Math 7410 Graph Theory

Bogdan Oporowski

Department of Mathematics
Louisiana State University

January 17, 2023

Definition of a graph

Definition 1.1

A graph G is a triple $(V, E, \mathcal{I})$ where

Example 1.2

Definition of a graph

Definition 1.1

A graph G is a triple $(V, E, \mathcal{I})$ where

- V (or $V(G)$) is a finite set whose elements are called vertices;

Example 1.2

- $V = \{v_1, v_2, v_3, v_4\}$

Definition of a graph

Definition 1.1

A graph G is a triple $(V, E, \mathcal{I})$ where

- V (or $V(G)$) is a finite set whose elements are called vertices;
- E (or $E(G)$) is a finite set disjoint from V whose elements are called edges; and

Example 1.2

- $V = \{v_1, v_2, v_3, v_4\}$
- $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$

Definition of a graph

Definition 1.1

A graph G is a triple $(V, E, \mathcal{I})$ where

- V (or $V(G)$) is a finite set whose elements are called vertices;
- E (or $E(G)$) is a finite set disjoint from V whose elements are called edges; and
- $\mathcal{I}$, called the incidence relation, is a subset of $V \times E$ in which each edge is in relation with exactly one or two vertices.

Example 1.2

- $V = \{v_1, v_2, v_3, v_4\}$
- $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$
- $\mathcal{I} = \{(v_1, e_1), (v_1, e_4), (v_1, e_5), (v_1, e_6), (v_2, e_1), (v_2, e_2), (v_3, e_2), (v_3, e_3), (v_3, e_5), (v_3, e_6), (v_4, e_3), (v_4, e_4), (v_4, e_7)\}$

Definition of a graph

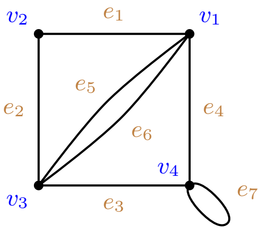
Definition 1.1

A graph G is a triple $(V, E, \mathcal{I})$ where

- V (or $V(G)$) is a finite set whose elements are called vertices;
- E (or $E(G)$) is a finite set disjoint from V whose elements are called edges; and
- $\mathcal{I}$, called the incidence relation, is a subset of $V \times E$ in which each edge is in relation with exactly one or two vertices.

Example 1.2

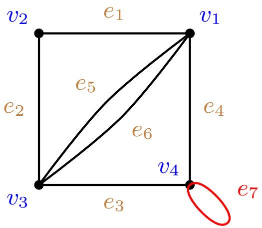
- $V = \{v_1, v_2, v_3, v_4\}$
- $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$
- $\mathcal{I} = \{(v_1, e_1), (v_1, e_4), (v_1, e_5), (v_1, e_6), (v_2, e_1), (v_2, e_2), (v_3, e_2), (v_3, e_3), (v_3, e_5), (v_3, e_6), (v_4, e_3), (v_4, e_4), (v_4, e_7)\}$



Simple graphs

Definition 1.3

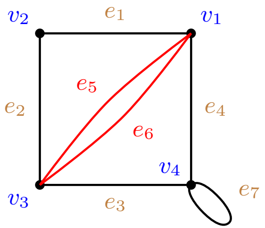
- Edges incident with just one vertex are loops.



Simple graphs

Definition 1.3

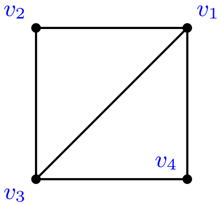
- Edges incident with just one vertex are loops.
- Edges incident with the same pair of vertices are parallel.



Simple graphs

Definition 1.3

- Edges incident with just one vertex are loops.
- Edges incident with the same pair of vertices are parallel.
- Graphs with no parallel edges and no loops are called simple.

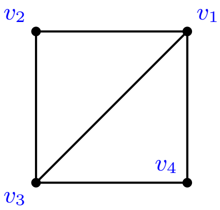


The simplification of the original graph.

Simple graphs

Definition 1.3

- Edges incident with just one vertex are loops.
- Edges incident with the same pair of vertices are parallel.
- Graphs with no parallel edges and no loops are called simple.



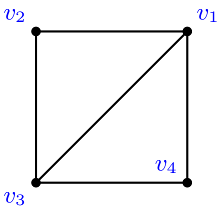
The simplification of the original graph.

Edges of a simple graph can be described as two-element subsets of the vertex set.

Simple graphs

Definition 1.3

- Edges incident with just one vertex are loops.
- Edges incident with the same pair of vertices are parallel.
- Graphs with no parallel edges and no loops are called simple.



The simplification of the original graph.

Edges of a simple graph can be described as two-element subsets of the vertex set.

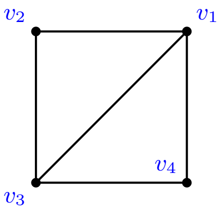
Example 1.4

$E = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\},$
$\{v_1, v_4\}, \{v_1, v_3\}\}.$

Simple graphs

Definition 1.3

- Edges incident with just one vertex are loops.
- Edges incident with the same pair of vertices are parallel.
- Graphs with no parallel edges and no loops are called simple.



The simplification of the original graph.

Edges of a simple graph can be described as two-element subsets of the vertex set.

Example 1.4

$E = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_1, v_4\}, \{v_1, v_3\}\}.$

Note 1.5

In some books, what we defined as a graph is called a multigraph and what we defined as a simple graph is called a graph.

Graph Terminology

Definition 1.6

- The graph G is empty if $V = \emptyset$, and is trivial if $E = \emptyset$.

Graph Terminology

Definition 1.6

- The graph G is empty if $V = \emptyset$, and is trivial if $E = \emptyset$.
- The cardinality of the vertex-set of a graph G is called the order of G and denoted $|G|$.

Graph Terminology

Definition 1.6

- The graph G is empty if $V = \emptyset$, and is trivial if $E = \emptyset$.
- The cardinality of the vertex-set of a graph G is called the order of G and denoted $|G|$.
- The cardinality of the edge-set of a graph G is called the size of G and denoted $\|G\|$.

Graph Terminology

Definition 1.6

- The graph G is empty if $V = \emptyset$, and is trivial if $E = \emptyset$.
- The cardinality of the vertex-set of a graph G is called the order of G and denoted $|G|$.
- The cardinality of the edge-set of a graph G is called the size of G and denoted $\|G\|$.
- Two vertices incident with the same edge are adjacent or neighbors.

Graph Terminology

Definition 1.6

- The graph G is empty if $V = \emptyset$, and is trivial if $E = \emptyset$.
- The cardinality of the vertex-set of a graph G is called the order of G and denoted $|G|$.
- The cardinality of the edge-set of a graph G is called the size of G and denoted $\|G\|$.
- Two vertices incident with the same edge are adjacent or neighbors.
- Similarly, two edges incident with the same vertex are adjacent.

Graph Terminology

Definition 1.6

- The graph G is empty if $V = \emptyset$, and is trivial if $E = \emptyset$.
- The cardinality of the vertex-set of a graph G is called the order of G and denoted $|G|$.
- The cardinality of the edge-set of a graph G is called the size of G and denoted $\|G\|$.
- Two vertices incident with the same edge are adjacent or neighbors.
- Similarly, two edges incident with the same vertex are adjacent.
- The number of edges incident with a vertex v of G, with loops counted twice, is the degree of v and is written as $d(v)$ or $d_G(v)$.

Graph Terminology

Definition 1.6

- The graph G is empty if $V = \emptyset$, and is trivial if $E = \emptyset$.
- The cardinality of the vertex-set of a graph G is called the order of G and denoted $|G|$.
- The cardinality of the edge-set of a graph G is called the size of G and denoted $\|G\|$.
- Two vertices incident with the same edge are adjacent or neighbors.
- Similarly, two edges incident with the same vertex are adjacent.
- The number of edges incident with a vertex v of G, with loops counted twice, is the degree of v and is written as $d(v)$ or $d_G(v)$.
- The set of neighbors of a vertex v of G, other than v itself, is denoted by $N(v)$ or by $N_G(v)$.

Graph Terminology

Definition 1.6

- The graph G is empty if $V = \emptyset$, and is trivial if $E = \emptyset$.
- The cardinality of the vertex-set of a graph G is called the order of G and denoted $|G|$.
- The cardinality of the edge-set of a graph G is called the size of G and denoted $\|G\|$.
- Two vertices incident with the same edge are adjacent or neighbors.
- Similarly, two edges incident with the same vertex are adjacent.
- The number of edges incident with a vertex v of G, with loops counted twice, is the degree of v and is written as $d(v)$ or $d_G(v)$.
- The set of neighbors of a vertex v of G, other than v itself, is denoted by $N(v)$ or by $N_G(v)$.
- Similarly, if U is a subset of the vertex set of G, then $N(U)$ is the set of those vertices that are not in U, but are adjacent to a vertex in U.

Isomorphism

Definition 1.7

The graphs $G_1 = (V_1, E_1, \mathcal{I}_1)$ and $G_2 = (V_2, E_2, \mathcal{I}_2)$ are isomorphic, written $G_1 \cong G_2$, if there are bijections $\varphi : V_1 \to V_2$ and $\psi : E_1 \to E_2$ such that $(v, e) \in \mathcal{I}_1$ if and only if $(\varphi(v), \psi(e)) \in \mathcal{I}_2$. Such a pair of bijections is an isomorphism.

Note 1.8

Isomorphism

Definition 1.7

The graphs $G_1 = (V_1, E_1, \mathcal{I}_1)$ and $G_2 = (V_2, E_2, \mathcal{I}_2)$ are isomorphic, written $G_1 \cong G_2$, if there are bijections $\varphi : V_1 \to V_2$ and $\psi : E_1 \to E_2$ such that $(v, e) \in \mathcal{I}_1$ if and only if $(\varphi(v), \psi(e)) \in \mathcal{I}_2$. Such a pair of bijections is an isomorphism.

Note 1.8

- *If G_1 and G_2 are simple, then an isomorphism may be defined as a bijection $\varphi : V_1 \to V_2$ such that u and v are adjacent in G_1 if and only if $\varphi(u)$ and $\varphi(v)$ are adjacent in G_2.*

Isomorphism

Definition 1.7

The graphs $G_1 = (V_1, E_1, \mathcal{I}_1)$ and $G_2 = (V_2, E_2, \mathcal{I}_2)$ are isomorphic, written $G_1 \cong G_2$, if there are bijections $\varphi : V_1 \to V_2$ and $\psi : E_1 \to E_2$ such that $(v, e) \in \mathcal{I}_1$ if and only if $(\varphi(v), \psi(e)) \in \mathcal{I}_2$. Such a pair of bijections is an isomorphism.

Note 1.8

- *If G_1 and G_2 are simple, then an isomorphism may be defined as a bijection $\varphi : V_1 \to V_2$ such that u and v are adjacent in G_1 if and only if $\varphi(u)$ and $\varphi(v)$ are adjacent in G_2.*
- *Isomorphic graphs are usually considered "the same".*

Isomorphism

Definition 1.7

The graphs $G_1 = (V_1, E_1, \mathcal{I}_1)$ and $G_2 = (V_2, E_2, \mathcal{I}_2)$ are isomorphic, written $G_1 \cong G_2$, if there are bijections $\varphi : V_1 \to V_2$ and $\psi : E_1 \to E_2$ such that $(v, e) \in \mathcal{I}_1$ if and only if $(\varphi(v), \psi(e)) \in \mathcal{I}_2$. Such a pair of bijections is an isomorphism.

Note 1.8

- *If G_1 and G_2 are simple, then an isomorphism may be defined as a bijection $\varphi : V_1 \to V_2$ such that u and v are adjacent in G_1 if and only if $\varphi(u)$ and $\varphi(v)$ are adjacent in G_2.*
- *Isomorphic graphs are usually considered "the same".*

Theorem 1.9 (Babai, 2015–2016)

Graph isomorphism problem can be solved in quasi-polynomial time.

Isomorphism

Definition 1.7

The graphs $G_1 = (V_1, E_1, \mathcal{I}_1)$ and $G_2 = (V_2, E_2, \mathcal{I}_2)$ are isomorphic, written $G_1 \cong G_2$, if there are bijections $\varphi : V_1 \to V_2$ and $\psi : E_1 \to E_2$ such that $(v, e) \in \mathcal{I}_1$ if and only if $(\varphi(v), \psi(e)) \in \mathcal{I}_2$. Such a pair of bijections is an isomorphism.

Note 1.8

- *If G_1 and G_2 are simple, then an isomorphism may be defined as a bijection $\varphi : V_1 \to V_2$ such that u and v are adjacent in G_1 if and only if $\varphi(u)$ and $\varphi(v)$ are adjacent in G_2.*
- *Isomorphic graphs are usually considered "the same".*

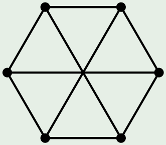
Theorem 1.9 (Babai, 2015–2016)

Graph isomorphism problem can be solved in quasi-polynomial time.
There is a constant c and an algorithm that can decide whether two graphs on n vertices are isomorphic or not in at most $2^{\mathcal{O}((\log n)^c)}$ steps.

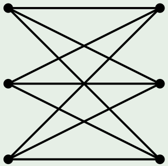
Isomorphism Example

Example 1.10

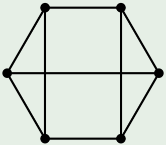
Which of the following graphs are isomorphic?



G_1



G_2

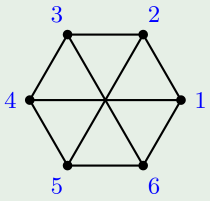


G_3

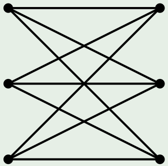
Isomorphism Example

Example 1.10

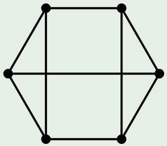
Which of the following graphs are isomorphic?



G_1



G_2

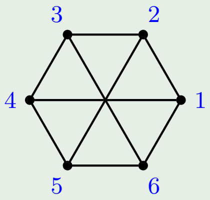


G_3

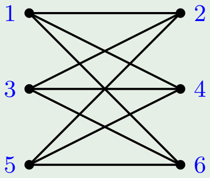
Isomorphism Example

Example 1.10

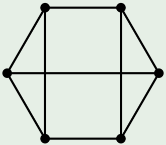
Which of the following graphs are isomorphic?



G_1



G_2

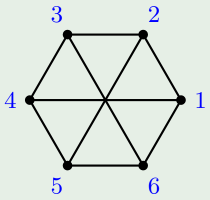


G_3

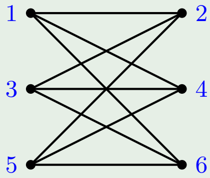
Isomorphism Example

Example 1.10

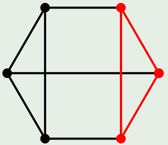
Which of the following graphs are isomorphic?



G_1



G_2

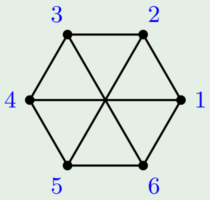


G_3

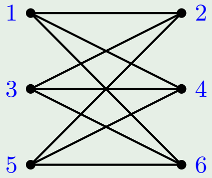
Isomorphism Example

Example 1.10

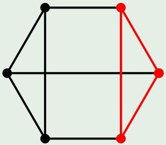
Which of the following graphs are isomorphic?



G_1



G_2



G_3

$$G_1 \cong G_2 \ncong G_3$$

Automorphism

Definition 1.11

An automorphism of a graph is an isomorphism from the graph to itself.

Automorphism

Definition 1.11

An automorphism of a graph is an isomorphism from the graph to itself.

Note 1.12

- *The automorphisms of a graph form a group.*

Automorphism

Definition 1.11

An automorphism of a graph is an isomorphism from the graph to itself.

Note 1.12

- *The automorphisms of a graph form a group.*
- *Computer software for finding automorphism groups of graphs is a part of the Sage system, available at* `http://sagemath.org`.

Automorphism

Definition 1.11

An automorphism of a graph is an isomorphism from the graph to itself.

Note 1.12

- *The automorphisms of a graph form a group.*
- *Computer software for finding automorphism groups of graphs is a part of the Sage system, available at* `http://sagemath.org`.

Theorem 1.13 (Frucht, 1938)

For every finite group X there is a graph whose automorphism group is X.

Automorphism

Definition 1.11

An automorphism of a graph is an isomorphism from the graph to itself.

Note 1.12

- *The automorphisms of a graph form a group.*
- *Computer software for finding automorphism groups of graphs is a part of the Sage system, available at* `http://sagemath.org`.

Theorem 1.13 (Frucht, 1938)

For every finite group X there is a graph whose automorphism group is X.

Problem 1

For every positive integer n, construct a simple graph with exactly n automorphisms.

Subgraphs

Definition 1.14

A graph $G_1 = (V_1, E_1, \mathfrak{I}_1)$ is a subgraph of a graph $G_2 = (V_2, E_2, \mathfrak{I}_2)$, written $G_1 \leqslant_{\mathrm{s}} G_2$, if

Subgraphs

Definition 1.14

A graph $G_1 = (V_1, E_1, \mathcal{I}_1)$ is a subgraph of a graph $G_2 = (V_2, E_2, \mathcal{I}_2)$, written $G_1 \leqslant_{\mathsf{s}} G_2$, if

- $V_1 \subseteq V_2$,

Subgraphs

Definition 1.14

A graph $G_1 = (V_1, E_1, \mathcal{I}_1)$ is a subgraph of a graph $G_2 = (V_2, E_2, \mathcal{I}_2)$, written $G_1 \leqslant_{\mathsf{s}} G_2$, if

- $V_1 \subseteq V_2$,
- $E_1 \subseteq E_2$, and

Subgraphs

Definition 1.14

A graph $G_1 = (V_1, E_1, \mathcal{I}_1)$ is a subgraph of a graph $G_2 = (V_2, E_2, \mathcal{I}_2)$, written $G_1 \leqslant_{\mathsf{s}} G_2$, if

- $V_1 \subseteq V_2$,
- $E_1 \subseteq E_2$, and
- $\mathcal{I}_1$ is induced by $\mathcal{I}_2$.

Subgraphs

Definition 1.14

A graph $G_1 = (V_1, E_1, \mathcal{I}_1)$ is a subgraph of a graph $G_2 = (V_2, E_2, \mathcal{I}_2)$, written $G_1 \leqslant_{\mathsf{s}} G_2$, if

- $V_1 \subseteq V_2$,
- $E_1 \subseteq E_2$, and
- $\mathcal{I}_1$ is induced by $\mathcal{I}_2$.

Alternately, we may think of G_1 as obtained from G_2 by

Subgraphs

Definition 1.14

A graph $G_1 = (V_1, E_1, \mathcal{I}_1)$ is a subgraph of a graph $G_2 = (V_2, E_2, \mathcal{I}_2)$, written $G_1 \leqslant_{\mathsf{s}} G_2$, if

- $V_1 \subseteq V_2$,
- $E_1 \subseteq E_2$, and
- $\mathcal{I}_1$ is induced by $\mathcal{I}_2$.

Alternately, we may think of G_1 as obtained from G_2 by

- Deleting vertices (denoted $G - v$ or $G - U$), and

Subgraphs

Definition 1.14

A graph $G_1 = (V_1, E_1, \mathcal{I}_1)$ is a subgraph of a graph $G_2 = (V_2, E_2, \mathcal{I}_2)$, written $G_1 \leqslant_{\mathsf{s}} G_2$, if

- $V_1 \subseteq V_2$,
- $E_1 \subseteq E_2$, and
- $\mathcal{I}_1$ is induced by $\mathcal{I}_2$.

Alternately, we may think of G_1 as obtained from G_2 by

- Deleting vertices (denoted $G - v$ or $G - U$), and
- Deleting edges (denoted $G \setminus e$ or $G \setminus F$).

Subgraphs

Definition 1.14

A graph $G_1 = (V_1, E_1, \mathcal{I}_1)$ is a subgraph of a graph $G_2 = (V_2, E_2, \mathcal{I}_2)$, written $G_1 \leqslant_s G_2$, if

- $V_1 \subseteq V_2$,
- $E_1 \subseteq E_2$, and
- $\mathcal{I}_1$ is induced by $\mathcal{I}_2$.

Alternately, we may think of G_1 as obtained from G_2 by

- Deleting vertices (denoted $G - v$ or $G - U$), and
- Deleting edges (denoted $G \setminus e$ or $G \setminus F$).

Definition 1.15

G_1 is an induced subgraph of G_2 if E_1 consists of all those elements of E_2 whose incident vertices lie in V_1.

Subgraphs

Definition 1.14

A graph $G_1 = (V_1, E_1, \mathcal{I}_1)$ is a subgraph of a graph $G_2 = (V_2, E_2, \mathcal{I}_2)$, written $G_1 \leqslant_{\mathrm{s}} G_2$, if

- $V_1 \subseteq V_2$,
- $E_1 \subseteq E_2$, and
- $\mathcal{I}_1$ is induced by $\mathcal{I}_2$.

Alternately, we may think of G_1 as obtained from G_2 by

- Deleting vertices (denoted $G - v$ or $G - U$), and
- Deleting edges (denoted $G \setminus e$ or $G \setminus F$).

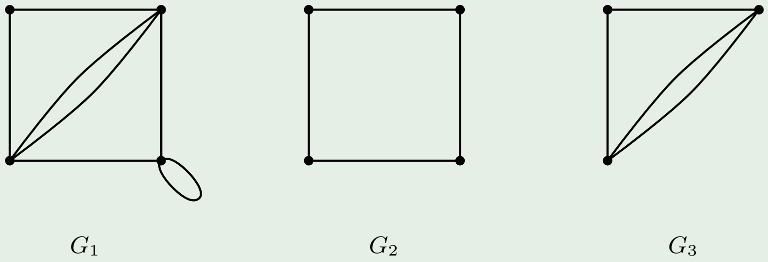
Definition 1.15

G_1 is an induced subgraph of G_2 if E_1 consists of all those elements of E_2 whose incident vertices lie in V_1.

Altrenately, we may think of G_1 as obtained from G_2 by deleting only vertices.

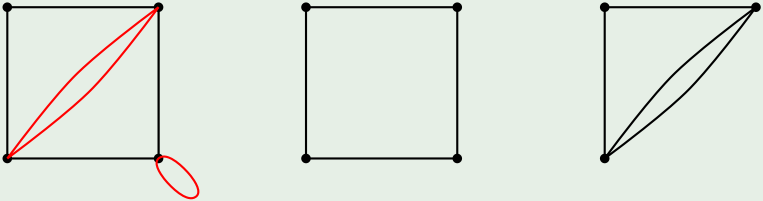
Subgraph Example

Example 1.16



Subgraph Example

Example 1.16

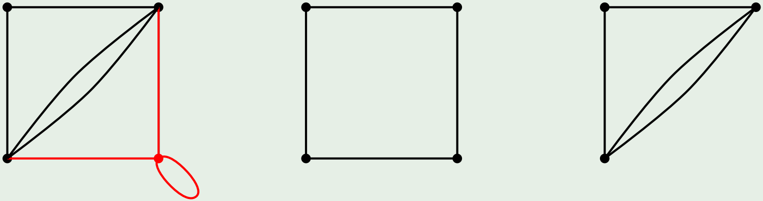


G_1 G_2 G_3

- G_2 is a subgraph of G_1, but it is not an induced subgraph.

Subgraph Example

Example 1.16



G_1 G_2 G_3

- G_2 is a subgraph of G_1, but it is not an induced subgraph.
- G_3 is an induced subgraph of G_1.

Reconstruction Conjectures

The deck of a graph G is the collection of graphs $G - v$ over all $v \in V(G)$. A graph is reconstructible if no other graph (up to isomorphism) has the same deck.

The deck of a graph G is the collection of graphs $G - v$ over all $v \in V(G)$. A graph is reconstructible if no other graph (up to isomorphism) has the same deck.

Conjecture 1.17 (Reconstruction Conjecture)

Every simple graph on at least three vertices is reconstructible.

The deck of a graph G is the collection of graphs $G - v$ over all $v \in V(G)$. A graph is reconstructible if no other graph (up to isomorphism) has the same deck.

Conjecture 1.17 (Reconstruction Conjecture)

Every simple graph on at least three vertices is reconstructible.

The edge-deck of a graph G is the collection of graphs $G \setminus e$ over all $e \in E(G)$. A graph is edge-reconstructible is no other graph has the same edge-deck.

Reconstruction Conjectures

The deck of a graph G is the collection of graphs $G - v$ over all $v \in V(G)$. A graph is reconstructible if no other graph (up to isomorphism) has the same deck.

Conjecture 1.17 (Reconstruction Conjecture)

Every simple graph on at least three vertices is reconstructible.

The edge-deck of a graph G is the collection of graphs $G \setminus e$ over all $e \in E(G)$. A graph is edge-reconstructible is no other graph has the same edge-deck.

Conjecture 1.18 (Edge-Reconstruction Conjecture)

Every simple graph on at least four edges is edge-reconstructible.

Walks, Trails, Paths, and Cycles

Definition 1.19

- A walk is a sequence v_0, e_1, v_1, e_2, v_2, ..., e_n, v_n, where each edge e_i is incident with vertices v_{i-1} and v_i.

Walks, Trails, Paths, and Cycles

Definition 1.19

- A walk is a sequence v_0, e_1, v_1, e_2, v_2, ..., e_n, v_n, where each edge e_i is incident with vertices v_{i-1} and v_i.
- The length of a walk is the number of edges in it.

Walks, Trails, Paths, and Cycles

Definition 1.19

- A walk is a sequence v_0, e_1, v_1, e_2, v_2, $\ldots$, e_n, v_n, where each edge e_i is incident with vertices v_{i-1} and v_i.
- The length of a walk is the number of edges in it.
- A walk is closed if its first and last vertices coincide.

Walks, Trails, Paths, and Cycles

Definition 1.19

- A walk is a sequence v_0, e_1, v_1, e_2, v_2, $\ldots$, e_n, v_n, where each edge e_i is incident with vertices v_{i-1} and v_i.
- The length of a walk is the number of edges in it.
- A walk is closed if its first and last vertices coincide.
- A trail is a walk in which no edge is repeated.

Walks, Trails, Paths, and Cycles

Definition 1.19

- A walk is a sequence v_0, e_1, v_1, e_2, v_2, ..., e_n, v_n, where each edge e_i is incident with vertices v_{i-1} and v_i.
- The length of a walk is the number of edges in it.
- A walk is closed if its first and last vertices coincide.
- A trail is a walk in which no edge is repeated.
- A path is a trail with no repeated vertices.

Walks, Trails, Paths, and Cycles

Definition 1.19

- A walk is a sequence v_0, e_1, v_1, e_2, v_2, ..., e_n, v_n, where each edge e_i is incident with vertices v_{i-1} and v_i.
- The length of a walk is the number of edges in it.
- A walk is closed if its first and last vertices coincide.
- A trail is a walk in which no edge is repeated.
- A path is a trail with no repeated vertices.
- A cycle is a trail with no vertices repeated except that the first vertex is the same as the last.

Walks, Trails, Paths, and Cycles

Definition 1.19

- A walk is a sequence $v_0, e_1, v_1, e_2, v_2, \ldots, e_n, v_n$, where each edge e_i is incident with vertices v_{i-1} and v_i.
- The length of a walk is the number of edges in it.
- A walk is closed if its first and last vertices coincide.
- A trail is a walk in which no edge is repeated.
- A path is a trail with no repeated vertices.
- A cycle is a trail with no vertices repeated except that the first vertex is the same as the last.
- For a path or a cycle, we will often blur the distinction between the sequence of vertices and edges, and the graph it forms.

Walks, Trails, Paths, and Cycles

Definition 1.19

- A walk is a sequence $v_0, e_1, v_1, e_2, v_2, \ldots, e_n, v_n$, where each edge e_i is incident with vertices v_{i-1} and v_i.
- The length of a walk is the number of edges in it.
- A walk is closed if its first and last vertices coincide.
- A trail is a walk in which no edge is repeated.
- A path is a trail with no repeated vertices.
- A cycle is a trail with no vertices repeated except that the first vertex is the same as the last.
- For a path or a cycle, we will often blur the distinction between the sequence of vertices and edges, and the graph it forms.
- The graph that is a path on n vertices (which has length $n-1$) will be denoted as P_n.

Walks, Trails, Paths, and Cycles

Definition 1.19

- A walk is a sequence $v_0, e_1, v_1, e_2, v_2, \ldots, e_n, v_n$, where each edge e_i is incident with vertices v_{i-1} and v_i.
- The length of a walk is the number of edges in it.
- A walk is closed if its first and last vertices coincide.
- A trail is a walk in which no edge is repeated.
- A path is a trail with no repeated vertices.
- A cycle is a trail with no vertices repeated except that the first vertex is the same as the last.
- For a path or a cycle, we will often blur the distinction between the sequence of vertices and edges, and the graph it forms.
- The graph that is a path on n vertices (which has length $n-1$) will be denoted as P_n.
- The graph that is a cycle on n vertices (which has length n) will be denoted as C_n.

Walks, Trails, Paths, and Cycles

Definition 1.19

- A walk is a sequence $v_0,\ e_1,\ v_1,\ e_2,\ v_2,\ \ldots,\ e_n,\ v_n$, where each edge e_i is incident with vertices v_{i-1} and v_i.
- The length of a walk is the number of edges in it.
- A walk is closed if its first and last vertices coincide.
- A trail is a walk in which no edge is repeated.
- A path is a trail with no repeated vertices.
- A cycle is a trail with no vertices repeated except that the first vertex is the same as the last.
- For a path or a cycle, we will often blur the distinction between the sequence of vertices and edges, and the graph it forms.
- The graph that is a path on n vertices (which has length $n-1$) will be denoted as P_n.
- The graph that is a cycle on n vertices (which has length n) will be denoted as C_n.
- A graph is connected if each pair of its vertices can be connected by a walk (equivalently, a trail or a path).

Complete Graphs and Complements

Definition 1.20

- A complete graph on n vertices, denoted by K_n, is a simple graph in which every two of its n vertices are connected by an edge.

Complete Graphs and Complements

Definition 1.20

- A complete graph on n vertices, denoted by K_n, is a simple graph in which every two of its n vertices are connected by an edge.
- If G is a simple graph, then the complement of G, denoted by $\overline{G}$, is the simple graph on the same vertex set as G, and in which two vertices are adjacent if and only if they are not adjacent in G.

Complete Graphs and Complements

Definition 1.20

- A complete graph on n vertices, denoted by K_n, is a simple graph in which every two of its n vertices are connected by an edge.
- If G is a simple graph, then the complement of G, denoted by $\overline{G}$, is the simple graph on the same vertex set as G, and in which two vertices are adjacent if and only if they are not adjacent in G.
- A simple graph is self-complementary if it is isomorphic to its own complement.

Complete Graphs and Complements

Definition 1.20

- A complete graph on n vertices, denoted by K_n, is a simple graph in which every two of its n vertices are connected by an edge.
- If G is a simple graph, then the complement of G, denoted by $\overline{G}$, is the simple graph on the same vertex set as G, and in which two vertices are adjacent if and only if they are not adjacent in G.
- A simple graph is self-complementary if it is isomorphic to its own complement.

Problem 2

Suppose that G is a non-trivial simple graph such that both G and $\overline{G}$ are connected. Prove that G has P_4 as an induced subgraph.

Complete Graphs and Complements

Definition 1.20

- A complete graph on n vertices, denoted by K_n, is a simple graph in which every two of its n vertices are connected by an edge.
- If G is a simple graph, then the complement of G, denoted by $\overline{G}$, is the simple graph on the same vertex set as G, and in which two vertices are adjacent if and only if they are not adjacent in G.
- A simple graph is self-complementary if it is isomorphic to its own complement.

Problem 2

Suppose that G is a non-trivial simple graph such that both G and $\overline{G}$ are connected. Prove that G has P_4 as an induced subgraph.

Problem 3

(a) *Show that the order of a self-complementary graph is congruent to 0 or 1 modulo 4.*

Complete Graphs and Complements

Definition 1.20

- A complete graph on n vertices, denoted by K_n, is a simple graph in which every two of its n vertices are connected by an edge.
- If G is a simple graph, then the complement of G, denoted by $\overline{G}$, is the simple graph on the same vertex set as G, and in which two vertices are adjacent if and only if they are not adjacent in G.
- A simple graph is self-complementary if it is isomorphic to its own complement.

Problem 2

Suppose that G is a non-trivial simple graph such that both G and $\overline{G}$ are connected. Prove that G has P_4 as an induced subgraph.

Problem 3

(a) *Show that the order of a self-complementary graph is congruent to 0 or 1 modulo 4.*

(b) *Construct a self-complementary graph of order n for every positive integer n congruent to 0 or 1 modulo 4.*

Hand-Shaking Lemma

Theorem 1.21 (Hand-Shaking Lemma)

$$\sum_{v \in V(G)} d(v) = 2\|G\|$$

Corollary 1.22

The number of vertices of odd degree is even.

Trees

Definition 2.1

- A graph having no cycles is acyclic or a forest.

Trees

Definition 2.1

- A graph having no cycles is acyclic or a forest.
- A connected forest is a tree.

Trees

Definition 2.1

- A graph having no cycles is acyclic or a forest.
- A connected forest is a tree.
- A leaf or a pendant vertex is a vertex of degree one.

Trees

Definition 2.1

- A graph having no cycles is acyclic or a forest.
- A connected forest is a tree.
- A leaf or a pendant vertex is a vertex of degree one.
- A subgraph of G is spanning if it has all the vertices of G.

Definition 2.1

- A graph having no cycles is acyclic or a forest.
- A connected forest is a tree.
- A leaf or a pendant vertex is a vertex of degree one.
- A subgraph of G is spanning if it has all the vertices of G.
- The distance between vertices u and v of G, written $d(u, v)$ or $d_G(u, v)$, is the length of the shortest path in G that contains both u and v. (Such a path is called a uv-path and u and v are its ends.)

Trees

Definition 2.1

- A graph having no cycles is acyclic or a forest.
- A connected forest is a tree.
- A leaf or a pendant vertex is a vertex of degree one.
- A subgraph of G is spanning if it has all the vertices of G.
- The distance between vertices u and v of G, written $d(u, v)$ or $d_G(u, v)$, is the length of the shortest path in G that contains both u and v. (Such a path is called a uv-path and u and v are its ends.) If a uv-path does not exist, then $d(u, v) = \infty$.

Definition 2.1

- A graph having no cycles is acyclic or a forest.
- A connected forest is a tree.
- A leaf or a pendant vertex is a vertex of degree one.
- A subgraph of G is spanning if it has all the vertices of G.
- The distance between vertices u and v of G, written $d(u, v)$ or $d_G(u, v)$, is the length of the shortest path in G that contains both u and v. (Such a path is called a uv-path and u and v are its ends.) If a uv-path does not exist, then $d(u, v) = \infty$.
- The distance between sets U and W of vertices of G, written $d(U, W)$, is the length of a shortest uw-path where $u \in U$ and $w \in W$, or infinity if no such path exists.

Induction for Trees

Theorem 2.2

Every tree with at least two vertices has at least two leaves.

Induction for Trees

Theorem 2.2

Every tree with at least two vertices has at least two leaves. Deleting a leaf from a tree of order n produces a tree of order $n-1$.

Induction for Trees

Theorem 2.2

Every tree with at least two vertices has at least two leaves. Deleting a leaf from a tree of order n produces a tree of order $n-1$.

Proof.

In an acyclic graph, the ends of a maximal non-trivial path have degree one. $\square$

Induction for Trees

Theorem 2.2

Every tree with at least two vertices has at least two leaves. Deleting a leaf from a tree of order n produces a tree of order $n-1$.

Proof.

In an acyclic graph, the ends of a maximal non-trivial path have degree one. Let v be a leaf of a tree T and let $T' = T - v$.

$\square$

Induction for Trees

Theorem 2.2

Every tree with at least two vertices has at least two leaves. Deleting a leaf from a tree of order n produces a tree of order $n-1$.

Proof.

In an acyclic graph, the ends of a maximal non-trivial path have degree one.
Let v be a leaf of a tree T and let $T' = T - v$.
Then T' is acyclic.

□

Induction for Trees

Theorem 2.2

Every tree with at least two vertices has at least two leaves. Deleting a leaf from a tree of order n produces a tree of order $n-1$.

Proof.

In an acyclic graph, the ends of a maximal non-trivial path have degree one.
Let v be a leaf of a tree T and let $T' = T - v$.
Then T' is acyclic.
Suppose u and w are vertices of T'. Then, in T there is a uw-path P.

$\square$

Induction for Trees

Theorem 2.2

Every tree with at least two vertices has at least two leaves. Deleting a leaf from a tree of order n produces a tree of order $n-1$.

Proof.

In an acyclic graph, the ends of a maximal non-trivial path have degree one.
Let v be a leaf of a tree T and let $T' = T - v$.
Then T' is acyclic.
Suppose u and w are vertices of T'. Then, in T there is a uw-path P.
But P cannot contain v as $d_T(v) = 1$, and so it also lies in T'. $\square$

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) G *is connected and acyclic;*

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) G *is connected and acyclic;*

(B) G *is connected and has size* $n-1$*;*

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) *G is connected and acyclic;*

(B) *G is connected and has size $n-1$;*

(C) *G is acyclic and has size $n-1$; and*

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) *G is connected and acyclic;*

(B) *G is connected and has size $n-1$;*

(C) *G is acyclic and has size $n-1$; and*

(D) *For every two vertices u and v, the graph G contains exactly one uv-path.*

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) *G is connected and acyclic;*

(B) *G is connected and has size $n-1$;*

(C) *G is acyclic and has size $n-1$; and*

(D) *For every two vertices u and v, the graph G contains exactly one uv-path.*

A $\Rightarrow$ B.

Induction on n. Trivial for $n = 1$. $\square$

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) *G is connected and acyclic;*

(B) *G is connected and has size $n-1$;*

(C) *G is acyclic and has size $n-1$; and*

(D) *For every two vertices u and v, the graph G contains exactly one uv-path.*

A $\Rightarrow$ B.

Induction on n. Trivial for $n=1$. For the inductive step, let v be a leaf, which exists by 2.2, and consider $G-v$. $\square$

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) *G is connected and acyclic;*

(B) *G is connected and has size $n-1$;*

(C) *G is acyclic and has size $n-1$; and*

(D) *For every two vertices u and v, the graph G contains exactly one uv-path.*

A $\Rightarrow$ B.

Induction on n. Trivial for $n=1$. For the inductive step, let v be a leaf, which exists by 2.2, and consider $G-v$. By the induction hypothesis, $G-v$ has size $n-2$, so G has size $n-1$. □

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) *G is connected and acyclic;*

(B) *G is connected and has size $n-1$;*

(C) *G is acyclic and has size $n-1$; and*

(D) *For every two vertices u and v, the graph G contains exactly one uv-path.*

A $\Rightarrow$ B.

Induction on n. Trivial for $n=1$. For the inductive step, let v be a leaf, which exists by 2.2, and consider $G-v$. By the induction hypothesis, $G-v$ has size $n-2$, so G has size $n-1$. $\square$

B $\Rightarrow$ C.

Delete edges from G, one by one, until the graph has no cycles, and call the resulting connected and acyclic graph G'. $\square$

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) *G is connected and acyclic;*

(B) *G is connected and has size $n-1$;*

(C) *G is acyclic and has size $n-1$; and*

(D) *For every two vertices u and v, the graph G contains exactly one uv-path.*

A $\Rightarrow$ B.

Induction on n. Trivial for $n=1$. For the inductive step, let v be a leaf, which exists by 2.2, and consider $G-v$. By the induction hypothesis, $G-v$ has size $n-2$, so G has size $n-1$. $\square$

B $\Rightarrow$ C.

Delete edges from G, one by one, until the graph has no cycles, and call the resulting connected and acyclic graph G'. Then G' satisfies (A), and so also satisfies (B), and so has size $n-1$. $\square$

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) *G is connected and acyclic;*

(B) *G is connected and has size $n-1$;*

(C) *G is acyclic and has size $n-1$; and*

(D) *For every two vertices u and v, the graph G contains exactly one uv-path.*

A $\Rightarrow$ B.

Induction on n. Trivial for $n = 1$. For the inductive step, let v be a leaf, which exists by 2.2, and consider $G - v$. By the induction hypothesis, $G - v$ has size $n - 2$, so G has size $n - 1$. $\square$

B $\Rightarrow$ C.

Delete edges from G, one by one, until the graph has no cycles, and call the resulting connected and acyclic graph G'. Then G' satisfies (A), and so also satisfies (B), and so has size $n-1$. This implies that $G' = G$. $\square$

Characterization of Trees

Theorem 2.3

For a simple graph G of order n the following are equivalent:

(A) *G is connected and acyclic;*

(B) *G is connected and has size $n-1$;*

(C) *G is acyclic and has size $n-1$; and*

(D) *For every two vertices u and v, the graph G contains exactly one uv-path.*

A $\Rightarrow$ B.

Induction on n. Trivial for $n = 1$. For the inductive step, let v be a leaf, which exists by 2.2, and consider $G - v$. By the induction hypothesis, $G - v$ has size $n - 2$, so G has size $n - 1$. □

B $\Rightarrow$ C.

Delete edges from G, one by one, until the graph has no cycles, and call the resulting connected and acyclic graph G'. Then G' satisfies (A), and so also satisfies (B), and so has size $n - 1$. This implies that $G' = G$. □

The remainder of the proof is left as an exercise.

Edge Exchange

Theorem 2.4

If T and T' are two spanning trees of a connected graph G and $e \in E(T) \setminus E(T')$, then there is an edge $e' \in E(T') \setminus E(T)$ such that $T \setminus e \cup e'$ is a spanning tree of G.

Edge Exchange

Theorem 2.4

If T and T' are two spanning trees of a connected graph G and $e \in E(T) \setminus E(T')$, then there is an edge $e' \in E(T') \setminus E(T)$ such that $T \setminus e \cup e'$ is a spanning tree of G.

Proof.

Consider $T \setminus e$: it is disconnected with exactly two connected components (maximal connected subgraphs) S and S'. $\square$

Edge Exchange

Theorem 2.4

If T and T' are two spanning trees of a connected graph G and $e \in E(T) \setminus E(T')$, then there is an edge $e' \in E(T') \setminus E(T)$ such that $T \setminus e \cup e'$ is a spanning tree of G.

Proof.

Consider $T \setminus e$: it is disconnected with exactly two connected components (maximal connected subgraphs) S and S'. Since T' is connected, it must have an edge e' with one endpoint in each S and S'. $\square$

Edge Exchange

Theorem 2.4

If T and T' are two spanning trees of a connected graph G and $e \in E(T) \setminus E(T')$, then there is an edge $e' \in E(T') \setminus E(T)$ such that $T \setminus e \cup e'$ is a spanning tree of G.

Proof.

Consider $T \setminus e$: it is disconnected with exactly two connected components (maximal connected subgraphs) S and S'. Since T' is connected, it must have an edge e' with one endpoint in each S and S'. Clearly, $T \setminus e \cup e'$ is a spanning tree of G. $\square$

Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function.

Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$.

Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with* $V(T) = V(G)$ *and* $E(T) = \emptyset$.

Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with* $V(T) = V(G)$ *and* $E(T) = \emptyset$.
- *Order the edges of* G *so that their costs are non-decreasing.*

Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with $V(T) = V(G)$ and $E(T) = \emptyset$.*
- *Order the edges of G so that their costs are non-decreasing.*
- *Proceed with each edge of G, one by one, in the above order: if its joins two components of T, add it to T; otherwise do nothing.*

Minimum Cost Spanning Tree

Suppose G is a graph and $c: E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with* $V(T) = V(G)$ *and* $E(T) = \emptyset$.
- *Order the edges of* G *so that their costs are non-decreasing.*
- *Proceed with each edge of* G*, one by one, in the above order: if its joins two components of* T*, add it to* T*; otherwise do nothing.*

Example 2.6

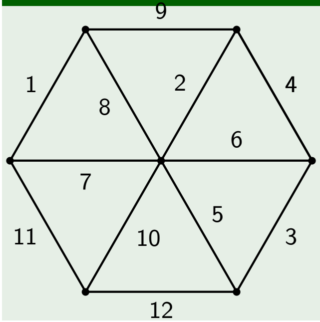
Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with* $V(T) = V(G)$ *and* $E(T) = \emptyset$.
- *Order the edges of* G *so that their costs are non-decreasing.*
- *Proceed with each edge of* G*, one by one, in the above order: if its joins two components of* T*, add it to* T*; otherwise do nothing.*

Example 2.6



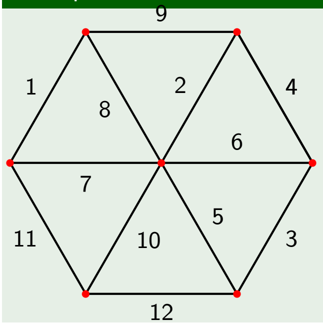
Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with $V(T) = V(G)$ and $E(T) = \emptyset$.*
- *Order the edges of G so that their costs are non-decreasing.*
- *Proceed with each edge of G, one by one, in the above order: if its joins two components of T, add it to T; otherwise do nothing.*

Example 2.6



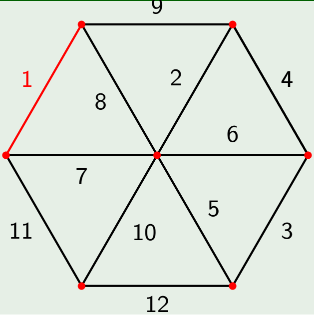
Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with* $V(T) = V(G)$ *and* $E(T) = \emptyset$.
- *Order the edges of* G *so that their costs are non-decreasing.*
- *Proceed with each edge of* G*, one by one, in the above order: if its joins two components of* T*, add it to* T*; otherwise do nothing.*

Example 2.6



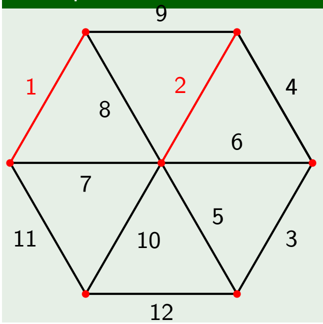
Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with $V(T) = V(G)$ and $E(T) = \emptyset$.*
- *Order the edges of G so that their costs are non-decreasing.*
- *Proceed with each edge of G, one by one, in the above order: if its joins two components of T, add it to T; otherwise do nothing.*

Example 2.6

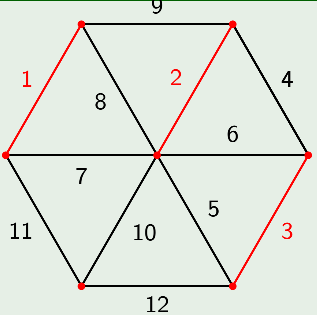


Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with* $V(T) = V(G)$ *and* $E(T) = \emptyset$.
- *Order the edges of* G *so that their costs are non-decreasing.*
- *Proceed with each edge of* G*, one by one, in the above order: if its joins two components of* T*, add it to* T*; otherwise do nothing.*

Example 2.6



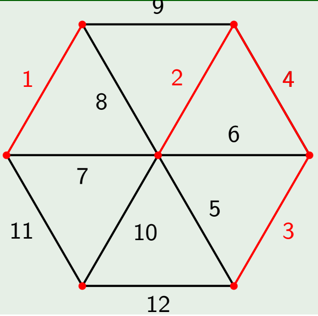
Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with $V(T) = V(G)$ and $E(T) = \emptyset$.*
- *Order the edges of G so that their costs are non-decreasing.*
- *Proceed with each edge of G, one by one, in the above order: if its joins two components of T, add it to T; otherwise do nothing.*

Example 2.6



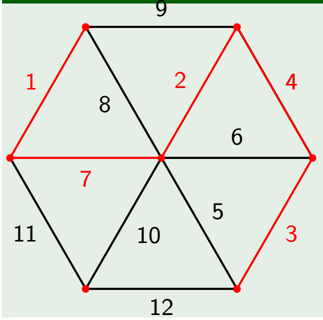
Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with* $V(T) = V(G)$ *and* $E(T) = \emptyset$.
- *Order the edges of* G *so that their costs are non-decreasing.*
- *Proceed with each edge of* G*, one by one, in the above order: if its joins two components of* T*, add it to* T*; otherwise do nothing.*

Example 2.6



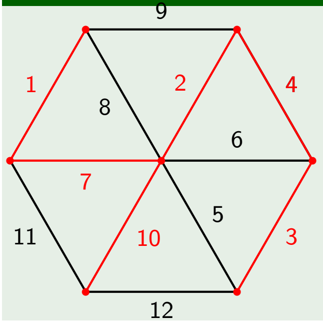
Minimum Cost Spanning Tree

Suppose G is a graph and $c : E(G) \to \mathbb{N}$ is a cost function. The cost of a subgraph H of G is $\sum_{e \in E(H)} c(e)$. We want to find a minimum-cost spanning tree T of G.

Algorithm 2.5 (Kruskal)

- *Start with* $V(T) = V(G)$ *and* $E(T) = \emptyset$.
- *Order the edges of* G *so that their costs are non-decreasing.*
- *Proceed with each edge of* G*, one by one, in the above order: if its joins two components of* T*, add it to* T*; otherwise do nothing.*

Example 2.6



Proof of Kruskal's Theorem

Theorem 2.7 (Kruskal)

In a connected graph, Kruskal's Algorithm produces a minimum-cost spanning tree.

Proof of Kruskal's Theorem

Theorem 2.7 (Kruskal)

In a connected graph, Kruskal's Algorithm produces a minimum-cost spanning tree.

Proof.

It is clear that the algorithm produces a spanning tree. □

Proof of Kruskal's Theorem

Theorem 2.7 (Kruskal)

In a connected graph, Kruskal's Algorithm produces a minimum-cost spanning tree.

Proof.

It is clear that the algorithm produces a spanning tree.
Let T be the resulting graph, and suppose T' is a spanning tree of minimum cost.

$\square$

Proof of Kruskal's Theorem

Theorem 2.7 (Kruskal)

In a connected graph, Kruskal's Algorithm produces a minimum-cost spanning tree.

Proof.

It is clear that the algorithm produces a spanning tree.
Let T be the resulting graph, and suppose T' is a spanning tree of minimum cost. If $T' = T$, then there is nothing to prove.

□

Proof of Kruskal's Theorem

Theorem 2.7 (Kruskal)

In a connected graph, Kruskal's Algorithm produces a minimum-cost spanning tree.

Proof.

It is clear that the algorithm produces a spanning tree.
Let T be the resulting graph, and suppose T' is a spanning tree of minimum cost. If $T' = T$, then there is nothing to prove. If $T \neq T'$, let e be the first edge chosen for T that is not in T'.

$\square$

Proof of Kruskal's Theorem

Theorem 2.7 (Kruskal)

In a connected graph, Kruskal's Algorithm produces a minimum-cost spanning tree.

Proof.

It is clear that the algorithm produces a spanning tree.
Let T be the resulting graph, and suppose T' is a spanning tree of minimum cost. If $T' = T$, then there is nothing to prove. If $T \neq T'$, let e be the first edge chosen for T that is not in T'. Adding e to T' creates a cycle C, but since T does not have cycles, T' has an edge $e' \notin E(T)$.

$\square$

Proof of Kruskal's Theorem

Theorem 2.7 (Kruskal)

In a connected graph, Kruskal's Algorithm produces a minimum-cost spanning tree.

Proof.

It is clear that the algorithm produces a spanning tree.
Let T be the resulting graph, and suppose T' is a spanning tree of minimum cost. If $T' = T$, then there is nothing to prove. If $T \neq T'$, let e be the first edge chosen for T that is not in T'. Adding e to T' creates a cycle C, but since T does not have cycles, T' has an edge $e' \notin E(T)$. Consider the spanning tree $T' \setminus e' \cup e$.

$\square$

Proof of Kruskal's Theorem

Theorem 2.7 (Kruskal)

In a connected graph, Kruskal's Algorithm produces a minimum-cost spanning tree.

Proof.

It is clear that the algorithm produces a spanning tree.
Let T be the resulting graph, and suppose T' is a spanning tree of minimum cost. If $T' = T$, then there is nothing to prove. If $T \neq T'$, let e be the first edge chosen for T that is not in T'. Adding e to T' creates a cycle C, but since T does not have cycles, T' has an edge $e' \notin E(T)$. Consider the spanning tree $T' \setminus e' \cup e$.
Since T' contains e' and all edges of T chosen before e, both e and e' are available when the algorithm chooses e, and hence $c(e) \leq c(e')$. $\square$

Proof of Kruskal's Theorem

Theorem 2.7 (Kruskal)

In a connected graph, Kruskal's Algorithm produces a minimum-cost spanning tree.

Proof.

It is clear that the algorithm produces a spanning tree.
Let T be the resulting graph, and suppose T' is a spanning tree of minimum cost. If $T' = T$, then there is nothing to prove. If $T \neq T'$, let e be the first edge chosen for T that is not in T'. Adding e to T' creates a cycle C, but since T does not have cycles, T' has an edge $e' \notin E(T)$. Consider the spanning tree $T' \setminus e' \cup e$.
Since T' contains e' and all edges of T chosen before e, both e and e' are available when the algorithm chooses e, and hence $c(e) \leq c(e')$. Thus $T' \setminus e' \cup e$ is a spanning tree with cost at most T' that agrees with T for a longer initial list of edges than T' does. $\square$

Proof of Kruskal's Theorem

Theorem 2.7 (Kruskal)

In a connected graph, Kruskal's Algorithm produces a minimum-cost spanning tree.

Proof.

It is clear that the algorithm produces a spanning tree.
Let T be the resulting graph, and suppose T' is a spanning tree of minimum cost. If $T' = T$, then there is nothing to prove. If $T \neq T'$, let e be the first edge chosen for T that is not in T'. Adding e to T' creates a cycle C, but since T does not have cycles, T' has an edge $e' \notin E(T)$. Consider the spanning tree $T' \setminus e' \cup e$.
Since T' contains e' and all edges of T chosen before e, both e and e' are available when the algorithm chooses e, and hence $c(e) \leq c(e')$. Thus $T' \setminus e' \cup e$ is a spanning tree with cost at most T' that agrees with T for a longer initial list of edges than T' does. Repeating this argument yields a minimum-cost spanning tree that equals T, proving that the costs of T and T' are the same. □

Enumerating Labeled Trees

We would like to know how many different (and here we really mean different rather than non-isomorphic) trees with the vertex set $\{1, 2, \ldots, n\}$ are there?

Enumerating Labeled Trees

We would like to know how many different (and here we really mean different rather than non-isomorphic) trees with the vertex set $\{1, 2, \ldots, n\}$ are there?

Theorem 2.8 (Cayley's Formula)

There are n^{n-2} *trees with vertex set* $\{1, 2, \ldots, n\}$.

Enumerating Labeled Trees

We would like to know how many different (and here we really mean different rather than non-isomorphic) trees with the vertex set $\{1, 2, \ldots, n\}$ are there?

Theorem 2.8 (Cayley's Formula due to Borchardt (1860))

There are n^{n-2} trees with vertex set $\{1, 2, \ldots, n\}$.

We would like to know how many different (and here we really mean different rather than non-isomorphic) trees with the vertex set $\{1, 2, \ldots, n\}$ are there?

Theorem 2.8 (Cayley's Formula due to Borchardt (1860))

There are n^{n-2} *trees with vertex set* $\{1, 2, \ldots, n\}$.

Proof.

There are n^{n-2} sequences of length $n-2$ with entries from $\{1, 2, \ldots, n\}$. $\square$

Enumerating Labeled Trees

We would like to know how many different (and here we really mean different rather than non-isomorphic) trees with the vertex set $\{1, 2, \ldots, n\}$ are there?

Theorem 2.8 (Cayley's Formula due to Borchardt (1860))

There are n^{n-2} trees with vertex set $\{1, 2, \ldots, n\}$.

Proof.

There are n^{n-2} sequences of length $n-2$ with entries from $\{1, 2, \ldots, n\}$. We will establish a bijection between such sequences and trees on the vertex set $\{1, 2, \ldots, n\}$. $\square$

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

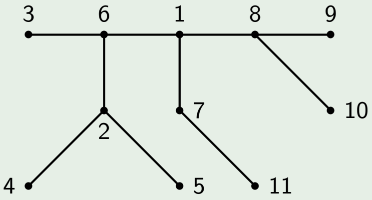
- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Example 2.9



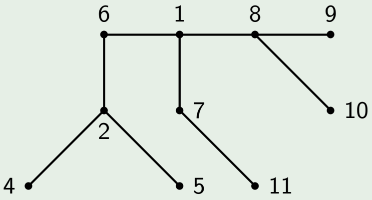
Prüfer sequence:

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Example 2.9



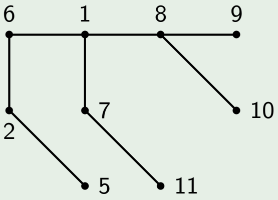
Prüfer sequence: 6

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Example 2.9



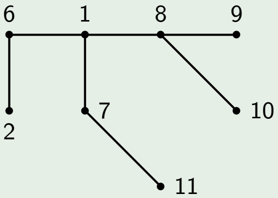
Prüfer sequence: 6 , 2

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Example 2.9



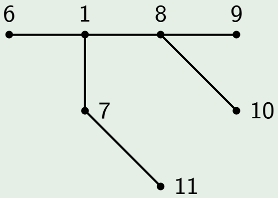
Prüfer sequence: 6 , 2 , 2

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Example 2.9



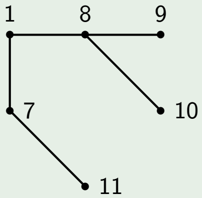
Prüfer sequence: 6 , 2 , 2 , 6

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Example 2.9



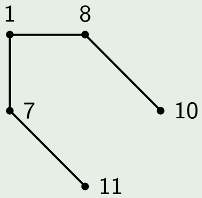
Prüfer sequence: 6 , 2 , 2 , 6 , 1

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Example 2.9



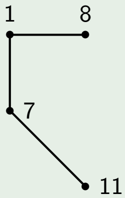
Prüfer sequence: 6 , 2 , 2 , 6 , 1 , 8

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Example 2.9



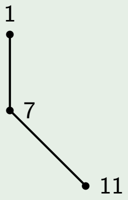
Prüfer sequence: 6 , 2 , 2 , 6 , 1 , 8 , 8

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Example 2.9



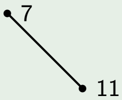
Prüfer sequence: 6 , 2 , 2 , 6 , 1 , 8 , 8 , 1

Prüfer Sequences

To find a Prüfer sequence $f(T)$ of a labeled tree T,

- delete the leaf with the smallest label, and
- append the label of its neighbor to the sequence until one edge remains.

Example 2.9



Prüfer sequence: 6 , 2 , 2 , 6 , 1 , 8 , 8 , 1 , 7

Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, $\ldots$, n.

Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, . . . , n.
- Proceed with all $n - 2$ elements of the sequence, and, at the ith step,

Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, . . . , n.
- Proceed with all $n - 2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.

Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, . . . , n.
- Proceed with all $n - 2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".

Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, ..., n.
- Proceed with all $n - 2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and

Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, ..., n.
- Proceed with all $n-2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.

Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, . . . , n.
- Proceed with all $n - 2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

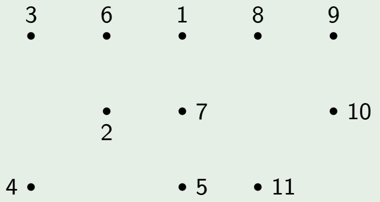
Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, . . . , n.
- Proceed with all $n-2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7 Finished:



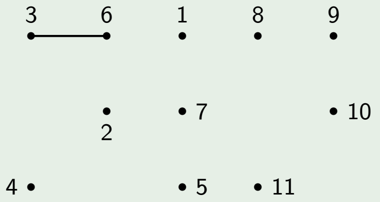
Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, . . . , n.
- Proceed with all $n - 2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7 Finished: 3



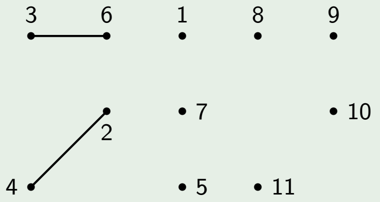
Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, . . . , n.
- Proceed with all $n-2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7 Finished: 3, 4



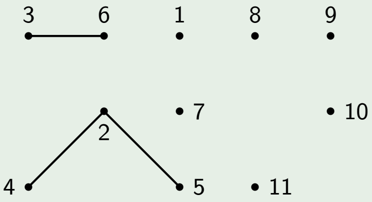
Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, ..., n.
- Proceed with all $n-2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7 Finished: 3, 4, 5



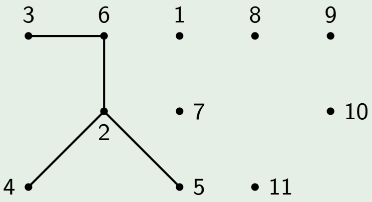
Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, ..., n.
- Proceed with all $n-2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7 Finished: 3, 4, 5, 2



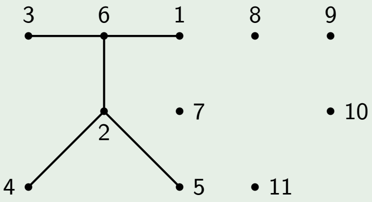
Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, . . . , n.
- Proceed with all $n - 2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7 Finished: 3, 4, 5, 2, 6



Trees from Sequences

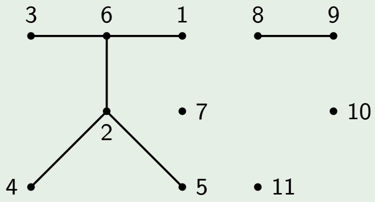
Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, ..., n.
- Proceed with all $n-2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7

Finished: 3, 4, 5, 2, 6, 9



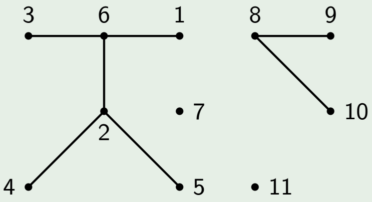
Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, $\ldots$, n.
- Proceed with all $n-2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7 Finished: 3, 4, 5, 2, 6, 9, 10



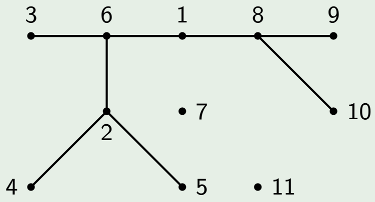
Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, $\ldots$, n.
- Proceed with all $n-2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7   Finished: 3, 4, 5, 2, 6, 9, 10, 8



Trees from Sequences

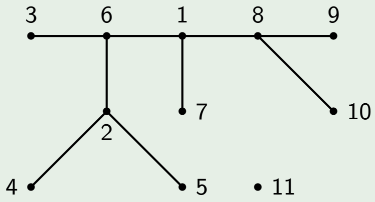
Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, ..., n.
- Proceed with all $n - 2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7

Finished: 3, 4, 5, 2, 6, 9, 10, 8, 1



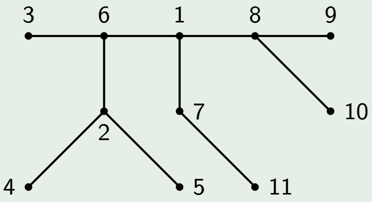
Trees from Sequences

Now we describe how to produce a tree from a Prüfer sequence.

- Begin with a forest having n isolated vertices labeled 1, 2, . . . , n.
- Proceed with all $n-2$ elements of the sequence, and, at the ith step,
  - let x be the label in position i.
  - let y be the smallest label that does not appear at the ith or later position and has not yet been marked as "finished".
  - add the edge xy, and
  - mark y as finished.
- Join the two remaining unfinished vertices with an edge.

Example 2.10

Sequence: 6, 2, 2, 6, 1, 8, 8, 1, 7 Finished: 3, 4, 5, 2, 6, 9, 10, 8, 1



Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.
First, we show that when we start with a sequence, we indeed produce a tree.

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.
First, we show that when we start with a sequence, we indeed produce a tree. Note that we start of the ith step with $n-i+1$ unfinished vertices and $n-i-1$ remaining vertices in the sequence.

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.

First, we show that when we start with a sequence, we indeed produce a tree. Note that we start of the ith step with $n - i + 1$ unfinished vertices and $n - i - 1$ remaining vertices in the sequence. Therefore y can be chosen as described, and the algorithm produces a graph of order n and size $n - 1$.

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.
First, we show that when we start with a sequence, we indeed produce a tree. Note that we start of the ith step with $n - i + 1$ unfinished vertices and $n - i - 1$ remaining vertices in the sequence. Therefore y can be chosen as described, and the algorithm produces a graph of order n and size $n - 1$. Each step joins two unfinished vertices and marks one of them as finished.

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.

First, we show that when we start with a sequence, we indeed produce a tree. Note that we start of the ith step with $n-i+1$ unfinished vertices and $n-i-1$ remaining vertices in the sequence. Therefore y can be chosen as described, and the algorithm produces a graph of order n and size $n-1$. Each step joins two unfinished vertices and marks one of them as finished. Thus after i steps the graph has $n-i$ components, each containing exactly one unfinished vertex.

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.
First, we show that when we start with a sequence, we indeed produce a tree. Note that we start of the ith step with $n - i + 1$ unfinished vertices and $n - i - 1$ remaining vertices in the sequence. Therefore y can be chosen as described, and the algorithm produces a graph of order n and size $n - 1$. Each step joins two unfinished vertices and marks one of them as finished. Thus after i steps the graph has $n - i$ components, each containing exactly one unfinished vertex. The final step connects the graph thereby creating a tree.

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.
First, we show that when we start with a sequence, we indeed produce a tree. Note that we start of the ith step with $n-i+1$ unfinished vertices and $n-i-1$ remaining vertices in the sequence. Therefore y can be chosen as described, and the algorithm produces a graph of order n and size $n-1$. Each step joins two unfinished vertices and marks one of them as finished. Thus after i steps the graph has $n-i$ components, each containing exactly one unfinished vertex. The final step connects the graph thereby creating a tree.
Now we need to show that the obtained tree is the same as the one that created the sequence.

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.
First, we show that when we start with a sequence, we indeed produce a tree. Note that we start of the ith step with $n - i + 1$ unfinished vertices and $n - i - 1$ remaining vertices in the sequence. Therefore y can be chosen as described, and the algorithm produces a graph of order n and size $n - 1$. Each step joins two unfinished vertices and marks one of them as finished. Thus after i steps the graph has $n - i$ components, each containing exactly one unfinished vertex. The final step connects the graph thereby creating a tree. Now we need to show that the obtained tree is the same as the one that created the sequence. In each step of computing the sequence, we can mark the deleted leaf as "finished".

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.
First, we show that when we start with a sequence, we indeed produce a tree. Note that we start of the ith step with $n - i + 1$ unfinished vertices and $n - i - 1$ remaining vertices in the sequence. Therefore y can be chosen as described, and the algorithm produces a graph of order n and size $n - 1$. Each step joins two unfinished vertices and marks one of them as finished. Thus after i steps the graph has $n - i$ components, each containing exactly one unfinished vertex. The final step connects the graph thereby creating a tree.
Now we need to show that the obtained tree is the same as the one that created the sequence. In each step of computing the sequence, we can mark the deleted leaf as "finished". The labels that do not yet appear in the remainder of the sequence we generate are the unfinished vertices that are not leaves.

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.
First, we show that when we start with a sequence, we indeed produce a tree. Note that we start of the ith step with $n-i+1$ unfinished vertices and $n-i-1$ remaining vertices in the sequence. Therefore y can be chosen as described, and the algorithm produces a graph of order n and size $n-1$. Each step joins two unfinished vertices and marks one of them as finished. Thus after i steps the graph has $n-i$ components, each containing exactly one unfinished vertex. The final step connects the graph thereby creating a tree.
Now we need to show that the obtained tree is the same as the one that created the sequence. In each step of computing the sequence, we can mark the deleted leaf as "finished". The labels that do not yet appear in the remainder of the sequence we generate are the unfinished vertices that are not leaves. Because the next leaf deleted is the least, the edge deleted in each stage of computing the sequence is precisely the edge added when constructing the graph.

Proof of Cayley's Formula

Now we show that the two operations described previously are inverses of each other.
First, we show that when we start with a sequence, we indeed produce a tree. Note that we start of the ith step with $n-i+1$ unfinished vertices and $n-i-1$ remaining vertices in the sequence. Therefore y can be chosen as described, and the algorithm produces a graph of order n and size $n-1$. Each step joins two unfinished vertices and marks one of them as finished. Thus after i steps the graph has $n-i$ components, each containing exactly one unfinished vertex. The final step connects the graph thereby creating a tree. Now we need to show that the obtained tree is the same as the one that created the sequence. In each step of computing the sequence, we can mark the deleted leaf as "finished". The labels that do not yet appear in the remainder of the sequence we generate are the unfinished vertices that are not leaves. Because the next leaf deleted is the least, the edge deleted in each stage of computing the sequence is precisely the edge added when constructing the graph. Therefore the correspondence between the sequences and labeled trees is a bijection.

Counting Trees with Prescribed Degrees

Corollary 2.11

The number of trees with vertex set $\{1, 2, \ldots, n\}$ *in which vertices* 1, 2, $\ldots$, n *have respective degrees* d_1, d_2, $\ldots$, d_n *is*

$$\frac{(n-2)!}{\prod (d_i - 1)!}.$$

Counting Trees with Prescribed Degrees

Corollary 2.11

The number of trees with vertex set $\{1, 2, \ldots, n\}$ *in which vertices* 1, 2, $\ldots$, n *have respective degrees* d_1, d_2, $\ldots$, d_n *is*

$$\frac{(n-2)!}{\prod(d_i - 1)!}.$$

Proof.

When we delete vertex x from T when constructing the Prüfer sequence, all neighbors of x except for one have already been deleted.

$\square$

Corollary 2.11

The number of trees with vertex set $\{1, 2, \ldots, n\}$ *in which vertices* 1, 2, $\ldots$, n *have respective degrees* d_1, d_2, $\ldots$, d_n *is*

$$\frac{(n-2)!}{\prod(d_i-1)!}.$$

Proof.

When we delete vertex x from T when constructing the Prüfer sequence, all neighbors of x except for one have already been deleted. We record x in the sequence once for each deleted neighbor and x does not appear in the sequence again.

$\square$

Counting Trees with Prescribed Degrees

Corollary 2.11

The number of trees with vertex set $\{1, 2, \ldots, n\}$ *in which vertices* 1, 2, $\ldots$, n *have respective degrees* d_1, d_2, $\ldots$, d_n *is*

$$\frac{(n-2)!}{\prod(d_i - 1)!}.$$

Proof.

When we delete vertex x from T when constructing the Prüfer sequence, all neighbors of x except for one have already been deleted. We record x in the sequence once for each deleted neighbor and x does not appear in the sequence again. Hence x appears in the sequence $d(x) - 1$ times.

$\square$

Counting Trees with Prescribed Degrees

Corollary 2.11

The number of trees with vertex set $\{1, 2, \dots, n\}$ *in which vertices* $1, 2, \dots, n$ *have respective degrees* $d_1, d_2, \dots, d_n$ *is*

$$\frac{(n-2)!}{\prod (d_i - 1)!}.$$

Proof.

When we delete vertex x from T when constructing the Prüfer sequence, all neighbors of x except for one have already been deleted. We record x in the sequence once for each deleted neighbor and x does not appear in the sequence again. Hence x appears in the sequence $d(x) - 1$ times.
Therefore we count the trees by counting sequences of length $n - 2$ having $d_i - 1$ copies of i, for each i. □

Counting Trees with Prescribed Degrees

Corollary 2.11

The number of trees with vertex set $\{1, 2, \ldots, n\}$ *in which vertices* $1, 2, \ldots, n$ *have respective degrees* $d_1, d_2, \ldots, d_n$ *is*

$$\frac{(n-2)!}{\prod(d_i - 1)!}.$$

Proof.

When we delete vertex x from T when constructing the Prüfer sequence, all neighbors of x except for one have already been deleted. We record x in the sequence once for each deleted neighbor and x does not appear in the sequence again. Hence x appears in the sequence $d(x) - 1$ times.

Therefore we count the trees by counting sequences of length $n - 2$ having $d_i - 1$ copies of i, for each i. If we distinguish between various copies of i, then there are $(n-2)!$ such sequences. $\square$

Counting Trees with Prescribed Degrees

Corollary 2.11

The number of trees with vertex set $\{1, 2, \ldots, n\}$ *in which vertices* 1, 2, $\ldots$, n *have respective degrees* d_1, d_2, $\ldots$, d_n *is*

$$\frac{(n-2)!}{\prod(d_i - 1)!}.$$

Proof.

When we delete vertex x from T when constructing the Prüfer sequence, all neighbors of x except for one have already been deleted. We record x in the sequence once for each deleted neighbor and x does not appear in the sequence again. Hence x appears in the sequence $d(x) - 1$ times.

Therefore we count the trees by counting sequences of length $n - 2$ having $d_i - 1$ copies of i, for each i. If we distinguish between various copies of i, then there are $(n-2)!$ such sequences. Since we really cannot distinguish between the copies, we have over-counted by a factor of $(d_i - 1)!$ for each i. □

Minors

Definition 2.12

- If e is an edge of G incident with two distinct vertices u and v, then the contraction of e is the operation of deleting e and identifying u and v.

Minors

Definition 2.12

- If e is an edge of G incident with two distinct vertices u and v, then the contraction of e is the operation of deleting e and identifying u and v.
- Contracting a loop is the same as deleting it.

Minors

Definition 2.12

- If e is an edge of G incident with two distinct vertices u and v, then the contraction of e is the operation of deleting e and identifying u and v.
- Contracting a loop is the same as deleting it.
- The graph obtained from G by contracting e is denoted G/e (extended to G/F if $F \subseteq E(G)$).

Minors

Definition 2.12

- If e is an edge of G incident with two distinct vertices u and v, then the contraction of e is the operation of deleting e and identifying u and v.
- Contracting a loop is the same as deleting it.
- The graph obtained from G by contracting e is denoted G/e (extended to G/F if $F \subseteq E(G)$).
- A graph H is a minor of G if it can be obtained from G by a sequence of operation each of which is one of the following:
  - deleting an edge;

Minors

Definition 2.12

- If e is an edge of G incident with two distinct vertices u and v, then the contraction of e is the operation of deleting e and identifying u and v.
- Contracting a loop is the same as deleting it.
- The graph obtained from G by contracting e is denoted G/e (extended to G/F if $F \subseteq E(G)$).
- A graph H is a minor of G if it can be obtained from G by a sequence of operation each of which is one of the following:
  - deleting an edge;
  - deleting an isolated vertex; and

Minors

Definition 2.12

- If e is an edge of G incident with two distinct vertices u and v, then the contraction of e is the operation of deleting e and identifying u and v.
- Contracting a loop is the same as deleting it.
- The graph obtained from G by contracting e is denoted G/e (extended to G/F if $F \subseteq E(G)$).
- A graph H is a minor of G if it can be obtained from G by a sequence of operation each of which is one of the following:
  - deleting an edge;
  - deleting an isolated vertex; and
  - contracting an edge.

Minors

Definition 2.12

- If e is an edge of G incident with two distinct vertices u and v, then the contraction of e is the operation of deleting e and identifying u and v.
- Contracting a loop is the same as deleting it.
- The graph obtained from G by contracting e is denoted G/e (extended to G/F if $F \subseteq E(G)$).
- A graph H is a minor of G if it can be obtained from G by a sequence of operation each of which is one of the following:
  - deleting an edge;
  - deleting an isolated vertex; and
  - contracting an edge.
- We write $H \leqslant_{\mathrm{m}} G$ to indicate that H is isomorphic to a minor of G.

Minors

Definition 2.12

- If e is an edge of G incident with two distinct vertices u and v, then the contraction of e is the operation of deleting e and identifying u and v.
- Contracting a loop is the same as deleting it.
- The graph obtained from G by contracting e is denoted G/e (extended to G/F if $F \subseteq E(G)$).
- A graph H is a minor of G if it can be obtained from G by a sequence of operation each of which is one of the following:
  - deleting an edge;
  - deleting an isolated vertex; and
  - contracting an edge.
- We write $H \leqslant_{\mathrm{m}} G$ to indicate that H is isomorphic to a minor of G.

Note 2.13

The order of operations of deleting and contracting to get a minor of a graph is irrelevant.

Counting Spanning Trees

Theorem 2.14

Let $\tau(G)$ denote the number of distinct spanning trees of a (labeled) graph G.

Counting Spanning Trees

Theorem 2.14

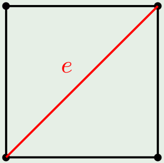
Let $\tau(G)$ denote the number of distinct spanning trees of a (labeled) graph G. If e is a non-loop edge of G, then $\tau(G) = \tau(G \setminus e) + \tau(G/e)$.

Counting Spanning Trees

Theorem 2.14

Let $\tau(G)$ denote the number of distinct spanning trees of a (labeled) graph G. If e is a non-loop edge of G, then $\tau(G) = \tau(G \setminus e) + \tau(G/e)$.

Example 2.15



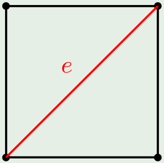
G

Counting Spanning Trees

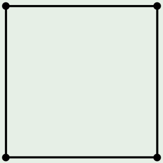
Theorem 2.14

Let $\tau(G)$ denote the number of distinct spanning trees of a (labeled) graph G. If e is a non-loop edge of G, then $\tau(G) = \tau(G \setminus e) + \tau(G/e)$.

Example 2.15



G



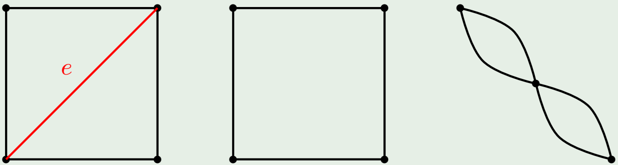
$G \setminus e$

Counting Spanning Trees

Theorem 2.14

Let $\tau(G)$ denote the number of distinct spanning trees of a (labeled) graph G. If e is a non-loop edge of G, then $\tau(G) = \tau(G \setminus e) + \tau(G/e)$.

Example 2.15



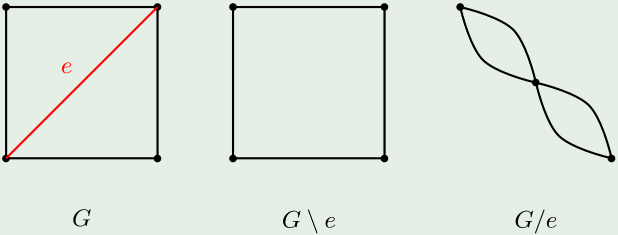
G $\quad$ $G \setminus e$ $\quad$ G/e

Counting Spanning Trees

Theorem 2.14

Let $\tau(G)$ denote the number of distinct spanning trees of a (labeled) graph G. If e is a non-loop edge of G, then $\tau(G) = \tau(G \setminus e) + \tau(G/e)$.

Example 2.15



$$\tau(G) = \tau(G \setminus e) + \tau(G/e) = 4 + 4 = 8$$

Proof of Spanning Tree Formula

- The spanning trees of $G \setminus e$ are precisely the spanning trees of G that avoid e.

Proof of Spanning Tree Formula

- The spanning trees of $G \setminus e$ are precisely the spanning trees of G that avoid e.
- The spanning trees of G/e correspond to the spanning trees of G using e.

Proof of Spanning Tree Formula

- The spanning trees of $G \setminus e$ are precisely the spanning trees of G that avoid e.
- The spanning trees of G/e correspond to the spanning trees of G using e. (If T is a spanning tree of G/e, then $E(T) \cup e$ form the edge-set of a spanning tree of G.)

Proof of Spanning Tree Formula

- The spanning trees of $G \setminus e$ are precisely the spanning trees of G that avoid e.
- The spanning trees of G/e correspond to the spanning trees of G using e. (If T is a spanning tree of G/e, then $E(T) \cup e$ form the edge-set of a spanning tree of G.)
- The formula follows.

Proof of Spanning Tree Formula

- The spanning trees of $G \setminus e$ are precisely the spanning trees of G that avoid e.
- The spanning trees of G/e correspond to the spanning trees of G using e. (If T is a spanning tree of G/e, then $E(T) \cup e$ form the edge-set of a spanning tree of G.)
- The formula follows.

Using the deletion-contraction formula for calculating the number of spanning trees is inefficient.

Proof of Spanning Tree Formula

- The spanning trees of $G \setminus e$ are precisely the spanning trees of G that avoid e.
- The spanning trees of G/e correspond to the spanning trees of G using e. (If T is a spanning tree of G/e, then $E(T) \cup e$ form the edge-set of a spanning tree of G.)
- The formula follows.

Using the deletion-contraction formula for calculating the number of spanning trees is inefficient. A much more efficient method is to construct a special matrix, called the Laplacian of the graph, and to compute its determinant.

Bipartite Graphs

Definition 3.1

A graph G is bipartite if the vertex set of G can be partitioned into sets X and Y such that every edge of G joins a vertex in X to a vertex in Y.

Bipartite Graphs

Definition 3.1

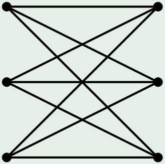
A graph G is bipartite if the vertex set of G can be partitioned into sets X and Y such that every edge of G joins a vertex in X to a vertex in Y. The simple bipartite graph in which each of the m vertices of X is joined to each of the n vertices of Y is called complete bipartite and denoted $K_{m,n}$.

Bipartite Graphs

Definition 3.1

A graph G is bipartite if the vertex set of G can be partitioned into sets X and Y such that every edge of G joins a vertex in X to a vertex in Y. The simple bipartite graph in which each of the m vertices of X is joined to each of the n vertices of Y is called complete bipartite and denoted $K_{m,n}$.

Example 3.2

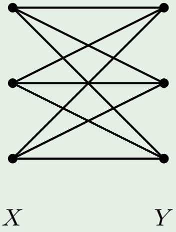


Bipartite Graphs

Definition 3.1

A graph G is bipartite if the vertex set of G can be partitioned into sets X and Y such that every edge of G joins a vertex in X to a vertex in Y. The simple bipartite graph in which each of the m vertices of X is joined to each of the n vertices of Y is called complete bipartite and denoted $K_{m,n}$.

Example 3.2

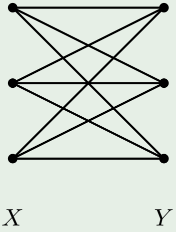


Bipartite Graphs

Definition 3.1

A graph G is bipartite if the vertex set of G can be partitioned into sets X and Y such that every edge of G joins a vertex in X to a vertex in Y. The simple bipartite graph in which each of the m vertices of X is joined to each of the n vertices of Y is called complete bipartite and denoted $K_{m,n}$.

Example 3.2



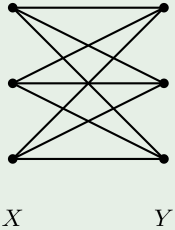
$K_{3,3}$

Bipartite Graphs

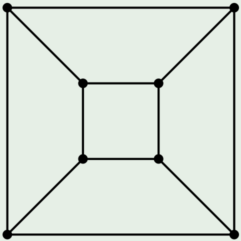
Definition 3.1

A graph G is bipartite if the vertex set of G can be partitioned into sets X and Y such that every edge of G joins a vertex in X to a vertex in Y. The simple bipartite graph in which each of the m vertices of X is joined to each of the n vertices of Y is called complete bipartite and denoted $K_{m,n}$.

Example 3.2



$K_{3,3}$

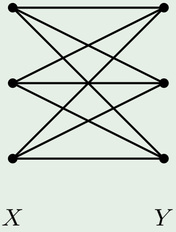


Bipartite Graphs

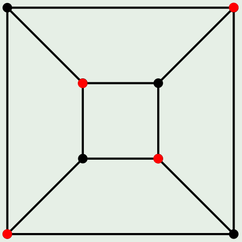
Definition 3.1

A graph G is bipartite if the vertex set of G can be partitioned into sets X and Y such that every edge of G joins a vertex in X to a vertex in Y. The simple bipartite graph in which each of the m vertices of X is joined to each of the n vertices of Y is called complete bipartite and denoted $K_{m,n}$.

Example 3.2



$K_{3,3}$



Characterization of Bipartite Graphs

Theorem 3.3

A graph is bipartite if and only if it has no cycles of odd length.

Characterization of Bipartite Graphs

Theorem 3.3

A graph is bipartite if and only if it has no cycles of odd length.

Proof.

Necessity is clear: every cycle of G must alternate between a vertex in X and a vertex in Y, and so it must be of even length.

$\square$

Characterization of Bipartite Graphs

Theorem 3.3

A graph is bipartite if and only if it has no cycles of odd length.

Proof.

Necessity is clear: every cycle of G must alternate between a vertex in X and a vertex in Y, and so it must be of even length.
For sufficiency, we may assume that G is connected. $\square$

Characterization of Bipartite Graphs

Theorem 3.3

A graph is bipartite if and only if it has no cycles of odd length.

Proof.

Necessity is clear: every cycle of G must alternate between a vertex in X and a vertex in Y, and so it must be of even length.
For sufficiency, we may assume that G is connected. Now, pick a vertex x of G, and let X be the set of vertices whose distance from x is even, and let Y be the set of vertices whose distance from x is odd. $\square$

Theorem 3.3

A graph is bipartite if and only if it has no cycles of odd length.

Proof.

Necessity is clear: every cycle of G must alternate between a vertex in X and a vertex in Y, and so it must be of even length.
For sufficiency, we may assume that G is connected. Now, pick a vertex x of G, and let X be the set of vertices whose distance from x is even, and let Y be the set of vertices whose distance from x is odd. Clearly, $\{X, Y\}$ is a partition of $V(G)$. $\square$

Theorem 3.3

A graph is bipartite if and only if it has no cycles of odd length.

Proof.

Necessity is clear: every cycle of G must alternate between a vertex in X and a vertex in Y, and so it must be of even length.
For sufficiency, we may assume that G is connected. Now, pick a vertex x of G, and let X be the set of vertices whose distance from x is even, and let Y be the set of vertices whose distance from x is odd. Clearly, $\{X, Y\}$ is a partition of $V(G)$. Suppose now that some two vertices of X or some two vertices of Y, say x_1 and x_2, are adjacent. □

Characterization of Bipartite Graphs

Theorem 3.3

A graph is bipartite if and only if it has no cycles of odd length.

Proof.

Necessity is clear: every cycle of G must alternate between a vertex in X and a vertex in Y, and so it must be of even length.
For sufficiency, we may assume that G is connected. Now, pick a vertex x of G, and let X be the set of vertices whose distance from x is even, and let Y be the set of vertices whose distance from x is odd. Clearly, $\{X, Y\}$ is a partition of $V(G)$. Suppose now that some two vertices of X or some two vertices of Y, say x_1 and x_2, are adjacent. Let P_1 be a shortest path from x to x_1 and let P_2 be a shortest path from x to x_2. □

Characterization of Bipartite Graphs

Theorem 3.3

A graph is bipartite if and only if it has no cycles of odd length.

Proof.

Necessity is clear: every cycle of G must alternate between a vertex in X and a vertex in Y, and so it must be of even length.

For sufficiency, we may assume that G is connected. Now, pick a vertex x of G, and let X be the set of vertices whose distance from x is even, and let Y be the set of vertices whose distance from x is odd. Clearly, $\{X, Y\}$ is a partition of $V(G)$. Suppose now that some two vertices of X or some two vertices of Y, say x_1 and x_2, are adjacent. Let P_1 be a shortest path from x to x_1 and let P_2 be a shortest path from x to x_2. Let u be vertex on $P_1 \cap P_2$ that the the farthest from x, and let P_1' and P_2' be the subpaths of, respectively, P_1 and P_2, from u to x_1 and from u to x_2. □

Characterization of Bipartite Graphs

Theorem 3.3

A graph is bipartite if and only if it has no cycles of odd length.

Proof.

Necessity is clear: every cycle of G must alternate between a vertex in X and a vertex in Y, and so it must be of even length.

For sufficiency, we may assume that G is connected. Now, pick a vertex x of G, and let X be the set of vertices whose distance from x is even, and let Y be the set of vertices whose distance from x is odd. Clearly, $\{X, Y\}$ is a partition of $V(G)$. Suppose now that some two vertices of X or some two vertices of Y, say x_1 and x_2, are adjacent. Let P_1 be a shortest path from x to x_1 and let P_2 be a shortest path from x to x_2. Let u be vertex on $P_1 \cap P_2$ that the the farthest from x, and let P_1' and P_2' be the subpaths of, respectively, P_1 and P_2, from u to x_1 and from u to x_2. Then P_1' and P_2' have the same length, and so the cycle $P_1' \cup P_2' \cup x_1x_2$ has odd length. $\square$

Characterization of Bipartite Graphs

Theorem 3.3

A graph is bipartite if and only if it has no cycles of odd length.

Proof.

Necessity is clear: every cycle of G must alternate between a vertex in X and a vertex in Y, and so it must be of even length.

For sufficiency, we may assume that G is connected. Now, pick a vertex x of G, and let X be the set of vertices whose distance from x is even, and let Y be the set of vertices whose distance from x is odd. Clearly, $\{X, Y\}$ is a partition of $V(G)$. Suppose now that some two vertices of X or some two vertices of Y, say x_1 and x_2, are adjacent. Let P_1 be a shortest path from x to x_1 and let P_2 be a shortest path from x to x_2. Let u be vertex on $P_1 \cap P_2$ that the the farthest from x, and let P_1' and P_2' be the subpaths of, respectively, P_1 and P_2, from u to x_1 and from u to x_2. Then P_1' and P_2' have the same length, and so the cycle $P_1' \cup P_2' \cup x_1x_2$ has odd length. This proves that x_1 and x_2 cannot be adjacent. $\square$

Matching

Definition 3.4

- A matching is a set of pairwise non-adjacent edges.

Matching

Definition 3.4

- A matching is a set of pairwise non-adjacent edges.
- A matching is perfect (is a 1-factor) if it meets every vertex of the graph.

Matching

Definition 3.4

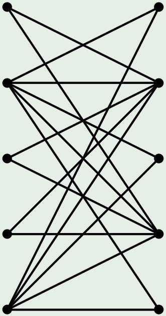
- A matching is a set of pairwise non-adjacent edges.
- A matching is perfect (is a 1-factor) if it meets every vertex of the graph.
- A matching saturates the set X of vertices if each vertex in X is incident with an edge in the matching.

Matching

Definition 3.4

- A matching is a set of pairwise non-adjacent edges.
- A matching is perfect (is a 1-factor) if it meets every vertex of the graph.
- A matching saturates the set X of vertices if each vertex in X is incident with an edge in the matching.

Example 3.5



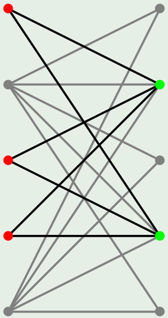
Does G have a matching that saturates all vertices on the left side?

Matching

Definition 3.4

- A matching is a set of pairwise non-adjacent edges.
- A matching is perfect (is a 1-factor) if it meets every vertex of the graph.
- A matching saturates the set X of vertices if each vertex in X is incident with an edge in the matching.

Example 3.5



Does G have a matching that saturates all vertices on the left side?
No! Look at S, which has 3 elements, and $N(S)$, which has only 2 elements.

Hall's Marriage Theorem

Theorem 3.6 (Hall's Marriage Theorem, 1935)

Suppose G is a bipartite graph with bipartition $\{X, Y\}$. The graph G has a matching saturating X if and only if $|N(S)| \geq |S|$ for every subset S of X.

Hall's Marriage Theorem

Theorem 3.6 (Hall's Marriage Theorem, 1935)

Suppose G is a bipartite graph with bipartition $\{X, Y\}$. The graph G has a matching saturating X if and only if $|N(S)| \geq |S|$ for every subset S of X.

Definition 3.7

- Given a matching M, an M-alternating path is a path that alternates between edges in M and edges not in M.

Hall's Marriage Theorem

Theorem 3.6 (Hall's Marriage Theorem, 1935)

Suppose G is a bipartite graph with bipartition $\{X, Y\}$. The graph G has a matching saturating X if and only if $|N(S)| \geq |S|$ for every subset S of X.

Definition 3.7

- Given a matching M, an M-alternating path is a path that alternates between edges in M and edges not in M.
- A non-trivial M-alternating path P that begins and ends at M-unsaturated vertices is an M-augmenting path.

Hall's Marriage Theorem

Theorem 3.6 (Hall's Marriage Theorem, 1935)

Suppose G is a bipartite graph with bipartition $\{X, Y\}$. The graph G has a matching saturating X if and only if $|N(S)| \geq |S|$ for every subset S of X.

Definition 3.7

- Given a matching M, an M-alternating path is a path that alternates between edges in M and edges not in M.
- A non-trivial M-alternating path P that begins and ends at M-unsaturated vertices is an M-augmenting path.
- Replacing $M \cap E(P)$ by $E(P) \setminus M$ produces a new matching M' that has one more edge than M.

Hall's Marriage Theorem

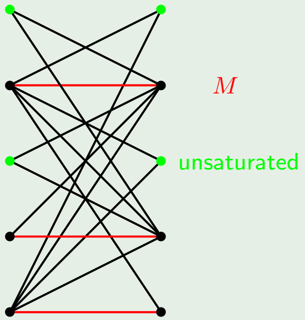
Theorem 3.6 (Hall's Marriage Theorem, 1935)

Suppose G is a bipartite graph with bipartition $\{X, Y\}$. The graph G has a matching saturating X if and only if $|N(S)| \geq |S|$ for every subset S of X.

Definition 3.7

- Given a matching M, an M-alternating path is a path that alternates between edges in M and edges not in M.
- A non-trivial M-alternating path P that begins and ends at M-unsaturated vertices is an M-augmenting path.
- Replacing $M \cap E(P)$ by $E(P) \setminus M$ produces a new matching M' that has one more edge than M.

Example 3.8



Hall's Marriage Theorem

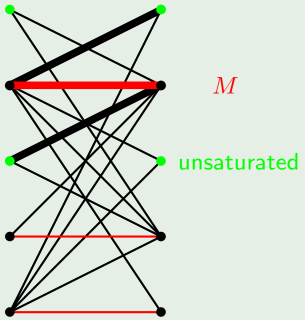
Theorem 3.6 (Hall's Marriage Theorem, 1935)

Suppose G is a bipartite graph with bipartition $\{X, Y\}$. The graph G has a matching saturating X if and only if $|N(S)| \geq |S|$ for every subset S of X.

Definition 3.7

- Given a matching M, an M-alternating path is a path that alternates between edges in M and edges not in M.
- A non-trivial M-alternating path P that begins and ends at M-unsaturated vertices is an M-augmenting path.
- Replacing $M \cap E(P)$ by $E(P) \setminus M$ produces a new matching M' that has one more edge than M.

Example 3.8



Hall's Marriage Theorem

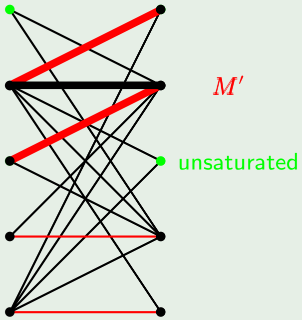
Theorem 3.6 (Hall's Marriage Theorem, 1935)

Suppose G is a bipartite graph with bipartition $\{X, Y\}$. The graph G has a matching saturating X if and only if $|N(S)| \geq |S|$ for every subset S of X.

Definition 3.7

- Given a matching M, an M-alternating path is a path that alternates between edges in M and edges not in M.
- A non-trivial M-alternating path P that begins and ends at M-unsaturated vertices is an M-augmenting path.
- Replacing $M \cap E(P)$ by $E(P) \setminus M$ produces a new matching M' that has one more edge than M.

Example 3.8



Augmenting Paths

Theorem 3.9 (Berge 1957)

A matching M in a bipartite graph G is a maximum matching in G if and only if G has no M-augmenting path.

Augmenting Paths

Theorem 3.9 (Berge 1957)

A matching M in a bipartite graph G is a maximum matching in G if and only if G has no M-augmenting path.

Proof.

It is clear that if G has an M-augmenting path, then M is not maximum. $\square$

Augmenting Paths

Theorem 3.9 (Berge 1957)

A matching M in a bipartite graph G is a maximum matching in G if and only if G has no M-augmenting path.

Proof.

It is clear that if G has an M-augmenting path, then M is not maximum. Suppose now that G has a matching M' that is larger than M and let F be the subgraph of G induced by the symmetric difference of M and M', that is, by all those edges that are in exactly one of M and M'.

$\square$

Augmenting Paths

Theorem 3.9 (Berge 1957)

A matching M in a bipartite graph G is a maximum matching in G if and only if G has no M-augmenting path.

Proof.

It is clear that if G has an M-augmenting path, then M is not maximum. Suppose now that G has a matching M' that is larger than M and let F be the subgraph of G induced by the symmetric difference of M and M', that is, by all those edges that are in exactly one of M and M'. The the maximum degree of F is at most 2, each component of F is a path or a cycle. $\square$

Augmenting Paths

Theorem 3.9 (Berge 1957)

A matching M in a bipartite graph G is a maximum matching in G if and only if G has no M-augmenting path.

Proof.

It is clear that if G has an M-augmenting path, then M is not maximum. Suppose now that G has a matching M' that is larger than M and let F be the subgraph of G induced by the symmetric difference of M and M', that is, by all those edges that are in exactly one of M and M'. The the maximum degree of F is at most 2, each component of F is a path or a cycle. Every path and every cycle in F alternates between edges in M and edges in M'.

□

Augmenting Paths

Theorem 3.9 (Berge 1957)

A matching M in a bipartite graph G is a maximum matching in G if and only if G has no M-augmenting path.

Proof.

It is clear that if G has an M-augmenting path, then M is not maximum. Suppose now that G has a matching M' that is larger than M and let F be the subgraph of G induced by the symmetric difference of M and M', that is, by all those edges that are in exactly one of M and M'. The the maximum degree of F is at most 2, each component of F is a path or a cycle. Every path and every cycle in F alternates between edges in M and edges in M'. Thus each cycle in F has the same number of edges from M and from M'.

□

Augmenting Paths

Theorem 3.9 (Berge 1957)

A matching M in a bipartite graph G is a maximum matching in G if and only if G has no M-augmenting path.

Proof.

It is clear that if G has an M-augmenting path, then M is not maximum. Suppose now that G has a matching M' that is larger than M and let F be the subgraph of G induced by the symmetric difference of M and M', that is, by all those edges that are in exactly one of M and M'. The the maximum degree of F is at most 2, each component of F is a path or a cycle. Every path and every cycle in F alternates between edges in M and edges in M'. Thus each cycle in F has the same number of edges from M and from M'. Since $|M'| > |M|$, there must be a component of F that is a path with more edges from M' than from M—an M-augmenting path. □

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.
Necessity is clear.

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.
Necessity is clear.
To prove sufficiency, suppose the Hall's condition holds, let M be a maximum matching, and suppose $u \in X$ is unsaturated.

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.
Necessity is clear.
To prove sufficiency, suppose the Hall's condition holds, let M be a maximum matching, and suppose $u \in X$ is unsaturated. Let S and T be subsets of X and Y, respectively, that are reachable from u by M-alternating paths.

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.
Necessity is clear.
To prove sufficiency, suppose the Hall's condition holds, let M be a maximum matching, and suppose $u \in X$ is unsaturated. Let S and T be subsets of X and Y, respectively, that are reachable from u by M-alternating paths. These paths reach Y from u along edges not in M, and reach X along edges in M.

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.
Necessity is clear.
To prove sufficiency, suppose the Hall's condition holds, let M be a maximum matching, and suppose $u \in X$ is unsaturated. Let S and T be subsets of X and Y, respectively, that are reachable from u by M-alternating paths. These paths reach Y from u along edges not in M, and reach X along edges in M. Hence every vertex in $S - u$ is reached along an edge in M from a vertex in T.

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.
Necessity is clear.
To prove sufficiency, suppose the Hall's condition holds, let M be a maximum matching, and suppose $u \in X$ is unsaturated. Let S and T be subsets of X and Y, respectively, that are reachable from u by M-alternating paths. These paths reach Y from u along edges not in M, and reach X along edges in M.
Hence every vertex in $S - u$ is reached along an edge in M from a vertex in T.
Since there are no augmenting paths, every vertex in T is saturated.

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.
Necessity is clear.
To prove sufficiency, suppose the Hall's condition holds, let M be a maximum matching, and suppose $u \in X$ is unsaturated. Let S and T be subsets of X and Y, respectively, that are reachable from u by M-alternating paths. These paths reach Y from u along edges not in M, and reach X along edges in M. Hence every vertex in $S - u$ is reached along an edge in M from a vertex in T. Since there are no augmenting paths, every vertex in T is saturated. Hence the edges of M establish a bijection between T and $S - u$.

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.
Necessity is clear.
To prove sufficiency, suppose the Hall's condition holds, let M be a maximum matching, and suppose $u \in X$ is unsaturated. Let S and T be subsets of X and Y, respectively, that are reachable from u by M-alternating paths. These paths reach Y from u along edges not in M, and reach X along edges in M. Hence every vertex in $S - u$ is reached along an edge in M from a vertex in T. Since there are no augmenting paths, every vertex in T is saturated. Hence the edges of M establish a bijection between T and $S - u$. Note that an edge between S and $y \in Y - T$ would be an edge not in M, and thus create an M-augmenting path to y, which contradicts $y \notin T$.

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.
Necessity is clear.
To prove sufficiency, suppose the Hall's condition holds, let M be a maximum matching, and suppose $u \in X$ is unsaturated. Let S and T be subsets of X and Y, respectively, that are reachable from u by M-alternating paths. These paths reach Y from u along edges not in M, and reach X along edges in M. Hence every vertex in $S - u$ is reached along an edge in M from a vertex in T. Since there are no augmenting paths, every vertex in T is saturated. Hence the edges of M establish a bijection between T and $S - u$. Note that an edge between S and $y \in Y - T$ would be an edge not in M, and thus create an M-augmenting path to y, which contradicts $y \notin T$. Hence $T = N(S)$,

Proof of Hall's Theorem

Recall the Hall's Condition: $|N(S)| \geq |S|$ for every $S \subseteq X$.
Necessity is clear.
To prove sufficiency, suppose the Hall's condition holds, let M be a maximum matching, and suppose $u \in X$ is unsaturated. Let S and T be subsets of X and Y, respectively, that are reachable from u by M-alternating paths. These paths reach Y from u along edges not in M, and reach X along edges in M. Hence every vertex in $S - u$ is reached along an edge in M from a vertex in T. Since there are no augmenting paths, every vertex in T is saturated. Hence the edges of M establish a bijection between T and $S - u$. Note that an edge between S and $y \in Y - T$ would be an edge not in M, and thus create an M-augmenting path to y, which contradicts $y \notin T$. Hence $T = N(S)$, and $|N(S)| = |T| = |S| - 1 < |S|$; a contradiction.

Regular Graphs

Definition 3.10

- A graph is regular if all vertices have the same degree.

Regular Graphs

Definition 3.10

- A graph is regular if all vertices have the same degree.
- If every vertex degree has the same value k, then the graph is k-regular.

Regular Graphs

Definition 3.10

- A graph is regular if all vertices have the same degree.
- If every vertex degree has the same value k, then the graph is k-regular.
- A 3-regular graph is sometimes called cubic.

Regular Graphs

Definition 3.10

- A graph is regular if all vertices have the same degree.
- If every vertex degree has the same value k, then the graph is k-regular.
- A 3-regular graph is sometimes called cubic.

Corollary 3.11

If G is a k-regular bipartite graph for some $k > 0$, then G has a perfect matching.

Regular Graphs

Definition 3.10

- A graph is regular if all vertices have the same degree.
- If every vertex degree has the same value k, then the graph is k-regular.
- A 3-regular graph is sometimes called cubic.

Corollary 3.11

If G is a k-regular bipartite graph for some $k > 0$, then G has a perfect matching.

Proof.

Counting the edges by endpoints in X and by endpoints in Y, we conclude that $k|X| = k|Y|$, and so $|X| = |Y|$, and so every matching saturating X is perfect. $\square$

Regular Graphs

Definition 3.10

- A graph is regular if all vertices have the same degree.
- If every vertex degree has the same value k, then the graph is k-regular.
- A 3-regular graph is sometimes called cubic.

Corollary 3.11

If G is a k-regular bipartite graph for some $k > 0$, then G has a perfect matching.

Proof.

Counting the edges by endpoints in X and by endpoints in Y, we conclude that $k|X| = k|Y|$, and so $|X| = |Y|$, and so every matching saturating X is perfect. Consider $S \subseteq X$, and suppose that there are m edges between S and $N(S)$.

$\square$

Regular Graphs

Definition 3.10

- A graph is regular if all vertices have the same degree.
- If every vertex degree has the same value k, then the graph is k-regular.
- A 3-regular graph is sometimes called cubic.

Corollary 3.11

If G is a k-regular bipartite graph for some $k > 0$, then G has a perfect matching.

Proof.

Counting the edges by endpoints in X and by endpoints in Y, we conclude that $k|X| = k|Y|$, and so $|X| = |Y|$, and so every matching saturating X is perfect. Consider $S \subseteq X$, and suppose that there are m edges between S and $N(S)$. Since G is k-regular, we have $m = k|S|$.

$\square$

Regular Graphs

Definition 3.10

- A graph is regular if all vertices have the same degree.
- If every vertex degree has the same value k, then the graph is k-regular.
- A 3-regular graph is sometimes called cubic.

Corollary 3.11

If G is a k-regular bipartite graph for some $k > 0$, then G has a perfect matching.

Proof.

Counting the edges by endpoints in X and by endpoints in Y, we conclude that $k|X| = k|Y|$, and so $|X| = |Y|$, and so every matching saturating X is perfect. Consider $S \subseteq X$, and suppose that there are m edges between S and $N(S)$. Since G is k-regular, we have $m = k|S|$. Since these m edges are incident to $N(S)$, we have $m \leq k|N(S)|$.

$\square$

Regular Graphs

Definition 3.10

- A graph is regular if all vertices have the same degree.
- If every vertex degree has the same value k, then the graph is k-regular.
- A 3-regular graph is sometimes called cubic.

Corollary 3.11

If G is a k-regular bipartite graph for some $k > 0$, then G has a perfect matching.

Proof.

Counting the edges by endpoints in X and by endpoints in Y, we conclude that $k|X| = k|Y|$, and so $|X| = |Y|$, and so every matching saturating X is perfect. Consider $S \subseteq X$, and suppose that there are m edges between S and $N(S)$. Since G is k-regular, we have $m = k|S|$. Since these m edges are incident to $N(S)$, we have $m \leq k|N(S)|$. Hence $k|S| \leq k|N(S)|$ and the Hall's condition holds. $\square$

Vertex Covers

Definition 3.12

- A vertex cover of G is a set S of vertices such that every edge of G is incident with at least one element of S.

Vertex Covers

Definition 3.12

- A vertex cover of G is a set S of vertices such that every edge of G is incident with at least one element of S.
- The vertices in S cover the edges of G.

Vertex Covers

Definition 3.12

- A vertex cover of G is a set S of vertices such that every edge of G is incident with at least one element of S.
- The vertices in S cover the edges of G.

Theorem 3.13 (König-Egerváry 1931)

If G is a bipartite graph, then the maximum size of a matching in G equals the minimum size of a vertex cover in G.

Vertex Covers

Definition 3.12

- A vertex cover of G is a set S of vertices such that every edge of G is incident with at least one element of S.
- The vertices in S cover the edges of G.

Theorem 3.13 (König-Egerváry 1931)

If G is a bipartite graph, then the maximum size of a matching in G equals the minimum size of a vertex cover in G.

Easy Direction.

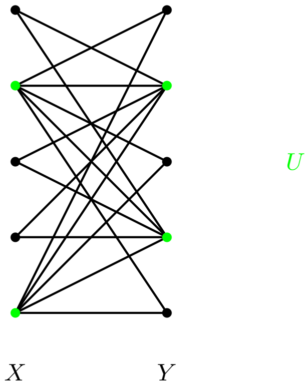
Since distinct vertices must be used to cover the edges of a matching, we have $|U| \geq |M|$ whenever U is a vertex cover and M is a matching. $\square$

Proof of König-Egerváry Theorem, Continued

Given a minimum vertex cover U, we construct a matching of size $|U|$.

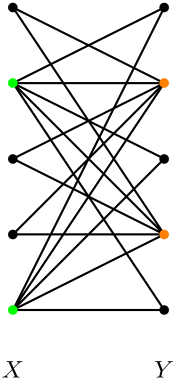
Proof of König-Egerváry Theorem, Continued

Given a minimum vertex cover U, we construct a matching of size $|U|$. Suppose G has bipartition $\{X, Y\}$.



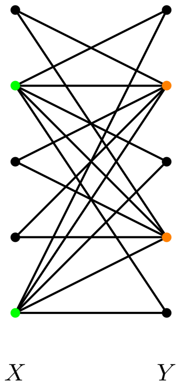
Proof of König-Egerváry Theorem, Continued

Given a minimum vertex cover U, we construct a matching of size $|U|$. Suppose G has bipartition $\{X, Y\}$. Let $R = U \cap X$ and $T = U \cap Y$.



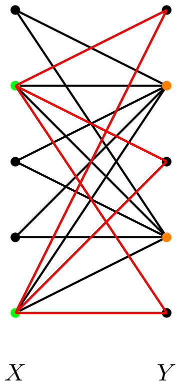
Proof of König-Egerváry Theorem, Continued

Given a minimum vertex cover U, we construct a matching of size $|U|$. Suppose G has bipartition $\{X, Y\}$. Let $R = U \cap X$ and $T = U \cap Y$. Let H and H' be the subgraphs of G induced by $R \cup (Y - T)$ and $T \cup (X - R)$, respectively.



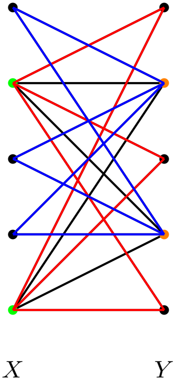
Proof of König-Egerváry Theorem, Continued

Given a minimum vertex cover U, we construct a matching of size $|U|$. Suppose G has bipartition $\{X, Y\}$. Let $R = U \cap X$ and $T = U \cap Y$. Let H and H' be the subgraphs of G induced by $R \cup (Y - T)$ and $T \cup (X - R)$, respectively. We use 3.6 to show H has a matching saturating R, and H' has a matching saturating T.



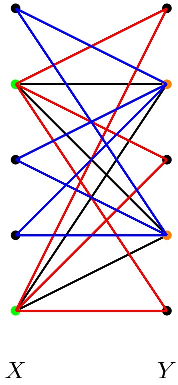
Proof of König-Egerváry Theorem, Continued

Given a minimum vertex cover U, we construct a matching of size $|U|$. Suppose G has bipartition $\{X, Y\}$. Let $R = U \cap X$ and $T = U \cap Y$. Let H and H' be the subgraphs of G induced by $R \cup (Y - T)$ and $T \cup (X - R)$, respectively. We use 3.6 to show H has a matching saturating R, and H' has a matching saturating T. Suppose $S \subseteq R$ and consider $N_H(S) \subseteq Y - T$.



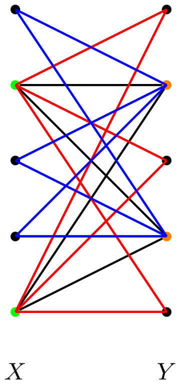
Proof of König-Egerváry Theorem, Continued

Given a minimum vertex cover U, we construct a matching of size $|U|$. Suppose G has bipartition $\{X, Y\}$. Let $R = U \cap X$ and $T = U \cap Y$. Let H and H' be the subgraphs of G induced by $R \cup (Y - T)$ and $T \cup (X - R)$, respectively. We use 3.6 to show H has a matching saturating R, and H' has a matching saturating T. Suppose $S \subseteq R$ and consider $N_H(S) \subseteq Y - T$. If $|N_H(S)| < |S|$, then we can substitute $N_H(S)$ for S in U to obtain a smaller vertex cover, which is impossible.



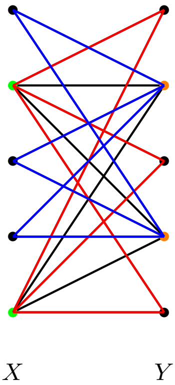
Proof of König-Egerváry Theorem, Continued

Given a minimum vertex cover U, we construct a matching of size $|U|$. Suppose G has bipartition $\{X, Y\}$. Let $R = U \cap X$ and $T = U \cap Y$. Let H and H' be the subgraphs of G induced by $R \cup (Y - T)$ and $T \cup (X - R)$, respectively. We use 3.6 to show H has a matching saturating R, and H' has a matching saturating T. Suppose $S \subseteq R$ and consider $N_H(S) \subseteq Y - T$. If $|N_H(S)| < |S|$, then we can substitute $N_H(S)$ for S in U to obtain a smaller vertex cover, which is impossible. Hence H satisfies the Hall's condition and so has a matching of size $|R|$.



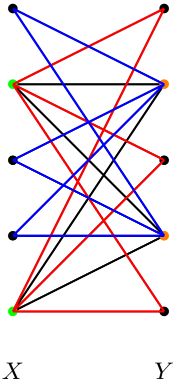
Proof of König-Egerváry Theorem, Continued

Given a minimum vertex cover U, we construct a matching of size $|U|$. Suppose G has bipartition $\{X, Y\}$. Let $R = U \cap X$ and $T = U \cap Y$. Let H and H' be the subgraphs of G induced by $R \cup (Y - T)$ and $T \cup (X - R)$, respectively. We use 3.6 to show H has a matching saturating R, and H' has a matching saturating T. Suppose $S \subseteq R$ and consider $N_H(S) \subseteq Y - T$. If $|N_H(S)| < |S|$, then we can substitute $N_H(S)$ for S in U to obtain a smaller vertex cover, which is impossible. Hence H satisfies the Hall's condition and so has a matching of size $|R|$. Likewise, H' has a matching of size $|T|$.



Proof of König-Egerváry Theorem, Continued

Given a minimum vertex cover U, we construct a matching of size $|U|$. Suppose G has bipartition $\{X, Y\}$. Let $R = U \cap X$ and $T = U \cap Y$. Let H and H' be the subgraphs of G induced by $R \cup (Y - T)$ and $T \cup (X - R)$, respectively. We use 3.6 to show H has a matching saturating R, and H' has a matching saturating T. Suppose $S \subseteq R$ and consider $N_H(S) \subseteq Y - T$. If $|N_H(S)| < |S|$, then we can substitute $N_H(S)$ for S in U to obtain a smaller vertex cover, which is impossible. Hence H satisfies the Hall's condition and so has a matching of size $|R|$. Likewise, H' has a matching of size $|T|$. The union of these two matchings is a matching of G of size $|U|$.



Matchings in Non-Bipartite Graphs

Note 3.14

- *Hall's Marriage Theorem 3.6 does not make sense for non-bipartite graphs.*

Matchings in Non-Bipartite Graphs

Note 3.14

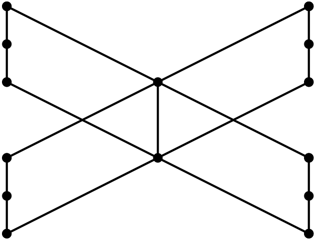
- *Hall's Marriage Theorem 3.6 does not make sense for non-bipartite graphs.*
- *König-Egerváry Theorem 3.13 fails, in general, for non-bipartite graphs.*

Matchings in Non-Bipartite Graphs

Note 3.14

- *Hall's Marriage Theorem 3.6 does not make sense for non-bipartite graphs.*
- *König-Egerváry Theorem 3.13 fails, in general, for non-bipartite graphs.*

Does the graph below have a perfect matching?

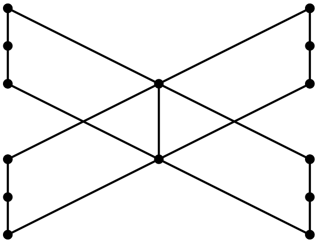


Matchings in Non-Bipartite Graphs

Note 3.14

- *Hall's Marriage Theorem 3.6 does not make sense for non-bipartite graphs.*
- *König-Egerváry Theorem 3.13 fails, in general, for non-bipartite graphs.*

Does the graph below have a perfect matching?



No, since removing the two vertices in the middle leaves more than two components of odd order.

Tutte's 1-Factor Theorem

Definition 3.15

- A graph (or component) is odd (even) if it has odd (even) order.

Tutte's 1-Factor Theorem

Definition 3.15

- A graph (or component) is odd (even) if it has odd (even) order.
- The number of odd components in a graph G will be denoted by $q(G)$.

Tutte's 1-Factor Theorem

Definition 3.15

- A graph (or component) is odd (even) if it has odd (even) order.
- The number of odd components in a graph G will be denoted by $q(G)$.

Theorem 3.16 (Tutte 1-Factor)

A graph G has a perfect matching if and only if $q(G - S) \leq |S|$ for every $S \subseteq V(G)$.

Tutte's 1-Factor Theorem

Definition 3.15

- A graph (or component) is odd (even) if it has odd (even) order.
- The number of odd components in a graph G will be denoted by $q(G)$.

Theorem 3.16 (Tutte 1-Factor)

A graph G has a perfect matching if and only if $q(G - S) \leq |S|$ for every $S \subseteq V(G)$.

Necessity.

If Q is an odd component of $G - S$, then a perfect matching must contain at least one edge between Q and S. $\square$

Tutte's 1-Factor Theorem

Definition 3.15

- A graph (or component) is odd (even) if it has odd (even) order.
- The number of odd components in a graph G will be denoted by $q(G)$.

Theorem 3.16 (Tutte 1-Factor)

A graph G has a perfect matching if and only if $q(G - S) \leq |S|$ for every $S \subseteq V(G)$.

Necessity.

If Q is an odd component of $G - S$, then a perfect matching must contain at least one edge between Q and S. Since edges in a matching are non-adjacent, the condition follows. $\square$

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G - S) \leq |S|$ for every $S \subseteq V(G)$.

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G-S) \leq |S|$ for every $S \subseteq V(G)$.
We assume that the condition holds and produce a perfect matching.

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G - S) \leq |S|$ for every $S \subseteq V(G)$.
We assume that the condition holds and produce a perfect matching. We proceed by induction on the order of G.

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G-S) \leq |S|$ for every $S \subseteq V(G)$.
We assume that the condition holds and produce a perfect matching. We proceed by induction on the order of G. The claim is trivial if $|G| \leq 2$.

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G - S) \leq |S|$ for every $S \subseteq V(G)$.
We assume that the condition holds and produce a perfect matching. We proceed by induction on the order of G. The claim is trivial if $|G| \leq 2$.

Now suppose that the Tutte Condition holds for G, which has order $n > 2$, and that the theorem holds for all graphs of smaller order.

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G-S) \leq |S|$ for every $S \subseteq V(G)$.
We assume that the condition holds and produce a perfect matching. We proceed by induction on the order of G. The claim is trivial if $|G| \leq 2$.

Now suppose that the Tutte Condition holds for G, which has order $n > 2$, and that the theorem holds for all graphs of smaller order. First note that $q(G-v) = 1 = |\{v\}|$, and so we may pick S_0 to be a maximal subset of $V(G)$ such that $q(G-S_0) = |S_0|$.

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G - S) \leq |S|$ for every $S \subseteq V(G)$.
We assume that the condition holds and produce a perfect matching. We proceed by induction on the order of G. The claim is trivial if $|G| \leq 2$.

Now suppose that the Tutte Condition holds for G, which has order $n > 2$, and that the theorem holds for all graphs of smaller order. First note that $q(G - v) = 1 = |\{v\}|$, and so we may pick S_0 to be a maximal subset of $V(G)$ such that $q(G - S_0) = |S_0|$. Let Q_1, Q_2, ..., Q_m be the odd components of $G - S_0$, and let D_1, D_2, ..., D_k be the even components of $G - S_0$.

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G - S) \leq |S|$ for every $S \subseteq V(G)$.
We assume that the condition holds and produce a perfect matching. We proceed by induction on the order of G. The claim is trivial if $|G| \leq 2$.

Now suppose that the Tutte Condition holds for G, which has order $n > 2$, and that the theorem holds for all graphs of smaller order. First note that $q(G - v) = 1 = |\{v\}|$, and so we may pick S_0 to be a maximal subset of $V(G)$ such that $q(G - S_0) = |S_0|$. Let $Q_1, Q_2, \ldots, Q_m$ be the odd components of $G - S_0$, and let $D_1, D_2, \ldots, D_k$ be the even components of $G - S_0$. We will show that:

(1) each D_j has a perfect matching;

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G - S) \leq |S|$ for every $S \subseteq V(G)$.
We assume that the condition holds and produce a perfect matching. We proceed by induction on the order of G. The claim is trivial if $|G| \leq 2$.

Now suppose that the Tutte Condition holds for G, which has order $n > 2$, and that the theorem holds for all graphs of smaller order. First note that $q(G - v) = 1 = |\{v\}|$, and so we may pick S_0 to be a maximal subset of $V(G)$ such that $q(G - S_0) = |S_0|$. Let $Q_1, Q_2, \ldots, Q_m$ be the odd components of $G - S_0$, and let $D_1, D_2, \ldots, D_k$ be the even components of $G - S_0$. We will show that:

(1) each D_j has a perfect matching;

(2) if $v \in V(Q_i)$, then $Q_i - v$ has a perfect matching; and

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G-S) \le |S|$ for every $S \subseteq V(G)$.
We assume that the condition holds and produce a perfect matching. We proceed by induction on the order of G. The claim is trivial if $|G| \le 2$.

Now suppose that the Tutte Condition holds for G, which has order $n > 2$, and that the theorem holds for all graphs of smaller order. First note that $q(G-v) = 1 = |\{v\}|$, and so we may pick S_0 to be a maximal subset of $V(G)$ such that $q(G-S_0) = |S_0|$. Let Q_1, Q_2, ..., Q_m be the odd components of $G-S_0$, and let D_1, D_2, ..., D_k be the even components of $G-S_0$. We will show that:

(1) each D_j has a perfect matching;

(2) if $v \in V(Q_i)$, then $Q_i - v$ has a perfect matching; and

(3) G contains a set s_1v_1, s_2v_2, ..., s_mv_m of edges such that $S_0 = \{s_1, s_2, \ldots, s_m\}$ and $v_i \in V(Q_i)$ for all i.

Proof of Tutte's 1-Factor Theorem

Recall the Tutte Condition: $q(G - S) \leq |S|$ for every $S \subseteq V(G)$.
We assume that the condition holds and produce a perfect matching. We proceed by induction on the order of G. The claim is trivial if $|G| \leq 2$.

Now suppose that the Tutte Condition holds for G, which has order $n > 2$, and that the theorem holds for all graphs of smaller order. First note that $q(G - v) = 1 = |\{v\}|$, and so we may pick S_0 to be a maximal subset of $V(G)$ such that $q(G - S_0) = |S_0|$. Let $Q_1, Q_2, \dots, Q_m$ be the odd components of $G - S_0$, and let $D_1, D_2, \dots, D_k$ be the even components of $G - S_0$. We will show that:

(1) each D_j has a perfect matching;

(2) if $v \in V(Q_i)$, then $Q_i - v$ has a perfect matching; and

(3) G contains a set $s_1v_1, s_2v_2, \dots, s_mv_m$ of edges such that $S_0 = \{s_1, s_2, \dots, s_m\}$ and $v_i \in V(Q_i)$ for all i.

Note that after (1)–(3) are established, the proof is complete.

Proof of Tutte's 1-Factor Theorem, Continued

To prove (1), which says that every D_j has a perfect matching, we want to apply the induction hypothesis,

Proof of Tutte's 1-Factor Theorem, Continued

To prove (1), which says that every D_j has a perfect matching, we want to apply the induction hypothesis, and so we need to verify that every D_j satisfies the Tutte Condition.

Proof of Tutte's 1-Factor Theorem, Continued

To prove (1), which says that every D_j has a perfect matching, we want to apply the induction hypothesis, and so we need to verify that every D_j satisfies the Tutte Condition. Suppose $S \subseteq V(D_j)$.

Proof of Tutte's 1-Factor Theorem, Continued

To prove (1), which says that every D_j has a perfect matching, we want to apply the induction hypothesis, and so we need to verify that every D_j satisfies the Tutte Condition. Suppose $S \subseteq V(D_j)$. Since the Tutte Condition holds for G, we have $q(G - (S \cup S_0)) \leq |S \cup S_0| = |S| + |S_0|$.

Proof of Tutte's 1-Factor Theorem, Continued

To prove (1), which says that every D_j has a perfect matching, we want to apply the induction hypothesis, and so we need to verify that every D_j satisfies the Tutte Condition. Suppose $S \subseteq V(D_j)$. Since the Tutte Condition holds for G, we have $q(G - (S \cup S_0)) \leq |S \cup S_0| = |S| + |S_0|$.
To count the odd components of $G - (S \cup S_0) = (G - S_0) - S$, note that when S is deleted from $G - S_0$, none of the Q_i's is affected,

Proof of Tutte's 1-Factor Theorem, Continued

To prove (1), which says that every D_j has a perfect matching, we want to apply the induction hypothesis, and so we need to verify that every D_j satisfies the Tutte Condition. Suppose $S \subseteq V(D_j)$. Since the Tutte Condition holds for G, we have $q(G - (S \cup S_0)) \leq |S \cup S_0| = |S| + |S_0|$.
To count the odd components of $G - (S \cup S_0) = (G - S_0) - S$, note that when S is deleted from $G - S_0$, none of the Q_i's is affected, and so $q(G - (S \cup S_0)) = q(G - S_0) + q(D_j - S)$

Proof of Tutte's 1-Factor Theorem, Continued

To prove (1), which says that every D_j has a perfect matching, we want to apply the induction hypothesis, and so we need to verify that every D_j satisfies the Tutte Condition. Suppose $S \subseteq V(D_j)$. Since the Tutte Condition holds for G, we have $q(G-(S \cup S_0)) \leq |S \cup S_0| = |S| + |S_0|$.
To count the odd components of $G-(S \cup S_0) = (G-S_0)-S$, note that when S is deleted from $G-S_0$, none of the Q_i's is affected, and so $q(G-(S \cup S_0)) = q(G-S_0) + q(D_j - S) = |S_0| + q(D_j - S)$.

Proof of Tutte's 1-Factor Theorem, Continued

To prove (1), which says that every D_j has a perfect matching, we want to apply the induction hypothesis, and so we need to verify that every D_j satisfies the Tutte Condition. Suppose $S \subseteq V(D_j)$. Since the Tutte Condition holds for G, we have $q(G-(S \cup S_0)) \leq |S \cup S_0| = |S| + |S_0|$.
To count the odd components of $G-(S \cup S_0) = (G-S_0)-S$, note that when S is deleted from $G-S_0$, none of the Q_i's is affected, and so $q(G-(S \cup S_0)) = q(G-S_0) + q(D_j - S) = |S_0| + q(D_j - S)$. Combining the previous inequality with the last equation, we get
$|S_0| + q(D_j - S) \leq |S| + |S_0|$,

Proof of Tutte's 1-Factor Theorem, Continued

To prove (1), which says that every D_j has a perfect matching, we want to apply the induction hypothesis, and so we need to verify that every D_j satisfies the Tutte Condition. Suppose $S \subseteq V(D_j)$. Since the Tutte Condition holds for G, we have $q(G-(S \cup S_0)) \leq |S \cup S_0| = |S| + |S_0|$.
To count the odd components of $G-(S \cup S_0) = (G-S_0)-S$, note that when S is deleted from $G-S_0$, none of the Q_i's is affected, and so $q(G-(S \cup S_0)) = q(G-S_0) + q(D_j - S) = |S_0| + q(D_j - S)$. Combining the previous inequality with the last equation, we get $|S_0| + q(D_j - S) \leq |S| + |S_0|$, and so $q(D_j - S) \leq |S|$, which means that Tutte Condition holds for D_j, as required.

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching.

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$,

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$.

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Reducing this equation modulo 2, gives

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Reducing this equation modulo 2, gives $1 \equiv |S| + 1 + q(Q_i - v - S) \pmod 2$,

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Reducing this equation modulo 2, gives $1 \equiv |S| + 1 + q(Q_i - v - S) \pmod 2$, and thus $q(Q_i - v - S) \equiv |S| \pmod 2$,

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Reducing this equation modulo 2, gives $1 \equiv |S| + 1 + q(Q_i - v - S) \pmod 2$, and thus $q(Q_i - v - S) \equiv |S| \pmod 2$, and so $q(Q_i - v - S) \geq |S| + 2$.

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Reducing this equation modulo 2, gives $1 \equiv |S| + 1 + q(Q_i - v - S) \pmod 2$, and thus $q(Q_i - v - S) \equiv |S| \pmod 2$, and so $q(Q_i - v - S) \geq |S| + 2$. Now notice that upon deleting $\{v\} \cup S$ from $G - S_0$ the only component of $G - S_0$ that is affected is Q_i, which is lost, and the number of new odd components formed is $q(Q_i - v - S)$.

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Reducing this equation modulo 2, gives $1 \equiv |S| + 1 + q(Q_i - v - S) \pmod 2$, and thus $q(Q_i - v - S) \equiv |S| \pmod 2$, and so $q(Q_i - v - S) \geq |S| + 2$. Now notice that upon deleting $\{v\} \cup S$ from $G - S_0$ the only component of $G - S_0$ that is affected is Q_i, which is lost, and the number of new odd components formed is $q(Q_i - v - S)$. Hence $q(G - S_0 - v - S) = q(G - S_0) - 1 + q(Q_i - v - S)$.

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Reducing this equation modulo 2, gives $1 \equiv |S| + 1 + q(Q_i - v - S) \pmod 2$, and thus $q(Q_i - v - S) \equiv |S| \pmod 2$, and so $q(Q_i - v - S) \geq |S| + 2$. Now notice that upon deleting $\{v\} \cup S$ from $G - S_0$ the only component of $G - S_0$ that is affected is Q_i, which is lost, and the number of new odd components formed is $q(Q_i - v - S)$. Hence $q(G - S_0 - v - S) = q(G - S_0) - 1 + q(Q_i - v - S)$. Now, since G satisfies the Tutte Condition for $S_0 \cup \{v\} \cup S$, we have $|S_0| + 1 + |S| \geq q(G - S_0 - v - S)$

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Reducing this equation modulo 2, gives $1 \equiv |S| + 1 + q(Q_i - v - S) \pmod 2$, and thus $q(Q_i - v - S) \equiv |S| \pmod 2$, and so $q(Q_i - v - S) \geq |S| + 2$. Now notice that upon deleting $\{v\} \cup S$ from $G - S_0$ the only component of $G - S_0$ that is affected is Q_i, which is lost, and the number of new odd components formed is $q(Q_i - v - S)$. Hence $q(G - S_0 - v - S) = q(G - S_0) - 1 + q(Q_i - v - S)$. Now, since G satisfies the Tutte Condition for $S_0 \cup \{v\} \cup S$, we have $|S_0| + 1 + |S| \geq q(G - S_0 - v - S)$ $= q(G - S_0) - 1 + q(Q_i - v - S)$

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Reducing this equation modulo 2, gives $1 \equiv |S| + 1 + q(Q_i - v - S) \pmod 2$, and thus $q(Q_i - v - S) \equiv |S| \pmod 2$, and so $q(Q_i - v - S) \geq |S| + 2$. Now notice that upon deleting $\{v\} \cup S$ from $G - S_0$ the only component of $G - S_0$ that is affected is Q_i, which is lost, and the number of new odd components formed is $q(Q_i - v - S)$. Hence $q(G - S_0 - v - S) = q(G - S_0) - 1 + q(Q_i - v - S)$. Now, since G satisfies the Tutte Condition for $S_0 \cup \{v\} \cup S$, we have $|S_0| + 1 + |S| \geq q(G - S_0 - v - S)$ $= q(G - S_0) - 1 + q(Q_i - v - S) \geq |S_0| - 1 + |S| + 2$.

Proof of Tutte's 1-Factor Theorem, Continued

Now, we prove (2), which states that each $Q_i - v$ has a perfect matching. Let $v \in V(Q_i)$ and suppose that the Tutte Conditions fails for $Q_i - v$, that is, there is a set $S \subseteq Q_i - v$ such that $q(Q_i - v - S) > |S|$. Now,

$$|V(Q_i)| = |S \cup \{v\}| + \sum_{\substack{\text{even components} \\ B_t \text{ of } Q_i - v - S}} |V(B_t)| + \sum_{\substack{\text{odd components} \\ R_s \text{ of } Q_i - v - S}} |V(R_s)|.$$

Reducing this equation modulo 2, gives $1 \equiv |S| + 1 + q(Q_i - v - S) \pmod 2$, and thus $q(Q_i - v - S) \equiv |S| \pmod 2$, and so $q(Q_i - v - S) \geq |S| + 2$. Now notice that upon deleting $\{v\} \cup S$ from $G - S_0$ the only component of $G - S_0$ that is affected is Q_i, which is lost, and the number of new odd components formed is $q(Q_i - v - S)$. Hence $q(G - S_0 - v - S) = q(G - S_0) - 1 + q(Q_i - v - S)$. Now, since G satisfies the Tutte Condition for $S_0 \cup \{v\} \cup S$, we have $|S_0| + 1 + |S| \geq q(G - S_0 - v - S)$ $= q(G - S_0) - 1 + q(Q_i - v - S) \geq |S_0| - 1 + |S| + 2$. But that implies that $q(G - S_0 - v - S) = |S_0 \cup \{v\} \cup S|$, which contradicts the maximality if S_0, and so (2) follows.

Proof of Tutte's 1-Factor Theorem, Continued

Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, ..., s_mv_m of edges such that $S_0 = \{s_1, s_2, \ldots, s_m\}$ and $v_i \in V(Q_i)$ for all i.

Proof of Tutte's 1-Factor Theorem, Continued

Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, . . . , s_mv_m of edges such that $S_0 = \{s_1, s_2, \ldots, s_m\}$ and $v_i \in V(Q_i)$ for all i. For that, we form a bipartite graph H with $X = \{Q_1, Q_2, \ldots, Q_m\}$ and $Y = S_0$,

Proof of Tutte's 1-Factor Theorem, Continued

Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, ..., s_mv_m of edges such that $S_0 = \{s_1, s_2, \ldots, s_m\}$ and $v_i \in V(Q_i)$ for all i. For that, we form a bipartite graph H with $X = \{Q_1, Q_2, \ldots, Q_m\}$ and $Y = S_0$, in which Q_i is joined to a vertex s_j in S_0 if and only if G has an edge from s_j to Q_i.

Proof of Tutte's 1-Factor Theorem, Continued

Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, ..., s_mv_m of edges such that $S_0 = \{s_1, s_2, \ldots, s_m\}$ and $v_i \in V(Q_i)$ for all i. For that, we form a bipartite graph H with $X = \{Q_1, Q_2, \ldots, Q_m\}$ and $Y = S_0$, in which Q_i is joined to a vertex s_j in S_0 if and only if G has an edge from s_j to Q_i. To prove (3), we need to show that H has a perfect matching.

Proof of Tutte's 1-Factor Theorem, Continued

Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, ..., s_mv_m of edges such that $S_0 = \{s_1, s_2, \ldots, s_m\}$ and $v_i \in V(Q_i)$ for all i. For that, we form a bipartite graph H with $X = \{Q_1, Q_2, \ldots, Q_m\}$ and $Y = S_0$, in which Q_i is joined to a vertex s_j in S_0 if and only if G has an edge from s_j to Q_i. To prove (3), we need to show that H has a perfect matching. We need to check that H satisfies the Hall Condition.

Proof of Tutte's 1-Factor Theorem, Continued

Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, . . . , s_mv_m of edges such that $S_0 = \{s_1, s_2, \ldots, s_m\}$ and $v_i \in V(Q_i)$ for all i. For that, we form a bipartite graph H with $X = \{Q_1, Q_2, \ldots, Q_m\}$ and $Y = S_0$, in which Q_i is joined to a vertex s_j in S_0 if and only if G has an edge from s_j to Q_i. To prove (3), we need to show that H has a perfect matching. We need to check that H satisfies the Hall Condition. Let $A \subseteq X$.

Proof of Tutte's 1-Factor Theorem, Continued

Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, ..., s_mv_m of edges such that $S_0 = \{s_1, s_2, \ldots, s_m\}$ and $v_i \in V(Q_i)$ for all i. For that, we form a bipartite graph H with $X = \{Q_1, Q_2, \ldots, Q_m\}$ and $Y = S_0$, in which Q_i is joined to a vertex s_j in S_0 if and only if G has an edge from s_j to Q_i. To prove (3), we need to show that H has a perfect matching. We need to check that H satisfies the Hall Condition. Let $A \subseteq X$. But $N_H(A)$ is also a set of vertices of G, so G satisfies the Tutte Condition for $N_H(A)$, Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, ..., s_mv_m of edges such that $S_0 = \{s_1, s_2, \dots, s_m\}$ and $v_i \in V(Q_i)$ for all i. For that, we form a bipartite graph H with $X = \{Q_1, Q_2, \dots, Q_m\}$ and $Y = S_0$, in which Q_i is joined to a vertex s_j in S_0 if and only if G has an edge from s_j to Q_i. To prove (3), we need to show that H has a perfect matching. We need to check that H satisfies the Hall Condition. Let $A \subseteq X$. But $N_H(A)$ is also a set of vertices of G, so G satisfies the Tutte Condition for $N_H(A)$, that is, $q(G - N_H(A)) \leq |N_H(A)|$.

Proof of Tutte's 1-Factor Theorem, Continued

Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, . . . , s_mv_m of edges such that $S_0 = \{s_1, s_2, \ldots, s_m\}$ and $v_i \in V(Q_i)$ for all i. For that, we form a bipartite graph H with $X = \{Q_1, Q_2, \ldots, Q_m\}$ and $Y = S_0$, in which Q_i is joined to a vertex s_j in S_0 if and only if G has an edge from s_j to Q_i. To prove (3), we need to show that H has a perfect matching. We need to check that H satisfies the Hall Condition. Let $A \subseteq X$. But $N_H(A)$ is also a set of vertices of G, so G satisfies the Tutte Condition for $N_H(A)$, that is, $q(G - N_H(A)) \leq |N_H(A)|$. But every odd component Q of $G - S_0$ that is in A is also a component of $G - N_H(A)$.

Proof of Tutte's 1-Factor Theorem, Continued

Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, ..., s_mv_m of edges such that $S_0 = \{s_1, s_2, \ldots, s_m\}$ and $v_i \in V(Q_i)$ for all i. For that, we form a bipartite graph H with $X = \{Q_1, Q_2, \ldots, Q_m\}$ and $Y = S_0$, in which Q_i is joined to a vertex s_j in S_0 if and only if G has an edge from s_j to Q_i. To prove (3), we need to show that H has a perfect matching. We need to check that H satisfies the Hall Condition. Let $A \subseteq X$. But $N_H(A)$ is also a set of vertices of G, so G satisfies the Tutte Condition for $N_H(A)$, that is, $q(G - N_H(A)) \leq |N_H(A)|$. But every odd component Q of $G - S_0$ that is in A is also a component of $G - N_H(A)$. Thus $q(G - N_H(A)) \geq |A|$, and so $|N_H(A)| \geq |A|$, as required.

Proof of Tutte's 1-Factor Theorem, Continued

Now we turn to (3), which states that G contains a set s_1v_1, s_2v_2, ..., s_mv_m of edges such that $S_0 = \{s_1, s_2, \dots, s_m\}$ and $v_i \in V(Q_i)$ for all i. For that, we form a bipartite graph H with $X = \{Q_1, Q_2, \dots, Q_m\}$ and $Y = S_0$, in which Q_i is joined to a vertex s_j in S_0 if and only if G has an edge from s_j to Q_i. To prove (3), we need to show that H has a perfect matching. We need to check that H satisfies the Hall Condition. Let $A \subseteq X$. But $N_H(A)$ is also a set of vertices of G, so G satisfies the Tutte Condition for $N_H(A)$, that is, $q(G - N_H(A)) \leq |N_H(A)|$. But every odd component Q of $G - S_0$ that is in A is also a component of $G - N_H(A)$. Thus $q(G - N_H(A)) \geq |A|$, and so $|N_H(A)| \geq |A|$, as required. Hence H has a perfect matching, and hence (3) is proved, and so is Tutte's 1-Factor Theorem.

Homework Set 2

Problem 4

Derive the sufficiency (the non-obvious direction) of the Hall's Marriage Theorem from the Tutte's 1*-Factor Theorem.*

Homework Set 2

Problem 4

Derive the sufficiency (the non-obvious direction) of the Hall's Marriage Theorem from the Tutte's 1-Factor Theorem.

Problem 5

Prove that a 3-regular simple graph has a 1-factor if and only if it decomposes into copies of P_4.

Homework Set 2

Problem 4

Derive the sufficiency (the non-obvious direction) of the Hall's Marriage Theorem from the Tutte's 1*-Factor Theorem.*

Problem 5

Prove that a 3*-regular simple graph has a* 1*-factor if and only if it decomposes into copies of* P_4*.*

Problem 6

Prove that a tree T *has a perfect matching if and only if* $q(T - v) = 1$ *for every* $v \in V(T)$*. Do not invoke Tutte's* 1*-Factor Theorem.*

Petersen's Theorem

Definition 3.17

A an edge e of G is a cut-edge if $G \setminus e$ has more connected components than G.

Petersen's Theorem

Definition 3.17

A an edge e of G is a cut-edge if $G \setminus e$ has more connected components than G.

Corollary 3.18 (Petersen 1891)

Every simple 3-regular graph with no cut-edge has a perfect matching.

Petersen's Theorem

Definition 3.17

A an edge e of G is a cut-edge if $G \setminus e$ has more connected components than G.

Corollary 3.18 (Petersen 1891)

Every simple 3-regular graph with no cut-edge has a perfect matching.

Proof.

We prove that G satisfies the Tutte Condition.

$\square$

Definition 3.17

A an edge e of G is a cut-edge if $G \setminus e$ has more connected components than G.

Corollary 3.18 (Petersen 1891)

Every simple 3-regular graph with no cut-edge has a perfect matching.

Proof.

We prove that G satisfies the Tutte Condition. Let $S \subseteq V(G)$, and count the edges between S and the odd components of $G - S$.

$\square$

Petersen's Theorem

Definition 3.17

A an edge e of G is a cut-edge if $G \setminus e$ has more connected components than G.

Corollary 3.18 (Petersen 1891)

Every simple 3-regular graph with no cut-edge has a perfect matching.

Proof.

We prove that G satisfies the Tutte Condition. Let $S \subseteq V(G)$, and count the edges between S and the odd components of $G - S$. Since G is 3-regular, every vertex in S is incident to at most three such edges.

□

Definition 3.17

A an edge e of G is a cut-edge if $G \setminus e$ has more connected components than G.

Corollary 3.18 (Petersen 1891)

Every simple 3-regular graph with no cut-edge has a perfect matching.

Proof.

We prove that G satisfies the Tutte Condition. Let $S \subseteq V(G)$, and count the edges between S and the odd components of $G - S$. Since G is 3-regular, every vertex in S is incident to at most three such edges. If each odd component H of $G - S$ is incident to at least three such edges, then $3q(G - S) \leq 3|S|$, and the Tutte Condition holds.

$\square$

Petersen's Theorem

Definition 3.17

A an edge e of G is a cut-edge if $G \setminus e$ has more connected components than G.

Corollary 3.18 (Petersen 1891)

Every simple 3-regular graph with no cut-edge has a perfect matching.

Proof.

We prove that G satisfies the Tutte Condition. Let $S \subseteq V(G)$, and count the edges between S and the odd components of $G - S$. Since G is 3-regular, every vertex in S is incident to at most three such edges. If each odd component H of $G - S$ is incident to at least three such edges, then $3q(G - S) \leq 3|S|$, and the Tutte Condition holds. Let m be the number of edges from S to H.

$\square$

Petersen's Theorem

Definition 3.17

A an edge e of G is a cut-edge if $G \setminus e$ has more connected components than G.

Corollary 3.18 (Petersen 1891)

Every simple 3-regular graph with no cut-edge has a perfect matching.

Proof.

We prove that G satisfies the Tutte Condition. Let $S \subseteq V(G)$, and count the edges between S and the odd components of $G - S$. Since G is 3-regular, every vertex in S is incident to at most three such edges. If each odd component H of $G - S$ is incident to at least three such edges, then $3q(G - S) \leq 3|S|$, and the Tutte Condition holds. Let m be the number of edges from S to H. The sum of vertex degrees in H is $3|H| - m$, which must be even.

□

Definition 3.17

A an edge e of G is a cut-edge if $G \setminus e$ has more connected components than G.

Corollary 3.18 (Petersen 1891)

Every simple 3-regular graph with no cut-edge has a perfect matching.

Proof.

We prove that G satisfies the Tutte Condition. Let $S \subseteq V(G)$, and count the edges between S and the odd components of $G - S$. Since G is 3-regular, every vertex in S is incident to at most three such edges. If each odd component H of $G - S$ is incident to at least three such edges, then $3q(G - S) \leq 3|S|$, and the Tutte Condition holds. Let m be the number of edges from S to H. The sum of vertex degrees in H is $3|H| - m$, which must be even. Since $|H|$ is odd, m must be also odd, but it cannot be 1 since G would have a cut-edge.

□

Petersen's Theorem

Definition 3.17

A an edge e of G is a cut-edge if $G \setminus e$ has more connected components than G.

Corollary 3.18 (Petersen 1891)

Every simple 3-regular graph with no cut-edge has a perfect matching.

Proof.

We prove that G satisfies the Tutte Condition. Let $S \subseteq V(G)$, and count the edges between S and the odd components of $G - S$. Since G is 3-regular, every vertex in S is incident to at most three such edges. If each odd component H of $G - S$ is incident to at least three such edges, then $3q(G - S) \leq 3|S|$, and the Tutte Condition holds. Let m be the number of edges from S to H. The sum of vertex degrees in H is $3|H| - m$, which must be even. Since $|H|$ is odd, m must be also odd, but it cannot be 1 since G would have a cut-edge. Thus m must be at least 3 and the Tutte Condition holds.

□

Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.

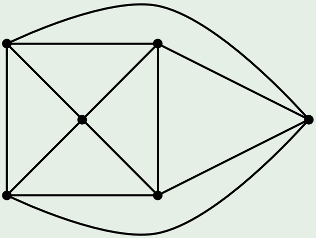
Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

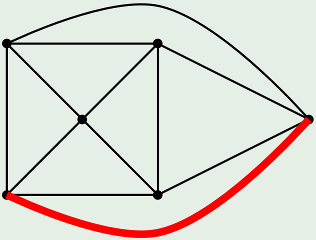
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

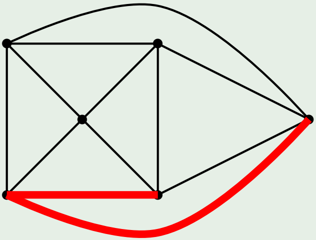
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

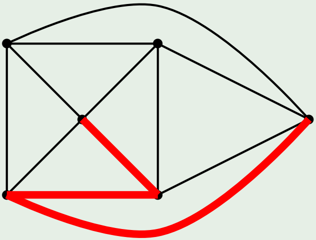
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

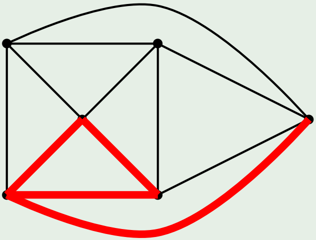
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

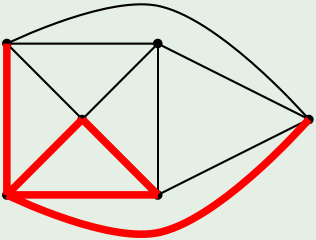
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

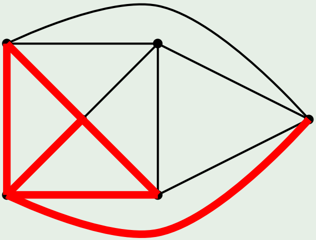
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

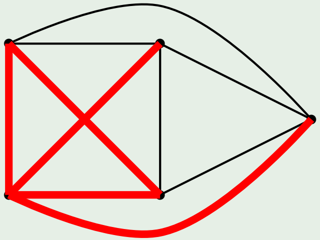
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

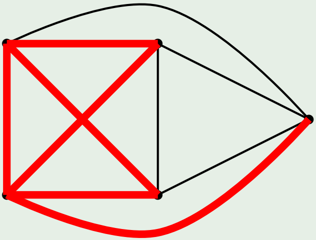
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

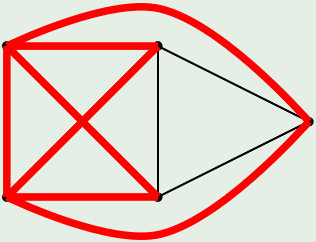
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

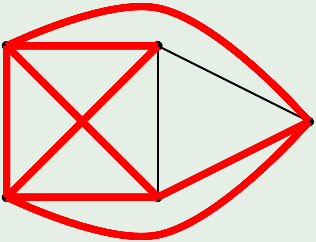
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

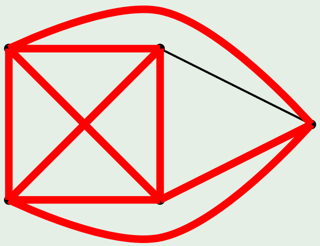
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

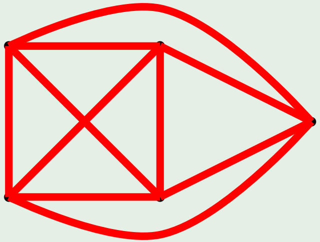
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

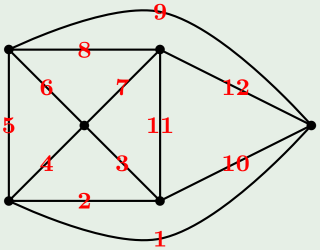
Example 4.2



Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

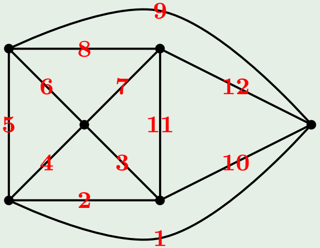
Example 4.2

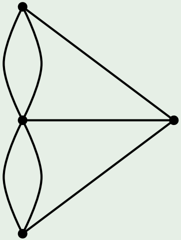


Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

Example 4.2

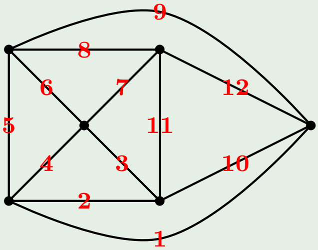


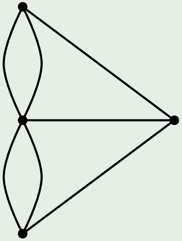


Definition 4.1

- A closed trail that uses every edge of the graph is called an Euler tour.
- A graph is Eulerian if it has an Euler tour.

Example 4.2





Not Eulerian!

Characterization of Eulerian Graphs

Theorem 4.3 (Euler 1873)

A graph is Eulerian if and only if all its vertices have even degrees

Characterization of Eulerian Graphs

Theorem 4.3 (Euler 1873)

A graph is Eulerian if and only if all its vertices have even degrees and all of its edges belong to a single component.

Characterization of Eulerian Graphs

Theorem 4.3 (Euler 1873)

A graph is Eulerian if and only if all its vertices have even degrees and all of its edges belong to a single component.

Lemma 4.4

Non-trivial maximal trails in graphs with all degrees even are closed.

Theorem 4.3 (Euler 1873)

A graph is Eulerian if and only if all its vertices have even degrees and all of its edges belong to a single component.

Lemma 4.4

Non-trivial maximal trails in graphs with all degrees even are closed.

Proof.

Let T be a maximal non-trivial trail in some graph G with all degrees even.

□

Characterization of Eulerian Graphs

Theorem 4.3 (Euler 1873)

A graph is Eulerian if and only if all its vertices have even degrees and all of its edges belong to a single component.

Lemma 4.4

Non-trivial maximal trails in graphs with all degrees even are closed.

Proof.

Let T be a maximal non-trivial trail in some graph G with all degrees even. Since T is maximal, it includes all edges of G incident with its final vertex v.

$\square$

Characterization of Eulerian Graphs

Theorem 4.3 (Euler 1873)

A graph is Eulerian if and only if all its vertices have even degrees and all of its edges belong to a single component.

Lemma 4.4

Non-trivial maximal trails in graphs with all degrees even are closed.

Proof.

Let T be a maximal non-trivial trail in some graph G with all degrees even. Since T is maximal, it includes all edges of G incident with its final vertex v. If T is not closed, then the degree of v must be odd, which is impossible. $\square$

Proof of Euler's Theorem

Necessity is clear.

Proof of Euler's Theorem

Necessity is clear.
For sufficiency, suppose that G is non-trivial with all degrees even and all edges in same component.

Proof of Euler's Theorem

Necessity is clear.
For sufficiency, suppose that G is non-trivial with all degrees even and all edges in same component. Let T be a trail of maximum length.

Proof of Euler's Theorem

Necessity is clear.
For sufficiency, suppose that G is non-trivial with all degrees even and all edges in same component. Let T be a trail of maximum length. By Lemma 4.4, T is closed.

Proof of Euler's Theorem

Necessity is clear.
For sufficiency, suppose that G is non-trivial with all degrees even and all edges in same component. Let T be a trail of maximum length. By Lemma 4.4, T is closed. Let $G' = G \setminus E(T)$ and suppose G' is non-trivial.

Proof of Euler's Theorem

Necessity is clear.
For sufficiency, suppose that G is non-trivial with all degrees even and all edges in same component. Let T be a trail of maximum length. By Lemma 4.4, T is closed. Let $G' = G \setminus E(T)$ and suppose G' is non-trivial. Since the degree of every vertex in G and in T is even, so it is in G'.

Proof of Euler's Theorem

Necessity is clear.
For sufficiency, suppose that G is non-trivial with all degrees even and all edges in same component. Let T be a trail of maximum length. By Lemma 4.4, T is closed. Let $G' = G \setminus E(T)$ and suppose G' is non-trivial. Since the degree of every vertex in G and in T is even, so it is in G'. Since all edges of G lie in the same component, there is an edge e of G' adjacent to an edge in T.

Proof of Euler's Theorem

Necessity is clear.
For sufficiency, suppose that G is non-trivial with all degrees even and all edges in same component. Let T be a trail of maximum length. By Lemma 4.4, T is closed. Let $G' = G \setminus E(T)$ and suppose G' is non-trivial. Since the degree of every vertex in G and in T is even, so it is in G'. Since all edges of G lie in the same component, there is an edge e of G' adjacent to an edge in T. Let T' be a maximal trail in G' with e as its first edge.

Proof of Euler's Theorem

Necessity is clear.
For sufficiency, suppose that G is non-trivial with all degrees even and all edges in same component. Let T be a trail of maximum length. By Lemma 4.4, T is closed. Let $G' = G \setminus E(T)$ and suppose G' is non-trivial. Since the degree of every vertex in G and in T is even, so it is in G'. Since all edges of G lie in the same component, there is an edge e of G' adjacent to an edge in T. Let T' be a maximal trail in G' with e as its first edge. Again by Lemma 4.4, T' is closed.

Proof of Euler's Theorem

Necessity is clear.
For sufficiency, suppose that G is non-trivial with all degrees even and all edges in same component. Let T be a trail of maximum length. By Lemma 4.4, T is closed. Let $G' = G \setminus E(T)$ and suppose G' is non-trivial. Since the degree of every vertex in G and in T is even, so it is in G'. Since all edges of G lie in the same component, there is an edge e of G' adjacent to an edge in T. Let T' be a maximal trail in G' with e as its first edge. Again by Lemma 4.4, T' is closed. Hence we may detour T along T' to produce a longer trail;

Proof of Euler's Theorem

Necessity is clear.

For sufficiency, suppose that G is non-trivial with all degrees even and all edges in same component. Let T be a trail of maximum length. By Lemma 4.4, T is closed. Let $G' = G \setminus E(T)$ and suppose G' is non-trivial. Since the degree of every vertex in G and in T is even, so it is in G'. Since all edges of G lie in the same component, there is an edge e of G' adjacent to an edge in T. Let T' be a maximal trail in G' with e as its first edge. Again by Lemma 4.4, T' is closed. Hence we may detour T along T' to produce a longer trail; a contradiction.

Vertex Connectivity

Definition 5.1

- A separating set or a vertex cut of a graph G is a set $S \subseteq V(G)$ such that $G - S$ has more than one component.

Vertex Connectivity

Definition 5.1

- A separating set or a vertex cut of a graph G is a set $S \subseteq V(G)$ such that $G - S$ has more than one component.
- Vertex connectivity or connectivity $\kappa(G)$ of a graph G is defined as follows:

Vertex Connectivity

Definition 5.1

- A separating set or a vertex cut of a graph G is a set $S \subseteq V(G)$ such that $G - S$ has more than one component.
- Vertex connectivity or connectivity $\kappa(G)$ of a graph G is defined as follows:
  - $\kappa(G) = 0$ if G is disconnected;

Vertex Connectivity

Definition 5.1

- A separating set or a vertex cut of a graph G is a set $S \subseteq V(G)$ such that $G - S$ has more than one component.
- Vertex connectivity or connectivity $\kappa(G)$ of a graph G is defined as follows:
  - $\kappa(G) = 0$ if G is disconnected;
  - $\kappa(G) = |G| - 1$ if G is connected, but has no pair of distinct non-adjacent vertices.

Vertex Connectivity

Definition 5.1

- A separating set or a vertex cut of a graph G is a set $S \subseteq V(G)$ such that $G - S$ has more than one component.
- Vertex connectivity or connectivity $\kappa(G)$ of a graph G is defined as follows:
  - $\kappa(G) = 0$ if G is disconnected;
  - $\kappa(G) = |G| - 1$ if G is connected, but has no pair of distinct non-adjacent vertices.
  - $\kappa(G) = j$ if G is connected, but has a pair of non-adjacent vertices, and j is the smallest integer such that G has a j-element vertex cut.

Vertex Connectivity

Definition 5.1

- A separating set or a vertex cut of a graph G is a set $S \subseteq V(G)$ such that $G - S$ has more than one component.
- Vertex connectivity or connectivity $\kappa(G)$ of a graph G is defined as follows:
  - $\kappa(G) = 0$ if G is disconnected;
  - $\kappa(G) = |G| - 1$ if G is connected, but has no pair of distinct non-adjacent vertices.
  - $\kappa(G) = j$ if G is connected, but has a pair of non-adjacent vertices, and j is the smallest integer such that G has a j-element vertex cut.
- If k is a positive integer, then G is k-connected or k-vertex-connected if $k \leq \kappa(G)$.

Vertex Connectivity

Definition 5.1

- A separating set or a vertex cut of a graph G is a set $S \subseteq V(G)$ such that $G - S$ has more than one component.
- Vertex connectivity or connectivity $\kappa(G)$ of a graph G is defined as follows:
  - $\kappa(G) = 0$ if G is disconnected;
  - $\kappa(G) = |G| - 1$ if G is connected, but has no pair of distinct non-adjacent vertices.
  - $\kappa(G) = j$ if G is connected, but has a pair of non-adjacent vertices, and j is the smallest integer such that G has a j-element vertex cut.
- If k is a positive integer, then G is k-connected or k-vertex-connected if $k \leq \kappa(G)$.

Note 5.2

- *Vertex connectivity is not affected by adding or deleting loops and parallel edges.*

Vertex Connectivity

Definition 5.1

- A separating set or a vertex cut of a graph G is a set $S \subseteq V(G)$ such that $G - S$ has more than one component.
- Vertex connectivity or connectivity $\kappa(G)$ of a graph G is defined as follows:
  - $\kappa(G) = 0$ if G is disconnected;
  - $\kappa(G) = |G| - 1$ if G is connected, but has no pair of distinct non-adjacent vertices.
  - $\kappa(G) = j$ if G is connected, but has a pair of non-adjacent vertices, and j is the smallest integer such that G has a j-element vertex cut.
- If k is a positive integer, then G is k-connected or k-vertex-connected if $k \leq \kappa(G)$.

Note 5.2

- *Vertex connectivity is not affected by adding or deleting loops and parallel edges.*
- K_1 *is connected although* $\kappa(K_1) = 0$.

Connectivity Examples

Example 5.3

- $\kappa(K_n) = n - 1$ for $n \geq 2$;

Connectivity Examples

Example 5.3

- $\kappa(K_n) = n - 1$ for $n \geq 2$;
- $\kappa(K_{m,n}) = \min(m, n)$;

Connectivity Examples

Example 5.3

- $\kappa(K_n) = n - 1$ for $n \geq 2$;
- $\kappa(K_{m,n}) = \min(m, n)$;
- If T is a non-trivial tree, then $\kappa(T) = 1$.

Connectivity Examples

Example 5.3

- $\kappa(K_n) = n - 1$ for $n \geq 2$;
- $\kappa(K_{m,n}) = \min(m, n)$;
- If T is a non-trivial tree, then $\kappa(T) = 1$.
- $\kappa(C_n) = 2$ for all $n \geq 3$.

Connectivity Examples

Example 5.3

- $\kappa(K_n) = n - 1$ for $n \geq 2$;
- $\kappa(K_{m,n}) = \min(m, n)$;
- If T is a non-trivial tree, then $\kappa(T) = 1$.
- $\kappa(C_n) = 2$ for all $n \geq 3$.
- An n-wheel W_n is obtained from C_n by adding a new vertex and joining it to all vertices of C_n. If $n \geq 3$, then $\kappa(W_n) = 3$.

Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.

Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.

Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.

Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.
- Given $S, T \subseteq V(G)$, we write $[S, T]$ for the set of edges with one endpoint in S and the other in T.

Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.
- Given $S, T \subseteq V(G)$, we write $[S, T]$ for the set of edges with one endpoint in S and the other in T.
- An edge cut is a set of edges of the form $[S, \overline{S}]$ where S is a non-empty proper subset of $V(G)$.

Edge Connectivity

Definition 5.4

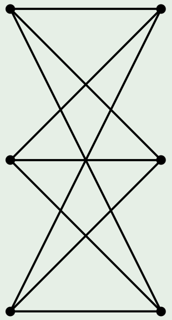
- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.
- Given $S, T \subseteq V(G)$, we write $[S, T]$ for the set of edges with one endpoint in S and the other in T.
- An edge cut is a set of edges of the form $[S, \overline{S}]$ where S is a non-empty proper subset of $V(G)$.
- A bond is a minimal non-empty edge cut.

Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.
- Given $S, T \subseteq V(G)$, we write $[S, T]$ for the set of edges with one endpoint in S and the other in T.
- An edge cut is a set of edges of the form $[S, \overline{S}]$ where S is a non-empty proper subset of $V(G)$.
- A bond is a minimal non-empty edge cut.

Example 5.5



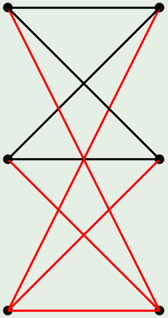
Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.
- Given $S, T \subseteq V(G)$, we write $[S, T]$ for the set of edges with one endpoint in S and the other in T.
- An edge cut is a set of edges of the form $[S, \overline{S}]$ where S is a non-empty proper subset of $V(G)$.
- A bond is a minimal non-empty edge cut.

Example 5.5

disconnecting set, but not an edge cut



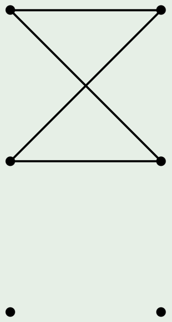
Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.
- Given $S, T \subseteq V(G)$, we write $[S, T]$ for the set of edges with one endpoint in S and the other in T.
- An edge cut is a set of edges of the form $[S, \overline{S}]$ where S is a non-empty proper subset of $V(G)$.
- A bond is a minimal non-empty edge cut.

Example 5.5

disconnecting set, but not an edge cut



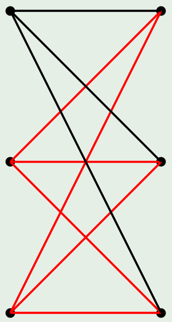
Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.
- Given $S, T \subseteq V(G)$, we write $[S, T]$ for the set of edges with one endpoint in S and the other in T.
- An edge cut is a set of edges of the form $[S, \overline{S}]$ where S is a non-empty proper subset of $V(G)$.
- A bond is a minimal non-empty edge cut.

Example 5.5

edge cut, but not a bond



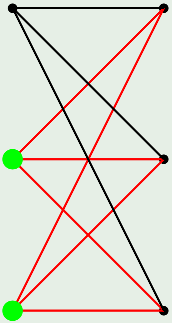
Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.
- Given $S, T \subseteq V(G)$, we write $[S, T]$ for the set of edges with one endpoint in S and the other in T.
- An edge cut is a set of edges of the form $[S, \overline{S}]$ where S is a non-empty proper subset of $V(G)$.
- A bond is a minimal non-empty edge cut.

Example 5.5

edge cut, but not a bond



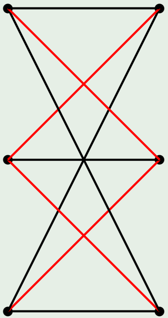
Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.
- Given $S, T \subseteq V(G)$, we write $[S, T]$ for the set of edges with one endpoint in S and the other in T.
- An edge cut is a set of edges of the form $[S, \overline{S}]$ where S is a non-empty proper subset of $V(G)$.
- A bond is a minimal non-empty edge cut.

Example 5.5

bond



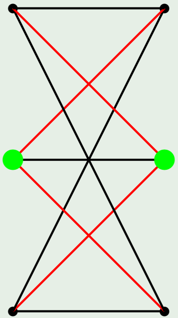
Edge Connectivity

Definition 5.4

- A disconnecting set of edges of a graph G with $|G| > 1$ is a set $F \subseteq E(G)$ such that $G \setminus F$ has more than one component.
- A graph is k-edge-connected if every disconnecting set has at least k edges.
- The edge connectivity of G, written $\kappa'(G)$ is the maximum k such that G is k-edge-connected.
- Given $S, T \subseteq V(G)$, we write $[S, T]$ for the set of edges with one endpoint in S and the other in T.
- An edge cut is a set of edges of the form $[S, \overline{S}]$ where S is a non-empty proper subset of $V(G)$.
- A bond is a minimal non-empty edge cut.

Example 5.5

bond



Whitney's Theorem

Note 5.6

The edge connectivity of a graph is unaffected by adding or deleting loops, but is affected by adding and deleting edges in parallel.

Whitney's Theorem

Note 5.6

The edge connectivity of a graph is unaffected by adding or deleting loops, but is affected by adding and deleting edges in parallel.

Theorem 5.7 (Whitney 1932)

If G is graph with $|G| > 1$, then $\kappa(G) \leq \kappa'(G) \leq \delta(G)$.

Whitney's Theorem

Note 5.6

The edge connectivity of a graph is unaffected by adding or deleting loops, but is affected by adding and deleting edges in parallel.

Theorem 5.7 (Whitney 1932)

If G is graph with $|G| > 1$, then $\kappa(G) \leq \kappa'(G) \leq \delta(G)$.

Proof.

The edges incident to a vertex form a disconnecting set, so $\kappa' \leq \delta$.

□

Whitney's Theorem

Note 5.6

The edge connectivity of a graph is unaffected by adding or deleting loops, but is affected by adding and deleting edges in parallel.

Theorem 5.7 (Whitney 1932)

If G is graph with $|G| > 1$, then $\kappa(G) \leq \kappa'(G) \leq \delta(G)$.

Proof.

The edges incident to a vertex form a disconnecting set, so $\kappa' \leq \delta$.
Clearly, $\kappa(G) \leq |G| - 1$.

$\square$

Whitney's Theorem

Note 5.6

The edge connectivity of a graph is unaffected by adding or deleting loops, but is affected by adding and deleting edges in parallel.

Theorem 5.7 (Whitney 1932)

If G is graph with $|G| > 1$, then $\kappa(G) \leq \kappa'(G) \leq \delta(G)$.

Proof.

The edges incident to a vertex form a disconnecting set, so $\kappa' \leq \delta$. Clearly, $\kappa(G) \leq |G| - 1$. Suppose $[S, \overline{S}]$ is a minimum edge cut of size $k' = \kappa'(G)$.

$\square$

Whitney's Theorem

Note 5.6

The edge connectivity of a graph is unaffected by adding or deleting loops, but is affected by adding and deleting edges in parallel.

Theorem 5.7 (Whitney 1932)

If G is graph with $|G| > 1$, then $\kappa(G) \leq \kappa'(G) \leq \delta(G)$.

Proof.

The edges incident to a vertex form a disconnecting set, so $\kappa' \leq \delta$. Clearly, $\kappa(G) \leq |G| - 1$. Suppose $[S, \overline{S}]$ is a minimum edge cut of size $k' = \kappa'(G)$. If every vertex in S is adjacent to every vertex in $\overline{S}$, then $k' = |S||\overline{S}| \geq |G| - 1$, and the inequality follows.

□

Whitney's Theorem

Note 5.6

The edge connectivity of a graph is unaffected by adding or deleting loops, but is affected by adding and deleting edges in parallel.

Theorem 5.7 (Whitney 1932)

If G is graph with $|G| > 1$, then $\kappa(G) \leq \kappa'(G) \leq \delta(G)$.

Proof.

The edges incident to a vertex form a disconnecting set, so $\kappa' \leq \delta$. Clearly, $\kappa(G) \leq |G| - 1$. Suppose $[S, \overline{S}]$ is a minimum edge cut of size $k' = \kappa'(G)$. If every vertex in S is adjacent to every vertex in $\overline{S}$, then $k' = |S||\overline{S}| \geq |G| - 1$, and the inequality follows. Hence we may assume that there are vertices $s \in S$ and $\overline{s} \in \overline{S}$ that are non-adjacent.

$\square$

Whitney's Theorem

Note 5.6

The edge connectivity of a graph is unaffected by adding or deleting loops, but is affected by adding and deleting edges in parallel.

Theorem 5.7 (Whitney 1932)

If G is graph with $|G| > 1$, then $\kappa(G) \leq \kappa'(G) \leq \delta(G)$.

Proof.

The edges incident to a vertex form a disconnecting set, so $\kappa' \leq \delta$. Clearly, $\kappa(G) \leq |G| - 1$. Suppose $[S, \overline{S}]$ is a minimum edge cut of size $k' = \kappa'(G)$. If every vertex in S is adjacent to every vertex in $\overline{S}$, then $k' = |S||\overline{S}| \geq |G| - 1$, and the inequality follows. Hence we may assume that there are vertices $s \in S$ and $\overline{s} \in \overline{S}$ that are non-adjacent. Let T be the vertex set consisting of all neighbors of s in $\overline{S}$ and all vertices in $S - s$ that have neighbors in $\overline{S}$.

□

Whitney's Theorem

Note 5.6

The edge connectivity of a graph is unaffected by adding or deleting loops, but is affected by adding and deleting edges in parallel.

Theorem 5.7 (Whitney 1932)

If G is graph with $|G| > 1$, then $\kappa(G) \leq \kappa'(G) \leq \delta(G)$.

Proof.

The edges incident to a vertex form a disconnecting set, so $\kappa' \leq \delta$. Clearly, $\kappa(G) \leq |G| - 1$. Suppose $[S, \overline{S}]$ is a minimum edge cut of size $k' = \kappa'(G)$. If every vertex in S is adjacent to every vertex in $\overline{S}$, then $k' = |S||\overline{S}| \geq |G| - 1$, and the inequality follows. Hence we may assume that there are vertices $s \in S$ and $\overline{s} \in \overline{S}$ that are non-adjacent. Let T be the vertex set consisting of all neighbors of s in $\overline{S}$ and all vertices in $S - s$ that have neighbors in $\overline{S}$. Then T is a vertex cut consisting of one endpoint of each edge in $[S, \overline{S}]$. $\square$

Whitney's Theorem

Note 5.6

The edge connectivity of a graph is unaffected by adding or deleting loops, but is affected by adding and deleting edges in parallel.

Theorem 5.7 (Whitney 1932)

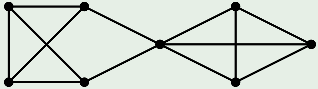
If G is graph with $|G| > 1$, then $\kappa(G) \leq \kappa'(G) \leq \delta(G)$.

Proof.

The edges incident to a vertex form a disconnecting set, so $\kappa' \leq \delta$. Clearly, $\kappa(G) \leq |G| - 1$. Suppose $[S, \overline{S}]$ is a minimum edge cut of size $k' = \kappa'(G)$. If every vertex in S is adjacent to every vertex in $\overline{S}$, then $k' = |S||\overline{S}| \geq |G| - 1$, and the inequality follows. Hence we may assume that there are vertices $s \in S$ and $\overline{s} \in \overline{S}$ that are non-adjacent. Let T be the vertex set consisting of all neighbors of s in $\overline{S}$ and all vertices in $S - s$ that have neighbors in $\overline{S}$. Then T is a vertex cut consisting of one endpoint of each edge in $[S, \overline{S}]$. Hence $\kappa \leq \kappa'$. □

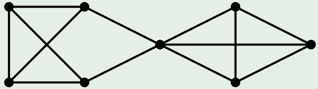
Connectivity Example

Example 5.8



Connectivity Example

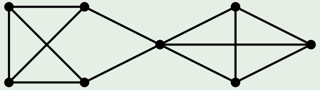
Example 5.8



$$\kappa = 1 < \kappa' = 2 < \delta = 3$$

Connectivity Example

Example 5.8



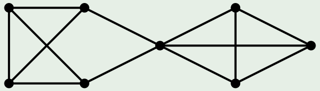
$$\kappa = 1 < \kappa' = 2 < \delta = 3$$

Theorem 5.9

If G is a connected graph and S is a non-empty proper subset of $V(G)$, then $F = [S, \overline{S}]$ is a bond if and only if $G \setminus F$ has two components.

Connectivity Example

Example 5.8



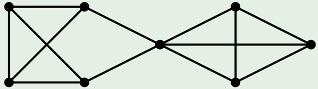
$\kappa = 1 < \kappa' = 2 < \delta = 3$

Theorem 5.9

If G is a connected graph and S is a non-empty proper subset of $V(G)$, then $F = [S, \overline{S}]$ is a bond if and only if $G \setminus F$ has two components. Equivalently, if and only if the subgraphs of G induced by each of S and $\overline{S}$ are connected.

Connectivity Example

Example 5.8



$\kappa = 1 < \kappa' = 2 < \delta = 3$

Theorem 5.9

If G is a connected graph and S is a non-empty proper subset of $V(G)$, then $F = [S, \overline{S}]$ is a bond if and only if $G \setminus F$ has two components. Equivalently, if and only if the subgraphs of G induced by each of S and $\overline{S}$ are connected.

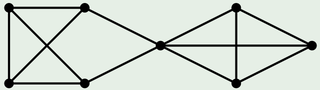
Proof.

If $G \setminus F$ has two components, then F is a bond, since $G \setminus F'$ is connected for every proper subset F' of F.

$\square$

Connectivity Example

Example 5.8



$$\kappa = 1 < \kappa' = 2 < \delta = 3$$

Theorem 5.9

If G is a connected graph and S is a non-empty proper subset of $V(G)$, then $F = [S, \overline{S}]$ is a bond if and only if $G \setminus F$ has two components. Equivalently, if and only if the subgraphs of G induced by each of S and $\overline{S}$ are connected.

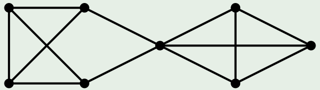
Proof.

If $G \setminus F$ has two components, then F is a bond, since $G \setminus F'$ is connected for every proper subset F' of F.
If $G \setminus F$ has more than two components, then we may assume $S = A \cup B$ with no edges between A and B.

$\square$

Connectivity Example

Example 5.8



$$\kappa = 1 < \kappa' = 2 < \delta = 3$$

Theorem 5.9

If G is a connected graph and S is a non-empty proper subset of $V(G)$, then $F = [S, \overline{S}]$ is a bond if and only if $G \setminus F$ has two components. Equivalently, if and only if the subgraphs of G induced by each of S and $\overline{S}$ are connected.

Proof.

If $G \setminus F$ has two components, then F is a bond, since $G \setminus F'$ is connected for every proper subset F' of F.
If $G \setminus F$ has more than two components, then we may assume $S = A \cup B$ with no edges between A and B. Then $[A, \overline{A}]$ is an edge cut which is a proper subset of F; a contradiction. □

Tutte Connectivity

Definition 5.10

- A *k-separation* of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.

Tutte Connectivity

Definition 5.10

- A k-separation of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then Tutte connectivity of G is $\min\{j : G \text{ has a } j \text{ separation}\}$, and ∞ if no k-separation exists.

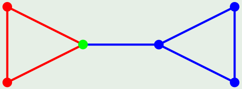
Tutte Connectivity

Definition 5.10

- A *k*-separation of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then Tutte connectivity of G is $\min\{j : G \text{ has a } j \text{ separation}\}$, and ∞ if no k-separation exists.

Example 5.11

1-separations:



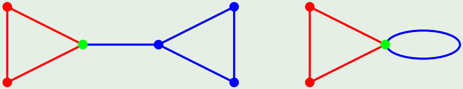
Tutte Connectivity

Definition 5.10

- A *k-separation* of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then Tutte connectivity of G is $\min\{j : G \text{ has a } j \text{ separation}\}$, and ∞ if no k-separation exists.

Example 5.11

1-separations:



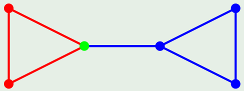
Tutte Connectivity

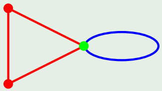
Definition 5.10

- A k-separation of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then Tutte connectivity of G is $\min\{j : G \text{ has a } j \text{ separation}\}$, and ∞ if no k-separation exists.

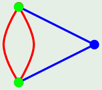
Example 5.11

1-separations:





2-separations:



Tutte Connectivity

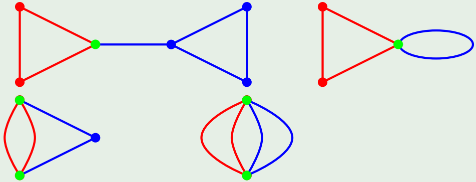
Definition 5.10

- A *k-separation* of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then *Tutte connectivity* of G is $\min\{j : G$ has a j separation$\}$, and ∞ if no k-separation exists.

Example 5.11

1-separations:

2-separations:



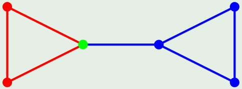
Tutte Connectivity

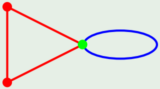
Definition 5.10

- A *k-separation* of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then Tutte connectivity of G is $\min\{j : G \text{ has a } j \text{ separation}\}$, and ∞ if no k-separation exists.

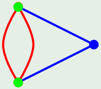
Example 5.11

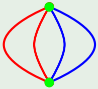
1-separations:





2-separations:





∞-te Tutte connectivity:

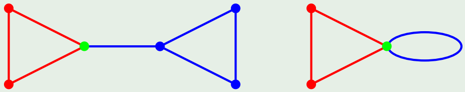
Tutte Connectivity

Definition 5.10

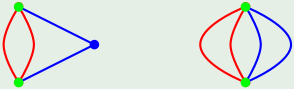
- A *k-separation* of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then Tutte connectivity of G is $\min\{j : G$ has a j separation$\}$, and ∞ if no k-separation exists.

Example 5.11

1-separations:



2-separations:

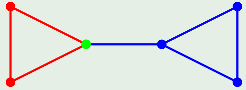
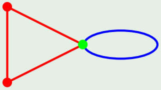


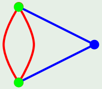
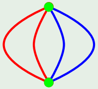
∞-te Tutte connectivity: $\emptyset$

Definition 5.10

- A *k-separation* of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then *Tutte connectivity* of G is $\min\{j : G$ has a j separation$\}$, and ∞ if no k-separation exists.

Example 5.11

1-separations:  

2-separations:  

∞-te Tutte connectivity: $\emptyset$ 

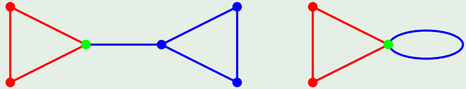
Tutte Connectivity

Definition 5.10

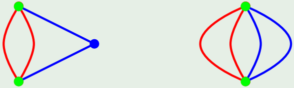
- A *k-separation* of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then Tutte connectivity of G is $\min\{j : G$ has a j separation$\}$, and ∞ if no k-separation exists.

Example 5.11

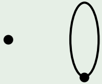
1-separations:



2-separations:



∞-te Tutte connectivity: $\emptyset$



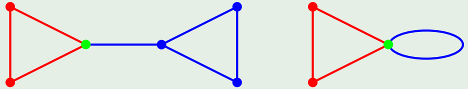
Tutte Connectivity

Definition 5.10

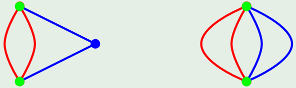
- A *k-separation* of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then *Tutte connectivity* of G is $\min\{j : G$ has a j separation$\}$, and ∞ if no k-separation exists.

Example 5.11

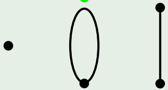
1-separations:



2-separations:



∞-te Tutte connectivity: $\emptyset$



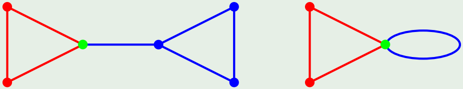
Tutte Connectivity

Definition 5.10

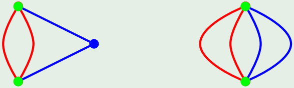
- A k-separation of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then Tutte connectivity of G is $\min\{j : G \text{ has a } j \text{ separation}\}$, and ∞ if no k-separation exists.

Example 5.11

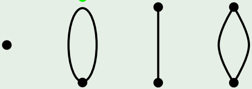
1-separations:



2-separations:



∞-te Tutte connectivity: $\emptyset$



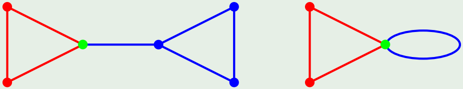
Tutte Connectivity

Definition 5.10

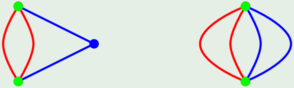
- A k-separation of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then Tutte connectivity of G is $\min\{j : G \text{ has a } j \text{ separation}\}$, and ∞ if no k-separation exists.

Example 5.11

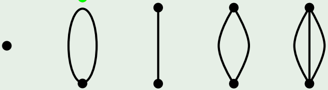
1-separations:



2-separations:



∞-te Tutte connectivity: $\emptyset$



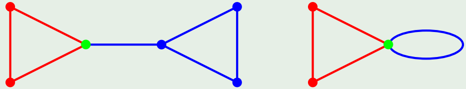
Tutte Connectivity

Definition 5.10

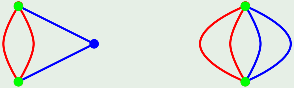
- A k-separation of a graph G is a pair of subgraphs $\{A, B\}$ of G such that each of A and B has size at least k, $A \neq G$, $B \neq G$, $A \cup B = G$, and $A \cap B$ is trivial of order at most k.
- If G has a k-separation for some k, then Tutte connectivity of G is $\min\{j : G \text{ has a } j \text{ separation}\}$, and ∞ if no k-separation exists.

Example 5.11

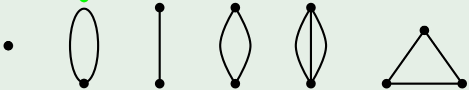
1-separations:



2-separations:



∞-te Tutte connectivity: $\emptyset$



Tutte Connectivity vs. Vertex Connectivity

Theorem 5.12

If G is a graph on at least 3 vertices and $G \ncong K_3$, then the Tutte connectivity of G is $\min(\kappa(G), g(G))$, where $g(G)$ is the girth of G, that is, the length of a shortest cycle in G.

Tutte Connectivity vs. Vertex Connectivity

Theorem 5.12

If G is a graph on at least 3 vertices and $G \ncong K_3$, then the Tutte connectivity of G is $\min(\kappa(G), g(G))$, where $g(G)$ is the girth of G, that is, the length of a shortest cycle in G.

Proof: Exercise.

Tutte Connectivity vs. Vertex Connectivity

Theorem 5.12

If G is a graph on at least 3 vertices and $G \ncong K_3$, then the Tutte connectivity of G is $\min(\kappa(G), g(G))$, where $g(G)$ is the girth of G, that is, the length of a shortest cycle in G.

Proof: Exercise.

Definition 5.13

- A component of a graph G is a maximal subgraph of G that has Tutte connectivity at least 1.

Tutte Connectivity vs. Vertex Connectivity

Theorem 5.12

If G is a graph on at least 3 vertices and $G \ncong K_3$, then the Tutte connectivity of G is $\min(\kappa(G), g(G))$, where $g(G)$ is the girth of G, that is, the length of a shortest cycle in G.

Proof: Exercise.

Definition 5.13

- A component of a graph G is a maximal subgraph of G that has Tutte connectivity at least 1.
- A block of a graph G is a maximal subgraph of G that has Tutte connectivity at least 2.

Tutte Connectivity vs. Vertex Connectivity

Theorem 5.12

If G is a graph on at least 3 vertices and $G \not\cong K_3$, then the Tutte connectivity of G is $\min(\kappa(G), g(G))$, where $g(G)$ is the girth of G, that is, the length of a shortest cycle in G.

Proof: Exercise.

Definition 5.13

- A component of a graph G is a maximal subgraph of G that has Tutte connectivity at least 1.
- A block of a graph G is a maximal subgraph of G that has Tutte connectivity at least 2.

Note 5.14

A block of a non-empty graph is an isolated vertex, a loop-graph, a graph on two vertices with a positive number of edges between those vertices, or is vertex-2-connected.

Block Tree

Note 5.15

Two distinct blocks in a graph share at most one vertex since otherwise their union would be Tutte-2-connected.

Block Tree

Note 5.15

Two distinct blocks in a graph share at most one vertex since otherwise their union would be Tutte-2-connected.

Definition 5.16

The block-tree of a connected graph G is a tree T whose vertex set is the disjoint union of the blocks of G

Block Tree

Note 5.15

Two distinct blocks in a graph share at most one vertex since otherwise their union would be Tutte-2-connected.

Definition 5.16

The block-tree of a connected graph G is a tree T whose vertex set is the disjoint union of the blocks of G and those vertices of G that belong to more than one block.

Block Tree

Note 5.15

Two distinct blocks in a graph share at most one vertex since otherwise their union would be Tutte-2-connected.

Definition 5.16

The block-tree of a connected graph G is a tree T whose vertex set is the disjoint union of the blocks of G and those vertices of G that belong to more than one block. The only edges in T are those that join vertices of G to bloks that contain them.

Block Tree

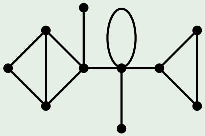
Note 5.15

Two distinct blocks in a graph share at most one vertex since otherwise their union would be Tutte-2-connected.

Definition 5.16

The block-tree of a connected graph G is a tree T whose vertex set is the disjoint union of the blocks of G and those vertices of G that belong to more than one block. The only edges in T are those that join vertices of G to bloks that contain them.

Example 5.17



Block Tree

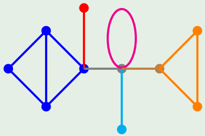
Note 5.15

Two distinct blocks in a graph share at most one vertex since otherwise their union would be Tutte-2-connected.

Definition 5.16

The block-tree of a connected graph G is a tree T whose vertex set is the disjoint union of the blocks of G and those vertices of G that belong to more than one block. The only edges in T are those that join vertices of G to bloks that contain them.

Example 5.17



Block Tree

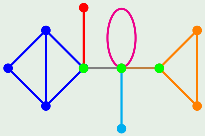
Note 5.15

Two distinct blocks in a graph share at most one vertex since otherwise their union would be Tutte-2-connected.

Definition 5.16

The block-tree of a connected graph G is a tree T whose vertex set is the disjoint union of the blocks of G and those vertices of G that belong to more than one block. The only edges in T are those that join vertices of G to bloks that contain them.

Example 5.17



Block Tree

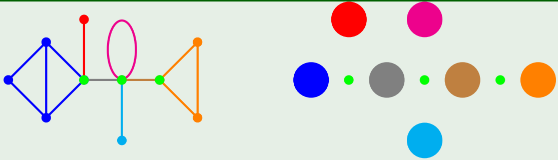
Note 5.15

Two distinct blocks in a graph share at most one vertex since otherwise their union would be Tutte-2-connected.

Definition 5.16

The block-tree of a connected graph G is a tree T whose vertex set is the disjoint union of the blocks of G and those vertices of G that belong to more than one block. The only edges in T are those that join vertices of G to bloks that contain them.

Example 5.17



Block Tree

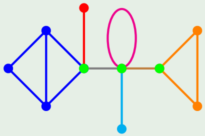
Note 5.15

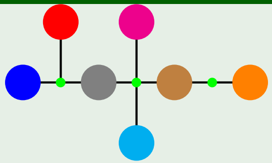
Two distinct blocks in a graph share at most one vertex since otherwise their union would be Tutte-2-connected.

Definition 5.16

The block-tree of a connected graph G is a tree T whose vertex set is the disjoint union of the blocks of G and those vertices of G that belong to more than one block. The only edges in T are those that join vertices of G to bloks that contain them.

Example 5.17





Whitney's Characterization of 2-Connected Graphs

Definition 5.18

Two paths are internally-disjoint if neither contains a non-endpoint of the other.

Whitney's Characterization of 2-Connected Graphs

Definition 5.18

Two paths are internally-disjoint if neither contains a non-endpoint of the other.

Theorem 5.19 (Whitney)

A graph with at least three vertices is 2-connected if and only if each pair u and v of vertices is connected by a pair internally-disjoint uv-paths.

Whitney's Characterization of 2-Connected Graphs

Definition 5.18

Two paths are internally-disjoint if neither contains a non-endpoint of the other.

Theorem 5.19 (Whitney)

A graph with at least three vertices is 2-connected if and only if each pair u and v of vertices is connected by a pair internally-disjoint uv-paths.

Proof.

If G has two internally-disjoint uv-paths, then deletion of one vertex cannot separate u from v.

$\square$

Whitney's Characterization of 2-Connected Graphs

Definition 5.18

Two paths are internally-disjoint if neither contains a non-endpoint of the other.

Theorem 5.19 (Whitney)

A graph with at least three vertices is 2-connected if and only if each pair u and v of vertices is connected by a pair internally-disjoint uv-paths.

Proof.

If G has two internally-disjoint uv-paths, then deletion of one vertex cannot separate u from v. Hence G has no one-element vertex-cuts and so is 2-connected. $\square$

Proof of Sufficiency

For the converse, suppose that G is 2-connected.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u, v)$ that G has two internally-disjoint uv-paths.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u,v)$ that G has two internally-disjoint uv-paths. When $d(u,v) = 1$, the graph $G \setminus uv$ is connected since $\kappa'(G) \geq \kappa(G) \geq 2$.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u, v)$ that G has two internally-disjoint uv-paths. When $d(u, v) = 1$, the graph $G \setminus uv$ is connected since $\kappa'(G) \geq \kappa(G) \geq 2$. A uv-path in $G \setminus uv$ is internally disjoint from the uv-path consisting of the edge uv only.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u, v)$ that G has two internally-disjoint uv-paths. When $d(u, v) = 1$, the graph $G \setminus uv$ is connected since $\kappa'(G) \geq \kappa(G) \geq 2$. A uv-path in $G \setminus uv$ is internally disjoint from the uv-path consisting of the edge uv only.
For the induction step, consider $d(u, v) = k > 1$ and assume that G has internally-disjoint xy-paths whenever $1 \leq d(x, y) < k$.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u, v)$ that G has two internally-disjoint uv-paths. When $d(u, v) = 1$, the graph $G \setminus uv$ is connected since $\kappa'(G) \geq \kappa(G) \geq 2$. A uv-path in $G \setminus uv$ is internally disjoint from the uv-path consisting of the edge uv only.
For the induction step, consider $d(u, v) = k > 1$ and assume that G has internally-disjoint xy-paths whenever $1 \leq d(x, y) < k$. Let w be the vertex just before v on a shortest uv-path.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u, v)$ that G has two internally-disjoint uv-paths. When $d(u, v) = 1$, the graph $G \setminus uv$ is connected since $\kappa'(G) \geq \kappa(G) \geq 2$. A uv-path in $G \setminus uv$ is internally disjoint from the uv-path consisting of the edge uv only.
For the induction step, consider $d(u, v) = k > 1$ and assume that G has internally-disjoint xy-paths whenever $1 \leq d(x, y) < k$. Let w be the vertex just before v on a shortest uv-path. Then $d(u, w) = k - 1$ and, by the induction hypothesis, G has two internally-disjoint uw-paths P and Q.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u, v)$ that G has two internally-disjoint uv-paths. When $d(u, v) = 1$, the graph $G \setminus uv$ is connected since $\kappa'(G) \geq \kappa(G) \geq 2$. A uv-path in $G \setminus uv$ is internally disjoint from the uv-path consisting of the edge uv only.
For the induction step, consider $d(u, v) = k > 1$ and assume that G has internally-disjoint xy-paths whenever $1 \leq d(x, y) < k$. Let w be the vertex just before v on a shortest uv-path. Then $d(u, w) = k - 1$ and, by the induction hypothesis, G has two internally-disjoint uw-paths P and Q. Since $G - w$ is connected, it has a uv-path R.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u, v)$ that G has two internally-disjoint uv-paths. When $d(u, v) = 1$, the graph $G \setminus uv$ is connected since $\kappa'(G) \geq \kappa(G) \geq 2$. A uv-path in $G \setminus uv$ is internally disjoint from the uv-path consisting of the edge uv only.
For the induction step, consider $d(u, v) = k > 1$ and assume that G has internally-disjoint xy-paths whenever $1 \leq d(x, y) < k$. Let w be the vertex just before v on a shortest uv-path. Then $d(u, w) = k - 1$ and, by the induction hypothesis, G has two internally-disjoint uw-paths P and Q. Since $G - w$ is connected, it has a uv-path R. If R meets P and Q only in u, then the conclusion follows.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u,v)$ that G has two internally-disjoint uv-paths. When $d(u,v) = 1$, the graph $G \setminus uv$ is connected since $\kappa'(G) \geq \kappa(G) \geq 2$. A uv-path in $G \setminus uv$ is internally disjoint from the uv-path consisting of the edge uv only.
For the induction step, consider $d(u,v) = k > 1$ and assume that G has internally-disjoint xy-paths whenever $1 \leq d(x,y) < k$. Let w be the vertex just before v on a shortest uv-path. Then $d(u,w) = k-1$ and, by the induction hypothesis, G has two internally-disjoint uw-paths P and Q. Since $G - w$ is connected, it has a uv-path R. If R meets P and Q only in u, then the conclusion follows. Let z be the last vertex on R that belongs to $P \cup Q$.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u, v)$ that G has two internally-disjoint uv-paths. When $d(u, v) = 1$, the graph $G \setminus uv$ is connected since $\kappa'(G) \geq \kappa(G) \geq 2$. A uv-path in $G \setminus uv$ is internally disjoint from the uv-path consisting of the edge uv only.
For the induction step, consider $d(u, v) = k > 1$ and assume that G has internally-disjoint xy-paths whenever $1 \leq d(x, y) < k$. Let w be the vertex just before v on a shortest uv-path. Then $d(u, w) = k - 1$ and, by the induction hypothesis, G has two internally-disjoint uw-paths P and Q. Since $G - w$ is connected, it has a uv-path R. If R meets P and Q only in u, then the conclusion follows. Let z be the last vertex on R that belongs to $P \cup Q$. By symmetry, we may assume that $z \in V(P)$.

Proof of Sufficiency

For the converse, suppose that G is 2-connected. We prove by induction on $d(u,v)$ that G has two internally-disjoint uv-paths. When $d(u,v) = 1$, the graph $G \setminus uv$ is connected since $\kappa'(G) \geq \kappa(G) \geq 2$. A uv-path in $G \setminus uv$ is internally disjoint from the uv-path consisting of the edge uv only.
For the induction step, consider $d(u,v) = k > 1$ and assume that G has internally-disjoint xy-paths whenever $1 \leq d(x,y) < k$. Let w be the vertex just before v on a shortest uv-path. Then $d(u,w) = k-1$ and, by the induction hypothesis, G has two internally-disjoint uw-paths P and Q. Since $G - w$ is connected, it has a uv-path R. If R meets P and Q only in u, then the conclusion follows. Let z be the last vertex on R that belongs to $P \cup Q$. By symmetry, we may assume that $z \in V(P)$. We combine the uz-subpath of P with the zv-subpath of R to obtain a uv-path internally-disjoint from $Q \cup wv$.

Expansion Lemma

Lemma 5.20 (Expansion Lemma)

If G is a k-connected graph and G' is obtained from G by adding a new vertex y adjacent to at least k vertices of G, then G' is also k-connected.

Expansion Lemma

Lemma 5.20 (Expansion Lemma)

If G is a k-connected graph and G' is obtained from G by adding a new vertex y adjacent to at least k vertices of G, then G' is also k-connected.

Proof.

Suppose S is a separating set of G'.

$\square$

Expansion Lemma

Lemma 5.20 (Expansion Lemma)

If G is a k-connected graph and G' is obtained from G by adding a new vertex y adjacent to at least k vertices of G, then G' is also k-connected.

Proof.

Suppose S is a separating set of G'. If $y \in S$, then $S - y$ separates G, so $|S| \geq k + 1$.

□

Expansion Lemma

Lemma 5.20 (Expansion Lemma)

If G is a k-connected graph and G' is obtained from G by adding a new vertex y adjacent to at least k vertices of G, then G' is also k-connected.

Proof.

Suppose S is a separating set of G'. If $y \in S$, then $S - y$ separates G, so $|S| \geq k + 1$. If $y \notin S$ and $N(y) \subseteq S$, then $|S| \geq k$.

□

Expansion Lemma

Lemma 5.20 (Expansion Lemma)

If G is a k-connected graph and G' is obtained from G by adding a new vertex y adjacent to at least k vertices of G, then G' is also k-connected.

Proof.

Suppose S is a separating set of G'. If $y \in S$, then $S - y$ separates G, so $|S| \geq k + 1$. If $y \notin S$ and $N(y) \subseteq S$, then $|S| \geq k$. Otherwise, S must separate G, and again $|S| \geq k$. $\square$

Characterization of 2-Connected Graphs

Theorem 5.21

If G is simple and $|G| \geq 3$, then the following are equivalent (and characterize simple 2-connected graphs):

Characterization of 2-Connected Graphs

Theorem 5.21

If G is simple and $|G| \geq 3$, then the following are equivalent (and characterize simple 2-connected graphs):

(A) *G is connected and has no cut-vertex;*

Characterization of 2-Connected Graphs

Theorem 5.21

If G is simple and $|G| \geq 3$, then the following are equivalent (and characterize simple 2-connected graphs):

(A) *G is connected and has no cut-vertex;*

(B) *For every two vertices x and y of G, there are two internally-disjoint xy-paths;*

Theorem 5.21

If G is simple and $|G| \geq 3$, then the following are equivalent (and characterize simple 2-connected graphs):

(A) *G is connected and has no cut-vertex;*

(B) *For every two vertices x and y of G, there are two internally-disjoint xy-paths;*

(C) *For every two vertices x and y of G, there is a cycle through x and y.*

Characterization of 2-Connected Graphs

Theorem 5.21

If G is simple and $|G| \geq 3$, then the following are equivalent (and characterize simple 2-connected graphs):

(A) *G is connected and has no cut-vertex;*

(B) *For every two vertices x and y of G, there are two internally-disjoint xy-paths;*

(C) *For every two vertices x and y of G, there is a cycle through x and y.*

(D) *$\delta \geq 1$ and every pair of edges of G lies on a common cycle.*

Characterization of 2-Connected Graphs

Theorem 5.21

If G is simple and $|G| \geq 3$, then the following are equivalent (and characterize simple 2-connected graphs):

(A) *G is connected and has no cut-vertex;*

(B) *For every two vertices x and y of G, there are two internally-disjoint xy-paths;*

(C) *For every two vertices x and y of G, there is a cycle through x and y.*

(D) *$\delta \geq 1$ and every pair of edges of G lies on a common cycle.*

Proof.

Whitney's Theorem 5.19 establishes the equivalence of (A) and (B).

$\square$

Characterization of 2-Connected Graphs

Theorem 5.21

If G is simple and $|G| \geq 3$, then the following are equivalent (and characterize simple 2-connected graphs):

(A) *G is connected and has no cut-vertex;*

(B) *For every two vertices x and y of G, there are two internally-disjoint xy-paths;*

(C) *For every two vertices x and y of G, there is a cycle through x and y.*

(D) *$\delta \geq 1$ and every pair of edges of G lies on a common cycle.*

Proof.

Whitney's Theorem 5.19 establishes the equivalence of (A) and (B). Clearly, (B) and (C) are equivalent.

□

Characterization of 2-Connected Graphs

Theorem 5.21

If G is simple and $|G| \geq 3$, then the following are equivalent (and characterize simple 2-connected graphs):

(A) *G is connected and has no cut-vertex;*

(B) *For every two vertices x and y of G, there are two internally-disjoint xy-paths;*

(C) *For every two vertices x and y of G, there is a cycle through x and y.*

(D) *$\delta \geq 1$ and every pair of edges of G lies on a common cycle.*

Proof.

Whitney's Theorem 5.19 establishes the equivalence of (A) and (B). Clearly, (B) and (C) are equivalent. To see that (D) implies (C), apply (D) to edges incident to the desired x and y. $\square$

Proof, Continued

(A) G is connected and has no cut-vertex;

(C) For every two vertices x and y of G, there is a cycle through x and y.

(D) $\delta \geq 1$ and every pair of edges of G lies on a common cycle.

Proof, Continued

(A) G is connected and has no cut-vertex;

(C) For every two vertices x and y of G, there is a cycle through x and y.

(D) $\delta \geq 1$ and every pair of edges of G lies on a common cycle.

We prove that (A) and (C) imply (D).

Proof, Continued

(A) G is connected and has no cut-vertex;

(C) For every two vertices x and y of G, there is a cycle through x and y.

(D) $\delta \geq 1$ and every pair of edges of G lies on a common cycle.

We prove that (A) and (C) imply (D). Suppose G is 2-connected and uv and xy are edges of G.

Proof, Continued

(A) G is connected and has no cut-vertex;

(C) For every two vertices x and y of G, there is a cycle through x and y.

(D) $\delta \geq 1$ and every pair of edges of G lies on a common cycle.

We prove that (A) and (C) imply (D). Suppose G is 2-connected and uv and xy are edges of G. Add to G vertices w and z, and connect w with u and v, and connect z to x and y.

Proof, Continued

(A) G is connected and has no cut-vertex;

(C) For every two vertices x and y of G, there is a cycle through x and y.

(D) $\delta \geq 1$ and every pair of edges of G lies on a common cycle.

We prove that (A) and (C) imply (D). Suppose G is 2-connected and uv and xy are edges of G. Add to G vertices w and z, and connect w with u and v, and connect z to x and y. By The Expansion Lemma 5.20, the resulting graph G' is also 2-connected.

Proof, Continued

(A) G is connected and has no cut-vertex;

(C) For every two vertices x and y of G, there is a cycle through x and y.

(D) $\delta \geq 1$ and every pair of edges of G lies on a common cycle.

We prove that (A) and (C) imply (D). Suppose G is 2-connected and uv and xy are edges of G. Add to G vertices w and z, and connect w with u and v, and connect z to x and y. By The Expansion Lemma 5.20, the resulting graph G' is also 2-connected. Hence w and z lie on a common cycle C' of G'.

Proof, Continued

(A) G is connected and has no cut-vertex;

(C) For every two vertices x and y of G, there is a cycle through x and y.

(D) $\delta \geq 1$ and every pair of edges of G lies on a common cycle.

We prove that (A) and (C) imply (D). Suppose G is 2-connected and uv and xy are edges of G. Add to G vertices w and z, and connect w with u and v, and connect z to x and y. By The Expansion Lemma 5.20, the resulting graph G' is also 2-connected. Hence w and z lie on a common cycle C' of G'. Replace the paths uwv and xyz by uv and xz, respectively, to obtain the desired cycle of G.

Subdivisions

Definition 5.22

- Subdividing an edge uv of a graph G is the operation of deleting uv and adding a path uwv through a new vertex w.

Subdivisions

Definition 5.22

- Subdividing an edge uv of a graph G is the operation of deleting uv and adding a path uwv through a new vertex w.
- A graph G is a subdivision of a graph H if G can be obtained from H by successively subdividing (zero or more) edges.

Subdivisions

Definition 5.22

- Subdividing an edge uv of a graph G is the operation of deleting uv and adding a path uwv through a new vertex w.
- A graph G is a subdivision of a graph H if G can be obtained from H by successively subdividing (zero or more) edges.
- A graph H is a topological minor of G, written $H \leqslant_{\mathrm{t}} G$, if a subgraph of G is a subdivision of H.

Subdivisions

Definition 5.22

- Subdividing an edge uv of a graph G is the operation of deleting uv and adding a path uwv through a new vertex w.
- A graph G is a subdivision of a graph H if G can be obtained from H by successively subdividing (zero or more) edges.
- A graph H is a topological minor of G, written $H \leqslant_{\mathrm{t}} G$, if a subgraph of G is a subdivision of H.
- A graph is a topological minor of G if it can be obtained from G by a sequence of operations each of which is one of the following:

Subdivisions

Definition 5.22

- Subdividing an edge uv of a graph G is the operation of deleting uv and adding a path uwv through a new vertex w.
- A graph G is a subdivision of a graph H if G can be obtained from H by successively subdividing (zero or more) edges.
- A graph H is a topological minor of G, written $H \leqslant_{\mathrm{t}} G$, if a subgraph of G is a subdivision of H.
- A graph is a topological minor of G if it can be obtained from G by a sequence of operations each of which is one of the following:
  - deleting an edge;

Subdivisions

Definition 5.22

- Subdividing an edge uv of a graph G is the operation of deleting uv and adding a path uwv through a new vertex w.
- A graph G is a subdivision of a graph H if G can be obtained from H by successively subdividing (zero or more) edges.
- A graph H is a topological minor of G, written $H \leqslant_{\mathrm{t}} G$, if a subgraph of G is a subdivision of H.
- A graph is a topological minor of G if it can be obtained from G by a sequence of operations each of which is one of the following:
  - deleting an edge;
  - deleting a vertex; and

Subdivisions

Definition 5.22

- Subdividing an edge uv of a graph G is the operation of deleting uv and adding a path uwv through a new vertex w.
- A graph G is a subdivision of a graph H if G can be obtained from H by successively subdividing (zero or more) edges.
- A graph H is a topological minor of G, written $H \leqslant_{\mathrm{t}} G$, if a subgraph of G is a subdivision of H.
- A graph is a topological minor of G if it can be obtained from G by a sequence of operations each of which is one of the following:
  - deleting an edge;
  - deleting a vertex; and
  - contracting an edge incident with a vertex of degree two (un-subdividing an edge).

Subdivisions and 2-Connectedness

Corollary 5.23

A subdivision of a 2-connected graph is also 2-connected.

Subdivisions and 2-Connectedness

Corollary 5.23

A subdivision of a 2-connected graph is also 2-connected.

Proof.

Suppose G' is formed by subdividing an edge uv of G with a new vertex w.

$\square$

Subdivisions and 2-Connectedness

Corollary 5.23

A subdivision of a 2-connected graph is also 2-connected.

Proof.

Suppose G' is formed by subdividing an edge uv of G with a new vertex w. By Theorem 5.21, it suffices to find a cycle through two arbitrary edges e and f of G'.

□

Corollary 5.23

A subdivision of a 2-connected graph is also 2-connected.

Proof.

Suppose G' is formed by subdividing an edge uv of G with a new vertex w. By Theorem 5.21, it suffices to find a cycle through two arbitrary edges e and f of G'. If $e, f \in E(G)$, then we can use the cycle of G,

$\square$

Corollary 5.23

A subdivision of a 2-connected graph is also 2-connected.

Proof.

Suppose G' is formed by subdividing an edge uv of G with a new vertex w. By Theorem 5.21, it suffices to find a cycle through two arbitrary edges e and f of G'. If $e, f \in E(G)$, then we can use the cycle of G, unless it uses uv,

□

Subdivisions and 2-Connectedness

Corollary 5.23

A subdivision of a 2-connected graph is also 2-connected.

Proof.

Suppose G' is formed by subdividing an edge uv of G with a new vertex w. By Theorem 5.21, it suffices to find a cycle through two arbitrary edges e and f of G'. If $e, f \in E(G)$, then we can use the cycle of G, unless it uses uv, in which case we reroute the cycle through w.

□

Corollary 5.23

A subdivision of a 2-connected graph is also 2-connected.

Proof.

Suppose G' is formed by subdividing an edge uv of G with a new vertex w. By Theorem 5.21, it suffices to find a cycle through two arbitrary edges e and f of G'. If $e, f \in E(G)$, then we can use the cycle of G, unless it uses uv, in which case we reroute the cycle through w. When $e \in E(G)$ and $f \in \{uw, wv\}$, we modify a cycle passing through e and uv.

$\square$

Subdivisions and 2-Connectedness

Corollary 5.23

A subdivision of a 2-connected graph is also 2-connected.

Proof.

Suppose G' is formed by subdividing an edge uv of G with a new vertex w. By Theorem 5.21, it suffices to find a cycle through two arbitrary edges e and f of G'. If $e, f \in E(G)$, then we can use the cycle of G, unless it uses uv, in which case we reroute the cycle through w. When $e \in E(G)$ and $f \in \{uw, wv\}$, we modify a cycle passing through e and uv. When $\{e, f\} = \{uw, wv\}$, we modify a cycle through uv. $\square$

Ears

Definition 5.24

- A path addition to G is the addition to G of a path of length $\ell \geq 1$ between two vertices of G, introducing $\ell - 1$ new vertices.

Ears

Definition 5.24

- A path addition to G is the addition to G of a path of length $\ell \geq 1$ between two vertices of G, introducing $\ell - 1$ new vertices.
- The added path is an ear.

Ears

Definition 5.24

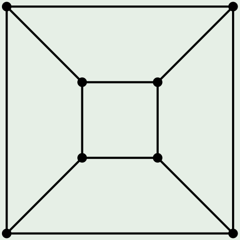
- A path addition to G is the addition to G of a path of length $\ell \geq 1$ between two vertices of G, introducing $\ell - 1$ new vertices.
- The added path is an ear.
- An ear decomposition is a partition of $E(G)$ into sets R_0, R_1, ..., R_k so that $C = R_0$ is a cycle, and R_i, for $i > 0$, is a path addition to the graph $R_0 \cup R_1 \cup \ldots \cup R_{i-1}$.

Ears

Definition 5.24

- A path addition to G is the addition to G of a path of length $\ell \geq 1$ between two vertices of G, introducing $\ell - 1$ new vertices.
- The added path is an ear.
- An ear decomposition is a partition of $E(G)$ into sets R_0, R_1, ..., R_k so that $C = R_0$ is a cycle, and R_i, for $i > 0$, is a path addition to the graph $R_0 \cup R_1 \cup \ldots \cup R_{i-1}$.

Example 5.25

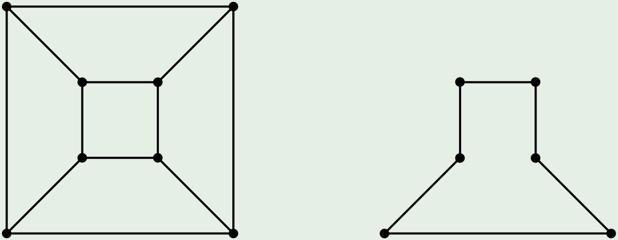


Ears

Definition 5.24

- A path addition to G is the addition to G of a path of length $\ell \geq 1$ between two vertices of G, introducing $\ell - 1$ new vertices.
- The added path is an ear.
- An ear decomposition is a partition of $E(G)$ into sets $R_0, R_1, \dots, R_k$ so that $C = R_0$ is a cycle, and R_i, for $i > 0$, is a path addition to the graph $R_0 \cup R_1 \cup \dots \cup R_{i-1}$.

Example 5.25

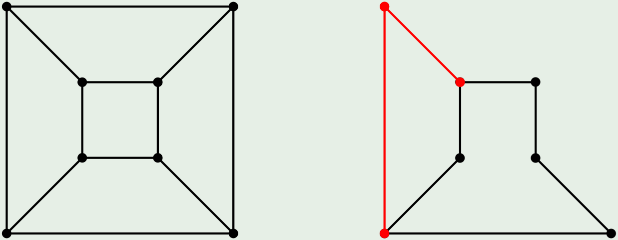


Ears

Definition 5.24

- A path addition to G is the addition to G of a path of length $\ell \geq 1$ between two vertices of G, introducing $\ell - 1$ new vertices.
- The added path is an ear.
- An ear decomposition is a partition of $E(G)$ into sets $R_0, R_1, \ldots, R_k$ so that $C = R_0$ is a cycle, and R_i, for $i > 0$, is a path addition to the graph $R_0 \cup R_1 \cup \ldots \cup R_{i-1}$.

Example 5.25

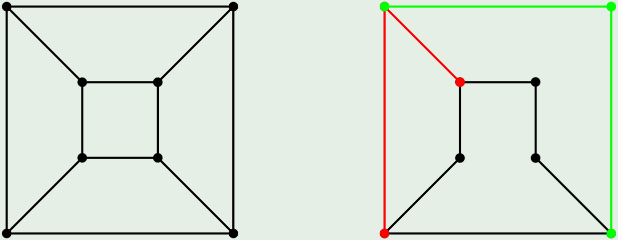


Ears

Definition 5.24

- A path addition to G is the addition to G of a path of length $\ell \geq 1$ between two vertices of G, introducing $\ell - 1$ new vertices.
- The added path is an ear.
- An ear decomposition is a partition of $E(G)$ into sets $R_0, R_1, \ldots, R_k$ so that $C = R_0$ is a cycle, and R_i, for $i > 0$, is a path addition to the graph $R_0 \cup R_1 \cup \ldots \cup R_{i-1}$.

Example 5.25

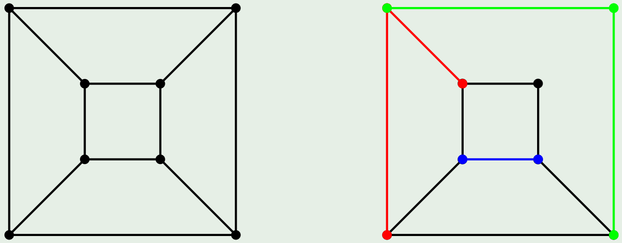


Ears

Definition 5.24

- A path addition to G is the addition to G of a path of length $\ell \geq 1$ between two vertices of G, introducing $\ell - 1$ new vertices.
- The added path is an ear.
- An ear decomposition is a partition of $E(G)$ into sets R_0, R_1, ..., R_k so that $C = R_0$ is a cycle, and R_i, for $i > 0$, is a path addition to the graph $R_0 \cup R_1 \cup \ldots \cup R_{i-1}$.

Example 5.25

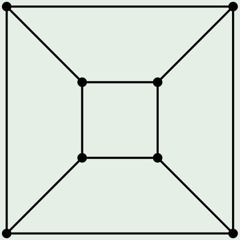


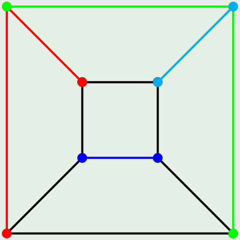
Ears

Definition 5.24

- A path addition to G is the addition to G of a path of length $\ell \geq 1$ between two vertices of G, introducing $\ell - 1$ new vertices.
- The added path is an ear.
- An ear decomposition is a partition of $E(G)$ into sets R_0, R_1, ..., R_k so that $C = R_0$ is a cycle, and R_i, for $i > 0$, is a path addition to the graph $R_0 \cup R_1 \cup \ldots \cup R_{i-1}$.

Example 5.25





Whitney's Ear Decomposition

Theorem 5.26 (Whitney's Ear Decomposition)

A simple graph is 2-connected if and only if it has an ear decomposition.

Whitney's Ear Decomposition

Theorem 5.26 (Whitney's Ear Decomposition)

A simple graph is 2-connected if and only if it has an ear decomposition. Furthermore, every cycle in a 2-connected graph is the initial cycle of some ear decomposition.

Whitney's Ear Decomposition

Theorem 5.26 (Whitney's Ear Decomposition)

A simple graph is 2-connected if and only if it has an ear decomposition. Furthermore, every cycle in a 2-connected graph is the initial cycle of some ear decomposition.

Proof.

First we prove that graph with an ear decomposition is 2-connected.

□

Whitney's Ear Decomposition

Theorem 5.26 (Whitney's Ear Decomposition)

A simple graph is 2-connected if and only if it has an ear decomposition. Furthermore, every cycle in a 2-connected graph is the initial cycle of some ear decomposition.

Proof.

First we prove that graph with an ear decomposition is 2-connected. Since cycles in simple graphs are 2-connected, it suffices to show that path addition preserves 2-connectedness.

$\square$

Whitney's Ear Decomposition

Theorem 5.26 (Whitney's Ear Decomposition)

A simple graph is 2-connected if and only if it has an ear decomposition. Furthermore, every cycle in a 2-connected graph is the initial cycle of some ear decomposition.

Proof.

First we prove that graph with an ear decomposition is 2-connected. Since cycles in simple graphs are 2-connected, it suffices to show that path addition preserves 2-connectedness. Let u and v be the endpoints of an ear P to be added to a 2-connected graph G.

□

Whitney's Ear Decomposition

Theorem 5.26 (Whitney's Ear Decomposition)

A simple graph is 2-connected if and only if it has an ear decomposition. Furthermore, every cycle in a 2-connected graph is the initial cycle of some ear decomposition.

Proof.

First we prove that graph with an ear decomposition is 2-connected. Since cycles in simple graphs are 2-connected, it suffices to show that path addition preserves 2-connectedness. Let u and v be the endpoints of an ear P to be added to a 2-connected graph G. Adding an edge e joining u and v (if u and v are non-adjacent) cannot reduce vertex-connectivity, so $G \cup e$ is 2-connected.

□

Whitney's Ear Decomposition

Theorem 5.26 (Whitney's Ear Decomposition)

A simple graph is 2-connected if and only if it has an ear decomposition. Furthermore, every cycle in a 2-connected graph is the initial cycle of some ear decomposition.

Proof.

First we prove that graph with an ear decomposition is 2-connected. Since cycles in simple graphs are 2-connected, it suffices to show that path addition preserves 2-connectedness. Let u and v be the endpoints of an ear P to be added to a 2-connected graph G. Adding an edge e joining u and v (if u and v are non-adjacent) cannot reduce vertex-connectivity, so $G \cup e$ is 2-connected. A succession of subdivions covers $G \cup e$ into $G \cup P$.

$\square$

Whitney's Ear Decomposition

Theorem 5.26 (Whitney's Ear Decomposition)

A simple graph is 2-connected if and only if it has an ear decomposition. Furthermore, every cycle in a 2-connected graph is the initial cycle of some ear decomposition.

Proof.

First we prove that graph with an ear decomposition is 2-connected. Since cycles in simple graphs are 2-connected, it suffices to show that path addition preserves 2-connectedness. Let u and v be the endpoints of an ear P to be added to a 2-connected graph G. Adding an edge e joining u and v (if u and v are non-adjacent) cannot reduce vertex-connectivity, so $G \cup e$ is 2-connected. A succession of subdivions covers $G \cup e$ into $G \cup P$. By Corollary 5.23, each subdivision preserves 2-connectedness. □

Proof of Sufficiency

Now, given a 2-connected graph G, we build an ear decomposition of G from a cycle C of G.

Proof of Sufficiency

Now, given a 2-connected graph G, we build an ear decomposition of G from a cycle C of G. Let $G_0 = C$.

Proof of Sufficiency

Now, given a 2-connected graph G, we build an ear decomposition of G from a cycle C of G. Let $G_0 = C$. Suppose we have built up a subgraph G_i by adding ears.

Proof of Sufficiency

Now, given a 2-connected graph G, we build an ear decomposition of G from a cycle C of G. Let $G_0 = C$. Suppose we have built up a subgraph G_i by adding ears. If $G_i \neq G$, then we may choose an edge uv of $G \setminus E(G_i)$ and an edge $xy \in E(G_i)$.

Proof of Sufficiency

Now, given a 2-connected graph G, we build an ear decomposition of G from a cycle C of G. Let $G_0 = C$. Suppose we have built up a subgraph G_i by adding ears. If $G_i \neq G$, then we may choose an edge uv of $G \setminus E(G_i)$ and an edge $xy \in E(G_i)$. Because G is 2-connected, uv and xy lie on a common cycle C'.

Proof of Sufficiency

Now, given a 2-connected graph G, we build an ear decomposition of G from a cycle C of G. Let $G_0 = C$. Suppose we have built up a subgraph G_i by adding ears. If $G_i \neq G$, then we may choose an edge uv of $G \setminus E(G_i)$ and an edge $xy \in E(G_i)$. Because G is 2-connected, uv and xy lie on a common cycle C'. Let P be the path of C that contains uv and exactly two vertices of G_i, one at each end of P.

Proof of Sufficiency

Now, given a 2-connected graph G, we build an ear decomposition of G from a cycle C of G. Let $G_0 = C$. Suppose we have built up a subgraph G_i by adding ears. If $G_i \neq G$, then we may choose an edge uv of $G \setminus E(G_i)$ and an edge $xy \in E(G_i)$. Because G is 2-connected, uv and xy lie on a common cycle C'. Let P be the path of C that contains uv and exactly two vertices of G_i, one at each end of P. Now P is an ear that can be added to G_i to obtain a larger subgraph G_{i+1} of G.

Proof of Sufficiency

Now, given a 2-connected graph G, we build an ear decomposition of G from a cycle C of G. Let $G_0 = C$. Suppose we have built up a subgraph G_i by adding ears. If $G_i \neq G$, then we may choose an edge uv of $G \setminus E(G_i)$ and an edge $xy \in E(G_i)$. Because G is 2-connected, uv and xy lie on a common cycle C'. Let P be the path of C that contains uv and exactly two vertices of G_i, one at each end of P. Now P is an ear that can be added to G_i to obtain a larger subgraph G_{i+1} of G. The process ends when all edges of G have been absorbed.

Closed-Ear Decomposition

Definition 5.27

A closed-ear decomposition of a graph G is a partition of $E(G)$ into sets R_0, $R_1, \ldots, R_k$ such that R_0 is a cycle and R_i for $i > 0$ is either a path addition

Closed-Ear Decomposition

Definition 5.27

A closed-ear decomposition of a graph G is a partition of $E(G)$ into sets R_0, R_1, ..., R_k such that R_0 is a cycle and R_i for $i > 0$ is either a path addition or a cycle with exactly one vertex in $R_0 \cup R_1 \cup \ldots R_{i-1}$ (closed ear).

Closed-Ear Decomposition

Definition 5.27

A closed-ear decomposition of a graph G is a partition of $E(G)$ into sets R_0, R_1, ..., R_k such that R_0 is a cycle and R_i for $i > 0$ is either a path addition or a cycle with exactly one vertex in $R_0 \cup R_1 \cup \ldots R_{i-1}$ (closed ear).

Theorem 5.28

A simple graph is 2-edge-connected if and only if it has a closed-ear decomposition.

Closed-Ear Decomposition

Definition 5.27

A closed-ear decomposition of a graph G is a partition of $E(G)$ into sets R_0, R_1, ..., R_k such that R_0 is a cycle and R_i for $i > 0$ is either a path addition or a cycle with exactly one vertex in $R_0 \cup R_1 \cup \ldots R_{i-1}$ (closed ear).

Theorem 5.28

A simple graph is 2-edge-connected if and only if it has a closed-ear decomposition. Moreover, every cycle in a 2-edge-connected graph is the initial cycle in some closed-ear decomposition.

Proof omitted.

The Menger Theorem

Theorem 5.29 (Menger 1927)

If x and y are non-adjacent distinct vertices of a graph G, then the minimum size of a vertex-cut separating x from y equals the maximum number of pairwise internally-disjoint xy-paths.

The Menger Theorem

Theorem 5.29 (Menger 1927)

If x and y are non-adjacent distinct vertices of a graph G, then the minimum size of a vertex-cut separating x from y equals the maximum number of pairwise internally-disjoint xy-paths.

Proof.

Let $\kappa(x, y)$ denote the minimum size of a vertex-cut separating x from y.

$\square$

The Menger Theorem

Theorem 5.29 (Menger 1927)

If x and y are non-adjacent distinct vertices of a graph G, then the minimum size of a vertex-cut separating x from y equals the maximum number of pairwise internally-disjoint xy-paths.

Proof.

Let $\kappa(x, y)$ denote the minimum size of a vertex-cut separating x from y. Let $\lambda(x, y)$ denote the maximum number of pairwise internally-disjoint xy-paths.

$\square$

The Menger Theorem

Theorem 5.29 (Menger 1927)

If x and y are non-adjacent distinct vertices of a graph G, then the minimum size of a vertex-cut separating x from y equals the maximum number of pairwise internally-disjoint xy-paths.

Proof.

Let $\kappa(x, y)$ denote the minimum size of a vertex-cut separating x from y. Let $\lambda(x, y)$ denote the maximum number of pairwise internally-disjoint xy-paths. An vertex-cut separating x from y must contain an internal vertex from every xy-path, and so $\kappa(x, y) \geq \lambda(x, y)$.

$\square$

The Menger Theorem

Theorem 5.29 (Menger 1927)

If x and y are non-adjacent distinct vertices of a graph G, then the minimum size of a vertex-cut separating x from y equals the maximum number of pairwise internally-disjoint xy-paths.

Proof.

Let $\kappa(x, y)$ denote the minimum size of a vertex-cut separating x from y. Let $\lambda(x, y)$ denote the maximum number of pairwise internally-disjoint xy-paths.
An vertex-cut separating x from y must contain an internal vertex from every xy-path, and so $\kappa(x, y) \geq \lambda(x, y)$.
To prove the opposite inequality, we use induction on $|G|$.

□

The Menger Theorem

Theorem 5.29 (Menger 1927)

If x and y are non-adjacent distinct vertices of a graph G, then the minimum size of a vertex-cut separating x from y equals the maximum number of pairwise internally-disjoint xy-paths.

Proof.

Let $\kappa(x, y)$ denote the minimum size of a vertex-cut separating x from y. Let $\lambda(x, y)$ denote the maximum number of pairwise internally-disjoint xy-paths.
An vertex-cut separating x from y must contain an internal vertex from every xy-path, and so $\kappa(x, y) \geq \lambda(x, y)$.
To prove the opposite inequality, we use induction on $|G|$. If $|G| = 2$, then $\kappa(x, y) = \lambda(x, y) = 0$.

□

The Menger Theorem

Theorem 5.29 (Menger 1927)

If x and y are non-adjacent distinct vertices of a graph G, then the minimum size of a vertex-cut separating x from y equals the maximum number of pairwise internally-disjoint xy-paths.

Proof.

Let $\kappa(x, y)$ denote the minimum size of a vertex-cut separating x from y. Let $\lambda(x, y)$ denote the maximum number of pairwise internally-disjoint xy-paths. An vertex-cut separating x from y must contain an internal vertex from every xy-path, and so $\kappa(x, y) \geq \lambda(x, y)$.
To prove the opposite inequality, we use induction on $|G|$. If $|G| = 2$, then $\kappa(x, y) = \lambda(x, y) = 0$. For the induction step, suppose $|G| > 2$ and let $k = \kappa(x, y)$; we construct k pairwise internally-disjoint xy-paths. $\square$

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$.

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths.

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths. Clearly, $V_1 \cap V_2 = S$.

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths. Clearly, $V_1 \cap V_2 = S$. Form H_1 from the subgraph of G induced by V_1 by adding a vertex y' and connecting it to all edges in S.

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths. Clearly, $V_1 \cap V_2 = S$. Form H_1 from the subgraph of G induced by V_1 by adding a vertex y' and connecting it to all edges in S. Similarly, form H_2 from the subgraph of G induced by V_2 by adding a vertex x' and connecting it to all edges in S.

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths. Clearly, $V_1 \cap V_2 = S$. Form H_1 from the subgraph of G induced by V_1 by adding a vertex y' and connecting it to all edges in S. Similarly, form H_2 from the subgraph of G induced by V_2 by adding a vertex x' and connecting it to all edges in S. Every xy-path in G starts with an xS-path (which is contained in H_1),

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths. Clearly, $V_1 \cap V_2 = S$. Form H_1 from the subgraph of G induced by V_1 by adding a vertex y' and connecting it to all edges in S. Similarly, form H_2 from the subgraph of G induced by V_2 by adding a vertex x' and connecting it to all edges in S. Every xy-path in G starts with an xS-path (which is contained in H_1), so every xy'-cut in H_1 is an xy-cut in G.

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths. Clearly, $V_1 \cap V_2 = S$. Form H_1 from the subgraph of G induced by V_1 by adding a vertex y' and connecting it to all edges in S. Similarly, form H_2 from the subgraph of G induced by V_2 by adding a vertex x' and connecting it to all edges in S. Every xy-path in G starts with an xS-path (which is contained in H_1), so every xy'-cut in H_1 is an xy-cut in G. Hence $\kappa_{H_1}(x, y') = k$.

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths. Clearly, $V_1 \cap V_2 = S$. Form H_1 from the subgraph of G induced by V_1 by adding a vertex y' and connecting it to all edges in S. Similarly, form H_2 from the subgraph of G induced by V_2 by adding a vertex x' and connecting it to all edges in S. Every xy-path in G starts with an xS-path (which is contained in H_1), so every xy'-cut in H_1 is an xy-cut in G. Hence $\kappa_{H_1}(x, y') = k$. Hence $\kappa_{H_2}(x', y) = k$.

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths. Clearly, $V_1 \cap V_2 = S$. Form H_1 from the subgraph of G induced by V_1 by adding a vertex y' and connecting it to all edges in S. Similarly, form H_2 from the subgraph of G induced by V_2 by adding a vertex x' and connecting it to all edges in S. Every xy-path in G starts with an xS-path (which is contained in H_1), so every xy'-cut in H_1 is an xy-cut in G. Hence $\kappa_{H_1}(x, y') = k$. Hence $\kappa_{H_2}(x', y) = k$. Since V_1 omits $y \cup N(y) - S$ and V_2 omits $x \cup N(x) - S$, each of H_1 and H_2 has fewer vertices than G.

Proof of the Menger Theorem, Case 1

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths. Clearly, $V_1 \cap V_2 = S$. Form H_1 from the subgraph of G induced by V_1 by adding a vertex y' and connecting it to all edges in S. Similarly, form H_2 from the subgraph of G induced by V_2 by adding a vertex x' and connecting it to all edges in S. Every xy-path in G starts with an xS-path (which is contained in H_1), so every xy'-cut in H_1 is an xy-cut in G. Hence $\kappa_{H_1}(x, y') = k$. Hence $\kappa_{H_2}(x', y) = k$. Since V_1 omits $y \cup N(y) - S$ and V_2 omits $x \cup N(x) - S$, each of H_1 and H_2 has fewer vertices than G. Hence the induction hypothesis gives $\lambda_{H_1}(x, y') = k = \lambda_{H_2}(x', y)$.

Case 1: G has a minimum xy-vertex-cut S not containing $N(x)$ and not containing $N(y)$. Let V_1 be the set of vertices on xS-paths, and let V_2 be the set of vertices on Sy-paths. Clearly, $V_1 \cap V_2 = S$. Form H_1 from the subgraph of G induced by V_1 by adding a vertex y' and connecting it to all edges in S. Similarly, form H_2 from the subgraph of G induced by V_2 by adding a vertex x' and connecting it to all edges in S. Every xy-path in G starts with an xS-path (which is contained in H_1), so every xy'-cut in H_1 is an xy-cut in G. Hence $\kappa_{H_1}(x, y') = k$. Hence $\kappa_{H_2}(x', y) = k$. Since V_1 omits $y \cup N(y) - S$ and V_2 omits $x \cup N(x) - S$, each of H_1 and H_2 has fewer vertices than G. Hence the induction hypothesis gives $\lambda_{H_1}(x, y') = k = \lambda_{H_2}(x', y)$. Combining k xS-paths in H_1 that meet only at x with k Sy-paths in H_2 that meet only in y, we get k pairwise internally-disjoint xy-paths in G.

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$.

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.
If G has a vertex v outside $\{x, y\} \cup N(x) \cup N(y)$, then v is in no minimum xy-vertex-cut;

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.

If G has a vertex v outside $\{x, y\} \cup N(x) \cup N(y)$, then v is in no minimum xy-vertex-cut; hence $\kappa_{G-v}(x, y) = k$ and applying the induction hypothesis to $G - v$ yields the desired xy-paths in G.

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.

If G has a vertex v outside $\{x, y\} \cup N(x) \cup N(y)$, then v is in no minimum xy-vertex-cut; hence $\kappa_{G-v}(x, y) = k$ and applying the induction hypothesis to $G - v$ yields the desired xy-paths in G.

If $v \in N(x) \cap N(y)$, then v appears in every xy-vertex-cut,

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.
If G has a vertex v outside $\{x, y\} \cup N(x) \cup N(y)$, then v is in no minimum xy-vertex-cut; hence $\kappa_{G-v}(x, y) = k$ and applying the induction hypothesis to $G - v$ yields the desired xy-paths in G.
If $v \in N(x) \cap N(y)$, then v appears in every xy-vertex-cut, and so $k_{G-v}(x, y) = k - 1$.

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.

If G has a vertex v outside $\{x, y\} \cup N(x) \cup N(y)$, then v is in no minimum xy-vertex-cut; hence $\kappa_{G-v}(x, y) = k$ and applying the induction hypothesis to $G - v$ yields the desired xy-paths in G.

If $v \in N(x) \cap N(y)$, then v appears in every xy-vertex-cut, and so $k_{G-v}(x, y) = k - 1$. Now, applying the induction hypothesis to $G - v$ yields $k - 1$ xy-paths, which combine with the path xvy to get the desired k paths.

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.
If G has a vertex v outside $\{x, y\} \cup N(x) \cup N(y)$, then v is in no minimum xy-vertex-cut; hence $\kappa_{G-v}(x, y) = k$ and applying the induction hypothesis to $G - v$ yields the desired xy-paths in G.
If $v \in N(x) \cap N(y)$, then v appears in every xy-vertex-cut, and so $k_{G-v}(x, y) = k - 1$. Now, applying the induction hypothesis to $G - v$ yields $k - 1$ xy-paths, which combine with the path xvy to get the desired k paths.
We may thus assume that $N(x)$ and $N(y)$ are disjoint and together with $\{x, y\}$ form the entire $V(G)$.

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.

If G has a vertex v outside $\{x, y\} \cup N(x) \cup N(y)$, then v is in no minimum xy-vertex-cut; hence $\kappa_{G-v}(x, y) = k$ and applying the induction hypothesis to $G - v$ yields the desired xy-paths in G.

If $v \in N(x) \cap N(y)$, then v appears in every xy-vertex-cut, and so $k_{G-v}(x, y) = k - 1$. Now, applying the induction hypothesis to $G - v$ yields $k - 1$ xy-paths, which combine with the path xvy to get the desired k paths.

We may thus assume that $N(x)$ and $N(y)$ are disjoint and together with $\{x, y\}$ form the entire $V(G)$. Let G' be the bipartite graph with bipartition $N(x)$ and $N(y)$ and the edge set $[N(x), N(y)]$.

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.
If G has a vertex v outside $\{x, y\} \cup N(x) \cup N(y)$, then v is in no minimum xy-vertex-cut; hence $\kappa_{G-v}(x, y) = k$ and applying the induction hypothesis to $G - v$ yields the desired xy-paths in G.
If $v \in N(x) \cap N(y)$, then v appears in every xy-vertex-cut, and so $k_{G-v}(x, y) = k - 1$. Now, applying the induction hypothesis to $G - v$ yields $k - 1$ xy-paths, which combine with the path xvy to get the desired k paths.
We may thus assume that $N(x)$ and $N(y)$ are disjoint and together with $\{x, y\}$ form the entire $V(G)$. Let G' be the bipartite graph with bipartition $N(x)$ and $N(y)$ and the edge set $[N(x), N(y)]$. Every xy-path in G uses some edge of G', and so the xy-vertex-cuts are precisely the vertex covers of G'.

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.

If G has a vertex v outside $\{x, y\} \cup N(x) \cup N(y)$, then v is in no minimum xy-vertex-cut; hence $\kappa_{G-v}(x, y) = k$ and applying the induction hypothesis to $G - v$ yields the desired xy-paths in G.

If $v \in N(x) \cap N(y)$, then v appears in every xy-vertex-cut, and so $k_{G-v}(x, y) = k - 1$. Now, applying the induction hypothesis to $G - v$ yields $k - 1$ xy-paths, which combine with the path xvy to get the desired k paths.

We may thus assume that $N(x)$ and $N(y)$ are disjoint and together with $\{x, y\}$ form the entire $V(G)$. Let G' be the bipartite graph with bipartition $N(x)$ and $N(y)$ and the edge set $[N(x), N(y)]$. Every xy-path in G uses some edge of G', and so the xy-vertex-cuts are precisely the vertex covers of G'. By the König-Egerváry Theorem 3.13, G has a matching of size k.

Proof of the Menger Theorem, Case 2

Case 2: Every minimum xy-vertex-cut contains $N(x)$ or $N(y)$. Since $N(x)$ and $N(y)$ are themselves xy-vertex-cuts, the condition implies that that there are no minimum xy-vertex-cuts other than $N(x)$ and $N(y)$.
If G has a vertex v outside $\{x, y\} \cup N(x) \cup N(y)$, then v is in no minimum xy-vertex-cut; hence $\kappa_{G-v}(x, y) = k$ and applying the induction hypothesis to $G - v$ yields the desired xy-paths in G.
If $v \in N(x) \cap N(y)$, then v appears in every xy-vertex-cut, and so $k_{G-v}(x, y) = k - 1$. Now, applying the induction hypothesis to $G - v$ yields $k - 1$ xy-paths, which combine with the path xvy to get the desired k paths.
We may thus assume that $N(x)$ and $N(y)$ are disjoint and together with $\{x, y\}$ form the entire $V(G)$. Let G' be the bipartite graph with bipartition $N(x)$ and $N(y)$ and the edge set $[N(x), N(y)]$. Every xy-path in G uses some edge of G', and so the xy-vertex-cuts are precisely the vertex covers of G'. By the König-Egerváry Theorem 3.13, G has a matching of size k. The edges of the matching together with the edges incident with x and y form the desired k paths.

The Edge Version of Menger's Theorem

Theorem 5.30 (Edge Version of Menger's Theorem)

If x and y are distinct vertices of a graph, then the minimum size $\kappa'(x,y)$ of the set of edges that separate x from y equals the maximum number $\lambda'(x,y)$ of pairwise edge-disjoint xy-paths.

The Edge Version of Menger's Theorem

Theorem 5.30 (Edge Version of Menger's Theorem)

If x and y are distinct vertices of a graph, then the minimum size $\kappa'(x, y)$ of the set of edges that separate x from y equals the maximum number $\lambda'(x, y)$ of pairwise edge-disjoint xy-paths.

Definition 5.31

The line graph of a graph G, written $L(G)$, is a simple graph whose vertex set is $E(G)$ with two vertices adjacent if the corresponding edges are adjacent in G.

The Edge Version of Menger's Theorem

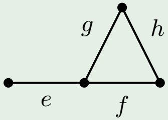
Theorem 5.30 (Edge Version of Menger's Theorem)

If x and y are distinct vertices of a graph, then the minimum size $\kappa'(x, y)$ of the set of edges that separate x from y equals the maximum number $\lambda'(x, y)$ of pairwise edge-disjoint xy-paths.

Definition 5.31

The line graph of a graph G, written $L(G)$, is a simple graph whose vertex set is $E(G)$ with two vertices adjacent if the corresponding edges are adjacent in G.

Example 5.32



The Edge Version of Menger's Theorem

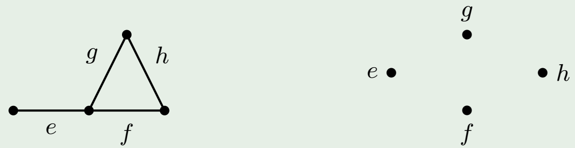
Theorem 5.30 (Edge Version of Menger's Theorem)

If x and y are distinct vertices of a graph, then the minimum size $\kappa'(x, y)$ of the set of edges that separate x from y equals the maximum number $\lambda'(x, y)$ of pairwise edge-disjoint xy-paths.

Definition 5.31

The line graph of a graph G, written $L(G)$, is a simple graph whose vertex set is $E(G)$ with two vertices adjacent if the corresponding edges are adjacent in G.

Example 5.32



The Edge Version of Menger's Theorem

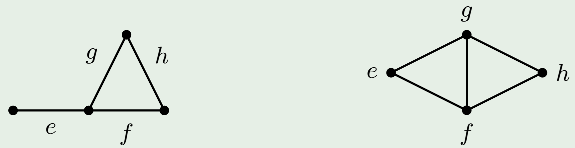
Theorem 5.30 (Edge Version of Menger's Theorem)

If x and y are distinct vertices of a graph, then the minimum size $\kappa'(x, y)$ of the set of edges that separate x from y equals the maximum number $\lambda'(x, y)$ of pairwise edge-disjoint xy-paths.

Definition 5.31

The line graph of a graph G, written $L(G)$, is a simple graph whose vertex set is $E(G)$ with two vertices adjacent if the corresponding edges are adjacent in G.

Example 5.32



Proof of Edge Version of Menger's Theorem

Modify G to obtain G' by adding two vertices s and t and two new edges sx and yt.

Proof of Edge Version of Menger's Theorem

Modify G to obtain G' by adding two vertices s and t and two new edges sx and yt. This operation does not affect $\kappa'(x, y)$ and $\lambda'(x, y)$.

Proof of Edge Version of Menger's Theorem

Modify G to obtain G' by adding two vertices s and t and two new edges sx and yt. This operation does not affect $\kappa'(x, y)$ and $\lambda'(x, y)$. A set of edges disconnects x from y in G if and only if the corresponding vertices of $L(G')$ form a set of edges separating sx from yt.

Proof of Edge Version of Menger's Theorem

Modify G to obtain G' by adding two vertices s and t and two new edges sx and yt. This operation does not affect $\kappa'(x,y)$ and $\lambda'(x,y)$. A set of edges disconnects x from y in G if and only if the corresponding vertices of $L(G')$ form a set of edges separating sx from yt. Similarly, edge-disjoint xy-paths in G become internally-disjoint paths from sx to yt in $L(G')$, and vice versa.

Proof of Edge Version of Menger's Theorem

Modify G to obtain G' by adding two vertices s and t and two new edges sx and yt. This operation does not affect $\kappa'(x, y)$ and $\lambda'(x, y)$. A set of edges disconnects x from y in G if and only if the corresponding vertices of $L(G')$ form a set of edges separating sx from yt. Similarly, edge-disjoint xy-paths in G become internally-disjoint paths from sx to yt in $L(G')$, and vice versa. Applying the vertex version of the Menger Theorem 5.29 yields $\kappa'_G(x, y) = \kappa_{L(G')}(sx, yt) = \lambda_{L(G')}(sx, yt) = \lambda'_G(x, y)$.

Proof of Edge Version of Menger's Theorem

Modify G to obtain G' by adding two vertices s and t and two new edges sx and yt. This operation does not affect $\kappa'(x,y)$ and $\lambda'(x,y)$. A set of edges disconnects x from y in G if and only if the corresponding vertices of $L(G')$ form a set of edges separating sx from yt. Similarly, edge-disjoint xy-paths in G become internally-disjoint paths from sx to yt in $L(G')$, and vice versa. Applying the vertex version of the Menger Theorem 5.29 yields $\kappa'_G(x,y) = \kappa_{L(G')}(sx,yt) = \lambda_{L(G')}(sx,yt) = \lambda'_G(x,y)$.

Theorem 5.33

The connectivity of G equals the maximum k such that $\lambda(x,y) \geq k$ for all vertices x and y of G.

Proof of Edge Version of Menger's Theorem

Modify G to obtain G' by adding two vertices s and t and two new edges sx and yt. This operation does not affect $\kappa'(x, y)$ and $\lambda'(x, y)$. A set of edges disconnects x from y in G if and only if the corresponding vertices of $L(G')$ form a set of edges separating sx from yt. Similarly, edge-disjoint xy-paths in G become internally-disjoint paths from sx to yt in $L(G')$, and vice versa. Applying the vertex version of the Menger Theorem 5.29 yields
$\kappa'_G(x, y) = \kappa_{L(G')}(sx, yt) = \lambda_{L(G')}(sx, yt) = \lambda'_G(x, y)$.

Theorem 5.33

The connectivity of G equals the maximum k such that $\lambda(x, y) \geq k$ for all vertices x and y of G. The edge connectivity of G equals the maximum k such that $\lambda'(x, y) \geq k$ for all vertices x and y of G.

Proof of Edge Version of Menger's Theorem

Modify G to obtain G' by adding two vertices s and t and two new edges sx and yt. This operation does not affect $\kappa'(x, y)$ and $\lambda'(x, y)$. A set of edges disconnects x from y in G if and only if the corresponding vertices of $L(G')$ form a set of edges separating sx from yt. Similarly, edge-disjoint xy-paths in G become internally-disjoint paths from sx to yt in $L(G')$, and vice versa. Applying the vertex version of the Menger Theorem 5.29 yields $\kappa'_G(x, y) = \kappa_{L(G')}(sx, yt) = \lambda_{L(G')}(sx, yt) = \lambda'_G(x, y)$.

Theorem 5.33

The connectivity of G equals the maximum k such that $\lambda(x, y) \geq k$ for all vertices x and y of G. The edge connectivity of G equals the maximum k such that $\lambda'(x, y) \geq k$ for all vertices x and y of G.

Proof.

The edge version follows immediately from Theorem 5.30 since $\kappa'(G) = \min_{x,y \in V(G)} \kappa'(x, y)$.

$\square$

Proof of Edge Version of Menger's Theorem

Modify G to obtain G' by adding two vertices s and t and two new edges sx and yt. This operation does not affect $\kappa'(x,y)$ and $\lambda'(x,y)$. A set of edges disconnects x from y in G if and only if the corresponding vertices of $L(G')$ form a set of edges separating sx from yt. Similarly, edge-disjoint xy-paths in G become internally-disjoint paths from sx to yt in $L(G')$, and vice versa. Applying the vertex version of the Menger Theorem 5.29 yields $\kappa'_G(x,y) = \kappa_{L(G')}(sx,yt) = \lambda_{L(G')}(sx,yt) = \lambda'_G(x,y)$.

Theorem 5.33

The connectivity of G equals the maximum k such that $\lambda(x,y) \geq k$ for all vertices x and y of G. The edge connectivity of G equals the maximum k such that $\lambda'(x,y) \geq k$ for all vertices x and y of G.

Proof.

The edge version follows immediately from Theorem 5.30 since $\kappa'(G) = \min_{x,y\in V(G)} \kappa'(x,y)$. For connectivity, we get $\kappa(x,y) = \lambda(x,y)$ if x and y are non-adjacent, and $\kappa(G)$ is the minimum of these values.

□

Proof of Edge Version of Menger's Theorem

Modify G to obtain G' by adding two vertices s and t and two new edges sx and yt. This operation does not affect $\kappa'(x, y)$ and $\lambda'(x, y)$. A set of edges disconnects x from y in G if and only if the corresponding vertices of $L(G')$ form a set of edges separating sx from yt. Similarly, edge-disjoint xy-paths in G become internally-disjoint paths from sx to yt in $L(G')$, and vice versa. Applying the vertex version of the Menger Theorem 5.29 yields $\kappa'_G(x, y) = \kappa_{L(G')}(sx, yt) = \lambda_{L(G')}(sx, yt) = \lambda'_G(x, y)$.

Theorem 5.33

The connectivity of G equals the maximum k such that $\lambda(x, y) \geq k$ for all vertices x and y of G. The edge connectivity of G equals the maximum k such that $\lambda'(x, y) \geq k$ for all vertices x and y of G.

Proof.

The edge version follows immediately from Theorem 5.30 since $\kappa'(G) = \min_{x,y \in V(G)} \kappa'(x, y)$. For connectivity, we get $\kappa(x, y) = \lambda(x, y)$ if x and y are non-adjacent, and $\kappa(G)$ is the minimum of these values. If x and y are adjacent, we get $\lambda_G(x, y) = 1 + \lambda_{G \setminus xy}(x, y) = 1 + \kappa_{G \setminus xy}(x, y) \geq 1 + \kappa(G \setminus xy) \geq \kappa(G)$. □

Tutte's Wheel Theorem

Theorem 5.34 (Tutte's Wheel Theorem)

If G is a Tutte-3-connected graph on at least four vertices that is not a wheel, then there is an edge e of G such that at least one of G/e and $G \setminus e$ is also Tutte-3-connected.

Theorem 5.34 (Tutte's Wheel Theorem)

If G is a Tutte-3-connected graph on at least four vertices that is not a wheel, then there is an edge e of G such that at least one of G/e and $G \setminus e$ is also Tutte-3-connected.

Lemma 5.35 (Thomassen 1980)

Every 3-connected graph G on at least five vertices has an edge e such that G/e is 3-connected.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut. Then $\{x, y, z\}$ is a vertex cut in G.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut. Then $\{x, y, z\}$ is a vertex cut in G. Choose $e = xy$ and the corresponding z so that the graph $G - \{x, y, z\}$ has a component H with the largest possible order.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut. Then $\{x, y, z\}$ is a vertex cut in G. Choose $e = xy$ and the corresponding z so that the graph $G - \{x, y, z\}$ has a component H with the largest possible order. Let H' be another component of $G - \{x, y, z\}$.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut. Then $\{x, y, z\}$ is a vertex cut in G. Choose $e = xy$ and the corresponding z so that the graph $G - \{x, y, z\}$ has a component H with the largest possible order. Let H' be another component of $G - \{x, y, z\}$. Since $\{x, y, z\}$ is a minimal vertex cut, each of x, y, and z has a neighbor in ech of H and H'.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut. Then $\{x, y, z\}$ is a vertex cut in G. Choose $e = xy$ and the corresponding z so that the graph $G - \{x, y, z\}$ has a component H with the largest possible order. Let H' be another component of $G - \{x, y, z\}$. Since $\{x, y, z\}$ is a minimal vertex cut, each of x, y, and z has a neighbor in ech of H and H'. Let u be a neighbor of z in H',

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut. Then $\{x, y, z\}$ is a vertex cut in G. Choose $e = xy$ and the corresponding z so that the graph $G - \{x, y, z\}$ has a component H with the largest possible order. Let H' be another component of $G - \{x, y, z\}$. Since $\{x, y, z\}$ is a minimal vertex cut, each of x, y, and z has a neighbor in ech of H and H'. Let u be a neighbor of z in H', and let v be a vertex such that $\{z, u, v\}$ disconnects G.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut. Then $\{x, y, z\}$ is a vertex cut in G. Choose $e = xy$ and the corresponding z so that the graph $G - \{x, y, z\}$ has a component H with the largest possible order. Let H' be another component of $G - \{x, y, z\}$. Since $\{x, y, z\}$ is a minimal vertex cut, each of x, y, and z has a neighbor in ech of H and H'. Let u be a neighbor of z in H', and let v be a vertex such that $\{z, u, v\}$ disconnects G. The subgraph of G induced by $V(H) \cup \{x, y\}$ is connected.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut. Then $\{x, y, z\}$ is a vertex cut in G. Choose $e = xy$ and the corresponding z so that the graph $G - \{x, y, z\}$ has a component H with the largest possible order. Let H' be another component of $G - \{x, y, z\}$. Since $\{x, y, z\}$ is a minimal vertex cut, each of x, y, and z has a neighbor in ech of H and H'. Let u be a neighbor of z in H', and let v be a vertex such that $\{z, u, v\}$ disconnects G. The subgraph of G induced by $V(H) \cup \{x, y\}$ is connected. Deleting v from this subgraph (if it occurs there) cannot disconnect it,

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut. Then $\{x, y, z\}$ is a vertex cut in G. Choose $e = xy$ and the corresponding z so that the graph $G - \{x, y, z\}$ has a component H with the largest possible order. Let H' be another component of $G - \{x, y, z\}$. Since $\{x, y, z\}$ is a minimal vertex cut, each of x, y, and z has a neighbor in ech of H and H'. Let u be a neighbor of z in H', and let v be a vertex such that $\{z, u, v\}$ disconnects G. The subgraph of G induced by $V(H) \cup \{x, y\}$ is connected. Deleting v from this subgraph (if it occurs there) cannot disconnect it, since then $\{z, v\}$ would disconnect G.

Proof of the Lemma

Assume that for each edge e the graph G/e is not 3-connected, and so has a 2-element vertex cut. Since G is 3-connected, one of the elements of this vertex cut must come from contracting $e = xy$. Let z be the other element of this vertex cut. Then $\{x, y, z\}$ is a vertex cut in G. Choose $e = xy$ and the corresponding z so that the graph $G - \{x, y, z\}$ has a component H with the largest possible order. Let H' be another component of $G - \{x, y, z\}$. Since $\{x, y, z\}$ is a minimal vertex cut, each of x, y, and z has a neighbor in ech of H and H'. Let u be a neighbor of z in H', and let v be a vertex such that $\{z, u, v\}$ disconnects G. The subgraph of G induced by $V(H) \cup \{x, y\}$ is connected. Deleting v from this subgraph (if it occurs there) cannot disconnect it, since then $\{z, v\}$ would disconnect G. Therefore all elements of $V(H) \cup \{x, y\} - v$ belong to the same component of $G - \{z, u, v\}$, which has more vertices than H; a contradiction.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) v_0, v_1, $\ldots$, v_k, and a vertex c adjacent to all vertices of the path;

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) v_0, v_1, ..., v_k, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) v_0, v_1, ..., v_k, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;
- each of v_1, v_2, ..., v_{k-1} has degree 3 in G; and

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) $v_0, v_1, \ldots, v_k$, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;
- each of $v_1, v_2, \ldots, v_{k-1}$ has degree 3 in G; and
- k is maximal, subject to the conditions above.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) $v_0, v_1, \ldots, v_k$, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;
- each of $v_1, v_2, \ldots, v_{k-1}$ has degree 3 in G; and
- k is maximal, subject to the conditions above.

Lemma 5.35 implies that $k \geq 1$.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) v_0, v_1, ..., v_k, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;
- each of v_1, v_2, ..., v_{k-1} has degree 3 in G; and
- k is maximal, subject to the conditions above.

Lemma 5.35 implies that $k \geq 1$. The graph $G \setminus cv_k$ is not 3-connected, so it has a vertex cut of size 2.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) $v_0, v_1, \ldots, v_k$, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;
- each of $v_1, v_2, \ldots, v_{k-1}$ has degree 3 in G; and
- k is maximal, subject to the conditions above.

Lemma 5.35 implies that $k \geq 1$. The graph $G \setminus cv_k$ is not 3-connected, so it has a vertex cut of size 2. One of the cut vertices is v_{k-1}; let's name the other one u.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) v_0, v_1, ..., v_k, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;
- each of v_1, v_2, ..., v_{k-1} has degree 3 in G; and
- k is maximal, subject to the conditions above.

Lemma 5.35 implies that $k \geq 1$. The graph $G \setminus cv_k$ is not 3-connected, so it has a vertex cut of size 2. One of the cut vertices is v_{k-1}; let's name the other one u. Vertices c and v_k are in distinct components A and B, respectively, of $G \setminus cv_k - \{v_{k-1}, u\}$.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) $v_0, v_1, \ldots, v_k$, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;
- each of $v_1, v_2, \ldots, v_{k-1}$ has degree 3 in G; and
- k is maximal, subject to the conditions above.

Lemma 5.35 implies that $k \geq 1$. The graph $G \setminus cv_k$ is not 3-connected, so it has a vertex cut of size 2. One of the cut vertices is v_{k-1}; let's name the other one u. Vertices c and v_k are in distinct components A and B, respectively, of $G \setminus cv_k - \{v_{k-1}, u\}$. Consider $G/v_{k-1}v_k$. It is 3-connected,

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) $v_0, v_1, \ldots, v_k$, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;
- each of $v_1, v_2, \ldots, v_{k-1}$ has degree 3 in G; and
- k is maximal, subject to the conditions above.

Lemma 5.35 implies that $k \geq 1$. The graph $G \setminus cv_k$ is not 3-connected, so it has a vertex cut of size 2. One of the cut vertices is v_{k-1}; let's name the other one u. Vertices c and v_k are in distinct components A and B, respectively, of $G \setminus cv_k - \{v_{k-1}, u\}$. Consider $G/v_{k-1}v_k$. It is 3-connected, which implies that v_k is the only vertex of B, and so v_k also has degree 3.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) $v_0, v_1, \ldots, v_k$, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;
- each of $v_1, v_2, \ldots, v_{k-1}$ has degree 3 in G; and
- k is maximal, subject to the conditions above.

Lemma 5.35 implies that $k \geq 1$. The graph $G \setminus cv_k$ is not 3-connected, so it has a vertex cut of size 2. One of the cut vertices is v_{k-1}; let's name the other one u. Vertices c and v_k are in distinct components A and B, respectively, of $G \setminus cv_k - \{v_{k-1}, u\}$. Consider $G/v_{k-1}v_k$. It is 3-connected, which implies that v_k is the only vertex of B, and so v_k also has degree 3. Note that each of v_1, $v_2, \ldots, v_{k-2}$ has degree 3 and is not adjacent to v_k, which implies that u is not among those vertices.

Proof of Tutte's Wheel Theorem

Every T3C graph G on at least 4 vertices has an edge e such that G/e or $G \setminus e$ is T3C, unless G is a wheel.

Suppose G is T3C and has at least 4 vertices, but has no edge whose deletion or contraction results in a T3C graph. If $|G| = 4$, then $G \cong K_4$, and the conclusion holds; so we may assume that G has at least 5 vertices. Then Lemma 5.35 implies that G has an edge whose contraction results in a 3-connected graph. Let F (a k-fan) be a subgraph of G such that:

- F consists of a path on vertices (listed in order) v_0, v_1, ..., v_k, and a vertex c adjacent to all vertices of the path;
- $G/v_{k-1}v_k$ is 3-connected;
- each of v_1, v_2, ..., v_{k-1} has degree 3 in G; and
- k is maximal, subject to the conditions above.

Lemma 5.35 implies that $k \geq 1$. The graph $G \setminus cv_k$ is not 3-connected, so it has a vertex cut of size 2. One of the cut vertices is v_{k-1}; let's name the other one u. Vertices c and v_k are in distinct components A and B, respectively, of $G \setminus cv_k - \{v_{k-1}, u\}$. Consider $G/v_{k-1}v_k$. It is 3-connected, which implies that v_k is the only vertex of B, and so v_k also has degree 3. Note that each of v_1, v_2, ..., v_{k-2} has degree 3 and is not adjacent to v_k, which implies that u is not among those vertices. Clearly, $u \neq v_k$ and $u \neq c$.

Proof, continued

Case 1: $u \neq v_0$

Proof, continued

Case 1: $u \neq v_0$
Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k.

Proof, continued

Case 1: $u \neq v_0$
Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction.

Proof, continued

Case 1: $u \neq v_0$

Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple.

Proof, continued

Case 1: $u \neq v_0$
Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple. One possibility of this happening is that u, c, and v_k form a triangle,

Proof, continued

Case 1: $u \neq v_0$
Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple. One possibility of this happening is that u, c, and v_k form a triangle, but that would imply that G has a larger k-fan; a contradiction.

Proof, continued

Case 1: $u \neq v_0$

Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple. One possibility of this happening is that u, c, and v_k form a triangle, but that would imply that G has a larger k-fan; a contradiction. Otherwise, u, v_{k-1}, and v_k form a triangle.

Proof, continued

Case 1: $u \neq v_0$

Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple. One possibility of this happening is that u, c, and v_k form a triangle, but that would imply that G has a larger k-fan; a contradiction. Otherwise, u, v_{k-1}, and v_k form a triangle. But then, if $k-1>0$, the vertex v_{k-1} would be adjacent to v_{k-2}, v_k, c, and u, which is impossible, the degree of v_{k-1} is 3.

Proof, continued

Case 1: $u \neq v_0$

Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple. One possibility of this happening is that u, c, and v_k form a triangle, but that would imply that G has a larger k-fan; a contradiction. Otherwise, u, v_{k-1}, and v_k form a triangle. But then, if $k-1>0$, the vertex v_{k-1} would be adjacent to v_{k-2}, v_k, c, and u, which is impossible, the degree of v_{k-1} is 3. So $k-1=0$.

Proof, continued

Case 1: $u \neq v_0$
Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple. One possibility of this happening is that u, c, and v_k form a triangle, but that would imply that G has a larger k-fan; a contradiction. Otherwise, u, v_{k-1}, and v_k form a triangle. But then, if $k-1>0$, the vertex v_{k-1} would be adjacent to v_{k-2}, v_k, c, and u, which is impossible, the degree of v_{k-1} is 3. So $k-1=0$. In that case, however, v_0, v_1, c, and u form a 2-fan; again a contradiction.

Proof, continued

Case 1: $u \neq v_0$
Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple. One possibility of this happening is that u, c, and v_k form a triangle, but that would imply that G has a larger k-fan; a contradiction. Otherwise, u, v_{k-1}, and v_k form a triangle. But then, if $k-1>0$, the vertex v_{k-1} would be adjacent to v_{k-2}, v_k, c, and u, which is impossible, the degree of v_{k-1} is 3. So $k-1=0$. In that case, however, v_0, v_1, c, and u form a 2-fan; again a contradiction.
Case 2: $u = v_0$

Proof, continued

Case 1: $u \neq v_0$
Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple. One possibility of this happening is that u, c, and v_k form a triangle, but that would imply that G has a larger k-fan; a contradiction. Otherwise, u, v_{k-1}, and v_k form a triangle. But then, if $k-1>0$, the vertex v_{k-1} would be adjacent to v_{k-2}, v_k, c, and u, which is impossible, the degree of v_{k-1} is 3. So $k-1=0$. In that case, however, v_0, v_1, c, and u form a 2-fan; again a contradiction.
Case 2: $u = v_0$
Note that v_1, v_2, $\ldots$, v_k have degree 3.

Proof, continued

Case 1: $u \neq v_0$
Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple. One possibility of this happening is that u, c, and v_k form a triangle, but that would imply that G has a larger k-fan; a contradiction. Otherwise, u, v_{k-1}, and v_k form a triangle. But then, if $k-1>0$, the vertex v_{k-1} would be adjacent to v_{k-2}, v_k, c, and u, which is impossible, the degree of v_{k-1} is 3. So $k-1=0$. In that case, however, v_0, v_1, c, and u form a 2-fan; again a contradiction.
Case 2: $u = v_0$
Note that $v_1, v_2, \ldots, v_k$ have degree 3. So if G contained another vertex, say z, it would be disconnected from $v_1, v_2, \ldots, v_k$, by deleting v_0 and c; which is impossible.

Proof, continued

Case 1: $u \neq v_0$
Look at G/uv_k. If this graph is not 3-connected, then G has a vertex-cut of size 3 containing u and v_k. But we concluded that the degree of v_k is 3, which means that the vertex cut containing u and v_k separates v_{k-1} from c, which are adjacent; a contradiction. This means that G/uv_k is not simple. One possibility of this happening is that u, c, and v_k form a triangle, but that would imply that G has a larger k-fan; a contradiction. Otherwise, u, v_{k-1}, and v_k form a triangle. But then, if $k-1>0$, the vertex v_{k-1} would be adjacent to v_{k-2}, v_k, c, and u, which is impossible, the degree of v_{k-1} is 3. So $k-1=0$. In that case, however, v_0, v_1, c, and u form a 2-fan; again a contradiction.
Case 2: $u = v_0$
Note that $v_1, v_2, \ldots, v_k$ have degree 3. So if G contained another vertex, say z, it would be disconnected from $v_1, v_2, \ldots, v_k$, by deleting v_0 and c; which is impossible. It follows that G is a $(k+1)$-wheel, which completes the proof.

Clique Sums

Definition 5.36

- A clique-sum of two graphs G and H is obtained from the disjoint union of G and H by identifying a complete subgraph of G with a complete subgraph (of the same order) of H, and then deleting the edges of the identified subgraph.

Clique Sums

Definition 5.36

- A clique-sum of two graphs G and H is obtained from the disjoint union of G and H by identifying a complete subgraph of G with a complete subgraph (of the same order) of H, and then deleting the edges of the identified subgraph.
- If the identified complete subgraph has order k, then the clique-sum is called k-sum and written $G \oplus_k H$.

Clique Sums

Definition 5.36

- A clique-sum of two graphs G and H is obtained from the disjoint union of G and H by identifying a complete subgraph of G with a complete subgraph (of the same order) of H, and then deleting the edges of the identified subgraph.
- If the identified complete subgraph has order k, then the clique-sum is called k-sum and written $G \oplus_k H$.

Example 5.37

- A 0-sum is a disjoint union.

Clique Sums

Definition 5.36

- A clique-sum of two graphs G and H is obtained from the disjoint union of G and H by identifying a complete subgraph of G with a complete subgraph (of the same order) of H, and then deleting the edges of the identified subgraph.
- If the identified complete subgraph has order k, then the clique-sum is called k-sum and written $G \oplus_k H$.

Example 5.37

- A 0-sum is a disjoint union.
- A 1-sum consists of two subgraphs that share exactly one vertex.

Clique Sums

Definition 5.36

- A clique-sum of two graphs G and H is obtained from the disjoint union of G and H by identifying a complete subgraph of G with a complete subgraph (of the same order) of H, and then deleting the edges of the identified subgraph.
- If the identified complete subgraph has order k, then the clique-sum is called k-sum and written $G \oplus_k H$.

Example 5.37

- A 0-sum is a disjoint union.
- A 1-sum consists of two subgraphs that share exactly one vertex.
- Every graph can be obtained by repeatedly 0-summing graphs, starting with connected graphs.

Clique Sums

Definition 5.36

- A clique-sum of two graphs G and H is obtained from the disjoint union of G and H by identifying a complete subgraph of G with a complete subgraph (of the same order) of H, and then deleting the edges of the identified subgraph.
- If the identified complete subgraph has order k, then the clique-sum is called k-sum and written $G \oplus_k H$.

Example 5.37

- A 0-sum is a disjoint union.
- A 1-sum consists of two subgraphs that share exactly one vertex.
- Every graph can be obtained by repeatedly 0-summing graphs, starting with connected graphs.
- Every connected graph can be obtained by repeatedly 1-summing graphs, starting with blocks.

Decomposition of 2-Connected Graphs

Definition 5.38

A 3-block is a cycle of length at least 3,

Decomposition of 2-Connected Graphs

Definition 5.38

A 3-block is a cycle of length at least 3, a loopless graph on two vertices with at least 3 edges between them (co-cycles), or

Decomposition of 2-Connected Graphs

Definition 5.38

A 3-block is a cycle of length at least 3, a loopless graph on two vertices with at least 3 edges between them (co-cycles), or Tutte-3-connected graph.

Decomposition of 2-Connected Graphs

Definition 5.38

A 3-block is a cycle of length at least 3, a loopless graph on two vertices with at least 3 edges between them (co-cycles), or Tutte-3-connected graph.

Theorem 5.39

Every Tutte-2-connected graph of size at least 3 can be obtained by repeatedly 2-summing graphs, starting with 3-blocks.

Decomposition of 2-Connected Graphs

Definition 5.38

A 3-block is a cycle of length at least 3, a loopless graph on two vertices with at least 3 edges between them (co-cycles), or Tutte-3-connected graph.

Theorem 5.39

Every Tutte-2-connected graph of size at least 3 can be obtained by repeatedly 2-summing graphs, starting with 3-blocks. Moreover, in this process, no two cycles are 2-summed together, and two co-cycles are 2-summed together.

Decomposition of 2-Connected Graphs

Definition 5.38

A 3-block is a cycle of length at least 3, a loopless graph on two vertices with at least 3 edges between them (co-cycles), or Tutte-3-connected graph.

Theorem 5.39

Every Tutte-2-connected graph of size at least 3 can be obtained by repeatedly 2-summing graphs, starting with 3-blocks. Moreover, in this process, no two cycles are 2-summed together, and two co-cycles are 2-summed together. The decomposition is unique.

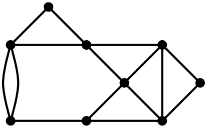
Decomposition of 2-Connected Graphs

Definition 5.38

A 3-block is a cycle of length at least 3, a loopless graph on two vertices with at least 3 edges between them (co-cycles), or Tutte-3-connected graph.

Theorem 5.39

Every Tutte-2-connected graph of size at least 3 can be obtained by repeatedly 2-summing graphs, starting with 3-blocks. Moreover, in this process, no two cycles are 2-summed together, and two co-cycles are 2-summed together. The decomposition is unique.



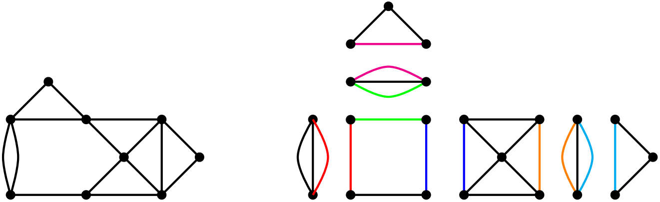
Decomposition of 2-Connected Graphs

Definition 5.38

A 3-block is a cycle of length at least 3, a loopless graph on two vertices with at least 3 edges between them (co-cycles), or Tutte-3-connected graph.

Theorem 5.39

Every Tutte-2-connected graph of size at least 3 can be obtained by repeatedly 2-summing graphs, starting with 3-blocks. Moreover, in this process, no two cycles are 2-summed together, and two co-cycles are 2-summed together. The decomposition is unique.



Homework Set 3

Problem 7

Suppose G is a graph that is non-trivial, connected, and such that every edge e is in some two cycles that meet only at e. What is the highest edge-connectivity of G that can be inferred from these properties?

Problem 8

Find all non-negative integers k for which the following statement is true: For every simple k-regular graph G on at least two vertices, $\kappa(G) = \kappa'(G)$.

Problem 9

Suppose G is a simple r-connected graph of even order with no $K_{1,r+1}$ as an induced subgraph for a positive integer r. Prove that G has a perfect matching.

Polygonal Paths

Note 6.1

A graph may be viewed as a topological space when

Polygonal Paths

Note 6.1

A graph may be viewed as a topological space when

- *the vertices are points;*

Polygonal Paths

Note 6.1

A graph may be viewed as a topological space when

- *the vertices are points;*
- *the edges are homeomorphic images of the unit interval; and*

Polygonal Paths

Note 6.1

A graph may be viewed as a topological space when

- *the vertices are points;*
- *the edges are homeomorphic images of the unit interval; and*
- *the incidences are reflected by the vertices (points) being in the closure of the edges.*

Polygonal Paths

Note 6.1

A graph may be viewed as a topological space when

- *the vertices are points;*
- *the edges are homeomorphic images of the unit interval; and*
- *the incidences are reflected by the vertices (points) being in the closure of the edges.*

Definition 6.2

A polygonal curve in the plane is the union of finitely many line segments such that each segment starts at the end of the previous one and no point lies in more than one segment, except the end of one segment and the beginning of the next one coincide.
A simple open polygonal curve is one homeomorphic to a closed interval.
A simple closed polygonal curve is one homeomorphic to a unit circle.

Plane Graphs

Definition 6.3

- A drawing of a graph G is a function that maps each vertex $v \in V(G)$ to a point $f(v)$ in the plane, and each uv-edge to a simple polygonal $f(u)f(v)$-curve in the plane.

Plane Graphs

Definition 6.3

- A drawing of a graph G is a function that maps each vertex $v \in V(G)$ to a point $f(v)$ in the plane, and each uv-edge to a simple polygonal $f(u)f(v)$-curve in the plane.
- A point $f(e) \cap f(e')$ other than the a common endpoint is a crossing.

Plane Graphs

Definition 6.3

- A drawing of a graph G is a function that maps each vertex $v \in V(G)$ to a point $f(v)$ in the plane, and each uv-edge to a simple polygonal $f(u)f(v)$-curve in the plane.
- A point $f(e) \cap f(e')$ other than the a common endpoint is a crossing.
- A graph is planar if it has a drawing without crossings.

Plane Graphs

Definition 6.3

- A drawing of a graph G is a function that maps each vertex $v \in V(G)$ to a point $f(v)$ in the plane, and each uv-edge to a simple polygonal $f(u)f(v)$-curve in the plane.
- A point $f(e) \cap f(e')$ other than the a common endpoint is a crossing.
- A graph is planar if it has a drawing without crossings. Such a drawing is a planar embedding of G.

Plane Graphs

Definition 6.3

- A drawing of a graph G is a function that maps each vertex $v \in V(G)$ to a point $f(v)$ in the plane, and each uv-edge to a simple polygonal $f(u)f(v)$-curve in the plane.
- A point $f(e) \cap f(e')$ other than the a common endpoint is a crossing.
- A graph is planar if it has a drawing without crossings. Such a drawing is a planar embedding of G.
- A plane graph is a particular drawing of a a graph in the plane with no crossings.

Plane Graphs

Definition 6.3

- A drawing of a graph G is a function that maps each vertex $v \in V(G)$ to a point $f(v)$ in the plane, and each uv-edge to a simple polygonal $f(u)f(v)$-curve in the plane.
- A point $f(e) \cap f(e')$ other than the a common endpoint is a crossing.
- A graph is planar if it has a drawing without crossings. Such a drawing is a planar embedding of G.
- A plane graph is a particular drawing of a a graph in the plane with no crossings.

Note 6.4

A plane embedding corresponds to an embedding of the graph in the sphere through a stereographic projection.

Faces

Theorem 6.5 (Jordan Curve Theorem)

If C is a simple closed polygonal curve in the plane, then the complement of C in the plane consists of two connected components each with C as the boundary.

Faces

Theorem 6.5 (Jordan Curve Theorem)

If C is a simple closed polygonal curve in the plane, then the complement of C in the plane consists of two connected components each with C as the boundary.

Definition 6.6

- The connected components of the complement of a plane graph are the faces of the embedding.

Faces

Theorem 6.5 (Jordan Curve Theorem)

If C is a simple closed polygonal curve in the plane, then the complement of C in the plane consists of two connected components each with C as the boundary.

Definition 6.6

- The connected components of the complement of a plane graph are the faces of the embedding.
- The length of a face is the number of edges in the boundary of the face, with cut-edges counted twice.

Faces

Theorem 6.5 (Jordan Curve Theorem)

If C is a simple closed polygonal curve in the plane, then the complement of C in the plane consists of two connected components each with C as the boundary.

Definition 6.6

- The connected components of the complement of a plane graph are the faces of the embedding.
- The length of a face is the number of edges in the boundary of the face, with cut-edges counted twice.
- The dual graph G^* of a non-empty plane graph G is the graph such that
  - the vertices of G^* are the faces of G;

Faces

Theorem 6.5 (Jordan Curve Theorem)

If C is a simple closed polygonal curve in the plane, then the complement of C in the plane consists of two connected components each with C as the boundary.

Definition 6.6

- The connected components of the complement of a plane graph are the faces of the embedding.
- The length of a face is the number of edges in the boundary of the face, with cut-edges counted twice.
- The dual graph G^* of a non-empty plane graph G is the graph such that
  - the vertices of G^* are the faces of G;
  - the edges of G^* are the edges of G;

Faces

Theorem 6.5 (Jordan Curve Theorem)

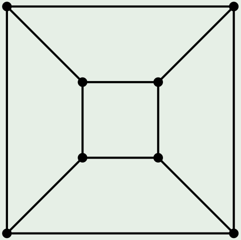
If C is a simple closed polygonal curve in the plane, then the complement of C in the plane consists of two connected components each with C as the boundary.

Definition 6.6

- The connected components of the complement of a plane graph are the faces of the embedding.
- The length of a face is the number of edges in the boundary of the face, with cut-edges counted twice.
- The dual graph G^* of a non-empty plane graph G is the graph such that
  - the vertices of G^* are the faces of G;
  - the edges of G^* are the edges of G;
  - a vertex and an edge of G^* are incident if and only if the edge is the boundary of the corresponding face of G.

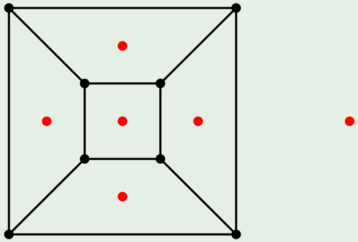
Example of a Dual Graph

Example 6.7



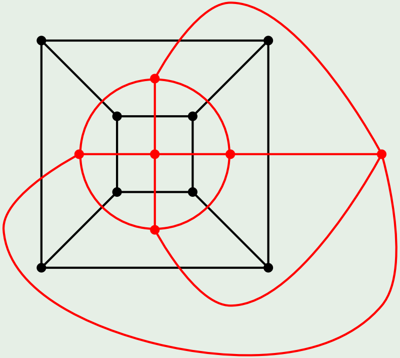
Example of a Dual Graph

Example 6.7



Example of a Dual Graph

Example 6.7



Properties of Dual Graphs

Note 6.8

Different graphs may have the same dual.

Properties of Dual Graphs

Note 6.8

Different graphs may have the same dual. $(G^*)^* = G$ *if and only if* G *is connected.*

Properties of Dual Graphs

Note 6.8

Different graphs may have the same dual. $(G^*)^* = G$ *if and only if* G *is connected.*

Theorem 6.9

Edges in a plane graph form a cycle if and only if the edges in the dual graph form a bond.

Properties of Dual Graphs

Note 6.8

Different graphs may have the same dual. $(G^*)^* = G$ *if and only if* G *is connected.*

Theorem 6.9

Edges in a plane graph form a cycle if and only if the edges in the dual graph form a bond.

Proof.

Suppose D is a set of edges of G that contains a cycle.

$\square$

Properties of Dual Graphs

Note 6.8

Different graphs may have the same dual. $(G^*)^* = G$ if and only if G is connected.

Theorem 6.9

Edges in a plane graph form a cycle if and only if the edges in the dual graph form a bond.

Proof.

Suppose D is a set of edges of G that contains a cycle. By Jordan Curve Theorem 6.5, some face u^* of G lies inside this cycle, and some other v^* lies outside.

□

Properties of Dual Graphs

Note 6.8

Different graphs may have the same dual. $(G^*)^* = G$ if and only if G is connected.

Theorem 6.9

Edges in a plane graph form a cycle if and only if the edges in the dual graph form a bond.

Proof.

Suppose D is a set of edges of G that contains a cycle. By Jordan Curve Theorem 6.5, some face u^* of G lies inside this cycle, and some other v^* lies outside. Then every $u^* v^*$-path in G^* must contain an edge of D^*.

$\square$

Properties of Dual Graphs

Note 6.8

Different graphs may have the same dual. $(G^*)^* = G$ *if and only if* G *is connected.*

Theorem 6.9

Edges in a plane graph form a cycle if and only if the edges in the dual graph form a bond.

Proof.

Suppose D is a set of edges of G that contains a cycle. By Jordan Curve Theorem 6.5, some face u^* of G lies inside this cycle, and some other v^* lies outside. Then every u^*v^*-path in G^* must contain an edge of D^*.
Conversely, suppose D contains no cycle.

□

Properties of Dual Graphs

Note 6.8

Different graphs may have the same dual. $(G^*)^* = G$ *if and only if* G *is connected.*

Theorem 6.9

Edges in a plane graph form a cycle if and only if the edges in the dual graph form a bond.

Proof.

Suppose D is a set of edges of G that contains a cycle. By Jordan Curve Theorem 6.5, some face u^* of G lies inside this cycle, and some other v^* lies outside. Then every u^*v^*-path in G^* must contain an edge of D^*.
Conversely, suppose D contains no cycle. Then it is possible to reach every face of G^* from every other without crossing D^*.

□

Properties of Dual Graphs

Note 6.8

Different graphs may have the same dual. $(G^*)^* = G$ if and only if G is connected.

Theorem 6.9

Edges in a plane graph form a cycle if and only if the edges in the dual graph form a bond.

Proof.

Suppose D is a set of edges of G that contains a cycle. By Jordan Curve Theorem 6.5, some face u^* of G lies inside this cycle, and some other v^* lies outside. Then every u^*v^*-path in G^* must contain an edge of D^*.
Conversely, suppose D contains no cycle. Then it is possible to reach every face of G^* from every other without crossing D^*. Hence $G^* \setminus D^*$ is connected so D^* contains no bond. $\square$

Properties of Dual Graphs, Continued

Theorem 6.10

If a plane graph G is connected, then the Tutte connectivity of G is the same as the Tutte connectivity of G^*.

Proof: Exercise.

Properties of Dual Graphs, Continued

Theorem 6.10

If a plane graph G is connected, then the Tutte connectivity of G is the same as the Tutte connectivity of G^*.

Proof: Exercise.

Theorem 6.11

The following are equivalent for a plane graph G:

Properties of Dual Graphs, Continued

Theorem 6.10

If a plane graph G is connected, then the Tutte connectivity of G is the same as the Tutte connectivity of G^*.

Proof: Exercise.

Theorem 6.11

The following are equivalent for a plane graph G:

(A) *G is bipartite;*

Properties of Dual Graphs, Continued

Theorem 6.10

If a plane graph G is connected, then the Tutte connectivity of G is the same as the Tutte connectivity of G^*.

Proof: Exercise.

Theorem 6.11

The following are equivalent for a plane graph G:

(A) *G is bipartite;*

(B) *every face of G has even length;*

Properties of Dual Graphs, Continued

Theorem 6.10

If a plane graph G is connected, then the Tutte connectivity of G is the same as the Tutte connectivity of G^*.

Proof: Exercise.

Theorem 6.11

The following are equivalent for a plane graph G:

(A) *G is bipartite;*

(B) *every face of G has even length;*

(C) *G^* is Eulerian.*

Properties of Dual Graphs, Continued

Theorem 6.10

If a plane graph G is connected, then the Tutte connectivity of G is the same as the Tutte connectivity of G^*.

Proof: Exercise.

Theorem 6.11

The following are equivalent for a plane graph G:

(A) *G is bipartite;*

(B) *every face of G has even length;*

(C) *G^* is Eulerian.*

Proof.

To see that (A) implies (B), note that the boundary of every face of G is the union of closed walks,

$\square$

Properties of Dual Graphs, Continued

Theorem 6.10

If a plane graph G is connected, then the Tutte connectivity of G is the same as the Tutte connectivity of G^*.

Proof: Exercise.

Theorem 6.11

The following are equivalent for a plane graph G:

(A) *G is bipartite;*

(B) *every face of G has even length;*

(C) *G^* is Eulerian.*

Proof.

To see that (A) implies (B), note that the boundary of every face of G is the union of closed walks, and if the total length is odd, then one of the walks must be of odd length, $\square$

Properties of Dual Graphs, Continued

Theorem 6.10

If a plane graph G is connected, then the Tutte connectivity of G is the same as the Tutte connectivity of G^*.

Proof: Exercise.

Theorem 6.11

The following are equivalent for a plane graph G:

(A) *G is bipartite;*

(B) *every face of G has even length;*

(C) *G^* is Eulerian.*

Proof.

To see that (A) implies (B), note that the boundary of every face of G is the union of closed walks, and if the total length is odd, then one of the walks must be of odd length, and so contain an odd-length cycle. $\square$

Proof, Continued

Conversely, suppose that G has an odd cycle C.

Proof, Continued

Conversely, suppose that G has an odd cycle C. Since G has no crossings, C is laid out as a simple closed polygonal curve.

Proof, Continued

Conversely, suppose that G has an odd cycle C. Since G has no crossings, C is laid out as a simple closed polygonal curve. Let F be the region enclosed by C.

Proof, Continued

Conversely, suppose that G has an odd cycle C. Since G has no crossings, C is laid out as a simple closed polygonal curve. Let F be the region enclosed by C. Every face of G is completely within F, or completely outside of F.

Proof, Continued

Conversely, suppose that G has an odd cycle C. Since G has no crossings, C is laid out as a simple closed polygonal curve. Let F be the region enclosed by C. Every face of G is completely within F, or completely outside of F. Summing up the face lengths for the faces inside F gives an even number since every face is even.

Proof, Continued

Conversely, suppose that G has an odd cycle C. Since G has no crossings, C is laid out as a simple closed polygonal curve. Let F be the region enclosed by C. Every face of G is completely within F, or completely outside of F. Summing up the face lengths for the faces inside F gives an even number since every face is even. This sum counts each edge of C once, and every edge inside F twice.

Proof, Continued

Conversely, suppose that G has an odd cycle C. Since G has no crossings, C is laid out as a simple closed polygonal curve. Let F be the region enclosed by C. Every face of G is completely within F, or completely outside of F. Summing up the face lengths for the faces inside F gives an even number since every face is even. This sum counts each edge of C once, and every edge inside F twice. Hence C is even; a contradiction.

Proof, Continued

Conversely, suppose that G has an odd cycle C. Since G has no crossings, C is laid out as a simple closed polygonal curve. Let F be the region enclosed by C. Every face of G is completely within F, or completely outside of F. Summing up the face lengths for the faces inside F gives an even number since every face is even. This sum counts each edge of C once, and every edge inside F twice. Hence C is even; a contradiction.

The equivalence of (B) and (C) follows from the fact the the dual graph is connected and its vertex degrees are the face lengths of G.

Proof, Continued

Conversely, suppose that G has an odd cycle C. Since G has no crossings, C is laid out as a simple closed polygonal curve. Let F be the region enclosed by C. Every face of G is completely within F, or completely outside of F. Summing up the face lengths for the faces inside F gives an even number since every face is even. This sum counts each edge of C once, and every edge inside F twice. Hence C is even; a contradiction.
The equivalence of (B) and (C) follows from the fact the the dual graph is connected and its vertex degrees are the face lengths of G.

Note 6.12

- *Deleting an edge or a vertex from a plane graph results in a plane graph.*

Proof, Continued

Conversely, suppose that G has an odd cycle C. Since G has no crossings, C is laid out as a simple closed polygonal curve. Let F be the region enclosed by C. Every face of G is completely within F, or completely outside of F. Summing up the face lengths for the faces inside F gives an even number since every face is even. This sum counts each edge of C once, and every edge inside F twice. Hence C is even; a contradiction.

The equivalence of (B) and (C) follows from the fact the the dual graph is connected and its vertex degrees are the face lengths of G.

Note 6.12

- *Deleting an edge or a vertex from a plane graph results in a plane graph.*
- *Contracting an edge in a plane graph can be visualized as sliding the two endvertices towards each other until they meet, pulling all incident edges along.*

Proof, Continued

Conversely, suppose that G has an odd cycle C. Since G has no crossings, C is laid out as a simple closed polygonal curve. Let F be the region enclosed by C. Every face of G is completely within F, or completely outside of F. Summing up the face lengths for the faces inside F gives an even number since every face is even. This sum counts each edge of C once, and every edge inside F twice. Hence C is even; a contradiction.

The equivalence of (B) and (C) follows from the fact the the dual graph is connected and its vertex degrees are the face lengths of G.

Note 6.12

- *Deleting an edge or a vertex from a plane graph results in a plane graph.*
- *Contracting an edge in a plane graph can be visualized as sliding the two endvertices towards each other until they meet, pulling all incident edges along.*
- *Thus the class of planar graphs is minor-closed, that is, all minors of planar graphs are also planar.*

Euler's Formula

Theorem 6.13 (Euler's Formula)

If a connected non-empty plane graph has v vertices, e edges, and f faces, then $v - e + f = 2$.

Euler's Formula

Theorem 6.13 (Euler's Formula)

If a connected non-empty plane graph has v vertices, e edges, and f faces, then $v - e + f = 2$.

Proof.

We proceed by induction on v.

$\square$

Euler's Formula

Theorem 6.13 (Euler's Formula)

If a connected non-empty plane graph has v vertices, e edges, and f faces, then $v - e + f = 2$.

Proof.

We proceed by induction on v. If $v = 1$, then G has only loops, each a closed curve in the embedding.

□

Euler's Formula

Theorem 6.13 (Euler's Formula)

If a connected non-empty plane graph has v vertices, e edges, and f faces, then $v - e + f = 2$.

Proof.

We proceed by induction on v. If $v = 1$, then G has only loops, each a closed curve in the embedding. If $e = 0$, then $f = 1$, and the formula holds.

□

Euler's Formula

Theorem 6.13 (Euler's Formula)

If a connected non-empty plane graph has v vertices, e edges, and f faces, then $v - e + f = 2$.

Proof.

We proceed by induction on v. If $v = 1$, then G has only loops, each a closed curve in the embedding. If $e = 0$, then $f = 1$, and the formula holds. Each added loop adds one more edge and one more face, and so the formula holds when $v = 1$.

□

Euler's Formula

Theorem 6.13 (Euler's Formula)

If a connected non-empty plane graph has v vertices, e edges, and f faces, then $v - e + f = 2$.

Proof.

We proceed by induction on v. If $v = 1$, then G has only loops, each a closed curve in the embedding. If $e = 0$, then $f = 1$, and the formula holds. Each added loop adds one more edge and one more face, and so the formula holds when $v = 1$.

Suppose $v > 1$.

□

Euler's Formula

Theorem 6.13 (Euler's Formula)

If a connected non-empty plane graph has v vertices, e edges, and f faces, then $v - e + f = 2$.

Proof.

We proceed by induction on v. If $v = 1$, then G has only loops, each a closed curve in the embedding. If $e = 0$, then $f = 1$, and the formula holds. Each added loop adds one more edge and one more face, and so the formula holds when $v = 1$.

Suppose $v > 1$. Since G is connected, it has a non-loop edge.

□

Euler's Formula

Theorem 6.13 (Euler's Formula)

If a connected non-empty plane graph has v vertices, e edges, and f faces, then $v - e + f = 2$.

Proof.

We proceed by induction on v. If $v = 1$, then G has only loops, each a closed curve in the embedding. If $e = 0$, then $f = 1$, and the formula holds. Each added loop adds one more edge and one more face, and so the formula holds when $v = 1$.

Suppose $v > 1$. Since G is connected, it has a non-loop edge. Contract such an edge to obtain a plane graph with $v' = v - 1$ vertices, $e' = e - 1$ edges, and $f' = f$ faces.

$\square$

Euler's Formula

Theorem 6.13 (Euler's Formula)

If a connected non-empty plane graph has v vertices, e edges, and f faces, then $v - e + f = 2$.

Proof.

We proceed by induction on v. If $v = 1$, then G has only loops, each a closed curve in the embedding. If $e = 0$, then $f = 1$, and the formula holds. Each added loop adds one more edge and one more face, and so the formula holds when $v = 1$.

Suppose $v > 1$. Since G is connected, it has a non-loop edge. Contract such an edge to obtain a plane graph with $v' = v - 1$ vertices, $e' = e - 1$ edges, and $f' = f$ faces. Applying the inductive hypothesis, we get $v' - e' + f' = 2$, $\square$

Euler's Formula

Theorem 6.13 (Euler's Formula)

If a connected non-empty plane graph has v vertices, e edges, and f faces, then $v - e + f = 2$.

Proof.

We proceed by induction on v. If $v = 1$, then G has only loops, each a closed curve in the embedding. If $e = 0$, then $f = 1$, and the formula holds. Each added loop adds one more edge and one more face, and so the formula holds when $v = 1$.

Suppose $v > 1$. Since G is connected, it has a non-loop edge. Contract such an edge to obtain a plane graph with $v' = v - 1$ vertices, $e' = e - 1$ edges, and $f' = f$ faces. Applying the inductive hypothesis, we get $v' - e' + f' = 2$, and so $(v - 1) - (e - 1) + (f) = v - e + f = 2$, as desired. □

Corollaries of Euler's Formula

Note 6.14

- *Euler's Formula implies that all plane embeddings of connected graphs have the same number of faces.*

Corollaries of Euler's Formula

Note 6.14

- *Euler's Formula implies that all plane embeddings of connected graphs have the same number of faces.*
- *Contracting a non-loop edge of G has the effect of deleting the corresponding edge in G^*. Similarly, deleting a non-cut edge of G has the effect of contracting the corresponding edge in G^*.*

Corollaries of Euler's Formula

Note 6.14

- *Euler's Formula implies that all plane embeddings of connected graphs have the same number of faces.*
- *Contracting a non-loop edge of G has the effect of deleting the corresponding edge in G^*. Similarly, deleting a non-cut edge of G has the effect of contracting the corresponding edge in G^*.*
- *Euler's Formula (as stated) fails for disconnected graphs.*

Corollaries of Euler's Formula

Note 6.14

- *Euler's Formula implies that all plane embeddings of connected graphs have the same number of faces.*
- *Contracting a non-loop edge of G has the effect of deleting the corresponding edge in G^*. Similarly, deleting a non-cut edge of G has the effect of contracting the corresponding edge in G^*.*
- *Euler's Formula (as stated) fails for disconnected graphs.*

Corollary 6.15

If G is a planar graph whose order v is at least 3, whose size is e, and whose girth g is at least 3 but finite, then

$$e \leq \frac{(v-2)g}{g-2}.$$

Corollaries of Euler's Formula

Note 6.14

- *Euler's Formula implies that all plane embeddings of connected graphs have the same number of faces.*
- *Contracting a non-loop edge of G has the effect of deleting the corresponding edge in G^*. Similarly, deleting a non-cut edge of G has the effect of contracting the corresponding edge in G^*.*
- *Euler's Formula (as stated) fails for disconnected graphs.*

Corollary 6.15

If G is a planar graph whose order v is at least 3, whose size is e, and whose girth g is at least 3 but finite, then

$$e \leq \frac{(v-2)g}{g-2}.$$

If G is simple, then $e \leq 3v - 6$.

Proof

Without loss of generality, we may assume that G is plane and connected.

Proof

Without loss of generality, we may assume that G is plane and connected. Let f_i denote the number of faces of G of length i.

Proof

Without loss of generality, we may assume that G is plane and connected. Let f_i denote the number of faces of G of length i. Since every edge appears in two faces or in the same face twice, we have $2e = \sum if_i \geq gf$.

Proof

Without loss of generality, we may assume that G is plane and connected. Let f_i denote the number of faces of G of length i. Since every edge appears in two faces or in the same face twice, we have $2e = \sum if_i \geq gf$. Substituting this into Euler's Formula gives $v - e + 2e/g \geq 2$.

Proof

Without loss of generality, we may assume that G is plane and connected. Let f_i denote the number of faces of G of length i. Since every edge appears in two faces or in the same face twice, we have $2e = \sum if_i \geq gf$. Substituting this into Euler's Formula gives $v - e + 2e/g \geq 2$.

$$e(\frac{2}{g} - 1) \geq 2 - v$$

Proof

Without loss of generality, we may assume that G is plane and connected. Let f_i denote the number of faces of G of length i. Since every edge appears in two faces or in the same face twice, we have $2e = \sum if_i \geq gf$. Substituting this into Euler's Formula gives $v - e + 2e/g \geq 2$.

$$e(\frac{2}{g} - 1) \geq 2 - v$$

$$e \leq \frac{v-2}{\frac{2}{g} - \frac{g}{g}} = \frac{(v-2)g}{g-2}$$

Proof

Without loss of generality, we may assume that G is plane and connected. Let f_i denote the number of faces of G of length i. Since every edge appears in two faces or in the same face twice, we have $2e = \sum if_i \geq gf$. Substituting this into Euler's Formula gives $v - e + 2e/g \geq 2$.

$$e(\frac{2}{g} - 1) \geq 2 - v$$

$$e \leq \frac{v-2}{\frac{2}{g} - \frac{g}{g}} = \frac{(v-2)g}{g-2}$$

Note that when G is simple, $g \geq 3$ and so $e \leq 3v - 6$.

Proof

Without loss of generality, we may assume that G is plane and connected. Let f_i denote the number of faces of G of length i. Since every edge appears in two faces or in the same face twice, we have $2e = \sum if_i \geq gf$. Substituting this into Euler's Formula gives $v - e + 2e/g \geq 2$.

$$e(\frac{2}{g} - 1) \geq 2 - v$$

$$e \leq \frac{v-2}{\frac{2}{g} - \frac{g}{g}} = \frac{(v-2)g}{g-2}$$

Note that when G is simple, $g \geq 3$ and so $e \leq 3v - 6$.

Example 6.16

Is K_5 planar?

Proof

Without loss of generality, we may assume that G is plane and connected. Let f_i denote the number of faces of G of length i. Since every edge appears in two faces or in the same face twice, we have $2e = \sum if_i \geq gf$. Substituting this into Euler's Formula gives $v - e + 2e/g \geq 2$.

$$e(\frac{2}{g} - 1) \geq 2 - v$$

$$e \leq \frac{v-2}{\frac{2}{g} - \frac{g}{g}} = \frac{(v-2)g}{g-2}$$

Note that when G is simple, $g \geq 3$ and so $e \leq 3v - 6$.

Example 6.16

Is K_5 planar?
No, since $e = 10 > 3v - 6 = 9$.

Proof

Without loss of generality, we may assume that G is plane and connected. Let f_i denote the number of faces of G of length i. Since every edge appears in two faces or in the same face twice, we have $2e = \sum if_i \geq gf$. Substituting this into Euler's Formula gives $v - e + 2e/g \geq 2$.

$$e(\frac{2}{g} - 1) \geq 2 - v$$

$$e \leq \frac{v-2}{\frac{2}{g} - \frac{g}{g}} = \frac{(v-2)g}{g-2}$$

Note that when G is simple, $g \geq 3$ and so $e \leq 3v - 6$.

Example 6.16

Is K_5 planar?
No, since $e = 10 > 3v - 6 = 9$.
Is $K_{3,3}$ planar?

Proof

Without loss of generality, we may assume that G is plane and connected. Let f_i denote the number of faces of G of length i. Since every edge appears in two faces or in the same face twice, we have $2e = \sum if_i \geq gf$. Substituting this into Euler's Formula gives $v - e + 2e/g \geq 2$.

$$e(\frac{2}{g} - 1) \geq 2 - v$$

$$e \leq \frac{v-2}{\frac{2}{g} - \frac{g}{g}} = \frac{(v-2)g}{g-2}$$

Note that when G is simple, $g \geq 3$ and so $e \leq 3v - 6$.

Example 6.16

Is K_5 planar?
No, since $e = 10 > 3v - 6 = 9$.
Is $K_{3,3}$ planar?
No, since

$$e = 9 > \frac{(v-2)g}{g-2} = 8.$$

Platonic Solids

We want to find all graphs that are

Platonic Solids

We want to find all graphs that are

- planar,

Platonic Solids

We want to find all graphs that are

- planar,
- simple,

Platonic Solids

We want to find all graphs that are

- planar,
- simple,
- 3-connected,

Platonic Solids

We want to find all graphs that are

- planar,
- simple,
- 3-connected,
- k-regular ($k \geq 3$),

Platonic Solids

We want to find all graphs that are

- planar,
- simple,
- 3-connected,
- k-regular ($k \geq 3$),
- l-co-regular (that is, each face has same length $l \geq 3$).

Platonic Solids

We want to find all graphs that are

- planar,
- simple,
- 3-connected,
- k-regular ($k \geq 3$),
- l-co-regular (that is, each face has same length $l \geq 3$).

We have $kv = 2e = lf$,

Platonic Solids

We want to find all graphs that are

- planar,
- simple,
- 3-connected,
- k-regular ($k \geq 3$),
- l-co-regular (that is, each face has same length $l \geq 3$).

We have $kv = 2e = lf$, and so the Euler's Formula 6.13 gives us $2e/k - e + 2e/l = 2$.

Platonic Solids

We want to find all graphs that are

- planar,
- simple,
- 3-connected,
- k-regular ($k \geq 3$),
- l-co-regular (that is, each face has same length $l \geq 3$).

We have $kv = 2e = lf$, and so the Euler's Formula 6.13 gives us $2e/k - e + 2e/l = 2$. Thus $e(2/k - 1 + 2/l) = 2$ and

$$e = \frac{2kl}{2k + 2l - kl}.$$

Platonic Solids

We want to find all graphs that are

- planar,
- simple,
- 3-connected,
- k-regular ($k \geq 3$),
- l-co-regular (that is, each face has same length $l \geq 3$).

We have $kv = 2e = lf$, and so the Euler's Formula 6.13 gives us $2e/k - e + 2e/l = 2$. Thus $e(2/k - 1 + 2/l) = 2$ and

$$e = \frac{2kl}{2k + 2l - kl}.$$

Then $-kl + 2l + 2k > 0$,

Platonic Solids

We want to find all graphs that are

- planar,
- simple,
- 3-connected,
- k-regular ($k \geq 3$),
- l-co-regular (that is, each face has same length $l \geq 3$).

We have $kv = 2e = lf$, and so the Euler's Formula 6.13 gives us $2e/k - e + 2e/l = 2$. Thus $e(2/k - 1 + 2/l) = 2$ and

$$e = \frac{2kl}{2k + 2l - kl}.$$

Then $-kl + 2l + 2k > 0$, and so $-kl + 2l + 2k - 4 > -4$,

Platonic Solids

We want to find all graphs that are

- planar,
- simple,
- 3-connected,
- k-regular ($k \geq 3$),
- l-co-regular (that is, each face has same length $l \geq 3$).

We have $kv = 2e = lf$, and so the Euler's Formula 6.13 gives us $2e/k - e + 2e/l = 2$. Thus $e(2/k - 1 + 2/l) = 2$ and

$$e = \frac{2kl}{2k + 2l - kl}.$$

Then $-kl + 2l + 2k > 0$, and so $-kl + 2l + 2k - 4 > -4$, And so $(k-2)(l-2) < 4$,

Platonic Solids

We want to find all graphs that are

- planar,
- simple,
- 3-connected,
- k-regular ($k \geq 3$),
- l-co-regular (that is, each face has same length $l \geq 3$).

We have $kv = 2e = lf$, and so the Euler's Formula 6.13 gives us $2e/k - e + 2e/l = 2$. Thus $e(2/k - 1 + 2/l) = 2$ and

$$e = \frac{2kl}{2k + 2l - kl}.$$

Then $-kl + 2l + 2k > 0$, and so $-kl + 2l + 2k - 4 > -4$, And so $(k-2)(l-2) < 4$, and so $k, l \geq 3$ and $k, l \leq 5$.

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3					

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1				

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6			

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4		

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k+2l-kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4					

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2				

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12			

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8		

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3					

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2				

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12			

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6		

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5					

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k+2l-kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3				

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3	30			

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3	30	20		

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3	30	20	12	

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3	30	20	12	dodecahedron

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3	30	20	12	dodecahedron
5	3					

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3	30	20	12	dodecahedron
5	3	3				

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k+2l-kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3	30	20	12	dodecahedron
5	3	3	30			

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k+2l-kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3	30	20	12	dodecahedron
5	3	3	30	12		

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3	30	20	12	dodecahedron
5	3	3	30	12	20	

The Zoo of Platonic Solids

$$e = \frac{2kl}{2k + 2l - kl} \qquad v = \frac{2e}{k} \qquad f = \frac{2e}{l}$$

k	l	$(k-2)(l-2)$	e	v	f	name
3	3	1	6	4	4	tetrahedron
3	4	2	12	8	6	cube
4	3	2	12	6	8	octahedron
3	5	3	30	20	12	dodecahedron
5	3	3	30	12	20	icosahedron

Statement of the Kuratowski Theorem

Theorem 6.17 (Kuratowski 1930)

A graph is planar if and only if it has neither K_5 *nor* $K_{3,3}$ *as a topological minor.*

Statement of the Kuratowski Theorem

Theorem 6.17 (Kuratowski 1930)

A graph is planar if and only if it has neither K_5 nor $K_{3,3}$ as a topological minor.

Theorem 6.18 (Wagner)

A graph is planar if and only if it has neither K_5 nor $K_{3,3}$ as a minor.

Statement of the Kuratowski Theorem

Theorem 6.17 (Kuratowski 1930)

A graph is planar if and only if it has neither K_5 nor $K_{3,3}$ as a topological minor.

Theorem 6.18 (Wagner)

A graph is planar if and only if it has neither K_5 nor $K_{3,3}$ as a minor.

Lemma 6.19

If F is the edge-set of the boundary of a face of a plane graph G, then G has an plane embedding in which F is the boundary of the infinite face.

Statement of the Kuratowski Theorem

Theorem 6.17 (Kuratowski 1930)

A graph is planar if and only if it has neither K_5 nor $K_{3,3}$ as a topological minor.

Theorem 6.18 (Wagner)

A graph is planar if and only if it has neither K_5 nor $K_{3,3}$ as a minor.

Lemma 6.19

If F is the edge-set of the boundary of a face of a plane graph G, then G has an plane embedding in which F is the boundary of the infinite face.

Proof.

Apply stereographic projection twice. □

Minimally No-Planar Graphs

Definition 6.20

A graph is minimally non-planar if it is non-planar, but every proper subgraph of it is planar.

Minimally No-Planar Graphs

Definition 6.20

A graph is minimally non-planar if it is non-planar, but every proper subgraph of it is planar.

Lemma 6.21

Every minimal non-planar graph is 2-connected.

Minimally No-Planar Graphs

Definition 6.20

A graph is minimally non-planar if it is non-planar, but every proper subgraph of it is planar.

Lemma 6.21

Every minimal non-planar graph is 2-connected.

Proof.

If G is disconnected, we can embed one component of G inside one face of the rest of G.

$\square$

Minimally No-Planar Graphs

Definition 6.20

A graph is minimally non-planar if it is non-planar, but every proper subgraph of it is planar.

Lemma 6.21

Every minimal non-planar graph is 2-connected.

Proof.

If G is disconnected, we can embed one component of G inside one face of the rest of G. Similarly, if G has a cut-vertex v, let G_1, G_2, $\ldots$, G_k be the subgraphs of G induced by v together with the components of $G - v$.

$\square$

Minimally No-Planar Graphs

Definition 6.20

A graph is minimally non-planar if it is non-planar, but every proper subgraph of it is planar.

Lemma 6.21

Every minimal non-planar graph is 2-connected.

Proof.

If G is disconnected, we can embed one component of G inside one face of the rest of G. Similarly, if G has a cut-vertex v, let G_1, G_2, $\ldots$, G_k be the subgraphs of G induced by v together with the components of $G - v$. By the minimality of G, these subgraphs are planar.

□

Definition 6.20

A graph is minimally non-planar if it is non-planar, but every proper subgraph of it is planar.

Lemma 6.21

Every minimal non-planar graph is 2-connected.

Proof.

If G is disconnected, we can embed one component of G inside one face of the rest of G. Similarly, if G has a cut-vertex v, let G_1, G_2, $\dots$, G_k be the subgraphs of G induced by v together with the components of $G - v$. By the minimality of G, these subgraphs are planar. It is easy to see that the plane embeddings of these subgraphs can be put together to form a plane embedding of G. $\square$

Lemma 6.22

Suppose $G = H_1 \oplus_2 H_2$ is non-planar.

Lemma 6.22

Suppose $G = H_1 \oplus_2 H_2$ is non-planar. Then at least one of H_1 and H_2 is non-planar.

Lemma 6.22

Suppose $G = H_1 \oplus_2 H_2$ is non-planar. Then at least one of H_1 and H_2 is non-planar.

Proof.

Let e be the common edge of H_1 and H_2.

□

Lemma 6.22

Suppose $G = H_1 \oplus_2 H_2$ is non-planar. Then at least one of H_1 and H_2 is non-planar.

Proof.

Let e be the common edge of H_1 and H_2. Suppose both H_1 and H_2 are planar.

$\square$

Lemma 6.22

Suppose $G = H_1 \oplus_2 H_2$ is non-planar. Then at least one of H_1 and H_2 is non-planar.

Proof.

Let e be the common edge of H_1 and H_2. Suppose both H_1 and H_2 are planar. By Lemma 6.19, each of H_1 and H_2 can be embedded in the plane with e in the boundary of the infinite face.

□

Lemma 6.22

Suppose $G = H_1 \oplus_2 H_2$ is non-planar. Then at least one of H_1 and H_2 is non-planar.

Proof.

Let e be the common edge of H_1 and H_2. Suppose both H_1 and H_2 are planar. By Lemma 6.19, each of H_1 and H_2 can be embedded in the plane with e in the boundary of the infinite face. It is now easy to put together the embeddings of H_1 and H_2 into a plane embedding of G. $\square$

Kuratowski Graphs and Topological Minors

Lemma 6.22

Suppose $G = H_1 \oplus_2 H_2$ *is non-planar. Then at least one of* H_1 *and* H_2 *is non-planar.*

Proof.

Let e be the common edge of H_1 and H_2. Suppose both H_1 and H_2 are planar. By Lemma 6.19, each of H_1 and H_2 can be embedded in the plane with e in the boundary of the infinite face. It is now easy to put together the embeddings of H_1 and H_2 into a plane embedding of G. □

Definition 6.23

- A Kuratowski subgraph is a subgraph isomorphic to a subdivision of K_5 or of $K_{3,3}$.

Lemma 6.22

Suppose $G = H_1 \oplus_2 H_2$ is non-planar. Then at least one of H_1 and H_2 is non-planar.

Proof.

Let e be the common edge of H_1 and H_2. Suppose both H_1 and H_2 are planar. By Lemma 6.19, each of H_1 and H_2 can be embedded in the plane with e in the boundary of the infinite face. It is now easy to put together the embeddings of H_1 and H_2 into a plane embedding of G. □

Definition 6.23

- A Kuratowski subgraph is a subgraph isomorphic to a subdivision of K_5 or of $K_{3,3}$.
- A vertex of a graph G is a branch vertex of a Kuratowski subgraph H of G, if its degree in H exceeds two.

proof

Lemma 6.24

If G/e has a Kuratowski subgraph, then so does G.

proof

Lemma 6.24

If G/e has a Kuratowski subgraph, then so does G.

Proof.

Let H be a Kuratowski subgraph of $G' = G/e$,

□

proof

Lemma 6.24

If G/e has a Kuratowski subgraph, then so does G.

Proof.

Let H be a Kuratowski subgraph of $G' = G/e$, and let z be the vertex of G' obtained by contracting $e = xy$.

$\square$

proof

Lemma 6.24

If G/e has a Kuratowski subgraph, then so does G.

Proof.

Let H be a Kuratowski subgraph of $G' = G/e$, and let z be the vertex of G' obtained by contracting $e = xy$. If z is not a branch vertex of H, then G also has a Kuratowski subgraph obtained from H by lengthening a path through z if necessary.

$\square$

proof

Lemma 6.24

If G/e has a Kuratowski subgraph, then so does G.

Proof.

Let H be a Kuratowski subgraph of $G' = G/e$, and let z be the vertex of G' obtained by contracting $e = xy$. If z is not a branch vertex of H, then G also has a Kuratowski subgraph obtained from H by lengthening a path through z if necessary. If z is a branch vertex of H and at most one of the edges incident to z in H is incident to x in G,

□

proof

Lemma 6.24

If G/e has a Kuratowski subgraph, then so does G.

Proof.

Let H be a Kuratowski subgraph of $G' = G/e$, and let z be the vertex of G' obtained by contracting $e = xy$. If z is not a branch vertex of H, then G also has a Kuratowski subgraph obtained from H by lengthening a path through z if necessary. If z is a branch vertex of H and at most one of the edges incident to z in H is incident to x in G, then z can be expanded into xy to lengthen that path, and y becomes the corresponding branch vertex of a Kuratowski subgraph of G.

□

proof

Lemma 6.24

If G/e has a Kuratowski subgraph, then so does G.

Proof.

Let H be a Kuratowski subgraph of $G' = G/e$, and let z be the vertex of G' obtained by contracting $e = xy$. If z is not a branch vertex of H, then G also has a Kuratowski subgraph obtained from H by lengthening a path through z if necessary. If z is a branch vertex of H and at most one of the edges incident to z in H is incident to x in G, then z can be expanded into xy to lengthen that path, and y becomes the corresponding branch vertex of a Kuratowski subgraph of G.

The only remaining case to consider is when H is a subdivision of K_5, z is a branch vertex of H, and each of x and y is incident in G to two of the four edges incident to z in H.

□

Lemma 6.24

If G/e has a Kuratowski subgraph, then so does G.

Proof.

Let H be a Kuratowski subgraph of $G' = G/e$, and let z be the vertex of G' obtained by contracting $e = xy$. If z is not a branch vertex of H, then G also has a Kuratowski subgraph obtained from H by lengthening a path through z if necessary. If z is a branch vertex of H and at most one of the edges incident to z in H is incident to x in G, then z can be expanded into xy to lengthen that path, and y becomes the corresponding branch vertex of a Kuratowski subgraph of G.

The only remaining case to consider is when H is a subdivision of K_5, z is a branch vertex of H, and each of x and y is incident in G to two of the four edges incident to z in H. Let u_1, u_2 be the branch vertices of H that are at the other ends of paths leaving z on the edges incident with x,

□

proof

Lemma 6.24

If G/e has a Kuratowski subgraph, then so does G.

Proof.

Let H be a Kuratowski subgraph of $G' = G/e$, and let z be the vertex of G' obtained by contracting $e = xy$. If z is not a branch vertex of H, then G also has a Kuratowski subgraph obtained from H by lengthening a path through z if necessary. If z is a branch vertex of H and at most one of the edges incident to z in H is incident to x in G, then z can be expanded into xy to lengthen that path, and y becomes the corresponding branch vertex of a Kuratowski subgraph of G.

The only remaining case to consider is when H is a subdivision of K_5, z is a branch vertex of H, and each of x and y is incident in G to two of the four edges incident to z in H. Let u_1, u_2 be the branch vertices of H that are at the other ends of paths leaving z on the edges incident with x, and let v_1, v_2 be the other branch vertices of H.

□

Lemma 6.24

If G/e has a Kuratowski subgraph, then so does G.

Proof.

Let H be a Kuratowski subgraph of $G' = G/e$, and let z be the vertex of G' obtained by contracting $e = xy$. If z is not a branch vertex of H, then G also has a Kuratowski subgraph obtained from H by lengthening a path through z if necessary. If z is a branch vertex of H and at most one of the edges incident to z in H is incident to x in G, then z can be expanded into xy to lengthen that path, and y becomes the corresponding branch vertex of a Kuratowski subgraph of G.

The only remaining case to consider is when H is a subdivision of K_5, z is a branch vertex of H, and each of x and y is incident in G to two of the four edges incident to z in H. Let u_1, u_2 be the branch vertices of H that are at the other ends of paths leaving z on the edges incident with x, and let v_1, v_2 be the other branch vertices of H. By deleting the edges of the u_1u_2-path and the v_1v_2-path, we obtain a subdivision of $K_{3,3}$. □

Tutte's Version of Kuratowski's Theorem

Definition 6.25

A plane embedding is convex if every face except the infinite one is a convex polygon.

Tutte's Version of Kuratowski's Theorem

Definition 6.25

A plane embedding is convex if every face except the infinite one is a convex polygon.

Theorem 6.26 (Tutte 1960–63)

If G is a simple 3-connected graph with neither K_5 nor $K_{3,3}$ as the topological minor, then G has a convex embedding in the plane with no three vertices on a line.

Tutte's Version of Kuratowski's Theorem

Definition 6.25

A plane embedding is convex if every face except the infinite one is a convex polygon.

Theorem 6.26 (Tutte 1960–63)

If G is a simple 3-connected graph with neither K_5 nor $K_{3,3}$ as the topological minor, then G has a convex embedding in the plane with no three vertices on a line.

Proof.

We proceed by induction on $|G|$.

$\square$

Tutte's Version of Kuratowski's Theorem

Definition 6.25

A plane embedding is convex if every face except the infinite one is a convex polygon.

Theorem 6.26 (Tutte 1960–63)

If G is a simple 3-connected graph with neither K_5 nor $K_{3,3}$ as the topological minor, then G has a convex embedding in the plane with no three vertices on a line.

Proof.

We proceed by induction on $|G|$. The only 3-connected simple graph on at most 4 vertices is K_4 and it has such an embedding.

□

Tutte's Version of Kuratowski's Theorem

Definition 6.25

A plane embedding is convex if every face except the infinite one is a convex polygon.

Theorem 6.26 (Tutte 1960–63)

If G is a simple 3-connected graph with neither K_5 nor $K_{3,3}$ as the topological minor, then G has a convex embedding in the plane with no three vertices on a line.

Proof.

We proceed by induction on $|G|$. The only 3-connected simple graph on at most 4 vertices is K_4 and it has such an embedding. Let G be a graph on $n \geq 5$ vertices and suppose the theorem holds for all graphs on fewer than n vertices. $\square$

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected;

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e.

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph,

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph, and so by the induction hypothesis, $H = G/e$ has a convex embedding with no 3 vertices on a line.

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph, and so by the induction hypothesis, $H = G/e$ has a convex embedding with no 3 vertices on a line. Let C denote the cycle that is the boundary of the face $H - z$ that is not a face of H.

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph, and so by the induction hypothesis, $H = G/e$ has a convex embedding with no 3 vertices on a line. Let C denote the cycle that is the boundary of the face $H - z$ that is not a face of H.

Since we started with a convex embedding, the face bounded by C contains straight line segments from z to all of its neighbors.

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph, and so by the induction hypothesis, $H = G/e$ has a convex embedding with no 3 vertices on a line. Let C denote the cycle that is the boundary of the face $H - z$ that is not a face of H.
Since we started with a convex embedding, the face bounded by C contains straight line segments from z to all of its neighbors. Let x_1, x_2, ..., x_k be the neighbors of x in the cyclic order on C.

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph, and so by the induction hypothesis, $H = G/e$ has a convex embedding with no 3 vertices on a line. Let C denote the cycle that is the boundary of the face $H - z$ that is not a face of H.
Since we started with a convex embedding, the face bounded by C contains straight line segments from z to all of its neighbors. Let x_1, x_2, . . . , x_k be the neighbors of x in the cyclic order on C. If all neighbors of y lie on the portion of C from x_i to x_{i+1}, then we obtain a convex embedding of G.

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph, and so by the induction hypothesis, $H = G/e$ has a convex embedding with no 3 vertices on a line. Let C denote the cycle that is the boundary of the face $H - z$ that is not a face of H.
Since we started with a convex embedding, the face bounded by C contains straight line segments from z to all of its neighbors. Let x_1, x_2, . . . , x_k be the neighbors of x in the cyclic order on C. If all neighbors of y lie on the portion of C from x_i to x_{i+1}, then we obtain a convex embedding of G. If this does not occur, then either:

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph, and so by the induction hypothesis, $H = G/e$ has a convex embedding with no 3 vertices on a line. Let C denote the cycle that is the boundary of the face $H - z$ that is not a face of H.
Since we started with a convex embedding, the face bounded by C contains straight line segments from z to all of its neighbors. Let x_1, x_2, . . . , x_k be the neighbors of x in the cyclic order on C. If all neighbors of y lie on the portion of C from x_i to x_{i+1}, then we obtain a convex embedding of G. If this does not occur, then either:

1. y shares three neighbors with x,

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph, and so by the induction hypothesis, $H = G/e$ has a convex embedding with no 3 vertices on a line. Let C denote the cycle that is the boundary of the face $H - z$ that is not a face of H.
Since we started with a convex embedding, the face bounded by C contains straight line segments from z to all of its neighbors. Let x_1, x_2, ..., x_k be the neighbors of x in the cyclic order on C. If all neighbors of y lie on the portion of C from x_i to x_{i+1}, then we obtain a convex embedding of G. If this does not occur, then either:

1. y shares three neighbors with x, in which case we obtain a subdivision of K_5; or

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph, and so by the induction hypothesis, $H = G/e$ has a convex embedding with no 3 vertices on a line. Let C denote the cycle that is the boundary of the face $H - z$ that is not a face of H.
Since we started with a convex embedding, the face bounded by C contains straight line segments from z to all of its neighbors. Let x_1, x_2, . . . , x_k be the neighbors of x in the cyclic order on C. If all neighbors of y lie on the portion of C from x_i to x_{i+1}, then we obtain a convex embedding of G. If this does not occur, then either:

1. y shares three neighbors with x, in which case we obtain a subdivision of K_5; or
2. y has neighbors u, v in C that are in different components of $C - \{x_i x_{i+1}\}$ for some i,

Proof, Continued

By Lemma 5.35, G has an edge $e = xy$ such that G/e is also 3-connected; let z denote the vertex resulting from contracting e. By Lemma 6.24, G/e has no Kuratowski subgraph, and so by the induction hypothesis, $H = G/e$ has a convex embedding with no 3 vertices on a line. Let C denote the cycle that is the boundary of the face $H - z$ that is not a face of H.
Since we started with a convex embedding, the face bounded by C contains straight line segments from z to all of its neighbors. Let x_1, x_2, . . . , x_k be the neighbors of x in the cyclic order on C. If all neighbors of y lie on the portion of C from x_i to x_{i+1}, then we obtain a convex embedding of G. If this does not occur, then either:

1. y shares three neighbors with x, in which case we obtain a subdivision of K_5; or
2. y has neighbors u, v in C that are in different components of $C - \{x_i x_{i+1}\}$ for some i, in which case we obtain a subdivision of $K_{3,3}$.

Proof of Kuratowski Theorem

Recall: G is planar if and only if neither K_5 nor $K_{3,3}$ is a topological minor of G.

Proof.

Without loss of generality, we may asume that G is simple.

$\square$

Proof of Kuratowski Theorem

Recall: G is planar if and only if neither K_5 nor $K_{3,3}$ is a topological minor of G.

Proof.

Without loss of generality, we may asume that G is simple. We showed in Example 6.16 that K_5 and $K_{3,3}$ are both non-planar.

□

Proof of Kuratowski Theorem

Recall: G is planar if and only if neither K_5 nor $K_{3,3}$ is a topological minor of G.

Proof.

Without loss of generality, we may asume that G is simple. We showed in Example 6.16 that K_5 and $K_{3,3}$ are both non-planar. Therefore any subdivision of K_3 or of $K_{3,3}$ is also non-planar,

□

Proof of Kuratowski Theorem

Recall: G is planar if and only if neither K_5 nor $K_{3,3}$ is a topological minor of G.

Proof.

Without loss of generality, we may asume that G is simple. We showed in Example 6.16 that K_5 and $K_{3,3}$ are both non-planar. Therefore any subdivision of K_3 or of $K_{3,3}$ is also non-planar, as is any supergraph of such a subdivision.

□

Proof of Kuratowski Theorem

Recall: G is planar if and only if neither K_5 nor $K_{3,3}$ is a topological minor of G.

Proof.

Without loss of generality, we may asume that G is simple. We showed in Example 6.16 that K_5 and $K_{3,3}$ are both non-planar. Therefore any subdivision of K_3 or of $K_{3,3}$ is also non-planar, as is any supergraph of such a subdivision. Suppose the converse implication fails, and G is a counter-example of the possible smallest order,

□

Proof of Kuratowski Theorem

Recall: G is planar if and only if neither K_5 nor $K_{3,3}$ is a topological minor of G.

Proof.

Without loss of generality, we may asume that G is simple. We showed in Example 6.16 that K_5 and $K_{3,3}$ are both non-planar. Therefore any subdivision of K_3 or of $K_{3,3}$ is also non-planar, as is any supergraph of such a subdivision. Suppose the converse implication fails, and G is a counter-example of the possible smallest order, that is, G is non-planar but has no Kuratowski subgraph.

$\square$

Proof of Kuratowski Theorem

Recall: G is planar if and only if neither K_5 nor $K_{3,3}$ is a topological minor of G.

Proof.

Without loss of generality, we may asume that G is simple. We showed in Example 6.16 that K_5 and $K_{3,3}$ are both non-planar. Therefore any subdivision of K_3 or of $K_{3,3}$ is also non-planar, as is any supergraph of such a subdivision. Suppose the converse implication fails, and G is a counter-example of the possible smallest order, that is, G is non-planar but has no Kuratowski subgraph. Then G is minimally non-planar,

□

Proof of Kuratowski Theorem

Recall: G is planar if and only if neither K_5 nor $K_{3,3}$ is a topological minor of G.

Proof.

Without loss of generality, we may asume that G is simple. We showed in Example 6.16 that K_5 and $K_{3,3}$ are both non-planar. Therefore any subdivision of K_3 or of $K_{3,3}$ is also non-planar, as is any supergraph of such a subdivision. Suppose the converse implication fails, and G is a counter-example of the possible smallest order, that is, G is non-planar but has no Kuratowski subgraph. Then G is minimally non-planar, and, by Lemma 6.21, G is 2-connected.

$\square$

Proof of Kuratowski Theorem

Recall: G is planar if and only if neither K_5 nor $K_{3,3}$ is a topological minor of G.

Proof.

Without loss of generality, we may asume that G is simple. We showed in Example 6.16 that K_5 and $K_{3,3}$ are both non-planar. Therefore any subdivision of K_3 or of $K_{3,3}$ is also non-planar, as is any supergraph of such a subdivision. Suppose the converse implication fails, and G is a counter-example of the possible smallest order, that is, G is non-planar but has no Kuratowski subgraph. Then G is minimally non-planar, and, by Lemma 6.21, G is 2-connected. Now, Lemma 6.22 implies that G is 3-connected.

□

Proof of Kuratowski Theorem

Recall: G is planar if and only if neither K_5 nor $K_{3,3}$ is a topological minor of G.

Proof.

Without loss of generality, we may asume that G is simple. We showed in Example 6.16 that K_5 and $K_{3,3}$ are both non-planar. Therefore any subdivision of K_3 or of $K_{3,3}$ is also non-planar, as is any supergraph of such a subdivision. Suppose the converse implication fails, and G is a counter-example of the possible smallest order, that is, G is non-planar but has no Kuratowski subgraph. Then G is minimally non-planar, and, by Lemma 6.21, G is 2-connected. Now, Lemma 6.22 implies that G is 3-connected. But then we get a contradiction with the Tutte Theorem 6.26, which states that a 3-connected graph with no Kuratowski subgraph is planar. $\square$

Problem Set 4

Problem 10

Prove that every n-vertex plane graph isomorphic to its dual has $2n - 2$ edges. For each $n \geq 4$, construct a simple n-vertex plane graph isomorphic to its dual.

Problem Set 4

Problem 10

Prove that every n-vertex plane graph isomorphic to its dual has $2n-2$ edges. For each $n \geq 4$, construct a simple n-vertex plane graph isomorphic to its dual.

Problem 11

Let G be a connected plane graph such that every vertex of G is incident with two faces of length four, one face of length six, and no other faces. Use Euler's Formula to determine the number of vertices, edges, and faces of G. Draw G.

Problem Set 4

Problem 10

Prove that every n-vertex plane graph isomorphic to its dual has $2n-2$ edges. For each $n \geq 4$, construct a simple n-vertex plane graph isomorphic to its dual.

Problem 11

Let G be a connected plane graph such that every vertex of G is incident with two faces of length four, one face of length six, and no other faces. Use Euler's Formula to determine the number of vertices, edges, and faces of G. Draw G.

Problem 12

A plane graph is outerplane if it has a face incident with all the vertices. A graph is outerplanar if it isomorphic to an outerplane graph. Prove that a graph is outerplanar if and only if it has neither K_4 nor $K_{2,3}$ as a topological minor.

Chromatic Number

Definition 7.1

- A k-coloring of a graph G is a labeling $f : V(G) \to \{1, 2, \ldots, k\}$.

Chromatic Number

Definition 7.1

- A k-coloring of a graph G is a labeling $f : V(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors.

Chromatic Number

Definition 7.1

- A k-coloring of a graph G is a labeling $f : V(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors.
- The vertices with color i are a color class.

Chromatic Number

Definition 7.1

- A k-coloring of a graph G is a labeling $f : V(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors.
- The vertices with color i are a color class.
- A k-coloring f is proper if $f(x) \neq f(y)$ whenever x and y are adjacent.

Chromatic Number

Definition 7.1

- A k-coloring of a graph G is a labeling $f : V(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors.
- The vertices with color i are a color class.
- A k-coloring f is proper if $f(x) \neq f(y)$ whenever x and y are adjacent.
- The chromatic number $\chi(G)$ is the minimum k such that G is k-colorable.

Chromatic Number

Definition 7.1

- A k-coloring of a graph G is a labeling $f : V(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors.
- The vertices with color i are a color class.
- A k-coloring f is proper if $f(x) \neq f(y)$ whenever x and y are adjacent.
- The chromatic number $\chi(G)$ is the minimum k such that G is k-colorable.
- If $\chi(G) = k$, then G is k-chromatic.

Chromatic Number

Definition 7.1

- A k-coloring of a graph G is a labeling $f : V(G) \to \{1, 2, \dots, k\}$.
- The labels are colors.
- The vertices with color i are a color class.
- A k-coloring f is proper if $f(x) \neq f(y)$ whenever x and y are adjacent.
- The chromatic number $\chi(G)$ is the minimum k such that G is k-colorable.
- If $\chi(G) = k$, then G is k-chromatic.
- If $\chi(G) = k$, but $\chi(H) < k$ for every proper subgraph H of G, then G is k-color-critical or k-critical.

Chromatic Number

Definition 7.1

- A k-coloring of a graph G is a labeling $f : V(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors.
- The vertices with color i are a color class.
- A k-coloring f is proper if $f(x) \neq f(y)$ whenever x and y are adjacent.
- The chromatic number $\chi(G)$ is the minimum k such that G is k-colorable.
- If $\chi(G) = k$, then G is k-chromatic.
- If $\chi(G) = k$, but $\chi(H) < k$ for every proper subgraph H of G, then G is k-color-critical or k-critical.
- Let $\omega(G)$ denote the clique number of G, that is, the order of a largest complete subgraph of G.

Chromatic Number

Definition 7.1

- A k-coloring of a graph G is a labeling $f : V(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors.
- The vertices with color i are a color class.
- A k-coloring f is proper if $f(x) \neq f(y)$ whenever x and y are adjacent.
- The chromatic number $\chi(G)$ is the minimum k such that G is k-colorable.
- If $\chi(G) = k$, then G is k-chromatic.
- If $\chi(G) = k$, but $\chi(H) < k$ for every proper subgraph H of G, then G is k-color-critical or k-critical.
- Let $\omega(G)$ denote the clique number of G, that is, the order of a largest complete subgraph of G.
- Let $\alpha(G)$ denote the independence number of G, that is, the largest number of vertices of G no two of which are adjacent.

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$,

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.
- There are triangle-free graphs of arbitrarily high chromatic number (Mycielski)

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.
- There are triangle-free graphs of arbitrarily high chromatic number (Mycielski)
  - Let $M_2 = K_2$.

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.
- There are triangle-free graphs of arbitrarily high chromatic number (Mycielski)
  - Let $M_2 = K_2$.
  - Construct M_{n+1} from M_n by:

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.
- There are triangle-free graphs of arbitrarily high chromatic number (Mycielski)
  - Let $M_2 = K_2$.
  - Construct M_{n+1} from M_n by:
    - Start with a copy M of M_n;

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.
- There are triangle-free graphs of arbitrarily high chromatic number (Mycielski)
  - Let $M_2 = K_2$.
  - Construct M_{n+1} from M_n by:
    - Start with a copy M of M_n;
    - for each vertex v in M add a new vertex u and connect it to the neighbors of v in M;

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.
- There are triangle-free graphs of arbitrarily high chromatic number (Mycielski)
  - Let $M_2 = K_2$.
  - Construct M_{n+1} from M_n by:
    - Start with a copy M of M_n;
    - for each vertex v in M add a new vertex u and connect it to the neighbors of v in M;
    - add a new vertex w and connect it to all vertices not in M.

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.
- There are triangle-free graphs of arbitrarily high chromatic number (Mycielski)
  - Let $M_2 = K_2$.
  - Construct M_{n+1} from M_n by:
    - Start with a copy M of M_n;
    - for each vertex v in M add a new vertex u and connect it to the neighbors of v in M;
    - add a new vertex w and connect it to all vertices not in M.
- $\chi(G) \geq |G|/\alpha(G)$

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.
- There are triangle-free graphs of arbitrarily high chromatic number (Mycielski)
  - Let $M_2 = K_2$.
  - Construct M_{n+1} from M_n by:
    - Start with a copy M of M_n;
    - for each vertex v in M add a new vertex u and connect it to the neighbors of v in M;
    - add a new vertex w and connect it to all vertices not in M.
- $\chi(G) \geq |G|/\alpha(G)$
- $\chi(G) \leq \Delta(G) + 1$

Proof.

Color greedily:

□

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.
- There are triangle-free graphs of arbitrarily high chromatic number (Mycielski)
  - Let $M_2 = K_2$.
  - Construct M_{n+1} from M_n by:
    - Start with a copy M of M_n;
    - for each vertex v in M add a new vertex u and connect it to the neighbors of v in M;
    - add a new vertex w and connect it to all vertices not in M.
- $\chi(G) \geq |G|/\alpha(G)$
- $\chi(G) \leq \Delta(G) + 1$

Proof.

Color greedily: Order the vertices arbitrarily as $v_1, v_2, \ldots, v_n$.

$\square$

Facts About Chromatic Number

- For vertex coloring, all graphs will be considered simple.
- $\chi(G) \geq \omega(G)$, and $\chi(G) = \omega(G)$ when G is complete.
- $\chi(G)$ may exceed $\omega(G)$, for example, consider $C_{2r+1} \vee K_s$, that is, the graph formed from the disjoint union of C_{2r+1} and K_s by joining each vertex of C_{2r+1} to each vertex of K_s.
- There are triangle-free graphs of arbitrarily high chromatic number (Mycielski)
  - Let $M_2 = K_2$.
  - Construct M_{n+1} from M_n by:
    - Start with a copy M of M_n;
    - for each vertex v in M add a new vertex u and connect it to the neighbors of v in M;
    - add a new vertex w and connect it to all vertices not in M.
- $\chi(G) \geq |G|/\alpha(G)$
- $\chi(G) \leq \Delta(G) + 1$

Proof.

Color greedily: Order the vertices arbitrarily as $v_1, v_2, \ldots, v_n$. Starting with $k = 1$, color each vertex v_k with the smallest color not used among the vertices $v_1, v_2, \ldots, v_{k-1}$ that are neighbors of v_k. □

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$.

$\square$

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$.
If G is not not regular, then choose v_n so that $d(v_n) < k$.

$\square$

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \le \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$. If G is not not regular, then choose v_n so that $d(v_n) < k$. Since G is connected, we can grow a spanning tree of G from v_n, assigning indices in the decreasing order as we reach vertices.

□

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$. If G is not not regular, then choose v_n so that $d(v_n) < k$. Since G is connected, we can grow a spanning tree of G from v_n, assigning indices in the decreasing order as we reach vertices. Then each vertex v_i, other than v_n, has at least one neighbor among $v_{i+1}, v_{i+2}, \ldots, v_n$,

□

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$. If G is not not regular, then choose v_n so that $d(v_n) < k$. Since G is connected, we can grow a spanning tree of G from v_n, assigning indices in the decreasing order as we reach vertices. Then each vertex v_i, other than v_n, has at least one neighbor among v_{i+1}, v_{i+2}, ..., v_n, and so each vertex v_i (including v_n) has at most $k-1$ neighbors among v_1, v_2, ..., v_{i-1}.

$\square$

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$. If G is not not regular, then choose v_n so that $d(v_n) < k$. Since G is connected, we can grow a spanning tree of G from v_n, assigning indices in the decreasing order as we reach vertices. Then each vertex v_i, other than v_n, has at least one neighbor among $v_{i+1}, v_{i+2}, \ldots, v_n$, and so each vertex v_i (including v_n) has at most $k-1$ neighbors among $v_1, v_2, \ldots, v_{i-1}$. Hence the greedy coloring requires at most k colors.

$\square$

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$. If G is not not regular, then choose v_n so that $d(v_n) < k$. Since G is connected, we can grow a spanning tree of G from v_n, assigning indices in the decreasing order as we reach vertices. Then each vertex v_i, other than v_n, has at least one neighbor among v_{i+1}, v_{i+2}, ..., v_n, and so each vertex v_i (including v_n) has at most $k - 1$ neighbors among v_1, v_2, ..., v_{i-1}. Hence the greedy coloring requires at most k colors. In the remaining case, G is k-regular.

□

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$. If G is not not regular, then choose v_n so that $d(v_n) < k$. Since G is connected, we can grow a spanning tree of G from v_n, assigning indices in the decreasing order as we reach vertices. Then each vertex v_i, other than v_n, has at least one neighbor among v_{i+1}, v_{i+2}, ..., v_n, and so each vertex v_i (including v_n) has at most $k-1$ neighbors among v_1, v_2, ..., v_{i-1}. Hence the greedy coloring requires at most k colors. In the remaining case, G is k-regular. If G has a cut-vertex x, then let G' be the subgraph of G induced by a component of $G - x$ together with x.

□

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$. If G is not not regular, then choose v_n so that $d(v_n) < k$. Since G is connected, we can grow a spanning tree of G from v_n, assigning indices in the decreasing order as we reach vertices. Then each vertex v_i, other than v_n, has at least one neighbor among $v_{i+1}, v_{i+2}, \ldots, v_n$, and so each vertex v_i (including v_n) has at most $k - 1$ neighbors among $v_1, v_2, \ldots, v_{i-1}$. Hence the greedy coloring requires at most k colors. In the remaining case, G is k-regular. If G has a cut-vertex x, then let G' be the subgraph of G induced by a component of $G - x$ together with x. Then the degree of x in G' is less than k,

□

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$. If G is not not regular, then choose v_n so that $d(v_n) < k$. Since G is connected, we can grow a spanning tree of G from v_n, assigning indices in the decreasing order as we reach vertices. Then each vertex v_i, other than v_n, has at least one neighbor among $v_{i+1}, v_{i+2}, \ldots, v_n$, and so each vertex v_i (including v_n) has at most $k - 1$ neighbors among $v_1, v_2, \ldots, v_{i-1}$. Hence the greedy coloring requires at most k colors. In the remaining case, G is k-regular. If G has a cut-vertex x, then let G' be the subgraph of G induced by a component of $G - x$ together with x. Then the degree of x in G' is less than k, and so G' has a proper k-coloring.

□

Brooks' Theorem

Theorem 7.2 (Brooks 1941)

If G is a connected simple graph other than a clique and an odd cycle, then $\chi(G) \leq \Delta(G)$.

Proof.

Suppose G is connected, not a clique, not an odd cycle, and let $k = \Delta(G)$. If G is not not regular, then choose v_n so that $d(v_n) < k$. Since G is connected, we can grow a spanning tree of G from v_n, assigning indices in the decreasing order as we reach vertices. Then each vertex v_i, other than v_n, has at least one neighbor among $v_{i+1}, v_{i+2}, \ldots, v_n$, and so each vertex v_i (including v_n) has at most $k-1$ neighbors among $v_1, v_2, \ldots, v_{i-1}$. Hence the greedy coloring requires at most k colors. In the remaining case, G is k-regular. If G has a cut-vertex x, then let G' be the subgraph of G induced by a component of $G - x$ together with x. Then the degree of x in G' is less than k, and so G' has a proper k-coloring. After permuting the colors, if necessary, it is easy to combine the k-colorings of such subgraphs into a k-coloring of G. □

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected.

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected.

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, . . . , v_n.

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, ..., v_n. Then every vertex v_i other than v_n would have at most $k - 1$ neighbors among v_1, v_2, ..., v_{i-1},

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, $\ldots$, v_n. Then every vertex v_i other than v_n would have at most $k - 1$ neighbors among v_1, v_2, $\ldots$, v_{i-1}, while two of the neighbors of v_n, namely v_1 and v_2 would both receive color 1 by the greedy coloring.

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, $\ldots$, v_n. Then every vertex v_i other than v_n would have at most $k - 1$ neighbors among v_1, v_2, $\ldots$, v_{i-1}, while two of the neighbors of v_n, namely v_1 and v_2 would both receive color 1 by the greedy coloring. Hence, it suffices to show that every 2-connected k-regular graph G with $k \geq 3$, not a clique, not an odd cycle, has 3-vertex induced path v_1, v_n, v_2 such that $G - \{v_1, v_2\}$ is connected.

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, ..., v_n. Then every vertex v_i other than v_n would have at most $k-1$ neighbors among v_1, v_2, ..., v_{i-1}, while two of the neighbors of v_n, namely v_1 and v_2 would both receive color 1 by the greedy coloring. Hence, it suffices to show that every 2-connected k-regular graph G with $k \geq 3$, not a clique, not an odd cycle, has 3-vertex induced path v_1, v_n, v_2 such that $G - \{v_1, v_2\}$ is connected. Choose a vertex x. If $\kappa(G - x) \geq 2$, then let $v_1 = x$, and let v_2 be a vertex of distance two from x,

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, $\ldots$, v_n. Then every vertex v_i other than v_n would have at most $k - 1$ neighbors among v_1, v_2, $\ldots$, v_{i-1}, while two of the neighbors of v_n, namely v_1 and v_2 would both receive color 1 by the greedy coloring. Hence, it suffices to show that every 2-connected k-regular graph G with $k \geq 3$, not a clique, not an odd cycle, has 3-vertex induced path v_1, v_n, v_2 such that $G - \{v_1, v_2\}$ is connected. Choose a vertex x. If $\kappa(G - x) \geq 2$, then let $v_1 = x$, and let v_2 be a vertex of distance two from x, which exists, since G is connected, regular and not a clique.

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, . . . , v_n. Then every vertex v_i other than v_n would have at most $k - 1$ neighbors among v_1, v_2, . . . , v_{i-1}, while two of the neighbors of v_n, namely v_1 and v_2 would both receive color 1 by the greedy coloring. Hence, it suffices to show that every 2-connected k-regular graph G with $k \geq 3$, not a clique, not an odd cycle, has 3-vertex induced path v_1, v_n, v_2 such that $G - \{v_1, v_2\}$ is connected. Choose a vertex x. If $\kappa(G - x) \geq 2$, then let $v_1 = x$, and let v_2 be a vertex of distance two from x, which exists, since G is connected, regular and not a clique. If $\kappa(G - x) = 1$, then x has a neighbor in every block of $G - x$ that is a leaf in the block tree of $G - x$,

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, ..., v_n. Then every vertex v_i other than v_n would have at most $k - 1$ neighbors among v_1, v_2, ..., v_{i-1}, while two of the neighbors of v_n, namely v_1 and v_2 would both receive color 1 by the greedy coloring. Hence, it suffices to show that every 2-connected k-regular graph G with $k \geq 3$, not a clique, not an odd cycle, has 3-vertex induced path v_1, v_n, v_2 such that $G - \{v_1, v_2\}$ is connected. Choose a vertex x. If $\kappa(G - x) \geq 2$, then let $v_1 = x$, and let v_2 be a vertex of distance two from x, which exists, since G is connected, regular and not a clique. If $\kappa(G - x) = 1$, then x has a neighbor in every block of $G - x$ that is a leaf in the block tree of $G - x$, since G has no cut-vertex.

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, ..., v_n. Then every vertex v_i other than v_n would have at most $k - 1$ neighbors among v_1, v_2, ..., v_{i-1}, while two of the neighbors of v_n, namely v_1 and v_2 would both receive color 1 by the greedy coloring. Hence, it suffices to show that every 2-connected k-regular graph G with $k \geq 3$, not a clique, not an odd cycle, has 3-vertex induced path v_1, v_n, v_2 such that $G - \{v_1, v_2\}$ is connected. Choose a vertex x. If $\kappa(G - x) \geq 2$, then let $v_1 = x$, and let v_2 be a vertex of distance two from x, which exists, since G is connected, regular and not a clique. If $\kappa(G - x) = 1$, then x has a neighbor in every block of $G - x$ that is a leaf in the block tree of $G - x$, since G has no cut-vertex. Neighbors v_1 and v_2 of x in two such blocks are non-adjacent.

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, ..., v_n. Then every vertex v_i other than v_n would have at most $k - 1$ neighbors among v_1, v_2, ..., v_{i-1}, while two of the neighbors of v_n, namely v_1 and v_2 would both receive color 1 by the greedy coloring. Hence, it suffices to show that every 2-connected k-regular graph G with $k \geq 3$, not a clique, not an odd cycle, has 3-vertex induced path v_1, v_n, v_2 such that $G - \{v_1, v_2\}$ is connected. Choose a vertex x. If $\kappa(G - x) \geq 2$, then let $v_1 = x$, and let v_2 be a vertex of distance two from x, which exists, since G is connected, regular and not a clique. If $\kappa(G - x) = 1$, then x has a neighbor in every block of $G - x$ that is a leaf in the block tree of $G - x$, since G has no cut-vertex. Neighbors v_1 and v_2 of x in two such blocks are non-adjacent. Furthermore, $G - \{x, v_1, v_2\}$ is connected, since blocks have no cut-vertices.

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, $\ldots$, v_n. Then every vertex v_i other than v_n would have at most $k-1$ neighbors among v_1, v_2, $\ldots$, v_{i-1}, while two of the neighbors of v_n, namely v_1 and v_2 would both receive color 1 by the greedy coloring. Hence, it suffices to show that every 2-connected k-regular graph G with $k \geq 3$, not a clique, not an odd cycle, has 3-vertex induced path v_1, v_n, v_2 such that $G - \{v_1, v_2\}$ is connected. Choose a vertex x. If $\kappa(G - x) \geq 2$, then let $v_1 = x$, and let v_2 be a vertex of distance two from x, which exists, since G is connected, regular and not a clique. If $\kappa(G - x) = 1$, then x has a neighbor in every block of $G - x$ that is a leaf in the block tree of $G - x$, since G has no cut-vertex. Neighbors v_1 and v_2 of x in two such blocks are non-adjacent. Furthermore, $G - \{x, v_1, v_2\}$ is connected, since blocks have no cut-vertices. Now, $G - \{v_1, v_2\}$ is also connected, since $k \geq 3$ and so x has a neighbor other than v_1 and v_2.

Proof of Brooks' Theorem, Continued

Thus we may assume that G is 2-connected. Suppose that G has an induced 3-vertex path with vertices we label v_1, v_n, v_2, in order, such that $G - \{v_1, v_2\}$ is connected. We could then index the vertices of a spanning tree of $G - \{v_1, v_2\}$ as before, so that every v_i other than v_n has at least one neighbor among v_{i+1}, v_{i+2}, ..., v_n. Then every vertex v_i other than v_n would have at most $k-1$ neighbors among v_1, v_2, ..., v_{i-1}, while two of the neighbors of v_n, namely v_1 and v_2 would both receive color 1 by the greedy coloring. Hence, it suffices to show that every 2-connected k-regular graph G with $k \geq 3$, not a clique, not an odd cycle, has 3-vertex induced path v_1, v_n, v_2 such that $G - \{v_1, v_2\}$ is connected. Choose a vertex x. If $\kappa(G - x) \geq 2$, then let $v_1 = x$, and let v_2 be a vertex of distance two from x, which exists, since G is connected, regular and not a clique. If $\kappa(G - x) = 1$, then x has a neighbor in every block of $G - x$ that is a leaf in the block tree of $G - x$, since G has no cut-vertex. Neighbors v_1 and v_2 of x in two such blocks are non-adjacent. Furthermore, $G - \{x, v_1, v_2\}$ is connected, since blocks have no cut-vertices. Now, $G - \{v_1, v_2\}$ is also connected, since $k \geq 3$ and so x has a neighbor other than v_1 and v_2. Let $v_n = x$ to complete the proof.

5-Color Theorem

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5*-coloring.*

5-Color Theorem

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5-coloring.

Proof.

Suppose G is a plane graph that is a minimal counter-example.

$\square$

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5*-coloring.*

Proof.

Suppose G is a plane graph that is a minimal counter-example. Then G is simple, and so $\|G\| \leq 3|G| - 6$ by Corollary 6.15.

$\square$

5-Color Theorem

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5*-coloring.*

Proof.

Suppose G is a plane graph that is a minimal counter-example. Then G is simple, and so $\|G\| \leq 3|G| - 6$ by Corollary 6.15. It follows that G has a vertex v of degree 5 or less.

$\square$

5-Color Theorem

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5-coloring.

Proof.

Suppose G is a plane graph that is a minimal counter-example. Then G is simple, and so $\|G\| \leq 3|G| - 6$ by Corollary 6.15. It follows that G has a vertex v of degree 5 or less. Then $G - v$ has a 5-coloring f by the minimality of G.

$\square$

5-Color Theorem

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5-coloring.

Proof.

Suppose G is a plane graph that is a minimal counter-example. Then G is simple, and so $\|G\| \leq 3|G| - 6$ by Corollary 6.15. It follows that G has a vertex v of degree 5 or less. Then $G - v$ has a 5-coloring f by the minimality of G. Since G is not 5-colorable, each color appears at one of the neighbors of v (and so $d(v) = 5$).

□

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5*-coloring.*

Proof.

Suppose G is a plane graph that is a minimal counter-example. Then G is simple, and so $\|G\| \leq 3|G| - 6$ by Corollary 6.15. It follows that G has a vertex v of degree 5 or less. Then $G - v$ has a 5-coloring f by the minimality of G. Since G is not 5-colorable, each color appears at one of the neighbors of v (and so $d(v) = 5$). We may assume that the colors on the neighbors of v appear as 1, 2, 3, 4, and 5 as they are inspected clockwise.

□

5-Color Theorem

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5*-coloring.*

Proof.

Suppose G is a plane graph that is a minimal counter-example. Then G is simple, and so $\|G\| \leq 3|G| - 6$ by Corollary 6.15. It follows that G has a vertex v of degree 5 or less. Then $G - v$ has a 5-coloring f by the minimality of G. Since G is not 5-colorable, each color appears at one of the neighbors of v (and so $d(v) = 5$). We may assume that the colors on the neighbors of v appear as 1, 2, 3, 4, and 5 as they are inspected clockwise. Let $G_{i,j}$ denote the subgraph of $G - v$ induced by the colors i and j.

□

5-Color Theorem

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5*-coloring.*

Proof.

Suppose G is a plane graph that is a minimal counter-example. Then G is simple, and so $\|G\| \leq 3|G| - 6$ by Corollary 6.15. It follows that G has a vertex v of degree 5 or less. Then $G - v$ has a 5-coloring f by the minimality of G. Since G is not 5-colorable, each color appears at one of the neighbors of v (and so $d(v) = 5$). We may assume that the colors on the neighbors of v appear as 1, 2, 3, 4, and 5 as they are inspected clockwise. Let $G_{i,j}$ denote the subgraph of $G - v$ induced by the colors i and j. Note that we an exchange the two colors on any component of $G_{i,j}$ to obtain another proper coloring of $G - v$.

$\square$

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5*-coloring.*

Proof.

Suppose G is a plane graph that is a minimal counter-example. Then G is simple, and so $\|G\| \leq 3|G| - 6$ by Corollary 6.15. It follows that G has a vertex v of degree 5 or less. Then $G - v$ has a 5-coloring f by the minimality of G. Since G is not 5-colorable, each color appears at one of the neighbors of v (and so $d(v) = 5$). We may assume that the colors on the neighbors of v appear as 1, 2, 3, 4, and 5 as they are inspected clockwise. Let $G_{i,j}$ denote the subgraph of $G - v$ induced by the colors i and j. Note that we an exchange the two colors on any component of $G_{i,j}$ to obtain another proper coloring of $G - v$. If some two neighbors of v were in different components of $G_{i,j}$, then switching colors on one such component would result in two neighbors of G being colored the same,

$\square$

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5*-coloring.*

Proof.

Suppose G is a plane graph that is a minimal counter-example. Then G is simple, and so $\|G\| \leq 3|G| - 6$ by Corollary 6.15. It follows that G has a vertex v of degree 5 or less. Then $G - v$ has a 5-coloring f by the minimality of G. Since G is not 5-colorable, each color appears at one of the neighbors of v (and so $d(v) = 5$). We may assume that the colors on the neighbors of v appear as 1, 2, 3, 4, and 5 as they are inspected clockwise. Let $G_{i,j}$ denote the subgraph of $G - v$ induced by the colors i and j. Note that we an exchange the two colors on any component of $G_{i,j}$ to obtain another proper coloring of $G - v$. If some two neighbors of v were in different components of $G_{i,j}$, then switching colors on one such component would result in two neighbors of G being colored the same, Thus allowing to extend the coloring to v.

$\square$

Theorem 7.3 (Heawood 1890)

Every loopless planar graph has a proper 5-coloring.

Proof.

Suppose G is a plane graph that is a minimal counter-example. Then G is simple, and so $\|G\| \leq 3|G| - 6$ by Corollary 6.15. It follows that G has a vertex v of degree 5 or less. Then $G - v$ has a 5-coloring f by the minimality of G. Since G is not 5-colorable, each color appears at one of the neighbors of v (and so $d(v) = 5$). We may assume that the colors on the neighbors of v appear as 1, 2, 3, 4, and 5 as they are inspected clockwise. Let $G_{i,j}$ denote the subgraph of $G - v$ induced by the colors i and j. Note that we an exchange the two colors on any component of $G_{i,j}$ to obtain another proper coloring of $G - v$. If some two neighbors of v were in different components of $G_{i,j}$, then switching colors on one such component would result in two neighbors of G being colored the same, Thus allowing to extend the coloring to v. Thus we may assume that every two neighbors of v are in the same component of $G_{i,j}$. □

Proof, Continued

Let $P_{i,j}$ be a path in $G_{i,j}$ joining the neighbors of v colored i and j.

Proof, Continued

Let $P_{i,j}$ be a path in $G_{i,j}$ joining the neighbors of v colored i and j. Consider the cycle C of G induced by $P_{1,3}$ together with v,

Proof, Continued

Let $P_{i,j}$ be a path in $G_{i,j}$ joining the neighbors of v colored i and j. Consider the cycle C of G induced by $P_{1,3}$ together with v, which separates the neighbor of v colored 2 from the one colored 4.

Proof, Continued

Let $P_{i,j}$ be a path in $G_{i,j}$ joining the neighbors of v colored i and j. Consider the cycle C of G induced by $P_{1,3}$ together with v, which separates the neighbor of v colored 2 from the one colored 4. Hence $P_{2,4}$ must cross C, which is impossible.

Proof, Continued

Let $P_{i,j}$ be a path in $G_{i,j}$ joining the neighbors of v colored i and j. Consider the cycle C of G induced by $P_{1,3}$ together with v, which separates the neighbor of v colored 2 from the one colored 4. Hence $P_{2,4}$ must cross C, which is impossible.

Theorem 7.4 (4-Color Theorem, Appel and Haken 1977)

Every loopless planar graph has a proper 4-coloring.

Proof, Continued

Let $P_{i,j}$ be a path in $G_{i,j}$ joining the neighbors of v colored i and j. Consider the cycle C of G induced by $P_{1,3}$ together with v, which separates the neighbor of v colored 2 from the one colored 4. Hence $P_{2,4}$ must cross C, which is impossible.

Theorem 7.4 (4-Color Theorem, Appel and Haken 1977)

Every loopless planar graph has a proper 4-coloring.

Proof omitted.

Discharging Method

Theorem 7.5

Suppose G is a plane triangulation with $\delta(G) \geq 5$. Then G contains two adjacent vertices one of which has degree 5, and the other has degree 5 or 6.

$\square$

Discharging Method

Theorem 7.5

Suppose G is a plane triangulation with $\delta(G) \geq 5$. Then G contains two adjacent vertices one of which has degree 5, and the other has degree 5 or 6.

Proof.

Suppose G is as described, but the conclusion fails.

$\square$

Theorem 7.5

Suppose G is a plane triangulation with $\delta(G) \geq 5$. Then G contains two adjacent vertices one of which has degree 5, and the other has degree 5 or 6.

Proof.

Suppose G is as described, but the conclusion fails. Let v, e, and f be, respectively, the number of vertices, edges, and faces of G. Since G is a plane triangulation, $3f = 2e$, and the Euler Formula implies $e = 3v - 6$.

$\square$

Discharging Method

Theorem 7.5

Suppose G is a plane triangulation with $\delta(G) \geq 5$. Then G contains two adjacent vertices one of which has degree 5, and the other has degree 5 or 6.

Proof.

Suppose G is as described, but the conclusion fails. Let v, e, and f be, respectively, the number of vertices, edges, and faces of G. Since G is a plane triangulation, $3f = 2e$, and the Euler Formula implies $e = 3v - 6$. To each vertex u, assign a charge of $6 - d(u)$.

$\square$

Discharging Method

Theorem 7.5

Suppose G is a plane triangulation with $\delta(G) \geq 5$. Then G contains two adjacent vertices one of which has degree 5, and the other has degree 5 or 6.

Proof.

Suppose G is as described, but the conclusion fails. Let v, e, and f be, respectively, the number of vertices, edges, and faces of G. Since G is a plane triangulation, $3f = 2e$, and the Euler Formula implies $e = 3v - 6$. To each vertex u, assign a charge of $6 - d(u)$. Note that the sum of all the charges in G is

$$\sum_{u \in V(G)} (6 - d(u)) = 6v - 2e = 6v - 2(3v - 6) = 12.$$

□

Discharging Method

Theorem 7.5

Suppose G is a plane triangulation with $\delta(G) \geq 5$. Then G contains two adjacent vertices one of which has degree 5, and the other has degree 5 or 6.

Proof.

Suppose G is as described, but the conclusion fails. Let v, e, and f be, respectively, the number of vertices, edges, and faces of G. Since G is a plane triangulation, $3f = 2e$, and the Euler Formula implies $e = 3v - 6$. To each vertex u, assign a charge of $6 - d(u)$. Note that the sum of all the charges in G is

$$\sum_{u \in V(G)} (6 - d(u)) = 6v - 2e = 6v - 2(3v - 6) = 12.$$

Now, set the discharging rule that each degree-5 vertex sends $1/5$ to each of its neighbors.

□

Discharging Method

Theorem 7.5

Suppose G is a plane triangulation with $\delta(G) \geq 5$. Then G contains two adjacent vertices one of which has degree 5, and the other has degree 5 or 6.

Proof.

Suppose G is as described, but the conclusion fails. Let v, e, and f be, respectively, the number of vertices, edges, and faces of G. Since G is a plane triangulation, $3f = 2e$, and the Euler Formula implies $e = 3v - 6$. To each vertex u, assign a charge of $6 - d(u)$. Note that the sum of all the charges in G is

$$\sum_{u \in V(G)} (6 - d(u)) = 6v - 2e = 6v - 2(3v - 6) = 12.$$

Now, set the discharging rule that each degree-5 vertex sends $1/5$ to each of its neighbors. The final charge of vertices of degree 5 or 6 is zero.

□

Discharging Method

Theorem 7.5

Suppose G is a plane triangulation with $\delta(G) \geq 5$. Then G contains two adjacent vertices one of which has degree 5, and the other has degree 5 or 6.

Proof.

Suppose G is as described, but the conclusion fails. Let v, e, and f be, respectively, the number of vertices, edges, and faces of G. Since G is a plane triangulation, $3f = 2e$, and the Euler Formula implies $e = 3v - 6$. To each vertex u, assign a charge of $6 - d(u)$. Note that the sum of all the charges in G is

$$\sum_{u \in V(G)} (6 - d(u)) = 6v - 2e = 6v - 2(3v - 6) = 12.$$

Now, set the discharging rule that each degree-5 vertex sends $1/5$ to each of its neighbors. The final charge of vertices of degree 5 or 6 is zero. If $d(u) \geq 7$, then the final charge of u is at most $6 - d(u) + d(u)/5$, and so it can be positive only if $d(u) = 7$.

$\square$

Discharging Method

Theorem 7.5

Suppose G is a plane triangulation with $\delta(G) \geq 5$. Then G contains two adjacent vertices one of which has degree 5, and the other has degree 5 or 6.

Proof.

Suppose G is as described, but the conclusion fails. Let v, e, and f be, respectively, the number of vertices, edges, and faces of G. Since G is a plane triangulation, $3f = 2e$, and the Euler Formula implies $e = 3v - 6$. To each vertex u, assign a charge of $6 - d(u)$. Note that the sum of all the charges in G is

$$\sum_{u \in V(G)} (6 - d(u)) = 6v - 2e = 6v - 2(3v - 6) = 12.$$

Now, set the discharging rule that each degree-5 vertex sends $1/5$ to each of its neighbors. The final charge of vertices of degree 5 or 6 is zero. If $d(u) \geq 7$, then the final charge of u is at most $6 - d(u) + d(u)/5$, and so it can be positive only if $d(u) = 7$. But a degree-7 vertex with a positive final charge would have to have six neighbors of degree 5,

$\square$

Discharging Method

Theorem 7.5

Suppose G is a plane triangulation with $\delta(G) \geq 5$. Then G contains two adjacent vertices one of which has degree 5, and the other has degree 5 or 6.

Proof.

Suppose G is as described, but the conclusion fails. Let v, e, and f be, respectively, the number of vertices, edges, and faces of G. Since G is a plane triangulation, $3f = 2e$, and the Euler Formula implies $e = 3v - 6$. To each vertex u, assign a charge of $6 - d(u)$. Note that the sum of all the charges in G is

$$\sum_{u \in V(G)} (6 - d(u)) = 6v - 2e = 6v - 2(3v - 6) = 12.$$

Now, set the discharging rule that each degree-5 vertex sends $1/5$ to each of its neighbors. The final charge of vertices of degree 5 or 6 is zero. If $d(u) \geq 7$, then the final charge of u is at most $6 - d(u) + d(u)/5$, and so it can be positive only if $d(u) = 7$. But a degree-7 vertex with a positive final charge would have to have six neighbors of degree 5, which implies that two of such neighbors must be adjacent since G is a triangulation; a contradiction. $\square$

Chromatic Polynomial

Let $P_G(x)$ denote the number of ways to properly color a (labeled) graph G with x colors.

Chromatic Polynomial

Let $P_G(x)$ denote the number of ways to properly color a (labeled) graph G with x colors. If G has loops, then $P_G(x) = 0$.

Chromatic Polynomial

Let $P_G(x)$ denote the number of ways to properly color a (labeled) graph G with x colors. If G has loops, then $P_G(x) = 0$. If G is edgeless of order n, then $P_G(x) = x^n$.

Chromatic Polynomial

Let $P_G(x)$ denote the number of ways to properly color a (labeled) graph G with x colors. If G has loops, then $P_G(x) = 0$. If G is edgeless of order n, then $P_G(x) = x^n$. If e is a non-loop non-multiple edge of G incident with u and v, then the proper colorings of $G \setminus e$ with x colors can be partitioned into two sets: A, in which u and v receive the same color, and B, in which they do not.

Chromatic Polynomial

Let $P_G(x)$ denote the number of ways to properly color a (labeled) graph G with x colors. If G has loops, then $P_G(x) = 0$. If G is edgeless of order n, then $P_G(x) = x^n$. If e is a non-loop non-multiple edge of G incident with u and v, then the proper colorings of $G \setminus e$ with x colors can be partitioned into two sets: A, in which u and v receive the same color, and B, in which they do not. Then A corresponds to proper colorings of G/e with x colors, and B corresponds to proper colorings of G.

Chromatic Polynomial

Let $P_G(x)$ denote the number of ways to properly color a (labeled) graph G with x colors. If G has loops, then $P_G(x) = 0$. If G is edgeless of order n, then $P_G(x) = x^n$. If e is a non-loop non-multiple edge of G incident with u and v, then the proper colorings of $G \setminus e$ with x colors can be partitioned into two sets: A, in which u and v receive the same color, and B, in which they do not. Then A corresponds to proper colorings of G/e with x colors, and B corresponds to proper colorings of G. Hence

$$P_G(x) = P_{G\setminus e}(x) - P_{G/e}(x).$$

Chromatic Polynomial

Let $P_G(x)$ denote the number of ways to properly color a (labeled) graph G with x colors. If G has loops, then $P_G(x) = 0$. If G is edgeless of order n, then $P_G(x) = x^n$. If e is a non-loop non-multiple edge of G incident with u and v, then the proper colorings of $G \setminus e$ with x colors can be partitioned into two sets: A, in which u and v receive the same color, and B, in which they do not. Then A corresponds to proper colorings of G/e with x colors, and B corresponds to proper colorings of G. Hence

$$P_G(x) = P_{G\setminus e}(x) - P_{G/e}(x).$$

$P_G(x)$ is called the chromatic polynomial of G.

Chromatic Polynomial

Let $P_G(x)$ denote the number of ways to properly color a (labeled) graph G with x colors. If G has loops, then $P_G(x) = 0$. If G is edgeless of order n, then $P_G(x) = x^n$. If e is a non-loop non-multiple edge of G incident with u and v, then the proper colorings of $G \setminus e$ with x colors can be partitioned into two sets: A, in which u and v receive the same color, and B, in which they do not. Then A corresponds to proper colorings of G/e with x colors, and B corresponds to proper colorings of G. Hence

$$P_G(x) = P_{G\setminus e}(x) - P_{G/e}(x).$$

$P_G(x)$ is called the chromatic polynomial of G.

Theorem 7.6 (Four-Color Theorem, restated)

If G is a planar loopless graph, then $P_G(4) > 0$.

Perfect Graphs

Definition 7.7

A graph G is perfect if $\chi(H) = \omega(H)$ for every induced subgraph H of G.

Perfect Graphs

Definition 7.7

A graph G is perfect if $\chi(H) = \omega(H)$ for every induced subgraph H of G.

Theorem 7.8 (Perfect Graph Theorem, Lovász 1972)

A graph is perfect if and only if its complement is perfect.

Perfect Graphs

Definition 7.7

A graph G is perfect if $\chi(H) = \omega(H)$ for every induced subgraph H of G.

Theorem 7.8 (Perfect Graph Theorem, Lovász 1972)

A graph is perfect if and only if its complement is perfect.

Theorem 7.9 (Strong Graph Theorem (formerly Berge's Strong Graph Conjecture), Chudnovsky, Robertson, Seymour, Thomas 2002)

A graph is perfect if and only if it has no induced subgraph that is an odd cycle of length at least five or its complement.

Edge Colorings

Definition 7.10

- A *k-edge-coloring* of a graph G is a labeling $f : E(G) \to \{1, 2, \ldots, k\}$.

Edge Colorings

Definition 7.10

- A k-edge-coloring of a graph G is a labeling $f : E(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors and the edge-set with one color is a color class.

Edge Colorings

Definition 7.10

- A k-edge-coloring of a graph G is a labeling $f : E(G) \to \{1, 2, \dots, k\}$.
- The labels are colors and the edge-set with one color is a color class.
- A k-edge-coloring is proper if adjacent edges have different colors,

Edge Colorings

Definition 7.10

- A k-edge-coloring of a graph G is a labeling $f : E(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors and the edge-set with one color is a color class.
- A k-edge-coloring is proper if adjacent edges have different colors, or equivalently, if every color class is a matching.

Edge Colorings

Definition 7.10

- A k-edge-coloring of a graph G is a labeling $f : E(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors and the edge-set with one color is a color class.
- A k-edge-coloring is proper if adjacent edges have different colors, or equivalently, if every color class is a matching.
- A graph is k-edge-colorable if it has a proper k-edge-coloring.

Edge Colorings

Definition 7.10

- A k-edge-coloring of a graph G is a labeling $f : E(G) \to \{1, 2, \dots, k\}$.
- The labels are colors and the edge-set with one color is a color class.
- A k-edge-coloring is proper if adjacent edges have different colors, or equivalently, if every color class is a matching.
- A graph is k-edge-colorable if it has a proper k-edge-coloring.
- The chromatic index or edge chromatic number $\chi'(G)$ of a loopless graph G is the least k such that G is k-edge-colorable.

Edge Colorings

Definition 7.10

- A k-edge-coloring of a graph G is a labeling $f : E(G) \to \{1, 2, \ldots, k\}$.
- The labels are colors and the edge-set with one color is a color class.
- A k-edge-coloring is proper if adjacent edges have different colors, or equivalently, if every color class is a matching.
- A graph is k-edge-colorable if it has a proper k-edge-coloring.
- The chromatic index or edge chromatic number $\chi'(G)$ of a loopless graph G is the least k such that G is k-edge-colorable.

Note 7.11

$\Delta(G) \le \chi'(G)$.

Edge-Coloring of Bipartite Graphs

Theorem 7.12 (König 1916)

If G is bipartite, then $\chi'(G) = \Delta(G)$.

Edge-Coloring of Bipartite Graphs

Theorem 7.12 (König 1916)

If G is bipartite, then $\chi'(G) = \Delta(G)$.

Proof.

We showed in Corollary 3.11 that every non-trivial regular bipartite graph H has a perfect matching.

$\square$

Edge-Coloring of Bipartite Graphs

Theorem 7.12 (König 1916)

If G is bipartite, then $\chi'(G) = \Delta(G)$.

Proof.

We showed in Corollary 3.11 that every non-trivial regular bipartite graph H has a perfect matching. By induction on $\Delta(H)$, this yields a proper Δ-edge-coloring of H.

$\square$

Edge-Coloring of Bipartite Graphs

Theorem 7.12 (König 1916)

If G is bipartite, then $\chi'(G) = \Delta(G)$.

Proof.

We showed in Corollary 3.11 that every non-trivial regular bipartite graph H has a perfect matching. By induction on $\Delta(H)$, this yields a proper Δ-edge-coloring of H. Thus it suffices to prove that every bipartite graph G of maximum degree k is a subgraph of a k-regular bipartite graph H.

□

Edge-Coloring of Bipartite Graphs

Theorem 7.12 (König 1916)

If G is bipartite, then $\chi'(G) = \Delta(G)$.

Proof.

We showed in Corollary 3.11 that every non-trivial regular bipartite graph H has a perfect matching. By induction on $\Delta(H)$, this yields a proper Δ-edge-coloring of H. Thus it suffices to prove that every bipartite graph G of maximum degree k is a subgraph of a k-regular bipartite graph H. Add the vertices to the smaller side to equalize the sides, if necessary.

$\square$

Edge-Coloring of Bipartite Graphs

Theorem 7.12 (König 1916)

If G is bipartite, then $\chi'(G) = \Delta(G)$.

Proof.

We showed in Corollary 3.11 that every non-trivial regular bipartite graph H has a perfect matching. By induction on $\Delta(H)$, this yields a proper Δ-edge-coloring of H. Thus it suffices to prove that every bipartite graph G of maximum degree k is a subgraph of a k-regular bipartite graph H. Add the vertices to the smaller side to equalize the sides, if necessary. If the resulting graph is not k-regular, then each side has a vertex with degree less than k.

□

Edge-Coloring of Bipartite Graphs

Theorem 7.12 (König 1916)

If G is bipartite, then $\chi'(G) = \Delta(G)$.

Proof.

We showed in Corollary 3.11 that every non-trivial regular bipartite graph H has a perfect matching. By induction on $\Delta(H)$, this yields a proper Δ-edge-coloring of H. Thus it suffices to prove that every bipartite graph G of maximum degree k is a subgraph of a k-regular bipartite graph H. Add the vertices to the smaller side to equalize the sides, if necessary. If the resulting graph is not k-regular, then each side has a vertex with degree less than k. Add an edge joining such a pair of vertices.

□

Edge-Coloring of Bipartite Graphs

Theorem 7.12 (König 1916)

If G is bipartite, then $\chi'(G) = \Delta(G)$.

Proof.

We showed in Corollary 3.11 that every non-trivial regular bipartite graph H has a perfect matching. By induction on $\Delta(H)$, this yields a proper Δ-edge-coloring of H. Thus it suffices to prove that every bipartite graph G of maximum degree k is a subgraph of a k-regular bipartite graph H. Add the vertices to the smaller side to equalize the sides, if necessary. If the resulting graph is not k-regular, then each side has a vertex with degree less than k. Add an edge joining such a pair of vertices. Continue adding edges until the graph becomes regular (although not necessarily simple). □

Vizing's Theorem

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Proof.

Suppose uv is an edge left uncolored by a proper $(\Delta(G) + 1)$-edge-coloring of a proper subgraph G' of G.

$\square$

Vizing's Theorem

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Proof.

Suppose uv is an edge left uncolored by a proper $(\Delta(G)+1)$-edge-coloring of a proper subgraph G' of G. After possibly re-coloring some edges, we extend this coloring to include uv; call this an augmentation.

$\square$

Vizing's Theorem

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Proof.

Suppose uv is an edge left uncolored by a proper $(\Delta(G) + 1)$-edge-coloring of a proper subgraph G' of G. After possibly re-coloring some edges, we extend this coloring to include uv; call this an augmentation. After an appropriate number of augmentations, we obtain a proper $(\Delta(G) + 1)$-edge-coloring of G.

□

Vizing's Theorem

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Proof.

Suppose uv is an edge left uncolored by a proper $(\Delta(G) + 1)$-edge-coloring of a proper subgraph G' of G. After possibly re-coloring some edges, we extend this coloring to include uv; call this an augmentation. After an appropriate number of augmentations, we obtain a proper $(\Delta(G) + 1)$-edge-coloring of G. Since the number of colors exceeds $\Delta(G)$, every vertex has some color not appearing on its incident edges.

$\square$

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Proof.

Suppose uv is an edge left uncolored by a proper $(\Delta(G) + 1)$-edge-coloring of a proper subgraph G' of G. After possibly re-coloring some edges, we extend this coloring to include uv; call this an augmentation. After an appropriate number of augmentations, we obtain a proper $(\Delta(G) + 1)$-edge-coloring of G. Since the number of colors exceeds $\Delta(G)$, every vertex has some color not appearing on its incident edges. Let a_0 be a color missing at u and let a_1 be a color missing at v.

$\square$

Vizing's Theorem

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Proof.

Suppose uv is an edge left uncolored by a proper $(\Delta(G) + 1)$-edge-coloring of a proper subgraph G' of G. After possibly re-coloring some edges, we extend this coloring to include uv; call this an augmentation. After an appropriate number of augmentations, we obtain a proper $(\Delta(G) + 1)$-edge-coloring of G. Since the number of colors exceeds $\Delta(G)$, every vertex has some color not appearing on its incident edges. Let a_0 be a color missing at u and let a_1 be a color missing at v. We may assume that a_1 appears at u, or it could be used for the edge uv.

□

Vizing's Theorem

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Proof.

Suppose uv is an edge left uncolored by a proper $(\Delta(G)+1)$-edge-coloring of a proper subgraph G' of G. After possibly re-coloring some edges, we extend this coloring to include uv; call this an augmentation. After an appropriate number of augmentations, we obtain a proper $(\Delta(G)+1)$-edge-coloring of G. Since the number of colors exceeds $\Delta(G)$, every vertex has some color not appearing on its incident edges. Let a_0 be a color missing at u and let a_1 be a color missing at v. We may assume that a_1 appears at u, or it could be used for the edge uv. Let v_1 be the neighbor of u along the edge colored a_1.

□

Vizing's Theorem

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Proof.

Suppose uv is an edge left uncolored by a proper $(\Delta(G) + 1)$-edge-coloring of a proper subgraph G' of G. After possibly re-coloring some edges, we extend this coloring to include uv; call this an augmentation. After an appropriate number of augmentations, we obtain a proper $(\Delta(G) + 1)$-edge-coloring of G. Since the number of colors exceeds $\Delta(G)$, every vertex has some color not appearing on its incident edges. Let a_0 be a color missing at u and let a_1 be a color missing at v. We may assume that a_1 appears at u, or it could be used for the edge uv. Let v_1 be the neighbor of u along the edge colored a_1. At v_1 some color a_2 is missing.

□

Vizing's Theorem

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Proof.

Suppose uv is an edge left uncolored by a proper $(\Delta(G)+1)$-edge-coloring of a proper subgraph G' of G. After possibly re-coloring some edges, we extend this coloring to include uv; call this an augmentation. After an appropriate number of augmentations, we obtain a proper $(\Delta(G)+1)$-edge-coloring of G. Since the number of colors exceeds $\Delta(G)$, every vertex has some color not appearing on its incident edges. Let a_0 be a color missing at u and let a_1 be a color missing at v. We may assume that a_1 appears at u, or it could be used for the edge uv. Let v_1 be the neighbor of u along the edge colored a_1. At v_1 some color a_2 is missing. We may assume that a_2 appears at u,

□

Vizing's Theorem

Theorem 7.13 (Vizing 1964–65, Gupta 1966)

If G is simple, then $\chi'(G) \leq \Delta(G) + 1$.

Proof.

Suppose uv is an edge left uncolored by a proper $(\Delta(G)+1)$-edge-coloring of a proper subgraph G' of G. After possibly re-coloring some edges, we extend this coloring to include uv; call this an augmentation. After an appropriate number of augmentations, we obtain a proper $(\Delta(G)+1)$-edge-coloring of G. Since the number of colors exceeds $\Delta(G)$, every vertex has some color not appearing on its incident edges. Let a_0 be a color missing at u and let a_1 be a color missing at v. We may assume that a_1 appears at u, or it could be used for the edge uv. Let v_1 be the neighbor of u along the edge colored a_1. At v_1 some color a_2 is missing. We may assume that a_2 appears at u, or we could re-color uv_1 from a_1 to a_2, and then use a_1 on uv to augment the coloring. □

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i.

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i. If a_{i+1} is missing at u, then we shift color a_j from uv_j to uv_{j-1} for $1 \leq j \leq i$ (where $v_0 = v$) to complete the augmentation (downshifting from i).

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i. If a_{i+1} is missing at u, then we shift color a_j from uv_j to uv_{j-1} for $1 \leq j \leq i$ (where $v_0 = v$) to complete the augmentation (downshifting from i). We are finished unless a_{i+1} appears at u, in which case the process continues.

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i. If a_{i+1} is missing at u, then we shift color a_j from uv_j to uv_{j-1} for $1 \leq j \leq i$ (where $v_0 = v$) to complete the augmentation (downshifting from i). We are finished unless a_{i+1} appears at u, in which case the process continues. Since we have $\Delta(G) + 1$ to choose from, this iterative process of selecting a_{i+1} eventually repeats a color.

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i. If a_{i+1} is missing at u, then we shift color a_j from uv_j to uv_{j-1} for $1 \leq j \leq i$ (where $v_0 = v$) to complete the augmentation (downshifting from i). We are finished unless a_{i+1} appears at u, in which case the process continues. Since we have $\Delta(G) + 1$ to choose from, this iterative process of selecting a_{i+1} eventually repeats a color. (Note that we do not need the fact that each vertex has at most Δ neighbors to augment the coloring by one edge; we just need that there is a color missing at each vertex.)

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i. If a_{i+1} is missing at u, then we shift color a_j from uv_j to uv_{j-1} for $1 \leq j \leq i$ (where $v_0 = v$) to complete the augmentation (downshifting from i). We are finished unless a_{i+1} appears at u, in which case the process continues. Since we have $\Delta(G) + 1$ to choose from, this iterative process of selecting a_{i+1} eventually repeats a color. (Note that we do not need the fact that each vertex has at most Δ neighbors to augment the coloring by one edge; we just need that there is a color missing at each vertex.) Let l be the smallest index such that the color a_{l+1} missing at v_l is in the list $a_1, a_2, \ldots, a_l$; suppose $a_{l+1} = a_k$.

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i. If a_{i+1} is missing at u, then we shift color a_j from uv_j to uv_{j-1} for $1 \leq j \leq i$ (where $v_0 = v$) to complete the augmentation (downshifting from i). We are finished unless a_{i+1} appears at u, in which case the process continues. Since we have $\Delta(G) + 1$ to choose from, this iterative process of selecting a_{i+1} eventually repeats a color. (Note that we do not need the fact that each vertex has at most Δ neighbors to augment the coloring by one edge; we just need that there is a color missing at each vertex.) Let l be the smallest index such that the color a_{l+1} missing at v_l is in the list $a_1, a_2, \ldots, a_l$; suppose $a_{l+1} = a_k$. Note that if a_0 is missing at v_l,

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i. If a_{i+1} is missing at u, then we shift color a_j from uv_j to uv_{j-1} for $1 \leq j \leq i$ (where $v_0 = v$) to complete the augmentation (downshifting from i). We are finished unless a_{i+1} appears at u, in which case the process continues. Since we have $\Delta(G) + 1$ to choose from, this iterative process of selecting a_{i+1} eventually repeats a color. (Note that we do not need the fact that each vertex has at most Δ neighbors to augment the coloring by one edge; we just need that there is a color missing at each vertex.) Let l be the smallest index such that the color a_{l+1} missing at v_l is in the list $a_1, a_2, \ldots, a_l$; suppose $a_{l+1} = a_k$. Note that if a_0 is missing at v_l, then we downshift from v_l and use color a_0 on uv_l to complete the augmentation.

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i. If a_{i+1} is missing at u, then we shift color a_j from uv_j to uv_{j-1} for $1 \leq j \leq i$ (where $v_0 = v$) to complete the augmentation (downshifting from i). We are finished unless a_{i+1} appears at u, in which case the process continues. Since we have $\Delta(G) + 1$ to choose from, this iterative process of selecting a_{i+1} eventually repeats a color. (Note that we do not need the fact that each vertex has at most Δ neighbors to augment the coloring by one edge; we just need that there is a color missing at each vertex.) Let l be the smallest index such that the color a_{l+1} missing at v_l is in the list $a_1, a_2, \ldots, a_l$; suppose $a_{l+1} = a_k$. Note that if a_0 is missing at v_l, then we downshift from v_l and use color a_0 on uv_l to complete the augmentation.

Hence we may assume that a_0 appears at v_l, but a_k does not.

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i. If a_{i+1} is missing at u, then we shift color a_j from uv_j to uv_{j-1} for $1 \leq j \leq i$ (where $v_0 = v$) to complete the augmentation (downshifting from i). We are finished unless a_{i+1} appears at u, in which case the process continues. Since we have $\Delta(G) + 1$ to choose from, this iterative process of selecting a_{i+1} eventually repeats a color. (Note that we do not need the fact that each vertex has at most Δ neighbors to augment the coloring by one edge; we just need that there is a color missing at each vertex.) Let l be the smallest index such that the color a_{l+1} missing at v_l is in the list $a_1, a_2, \ldots, a_l$; suppose $a_{l+1} = a_k$. Note that if a_0 is missing at v_l, then we downshift from v_l and use color a_0 on uv_l to complete the augmentation.

Hence we may assume that a_0 appears at v_l, but a_k does not. Let P be the (unique) maximal path of edges colored a_0 or a_k that begins at v_l.

Proof of Vizing's Theorem, Continued

For $i \geq 2$, we continue this process: Having selected a new color a_i that appears at u, let v_i be the neighbor of u along the edge colored a_i. If a_{i+1} is missing at u, then we shift color a_j from uv_j to uv_{j-1} for $1 \leq j \leq i$ (where $v_0 = v$) to complete the augmentation (downshifting from i). We are finished unless a_{i+1} appears at u, in which case the process continues. Since we have $\Delta(G) + 1$ to choose from, this iterative process of selecting a_{i+1} eventually repeats a color. (Note that we do not need the fact that each vertex has at most Δ neighbors to augment the coloring by one edge; we just need that there is a color missing at each vertex.) Let l be the smallest index such that the color a_{l+1} missing at v_l is in the list $a_1, a_2, \ldots, a_l$; suppose $a_{l+1} = a_k$. Note that if a_0 is missing at v_l, then we downshift from v_l and use color a_0 on uv_l to complete the augmentation.

Hence we may assume that a_0 appears at v_l, but a_k does not. Let P be the (unique) maximal path of edges colored a_0 or a_k that begins at v_l. Switching on P means interchanging the colors a_0 and a_k on the edges of P.

Proof of Vizing's Theorem, Continued

If P reaches v_k, then it does so along an edge colored a_0, continues along the edge colored a_k, and stops at u.

Proof of Vizing's Theorem, Continued

If P reaches v_k, then it does so along an edge colored a_0, continues along the edge colored a_k, and stops at u. In this case, we downshift from k and switch on P.

Proof of Vizing's Theorem, Continued

If P reaches v_k, then it does so along an edge colored a_0, continues along the edge colored a_k, and stops at u. In this case, we downshift from k and switch on P. Similarly, if P reaches v_{k-1}, then it does so along an edge colored a_0, and stops there.

Proof of Vizing's Theorem, Continued

If P reaches v_k, then it does so along an edge colored a_0, continues along the edge colored a_k, and stops at u. In this case, we downshift from k and switch on P. Similarly, if P reaches v_{k-1}, then it does so along an edge colored a_0, and stops there. In that case, we downshift from $k-1$, give color a_0 to uv_{k-1}, and switch on P.

Proof of Vizing's Theorem, Continued

If P reaches v_k, then it does so along an edge colored a_0, continues along the edge colored a_k, and stops at u. In this case, we downshift from k and switch on P. Similarly, if P reaches v_{k-1}, then it does so along an edge colored a_0, and stops there. In that case, we downshift from $k-1$, give color a_0 to uv_{k-1}, and switch on P. Finally, suppose that P reaches neither v_k nor v_{k-1}, and so it ends outside $\{u, v_l, v_k, v_{k-1}\}$.

Proof of Vizing's Theorem, Continued

If P reaches v_k, then it does so along an edge colored a_0, continues along the edge colored a_k, and stops at u. In this case, we downshift from k and switch on P. Similarly, if P reaches v_{k-1}, then it does so along an edge colored a_0, and stops there. In that case, we downshift from $k-1$, give color a_0 to uv_{k-1}, and switch on P. Finally, suppose that P reaches neither v_k nor v_{k-1}, and so it ends outside $\{u, v_l, v_k, v_{k-1}\}$. In that case, we downshift from l, give color a_0 to uv_l, and switch on P.

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v.

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v. We say that G admits an $\mathcal{L}$-coloring if there is a proper coloring of G such that, for every vertex v, the color of v is in the list L_v.

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v. We say that G admits an $\mathcal{L}$-coloring if there is a proper coloring of G such that, for every vertex v, the color of v is in the list L_v. The graph G is k-list-colorable or k-choosable if G admits an $\mathcal{L}$-coloring for every $\mathcal{L} = (L_v)_{v \in V}$ with $|L_v| = k$ for every vertex v.

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v. We say that G admits an $\mathcal{L}$-coloring if there is a proper coloring of G such that, for every vertex v, the color of v is in the list L_v. The graph G is k-list-colorable or k-choosable if G admits an $\mathcal{L}$-coloring for every $\mathcal{L} = (L_v)_{v \in V}$ with $|L_v| = k$ for every vertex v. The smallest k such that G is k-choosable is called the list-chromatic number of G and is denoted by $\mathrm{ch}(G)$.

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v. We say that G admits an $\mathcal{L}$-coloring if there is a proper coloring of G such that, for every vertex v, the color of v is in the list L_v. The graph G is k-list-colorable or k-choosable if G admits an $\mathcal{L}$-coloring for every $\mathcal{L} = (L_v)_{v \in V}$ with $|L_v| = k$ for every vertex v. The smallest k such that G is k-choosable is called the list-chromatic number of G and is denoted by $\mathrm{ch}(G)$.
List colorings of edges are defined analogously, as is the list-chromatic index $\mathrm{ch}'(G)$.

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v. We say that G admits an $\mathcal{L}$-coloring if there is a proper coloring of G such that, for every vertex v, the color of v is in the list L_v. The graph G is k-list-colorable or k-choosable if G admits an $\mathcal{L}$-coloring for every $\mathcal{L} = (L_v)_{v \in V}$ with $|L_v| = k$ for every vertex v. The smallest k such that G is k-choosable is called the list-chromatic number of G and is denoted by $\mathrm{ch}(G)$.

List colorings of edges are defined analogously, as is the list-chromatic index $\mathrm{ch}'(G)$. Note that if $\mathcal{L} = (L_v)_{v \in V}$ is such that all L_v's are identical and of cardinality k, then G admitting an $\mathcal{L}$-coloring is equivalent to G being k-colorable.

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v. We say that G admits an $\mathcal{L}$-coloring if there is a proper coloring of G such that, for every vertex v, the color of v is in the list L_v. The graph G is k-list-colorable or k-choosable if G admits an $\mathcal{L}$-coloring for every $\mathcal{L} = (L_v)_{v \in V}$ with $|L_v| = k$ for every vertex v. The smallest k such that G is k-choosable is called the list-chromatic number of G and is denoted by $\mathrm{ch}(G)$.
List colorings of edges are defined analogously, as is the list-chromatic index $\mathrm{ch}'(G)$. Note that if $\mathcal{L} = (L_v)_{v \in V}$ is such that all L_v's are identical and of cardinality k, then G admitting an $\mathcal{L}$-coloring is equvalent to G being k-colorable. An analogous statement holds for edge-colorings.

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v. We say that G admits an $\mathcal{L}$-coloring if there is a proper coloring of G such that, for every vertex v, the color of v is in the list L_v. The graph G is k-list-colorable or k-choosable if G admits an $\mathcal{L}$-coloring for every $\mathcal{L} = (L_v)_{v \in V}$ with $|L_v| = k$ for every vertex v. The smallest k such that G is k-choosable is called the list-chromatic number of G and is denoted by $\mathrm{ch}(G)$.
List colorings of edges are defined analogously, as is the list-chromatic index $\mathrm{ch}'(G)$. Note that if $\mathcal{L} = (L_v)_{v \in V}$ is such that all L_v's are identical and of cardinality k, then G admitting an $\mathcal{L}$-coloring is equivalent to G being k-colorable. An analogous statement holds for edge-colorings. Thus

$$\mathrm{ch}(G) \geq \chi(G) \quad \text{and} \quad \mathrm{ch}'(G) \geq \chi'(G).$$

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v. We say that G admits an $\mathcal{L}$-coloring if there is a proper coloring of G such that, for every vertex v, the color of v is in the list L_v. The graph G is k-list-colorable or k-choosable if G admits an $\mathcal{L}$-coloring for every $\mathcal{L} = (L_v)_{v \in V}$ with $|L_v| = k$ for every vertex v. The smallest k such that G is k-choosable is called the list-chromatic number of G and is denoted by $\mathrm{ch}(G)$.
List colorings of edges are defined analogously, as is the list-chromatic index $\mathrm{ch}'(G)$. Note that if $\mathcal{L} = (L_v)_{v \in V}$ is such that all L_v's are identical and of cardinality k, then G admitting an $\mathcal{L}$-coloring is equivalent to G being k-colorable. An analogous statement holds for edge-colorings. Thus

$$\mathrm{ch}(G) \geq \chi(G) \quad \text{and} \quad \mathrm{ch}'(G) \geq \chi'(G).$$

But there are graphs for which $\mathrm{ch}(G) \neq \chi(G)$.

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v. We say that G admits an $\mathcal{L}$-coloring if there is a proper coloring of G such that, for every vertex v, the color of v is in the list L_v. The graph G is k-list-colorable or k-choosable if G admits an $\mathcal{L}$-coloring for every $\mathcal{L} = (L_v)_{v \in V}$ with $|L_v| = k$ for every vertex v. The smallest k such that G is k-choosable is called the list-chromatic number of G and is denoted by $\text{ch}(G)$.
List colorings of edges are defined analogously, as is the list-chromatic index $\text{ch}'(G)$. Note that if $\mathcal{L} = (L_v)_{v \in V}$ is such that all L_v's are identical and of cardinality k, then G admitting an $\mathcal{L}$-coloring is equivalent to G being k-colorable. An analogous statement holds for edge-colorings. Thus

$$\text{ch}(G) \geq \chi(G) \quad \text{and} \quad \text{ch}'(G) \geq \chi'(G).$$

But there are graphs for which $\text{ch}(G) \neq \chi(G)$. Consider $K_{3,3}$ where each side of the bipartition has lists $\{1, 2\}$, $\{1, 3\}$, and $\{2, 3\}$.

List Colorings

Suppose G is a graph with the vertex set V, and $\mathcal{L} = (L_v)_{v \in V}$ associates with each vertex v a list L_v of colors available to color v. We say that G admits an $\mathcal{L}$-coloring if there is a proper coloring of G such that, for every vertex v, the color of v is in the list L_v. The graph G is k-list-colorable or k-choosable if G admits an $\mathcal{L}$-coloring for every $\mathcal{L} = (L_v)_{v \in V}$ with $|L_v| = k$ for every vertex v. The smallest k such that G is k-choosable is called the list-chromatic number of G and is denoted by $\mathrm{ch}(G)$.
List colorings of edges are defined analogously, as is the list-chromatic index $\mathrm{ch}'(G)$. Note that if $\mathcal{L} = (L_v)_{v \in V}$ is such that all L_v's are identical and of cardinality k, then G admitting an $\mathcal{L}$-coloring is equivalent to G being k-colorable. An analogous statement holds for edge-colorings. Thus

$$\mathrm{ch}(G) \geq \chi(G) \quad \text{and} \quad \mathrm{ch}'(G) \geq \chi'(G).$$

But there are graphs for which $\mathrm{ch}(G) \neq \chi(G)$. Consider $K_{3,3}$ where each side of the bipartition has lists $\{1, 2\}$, $\{1, 3\}$, and $\{2, 3\}$. The list-chromatic number of this graph is 3, while the chromatic number is 2.

Every Planar Graph Is 5-Choosable

Theorem 7.14 (Thomassen 1994)

Every planar graph is 5-choosable.

Every Planar Graph Is 5-Choosable

Theorem 7.14 (Thomassen 1994)

Every planar graph is 5-choosable.

In fact, we will prove a somewhat stronger statement:

> *Suppose that G is a plane graph such that each internal face is a triangle, and the external face is bounded by a cycle C with vertices v_1, v_2, ... v_k (in this order). Let $\mathcal{L} = (L_v)_{v \in V(G)}$ be the set of lists such that $L_{v_1} = \{1\}$, $L_{v_2} = \{2\}$, $|L_{v_i}| \leq 3$ for all $i \in \{3, 4, \ldots, k\}$, and $|L_w| = 5$ for all vertices w not on C. Then G admits an $\mathcal{L}$-coloring.*

Every Planar Graph Is 5-Choosable

Theorem 7.14 (Thomassen 1994)

Every planar graph is 5-choosable.

In fact, we will prove a somewhat stronger statement:

> *Suppose that G is a plane graph such that each internal face is a triangle, and the external face is bounded by a cycle C with vertices v_1, v_2, ... v_k (in this order). Let $\mathcal{L} = (L_v)_{v \in V(G)}$ be the set of lists such that $L_{v_1} = \{1\}$, $L_{v_2} = \{2\}$, $|L_{v_i}| \leq 3$ for all $i \in \{3, 4, \ldots, k\}$, and $|L_w| = 5$ for all vertices w not on C. Then G admits an $\mathcal{L}$-coloring.*

We prove this by induction.

Every Planar Graph Is 5-Choosable

Theorem 7.14 (Thomassen 1994)

Every planar graph is 5-choosable.

In fact, we will prove a somewhat stronger statement:

> *Suppose that G is a plane graph such that each internal face is a triangle, and the external face is bounded by a cycle C with vertices v_1, v_2, ... v_k (in this order). Let $\mathcal{L} = (L_v)_{v \in V(G)}$ be the set of lists such that $L_{v_1} = \{1\}$, $L_{v_2} = \{2\}$, $|L_{v_i}| \leq 3$ for all $i \in \{3, 4, \dots, k\}$, and $|L_w| = 5$ for all vertices w not on C. Then G admits an $\mathcal{L}$-coloring.*

We prove this by induction. The claim is obvious for the smallest graph for which it makes sense, that is, a triangle.

Every Planar Graph Is 5-Choosable

Theorem 7.14 (Thomassen 1994)

Every planar graph is 5-choosable.

In fact, we will prove a somewhat stronger statement:

> *Suppose that G is a plane graph such that each internal face is a triangle, and the external face is bounded by a cycle C with vertices v_1, v_2, ... v_k (in this order). Let $\mathcal{L} = (L_v)_{v \in V(G)}$ be the set of lists such that $L_{v_1} = \{1\}$, $L_{v_2} = \{2\}$, $|L_{v_i}| \leq 3$ for all $i \in \{3, 4, \ldots, k\}$, and $|L_w| = 5$ for all vertices w not on C. Then G admits an $\mathcal{L}$-coloring.*

We prove this by induction. The claim is obvious for the smallest graph for which it makes sense, that is, a triangle. Suppose the claim is true for every graph on fewer than n vertices, and suppose that G is like described above, and $|G| = n$.

Case 1

Suppose that C has a chord vw.

Case 1

Suppose that C has a chord vw. By re-indexing the vertices if necessary, we may assume that $v_2 = w$.

Case 1

Suppose that C has a chord vw. By re-indexing the vertices if necessary, we may assume that $v_2 = w$. Consider the two cycles C_1 and C_2 contained in $C \cup vw$ with v_1 lying on C_1 but not on C_2, and the graphs G_1 and G_2 bounded by C_1 and C_2, respectively.

Case 1

Suppose that C has a chord vw. By re-indexing the vertices if necessary, we may assume that $v_2 = w$. Consider the two cycles C_1 and C_2 contained in $C \cup vw$ with v_1 lying on C_1 but not on C_2, and the graphs G_1 and G_2 bounded by C_1 and C_2, respectively. Create a list $\mathcal{L}_1$ by restricting $\mathcal{L}$ to $V(G_1)$.

Case 1

Suppose that C has a chord vw. By re-indexing the vertices if necessary, we may assume that $v_2 = w$. Consider the two cycles C_1 and C_2 contained in $C \cup vw$ with v_1 lying on C_1 but not on C_2, and the graphs G_1 and G_2 bounded by C_1 and C_2, respectively. Create a list $\mathcal{L}_1$ by restricting $\mathcal{L}$ to $V(G_1)$. Applying the inductive hypothesis to G_1 we get an $\mathcal{L}_1$ coloring of G_1.

Case 1

Suppose that C has a chord vw. By re-indexing the vertices if necessary, we may assume that $v_2 = w$. Consider the two cycles C_1 and C_2 contained in $C \cup vw$ with v_1 lying on C_1 but not on C_2, and the graphs G_1 and G_2 bounded by C_1 and C_2, respectively. Create a list $\mathcal{L}_1$ by restricting $\mathcal{L}$ to $V(G_1)$. Applying the inductive hypothesis to G_1 we get an $\mathcal{L}_1$ coloring of G_1. Let c be the color assigned by the coloring of G_1 to v, which must be different from the color of w.

Case 1

Suppose that C has a chord vw. By re-indexing the vertices if necessary, we may assume that $v_2 = w$. Consider the two cycles C_1 and C_2 contained in $C \cup vw$ with v_1 lying on C_1 but not on C_2, and the graphs G_1 and G_2 bounded by C_1 and C_2, respectively. Create a list $\mathcal{L}_1$ by restricting $\mathcal{L}$ to $V(G_1)$. Applying the inductive hypothesis to G_1 we get an $\mathcal{L}_1$ coloring of G_1. Let c be the color assigned by the coloring of G_1 to v, which must be different from the color of w. Construct a list $\mathcal{L}_2$ by restricting $\mathcal{L}$ to $V(G_2 - v)$ and assigning $\{c\}$ as the list for v.

Case 1

Suppose that C has a chord vw. By re-indexing the vertices if necessary, we may assume that $v_2 = w$. Consider the two cycles C_1 and C_2 contained in $C \cup vw$ with v_1 lying on C_1 but not on C_2, and the graphs G_1 and G_2 bounded by C_1 and C_2, respectively. Create a list $\mathcal{L}_1$ by restricting $\mathcal{L}$ to $V(G_1)$. Applying the inductive hypothesis to G_1 we get an $\mathcal{L}_1$ coloring of G_1. Let c be the color assigned by the coloring of G_1 to v, which must be different from the color of w. Construct a list $\mathcal{L}_2$ by restricting $\mathcal{L}$ to $V(G_2 - v)$ and assigning $\{c\}$ as the list for v. Now, applying the inductive hypothesis to G_2, we get an $\mathcal{L}_2$-coloring of G_2, which can be combined with the coloring of G_1 to get an $\mathcal{L}$-coloring of G.

Case 2

Suppose now that C has no chord.

Case 2

Suppose now that C has no chord. Consider the neighbors of v_k that are v_1, u_1, u_2, ..., u_m, v_{k-1}, in this order.

Case 2

Suppose now that C has no chord. Consider the neighbors of v_k that are v_1, u_1, u_2, ..., u_m, v_{k-1}, in this order. Let j, and l be two colors in L_{v_k} that are different from 1, and remove j and l (if present) from L_{u_i} for all $i \in \{1, 2, \ldots, m\}$ to create a list $\mathcal{L}'$.

Case 2

Suppose now that C has no chord. Consider the neighbors of v_k that are v_1, u_1, u_2, ..., u_m, v_{k-1}, in this order. Let j, and l be two colors in L_{v_k} that are different from 1, and remove j and l (if present) from L_{u_i} for all $i \in \{1, 2, \ldots, m\}$ to create a list $\mathcal{L}'$. Applying the inductive hypothesis to $G - v_k$ results in an $\mathcal{L}'$-coloring of $G - v_k$.

Case 2

Suppose now that C has no chord. Consider the neighbors of v_k that are v_1, u_1, u_2, ..., u_m, v_{k-1}, in this order. Let j, and l be two colors in L_{v_k} that are different from 1, and remove j and l (if present) from L_{u_i} for all $i \in \{1, 2, \ldots, m\}$ to create a list $\mathcal{L}'$. Applying the inductive hypothesis to $G - v_k$ results in an $\mathcal{L}'$-coloring of $G - v_k$. Extend the coloring to an $\mathcal{L}$-coloring of G by assigning to v_k the color from $\{j, l\}$ that is different from the color of v_{k-1}.

List Colorings

Theorem 7.15

There are simple planar graphs that are not 4-choosable.

Theorem 7.15

There are simple planar graphs that are not 4*-choosable.*

Conjecture 7.16

$ch'(G) = \chi'(G)$.

Flows and Circulations

A flow is an assignment of "values" to directed edges of a graph G so that for every vertex $x \in V(G)$ the net flow into x is zero.

Flows and Circulations

A flow is an assignment of "values" to directed edges of a graph G so that for every vertex $x \in V(G)$ the net flow into x is zero.

Definition 7.17

- Let $\vec{E} = \{(e, x, y) : e \in E, x \in V, y \in V, e = xy\}$. Thus an edge $e = xy$ with $x \neq y$ has two directions (e, x, y) and (e, y, x).

Flows and Circulations

A flow is an assignment of "values" to directed edges of a graph G so that for every vertex $x \in V(G)$ the net flow into x is zero.

Definition 7.17

- Let $\vec{E} = \{(e, x, y) : e \in E, x \in V, y \in V, e = xy\}$. Thus an edge $e = xy$ with $x \neq y$ has two directions (e, x, y) and (e, y, x). A loop $e = xx$ has only one direction.

Flows and Circulations

A flow is an assignment of "values" to directed edges of a graph G so that for every vertex $x \in V(G)$ the net flow into x is zero.

Definition 7.17

- Let $\vec{E} = \{(e, x, y) : e \in E, x \in V, y \in V, e = xy\}$. Thus an edge $e = xy$ with $x \neq y$ has two directions (e, x, y) and (e, y, x). A loop $e = xx$ has only one direction.
- Let $\mathbb{H}$ be an abelian group written additively with neutral element 0 (usually $\mathbb{H} = \mathbb{Z}$ or $\mathbb{H} = \mathbb{Z}_k$, that is, integers modulo k).

Flows and Circulations

A flow is an assignment of "values" to directed edges of a graph G so that for every vertex $x \in V(G)$ the net flow into x is zero.

Definition 7.17

- Let $\vec{E} = \{(e, x, y) : e \in E, x \in V, y \in V, e = xy\}$. Thus an edge $e = xy$ with $x \neq y$ has two directions (e, x, y) and (e, y, x). A loop $e = xx$ has only one direction.
- Let $\mathbb{H}$ be an abelian group written additively with neutral element 0 (usually $\mathbb{H} = \mathbb{Z}$ or $\mathbb{H} = \mathbb{Z}_k$, that is, integers modulo k).
- Given $X, Y \subseteq V(G)$ and $\vec{F} \subseteq \vec{E}$, define $\vec{F}(X, Y) = \{(e, x, y) \in \vec{F} : x \in X, y \in Y, x \neq y\}$.

Flows and Circulations

A flow is an assignment of "values" to directed edges of a graph G so that for every vertex $x \in V(G)$ the net flow into x is zero.

Definition 7.17

- Let $\vec{E} = \{(e, x, y) : e \in E, x \in V, y \in V, e = xy\}$. Thus an edge $e = xy$ with $x \neq y$ has two directions (e, x, y) and (e, y, x). A loop $e = xx$ has only one direction.
- Let $\mathbb{H}$ be an abelian group written additively with neutral element 0 (usually $\mathbb{H} = \mathbb{Z}$ or $\mathbb{H} = \mathbb{Z}_k$, that is, integers modulo k).
- Given $X, Y \subseteq V(G)$ and $\vec{F} \subseteq \vec{E}$, define $\vec{F}(X, Y) = \{(e, x, y) \in \vec{F} : x \in X, y \in Y, x \neq y\}$.
- Given $X, Y \subseteq V(G)$ and $f : \vec{E} \to \mathbb{H}$, we write

$$f(X, Y) = \sum_{\vec{e} \in \vec{E}(X,Y)} f(\vec{e}).$$

Flows and Circulations

Definition 7.18

- A function $f : \vec{E} \to \mathbb{H}$ is a circulation or $\mathbb{H}$-circulation if

Flows and Circulations

Definition 7.18

- A function $f : \vec{E} \to \mathbb{H}$ is a circulation or $\mathbb{H}$-circulation if

  (F1) $f(e, x, y) = -f(e, y, x)$ for all $(e, x, y) \in \vec{E}$; and

Flows and Circulations

Definition 7.18

- A function $f : \vec{E} \to \mathbb{H}$ is a circulation or $\mathbb{H}$-circulation if

  (F1) $f(e, x, y) = -f(e, y, x)$ for all $(e, x, y) \in \vec{E}$; and
  (F2) $f(x, V(G)) = 0$ for all $x \in V(G)$ (Kirchhoff's Law)

Flows and Circulations

Definition 7.18

- A function $f : \vec{E} \to \mathbb{H}$ is a circulation or $\mathbb{H}$-circulation if

  (F1) $f(e, x, y) = -f(e, y, x)$ for all $(e, x, y) \in \vec{E}$; and
  (F2) $f(x, V(G)) = 0$ for all $x \in V(G)$ (Kirchhoff's Law)

- A function $f : \vec{E} \to \mathbb{H}$ is nowhere-zero if $f(\vec{e}) \neq 0$ for all $\vec{e} \in \vec{E}$.

Flows and Circulations

Definition 7.18

- A function $f : \vec{E} \to \mathbb{H}$ is a circulation or $\mathbb{H}$-circulation if
  - (F1) $f(e, x, y) = -f(e, y, x)$ for all $(e, x, y) \in \vec{E}$; and
  - (F2) $f(x, V(G)) = 0$ for all $x \in V(G)$ (Kirchhoff's Law)
- A function $f : \vec{E} \to \mathbb{H}$ is nowhere-zero if $f(\vec{e}) \neq 0$ for all $\vec{e} \in \vec{E}$.
- A nowhere-zero $\mathbb{H}$-circulation is called an $\mathbb{H}$-flow.

Flows and Circulations

Definition 7.18

- A function $f : \vec{E} \to \mathbb{H}$ is a circulation or $\mathbb{H}$-circulation if
  (F1) $f(e, x, y) = -f(e, y, x)$ for all $(e, x, y) \in \vec{E}$; and
  (F2) $f(x, V(G)) = 0$ for all $x \in V(G)$ (Kirchhoff's Law)
- A function $f : \vec{E} \to \mathbb{H}$ is nowhere-zero if $f(\vec{e}) \neq 0$ for all $\vec{e} \in \vec{E}$.
- A nowhere-zero $\mathbb{H}$-circulation is called an $\mathbb{H}$-flow.

Note 7.19

- *If f satisfies (F1), then $f(X, X) = 0$ for all $X \subseteq V$.*

Flows and Circulations

Definition 7.18

- A function $f : \vec{E} \to \mathbb{H}$ is a circulation or $\mathbb{H}$-circulation if

  (F1) $f(e,x,y) = -f(e,y,x)$ for all $(e,x,y) \in \vec{E}$; and
  (F2) $f(x, V(G)) = 0$ for all $x \in V(G)$ (Kirchhoff's Law)
- A function $f : \vec{E} \to \mathbb{H}$ is nowhere-zero if $f(\vec{e}) \neq 0$ for all $\vec{e} \in \vec{E}$.
- A nowhere-zero $\mathbb{H}$-circulation is called an $\mathbb{H}$-flow.

Note 7.19

- *If f satisfies (F1), then $f(X, X) = 0$ for all $X \subseteq V$.*
- *If f satisfies (F2), then $f(X, V) = 0$.*

Flows and Circulations

Definition 7.18

- A function $f : \vec{E} \to \mathbb{H}$ is a circulation or $\mathbb{H}$-circulation if

  (F1) $f(e,x,y) = -f(e,y,x)$ for all $(e,x,y) \in \vec{E}$; and
  (F2) $f(x,V(G)) = 0$ for all $x \in V(G)$ (Kirchhoff's Law)

- A function $f : \vec{E} \to \mathbb{H}$ is nowhere-zero if $f(\vec{e}) \neq 0$ for all $\vec{e} \in \vec{E}$.
- A nowhere-zero $\mathbb{H}$-circulation is called an $\mathbb{H}$-flow.

Note 7.19

- *If f satisfies (F1), then $f(X,X) = 0$ for all $X \subseteq V$.*
- *If f satisfies (F2), then $f(X,V) = 0$.*
- *If f is a circulation, then $f(X,\overline{X}) = 0$ for every $X \subseteq V$.*

Definition 7.18

- A function $f : \vec{E} \to \mathbb{H}$ is a circulation or $\mathbb{H}$-circulation if
  (F1) $f(e, x, y) = -f(e, y, x)$ for all $(e, x, y) \in \vec{E}$; and
  (F2) $f(x, V(G)) = 0$ for all $x \in V(G)$ (Kirchhoff's Law)
- A function $f : \vec{E} \to \mathbb{H}$ is nowhere-zero if $f(\vec{e}) \neq 0$ for all $\vec{e} \in \vec{E}$.
- A nowhere-zero $\mathbb{H}$-circulation is called an $\mathbb{H}$-flow.

Note 7.19

- *If f satisfies (F1), then $f(X, X) = 0$ for all $X \subseteq V$.*
- *If f satisfies (F2), then $f(X, V) = 0$.*
- *If f is a circulation, then $f(X, \overline{X}) = 0$ for every $X \subseteq V$.*
- *If f is a circulation and $e = xy$ is a cut-edge, then $f(e, x, y) = 0$.*

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph.

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph. We want to find the number $F_G(x)$ of $\mathbb{H}$-flows in G.

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph. We want to find the number $F_G(x)$ of $\mathbb{H}$-flows in G. If G has a cut-edge, then $F_G(x) = 0$.

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph. We want to find the number $F_G(x)$ of $\mathbb{H}$-flows in G. If G has a cut-edge, then $F_G(x) = 0$. If G has n loops and no other edges, then $F_G(x) = (x-1)^n$.

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph. We want to find the number $F_G(x)$ of $\mathbb{H}$-flows in G. If G has a cut-edge, then $F_G(x) = 0$. If G has n loops and no other edges, then $F_G(x) = (x-1)^n$. Let e be a non-loop edge with endpoints u and v of G.

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph. We want to find the number $F_G(x)$ of $\mathbb{H}$-flows in G. If G has a cut-edge, then $F_G(x) = 0$. If G has n loops and no other edges, then $F_G(x) = (x-1)^n$. Let e be a non-loop edge with endpoints u and v of G. Count the number of $\mathbb{H}$-flows in G/e.

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph. We want to find the number $F_G(x)$ of $\mathbb{H}$-flows in G. If G has a cut-edge, then $F_G(x) = 0$. If G has n loops and no other edges, then $F_G(x) = (x-1)^n$. Let e be a non-loop edge with endpoints u and v of G. Count the number of $\mathbb{H}$-flows in G/e. Those flows can be partitioned into two sets: A, those that induce a flow in $G \setminus e$, and B, those that do not.

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph. We want to find the number $F_G(x)$ of $\mathbb{H}$-flows in G. If G has a cut-edge, then $F_G(x) = 0$. If G has n loops and no other edges, then $F_G(x) = (x-1)^n$. Let e be a non-loop edge with endpoints u and v of G. Count the number of $\mathbb{H}$-flows in G/e. Those flows can be partitioned into two sets: A, those that induce a flow in $G \setminus e$, and B, those that do not. Those flows in A cannot be extended to a flow on G, whereas those in B can.

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph. We want to find the number $F_G(x)$ of $\mathbb{H}$-flows in G. If G has a cut-edge, then $F_G(x) = 0$. If G has n loops and no other edges, then $F_G(x) = (x-1)^n$. Let e be a non-loop edge with endpoints u and v of G. Count the number of $\mathbb{H}$-flows in G/e. Those flows can be partitioned into two sets: A, those that induce a flow in $G \setminus e$, and B, those that do not. Those flows in A cannot be extended to a flow on G, whereas those in B can. So

$$F_G(x) = F_{G/e}(x) - F_{G \setminus e}(x).$$

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph. We want to find the number $F_G(x)$ of $\mathbb{H}$-flows in G. If G has a cut-edge, then $F_G(x) = 0$. If G has n loops and no other edges, then $F_G(x) = (x-1)^n$. Let e be a non-loop edge with endpoints u and v of G. Count the number of $\mathbb{H}$-flows in G/e. Those flows can be partitioned into two sets: A, those that induce a flow in $G \setminus e$, and B, those that do not. Those flows in A cannot be extended to a flow on G, whereas those in B can. So

$$F_G(x) = F_{G/e}(x) - F_{G \setminus e}(x).$$

Clearly, $F_G(x)$ is a polynomial, and is called the flow polynomial of G.

Flow polynomial

Suppose $|\mathbb{H}| = x$ and let G be a graph. We want to find the number $F_G(x)$ of $\mathbb{H}$-flows in G. If G has a cut-edge, then $F_G(x) = 0$. If G has n loops and no other edges, then $F_G(x) = (x-1)^n$. Let e be a non-loop edge with endpoints u and v of G. Count the number of $\mathbb{H}$-flows in G/e. Those flows can be partitioned into two sets: A, those that induce a flow in $G \setminus e$, and B, those that do not. Those flows in A cannot be extended to a flow on G, whereas those in B can. So

$$F_G(x) = F_{G/e}(x) - F_{G\setminus e}(x).$$

Clearly, $F_G(x)$ is a polynomial, and is called the flow polynomial of G. It follows:

Corollary 7.20

If $\mathbb{H}$ and $\mathbb{H}'$ are two finite abelian groups of equal order, then G has an $\mathbb{H}$-flow if and only if it has an $\mathbb{H}'$-flow.

Definition 7.21

- A $\mathbb{Z}$-flow f such that $0 < |f(\vec{e})| < k$ for all $\vec{e} \in \vec{E}$ is a k-flow.

Definition 7.21

- A $\mathbb{Z}$-flow f such that $0 < |f(\vec{e})| < k$ for all $\vec{e} \in \vec{E}$ is a k-flow.
- The flow number of a graph G, denoted by $\varphi(G)$, is the smallest k such that G has a k-flow, or infinite if no k-flow exists.

Definition 7.21

- A $\mathbb{Z}$-flow f such that $0 < |f(\vec{e})| < k$ for all $\vec{e} \in \vec{E}$ is a k-flow.
- The flow number of a graph G, denoted by $\varphi(G)$, is the smallest k such that G has a k-flow, or infinite if no k-flow exists.

Theorem 7.22 (Tutte 1950)

A graph admits a k-flow if and only if it admits a $\mathbb{Z}_k$-flow.

Definition 7.21

- A $\mathbb{Z}$-flow f such that $0 < |f(\vec{e})| < k$ for all $\vec{e} \in \vec{E}$ is a k-flow.
- The flow number of a graph G, denoted by $\varphi(G)$, is the smallest k such that G has a k-flow, or infinite if no k-flow exists.

Theorem 7.22 (Tutte 1950)

A graph admits a k-flow if and only if it admits a $\mathbb{Z}_k$-flow.

Proof of $\Rightarrow$ only.

Use the natural map $i \mapsto \bar{i}$ from $\mathbb{Z}$ to $\mathbb{Z}_k$. $\square$

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow,

$\square$

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2).

□

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. $\square$

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. $\square$

Theorem 7.24 (Tutte 1949)

A cubic graph has a 3-flow if and only if it is bipartite.

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. □

Theorem 7.24 (Tutte 1949)

A cubic graph has a 3-flow if and only if it is bipartite.

Proof.

Let G be a cubic graph.

□

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. □

Theorem 7.24 (Tutte 1949)

A cubic graph has a 3-flow if and only if it is bipartite.

Proof.

Let G be a cubic graph. Suppose first that G has a 3-flow, and thus a $\mathbb{Z}_3$-flow.

□

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. □

Theorem 7.24 (Tutte 1949)

A cubic graph has a 3-flow if and only if it is bipartite.

Proof.

Let G be a cubic graph. Suppose first that G has a 3-flow, and thus a $\mathbb{Z}_3$-flow. We show that every cycle $C = x_0 x_1 \dots x_l x_0$ has even length.

□

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. □

Theorem 7.24 (Tutte 1949)

A cubic graph has a 3-flow if and only if it is bipartite.

Proof.

Let G be a cubic graph. Suppose first that G has a 3-flow, and thus a $\mathbb{Z}_3$-flow. We show that every cycle $C = x_0 x_1 \dots x_l x_0$ has even length. Consider two consecutive edges of C: $e_{i-1} = x_{i-1} x_i$ and $e_i = x_1 x_{i+1}$.

□

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. □

Theorem 7.24 (Tutte 1949)

A cubic graph has a 3-flow if and only if it is bipartite.

Proof.

Let G be a cubic graph. Suppose first that G has a 3-flow, and thus a $\mathbb{Z}_3$-flow. We show that every cycle $C = x_0 x_1 \ldots x_l x_0$ has even length. Consider two consecutive edges of C: $e_{i-1} = x_{i-1} x_i$ and $e_i = x_1 x_{i+1}$. If $f(e_{i-1}, x_{i-1}, x_i) = f(e_i, x_i, x_{i+1})$, then f could not satisfy (F2) at x_i due to a non-zero value of the third edge at x_i.

□

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. □

Theorem 7.24 (Tutte 1949)

A cubic graph has a 3-flow if and only if it is bipartite.

Proof.

Let G be a cubic graph. Suppose first that G has a 3-flow, and thus a $\mathbb{Z}_3$-flow. We show that every cycle $C = x_0 x_1 \dots x_l x_0$ has even length. Consider two consecutive edges of C: $e_{i-1} = x_{i-1}x_i$ and $e_i = x_1 x_{i+1}$. If
$f(e_{i-1}, x_{i-1}, x_i) = f(e_i, x_i, x_{i+1})$, then f could not satisfy (F2) at x_i due to a non-zero value of the third edge at x_i. Therefore f assigns 1 and 2 to the edges of C alternately,

□

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. □

Theorem 7.24 (Tutte 1949)

A cubic graph has a 3-flow if and only if it is bipartite.

Proof.

Let G be a cubic graph. Suppose first that G has a 3-flow, and thus a $\mathbb{Z}_3$-flow. We show that every cycle $C = x_0 x_1 \dots x_l x_0$ has even length. Consider two consecutive edges of C: $e_{i-1} = x_{i-1}x_i$ and $e_i = x_1 x_{i+1}$. If $f(e_{i-1}, x_{i-1}, x_i) = f(e_i, x_i, x_{i+1})$, then f could not satisfy (F2) at x_i due to a non-zero value of the third edge at x_i. Therefore f assigns 1 and 2 to the edges of C alternately, and so C must be even.

□

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. □

Theorem 7.24 (Tutte 1949)

A cubic graph has a 3-flow if and only if it is bipartite.

Proof.

Let G be a cubic graph. Suppose first that G has a 3-flow, and thus a $\mathbb{Z}_3$-flow. We show that every cycle $C = x_0x_1 \dots x_lx_0$ has even length. Consider two consecutive edges of C: $e_{i-1} = x_{i-1}x_i$ and $e_i = x_1x_{i+1}$. If $f(e_{i-1}, x_{i-1}, x_i) = f(e_i, x_i, x_{i+1})$, then f could not satisfy (F2) at x_i due to a non-zero value of the third edge at x_i. Therefore f assigns 1 and 2 to the edges of C alternately, and so C must be even.
Conversely, let G be bipartite with bipartition (X, Y).

□

k-Flows for Small k

Theorem 7.23

A graph has a 2-flow if and only if all vertices have even degree.

Proof.

By Corollary 7.22, a graph has a 2-flow if and only if it has a $\mathbb{Z}_2$-flow, that is, the constant map $\vec{E} \to \mathbb{Z}_2$ with value 1 satisfies (F2). This is the case if and only if every vertex degree is even. □

Theorem 7.24 (Tutte 1949)

A cubic graph has a 3-flow if and only if it is bipartite.

Proof.

Let G be a cubic graph. Suppose first that G has a 3-flow, and thus a $\mathbb{Z}_3$-flow. We show that every cycle $C = x_0 x_1 \ldots x_l x_0$ has even length. Consider two consecutive edges of C: $e_{i-1} = x_{i-1} x_i$ and $e_i = x_1 x_{i+1}$. If $f(e_{i-1}, x_{i-1}, x_i) = f(e_i, x_i, x_{i+1})$, then f could not satisfy (F2) at x_i due to a non-zero value of the third edge at x_i. Therefore f assigns 1 and 2 to the edges of C alternately, and so C must be even.
Conversely, let G be bipartite with bipartition (X, Y). Since G is cubic, the map $\vec{E} \to \mathbb{Z}_3$ defined by $f(e, x, y) = 1$ and $(e, y, x) = 2$ for all edges xy with $x \in X$ and $y \in Y$ is a $\mathbb{Z}_3$-flow. □

Flow Number of Cliques

Theorem 7.25

$$\varphi(K_n) = \begin{cases} 2 & \textit{if } n \textit{ is odd;} \\ 4 & \textit{if } n = 4\textit{; and} \\ 3 & \textit{if } n \textit{ is even and exceeds } 4. \end{cases}$$

Flow Number of Cliques

Theorem 7.25

$$\varphi(K_n) = \begin{cases} 2 & \text{if } n \text{ is odd;} \\ 4 & \text{if } n = 4\text{; and} \\ 3 & \text{if } n \text{ is even and exceeds } 4. \end{cases}$$

Proof.

The case for n odd follows from Theorem 7.23, and $n = 4$ can be checked directly.

$\square$

Flow Number of Cliques

Theorem 7.25

$$\varphi(K_n) = \begin{cases} 2 & \textit{if } n \textit{ is odd;} \\ 4 & \textit{if } n = 4\textit{; and} \\ 3 & \textit{if } n \textit{ is even and exceeds } 4. \end{cases}$$

Proof.

The case for n odd follows from Theorem 7.23, and $n = 4$ can be checked directly. We handle the remaining cases by induction.

□

Flow Number of Cliques

Theorem 7.25

$$\varphi(K_n) = \begin{cases} 2 & \text{if } n \text{ is odd;} \\ 4 & \text{if } n = 4\text{; and} \\ 3 & \text{if } n \text{ is even and exceeds } 4. \end{cases}$$

Proof.

The case for n odd follows from Theorem 7.23, and $n = 4$ can be checked directly. We handle the remaining cases by induction.
Note that K_6 is the edge-disjoint union of G_1, G_2, and G_3 where $G_1 \cong G_2 \cong K_3$ and $G_3 \cong K_{3,3}$.

□

Flow Number of Cliques

Theorem 7.25

$$\varphi(K_n) = \begin{cases} 2 & \textit{if } n \textit{ is odd;} \\ 4 & \textit{if } n = 4\textit{; and} \\ 3 & \textit{if } n \textit{ is even and exceeds } 4. \end{cases}$$

Proof.

The case for n odd follows from Theorem 7.23, and $n = 4$ can be checked directly. We handle the remaining cases by induction.
Note that K_6 is the edge-disjoint union of G_1, G_2, and G_3 where $G_1 \cong G_2 \cong K_3$ and $G_3 \cong K_{3,3}$. Each of G_1 and G_2 has a 2-flow,

□

Flow Number of Cliques

Theorem 7.25

$$\varphi(K_n) = \begin{cases} 2 & \textit{if } n \textit{ is odd;} \\ 4 & \textit{if } n = 4\textit{; and} \\ 3 & \textit{if } n \textit{ is even and exceeds } 4. \end{cases}$$

Proof.

The case for n odd follows from Theorem 7.23, and $n = 4$ can be checked directly. We handle the remaining cases by induction.
Note that K_6 is the edge-disjoint union of G_1, G_2, and G_3 where $G_1 \cong G_2 \cong K_3$ and $G_3 \cong K_{3,3}$. Each of G_1 and G_2 has a 2-flow, while $K_{3,3}$ has a 3-flow by Theorem 7.24. $\square$

Flow Number of Cliques

Theorem 7.25

$$\varphi(K_n) = \begin{cases} 2 & \textit{if } n \textit{ is odd;} \\ 4 & \textit{if } n = 4\textit{; and} \\ 3 & \textit{if } n \textit{ is even and exceeds } 4. \end{cases}$$

Proof.

The case for n odd follows from Theorem 7.23, and $n = 4$ can be checked directly. We handle the remaining cases by induction.
Note that K_6 is the edge-disjoint union of G_1, G_2, and G_3 where $G_1 \cong G_2 \cong K_3$ and $G_3 \cong K_{3,3}$. Each of G_1 and G_2 has a 2-flow, while $K_{3,3}$ has a 3-flow by Theorem 7.24. The union of these flows is a 3-flow on G. □

Proof, Continued

Now let n be even and greater than 6, and assume that the assertion holds for $n - 2$.

Proof, Continued

Now let n be even and greater than 6, and assume that the assertion holds for $n-2$. Now G can be written as edge-disjoint union of K_{n-2} and $G' = \overline{K_{n-2}} \vee K_2$.

Proof, Continued

Now let n be even and greater than 6, and assume that the assertion holds for $n-2$. Now G can be written as edge-disjoint union of K_{n-2} and $G' = \overline{K_{n-2}} \vee K_2$. The K_{n-2} has a 3-flow by induction hypothesis.

Proof, Continued

Now let n be even and greater than 6, and assume that the assertion holds for $n-2$. Now G can be written as edge-disjoint union of K_{n-2} and $G' = \overline{K_{n-2}} \vee K_2$. The K_{n-2} has a 3-flow by induction hypothesis. Therefore it suffices to find a $\mathbb{Z}_3$-flow on G'.

Proof, Continued

Now let n be even and greater than 6, and assume that the assertion holds for $n-2$. Now G can be written as edge-disjoint union of K_{n-2} and $G' = \overline{K_{n-2}} \vee K_2$. The K_{n-2} has a 3-flow by induction hypothesis. Therefore it suffices to find a $\mathbb{Z}_3$-flow on G'. Let x and y be the vertices of K_2.

Proof, Continued

Now let n be even and greater than 6, and assume that the assertion holds for $n-2$. Now G can be written as edge-disjoint union of K_{n-2} and $G' = \overline{K_{n-2}} \vee K_2$. The K_{n-2} has a 3-flow by induction hypothesis. Therefore it suffices to find a $\mathbb{Z}_3$-flow on G'. Let x and y be the vertices of K_2. Then each triangle xyz has a constant $\mathbb{Z}_3$-flow.

Proof, Continued

Now let n be even and greater than 6, and assume that the assertion holds for $n-2$. Now G can be written as edge-disjoint union of K_{n-2} and $G' = \overline{K_{n-2}} \vee K_2$. The K_{n-2} has a 3-flow by induction hypothesis. Therefore it suffices to find a $\mathbb{Z}_3$-flow on G'. Let x and y be the vertices of K_2. Then each triangle xyz has a constant $\mathbb{Z}_3$-flow. Adding all of those flows produces a circulation on G' that is non-zero, except possibly on xy.

Proof, Continued

Now let n be even and greater than 6, and assume that the assertion holds for $n-2$. Now G can be written as edge-disjoint union of K_{n-2} and $G' = \overline{K_{n-2}} \vee K_2$. The K_{n-2} has a 3-flow by induction hypothesis. Therefore it suffices to find a $\mathbb{Z}_3$-flow on G'. Let x and y be the vertices of K_2. Then each triangle xyz has a constant $\mathbb{Z}_3$-flow. Adding all of those flows produces a circulation on G' that is non-zero, except possibly on xy. If that is the case, the multiply exactly one of the flows by 2 before adding them all up.

Proof, Continued

Now let n be even and greater than 6, and assume that the assertion holds for $n-2$. Now G can be written as edge-disjoint union of K_{n-2} and $G' = \overline{K_{n-2}} \vee K_2$. The K_{n-2} has a 3-flow by induction hypothesis. Therefore it suffices to find a $\mathbb{Z}_3$-flow on G'. Let x and y be the vertices of K_2. Then each triangle xyz has a constant $\mathbb{Z}_3$-flow. Adding all of those flows produces a circulation on G' that is non-zero, except possibly on xy. If that is the case, the multiply exactly one of the flows by 2 before adding them all up. The result follows.

4-Flows

Theorem 7.26

(i) *A graph has a* 4*-flow if and only if it is the union of two subgraphs whose vertices have all degrees even.*

Theorem 7.26

(i) *A graph has a 4-flow if and only if it is the union of two subgraphs whose vertices have all degrees even.*

(ii) *A cubic graph has a 4-flow if and only if it is 3-edge-colorable.*

4-Flows

Theorem 7.26

(i) *A graph has a 4-flow if and only if it is the union of two subgraphs whose vertices have all degrees even.*

(ii) *A cubic graph has a 4-flow if and only if it is 3-edge-colorable.*

Proof.

Let $\mathbb{H} = \mathbb{Z}_2 \times \mathbb{Z}_2$.

$\square$

4-Flows

Theorem 7.26

(i) *A graph has a 4-flow if and only if it is the union of two subgraphs whose vertices have all degrees even.*

(ii) *A cubic graph has a 4-flow if and only if it is 3-edge-colorable.*

Proof.

Let $\mathbb{H} = \mathbb{Z}_2 \times \mathbb{Z}_2$. By Theorems 7.20 and 7.22, a graph has a 4-flow if and only if it as an $\mathbb{H}$-flow.

$\square$

Theorem 7.26

(i) *A graph has a 4-flow if and only if it is the union of two subgraphs whose vertices have all degrees even.*

(ii) *A cubic graph has a 4-flow if and only if it is 3-edge-colorable.*

Proof.

Let $\mathbb{H} = \mathbb{Z}_2 \times \mathbb{Z}_2$. By Theorems 7.20 and 7.22, a graph has a 4-flow if and only if it as an $\mathbb{H}$-flow. Now (i) follows immediately from Theorem 7.23.

$\square$

Theorem 7.26

(i) *A graph has a 4-flow if and only if it is the union of two subgraphs whose vertices have all degrees even.*

(ii) *A cubic graph has a 4-flow if and only if it is 3-edge-colorable.*

Proof.

Let $\mathbb{H} = \mathbb{Z}_2 \times \mathbb{Z}_2$. By Theorems 7.20 and 7.22, a graph has a 4-flow if and only if it as an $\mathbb{H}$-flow. Now (i) follows immediately from Theorem 7.23.
Assume a cubic graph G has an $\mathbb{H}$-flow f.

$\square$

Theorem 7.26

(i) *A graph has a 4-flow if and only if it is the union of two subgraphs whose vertices have all degrees even.*

(ii) *A cubic graph has a 4-flow if and only if it is 3-edge-colorable.*

Proof.

Let $\mathbb{H} = \mathbb{Z}_2 \times \mathbb{Z}_2$. By Theorems 7.20 and 7.22, a graph has a 4-flow if and only if it as an $\mathbb{H}$-flow. Now (i) follows immediately from Theorem 7.23.
Assume a cubic graph G has an $\mathbb{H}$-flow f. It is easy to check that f gives a 3-edge-coloring.

$\square$

Theorem 7.26

(i) *A graph has a 4-flow if and only if it is the union of two subgraphs whose vertices have all degrees even.*

(ii) *A cubic graph has a 4-flow if and only if it is 3-edge-colorable.*

Proof.

Let $\mathbb{H} = \mathbb{Z}_2 \times \mathbb{Z}_2$. By Theorems 7.20 and 7.22, a graph has a 4-flow if and only if it as an $\mathbb{H}$-flow. Now (i) follows immediately from Theorem 7.23.
Assume a cubic graph G has an $\mathbb{H}$-flow f. It is easy to check that f gives a 3-edge-coloring. Conversely, since the non-zero elements of $\mathbb{H}$ sum up to 0, every proper 3-edge-coloring of G using colors $\mathbb{H} \setminus 0$ defines an $\mathbb{H}$-flow on G. $\square$

Flow-Coloring Duality

Theorem 7.27 (Tait 1878)

A simple 2-edge-connected 3-regular plane graph G is 3-edge-colorable if and only if its dual is 4-colorable.

Flow-Coloring Duality

Theorem 7.27 (Tait 1878)

A simple 2-edge-connected 3-regular plane graph G is 3-edge-colorable if and only if its dual is 4-colorable.

Proof.

Suppose G is 4-face-colored with elements of $\mathbb{Z}_2 \times \mathbb{Z}_2$.

$\square$

Flow-Coloring Duality

Theorem 7.27 (Tait 1878)

A simple 2-edge-connected 3-regular plane graph G is 3-edge-colorable if and only if its dual is 4-colorable.

Proof.

Suppose G is 4-face-colored with elements of $\mathbb{Z}_2 \times \mathbb{Z}_2$. For each edge, assign to it the color that is the sum of the colors of the two incident faces.

$\square$

Flow-Coloring Duality

Theorem 7.27 (Tait 1878)

A simple 2-edge-connected 3-regular plane graph G is 3-edge-colorable if and only if its dual is 4-colorable.

Proof.

Suppose G is 4-face-colored with elements of $\mathbb{Z}_2 \times \mathbb{Z}_2$. For each edge, assign to it the color that is the sum of the colors of the two incident faces. Then it is easy to check that this results in proper 3-edge-coloring.

□

Theorem 7.27 (Tait 1878)

A simple 2-edge-connected 3-regular plane graph G is 3-edge-colorable if and only if its dual is 4-colorable.

Proof.

Suppose G is 4-face-colored with elements of $\mathbb{Z}_2 \times \mathbb{Z}_2$. For each edge, assign to it the color that is the sum of the colors of the two incident faces. Then it is easy to check that this results in proper 3-edge-coloring.
Conversely, suppose the edges of G can be colored with colors from $\mathbb{Z}_2 \times \mathbb{Z}_2 \setminus \{(0,0)\}$.

□

Flow-Coloring Duality

Theorem 7.27 (Tait 1878)

A simple 2-edge-connected 3-regular plane graph G is 3-edge-colorable if and only if its dual is 4-colorable.

Proof.

Suppose G is 4-face-colored with elements of $\mathbb{Z}_2 \times \mathbb{Z}_2$. For each edge, assign to it the color that is the sum of the colors of the two incident faces. Then it is easy to check that this results in proper 3-edge-coloring.
Conversely, suppose the edges of G can be colored with colors from $\mathbb{Z}_2 \times \mathbb{Z}_2 \setminus \{(0,0)\}$. Let H_1 be the subgraph induced by the edges colored $(1,0)$ or $(1,1)$,

$\square$

Flow-Coloring Duality

Theorem 7.27 (Tait 1878)

A simple 2-edge-connected 3-regular plane graph G is 3-edge-colorable if and only if its dual is 4-colorable.

Proof.

Suppose G is 4-face-colored with elements of $\mathbb{Z}_2 \times \mathbb{Z}_2$. For each edge, assign to it the color that is the sum of the colors of the two incident faces. Then it is easy to check that this results in proper 3-edge-coloring.
Conversely, suppose the edges of G can be colored with colors from $\mathbb{Z}_2 \times \mathbb{Z}_2 \setminus \{(0,0)\}$. Let H_1 be the subgraph induced by the edges colored $(1,0)$ or $(1,1)$, and let H_2 be the subgraph induced by the edges colored $(0,1)$ or $(1,1)$.

$\square$

Flow-Coloring Duality

Theorem 7.27 (Tait 1878)

A simple 2-edge-connected 3-regular plane graph G is 3-edge-colorable if and only if its dual is 4-colorable.

Proof.

Suppose G is 4-face-colored with elements of $\mathbb{Z}_2 \times \mathbb{Z}_2$. For each edge, assign to it the color that is the sum of the colors of the two incident faces. Then it is easy to check that this results in proper 3-edge-coloring.
Conversely, suppose the edges of G can be colored with colors from $\mathbb{Z}_2 \times \mathbb{Z}_2 \setminus \{(0,0)\}$. Let H_1 be the subgraph induced by the edges colored $(1,0)$ or $(1,1)$, and let H_2 be the subgraph induced by the edges colored $(0,1)$ or $(1,1)$. Note that each of H_1 and H_2 is the disjoint union of cycles.

$\square$

Flow-Coloring Duality

Theorem 7.27 (Tait 1878)

A simple 2-edge-connected 3-regular plane graph G is 3-edge-colorable if and only if its dual is 4-colorable.

Proof.

Suppose G is 4-face-colored with elements of $\mathbb{Z}_2 \times \mathbb{Z}_2$. For each edge, assign to it the color that is the sum of the colors of the two incident faces. Then it is easy to check that this results in proper 3-edge-coloring.
Conversely, suppose the edges of G can be colored with colors from $\mathbb{Z}_2 \times \mathbb{Z}_2 \setminus \{(0,0)\}$. Let H_1 be the subgraph induced by the edges colored $(1,0)$ or $(1,1)$, and let H_2 be the subgraph induced by the edges colored $(0,1)$ or $(1,1)$. Note that each of H_1 and H_2 is the disjoint union of cycles. To each face of G, assign the color (p_1, p_2) where p_i is the parity (0 for even, 1 for odd) of the number of cycles that contain it inside. $\square$

Problem Set 5

Problem 13

Prove that every loopless planar graph on fewer than thirteen vertices admits a proper 4-coloring. In other words, prove the Four-Color Theorem for graphs on at most twelve vertices.

Problem 14

Show that every graph without a cut-edge admits a flow.

Problem 15

Show that if a graph has a spanning cycle, then it admits a 4-flow.

Tutte's Flow Conjectures

Conjecture 7.28 (Tutte)

- *(5-Flow Conjecture, 1954) Every graph with no cut-edge has a 5-flow.*

Tutte's Flow Conjectures

Conjecture 7.28 (Tutte)

- *(5-Flow Conjecture, 1954) Every graph with no cut-edge has a 5-flow.*
- *(4-Flow Conjecture, 1966) Every graph with no cut-edge and no Petersen graph minor has a 4-flow.*

Tutte's Flow Conjectures

Conjecture 7.28 (Tutte)

- *(5-Flow Conjecture, 1954) Every graph with no cut-edge has a 5-flow.*
- *(4-Flow Conjecture, 1966) Every graph with no cut-edge and no Petersen graph minor has a 4-flow.*
- *(3-Flow Conjecture, 1972) Every graph with no edge-cuts of size 1 and 3 has a 3-flow.*

Tutte's Flow Conjectures

Conjecture 7.28 (Tutte)

- *(5-Flow Conjecture, 1954) Every graph with no cut-edge has a 5-flow.*
- *(4-Flow Conjecture, 1966) Every graph with no cut-edge and no Petersen graph minor has a 4-flow.*
- *(3-Flow Conjecture, 1972) Every graph with no edge-cuts of size 1 and 3 has a 3-flow.*

Theorem 7.29 (Seymour 1981)

Every graph with no cut-edge has a 6-flow.

Tutte's Flow Conjectures

Conjecture 7.28 (Tutte)

- *(5-Flow Conjecture, 1954) Every graph with no cut-edge has a 5-flow.*
- *(4-Flow Conjecture, 1966) Every graph with no cut-edge and no Petersen graph minor has a 4-flow.*
- *(3-Flow Conjecture, 1972) Every graph with no edge-cuts of size 1 and 3 has a 3-flow.*

Theorem 7.29 (Seymour 1981)

Every graph with no cut-edge has a 6-flow.

Theorem 7.30 (Robertson, Sanders, Seymour, Thomas 2000)

Every cubic graph with no cut-edge and no Petersen graph minor has a 4-flow.

Tutte's Flow Conjectures

Conjecture 7.28 (Tutte)

- *(5-Flow Conjecture, 1954) Every graph with no cut-edge has a 5-flow.*
- *(4-Flow Conjecture, 1966) Every graph with no cut-edge and no Petersen graph minor has a 4-flow.*
- *(3-Flow Conjecture, 1972) Every graph with no edge-cuts of size 1 and 3 has a 3-flow.*

Theorem 7.29 (Seymour 1981)

Every graph with no cut-edge has a 6-flow.

Theorem 7.30 (Robertson, Sanders, Seymour, Thomas 2000)

Every cubic graph with no cut-edge and no Petersen graph minor has a 4-flow.

Theorem 7.31 (Grötzsch 1959)

Every planar graph with no edge-cuts of size 1 and 3 has a 3-flow.

Miscellany of Coloring Results

Theorem 7.32 (Tutte 1954)

$\chi(G) = \varphi(G^*)$

Proof.

$\square$

Miscellany of Coloring Results

Theorem 7.32 (Tutte 1954)

$\chi(G) = \varphi(G^*)$

Proof.

$\square$

Conjecture 7.33 (Hadwiger 1943)

If n is an integer exceeding 1, and G has no K_n-minor, then $\chi(G) < n$.

Miscellany of Coloring Results

Theorem 7.32 (Tutte 1954)

$\chi(G) = \varphi(G^*)$

Proof.

□

Conjecture 7.33 (Hadwiger 1943)

If n is an integer exceeding 1, and G has no K_n-minor, then $\chi(G) < n$.

Note 7.34

- *Trivial for $n = 2$.*

Miscellany of Coloring Results

Theorem 7.32 (Tutte 1954)

$\chi(G) = \varphi(G^*)$

Proof.

$\square$

Conjecture 7.33 (Hadwiger 1943)

If n is an integer exceeding 1, and G has no K_n-minor, then $\chi(G) < n$.

Note 7.34

- *Trivial for $n = 2$.*
- *Obvious for $n = 3$.*

Miscellany of Coloring Results

Theorem 7.32 (Tutte 1954)

$\chi(G) = \varphi(G^*)$

Proof.

$\square$

Conjecture 7.33 (Hadwiger 1943)

If n is an integer exceeding 1, and G has no K_n-minor, then $\chi(G) < n$.

Note 7.34

- *Trivial for $n = 2$.*
- *Obvious for $n = 3$.*
- *Easy for $n = 4$.*

Miscellany of Coloring Results

Theorem 7.32 (Tutte 1954)

$\chi(G) = \varphi(G^*)$

Proof.

$\square$

Conjecture 7.33 (Hadwiger 1943)

If n is an integer exceeding 1, and G has no K_n-minor, then $\chi(G) < n$.

Note 7.34

- *Trivial for $n = 2$.*
- *Obvious for $n = 3$.*
- *Easy for $n = 4$.*
- *Equivalent to Four-Color Theorem for $n = 5$.*

Miscellany of Coloring Results

Theorem 7.32 (Tutte 1954)

$\chi(G) = \varphi(G^*)$

Proof.

$\square$

Conjecture 7.33 (Hadwiger 1943)

If n is an integer exceeding 1, and G has no K_n-minor, then $\chi(G) < n$.

Note 7.34

- *Trivial for $n = 2$.*
- *Obvious for $n = 3$.*
- *Easy for $n = 4$.*
- *Equivalent to Four-Color Theorem for $n = 5$.*
- *Proved by Robertson, Seymour, and Thomas for $n = 6$.*

Miscellany of Coloring Results

Theorem 7.32 (Tutte 1954)

$\chi(G) = \varphi(G^*)$

Proof.

$\square$

Conjecture 7.33 (Hadwiger 1943)

If n is an integer exceeding 1, and G has no K_n-minor, then $\chi(G) < n$.

Note 7.34

- *Trivial for $n = 2$.*
- *Obvious for $n = 3$.*
- *Easy for $n = 4$.*
- *Equivalent to Four-Color Theorem for $n = 5$.*
- *Proved by Robertson, Seymour, and Thomas for $n = 6$.*
- *Unknown for $n \geq 7$.*

Hamilton Cycles

Definition 8.1

- A spanning subgraph that is a cycle or a path is called a Hamilton cycle or a Hamilton path.

Hamilton Cycles

Definition 8.1

- A spanning subgraph that is a cycle or a path is called a Hamilton cycle or a Hamilton path.
- A graph is Hamiltonian if it has a Hamilton cycle.

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described.

$\square$

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ *and* $\delta \geq n/2$ *is Hamiltonian.*

Proof.

Let G be a graph as described. Note that G is connected;

$\square$

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described. Note that G is connected; otherwise a vertex in a smallest component would have degree less than $n/2$.

$\square$

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described. Note that G is connected; otherwise a vertex in a smallest component would have degree less than $n/2$.
Let $P = x_0 x_1 \dots x_k$ be a longest path in G.

$\square$

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described. Note that G is connected; otherwise a vertex in a smallest component would have degree less than $n/2$.
Let $P = x_0 x_1 \dots x_k$ be a longest path in G. By the maximality of P, all neighbors of x_0 and all neighbors of x_k lie on P.

□

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described. Note that G is connected; otherwise a vertex in a smallest component would have degree less than $n/2$.
Let $P = x_0 x_1 \ldots x_k$ be a longest path in G. By the maximality of P, all neighbors of x_0 and all neighbors of x_k lie on P. Hence at least $n/2$ of the vertices x_0, x_1, ..., x_{k-1} are adjacent to x_k,

□

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described. Note that G is connected; otherwise a vertex in a smallest component would have degree less than $n/2$.
Let $P = x_0 x_1 \ldots x_k$ be a longest path in G. By the maximality of P, all neighbors of x_0 and all neighbors of x_k lie on P. Hence at least $n/2$ of the vertices $x_0, x_1, \ldots, x_{k-1}$ are adjacent to x_k, and at least $n/2$ of the same $k < n$ vertices x_i are such that $x_0 x_{i+1} \in E(G)$.

□

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described. Note that G is connected; otherwise a vertex in a smallest component would have degree less than $n/2$.
Let $P = x_0x_1 \ldots x_k$ be a longest path in G. By the maximality of P, all neighbors of x_0 and all neighbors of x_k lie on P. Hence at least $n/2$ of the vertices x_0, x_1, ..., x_{k-1} are adjacent to x_k, and at least $n/2$ of the same $k < n$ vertices x_i are such that $x_0x_{i+1} \in E(G)$. By the Pigeon-Hole Principle, there is a vertex x_i that has both properties,

□

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described. Note that G is connected; otherwise a vertex in a smallest component would have degree less than $n/2$.
Let $P = x_0x_1 \ldots x_k$ be a longest path in G. By the maximality of P, all neighbors of x_0 and all neighbors of x_k lie on P. Hence at least $n/2$ of the vertices x_0, x_1, ..., x_{k-1} are adjacent to x_k, and at least $n/2$ of the same $k < n$ vertices x_i are such that $x_0x_{i+1} \in E(G)$. By the Pigeon-Hole Principle, there is a vertex x_i that has both properties, that is, $x_0x_{i+1} \in E(G)$ and $x_ix_k \in E(G)$.

$\square$

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described. Note that G is connected; otherwise a vertex in a smallest component would have degree less than $n/2$.
Let $P = x_0 x_1 \dots x_k$ be a longest path in G. By the maximality of P, all neighbors of x_0 and all neighbors of x_k lie on P. Hence at least $n/2$ of the vertices $x_0, x_1, \dots, x_{k-1}$ are adjacent to x_k, and at least $n/2$ of the same $k < n$ vertices x_i are such that $x_0 x_{i+1} \in E(G)$. By the Pigeon-Hole Principle, there is a vertex x_i that has both properties, that is, $x_0 x_{i+1} \in E(G)$ and $x_i x_k \in E(G)$. Let C be the cycle obtained from P by deleting the edge $x_i x_{i+1}$ and adding edges $x_0 x_{i+1}$ and $x_i x_k$.

$\square$

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described. Note that G is connected; otherwise a vertex in a smallest component would have degree less than $n/2$.
Let $P = x_0 x_1 \ldots x_k$ be a longest path in G. By the maximality of P, all neighbors of x_0 and all neighbors of x_k lie on P. Hence at least $n/2$ of the vertices x_0, x_1, $\ldots$, x_{k-1} are adjacent to x_k, and at least $n/2$ of the same $k < n$ vertices x_i are such that $x_0 x_{i+1} \in E(G)$. By the Pigeon-Hole Principle, there is a vertex x_i that has both properties, that is, $x_0 x_{i+1} \in E(G)$ and $x_i x_k \in E(G)$. Let C be the cycle obtained from P by deleting the edge $x_i x_{i+1}$ and adding edges $x_0 x_{i+1}$ and $x_i x_k$. If C is not Hamilton, then, since G is connected, C would have a neighbor in $G - C$,

$\square$

Sufficient Condition for Hamiltonicity

Theorem 8.2 (Dirac 1952)

Every graph of order $n \geq 3$ and $\delta \geq n/2$ is Hamiltonian.

Proof.

Let G be a graph as described. Note that G is connected; otherwise a vertex in a smallest component would have degree less than $n/2$.
Let $P = x_0 x_1 \ldots x_k$ be a longest path in G. By the maximality of P, all neighbors of x_0 and all neighbors of x_k lie on P. Hence at least $n/2$ of the vertices x_0, x_1, ..., x_{k-1} are adjacent to x_k, and at least $n/2$ of the same $k < n$ vertices x_i are such that $x_0 x_{i+1} \in E(G)$. By the Pigeon-Hole Principle, there is a vertex x_i that has both properties, that is, $x_0 x_{i+1} \in E(G)$ and $x_i x_k \in E(G)$. Let C be the cycle obtained from P by deleting the edge $x_i x_{i+1}$ and adding edges $x_0 x_{i+1}$ and $x_i x_k$. If C is not Hamilton, then, since G is connected, C would have a neighbor in $G - C$, which would yield a path longer than P; a contradiction. □

Note on Dirac's Theorem

Note 8.3

Note that $n/2$ in Dirac's Theorem 8.2 is the best possible.

Note on Dirac's Theorem

Note 8.3

Note that $n/2$ *in Dirac's Theorem 8.2 is the best possible. We cannot replace it with* $\lfloor n/2 \rfloor$ *if* n *is odd, since then* G *which is a* 1*-sum of two copies of* $K^{\lceil n/2 \rceil}$

Note on Dirac's Theorem

Note 8.3

Note that $n/2$ in Dirac's Theorem 8.2 is the best possible. We cannot replace it with $\lfloor n/2 \rfloor$ if n is odd, since then G which is a 1-sum of two copies of $K^{\lceil n/2 \rceil}$ would have $\delta = \lfloor n/2 \rfloor$, but no Hamilton cycle.

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G.

$\square$

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$.

$\square$

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$. If C is not a Hamiltonian cycle, pick a vertex v not in C.

□

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$. If C is not a Hamiltonian cycle, pick a vertex v not in C. Let $\mathcal{F} = \{P_i : i \in I\}$ be a maximum-cardinality collection of vC-paths that pairwise meet only in v and so that P_i contains v_i.

□

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$. If C is not a Hamiltonian cycle, pick a vertex v not in C. Let $\mathcal{F} = \{P_i : i \in I\}$ be a maximum-cardinality collection of vC-paths that pairwise meet only in v and so that P_i contains v_i. Then $vv_j \notin E(G)$ for every $j \notin I$,

$\square$

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$. If C is not a Hamiltonian cycle, pick a vertex v not in C. Let $\mathcal{F} = \{P_i : i \in I\}$ be a maximum-cardinality collection of vC-paths that pairwise meet only in v and so that P_i contains v_i. Then $vv_j \notin E(G)$ for every $j \notin I$, and $|I| \geq \min\{k, |C|\}$ by Menger's Theorem 5.29.

□

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$. If C is not a Hamiltonian cycle, pick a vertex v not in C. Let $\mathcal{F} = \{P_i : i \in I\}$ be a maximum-cardinality collection of vC-paths that pairwise meet only in v and so that P_i contains v_i. Then $vv_j \notin E(G)$ for every $j \notin I$, and $|I| \geq \min\{k, |C|\}$ by Menger's Theorem 5.29. For every $i \in I$, we have $i + 1 \notin I$,

$\square$

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$. If C is not a Hamiltonian cycle, pick a vertex v not in C. Let $\mathcal{F} = \{P_i : i \in I\}$ be a maximum-cardinality collection of vC-paths that pairwise meet only in v and so that P_i contains v_i. Then $vv_j \notin E(G)$ for every $j \notin I$, and $|I| \geq \min\{k, |C|\}$ by Menger's Theorem 5.29. For every $i \in I$, we have $i + 1 \notin I$, otherwise $(C \cup P_i \cup P_{i+1}) \setminus v_i v_{i+1}$ would be a cycle longer than C.

□

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$. If C is not a Hamiltonian cycle, pick a vertex v not in C. Let $\mathcal{F} = \{P_i : i \in I\}$ be a maximum-cardinality collection of vC-paths that pairwise meet only in v and so that P_i contains v_i. Then $vv_j \notin E(G)$ for every $j \notin I$, and $|I| \geq \min\{k, |C|\}$ by Menger's Theorem 5.29. For every $i \in I$, we have $i + 1 \notin I$, otherwise $(C \cup P_i \cup P_{i+1}) \setminus v_i v_{i+1}$ would be a cycle longer than C. Thus $|I| < |C|$ and hence $|I| = |\mathcal{F}| \geq k$.

□

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$. If C is not a Hamiltonian cycle, pick a vertex v not in C. Let $\mathcal{F} = \{P_i : i \in I\}$ be a maximum-cardinality collection of vC-paths that pairwise meet only in v and so that P_i contains v_i. Then $vv_j \notin E(G)$ for every $j \notin I$, and $|I| \geq \min\{k, |C|\}$ by Menger's Theorem 5.29. For every $i \in I$, we have $i + 1 \notin I$, otherwise $(C \cup P_i \cup P_{i+1}) \setminus v_i v_{i+1}$ would be a cycle longer than C. Thus $|I| < |C|$ and hence $|I| = |\mathcal{F}| \geq k$. Furthermore, $v_{i+1} v_{j+1} \notin E(G)$ for all $i, j \in I$,

□

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$. If C is not a Hamiltonian cycle, pick a vertex v not in C. Let $\mathcal{F} = \{P_i : i \in I\}$ be a maximum-cardinality collection of vC-paths that pairwise meet only in v and so that P_i contains v_i. Then $vv_j \notin E(G)$ for every $j \notin I$, and $|I| \geq \min\{k, |C|\}$ by Menger's Theorem 5.29. For every $i \in I$, we have $i+1 \notin I$, otherwise $(C \cup P_i \cup P_{i+1}) \setminus v_i v_{i+1}$ would be a cycle longer than C. Thus $|I| < |C|$ and hence $|I| = |\mathcal{F}| \geq k$. Furthermore, $v_{i+1} v_{j+1} \notin E(G)$ for all $i, j \in I$, as otherwise $(C \cup P_i \cup P_j \cup v_{i+1} v_{j+1}) \setminus v_i v_{i+1} \setminus v_j v_{j+1}$ would be a cycle longer than C.

$\square$

Another Sufficient Condition

Theorem 8.4

Every graph G with $|G| \geq 3$ and $\kappa(G) \geq \alpha(G)$ is Hamiltonian.

Proof.

Let $k = \kappa(G)$ and let C be a longest cycle in G. Enumerate the vertices of C cyclically so that $V(C) = \{v_i : i \in \mathbb{Z}_n\}$ with $v_i v_{i+1} \in E(C)$ for all $i \in \mathbb{Z}_n$. If C is not a Hamiltonian cycle, pick a vertex v not in C. Let $\mathcal{F} = \{P_i : i \in I\}$ be a maximum-cardinality collection of vC-paths that pairwise meet only in v and so that P_i contains v_i. Then $vv_j \notin E(G)$ for every $j \notin I$, and $|I| \geq \min\{k, |C|\}$ by Menger's Theorem 5.29. For every $i \in I$, we have $i + 1 \notin I$, otherwise $(C \cup P_i \cup P_{i+1}) \setminus v_i v_{i+1}$ would be a cycle longer than C. Thus $|I| < |C|$ and hence $|I| = |\mathcal{F}| \geq k$. Furthermore, $v_{i+1} v_{j+1} \notin E(G)$ for all $i, j \in I$, as otherwise $(C \cup P_i \cup P_j \cup v_{i+1} v_{j+1}) \setminus v_i v_{i+1} \setminus v_j v_{j+1}$ would be a cycle longer than C. Hence $\{v_{i+1} : i \in I\} \cup \{v\}$ is a set of at least $k + 1$ independent vertices in G, contradicting the assumption that $\alpha(G) \leq k$. $\square$

A Necessary Condition

Theorem 8.5

If G is a Hamiltonian graph, then for every set $\emptyset \neq S \subseteq V(G)$, the graph $G - S$ has at most S components.

A Necessary Condition

Theorem 8.5

If G is a Hamiltonian graph, then for every set $\emptyset \neq S \subseteq V(G)$, the graph $G - S$ has at most S components.

Proof.

When leaving a component of $G - S$, a Hamilton cycle can go only to S and the arrivals in S must occur at different vertices of S. $\square$

A Necessary Condition

Theorem 8.5

If G is a Hamiltonian graph, then for every set $\emptyset \neq S \subseteq V(G)$, the graph $G - S$ has at most S components.

Proof.

When leaving a component of $G - S$, a Hamilton cycle can go only to S and the arrivals in S must occur at different vertices of S. Hence S must have at least as many vertices as $G - S$ has components. $\square$

Note that if we managed to prove that every 3-connected cubic plane graph is Hamiltonian, Note that if we managed to prove that every 3-connected cubic plane graph is Hamiltonian, then we woud have proved that every such graph has a 4-flow, Note that if we managed to prove that every 3-connected cubic plane graph is Hamiltonian, then we woud have proved that every such graph has a 4-flow, and so is 3-edge-colorable, Note that if we managed to prove that every 3-connected cubic plane graph is Hamiltonian, then we woud have proved that every such graph has a 4-flow, and so is 3-edge-colorable, and so is 4-face-colorable.

Note that if we managed to prove that every 3-connected cubic plane graph is Hamiltonian, then we woud have proved that every such graph has a 4-flow, and so is 3-edge-colorable, and so is 4-face-colorable. Unfortunately, there are 3-connected cubic plane graphs that are not Hamiltonian.

Grinberg's Theorem

Theorem 8.6 (Grinberg 1968)

If G is a loopless plane graph with a Hamilton cycle C, and G has f_i' faces of length i inside C and f_i'' faces of length i outside C, then $\sum_i (i-2)(f_i' - f_i'') = 0$.

Grinberg's Theorem

Theorem 8.6 (Grinberg 1968)

If G is a loopless plane graph with a Hamilton cycle C, and G has f_i' faces of length i inside C and f_i'' faces of length i outside C, then $\sum_i (i-2)(f_i' - f_i'') = 0$.

Proof.

Want to show that $\sum_i (i-2) f_i' = \sum_i (i-2) f_i''$.

□

Theorem 8.6 (Grinberg 1968)

If G is a loopless plane graph with a Hamilton cycle C, and G has f_i' faces of length i inside C and f_i'' faces of length i outside C, then $\sum_i (i-2)(f_i' - f_i'') = 0$.

Proof.

Want to show that $\sum_i (i-2) f_i' = \sum_i (i-2) f_i''$. It suffices to show that $\sum_i (i-2) f_i'$ remains invariant as we add edges inside a cycle C of length n.

□

Theorem 8.6 (Grinberg 1968)

If G is a loopless plane graph with a Hamilton cycle C, and G has f_i' faces of length i inside C and f_i'' faces of length i outside C, then $\sum_i (i-2)(f_i' - f_i'') = 0$.

Proof.

Want to show that $\sum_i (i-2) f_i' = \sum_i (i-2) f_i''$. It suffices to show that $\sum_i (i-2) f_i'$ remains invariant as we add edges inside a cycle C of length n. If there are no edges inside C, then the sum is $n-2$.

$\square$

Grinberg's Theorem

Theorem 8.6 (Grinberg 1968)

If G is a loopless plane graph with a Hamilton cycle C, and G has f_i' faces of length i inside C and f_i'' faces of length i outside C, then $\sum_i (i-2)(f_i' - f_i'') = 0$.

Proof.

Want to show that $\sum_i (i-2) f_i' = \sum_i (i-2) f_i''$. It suffices to show that $\sum_i (i-2) f_i'$ remains invariant as we add edges inside a cycle C of length n. If there are no edges inside C, then the sum is $n-2$. Suppose $\sum_i (i-2) f_i' = n-2$ for any graph with k edges inside C.

$\square$

Theorem 8.6 (Grinberg 1968)

If G is a loopless plane graph with a Hamilton cycle C, and G has f_i' faces of length i inside C and f_i'' faces of length i outside C, then $\sum_i (i-2)(f_i' - f_i'') = 0$.

Proof.

Want to show that $\sum_i (i-2)f_i' = \sum_i (i-2)f_i''$. It suffices to show that $\sum_i (i-2)f_i'$ remains invariant as we add edges inside a cycle C of length n. If there are no edges inside C, then the sum is $n-2$. Suppose $\sum_i (i-2)f_i' = n-2$ for any graph with k edges inside C. We can obtain any graph with $k+1$ edges inside C by adding an edge to such graph. $\square$

Theorem 8.6 (Grinberg 1968)

If G is a loopless plane graph with a Hamilton cycle C, and G has f_i' faces of length i inside C and f_i'' faces of length i outside C, then $\sum_i (i-2)(f_i' - f_i'') = 0$.

Proof.

Want to show that $\sum_i (i-2) f_i' = \sum_i (i-2) f_i''$. It suffices to show that $\sum_i (i-2) f_i'$ remains invariant as we add edges inside a cycle C of length n. If there are no edges inside C, then the sum is $n-2$. Suppose $\sum_i (i-2) f_i' = n-2$ for any graph with k edges inside C. We can obtain any graph with $k+1$ edges inside C by adding an edge to such graph. The edge addition cuts a face of length r into faces of lengths s and t.

$\square$

Grinberg's Theorem

Theorem 8.6 (Grinberg 1968)

If G is a loopless plane graph with a Hamilton cycle C, and G has f_i' faces of length i inside C and f_i'' faces of length i outside C, then $\sum_i (i-2)(f_i' - f_i'') = 0$.

Proof.

Want to show that $\sum_i (i-2) f_i' = \sum_i (i-2) f_i''$. It suffices to show that $\sum_i (i-2) f_i'$ remains invariant as we add edges inside a cycle C of length n. If there are no edges inside C, then the sum is $n-2$. Suppose $\sum_i (i-2) f_i' = n-2$ for any graph with k edges inside C. We can obtain any graph with $k+1$ edges inside C by adding an edge to such graph. The edge addition cuts a face of length r into faces of lengths s and t. We have $s+t = r+2$, and so $(s-2)+(t-2) = r-2$ and so the total contribution remains the same. $\square$

Corollary 8.7

The Tutte graph is not Hamiltonian.

Corollary 8.7

The Tutte graph is not Hamiltonian.

Theorem 8.8 (Tutte 1956)

Every 4*-connected planar graph is Hamiltonian.*

Corollary 8.7

The Tutte graph is not Hamiltonian.

Theorem 8.8 (Tutte 1956)

Every 4*-connected planar graph is Hamiltonian.*

Theorem 8.9 (Thomas, Yu 1994)

Every 4*-connected projective graph is Hamiltonian.*

Corollary 8.7

The Tutte graph is not Hamiltonian.

Theorem 8.8 (Tutte 1956)

Every 4-connected planar graph is Hamiltonian.

Theorem 8.9 (Thomas, Yu 1994)

Every 4-connected projective graph is Hamiltonian.

Theorem 8.10 (Thomas, Yu 1997)

Every 5-connected toroidal graph is Hamiltonian.

Suppose we have a finite family $\mathcal{F}$ of pairwise disjoint convex polygons (with interiors) in the plane with all sides of length 1.

Suppose we have a finite family $\mathcal{F}$ of pairwise disjoint convex polygons (with interiors) in the plane with all sides of length 1. Suppose that all these polygons together have m sides σ_1, σ_2, $\ldots$, σ_m where m is even.

Suppose we have a finite family $\mathcal{F}$ of pairwise disjoint convex polygons (with interiors) in the plane with all sides of length 1. Suppose that all these polygons together have m sides σ_1, σ_2, $\dots$, σ_m where m is even. Orient arbitrarily each of the sides σ_i by choosing one of its endpoints as the initial point, Suppose we have a finite family $\mathcal{F}$ of pairwise disjoint convex polygons (with interiors) in the plane with all sides of length 1. Suppose that all these polygons together have m sides σ_1, σ_2, ..., σ_m where m is even. Orient arbitrarily each of the sides σ_i by choosing one of its endpoints as the initial point, and choose an arbitrary partition of the sides into pairs.

Suppose we have a finite family $\mathcal{F}$ of pairwise disjoint convex polygons (with interiors) in the plane with all sides of length 1. Suppose that all these polygons together have m sides σ_1, σ_2, $\ldots$, σ_m where m is even. Orient arbitrarily each of the sides σ_i by choosing one of its endpoints as the initial point, and choose an arbitrary partition of the sides into pairs. From the disjoint union of polygons $\mathcal{F}$, form a topological space S by identifying sides in each pair of our partition so that the orientations agree.

Suppose we have a finite family $\mathcal{F}$ of pairwise disjoint convex polygons (with interiors) in the plane with all sides of length 1. Suppose that all these polygons together have m sides σ_1, σ_2, $\dots$, σ_m where m is even. Orient arbitrarily each of the sides σ_i by choosing one of its endpoints as the initial point, and choose an arbitrary partition of the sides into pairs. From the disjoint union of polygons $\mathcal{F}$, form a topological space S by identifying sides in each pair of our partition so that the orientations agree. If S is connected (which will be assumed from now on), we get a surface S, that is, a compact Hausdorff space that is locally homeomorphic to the unit disk in the plane.

Suppose we have a finite family $\mathcal{F}$ of pairwise disjoint convex polygons (with interiors) in the plane with all sides of length 1. Suppose that all these polygons together have m sides $\sigma_1, \sigma_2, \ldots, \sigma_m$ where m is even. Orient arbitrarily each of the sides σ_i by choosing one of its endpoints as the initial point, and choose an arbitrary partition of the sides into pairs. From the disjoint union of polygons $\mathcal{F}$, form a topological space S by identifying sides in each pair of our partition so that the orientations agree. If S is connected (which will be assumed from now on), we get a surface S, that is, a compact Hausdorff space that is locally homeomorphic to the unit disk in the plane. The identified edges of the polygons can be viewed as edges of a graph, the endpoints of those edges as vertices, and the interiors of the polygons are faces.

Suppose we have a finite family $\mathcal{F}$ of pairwise disjoint convex polygons (with interiors) in the plane with all sides of length 1. Suppose that all these polygons together have m sides $\sigma_1, \sigma_2, \ldots, \sigma_m$ where m is even. Orient arbitrarily each of the sides σ_i by choosing one of its endpoints as the initial point, and choose an arbitrary partition of the sides into pairs. From the disjoint union of polygons $\mathcal{F}$, form a topological space S by identifying sides in each pair of our partition so that the orientations agree. If S is connected (which will be assumed from now on), we get a surface S, that is, a compact Hausdorff space that is locally homeomorphic to the unit disk in the plane. The identified edges of the polygons can be viewed as edges of a graph, the endpoints of those edges as vertices, and the interiors of the polygons are faces. What results is an embedding of a connected graph into some surface, which will be called 2-cell embedding.

Suppose we have a finite family $\mathcal{F}$ of pairwise disjoint convex polygons (with interiors) in the plane with all sides of length 1. Suppose that all these polygons together have m sides $\sigma_1, \sigma_2, \ldots, \sigma_m$ where m is even. Orient arbitrarily each of the sides σ_i by choosing one of its endpoints as the initial point, and choose an arbitrary partition of the sides into pairs. From the disjoint union of polygons $\mathcal{F}$, form a topological space S by identifying sides in each pair of our partition so that the orientations agree. If S is connected (which will be assumed from now on), we get a surface S, that is, a compact Hausdorff space that is locally homeomorphic to the unit disk in the plane. The identified edges of the polygons can be viewed as edges of a graph, the endpoints of those edges as vertices, and the interiors of the polygons are faces. What results is an embedding of a connected graph into some surface, which will be called 2-cell embedding.

Theorem 9.1

Every surface is homeomorphic to a triangulated surface.

Proof omitted.

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G.

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree.

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree. We say that the surface S' is obtained from S by adding a handle.

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree. We say that the surface S' is obtained from S by adding a handle. If we identify T_1 and T_2 so that their orientations agree to obtain a surface S'', then S'' is obtained from S by adding a twisted handle.

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree. We say that the surface S' is obtained from S by adding a handle. If we identify T_1 and T_2 so that their orientations agree to obtain a surface S'', then S'' is obtained from S by adding a twisted handle. Finally, let T be a square in F.

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree. We say that the surface S' is obtained from S by adding a handle. If we identify T_1 and T_2 so that their orientations agree to obtain a surface S'', then S'' is obtained from S by adding a twisted handle. Finally, let T be a square in F. Let S''' be obtained from S by deleting the interior of T and identifying the diametrically opposite points of the square.

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree. We say that the surface S' is obtained from S by adding a handle. If we identify T_1 and T_2 so that their orientations agree to obtain a surface S'', then S'' is obtained from S by adding a twisted handle. Finally, let T be a square in F. Let S''' be obtained from S by deleting the interior of T and identifying the diametrically opposite points of the square. Then S''' is obtained from S by adding a crosscap.

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree. We say that the surface S' is obtained from S by adding a handle. If we identify T_1 and T_2 so that their orientations agree to obtain a surface S'', then S'' is obtained from S by adding a twisted handle. Finally, let T be a square in F. Let S''' be obtained from S by deleting the interior of T and identifying the diametrically opposite points of the square. Then S''' is obtained from S by adding a crosscap. It is easy to show that S', S'', and S''' are independent (up to homeomorphism) of where in F the triangles T_1 and T_2 and the square T come from,

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree. We say that the surface S' is obtained from S by adding a handle. If we identify T_1 and T_2 so that their orientations agree to obtain a surface S'', then S'' is obtained from S by adding a twisted handle. Finally, let T be a square in F. Let S''' be obtained from S by deleting the interior of T and identifying the diametrically opposite points of the square. Then S''' is obtained from S by adding a crosscap. It is easy to show that S', S'', and S''' are independent (up to homeomorphism) of where in F the triangles T_1 and T_2 and the square T come from, or whether they come from distinct faces.

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree. We say that the surface S' is obtained from S by adding a handle. If we identify T_1 and T_2 so that their orientations agree to obtain a surface S'', then S'' is obtained from S by adding a twisted handle. Finally, let T be a square in F. Let S''' be obtained from S by deleting the interior of T and identifying the diametrically opposite points of the square. Then S''' is obtained from S by adding a crosscap. It is easy to show that S', S'', and S''' are independent (up to homeomorphism) of where in F the triangles T_1 and T_2 and the square T come from, or whether they come from distinct faces.
Consider the sphere $\mathbb{S}_0$ (tetrahedron).

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree. We say that the surface S' is obtained from S by adding a handle. If we identify T_1 and T_2 so that their orientations agree to obtain a surface S'', then S'' is obtained from S by adding a twisted handle. Finally, let T be a square in F. Let S''' be obtained from S by deleting the interior of T and identifying the diametrically opposite points of the square. Then S''' is obtained from S by adding a crosscap. It is easy to show that S', S'', and S''' are independent (up to homeomorphism) of where in F the triangles T_1 and T_2 and the square T come from, or whether they come from distinct faces.
Consider the sphere $\mathbb{S}_0$ (tetrahedron). If we add h handles to $\mathbb{S}_0$, then we obtain $\mathbb{S}_h$, which we call the orientable surface of genus h.

Consider now two disjoint triangles T_1 and T_2 (such that all sides have same length) in a face F of a surface S with a 2-cell embedded graph G. We form a new surface S' by deleting from F the interiors of T_1 and T_2, and identifying T_1 with T_2 such that their clockwise orientations (as defined by F) disagree. We say that the surface S' is obtained from S by adding a handle. If we identify T_1 and T_2 so that their orientations agree to obtain a surface S'', then S'' is obtained from S by adding a twisted handle. Finally, let T be a square in F. Let S''' be obtained from S by deleting the interior of T and identifying the diametrically opposite points of the square. Then S''' is obtained from S by adding a crosscap. It is easy to show that S', S'', and S''' are independent (up to homeomorphism) of where in F the triangles T_1 and T_2 and the square T come from, or whether they come from distinct faces.
Consider the sphere $\mathbb{S}_0$ (tetrahedron). If we add h handles to $\mathbb{S}_0$, then we obtain $\mathbb{S}_h$, which we call the orientable surface of genus h. If we add h crosscaps to $\mathbb{S}_0$, we obtain $\mathbb{N}_h$, the non-orientable surface of genus h.

The surfaces $\mathbb{S}_1$, $\mathbb{S}_2$, $\mathbb{N}_1$, and $\mathbb{N}_2$ are called, respectively

The surfaces $\mathbb{S}_1$, $\mathbb{S}_2$, $\mathbb{N}_1$, and $\mathbb{N}_2$ are called, respectively torus, The surfaces $\mathbb{S}_1$, $\mathbb{S}_2$, $\mathbb{N}_1$, and $\mathbb{N}_2$ are called, respectively torus, double torus, The surfaces $\mathbb{S}_1$, $\mathbb{S}_2$, $\mathbb{N}_1$, and $\mathbb{N}_2$ are called, respectively torus, double torus, projective plane, and

The surfaces $\mathbb{S}_1$, $\mathbb{S}_2$, $\mathbb{N}_1$, and $\mathbb{N}_2$ are called, respectively torus, double torus, projective plane, and Klein bottle.

The surfaces $\mathbb{S}_1$, $\mathbb{S}_2$, $\mathbb{N}_1$, and $\mathbb{N}_2$ are called, respectively torus, double torus, projective plane, and Klein bottle.
A twisted handle can be always replaced with two crosscaps, and, as long as there is a crosscap, a handle can also be replaced by two crosscaps.

The surfaces $\mathbb{S}_1$, $\mathbb{S}_2$, $\mathbb{N}_1$, and $\mathbb{N}_2$ are called, respectively torus, double torus, projective plane, and Klein bottle.
A twisted handle can be always replaced with two crosscaps, and, as long as there is a crosscap, a handle can also be replaced by two crosscaps.

Theorem 9.2

Let S be the surface obtained from the sphere by adding h handles, t twisted handles, and c crosscaps. If $t = c = 0$, then $S = \mathbb{S}_h$. Otherwise, $S = \mathbb{N}_{2h+2t+c}$.

The surfaces $\mathbb{S}_1$, $\mathbb{S}_2$, $\mathbb{N}_1$, and $\mathbb{N}_2$ are called, respectively torus, double torus, projective plane, and Klein bottle.
A twisted handle can be always replaced with two crosscaps, and, as long as there is a crosscap, a handle can also be replaced by two crosscaps.

Theorem 9.2

Let S be the surface obtained from the sphere by adding h handles, t twisted handles, and c crosscaps. If $t = c = 0$, then $S = \mathbb{S}_h$. Otherwise, $S = \mathbb{N}_{2h+2t+c}$.

Theorem 9.3

Let S be a surface and let G be a graph that is 2-cell embedded in S with v vertices, e edges and f faces.

The surfaces $\mathbb{S}_1$, $\mathbb{S}_2$, $\mathbb{N}_1$, and $\mathbb{N}_2$ are called, respectively torus, double torus, projective plane, and Klein bottle.
A twisted handle can be always replaced with two crosscaps, and, as long as there is a crosscap, a handle can also be replaced by two crosscaps.

Theorem 9.2

Let S be the surface obtained from the sphere by adding h handles, t twisted handles, and c crosscaps. If $t = c = 0$, then $S = \mathbb{S}_h$. Otherwise, $S = \mathbb{N}_{2h+2t+c}$.

Theorem 9.3

Let S be a surface and let G be a graph that is 2-cell embedded in S with v vertices, e edges and f faces. Then S is homeomorphic to either $\mathbb{S}_h$ or $\mathbb{N}_k$, where $v - e + f = 2 - 2h = 2 - k$.

Euler's Characteristic

Definition 9.4

The Euler characteristic $\chi(S)$ of a surface S is defined as

$$\chi(S) = \begin{cases} 2 - 2h, & \text{if } S = \mathbb{S}_h; \\ 2 - k, & \text{if } S = \mathbb{N}_k. \end{cases}$$

π-Walks

Suppose G is a connected non-trivial graph.

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v.

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v. Let's consider a closed walk $W = v_1 e_1 v_2 e_2 v_3 \dots v_k e_k v_1$,

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v. Let's consider a closed walk $W = v_1 e_1 v_2 e_2 v_3 \ldots v_k e_k v_1$, which is determined by the first edge $e_1 = v_1 v_2$ and the requirement that for each i we have $\pi_{v_i}(e_i) = e_{i+1}$ where $e_{k+1} = e_1$ and k is minimal with this property.

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v. Let's consider a closed walk $W = v_1e_1v_2e_2v_3 \dots v_ke_kv_1$, which is determined by the first edge $e_1 = v_1v_2$ and the requirement that for each i we have $\pi_{v_i}(e_i) = e_{i+1}$ where $e_{k+1} = e_1$ and k is minimal with this property. Note, however, that some edges might occur in W twice, traversed in opposite directions.

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v. Let's consider a closed walk $W = v_1 e_1 v_2 e_2 v_3 \dots v_k e_k v_1$, which is determined by the first edge $e_1 = v_1 v_2$ and the requirement that for each i we have $\pi_{v_i}(e_i) = e_{i+1}$ where $e_{k+1} = e_1$ and k is minimal with this property. Note, however, that some edges might occur in W twice, traversed in opposite directions. We will not distinguish W from its cyclic shifts.

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v. Let's consider a closed walk $W = v_1 e_1 v_2 e_2 v_3 \ldots v_k e_k v_1$, which is determined by the first edge $e_1 = v_1 v_2$ and the requirement that for each i we have $\pi_{v_i}(e_i) = e_{i+1}$ where $e_{k+1} = e_1$ and k is minimal with this property. Note, however, that some edges might occur in W twice, traversed in opposite directions. We will not distinguish W from its cyclic shifts. If $\pi = \{\pi_v : v \in V(G)\}$ (the rotation system), then W is a π-walk.

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v. Let's consider a closed walk $W = v_1 e_1 v_2 e_2 v_3 \ldots v_k e_k v_1$, which is determined by the first edge $e_1 = v_1 v_2$ and the requirement that for each i we have $\pi_{v_i}(e_i) = e_{i+1}$ where $e_{k+1} = e_1$ and k is minimal with this property. Note, however, that some edges might occur in W twice, traversed in opposite directions. We will not distinguish W from its cyclic shifts. If $\pi = \{\pi_v : v \in V(G)\}$ (the rotation system), then W is a π-walk. For each π-walk, take a polygon with as many sides as the length of the walk, disjoint from other polygons—call it a π-polygon.

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v. Let's consider a closed walk $W = v_1 e_1 v_2 e_2 v_3 \ldots v_k e_k v_1$, which is determined by the first edge $e_1 = v_1 v_2$ and the requirement that for each i we have $\pi_{v_i}(e_i) = e_{i+1}$ where $e_{k+1} = e_1$ and k is minimal with this property. Note, however, that some edges might occur in W twice, traversed in opposite directions. We will not distinguish W from its cyclic shifts. If $\pi = \{\pi_v : v \in V(G)\}$ (the rotation system), then W is a π-walk. For each π-walk, take a polygon with as many sides as the length of the walk, disjoint from other polygons—call it a π-polygon. Now take all π-polygons.

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v. Let's consider a closed walk $W = v_1 e_1 v_2 e_2 v_3 \ldots v_k e_k v_1$, which is determined by the first edge $e_1 = v_1 v_2$ and the requirement that for each i we have $\pi_{v_i}(e_i) = e_{i+1}$ where $e_{k+1} = e_1$ and k is minimal with this property. Note, however, that some edges might occur in W twice, traversed in opposite directions. We will not distinguish W from its cyclic shifts. If $\pi = \{\pi_v : v \in V(G)\}$ (the rotation system), then W is a π-walk. For each π-walk, take a polygon with as many sides as the length of the walk, disjoint from other polygons—call it a π-polygon. Now take all π-polygons. Each edge appears exactly twice in the π-walks, and this determines their orientation.

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v. Let's consider a closed walk $W = v_1 e_1 v_2 e_2 v_3 \ldots v_k e_k v_1$, which is determined by the first edge $e_1 = v_1 v_2$ and the requirement that for each i we have $\pi_{v_i}(e_i) = e_{i+1}$ where $e_{k+1} = e_1$ and k is minimal with this property. Note, however, that some edges might occur in W twice, traversed in opposite directions. We will not distinguish W from its cyclic shifts. If $\pi = \{\pi_v : v \in V(G)\}$ (the rotation system), then W is a π-walk. For each π-walk, take a polygon with as many sides as the length of the walk, disjoint from other polygons—call it a π-polygon. Now take all π-polygons. Each edge appears exactly twice in the π-walks, and this determines their orientation. By identifying each side with its mate we obtain a 2-cell embedding of graph isomorphic to G in some orientable surface.

π-Walks

Suppose G is a connected non-trivial graph. Suppose that for each $v \in V(G)$ we have a cyclic permutation π_v of edges incident with v. Let's consider a closed walk $W = v_1 e_1 v_2 e_2 v_3 \dots v_k e_k v_1$, which is determined by the first edge $e_1 = v_1 v_2$ and the requirement that for each i we have $\pi_{v_i}(e_i) = e_{i+1}$ where $e_{k+1} = e_1$ and k is minimal with this property. Note, however, that some edges might occur in W twice, traversed in opposite directions. We will not distinguish W from its cyclic shifts. If $\pi = \{\pi_v : v \in V(G)\}$ (the rotation system), then W is a π-walk. For each π-walk, take a polygon with as many sides as the length of the walk, disjoint from other polygons—call it a π-polygon. Now take all π-polygons. Each edge appears exactly twice in the π-walks, and this determines their orientation. By identifying each side with its mate we obtain a 2-cell embedding of graph isomorphic to G in some orientable surface.

Theorem 9.5

Every cellular embedding (an embedding where each face is homeomorphic to an open disk) into an orientable surface is determined by its rotation system.

Embedding Schemes

An embedding scheme is a pair $\Pi = (\pi, \lambda)$ where π is a rotation system, and $\lambda : E(G) \to \{-1, 1\}$ is a signature.

Embedding Schemes

An embedding scheme is a pair $\Pi = (\pi, \lambda)$ where π is a rotation system, and $\lambda : E(G) \to \{-1, 1\}$ is a signature. Two embedding schemes are equivalent if one can be obtained from the other by a sequence of operations, each involving a change of clockwise to counter-clockwise orientation at a vertex v and the corresponding change of signatures of edges incident with v.

Embedding Schemes

An embedding scheme is a pair $\Pi = (\pi, \lambda)$ where π is a rotation system, and $\lambda : E(G) \to \{-1, 1\}$ is a signature. Two embedding schemes are equivalent if one can be obtained from the other by a sequence of operations, each involving a change of clockwise to counter-clockwise orientation at a vertex v and the corresponding change of signatures of edges incident with v. Now we define a surface embedding using π-walks and π-polygons as before, with the following modification.

Embedding Schemes

An embedding scheme is a pair $\Pi = (\pi, \lambda)$ where π is a rotation system, and $\lambda : E(G) \to \{-1, 1\}$ is a signature. Two embedding schemes are equivalent if one can be obtained from the other by a sequence of operations, each involving a change of clockwise to counter-clockwise orientation at a vertex v and the corresponding change of signatures of edges incident with v. Now we define a surface embedding using π-walks and π-polygons as before, with the following modification. Whenever we traverse an edge of negative signature, we switch the permutation at a vertex from $\pi(v)$ to $\pi^{-1}(v)$.

Embedding Schemes

An embedding scheme is a pair $\Pi = (\pi, \lambda)$ where π is a rotation system, and $\lambda : E(G) \rightarrow \{-1, 1\}$ is a signature. Two embedding schemes are equivalent if one can be obtained from the other by a sequence of operations, each involving a change of clockwise to counter-clockwise orientation at a vertex v and the corresponding change of signatures of edges incident with v. Now we define a surface embedding using π-walks and π-polygons as before, with the following modification. Whenever we traverse an edge of negative signature, we switch the permutation at a vertex from $\pi(v)$ to $\pi^{-1}(v)$. Traversing a π-walk stops whenever we are about to traverse the same edge in the same direction and we are in the same mode (clockwise or counter-clockwise).

Embedding Schemes

An embedding scheme is a pair $\Pi = (\pi, \lambda)$ where π is a rotation system, and $\lambda : E(G) \to \{-1, 1\}$ is a signature. Two embedding schemes are equivalent if one can be obtained from the other by a sequence of operations, each involving a change of clockwise to counter-clockwise orientation at a vertex v and the corresponding change of signatures of edges incident with v. Now we define a surface embedding using π-walks and π-polygons as before, with the following modification. Whenever we traverse an edge of negative signature, we switch the permutation at a vertex from $\pi(v)$ to $\pi^{-1}(v)$. Traversing a π-walk stops whenever we are about to traverse the same edge in the same direction and we are in the same mode (clockwise or counter-clockwise). The resulting surface is non-orientable if and only if G contains a cycle with an odd number of edges of negative signature.

Embedding Schemes

An embedding scheme is a pair $\Pi = (\pi, \lambda)$ where π is a rotation system, and $\lambda : E(G) \to \{-1, 1\}$ is a signature. Two embedding schemes are equivalent if one can be obtained from the other by a sequence of operations, each involving a change of clockwise to counter-clockwise orientation at a vertex v and the corresponding change of signatures of edges incident with v. Now we define a surface embedding using π-walks and π-polygons as before, with the following modification. Whenever we traverse an edge of negative signature, we switch the permutation at a vertex from $\pi(v)$ to $\pi^{-1}(v)$. Traversing a π-walk stops whenever we are about to traverse the same edge in the same direction and we are in the same mode (clockwise or counter-clockwise). The resulting surface is non-orientable if and only if G contains a cycle with an odd number of edges of negative signature.

Theorem 9.6

Every cellular embedding of a graph in some surface is uniquely determined, up to homeomorphism, by its embedding scheme.

Genus of a Graph

Definition 9.7

The genus $\gamma(G)$ and the non-orientable genus $\tilde{\gamma}(G)$ of a graph G are the minimum h and the minimum k, respectively, such that G has an embedding into the surface $\mathbb{S}_h$, respectively into $\mathbb{N}_k$.

Genus of a Graph

Definition 9.7

The genus $\gamma(G)$ and the non-orientable genus $\tilde{\gamma}(G)$ of a graph G are the minimum h and the minimum k, respectively, such that G has an embedding into the surface $\mathbb{S}_h$, respectively into $\mathbb{N}_k$. An embedding into such surface of minimum genus is minimum genus, respectively minimum non-orientable genus, embedding.

Genus of a Graph

Definition 9.7

The genus $\gamma(G)$ and the non-orientable genus $\tilde{\gamma}(G)$ of a graph G are the minimum h and the minimum k, respectively, such that G has an embedding into the surface $\mathbb{S}_h$, respectively into $\mathbb{N}_k$. An embedding into such surface of minimum genus is minimum genus, respectively minimum non-orientable genus, embedding.

Theorem 9.8

Every minimum (orientable) genus embedding of a connected graph is cellular.

Genus of a Graph

Definition 9.7

The genus $\gamma(G)$ and the non-orientable genus $\tilde{\gamma}(G)$ of a graph G are the minimum h and the minimum k, respectively, such that G has an embedding into the surface $\mathbb{S}_h$, respectively into $\mathbb{N}_k$. An embedding into such surface of minimum genus is minimum genus, respectively minimum non-orientable genus, embedding.

Theorem 9.8

Every minimum (orientable) genus embedding of a connected graph is cellular.

Theorem 9.9

Let G be a connected graph. If $\tilde{\gamma}(G) < 2\gamma(G) + 1$, then every non-orientable minimum genus embedding of G is cellular.

Genus of a Graph

Definition 9.7

The genus $\gamma(G)$ and the non-orientable genus $\tilde{\gamma}(G)$ of a graph G are the minimum h and the minimum k, respectively, such that G has an embedding into the surface $\mathbb{S}_h$, respectively into $\mathbb{N}_k$. An embedding into such surface of minimum genus is minimum genus, respectively minimum non-orientable genus, embedding.

Theorem 9.8

Every minimum (orientable) genus embedding of a connected graph is cellular.

Theorem 9.9

Let G be a connected graph. If $\tilde{\gamma}(G) < 2\gamma(G) + 1$, then every non-orientable minimum genus embedding of G is cellular. If $\tilde{\gamma}(G) = 2\gamma(G) + 1$ and G is not a tree, then G has both a cellular and a non-cellular embedding in $\mathbb{N}_{\tilde{\gamma}(G)}$.

Cycle Double-Cover Conjecture

Conjecture 9.10 (Cycle Double-Cover Conjecture)

Every 2-edge-connected graph G can be expressed as a union of cycles so that every edge of G appears in exactly two cycles.

Cycle Double-Cover Conjecture

Conjecture 9.10 (Cycle Double-Cover Conjecture)

Every 2-edge-connected graph G can be expressed as a union of cycles so that every edge of G appears in exactly two cycles.

Conjecture 9.11

Every 2-edge-connected graph has an embedding in some surface so that every face with the boundary is homeomorphic to the closed unit disk.

Cycle Double-Cover Conjecture

Conjecture 9.10 (Cycle Double-Cover Conjecture)

Every 2-edge-connected graph G can be expressed as a union of cycles so that every edge of G appears in exactly two cycles.

Conjecture 9.11

Every 2-edge-connected graph has an embedding in some surface so that every face with the boundary is homeomorphic to the closed unit disk. Holds for 4-connected graphs.

Comparisons of Graph Topological Parameters

- Euler Characteristic: $\chi = v - e + f$

Comparisons of Graph Topological Parameters

- Euler Characteristic: $\chi = v - e + f$
- Orientable Genus: $\chi = 2 - 2\gamma$

Comparisons of Graph Topological Parameters

- Euler Characteristic: $\chi = v - e + f$
- Orientable Genus: $\chi = 2 - 2\gamma \implies \gamma = 1 - \frac{\chi}{2}$

Comparisons of Graph Topological Parameters

- Euler Characteristic: $\chi = v - e + f$
- Orientable Genus: $\chi = 2 - 2\gamma \implies \gamma = 1 - \frac{\chi}{2}$
- Non-Orientable Genus: $\chi = 2 - \tilde{\gamma}$

Comparisons of Graph Topological Parameters

- Euler Characteristic: $\chi = v - e + f$
- Orientable Genus: $\chi = 2 - 2\gamma \implies \gamma = 1 - \frac{\chi}{2}$
- Non-Orientable Genus: $\chi = 2 - \tilde{\gamma} \implies \tilde{\gamma} = 2 - \chi$

Comparisons of Graph Topological Parameters

- Euler Characteristic: $\chi = v - e + f$
- Orientable Genus: $\chi = 2 - 2\gamma \implies \gamma = 1 - \frac{\chi}{2}$
- Non-Orientable Genus: $\chi = 2 - \tilde{\gamma} \implies \tilde{\gamma} = 2 - \chi$
- Euler Genus: $g = 2 - \chi$

Comparisons of Graph Topological Parameters

- Euler Characteristic: $\chi = v - e + f$
- Orientable Genus: $\chi = 2 - 2\gamma \implies \gamma = 1 - \frac{\chi}{2}$
- Non-Orientable Genus: $\chi = 2 - \tilde{\gamma} \implies \tilde{\gamma} = 2 - \chi$
- Euler Genus: $g = 2 - \chi \implies g = 2\gamma$

Comparisons of Graph Topological Parameters

- Euler Characteristic: $\chi = v - e + f$
- Orientable Genus: $\chi = 2 - 2\gamma \implies \gamma = 1 - \frac{\chi}{2}$
- Non-Orientable Genus: $\chi = 2 - \tilde{\gamma} \implies \tilde{\gamma} = 2 - \chi$
- Euler Genus: $g = 2 - \chi \Longrightarrow g = 2\gamma \quad \& \quad g = \tilde{\gamma}$

Bounds on Genus

Theorem 9.12

Let G be a simple connected graph with v vertices ($v \geq 3$) and e edges. Then

$$\gamma \geq \left\lceil \frac{e}{6} - \frac{v}{2} + 1 \right\rceil \qquad \textit{and} \qquad \tilde{\gamma} \geq \left\lceil \frac{e}{3} - v + 2 \right\rceil .$$

Bounds on Genus

Theorem 9.12

Let G be a simple connected graph with v vertices ($v \geq 3$) and e edges. Then

$$\gamma \geq \left\lceil \frac{e}{6} - \frac{v}{2} + 1 \right\rceil \qquad \text{and} \qquad \tilde{\gamma} \geq \left\lceil \frac{e}{3} - v + 2 \right\rceil .$$

Proof.

Let f denote the number of facial walks.

$\square$

Bounds on Genus

Theorem 9.12

Let G be a simple connected graph with v vertices ($v \geq 3$) and e edges. Then

$$\gamma \geq \left\lceil \frac{e}{6} - \frac{v}{2} + 1 \right\rceil \qquad \text{and} \qquad \tilde{\gamma} \geq \left\lceil \frac{e}{3} - v + 2 \right\rceil .$$

Proof.

Let f denote the number of facial walks. The sum of lengths of those walks is $2e$.

$\square$

Bounds on Genus

Theorem 9.12

Let G be a simple connected graph with v vertices ($v \geq 3$) and e edges. Then

$$\gamma \geq \left\lceil \frac{e}{6} - \frac{v}{2} + 1 \right\rceil \qquad \textit{and} \qquad \tilde{\gamma} \geq \left\lceil \frac{e}{3} - v + 2 \right\rceil .$$

Proof.

Let f denote the number of facial walks. The sum of lengths of those walks is $2e$. Since the graph is simple, $2e \geq 3f$.

□

Bounds on Genus

Theorem 9.12

Let G be a simple connected graph with v vertices ($v \geq 3$) and e edges. Then

$$\gamma \geq \left\lceil \frac{e}{6} - \frac{v}{2} + 1 \right\rceil \qquad \textit{and} \qquad \tilde{\gamma} \geq \left\lceil \frac{e}{3} - v + 2 \right\rceil .$$

Proof.

Let f denote the number of facial walks. The sum of lengths of those walks is $2e$. Since the graph is simple, $2e \geq 3f$. By Euler's Formula, $3\chi = 3v - 3e + 3f \leq 3v - e$, and so $\chi \leq v - e/3$.

□

Bounds on Genus

Theorem 9.12

Let G be a simple connected graph with v vertices ($v \geq 3$) and e edges. Then

$$\gamma \geq \left\lceil \frac{e}{6} - \frac{v}{2} + 1 \right\rceil \qquad \text{and} \qquad \tilde{\gamma} \geq \left\lceil \frac{e}{3} - v + 2 \right\rceil .$$

Proof.

Let f denote the number of facial walks. The sum of lengths of those walks is $2e$. Since the graph is simple, $2e \geq 3f$. By Euler's Formula, $3\chi = 3v - 3e + 3f \leq 3v - e$, and so $\chi \leq v - e/3$. Note that $\gamma = 1 - \chi/2$ and $\tilde{\gamma} = 2 - \chi$.

$\square$

Bounds on Genus

Theorem 9.12

Let G be a simple connected graph with v vertices ($v \geq 3$) and e edges. Then

$$\gamma \geq \left\lceil \frac{e}{6} - \frac{v}{2} + 1 \right\rceil \qquad \textit{and} \qquad \tilde{\gamma} \geq \left\lceil \frac{e}{3} - v + 2 \right\rceil .$$

Proof.

Let f denote the number of facial walks. The sum of lengths of those walks is $2e$. Since the graph is simple, $2e \geq 3f$. By Euler's Formula, $3\chi = 3v - 3e + 3f \leq 3v - e$, and so $\chi \leq v - e/3$. Note that $\gamma = 1 - \chi/2$ and $\tilde{\gamma} = 2 - \chi$. In the orientable case, we have

$$\gamma \geq 1 - \frac{v - e/3}{2} = \frac{e}{6} - \frac{v}{2} + 1.$$

□

Bounds on Genus

Theorem 9.12

Let G be a simple connected graph with v vertices ($v \geq 3$) and e edges. Then

$$\gamma \geq \left\lceil \frac{e}{6} - \frac{v}{2} + 1 \right\rceil \qquad \text{and} \qquad \tilde{\gamma} \geq \left\lceil \frac{e}{3} - v + 2 \right\rceil.$$

Proof.

Let f denote the number of facial walks. The sum of lengths of those walks is $2e$. Since the graph is simple, $2e \geq 3f$. By Euler's Formula, $3\chi = 3v - 3e + 3f \leq 3v - e$, and so $\chi \leq v - e/3$. Note that $\gamma = 1 - \chi/2$ and $\tilde{\gamma} = 2 - \chi$. In the orientable case, we have

$$\gamma \geq 1 - \frac{v - e/3}{2} = \frac{e}{6} - \frac{v}{2} + 1.$$

Similarly, in the nonorientable case, we have

$$\tilde{\gamma} \geq 2 - (v - \frac{e}{3}) = \frac{e}{3} - v + 2.$$

□

Bounds on Genus

Theorem 9.12

Let G be a simple connected graph with v vertices ($v \geq 3$) and e edges. Then

$$\gamma \geq \left\lceil \frac{e}{6} - \frac{v}{2} + 1 \right\rceil \qquad \textit{and} \qquad \tilde{\gamma} \geq \left\lceil \frac{e}{3} - v + 2 \right\rceil .$$

Proof.

Let f denote the number of facial walks. The sum of lengths of those walks is $2e$. Since the graph is simple, $2e \geq 3f$. By Euler's Formula, $3\chi = 3v - 3e + 3f \leq 3v - e$, and so $\chi \leq v - e/3$. Note that $\gamma = 1 - \chi/2$ and $\tilde{\gamma} = 2 - \chi$. In the orientable case, we have

$$\gamma \geq 1 - \frac{v - e/3}{2} = \frac{e}{6} - \frac{v}{2} + 1.$$

Similarly, in the nonorientable case, we have

$$\tilde{\gamma} \geq 2 - (v - \frac{e}{3}) = \frac{e}{3} - v + 2.$$

The theorem follows now from the fact that both γ and $\tilde{\gamma}$ are integers. $\square$

Genera of Complete Graphs

Corollary 9.13

$$\gamma(K_n) \geq \left\lceil \frac{(n-3)(n-4)}{12} \right\rceil \qquad \textit{and} \qquad \tilde{\gamma}(K_n) \geq \left\lceil \frac{(n-3)(n-4)}{6} \right\rceil$$

Heawood's Theorem

Theorem 9.14 (Ringel, Youngs)

If $n \geq 3$ and $n \neq 7$, then

$$\gamma(K_n) = \left\lceil \frac{(n-3)(n-4)}{12} \right\rceil \qquad \textit{and} \qquad \tilde{\gamma}(K_n) = \left\lceil \frac{(n-3)(n-4)}{6} \right\rceil$$

Heawood's Formula

Theorem 9.15 (Heawood's Formula)

Let S be a surface with Euler genus $g = 2 - (v - e + f) > 0$ and let G be a loopless graph embedded in S. Then

$$\chi(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 24g}}{2} \right\rfloor .$$

Heawood's Formula

Theorem 9.15 (Heawood's Formula)

Let S be a surface with Euler genus $g = 2 - (v - e + f) > 0$ and let G be a loopless graph embedded in S. Then

$$\chi(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 24g}}{2} \right\rfloor .$$

Proof.

Suppose the theorem fails and G is a minimal counter-example.

□

Heawood's Formula

Theorem 9.15 (Heawood's Formula)

Let S be a surface with Euler genus $g = 2 - (v - e + f) > 0$ and let G be a loopless graph embedded in S. Then

$$\chi(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 24g}}{2} \right\rfloor.$$

Proof.

Suppose the theorem fails and G is a minimal counter-example. Let $c = \chi(G)$, and note that since $g > 0$ and G is a counter-example, we have $c \geq 7$.

□

Heawood's Formula

Theorem 9.15 (Heawood's Formula)

Let S be a surface with Euler genus $g = 2 - (v - e + f) > 0$ and let G be a loopless graph embedded in S. Then

$$\chi(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 24g}}{2} \right\rfloor.$$

Proof.

Suppose the theorem fails and G is a minimal counter-example. Let $c = \chi(G)$, and note that since $g > 0$ and G is a counter-example, we have $c \geq 7$. From Theorem 9.12, we have $e \leq 3v - 6 + 3g$, and so $2e \leq 6v - 12 + 6g$.

□

Heawood's Formula

Theorem 9.15 (Heawood's Formula)

Let S be a surface with Euler genus $g = 2 - (v - e + f) > 0$ and let G be a loopless graph embedded in S. Then

$$\chi(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 24g}}{2} \right\rfloor.$$

Proof.

Suppose the theorem fails and G is a minimal counter-example. Let $c = \chi(G)$, and note that since $g > 0$ and G is a counter-example, we have $c \geq 7$. From Theorem 9.12, we have $e \leq 3v - 6 + 3g$, and so $2e \leq 6v - 12 + 6g$. The minimality of G implies that $\delta(G) \geq c - 1$.

□

Heawood's Formula

Theorem 9.15 (Heawood's Formula)

Let S be a surface with Euler genus $g = 2 - (v - e + f) > 0$ and let G be a loopless graph embedded in S. Then

$$\chi(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 24g}}{2} \right\rfloor .$$

Proof.

Suppose the theorem fails and G is a minimal counter-example. Let $c = \chi(G)$, and note that since $g > 0$ and G is a counter-example, we have $c \geq 7$. From Theorem 9.12, we have $e \leq 3v - 6 + 3g$, and so $2e \leq 6v - 12 + 6g$. The minimality of G implies that $\delta(G) \geq c - 1$. Clearly, $\delta v \leq 2e$, and since $c - 1 \leq \delta$, as implied by the minimality of G, we have $(c - 1)v \leq 2e$.

□

Heawood's Formula

Theorem 9.15 (Heawood's Formula)

Let S be a surface with Euler genus $g = 2 - (v - e + f) > 0$ and let G be a loopless graph embedded in S. Then

$$\chi(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 24g}}{2} \right\rfloor .$$

Proof.

Suppose the theorem fails and G is a minimal counter-example. Let $c = \chi(G)$, and note that since $g > 0$ and G is a counter-example, we have $c \geq 7$. From Theorem 9.12, we have $e \leq 3v - 6 + 3g$, and so $2e \leq 6v - 12 + 6g$. The minimality of G implies that $\delta(G) \geq c - 1$. Clearly, $\delta v \leq 2e$, and since $c - 1 \leq \delta$, as implied by the minimality of G, we have $(c - 1)v \leq 2e$. Combining this with an earlier inequality, we get $(c - 1)v \leq 6v - 12 + 6g$, and so $(c - 7)v + 12 - 6g \leq 0$.

□

Heawood's Formula

Theorem 9.15 (Heawood's Formula)

Let S be a surface with Euler genus $g = 2 - (v - e + f) > 0$ and let G be a loopless graph embedded in S. Then

$$\chi(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 24g}}{2} \right\rfloor.$$

Proof.

Suppose the theorem fails and G is a minimal counter-example. Let $c = \chi(G)$, and note that since $g > 0$ and G is a counter-example, we have $c \geq 7$. From Theorem 9.12, we have $e \leq 3v - 6 + 3g$, and so $2e \leq 6v - 12 + 6g$. The minimality of G implies that $\delta(G) \geq c - 1$. Clearly, $\delta v \leq 2e$, and since $c - 1 \leq \delta$, as implied by the minimality of G, we have $(c - 1)v \leq 2e$. Combining this with an earlier inequality, we get $(c - 1)v \leq 6v - 12 + 6g$, and so $(c - 7)v + 12 - 6g \leq 0$. Since $(c - 7) \geq 0$ and $c \leq v$, we have $c^2 - 7c + 12 - 6g \leq 0$. □

Heawood's Formula

Theorem 9.15 (Heawood's Formula)

Let S be a surface with Euler genus $g = 2 - (v - e + f) > 0$ and let G be a loopless graph embedded in S. Then

$$\chi(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 24g}}{2} \right\rfloor .$$

Proof.

Suppose the theorem fails and G is a minimal counter-example. Let $c = \chi(G)$, and note that since $g > 0$ and G is a counter-example, we have $c \geq 7$. From Theorem 9.12, we have $e \leq 3v - 6 + 3g$, and so $2e \leq 6v - 12 + 6g$. The minimality of G implies that $\delta(G) \geq c - 1$. Clearly, $\delta v \leq 2e$, and since $c - 1 \leq \delta$, as implied by the minimality of G, we have $(c - 1)v \leq 2e$. Combining this with an earlier inequality, we get $(c - 1)v \leq 6v - 12 + 6g$, and so $(c - 7)v + 12 - 6g \leq 0$. Since $(c - 7) \geq 0$ and $c \leq v$, we have $c^2 - 7c + 12 - 6g \leq 0$. Using the quadratic formula finishes the proof. □

Heawood's Formula

Theorem 9.15 (Heawood's Formula)

Let S be a surface with Euler genus $g = 2 - (v - e + f) > 0$ and let G be a loopless graph embedded in S. Then

$$\chi(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 24g}}{2} \right\rfloor .$$

Proof.

Suppose the theorem fails and G is a minimal counter-example. Let $c = \chi(G)$, and note that since $g > 0$ and G is a counter-example, we have $c \geq 7$. From Theorem 9.12, we have $e \leq 3v - 6 + 3g$, and so $2e \leq 6v - 12 + 6g$. The minimality of G implies that $\delta(G) \geq c - 1$. Clearly, $\delta v \leq 2e$, and since $c - 1 \leq \delta$, as implied by the minimality of G, we have $(c - 1)v \leq 2e$. Combining this with an earlier inequality, we get $(c - 1)v \leq 6v - 12 + 6g$, and so $(c - 7)v + 12 - 6g \leq 0$. Since $(c - 7) \geq 0$ and $c \leq v$, we have $c^2 - 7c + 12 - 6g \leq 0$. Using the quadratic formula finishes the proof. □

Theorem 9.16 (Ringel–Youngs)

The bound in Heawood's Formula is the best possible, except that maximum chromatic number of graphs embedded in the Klein bottle is 6.

Homework

Problem 16

Prove that for every number n there is a bipartite graph whose choosability number is greater than n.

Problem 17

Find the (orientable) genus of the Petersen graph.

Problem 18

Does K_5 have cellular embeddings into two different orientable surfaces? Into two different non-orientable surfaces?

Well-Quasi-Ordering

Definition 10.1

- A relation is a quasi-ordering if it is reflexive and transitive.

Well-Quasi-Ordering

Definition 10.1

- A relation is a quasi-ordering if it is reflexive and transitive.
- A quasi-ordering $\preccurlyeq$ on X is a well-quasi-ordering, or a wqo, if for every infinite sequence $x_0, x_1, \ldots$ in X there are indices $i < j$ such that $x_i \preccurlyeq x_j$.

Well-Quasi-Ordering

Definition 10.1

- A relation is a quasi-ordering if it is reflexive and transitive.
- A quasi-ordering $\preccurlyeq$ on X is a well-quasi-ordering, or a wqo, if for every infinite sequence $x_0, x_1, \ldots$ in X there are indices $i < j$ such that $x_i \preccurlyeq x_j$.
- Then (x_i, x_j) is a good pair for the sequence.

Well-Quasi-Ordering

Definition 10.1

- A relation is a quasi-ordering if it is reflexive and transitive.
- A quasi-ordering $\preccurlyeq$ on X is a well-quasi-ordering, or a wqo, if for every infinite sequence $x_0, x_1, \ldots$ in X there are indices $i < j$ such that $x_i \preccurlyeq x_j$.
- Then (x_i, x_j) is a good pair for the sequence.
- An infinite sequence containing a good pair is good;

Well-Quasi-Ordering

Definition 10.1

- A relation is a quasi-ordering if it is reflexive and transitive.
- A quasi-ordering $\preccurlyeq$ on X is a well-quasi-ordering, or a wqo, if for every infinite sequence $x_0, x_1, \ldots$ in X there are indices $i < j$ such that $x_i \preccurlyeq x_j$.
- Then (x_i, x_j) is a good pair for the sequence.
- An infinite sequence containing a good pair is good; otherwise it is bad.

Antichains and Decreasing Sequences

Theorem 10.2

A quasi ordering $\preccurlyeq$ on X is a wqo if and only if X contains neither an infinite antichain nor an infinite strictly descending chain $x_0 \succ x_1 \succ \dots$.

Antichains and Decreasing Sequences

Theorem 10.2

A quasi ordering $\preccurlyeq$ on X is a wqo if and only if X contains neither an infinite antichain nor an infinite strictly descending chain $x_0 \succ x_1 \succ \dots$.

Proof.

The forward implication is obvious.

$\square$

Antichains and Decreasing Sequences

Theorem 10.2

A quasi ordering $\preccurlyeq$ on X is a wqo if and only if X contains neither an infinite antichain nor an infinite strictly descending chain $x_0 \succ x_1 \succ \ldots$.

Proof.

The forward implication is obvious. Conversely, let $\mathbf{x} = x_0, x_1, \ldots$ be a bad sequence in X.

$\square$

Antichains and Decreasing Sequences

Theorem 10.2

A quasi ordering $\preccurlyeq$ on X is a wqo if and only if X contains neither an infinite antichain nor an infinite strictly descending chain $x_0 \succ x_1 \succ \dots$.

Proof.

The forward implication is obvious. Conversely, let $\mathbf{x} = x_0, x_1, \dots$ be a bad sequence in X. We define infinite subsequences $\mathbf{y}$, $\mathbf{x}_0$, $\mathbf{x}_1 \dots$ of $\mathbf{x}$, and two sets A and B as follows.

□

Antichains and Decreasing Sequences

Theorem 10.2

A quasi ordering $\preccurlyeq$ on X is a wqo if and only if X contains neither an infinite antichain nor an infinite strictly descending chain $x_0 \succ x_1 \succ \dots$.

Proof.

The forward implication is obvious. Conversely, let $\mathbf{x} = x_0, x_1, \dots$ be a bad sequence in X. We define infinite subsequences $\mathbf{y}$, $\mathbf{x}_0$, $\mathbf{x}_1 \dots$ of $\mathbf{x}$, and two sets A and B as follows.
We start with $\mathbf{y}$ as the empty sequence, A and B as empty sets, and $\mathbf{x}_0 = \mathbf{x}$.

□

Antichains and Decreasing Sequences

Theorem 10.2

A quasi ordering $\preccurlyeq$ on X is a wqo if and only if X contains neither an infinite antichain nor an infinite strictly descending chain $x_0 \succ x_1 \succ \dots$.

Proof.

The forward implication is obvious. Conversely, let $\mathbf{x} = x_0, x_1, \ \dots$ be a bad sequence in X. We define infinite subsequences $\mathbf{y}$, $\mathbf{x}_0$, $\mathbf{x}_1 \ \dots$ of $\mathbf{x}$, and two sets A and B as follows.
We start with $\mathbf{y}$ as the empty sequence, A and B as empty sets, and $\mathbf{x}_0 = \mathbf{x}$. Suppose that for some $i = 0, 1, \dots$, the first i elements of $\mathbf{y}$ and the sequence $\mathbf{x}_i$ have been defined, and each of the first i elements of $\mathbf{y}$ has been placed in exactly one of A and B.

$\square$

Antichains and Decreasing Sequences

Theorem 10.2

A quasi ordering $\preccurlyeq$ on X is a wqo if and only if X contains neither an infinite antichain nor an infinite strictly descending chain $x_0 \succ x_1 \succ \dots$.

Proof.

The forward implication is obvious. Conversely, let $\mathbf{x} = x_0, x_1, \dots$ be a bad sequence in X. We define infinite subsequences $\mathbf{y}$, $\mathbf{x}_0$, $\mathbf{x}_1 \dots$ of $\mathbf{x}$, and two sets A and B as follows.
We start with $\mathbf{y}$ as the empty sequence, A and B as empty sets, and $\mathbf{x}_0 = \mathbf{x}$.
Suppose that for some $i = 0, 1, \dots,$ the first i elements of $\mathbf{y}$ and the sequence $\mathbf{x}_i$ have been defined, and each of the first i elements of $\mathbf{y}$ has been placed in exactly one of A and B. Consider the first element x of $\mathbf{x}_i$.

$\square$

Antichains and Decreasing Sequences

Theorem 10.2

A quasi ordering $\preccurlyeq$ on X is a wqo if and only if X contains neither an infinite antichain nor an infinite strictly descending chain $x_0 \succ x_1 \succ \dots$.

Proof.

The forward implication is obvious. Conversely, let $\mathbf{x} = x_0, x_1, \dots$ be a bad sequence in X. We define infinite subsequences $\mathbf{y}$, $\mathbf{x}_0$, $\mathbf{x}_1$ ... of $\mathbf{x}$, and two sets A and B as follows.

We start with $\mathbf{y}$ as the empty sequence, A and B as empty sets, and $\mathbf{x}_0 = \mathbf{x}$. Suppose that for some $i = 0, 1, \dots$, the first i elements of $\mathbf{y}$ and the sequence $\mathbf{x}_i$ have been defined, and each of the first i elements of $\mathbf{y}$ has been placed in exactly one of A and B. Consider the first element x of $\mathbf{x}_i$.

- If infinitely many elements of $\mathbf{x}_i$ are incomparable with x, make $\mathbf{x}_{i+1}$ be an infinite subsequence of $\mathbf{x}_i$ consisting of the elements incomparable with x, and put x into A.

$\square$

Antichains and Decreasing Sequences

Theorem 10.2

A quasi ordering $\preccurlyeq$ on X is a wqo if and only if X contains neither an infinite antichain nor an infinite strictly descending chain $x_0 \succ x_1 \succ \dots$.

Proof.

The forward implication is obvious. Conversely, let $\mathbf{x} = x_0, x_1, \dots$ be a bad sequence in X. We define infinite subsequences $\mathbf{y}$, $\mathbf{x}_0$, $\mathbf{x}_1 \dots$ of $\mathbf{x}$, and two sets A and B as follows.

We start with $\mathbf{y}$ as the empty sequence, A and B as empty sets, and $\mathbf{x}_0 = \mathbf{x}$. Suppose that for some $i = 0, 1, \dots$, the first i elements of $\mathbf{y}$ and the sequence $\mathbf{x}_i$ have been defined, and each of the first i elements of $\mathbf{y}$ has been placed in exactly one of A and B. Consider the first element x of $\mathbf{x}_i$.

- If infinitely many elements of $\mathbf{x}_i$ are incomparable with x, make $\mathbf{x}_{i+1}$ be an infinite subsequence of $\mathbf{x}_i$ consisting of the elements incomparable with x, and put x into A.
- If this doesn't happen, there are infinitely many elements of $\mathbf{x}_i$ smaller than x. In that case, let $\mathbf{x}_{i+1}$ be an infinite subsequence of $\mathbf{x}_i$ consisting of the elements x' such that $x \succ x'$ and put x into B.

□

Proof, Continued

After this inductive construction, at least one of the sets A and B is infinite.

Proof, Continued

After this inductive construction, at least one of the sets A and B is infinite. If A is infinite, its elements form an antichain.

Proof, Continued

After this inductive construction, at least one of the sets A and B is infinite. If A is infinite, its elements form an antichain. If B is infinite, the sequence $\mathbf{y}$ restricted to B is an infinite strictly descending sequence.

Proof, Continued

After this inductive construction, at least one of the sets A and B is infinite. If A is infinite, its elements form an antichain. If B is infinite, the sequence $\mathbf{y}$ restricted to B is an infinite strictly descending sequence.

Theorem 10.3

If X is a wqo, then every infinite sequence in X has an infinite increasing subsequence.

Proof, Continued

After this inductive construction, at least one of the sets A and B is infinite. If A is infinite, its elements form an antichain. If B is infinite, the sequence $\mathbf{y}$ restricted to B is an infinite strictly descending sequence.

Theorem 10.3

If X is a wqo, then every infinite sequence in X has an infinite increasing subsequence.

Proof: Exercise.

Ordering of Sequences

Definition 10.4

- The set of all finite sequences of elements of a set X will be denoted by $X^{<\omega}$.

Ordering of Sequences

Definition 10.4

- The set of all finite sequences of elements of a set X will be denoted by $X^{<\omega}$.
- The set of all finite subsets of elements of a set X will be denoted by $[X]^{<\omega}$.

Ordering of Sequences

Definition 10.4

- The set of all finite sequences of elements of a set X will be denoted by $X^{<\omega}$.
- The set of all finite subsets of elements of a set X will be denoted by $[X]^{<\omega}$.
- The relation $\preccurlyeq$ on X is extended to $X^{<\omega}$ as follows: If $\mathbf{x} = (x_1, x_2, \ldots, x_m)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)$, we write $\mathbf{x} \preccurlyeq \mathbf{y}$ whenever there is a strictly increasing function $f : \{1, 2, \ldots, m\} \to \{1, 2, \ldots, n\}$ such that $x_i \preccurlyeq y_{f(i)}$ for all $i \in \{1, 2, \ldots, m\}$.

Ordering of Sequences

Definition 10.4

- The set of all finite sequences of elements of a set X will be denoted by $X^{<\omega}$.
- The set of all finite subsets of elements of a set X will be denoted by $[X]^{<\omega}$.
- The relation $\preccurlyeq$ on X is extended to $X^{<\omega}$ as follows: If $\mathbf{x} = (x_1, x_2, \ldots, x_m)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)$, we write $\mathbf{x} \preccurlyeq \mathbf{y}$ whenever there is a strictly increasing function $f : \{1, 2, \ldots, m\} \to \{1, 2, \ldots, n\}$ such that $x_i \preccurlyeq y_{f(i)}$ for all $i \in \{1, 2, \ldots, m\}$.
- The relation $\preccurlyeq$ on X is extended to $[X]^{<\omega}$ by saying that two elements $\mathbf{x}$ and $\mathbf{y}$ of $[X]^{<\omega}$ satisfy $\mathbf{x} \preccurlyeq \mathbf{y}$ whenever there is a injection f from $\mathbf{x}$ to $\mathbf{y}$ such that $x \preccurlyeq f(x)$ for all $x \in \mathbf{x}$.

Ordering of Sequences

Definition 10.4

- The set of all finite sequences of elements of a set X will be denoted by $X^{<\omega}$.
- The set of all finite subsets of elements of a set X will be denoted by $[X]^{<\omega}$.
- The relation $\preccurlyeq$ on X is extended to $X^{<\omega}$ as follows: If $\mathbf{x} = (x_1, x_2, \ldots, x_m)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)$, we write $\mathbf{x} \preccurlyeq \mathbf{y}$ whenever there is a strictly increasing function $f : \{1, 2, \ldots, m\} \to \{1, 2, \ldots, n\}$ such that $x_i \preccurlyeq y_{f(i)}$ for all $i \in \{1, 2, \ldots, m\}$.
- The relation $\preccurlyeq$ on X is extended to $[X]^{<\omega}$ by saying that two elements $\mathbf{x}$ and $\mathbf{y}$ of $[X]^{<\omega}$ satisfy $\mathbf{x} \preccurlyeq \mathbf{y}$ whenever there is a injection f from $\mathbf{x}$ to $\mathbf{y}$ such that $x \preccurlyeq f(x)$ for all $x \in \mathbf{x}$.
- A quasi-order X is well-founded if it has no infinite strictly descending chains.

Higman's Theorem

Theorem 10.5 (Higman)

If X is well-quasi-ordered by $\preccurlyeq$, then so is $X^{<\omega}$.

Higman's Theorem

Theorem 10.5 (Higman)

If X is well-quasi-ordered by $\preccurlyeq$, then so is $X^{<\omega}$.

Proof.

Suppose $X^{<\omega}$ is not a wqo.

$\square$

Higman's Theorem

Theorem 10.5 (Higman)

If X is well-quasi-ordered by $\preccurlyeq$, then so is $X^{<\omega}$.

Proof.

Suppose $X^{<\omega}$ is not a wqo. Observe that $X^{<\omega}$ is well-founded.

$\square$

Higman's Theorem

Theorem 10.5 (Higman)

If X is well-quasi-ordered by $\preccurlyeq$, then so is $X^{<\omega}$.

Proof.

Suppose $X^{<\omega}$ is not a wqo. Observe that $X^{<\omega}$ is well-founded. We construct a minimal bad sequence $(\mathbf{x}_n)_{n\in\mathbb{N}}$ in $X^{<\omega}$.

$\square$

Higman's Theorem

Theorem 10.5 (Higman)

If X is well-quasi-ordered by $\preccurlyeq$, then so is $X^{<\omega}$.

Proof.

Suppose $X^{<\omega}$ is not a wqo. Observe that $X^{<\omega}$ is well-founded. We construct a minimal bad sequence $(\mathbf{x}_n)_{n\in\mathbb{N}}$ in $X^{<\omega}$. Given $n \in \mathbb{N}$, assume inductively that $\mathbf{x}_i$ has been defined for every $i < n$ and that there is a bad sequence in $X^{<\omega}$ starting with $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{n-1}$.

$\square$

Theorem 10.5 (Higman)

If X is well-quasi-ordered by $\preccurlyeq$, then so is $X^{<\omega}$.

Proof.

Suppose $X^{<\omega}$ is not a wqo. Observe that $X^{<\omega}$ is well-founded. We construct a minimal bad sequence $(\mathbf{x}_n)_{n\in\mathbb{N}}$ in $X^{<\omega}$. Given $n \in \mathbb{N}$, assume inductively that $\mathbf{x}_i$ has been defined for every $i < n$ and that there is a bad sequence in $X^{<\omega}$ starting with $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{n-1}$. This is clearly true for $n = 0$.

$\square$

Higman's Theorem

Theorem 10.5 (Higman)

If X is well-quasi-ordered by $\preccurlyeq$, then so is $X^{<\omega}$.

Proof.

Suppose $X^{<\omega}$ is not a wqo. Observe that $X^{<\omega}$ is well-founded. We construct a minimal bad sequence $(\mathbf{x}_n)_{n\in\mathbb{N}}$ in $X^{<\omega}$. Given $n \in \mathbb{N}$, assume inductively that $\mathbf{x}_i$ has been defined for every $i < n$ and that there is a bad sequence in $X^{<\omega}$ starting with $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{n-1}$. This is clearly true for $n = 0$. Choose $\mathbf{x}_n \in X^{<\omega}$ so that some bad sequence starts with $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_n$ and $\mathbf{x}_n$ is minimal with this property (exists as $X^{<\omega}$ is well-founded).

$\square$

Proof of Higman's Theorem, Continued

Clearly, $(\mathbf{x}_n)_{n \in \mathbb{N}}$ is a bad sequence.

Proof of Higman's Theorem, Continued

Clearly, $(\mathbf{x}_n)_{n\in\mathbb{N}}$ is a bad sequence. For each n, let $\mathbf{y}_n$ be $\mathbf{x}_n$ with the last element x_n deleted.

Proof of Higman's Theorem, Continued

Clearly, $(\mathbf{x}_n)_{n\in\mathbb{N}}$ is a bad sequence. For each n, let $\mathbf{y}_n$ be $\mathbf{x}_n$ with the last element x_n deleted. By Theorem 10.3, the sequence $(x_n)_{n\in\mathbb{N}}$ has an infinite increasing subsequence $(x_{n_i})_{i\in\mathbb{N}}$.

Proof of Higman's Theorem, Continued

Clearly, $(\mathbf{x}_n)_{n\in\mathbb{N}}$ is a bad sequence. For each n, let $\mathbf{y}_n$ be $\mathbf{x}_n$ with the last element x_n deleted. By Theorem 10.3, the sequence $(x_n)_{n\in\mathbb{N}}$ has an infinite increasing subsequence $(x_{n_i})_{i\in\mathbb{N}}$. By the minimality of $\mathbf{x}_{n_0}$, the sequence $\mathbf{x}_0$, $\mathbf{x}_1, \ldots, \ \mathbf{x}_{n_0-1}, \mathbf{y}_{n_0}, \mathbf{y}_{n_1}, \ldots$ is good, so it has a good pair.

Proof of Higman's Theorem, Continued

Clearly, $(\mathbf{x}_n)_{n\in\mathbb{N}}$ is a bad sequence. For each n, let $\mathbf{y}_n$ be $\mathbf{x}_n$ with the last element x_n deleted. By Theorem 10.3, the sequence $(x_n)_{n\in\mathbb{N}}$ has an infinite increasing subsequence $(x_{n_i})_{i\in\mathbb{N}}$. By the minimality of $\mathbf{x}_{n_0}$, the sequence $\mathbf{x}_0$, $\mathbf{x}_1$, ..., $\mathbf{x}_{n_0-1}$, $\mathbf{y}_{n_0}$, $\mathbf{y}_{n_1}$, ... is good, so it has a good pair. This good pair must be of the form $(\mathbf{y}_i, \mathbf{y}_j)$.

Proof of Higman's Theorem, Continued

Clearly, $(\mathbf{x}_n)_{n\in\mathbb{N}}$ is a bad sequence. For each n, let $\mathbf{y}_n$ be $\mathbf{x}_n$ with the last element x_n deleted. By Theorem 10.3, the sequence $(x_n)_{n\in\mathbb{N}}$ has an infinite increasing subsequence $(x_{n_i})_{i\in\mathbb{N}}$. By the minimality of $\mathbf{x}_{n_0}$, the sequence $\mathbf{x}_0$, $\mathbf{x}_1, \ldots, \mathbf{x}_{n_0-1}, \mathbf{y}_{n_0}, \mathbf{y}_{n_1}, \ldots$ is good, so it has a good pair. This good pair must be of the form $(\mathbf{y}_i, \mathbf{y}_j)$. Extending the injection $\mathbf{y}_i \mapsto \mathbf{y}_j$ by $x_i \mapsto x_j$, we get a good pair $(\mathbf{x}_i, \mathbf{x}_j)$;

Proof of Higman's Theorem, Continued

Clearly, $(\mathbf{x}_n)_{n\in\mathbb{N}}$ is a bad sequence. For each n, let $\mathbf{y}_n$ be $\mathbf{x}_n$ with the last element x_n deleted. By Theorem 10.3, the sequence $(x_n)_{n\in\mathbb{N}}$ has an infinite increasing subsequence $(x_{n_i})_{i\in\mathbb{N}}$. By the minimality of $\mathbf{x}_{n_0}$, the sequence $\mathbf{x}_0$, $\mathbf{x}_1, \ldots, \mathbf{x}_{n_0-1}, \mathbf{y}_{n_0}, \mathbf{y}_{n_1}, \ldots$ is good, so it has a good pair. This good pair must be of the form $(\mathbf{y}_i, \mathbf{y}_j)$. Extending the injection $\mathbf{y}_i \mapsto \mathbf{y}_j$ by $x_i \mapsto x_j$, we get a good pair $(\mathbf{x}_i, \mathbf{x}_j)$; a contradiction.

Proof of Higman's Theorem, Continued

Clearly, $(\mathbf{x}_n)_{n\in\mathbb{N}}$ is a bad sequence. For each n, let $\mathbf{y}_n$ be $\mathbf{x}_n$ with the last element x_n deleted. By Theorem 10.3, the sequence $(x_n)_{n\in\mathbb{N}}$ has an infinite increasing subsequence $(x_{n_i})_{i\in\mathbb{N}}$. By the minimality of $\mathbf{x}_{n_0}$, the sequence $\mathbf{x}_0$, $\mathbf{x}_1, \ldots, \mathbf{x}_{n_0-1}, \mathbf{y}_{n_0}, \mathbf{y}_{n_1}, \ldots$ is good, so it has a good pair. This good pair must be of the form $(\mathbf{y}_i, \mathbf{y}_j)$. Extending the injection $\mathbf{y}_i \mapsto \mathbf{y}_j$ by $x_i \mapsto x_j$, we get a good pair $(\mathbf{x}_i, \mathbf{x}_j)$; a contradiction.

Corollary 10.6

If X is well-quasi-ordered by $\preccurlyeq$, then so is $[X]^{<\omega}$.

Proof of Higman's Theorem, Continued

Clearly, $(\mathbf{x}_n)_{n\in\mathbb{N}}$ is a bad sequence. For each n, let $\mathbf{y}_n$ be $\mathbf{x}_n$ with the last element x_n deleted. By Theorem 10.3, the sequence $(x_n)_{n\in\mathbb{N}}$ has an infinite increasing subsequence $(x_{n_i})_{i\in\mathbb{N}}$. By the minimality of $\mathbf{x}_{n_0}$, the sequence $\mathbf{x}_0$, $\mathbf{x}_1, \ldots, \mathbf{x}_{n_0-1}, \mathbf{y}_{n_0}, \mathbf{y}_{n_1}, \ldots$ is good, so it has a good pair. This good pair must be of the form $(\mathbf{y}_i, \mathbf{y}_j)$. Extending the injection $\mathbf{y}_i \mapsto \mathbf{y}_j$ by $x_i \mapsto x_j$, we get a good pair $(\mathbf{x}_i, \mathbf{x}_j)$; a contradiction.

Corollary 10.6

If X is well-quasi-ordered by $\preccurlyeq$, then so is $[X]^{<\omega}$.

Note 10.7 (Rado)

Higman's Theorem does not extend to infinite sequences.

Ordering Trees

Consider two trees T and T' with roots, respectively r and r'.

Ordering Trees

Consider two trees T and T' with roots, respectively r and r'. Note that the root r induces a natural partial order on the vertices of the tree T. Specifically, $u \leqslant_T v$ if u lies on the rv-path.

Ordering Trees

Consider two trees T and T' with roots, respectively r and r'. Note that the root r induces a natural partial order on the vertices of the tree T. Specifically, $u \leqslant_T v$ if u lies on the rv-path. We write $T \preccurlyeq T'$ if there is an isomorphism φ from some subdivision S of T to a subtree of S' of T' that preserves the tree order,

Ordering Trees

Consider two trees T and T' with roots, respectively r and r'. Note that the root r induces a natural partial order on the vertices of the tree T. Specifically, $u \leqslant_T v$ if u lies on the rv-path. We write $T \preccurlyeq T'$ if there is an isomorphism φ from some subdivision S of T to a subtree of S' of T' that preserves the tree order, that is, $u \leqslant_S v$ if and only if $\varphi(u) \leqslant_{S'} \varphi(v)$.

Ordering Trees

Consider two trees T and T' with roots, respectively r and r'. Note that the root r induces a natural partial order on the vertices of the tree T. Specifically, $u \leqslant_T v$ if u lies on the rv-path. We write $T \preccurlyeq T'$ if there is an isomorphism φ from some subdivision S of T to a subtree of S' of T' that preserves the tree order, that is, $u \leqslant_S v$ if and only if $\varphi(u) \leqslant_{S'} \varphi(v)$. Note that if $T \preccurlyeq T'$, then T is a topological minor of T'.

Ordering Trees

Consider two trees T and T' with roots, respectively r and r'. Note that the root r induces a natural partial order on the vertices of the tree T. Specifically, $u \leqslant_T v$ if u lies on the rv-path. We write $T \preccurlyeq T'$ if there is an isomorphism φ from some subdivision S of T to a subtree of S' of T' that preserves the tree order, that is, $u \leqslant_S v$ if and only if $\varphi(u) \leqslant_{S'} \varphi(v)$. Note that if $T \preccurlyeq T'$, then T is a topological minor of T'.

Theorem 10.8 (Kruskal 1960)

Trees are well-quasi-ordered by the topological minors relation.

Proof of Kruskal's Theorem

We show that rooted trees are well-quasi-ordered by $\preccurlyeq$.

Proof of Kruskal's Theorem

We show that rooted trees are well-quasi-ordered by $\preccurlyeq$. Suppose not.

Proof of Kruskal's Theorem

We show that rooted trees are well-quasi-ordered by $\preccurlyeq$. Suppose not. Let T_0, T_1, ... be a minimal bad sequence, with r_i being the root of T_i.

Proof of Kruskal's Theorem

We show that rooted trees are well-quasi-ordered by $\preccurlyeq$. Suppose not. Let T_0, T_1, ... be a minimal bad sequence, with r_i being the root of T_i. For each n, let A_n denote the set of components of $T_n - r_n$, made into rooted trees by choosing the neighbors of r_n as the roots.

Proof of Kruskal's Theorem

We show that rooted trees are well-quasi-ordered by $\preccurlyeq$. Suppose not. Let T_0, T_1, ... be a minimal bad sequence, with r_i being the root of T_i. For each n, let A_n denote the set of components of $T_n - r_n$, made into rooted trees by choosing the neighbors of r_n as the roots.
First, we show that $A = \bigcup A_n$ is a wqo.

Proof of Kruskal's Theorem

We show that rooted trees are well-quasi-ordered by $\preccurlyeq$. Suppose not. Let T_0, T_1, ... be a minimal bad sequence, with r_i being the root of T_i. For each n, let A_n denote the set of components of $T_n - r_n$, made into rooted trees by choosing the neighbors of r_n as the roots.
First, we show that $A = \bigcup A_n$ is a wqo. Let $(S_k)_{k \in \mathbb{N}}$ be a sequence of elements of A.

Proof of Kruskal's Theorem

We show that rooted trees are well-quasi-ordered by $\preccurlyeq$. Suppose not. Let T_0, T_1, ... be a minimal bad sequence, with r_i being the root of T_i. For each n, let A_n denote the set of components of $T_n - r_n$, made into rooted trees by choosing the neighbors of r_n as the roots.
First, we show that $A = \bigcup A_n$ is a wqo. Let $(S_k)_{k\in\mathbb{N}}$ be a sequence of elements of A. For each k, let $n = n(k)$ denote the A_n that contains S_k.

Proof of Kruskal's Theorem

We show that rooted trees are well-quasi-ordered by $\preccurlyeq$. Suppose not. Let T_0, T_1, ... be a minimal bad sequence, with r_i being the root of T_i. For each n, let A_n denote the set of components of $T_n - r_n$, made into rooted trees by choosing the neighbors of r_n as the roots.
First, we show that $A = \bigcup A_n$ is a wqo. Let $(S_k)_{k \in \mathbb{N}}$ be a sequence of elements of A. For each k, let $n = n(k)$ denote the A_n that contains S_k. Pick a k with the smallest $n(k)$. Then T_0, T_1, ..., $T_{n(k)-1}$, S_k, S_{k+1}, ... is a good sequence, by the minimality of (T_n).

Proof of Kruskal's Theorem

We show that rooted trees are well-quasi-ordered by $\preccurlyeq$. Suppose not. Let T_0, T_1, ... be a minimal bad sequence, with r_i being the root of T_i. For each n, let A_n denote the set of components of $T_n - r_n$, made into rooted trees by choosing the neighbors of r_n as the roots.
First, we show that $A = \bigcup A_n$ is a wqo. Let $(S_k)_{k\in\mathbb{N}}$ be a sequence of elements of A. For each k, let $n = n(k)$ denote the A_n that contains S_k. Pick a k with the smallest $n(k)$. Then T_0, T_1, ..., $T_{n(k)-1}$, S_k, S_{k+1}, ... is a good sequence, by the minimality of (T_n). Clearly, a good pair of that sequence must be of the form (S_i, S_j).

Proof of Kruskal's Theorem, Continued

By Corollary 10.6, the sequence $(A_n)_{n \in \mathbb{N}}$ has a good pair (A_i, A_j).

Proof of Kruskal's Theorem, Continued

By Corollary 10.6, the sequence $(A_n)_{n \in \mathbb{N}}$ has a good pair (A_i, A_j). Let $f : A_i \to A_j$ be an injection such that $T \preccurlyeq f(T)$ for all $T \in A_i$.

Proof of Kruskal's Theorem, Continued

By Corollary 10.6, the sequence $(A_n)_{n\in\mathbb{N}}$ has a good pair (A_i, A_j). Let $f : A_i \to A_j$ be an injection such that $T \preccurlyeq f(T)$ for all $T \in A_i$. We extend the union of those embeddings to a map φ from $V(T_i)$ to $V(T_j)$ by letting $\varphi(r_i) = r_j$.

Proof of Kruskal's Theorem, Continued

By Corollary 10.6, the sequence $(A_n)_{n\in\mathbb{N}}$ has a good pair (A_i, A_j). Let $f : A_i \to A_j$ be an injection such that $T \preccurlyeq f(T)$ for all $T \in A_i$. We extend the union of those embeddings to a map φ from $V(T_i)$ to $V(T_j)$ by letting $\varphi(r_i) = r_j$. The map φ is an embedding that preserves the tree order, proving that (T_i, T_j) is a good pair; a contradiction.

Tree-Decomposition

Let G be a graph, T be a tree, and let $\mathcal{V} = \{V_t\}_{t \in V(T)}$ be a family of vertex sets $V_t \subseteq V(G)$ (called bags).

Tree-Decomposition

Let G be a graph, T be a tree, and let $\mathcal{V} = \{V_t\}_{t \in V(T)}$ be a family of vertex sets $V_t \subseteq V(G)$ (called bags).

Definition 10.9

The pair $(T, \mathcal{V})$ is called a tree-decomposition of G if

Tree-Decomposition

Let G be a graph, T be a tree, and let $\mathcal{V} = \{V_t\}_{t \in V(T)}$ be a family of vertex sets $V_t \subseteq V(G)$ (called bags).

Definition 10.9

The pair $(T, \mathcal{V})$ is called a tree-decomposition of G if

(T1) $V(G) = \bigcup_{t \in V(T)} V_t$;

Tree-Decomposition

Let G be a graph, T be a tree, and let $\mathcal{V} = \{V_t\}_{t \in V(T)}$ be a family of vertex sets $V_t \subseteq V(G)$ (called bags).

Definition 10.9

The pair $(T, \mathcal{V})$ is called a tree-decomposition of G if

(T1) $V(G) = \bigcup_{t \in V(T)} V_t$;

(T2) For every edge e of G, there is a $t \in V(T)$ such that both endpoints of e are in V_t; and

Tree-Decomposition

Let G be a graph, T be a tree, and let $\mathcal{V} = \{V_t\}_{t \in V(T)}$ be a family of vertex sets $V_t \subseteq V(G)$ (called bags).

Definition 10.9

The pair $(T, \mathcal{V})$ is called a tree-decomposition of G if

(T1) $V(G) = \bigcup_{t \in V(T)} V_t$;

(T2) For every edge e of G, there is a $t \in V(T)$ such that both endpoints of e are in V_t; and

(T3) $V_r \cap V_t \subseteq V_s$ whenever s lies between r and t in T.

Tree-Decomposition

Let G be a graph, T be a tree, and let $\mathcal{V} = \{V_t\}_{t \in V(T)}$ be a family of vertex sets $V_t \subseteq V(G)$ (called bags).

Definition 10.9

The pair $(T, \mathcal{V})$ is called a tree-decomposition of G if

(T1) $V(G) = \bigcup_{t \in V(T)} V_t$;

(T2) For every edge e of G, there is a $t \in V(T)$ such that both endpoints of e are in V_t; and

(T3) $V_r \cap V_t \subseteq V_s$ whenever s lies between r and t in T.

(T3) (Alternate version) For every $v \in V(G)$, the subgraph T_v induced by those t for which $v \in V_t$ is connected.

Tree-Decomposition

Let G be a graph, T be a tree, and let $\mathcal{V} = \{V_t\}_{t \in V(T)}$ be a family of vertex sets $V_t \subseteq V(G)$ (called bags).

Definition 10.9

The pair $(T, \mathcal{V})$ is called a tree-decomposition of G if

(T1) $V(G) = \bigcup_{t \in V(T)} V_t$;

(T2) For every edge e of G, there is a $t \in V(T)$ such that both endpoints of e are in V_t; and

(T3) $V_r \cap V_t \subseteq V_s$ whenever s lies between r and t in T.

(T3) (Alternate version) For every $v \in V(G)$, the subgraph T_v induced by those t for which $v \in V_t$ is connected.

The width of the decomposition $(T, \mathcal{V})$ is the maximum of $|V_t| - 1$ taken over all $v \in V(T)$.

Tree-Decomposition

Let G be a graph, T be a tree, and let $\mathcal{V} = \{V_t\}_{t \in V(T)}$ be a family of vertex sets $V_t \subseteq V(G)$ (called bags).

Definition 10.9

The pair $(T, \mathcal{V})$ is called a tree-decomposition of G if

(T1) $V(G) = \bigcup_{t \in V(T)} V_t$;

(T2) For every edge e of G, there is a $t \in V(T)$ such that both endpoints of e are in V_t; and

(T3) $V_r \cap V_t \subseteq V_s$ whenever s lies between r and t in T.

(T3) (Alternate version) For every $v \in V(G)$, the subgraph T_v induced by those t for which $v \in V_t$ is connected.
The width of the decomposition $(T, \mathcal{V})$ is the maximum of $|V_t| - 1$ taken over all $v \in V(T)$. The tree-width of G, denoted by $\mathrm{tw}(G)$ is the minimum width over all possible tree-decomposisions.

Properties of Tree-Decompositions

Theorem 10.10

If H is a subgraph of G, and $(T, \{V_t\}_{t \in V(T)})$ is a tree-decomposition of G, then $(T, \{V_t \cap V(H)\}_{t \in V(T)})$ is a tree-decomposition of H.

Proof is very easy.

Properties of Tree-Decompositions

Theorem 10.10

If H is a subgraph of G, and $(T, \{V_t\}_{t \in V(T)})$ is a tree-decomposition of G, then $(T, \{V_t \cap V(H)\}_{t \in V(T)})$ is a tree-decomposition of H.

Proof is very easy.

Lemma 10.11

Let t_1t_2 be an edge of T, and let T_1 and T_2 be the components of $T \setminus t_1t_2$, with $t_1 \in V(T_1)$ and $t_2 \in V(T_2)$.

Theorem 10.10

If H is a subgraph of G, and $(T, \{V_t\}_{t \in V(T)})$ is a tree-decomposition of G, then $(T, \{V_t \cap V(H)\}_{t \in V(T)})$ is a tree-decomposition of H.

Proof is very easy.

Lemma 10.11

Let t_1t_2 be an edge of T, and let T_1 and T_2 be the components of $T \setminus t_1t_2$, with $t_1 \in V(T_1)$ and $t_2 \in V(T_2)$. Then $V_{t_1} \cap V_{t_2}$ separates $U_1 = \bigcup_{t \in V(T_1)} V_t$ from $U_2 = \bigcup_{t \in V(T_2)} V_t$.

Proof.

Both t_1 and t_2 lie on every $s_1 s_2$-path in T with $s_1 \in V(T_1)$ and $s_2 \in V(T_2)$. $\square$

Proof.

Both t_1 and t_2 lie on every $s_1 s_2$-path in T with $s_1 \in V(T_1)$ and $s_2 \in V(T_2)$. Therefore $U_1 \cap U_2 \subseteq V_{t_1} \cap V_{t_2}$ by (T3).

$\square$

Proof.

Both t_1 and t_2 lie on every $s_1 s_2$-path in T with $s_1 \in V(T_1)$ and $s_2 \in V(T_2)$. Therefore $U_1 \cap U_2 \subseteq V_{t_1} \cap V_{t_2}$ by (T3). What is left to show is that G has no edge $u_1 u_2$ with $u_1 \in U_1 - U_2$ and $u_2 \in U_2 - U_1$.

$\square$

Proof.

Both t_1 and t_2 lie on every s_1s_2-path in T with $s_1 \in V(T_1)$ and $s_2 \in V(T_2)$. Therefore $U_1 \cap U_2 \subseteq V_{t_1} \cap V_{t_2}$ by (T3). What is left to show is that G has no edge u_1u_2 with $u_1 \in U_1 - U_2$ and $u_2 \in U_2 - U_1$. If G has such an edge u_1u_2, then (T2) implies that there is a $t \in V(T)$ such that V_t contains both u_1 and u_2. □

Proof.

Both t_1 and t_2 lie on every $s_1 s_2$-path in T with $s_1 \in V(T_1)$ and $s_2 \in V(T_2)$. Therefore $U_1 \cap U_2 \subseteq V_{t_1} \cap V_{t_2}$ by (T3). What is left to show is that G has no edge $u_1 u_2$ with $u_1 \in U_1 - U_2$ and $u_2 \in U_2 - U_1$. If G has such an edge $u_1 u_2$, then (T2) implies that there is a $t \in V(T)$ such that V_t contains both u_1 and u_2. But t can be in neither T_1 nor in T_2; a contradiction. $\square$

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

(ii) *there are vertices w_1 and w_2 in W and an edge t_1t_2 of T such that w_1 and w_2 lie outside of $V_{t_1} \cap V_{t_2}$ and are separated by it in G.*

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

(ii) *there are vertices w_1 and w_2 in W and an edge t_1t_2 of T such that w_1 and w_2 lie outside of $V_{t_1} \cap V_{t_2}$ and are separated by it in G.*

Proof.

We orient the edges of T as follows.

$\square$

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

(ii) *there are vertices w_1 and w_2 in W and an edge t_1t_2 of T such that w_1 and w_2 lie outside of $V_{t_1} \cap V_{t_2}$ and are separated by it in G.*

Proof.

We orient the edges of T as follows. For each edge t_1t_2 of T, define U_1 and U_2 as in Lemma 10.11;

$\square$

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

(ii) *there are vertices w_1 and w_2 in W and an edge t_1t_2 of T such that w_1 and w_2 lie outside of $V_{t_1} \cap V_{t_2}$ and are separated by it in G.*

Proof.

We orient the edges of T as follows. For each edge t_1t_2 of T, define U_1 and U_2 as in Lemma 10.11; then $V_{t_1} \cap V_{t_2}$ separates U_1 from U_2.

$\square$

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

(ii) *there are vertices w_1 and w_2 in W and an edge t_1t_2 of T such that w_1 and w_2 lie outside of $V_{t_1} \cap V_{t_2}$ and are separated by it in G.*

Proof.

We orient the edges of T as follows. For each edge t_1t_2 of T, define U_1 and U_2 as in Lemma 10.11; then $V_{t_1} \cap V_{t_2}$ separates U_1 from U_2. If $V_{t_1} \cap V_{t_2}$ does not separate any two vertices of W, then $W \subseteq U_i$ for some $i \in \{1, 2\}$; we orient t_1t_2 towards that t_i.

□

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

(ii) *there are vertices w_1 and w_2 in W and an edge t_1t_2 of T such that w_1 and w_2 lie outside of $V_{t_1} \cap V_{t_2}$ and are separated by it in G.*

Proof.

We orient the edges of T as follows. For each edge t_1t_2 of T, define U_1 and U_2 as in Lemma 10.11; then $V_{t_1} \cap V_{t_2}$ separates U_1 from U_2. If $V_{t_1} \cap V_{t_2}$ does not separate any two vertices of W, then $W \subseteq U_i$ for some $i \in \{1, 2\}$; we orient t_1t_2 towards that t_i.
Let t be the last vertex of a maximal directed path in T; we claim that $W \subseteq V_t$.

□

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

(ii) *there are vertices w_1 and w_2 in W and an edge t_1t_2 of T such that w_1 and w_2 lie outside of $V_{t_1} \cap V_{t_2}$ and are separated by it in G.*

Proof.

We orient the edges of T as follows. For each edge t_1t_2 of T, define U_1 and U_2 as in Lemma 10.11; then $V_{t_1} \cap V_{t_2}$ separates U_1 from U_2. If $V_{t_1} \cap V_{t_2}$ does not separate any two vertices of W, then $W \subseteq U_i$ for some $i \in \{1, 2\}$; we orient t_1t_2 towards that t_i.

Let t be the last vertex of a maximal directed path in T; we claim that $W \subseteq V_t$. Suppose $w \in W$ and let $t' \in V(T)$ be such that $w \in V_{t'}$.

□

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

(ii) *there are vertices w_1 and w_2 in W and an edge t_1t_2 of T such that w_1 and w_2 lie outside of $V_{t_1} \cap V_{t_2}$ and are separated by it in G.*

Proof.

We orient the edges of T as follows. For each edge t_1t_2 of T, define U_1 and U_2 as in Lemma 10.11; then $V_{t_1} \cap V_{t_2}$ separates U_1 from U_2. If $V_{t_1} \cap V_{t_2}$ does not separate any two vertices of W, then $W \subseteq U_i$ for some $i \in \{1, 2\}$; we orient t_1t_2 towards that t_i.

Let t be the last vertex of a maximal directed path in T; we claim that $W \subseteq V_t$. Suppose $w \in W$ and let $t' \in V(T)$ be such that $w \in V_{t'}$. If $t' \neq t$, then the edge e at t that separates t from t' is directed towards t,

$\square$

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

(ii) *there are vertices w_1 and w_2 in W and an edge t_1t_2 of T such that w_1 and w_2 lie outside of $V_{t_1} \cap V_{t_2}$ and are separated by it in G.*

Proof.

We orient the edges of T as follows. For each edge t_1t_2 of T, define U_1 and U_2 as in Lemma 10.11; then $V_{t_1} \cap V_{t_2}$ separates U_1 from U_2. If $V_{t_1} \cap V_{t_2}$ does not separate any two vertices of W, then $W \subseteq U_i$ for some $i \in \{1, 2\}$; we orient t_1t_2 towards that t_i.

Let t be the last vertex of a maximal directed path in T; we claim that $W \subseteq V_t$. Suppose $w \in W$ and let $t' \in V(T)$ be such that $w \in V_{t'}$. If $t' \neq t$, then the edge e at t that separates t from t' is directed towards t, so w also lies in $V_{t''}$ for some t'' in the component of $T \setminus e$ containing t.

□

Lemma 10.12

For every $W \subseteq V(G)$, one of the following holds:

(i) *there is a $t \in V(T)$ such that $W \subseteq V_t$;*

(ii) *there are vertices w_1 and w_2 in W and an edge t_1t_2 of T such that w_1 and w_2 lie outside of $V_{t_1} \cap V_{t_2}$ and are separated by it in G.*

Proof.

We orient the edges of T as follows. For each edge t_1t_2 of T, define U_1 and U_2 as in Lemma 10.11; then $V_{t_1} \cap V_{t_2}$ separates U_1 from U_2. If $V_{t_1} \cap V_{t_2}$ does not separate any two vertices of W, then $W \subseteq U_i$ for some $i \in \{1, 2\}$; we orient t_1t_2 towards that t_i.

Let t be the last vertex of a maximal directed path in T; we claim that $W \subseteq V_t$. Suppose $w \in W$ and let $t' \in V(T)$ be such that $w \in V_{t'}$. If $t' \neq t$, then the edge e at t that separates t from t' is directed towards t, so w also lies in $V_{t''}$ for some t'' in the component of $T \setminus e$ containing t. Therefore $w \in V_t$ by (T3). $\square$

Corollary 10.13

If H is a complete subgraph of G, and $(T, \{V_t\}_{t \in V(T)})$ is a tree-decomposition of G, then there is a bag V_t that contains all vertices of H.

Corollary 10.13

If H is a complete subgraph of G, and $(T, \{V_t\}_{t \in V(T)})$ is a tree-decomposition of G, then there is a bag V_t that contains all vertices of H.

Theorem 10.14

If G is a minor of H, then $tw(G) \leq tw(H)$.

Corollary 10.13

If H is a complete subgraph of G, and $(T, \{V_t\}_{t \in V(T)})$ is a tree-decomposition of G, then there is a bag V_t that contains all vertices of H.

Theorem 10.14

If G is a minor of H, then $tw(G) \leq tw(H)$.

Proof.

If G is obtained from H by deleting an edge, then a tree-decomposition of H is also a tree-decomposition of G.

$\square$

Corollary 10.13

If H is a complete subgraph of G, and $(T, \{V_t\}_{t \in V(T)})$ is a tree-decomposition of G, then there is a bag V_t that contains all vertices of H.

Theorem 10.14

If G is a minor of H, then $tw(G) \leq tw(H)$.

Proof.

If G is obtained from H by deleting an edge, then a tree-decomposition of H is also a tree-decomposition of G. If G is obtained from H by deleting a vertex, then a tree-decomposition of H may be modified by removing the vertex from all bags to form a tree-decomposition of G.

$\square$

Corollary 10.13

If H is a complete subgraph of G, and $(T, \{V_t\}_{t \in V(T)})$ is a tree-decomposition of G, then there is a bag V_t that contains all vertices of H.

Theorem 10.14

If G is a minor of H, then $tw(G) \leq tw(H)$.

Proof.

If G is obtained from H by deleting an edge, then a tree-decomposition of H is also a tree-decomposition of G. If G is obtained from H by deleting a vertex, then a tree-decomposition of H may be modified by removing the vertex from all bags to form a tree-decomposition of G. If G is obtained from H by ontracting an edge uv to a new vertex w, then a tree-decomposition of H may be modified by replacing each occurence by u and v by w to form a tree-decomposition of G. □

Corollary 10.15

For every integer k, the class of graphs of tree-width at most k is closed under the taking of minors.

Corollary 10.15

For every integer k, the class of graphs of tree-width at most k is closed under the taking of minors.

Theorem 10.16 (Robertson-Seymour 1990)

For every positive integer k, the graphs of tree-width less than k are well-quasi-ordered by the minor relation.

Corollary 10.15

For every integer k, the class of graphs of tree-width at most k is closed under the taking of minors.

Theorem 10.16 (Robertson-Seymour 1990)

For every positive integer k, the graphs of tree-width less than k are well-quasi-ordered by the minor relation.

Theorem 10.17

- $\mathrm{tw}(G) < 2$ *if and only if* K_3 *is not a minor of* G.

Corollary 10.15

For every integer k, the class of graphs of tree-width at most k is closed under the taking of minors.

Theorem 10.16 (Robertson-Seymour 1990)

For every positive integer k, the graphs of tree-width less than k are well-quasi-ordered by the minor relation.

Theorem 10.17

- $\mathrm{tw}(G) < 2$ *if and only if* K_3 *is not a minor of* G.
- $\mathrm{tw}(G) < 3$ *if and only if* K_4 *is not a minor of* G.

Brambles

Definition 10.18

- Two subsets U and W of $V(G)$ touch if a vertex of U is in W or is a neighbor of a vertex in W.

Brambles

Definition 10.18

- Two subsets U and W of $V(G)$ touch if a vertex of U is in W or is a neighbor of a vertex in W.
- A set of mutually touching connected vertex sets in G is a bramble.

Brambles

Definition 10.18

- Two subsets U and W of $V(G)$ touch if a vertex of U is in W or is a neighbor of a vertex in W.
- A set of mutually touching connected vertex sets in G is a bramble.
- A subset of $V(G)$ covers a bramble if it meets each of its elements.

Brambles

Definition 10.18

- Two subsets U and W of $V(G)$ touch if a vertex of U is in W or is a neighbor of a vertex in W.
- A set of mutually touching connected vertex sets in G is a bramble.
- A subset of $V(G)$ covers a bramble if it meets each of its elements.
- The smallest number of vertices that cover a bramble is the order of the bramble.

Brambles

Definition 10.18

- Two subsets U and W of $V(G)$ touch if a vertex of U is in W or is a neighbor of a vertex in W.
- A set of mutually touching connected vertex sets in G is a bramble.
- A subset of $V(G)$ covers a bramble if it meets each of its elements.
- The smallest number of vertices that cover a bramble is the order of the bramble.

Lemma 10.19

Any set of vertices separating two covers of a bramble also covers that bramble.

Brambles

Definition 10.18

- Two subsets U and W of $V(G)$ touch if a vertex of U is in W or is a neighbor of a vertex in W.
- A set of mutually touching connected vertex sets in G is a bramble.
- A subset of $V(G)$ covers a bramble if it meets each of its elements.
- The smallest number of vertices that cover a bramble is the order of the bramble.

Lemma 10.19

Any set of vertices separating two covers of a bramble also covers that bramble.

Proof.

Since each set in a bramble is connected and meets both of the covers, it also meets any set separating these covers. □

Tree-width vs. Bramble

Theorem 10.20 (Seymour-Thomas 1993)

Let k be a non-negative integer. $\mathrm{tw}(G) \geq k$ if and only if G contains a bramble of order greater than k.

Tree-width vs. Bramble

Theorem 10.20 (Seymour-Thomas 1993)

Let k be a non-negative integer. $\mathrm{tw}(G) \geq k$ if and only if G contains a bramble of order greater than k.

Let $\mathcal{B}$ be a bramble in G. We show that every tree-decomposition $(T, (V_t)_{t \in V(T)})$ of G has a bag that covers $\mathcal{B}$.

Tree-width vs. Bramble

Theorem 10.20 (Seymour-Thomas 1993)

Let k be a non-negative integer. $\mathrm{tw}(G) \geq k$ if and only if G contains a bramble of order greater than k.

Let $\mathcal{B}$ be a bramble in G. We show that every tree-decomposition $(T, (V_t)_{t \in V(T)})$ of G has a bag that covers $\mathcal{B}$. Orient the edges $t_1 t_2$ as in the proof of Lemma 10.12.

Tree-width vs. Bramble

Theorem 10.20 (Seymour-Thomas 1993)

Let k be a non-negative integer. $\mathrm{tw}(G) \geq k$ if and only if G contains a bramble of order greater than k.

Let $\mathcal{B}$ be a bramble in G. We show that every tree-decomposition $(T, (V_t)_{t \in V(T)})$ of G has a bag that covers $\mathcal{B}$. Orient the edges $t_1 t_2$ as in the proof of Lemma 10.12. If $X = V_{t_1} \cap V_{t_2}$ covers $\mathcal{B}$, the conclusion holds.

Tree-width vs. Bramble

Theorem 10.20 (Seymour-Thomas 1993)

Let k be a non-negative integer. $\mathrm{tw}(G) \geq k$ if and only if G contains a bramble of order greater than k.

Let $\mathcal{B}$ be a bramble in G. We show that every tree-decomposition $(T, (V_t)_{t \in V(T)})$ of G has a bag that covers $\mathcal{B}$. Orient the edges $t_1 t_2$ as in the proof of Lemma 10.12. If $X = V_{t_1} \cap V_{t_2}$ covers $\mathcal{B}$, the conclusion holds. If not, then for each $B \in \mathcal{B}$ disjoint from X there is an $i \in \{1, 2\}$ such that $B \subseteq U_i \setminus X$.

Tree-width vs. Bramble

Theorem 10.20 (Seymour-Thomas 1993)

Let k be a non-negative integer. $\mathrm{tw}(G) \geq k$ if and only if G contains a bramble of order greater than k.

Let $\mathcal{B}$ be a bramble in G. We show that every tree-decomposition $(T, (V_t)_{t \in V(T)})$ of G has a bag that covers $\mathcal{B}$. Orient the edges $t_1 t_2$ as in the proof of Lemma 10.12. If $X = V_{t_1} \cap V_{t_2}$ covers $\mathcal{B}$, the conclusion holds. If not, then for each $B \in \mathcal{B}$ disjoint from X there is an $i \in \{1, 2\}$ such that $B \subseteq U_i \setminus X$. This i is the same for all such B, because they touch.

Tree-width vs. Bramble

Theorem 10.20 (Seymour-Thomas 1993)

Let k be a non-negative integer. $\mathrm{tw}(G) \geq k$ if and only if G contains a bramble of order greater than k.

Let $\mathcal{B}$ be a bramble in G. We show that every tree-decomposition $(T, (V_t)_{t \in V(T)})$ of G has a bag that covers $\mathcal{B}$. Orient the edges $t_1 t_2$ as in the proof of Lemma 10.12. If $X = V_{t_1} \cap V_{t_2}$ covers $\mathcal{B}$, the conclusion holds. If not, then for each $B \in \mathcal{B}$ disjoint from X there is an $i \in \{1, 2\}$ such that $B \subseteq U_i \setminus X$. This i is the same for all such B, because they touch. Orient the edge $t_1 t_2$ towards t_i.

Theorem 10.20 (Seymour-Thomas 1993)

Let k be a non-negative integer. $\operatorname{tw}(G) \geq k$ if and only if G contains a bramble of order greater than k.

Let $\mathcal{B}$ be a bramble in G. We show that every tree-decomposition $(T, (V_t)_{t \in V(T)})$ of G has a bag that covers $\mathcal{B}$. Orient the edges $t_1 t_2$ as in the proof of Lemma 10.12. If $X = V_{t_1} \cap V_{t_2}$ covers $\mathcal{B}$, the conclusion holds. If not, then for each $B \in \mathcal{B}$ disjoint from X there is an $i \in \{1, 2\}$ such that $B \subseteq U_i \setminus X$. This i is the same for all such B, because they touch. Orient the edge $t_1 t_2$ towards t_i. Then if t is the last vertex of a maximal directed path in T, then V_t covers $\mathcal{B}$.

Theorem 10.21

The tree-width of an $n \times n$ *grid* ($n > 1$) *is* n.

Theorem 10.21

The tree-width of an $n \times n$ grid ($n > 1$) is n.

Theorem 10.22 (Robertson-Seymour 1986)

For every integer r there is an integer k such that every graph of tree-width at least k has an $r \times r$ grid minor.

Theorem 10.21

The tree-width of an $n \times n$ grid ($n > 1$) is n.

Theorem 10.22 (Robertson-Seymour 1986)

For every integer r there is an integer k such that every graph of tree-width at least k has an $r \times r$ grid minor.

Theorem 10.23

Every planar graph is a minor of a sufficiently large grid.

Theorem 10.21

The tree-width of an $n \times n$ grid ($n > 1$) is n.

Theorem 10.22 (Robertson-Seymour 1986)

For every integer r there is an integer k such that every graph of tree-width at least k has an $r \times r$ grid minor.

Theorem 10.23

Every planar graph is a minor of a sufficiently large grid.

Theorem 10.24 (Robertson-Seymour)

Planar graphs are well-quasi-ordered by the minor relation.

Representativity

Definition 10.25

Suppose G is a graph embedded in a surface S. The representativity of G is the smallest number of points that a homotopically non-trivial cruve in S intersects the graph.

Representativity

Definition 10.25

Suppose G is a graph embedded in a surface S. The representativity of G is the smallest number of points that a homotopically non-trivial cruve in S intersects the graph. The S-representativity of an abstract graph H is the smallest representativity of all embbedings of H in S, or zero if no embedding exists.

Representativity

Definition 10.25

Suppose G is a graph embedded in a surface S. The representativity of G is the smallest number of points that a homotopically non-trivial cruve in S intersects the graph. The S-representativity of an abstract graph H is the smallest representativity of all embbedings of H in S, or zero if no embedding exists.

Theorem 10.26 (Robertson-Seymour)

Every graph embeddable on a surface S is a minor of a graph of sufficiently high S-representativity.

Representativity

Definition 10.25

Suppose G is a graph embedded in a surface S. The representativity of G is the smallest number of points that a homotopically non-trivial cruve in S intersects the graph. The S-representativity of an abstract graph H is the smallest representativity of all embbedings of H in S, or zero if no embedding exists.

Theorem 10.26 (Robertson-Seymour)

Every graph embeddable on a surface S is a minor of a graph of sufficiently high S-representativity.

Theorem 10.27

For every surface S (orientable or not), the graphs embeddable in S are well-quasi-ordered by the minors relation.

Graphs Almost Embedded on Surfaces

Let r, s, t, and u be non-negative integers.

Graphs Almost Embedded on Surfaces

Let r, s, t, and u be non-negative integers. Let $\mathcal{H}(r, s, t, u)$ is the class of graphs G such that after deleting at most r vortices each of depth at most s from G, and after deleting at most t vertices, the resulting graph has Euler genus at most u.

Graphs Almost Embedded on Surfaces

Let r, s, t, and u be non-negative integers. Let $\mathcal{H}(r, s, t, u)$ is the class of graphs G such that after deleting at most r vortices each of depth at most s from G, and after deleting at most t vertices, the resulting graph has Euler genus at most u. Let $\mathcal{G}(r, s, t, u)$ be the class of graphs that are obtained by repeated clique-summing graphs from $\mathcal{H}(r, s, t, u)$.

Let r, s, t, and u be non-negative integers. Let $\mathcal{H}(r,s,t,u)$ is the class of graphs G such that after deleting at most r vortices each of depth at most s from G, and after deleting at most t vertices, the resulting graph has Euler genus at most u. Let $\mathcal{G}(r,s,t,u)$ be the class of graphs that are obtained by repeated clique-summing graphs from $\mathcal{H}(r,s,t,u)$.

Theorem 10.28 (Robertson-Seymour)

The class $\mathcal{G}(r,s,t,u)$ is well-quasi-ordered by the minor relation.

Graph Minors Theorem

Theorem 10.29 (Robertson-Seymour)

For every integer k there are integers r, s, t, and u such that every graph without K_k-minor belongs to $\mathcal{G}(r, s, t, u)$.

Graph Minors Theorem

Theorem 10.29 (Robertson-Seymour)

For every integer k there are integers r, s, t, and u such that every graph without K_k-minor belongs to $\mathcal{G}(r, s, t, u)$.

Corollary 10.30

Every minor-closed class of graphs other than the class of all graphs is a subclass of some $\mathcal{G}(r, s, t, u)$.

Graph Minors Theorem

Theorem 10.29 (Robertson-Seymour)

For every integer k there are integers r, s, t, and u such that every graph without K_k-minor belongs to $\mathcal{G}(r, s, t, u)$.

Corollary 10.30

Every minor-closed class of graphs other than the class of all graphs is a subclass of some $\mathcal{G}(r, s, t, u)$.

Corollary 10.31

The class of all (finite) graphs is well-quasi-ordered by the minor relation.

Problem 19

For each integer n exceeding one, find a bramble of order $n+1$ in the $n \times n$ grid.

Problem 20

A tree T is a caterpillar if T contains a path P such that every vertex of T either lies on P or is adjacent to a vertex of P. A caterpillar forest is a disjoint union of caterpillars. Find the minor-minimal graphs that are not caterpillar forests.

Problem 21

What is the tree-width of the graph obtained from the Petersen graph by deleting one edge?

The Turán Graph

Question: Given a graph H, what is the greatest possible number of edges in a simple graph of order n that does not have H as a subgraph? We will answer this question when H is a complete graph.

The Turán Graph

Question: Given a graph H, what is the greatest possible number of edges in a simple graph of order n that does not have H as a subgraph? We will answer this question when H is a complete graph.

Definition 11.1

The unique complete r-partite graph on $n \geq r$ vertices whose partition sets differ by at most 1 is called the Turán graph $T^r(n)$.

The Turán Graph

Question: Given a graph H, what is the greatest possible number of edges in a simple graph of order n that does not have H as a subgraph? We will answer this question when H is a complete graph.

Definition 11.1

The unique complete r-partite graph on $n \geq r$ vertices whose partition sets differ by at most 1 is called the Turán graph $T^r(n)$. The size of $T^r(n)$ will be denoted by $t_r(n)$.

The Turán Graph

Question: Given a graph H, what is the greatest possible number of edges in a simple graph of order n that does not have H as a subgraph? We will answer this question when H is a complete graph.

Definition 11.1

The unique complete r-partite graph on $n \geq r$ vertices whose partition sets differ by at most 1 is called the Turán graph $T^r(n)$. The size of $T^r(n)$ will be denoted by $t_r(n)$.

Theorem 11.2 (Turán 1941)

Given integers r and n exceeding 1, the unique simple graph of order n without K_r as a subgraph of maximum possible size is $T^{r-1}(n)$.

Proof of Turán's Theorem

First, observe that among all simple k-partite $(k < r)$ graphs on n vertices, $T^{r-1}(n)$ has the largest size.

Proof of Turán's Theorem

First, observe that among all simple k-partite $(k < r)$ graphs on n vertices, $T^{r-1}(n)$ has the largest size. Observe also that $T^{r-1}(n)$ does not have K_r as a subgraph, but $T^r(n)$ does.

Proof of Turán's Theorem

First, observe that among all simple k-partite $(k < r)$ graphs on n vertices, $T^{r-1}(n)$ has the largest size. Observe also that $T^{r-1}(n)$ does not have K_r as a subgraph, but $T^r(n)$ does. Thus it suffices to show that a simple graph G of order n, the maximum size without K_r as a subgraph is complete multipartite.

Proof of Turán's Theorem

First, observe that among all simple k-partite $(k < r)$ graphs on n vertices, $T^{r-1}(n)$ has the largest size. Observe also that $T^{r-1}(n)$ does not have K_r as a subgraph, but $T^r(n)$ does. Thus it suffices to show that a simple graph G of order n, the maximum size without K_r as a subgraph is complete multipartite.

By duplicating a vertex v, we mean adding a new vertex v' and joining it to all neighbors of v (but not v itself).

Proof of Turán's Theorem

First, observe that among all simple k-partite $(k < r)$ graphs on n vertices, $T^{r-1}(n)$ has the largest size. Observe also that $T^{r-1}(n)$ does not have K_r as a subgraph, but $T^r(n)$ does. Thus it suffices to show that a simple graph G of order n, the maximum size without K_r as a subgraph is complete multipartite.

By duplicating a vertex v, we mean adding a new vertex v' and joining it to all neighbors of v (but not v itself).

If not, then non-adjacency is not an equivalence relation on $V(G)$,

Proof of Turán's Theorem

First, observe that among all simple k-partite $(k < r)$ graphs on n vertices, $T^{r-1}(n)$ has the largest size. Observe also that $T^{r-1}(n)$ does not have K_r as a subgraph, but $T^r(n)$ does. Thus it suffices to show that a simple graph G of order n, the maximum size without K_r as a subgraph is complete multipartite.

By duplicating a vertex v, we mean adding a new vertex v' and joining it to all neighbors of v (but not v itself).

If not, then non-adjacency is not an equivalence relation on $V(G)$, that is, there are vertices y_1, x, and y_2 such that y_1x and xy_2 do not form edges of G, but y_1y_2 does.

Proof of Turán's Theorem

First, observe that among all simple k-partite ($k < r$) graphs on n vertices, $T^{r-1}(n)$ has the largest size. Observe also that $T^{r-1}(n)$ does not have K_r as a subgraph, but $T^r(n)$ does. Thus it suffices to show that a simple graph G of order n, the maximum size without K_r as a subgraph is complete multipartite.
By duplicating a vertex v, we mean adding a new vertex v' and joining it to all neighbors of v (but not v itself).
If not, then non-adjacency is not an equivalence relation on $V(G)$, that is, there are vertices y_1, x, and y_2 such that y_1x and xy_2 do not form edges of G, but y_1y_2 does. If $d(y_1) > d(x)$, then deleting x and duplicating y_1 yields another K_r-free graph with more edges than G.

Proof of Turán's Theorem

First, observe that among all simple k-partite $(k < r)$ graphs on n vertices, $T^{r-1}(n)$ has the largest size. Observe also that $T^{r-1}(n)$ does not have K_r as a subgraph, but $T^r(n)$ does. Thus it suffices to show that a simple graph G of order n, the maximum size without K_r as a subgraph is complete multipartite.

By duplicating a vertex v, we mean adding a new vertex v' and joining it to all neighbors of v (but not v itself).

If not, then non-adjacency is not an equivalence relation on $V(G)$, that is, there are vertices y_1, x, and y_2 such that y_1x and xy_2 do not form edges of G, but y_1y_2 does. If $d(y_1) > d(x)$, then deleting x and duplicating y_1 yields another K_r-free graph with more edges than G. So $d(y_1) \leq d(x)$ and $d(y_2) \leq d(x)$.

Proof of Turán's Theorem

First, observe that among all simple k-partite $(k < r)$ graphs on n vertices, $T^{r-1}(n)$ has the largest size. Observe also that $T^{r-1}(n)$ does not have K_r as a subgraph, but $T^r(n)$ does. Thus it suffices to show that a simple graph G of order n, the maximum size without K_r as a subgraph is complete multipartite.

By duplicating a vertex v, we mean adding a new vertex v' and joining it to all neighbors of v (but not v itself).

If not, then non-adjacency is not an equivalence relation on $V(G)$, that is, there are vertices y_1, x, and y_2 such that y_1x and xy_2 do not form edges of G, but y_1y_2 does. If $d(y_1) > d(x)$, then deleting x and duplicating y_1 yields another K_r-free graph with more edges than G. So $d(y_1) \leq d(x)$ and $d(y_2) \leq d(x)$. But then deleting y_1 and y_2 and duplicating x twice gives a K_r-free graph with more edges than G; a contradiction.

Erdős-Stone Theorem

Corollary 11.3

If G is a simple graph of order n and size more than $t_{r-1}(n)$, then G contains K_r as a subgraph.

Erdős-Stone Theorem

Corollary 11.3

If G is a simple graph of order n and size more than $t_{r-1}(n)$, then G contains K_r as a subgraph.

Theorem 11.4 (Erdős-Stone, 1946)

For all integers $r \geq 2$ and $s \geq 1$, and every $\epsilon > 0$, there is an integer n_0 such that every simple graph of order $n \geq n_0$ and size at least $t_{r-1}(n) + \epsilon n^2$ contains the complete r-partite graph with each part of cardinality s.

Erdős-Stone Theorem

Corollary 11.3

If G is a simple graph of order n and size more than $t_{r-1}(n)$, then G contains K_r as a subgraph.

Theorem 11.4 (Erdős-Stone, 1946)

For all integers $r \geq 2$ and $s \geq 1$, and every $\epsilon > 0$, there is an integer n_0 such that every simple graph of order $n \geq n_0$ and size at least $t_{r-1}(n) + \epsilon n^2$ contains the complete r-partite graph with each part of cardinality s.

Proof omitted.

Erdős-Stone Theorem

Corollary 11.3

If G is a simple graph of order n and size more than $t_{r-1}(n)$, then G contains K_r as a subgraph.

Theorem 11.4 (Erdős-Stone, 1946)

For all integers $r \geq 2$ and $s \geq 1$, and every $\epsilon > 0$, there is an integer n_0 such that every simple graph of order $n \geq n_0$ and size at least $t_{r-1}(n) + \epsilon n^2$ contains the complete r-partite graph with each part of cardinality s.

Proof omitted.

Definition 11.5

Given a simple graph H and an integer n, let $h_n(H)$ denote the maximum edge density that a simple H-free graph of order n can have;

Erdős-Stone Theorem

Corollary 11.3

If G is a simple graph of order n and size more than $t_{r-1}(n)$, then G contains K_r as a subgraph.

Theorem 11.4 (Erdős-Stone, 1946)

For all integers $r \geq 2$ and $s \geq 1$, and every $\epsilon > 0$, there is an integer n_0 such that every simple graph of order $n \geq n_0$ and size at least $t_{r-1}(n) + \epsilon n^2$ contains the complete r-partite graph with each part of cardinality s.

Proof omitted.

Definition 11.5

Given a simple graph H and an integer n, let $h_n(H)$ denote the maximum edge density that a simple H-free graph of order n can have; that is, the maximum number of edges that a simple H-free graph of order n can have divided by $\binom{n}{2}$.

Erdős-Stone Theorem

Corollary 11.3

If G is a simple graph of order n and size more than $t_{r-1}(n)$, then G contains K_r as a subgraph.

Theorem 11.4 (Erdős-Stone, 1946)

For all integers $r \geq 2$ and $s \geq 1$, and every $\epsilon > 0$, there is an integer n_0 such that every simple graph of order $n \geq n_0$ and size at least $t_{r-1}(n) + \epsilon n^2$ contains the complete r-partite graph with each part of cardinality s.

Proof omitted.

Definition 11.5

Given a simple graph H and an integer n, let $h_n(H)$ denote the maximum edge density that a simple H-free graph of order n can have; that is, the maximum number of edges that a simple H-free graph of order n can have divided by $\binom{n}{2}$.

Lemma 11.6

$$\lim_{n\to\infty} h_n(K_r) = \frac{r-2}{r-1}.$$

Erdős-Stone Theorem

Corollary 11.3

If G is a simple graph of order n and size more than $t_{r-1}(n)$, then G contains K_r as a subgraph.

Theorem 11.4 (Erdős-Stone, 1946)

For all integers $r \geq 2$ and $s \geq 1$, and every $\epsilon > 0$, there is an integer n_0 such that every simple graph of order $n \geq n_0$ and size at least $t_{r-1}(n) + \epsilon n^2$ contains the complete r-partite graph with each part of cardinality s.

Proof omitted.

Definition 11.5

Given a simple graph H and an integer n, let $h_n(H)$ denote the maximum edge density that a simple H-free graph of order n can have; that is, the maximum number of edges that a simple H-free graph of order n can have divided by $\binom{n}{2}$.

Lemma 11.6

$$\lim_{n \to \infty} h_n(K_r) = \frac{r-2}{r-1}.$$

Corollary of Erdős-Stone Theorem

Corollary 11.7

For every simple, non-trivial graph H,

$$\lim_{n \to \infty} h_n(H) = \frac{\chi(H) - 2}{\chi(H) - 1}.$$

Corollary of Erdős-Stone Theorem

Corollary 11.7

For every simple, non-trivial graph H,

$$\lim_{n\to\infty} h_n(H) = \frac{\chi(H)-2}{\chi(H)-1}.$$

Proof.

Let $r = \chi(H)$.

□

Corollary of Erdős-Stone Theorem

Corollary 11.7

For every simple, non-trivial graph H,

$$\lim_{n\to\infty} h_n(H) = \frac{\chi(H)-2}{\chi(H)-1}.$$

Proof.

Let $r = \chi(H)$. Then H is not a subgraph of $T^{r-1}(n)$ for all n,

□

Corollary of Erdős-Stone Theorem

Corollary 11.7

For every simple, non-trivial graph H,

$$\lim_{n\to\infty} h_n(H) = \frac{\chi(H)-2}{\chi(H)-1}.$$

Proof.

Let $r = \chi(H)$. Then H is not a subgraph of $T^{r-1}(n)$ for all n, and so $h_n(K_r) \leq h_n(H)$.

$\square$

Corollary of Erdős-Stone Theorem

Corollary 11.7

For every simple, non-trivial graph H,

$$\lim_{n\to\infty} h_n(H) = \frac{\chi(H)-2}{\chi(H)-1}.$$

Proof.

Let $r = \chi(H)$. Then H is not a subgraph of $T^{r-1}(n)$ for all n, and so $h_n(K_r) \leq h_n(H)$.
On the other hand, if K_r^s denotes the complete r-partite graph on rs vertices with every part of cardinality s, then $h_n(H) \leq h_n(K_r^s)$ for all sufficiently large s.

$\square$

Corollary of Erdős-Stone Theorem

Corollary 11.7

For every simple, non-trivial graph H,

$$\lim_{n\to\infty} h_n(H) = \frac{\chi(H)-2}{\chi(H)-1}.$$

Proof.

Let $r = \chi(H)$. Then H is not a subgraph of $T^{r-1}(n)$ for all n, and so $h_n(K_r) \leq h_n(H)$.

On the other hand, if K_r^s denotes the complete r-partite graph on rs vertices with every part of cardinality s, then $h_n(H) \leq h_n(K_r^s)$ for all sufficiently large s. Moreover, Erdős-Stone Theorem 11.4 implies that for every $\epsilon > 0$ and n large enough

$$h_n(K_r^s) < h_n(K_r) + \frac{\epsilon n^2}{\binom{n}{2}}.$$

□

Corollary of Erdős-Stone Theorem

Corollary 11.7

For every simple, non-trivial graph H,

$$\lim_{n\to\infty} h_n(H) = \frac{\chi(H)-2}{\chi(H)-1}.$$

Proof.

Let $r=\chi(H)$. Then H is not a subgraph of $T^{r-1}(n)$ for all n, and so $h_n(K_r)\le h_n(H)$.
On the other hand, if K_r^s denotes the complete r-partite graph on rs vertices with every part of cardinality s, then $h_n(H)\le h_n(K_r^s)$ for all sufficiently large s. Moreover, Erdős-Stone Theorem 11.4 implies that for every $\epsilon>0$ and n large enough

$$h_n(K_r^s) < h_n(K_r) + \frac{\epsilon n^2}{\binom{n}{2}}.$$

Hence for large n, we have

$$h_n(K_r)\le h_n(H)\le h_n(K_r^s) < h_n(K_r)+\frac{\epsilon n^2}{\binom{n}{2}} = h_n(K_r)+\frac{2\epsilon}{1-1/n}\le h_n(K_r)+4\epsilon.$$

$\square$

Corollary of Erdős-Stone Theorem

Corollary 11.7

For every simple, non-trivial graph H,

$$\lim_{n\to\infty} h_n(H) = \frac{\chi(H)-2}{\chi(H)-1}.$$

Proof.

Let $r = \chi(H)$. Then H is not a subgraph of $T^{r-1}(n)$ for all n, and so $h_n(K_r) \le h_n(H)$.

On the other hand, if K_r^s denotes the complete r-partite graph on rs vertices with every part of cardinality s, then $h_n(H) \le h_n(K_r^s)$ for all sufficiently large s. Moreover, Erdős-Stone Theorem 11.4 implies that for every $\epsilon > 0$ and n large enough

$$h_n(K_r^s) < h_n(K_r) + \frac{\epsilon n^2}{\binom{n}{2}}.$$

Hence for large n, we have

$$h_n(K_r) \le h_n(H) \le h_n(K_r^s) < h_n(K_r) + \frac{\epsilon n^2}{\binom{n}{2}} = h_n(K_r) + \frac{2\epsilon}{1-1/n} \le h_n(K_r) + 4\epsilon.$$

Then Lemma 11.6 finishes the proof. $\square$

Ramsey Theorem

Theorem 11.8 (Ramsey 1930)

For every natural number r there is a natural number n such that every simple graph of order at least n contains either K_r or $\overline{K_r}$ as an induced subgraph.

Ramsey Theorem

Theorem 11.8 (Ramsey 1930)

For every natural number r there is a natural number n such that every simple graph of order at least n contains either K_r or $\overline{K_r}$ as an induced subgraph.

Proof.

Trivial for $r = 1$; assume $r \geq 2$.

$\square$

Ramsey Theorem

Theorem 11.8 (Ramsey 1930)

For every natural number r there is a natural number n such that every simple graph of order at least n contains either K_r or $\overline{K_r}$ as an induced subgraph.

Proof.

Trivial for $r = 1$; assume $r \geq 2$. Let $n = 2^{2r-3}$, and let G be a simple graph of order at least n.

$\square$

Ramsey Theorem

Theorem 11.8 (Ramsey 1930)

For every natural number r there is a natural number n such that every simple graph of order at least n contains either K_r or $\overline{K_r}$ as an induced subgraph.

Proof.

Trivial for $r = 1$; assume $r \geq 2$. Let $n = 2^{2r-3}$, and let G be a simple graph of order at least n. We will define a sequence V_1, V_2, ..., V_{2r-2} of sets and choose vertices $v_i \in V_i$ so that the following hold:

$\square$

Theorem 11.8 (Ramsey 1930)

For every natural number r there is a natural number n such that every simple graph of order at least n contains either K_r or $\overline{K_r}$ as an induced subgraph.

Proof.

Trivial for $r = 1$; assume $r \geq 2$. Let $n = 2^{2r-3}$, and let G be a simple graph of order at least n. We will define a sequence V_1, V_2, ..., V_{2r-2} of sets and choose vertices $v_i \in V_i$ so that the following hold:

(i) $|V_i| = 2^{2r-2-i}$ for all $i \in \{1, 2, \ldots, 2r-2\}$;

$\square$

Theorem 11.8 (Ramsey 1930)

For every natural number r there is a natural number n such that every simple graph of order at least n contains either K_r or $\overline{K_r}$ as an induced subgraph.

Proof.

Trivial for $r = 1$; assume $r \geq 2$. Let $n = 2^{2r-3}$, and let G be a simple graph of order at least n. We will define a sequence $V_1, V_2, \ldots, V_{2r-2}$ of sets and choose vertices $v_i \in V_i$ so that the following hold:

(i) $|V_i| = 2^{2r-2-i}$ for all $i \in \{1, 2, \ldots, 2r-2\}$;

(ii) $V_i \subseteq V_{i-1} \setminus \{v_{i-1}\}$ for all $i \in \{1, 2, \ldots, 2r-2\}$; and

□

Theorem 11.8 (Ramsey 1930)

For every natural number r there is a natural number n such that every simple graph of order at least n contains either K_r or $\overline{K_r}$ as an induced subgraph.

Proof.

Trivial for $r = 1$; assume $r \geq 2$. Let $n = 2^{2r-3}$, and let G be a simple graph of order at least n. We will define a sequence V_1, V_2, ..., V_{2r-2} of sets and choose vertices $v_i \in V_i$ so that the following hold:

(i) $|V_i| = 2^{2r-2-i}$ for all $i \in \{1, 2, \ldots, 2r-2\}$;

(ii) $V_i \subseteq V_{i-1} \setminus \{v_{i-1}\}$ for all $i \in \{1, 2, \ldots, 2r-2\}$; and

(iii) v_{i-1} is adjacent either to all vertices in V_i or to no vertex in V_i for all $i \in \{1, 2, \ldots, 2r-2\}$.

□

Ramsey Theorem

Theorem 11.8 (Ramsey 1930)

For every natural number r there is a natural number n such that every simple graph of order at least n contains either K_r or $\overline{K_r}$ as an induced subgraph.

Proof.

Trivial for $r = 1$; assume $r \geq 2$. Let $n = 2^{2r-3}$, and let G be a simple graph of order at least n. We will define a sequence $V_1, V_2, \ldots, V_{2r-2}$ of sets and choose vertices $v_i \in V_i$ so that the following hold:

(i) $|V_i| = 2^{2r-2-i}$ for all $i \in \{1, 2, \ldots, 2r-2\}$;

(ii) $V_i \subseteq V_{i-1} \setminus \{v_{i-1}\}$ for all $i \in \{1, 2, \ldots, 2r-2\}$; and

(iii) v_{i-1} is adjacent either to all vertices in V_i or to no vertex in V_i for all $i \in \{1, 2, \ldots, 2r-2\}$.

Let V_1 be any set of 2^{2r-3} vertices of and pick $v_1 \in V_1$ arbitrarily. Then (i)–(iii) hold trivially.

$\square$

Ramsey Theorem

Theorem 11.8 (Ramsey 1930)

For every natural number r there is a natural number n such that every simple graph of order at least n contains either K_r or $\overline{K_r}$ as an induced subgraph.

Proof.

Trivial for $r = 1$; assume $r \geq 2$. Let $n = 2^{2r-3}$, and let G be a simple graph of order at least n. We will define a sequence V_1, V_2, …, V_{2r-2} of sets and choose vertices $v_i \in V_i$ so that the following hold:

(i) $|V_i| = 2^{2r-2-i}$ for all $i \in \{1, 2, \ldots, 2r-2\}$;

(ii) $V_i \subseteq V_{i-1} \setminus \{v_{i-1}\}$ for all $i \in \{1, 2, \ldots, 2r-2\}$; and

(iii) v_{i-1} is adjacent either to all vertices in V_i or to no vertex in V_i for all $i \in \{1, 2, \ldots, 2r-2\}$.

Let V_1 be any set of 2^{2r-3} vertices of and pick $v_1 \in V_1$ arbitrarily. Then (i)–(iii) hold trivially.
Inductively, $|V_{i-1} \setminus \{v_{i-1}\}| = 2^{2r-1-i} - 1$, and $V_{i-1} \setminus \{v_{i-1}\}$ contains a subset V_i satisfying (i)–(iii); pick v_i arbitrarily in V_i.

$\square$

Ramsey Theorem

Theorem 11.8 (Ramsey 1930)

For every natural number r there is a natural number n such that every simple graph of order at least n contains either K_r or $\overline{K_r}$ as an induced subgraph.

Proof.

Trivial for $r = 1$; assume $r \geq 2$. Let $n = 2^{2r-3}$, and let G be a simple graph of order at least n. We will define a sequence V_1, V_2, $\ldots$, V_{2r-2} of sets and choose vertices $v_i \in V_i$ so that the following hold:

(i) $|V_i| = 2^{2r-2-i}$ for all $i \in \{1, 2, \ldots, 2r-2\}$;

(ii) $V_i \subseteq V_{i-1} \setminus \{v_{i-1}\}$ for all $i \in \{1, 2, \ldots, 2r-2\}$; and

(iii) v_{i-1} is adjacent either to all vertices in V_i or to no vertex in V_i for all $i \in \{1, 2, \ldots, 2r-2\}$.

Let V_1 be any set of 2^{2r-3} vertices of and pick $v_1 \in V_1$ arbitrarily. Then (i)–(iii) hold trivially.
Inductively, $|V_{i-1} \setminus \{v_{i-1}\}| = 2^{2r-1-i} - 1$, and $V_{i-1} \setminus \{v_{i-1}\}$ contains a subset V_i satisfying (i)–(iii); pick v_i arbitrarily in V_i. Among the vertices v_1, v_2 $\ldots$, v_{2r-3}, at least $r-1$ show the same behavior described in (iii). Those $r-1$ vertices together with v_{2r-2} induce either K_r or $\overline{K_r}$. $\square$

Infinite version of Ramsey's Theorem

Recall that $[X]^k$ denotes the set of k-elements subsets of a set X.

Infinite version of Ramsey's Theorem

Recall that $[X]^k$ denotes the set of k-elements subsets of a set X.
Given a c-coloring, that is, partitioning into c classes, of elements of $[X]^k$, we call a set $Y \subseteq X$ monochromatic if all elements of $[Y]^k$ receive the same color.

Infinite version of Ramsey's Theorem

Recall that $[X]^k$ denotes the set of k-elements subsets of a set X.
Given a c-coloring, that is, partitioning into c classes, of elements of $[X]^k$, we call a set $Y \subseteq X$ monochromatic if all elements of $[Y]^k$ receive the same color.

Then Ramsey's Theorem can be re-stated as: For every r there is an n such that if X is an n-element set and $[X]^2$ is 2-colored, then X has a monochromatic subset of cardinality r.

Infinite version of Ramsey's Theorem

Recall that $[X]^k$ denotes the set of k-elements subsets of a set X.
Given a c-coloring, that is, partitioning into c classes, of elements of $[X]^k$, we call a set $Y \subseteq X$ monochromatic if all elements of $[Y]^k$ receive the same color.

Then Ramsey's Theorem can be re-stated as: For every r there is an n such that if X is an n-element set and $[X]^2$ is 2-colored, then X has a monochromatic subset of cardinality r.

Theorem 11.9

Let k and c be positive integers, and let X be an infinite set. If $[X]^k$ is c-colored, then X has an infinite monochromatic subset.

Proof

We proceed by induction on k.

Proof

We proceed by induction on k. If $k = 1$, then the claim clearly holds.

Proof

We proceed by induction on k. If $k = 1$, then the claim clearly holds. Let $k > 0$ and assume that the theorem holds for all smaller values of k.

Proof

We proceed by induction on k. If $k = 1$, then the claim clearly holds. Let $k > 0$ and assume that the theorem holds for all smaller values of k. Let $[X]^k$ be colored with c colors. We will construct an infinite sequence X_0, X_1, ... of infinite subsets of X and choose elements $x_i \in X_i$ such that (for all i):

Proof

We proceed by induction on k. If $k = 1$, then the claim clearly holds. Let $k > 0$ and assume that the theorem holds for all smaller values of k. Let $[X]^k$ be colored with c colors. We will construct an infinite sequence $X_0, X_1, \ldots$ of infinite subsets of X and choose elements $x_i \in X_i$ such that (for all i):

(i) $X_{i+1} \subseteq X_i \setminus \{x_i\}$; and

Proof

We proceed by induction on k. If $k = 1$, then the claim clearly holds. Let $k > 0$ and assume that the theorem holds for all smaller values of k. Let $[X]^k$ be colored with c colors. We will construct an infinite sequence $X_0, X_1, \ldots$ of infinite subsets of X and choose elements $x_i \in X_i$ such that (for all i):

(i) $X_{i+1} \subseteq X_i \setminus \{x_i\}$; and

(ii) all k-element sets of the form $\{x_i\} \cup Z$ where $Z \subseteq [X]^{k-1}$ have the same color, which we associate with x_i.

Proof

We proceed by induction on k. If $k = 1$, then the claim clearly holds. Let $k > 0$ and assume that the theorem holds for all smaller values of k. Let $[X]^k$ be colored with c colors. We will construct an infinite sequence $X_0, X_1, \ldots$ of infinite subsets of X and choose elements $x_i \in X_i$ such that (for all i):

(i) $X_{i+1} \subseteq X_i \setminus \{x_i\}$; and

(ii) all k-element sets of the form $\{x_i\} \cup Z$ where $Z \subseteq [X]^{k-1}$ have the same color, which we associate with x_i.

Start with $X_0 = X$ and pick $x_0 \in X_0$ arbitrarily.

Proof

We proceed by induction on k. If $k = 1$, then the claim clearly holds. Let $k > 0$ and assume that the theorem holds for all smaller values of k. Let $[X]^k$ be colored with c colors. We will construct an infinite sequence $X_0, X_1, \ldots$ of infinite subsets of X and choose elements $x_i \in X_i$ such that (for all i):

(i) $X_{i+1} \subseteq X_i \setminus \{x_i\}$; and

(ii) all k-element sets of the form $\{x_i\} \cup Z$ where $Z \subseteq [X]^{k-1}$ have the same color, which we associate with x_i.

Start with $X_0 = X$ and pick $x_0 \in X_0$ arbitrarily. Having chosen X_i and $x_i \in X_i$, we c-color $[X_i \setminus \{x_i\}]^{k-1}$ by giving each set Z the color of $\{x_i\} \cup Z$ in our c-coloring of $[X]^k$.

Proof

We proceed by induction on k. If $k = 1$, then the claim clearly holds. Let $k > 0$ and assume that the theorem holds for all smaller values of k. Let $[X]^k$ be colored with c colors. We will construct an infinite sequence $X_0, X_1, \ldots$ of infinite subsets of X and choose elements $x_i \in X_i$ such that (for all i):

(i) $X_{i+1} \subseteq X_i \setminus \{x_i\}$; and

(ii) all k-element sets of the form $\{x_i\} \cup Z$ where $Z \subseteq [X]^{k-1}$ have the same color, which we associate with x_i.

Start with $X_0 = X$ and pick $x_0 \in X_0$ arbitrarily. Having chosen X_i and $x_i \in X_i$, we c-color $[X_i \setminus \{x_i\}]^{k-1}$ by giving each set Z the color of $\{x_i\} \cup Z$ in our c-coloring of $[X]^k$. By the induction hypothesis, $X_i \setminus \{x_i\}$ has an infinite monochromatic subset, which we choose as X_{k+1}. Pick $x_{k+1} \in X_{k+1}$ arbitrarily.

Proof

We proceed by induction on k. If $k = 1$, then the claim clearly holds. Let $k > 0$ and assume that the theorem holds for all smaller values of k. Let $[X]^k$ be colored with c colors. We will construct an infinite sequence $X_0, X_1, \ldots$ of infinite subsets of X and choose elements $x_i \in X_i$ such that (for all i):

(i) $X_{i+1} \subseteq X_i \setminus \{x_i\}$; and

(ii) all k-element sets of the form $\{x_i\} \cup Z$ where $Z \subseteq [X]^{k-1}$ have the same color, which we associate with x_i.

Start with $X_0 = X$ and pick $x_0 \in X_0$ arbitrarily. Having chosen X_i and $x_i \in X_i$, we c-color $[X_i \setminus \{x_i\}]^{k-1}$ by giving each set Z the color of $\{x_i\} \cup Z$ in our c-coloring of $[X]^k$. By the induction hypothesis, $X_i \setminus \{x_i\}$ has an infinite monochromatic subset, which we choose as X_{k+1}. Pick $x_{k+1} \in X_{k+1}$ arbitrarily.
Since c is finite, one of the colors is associated with infinitely many x_i—they form an infinite monochromatic subset of X.

Theorem 11.10 (König Infinity Lemma)

Let V_0, V_1, ... be an infinite sequence of disjoint non-empty finite sets, and let G be an infinite graph on their union. Assume that every vertex v in V_n, for $n \geq 1$, has a neighbor $f(v)$ in V_{n-1}. Then G contains a ray, that is a one-way-infinite path, $v_0 v_1 \ldots$ with $v_n \in V_n$ for all n.

Theorem 11.10 (König Infinity Lemma)

Let $V_0, V_1, \ldots$ be an infinite sequence of disjoint non-empty finite sets, and let G be an infinite graph on their union. Assume that every vertex v in V_n, for $n \geq 1$, has a neighbor $f(v)$ in V_{n-1}. Then G contains a ray, that is a one-way-infinite path, $v_0 v_1 \ldots$ with $v_n \in V_n$ for all n.

Proof.

Let $\mathcal{P}$ be the set of all finite paths of the form $v f(v) f(f(v)) \ldots$.

$\square$

Theorem 11.10 (König Infinity Lemma)

Let $V_0, V_1, \ldots$ be an infinite sequence of disjoint non-empty finite sets, and let G be an infinite graph on their union. Assume that every vertex v in V_n, for $n \geq 1$, has a neighbor $f(v)$ in V_{n-1}. Then G contains a ray, that is a one-way-infinite path, $v_0 v_1 \ldots$ with $v_n \in V_n$ for all n.

Proof.

Let $\mathcal{P}$ be the set of all finite paths of the form $v f(v) f(f(v)) \ldots$. Since V_0 is finite, but $\mathcal{P}$ is infinite, infinitely many of the paths in $\mathcal{P}$ begin at the same vertex v_0.

$\square$

Theorem 11.10 (König Infinity Lemma)

Let $V_0, V_1, \ldots$ be an infinite sequence of disjoint non-empty finite sets, and let G be an infinite graph on their union. Assume that every vertex v in V_n, for $n \geq 1$, has a neighbor $f(v)$ in V_{n-1}. Then G contains a ray, that is a one-way-infinite path, $v_0 v_1 \ldots$ with $v_n \in V_n$ for all n.

Proof.

Let $\mathcal{P}$ be the set of all finite paths of the form $v f(v) f(f(v)) \ldots$. Since V_0 is finite, but $\mathcal{P}$ is infinite, infinitely many of the paths in $\mathcal{P}$ begin at the same vertex v_0. On these infinitely many paths, infinitely many agree on $v_1 \in V_1$, because V_1 is finite.

□

Theorem 11.10 (König Infinity Lemma)

Let $V_0, V_1, \ldots$ be an infinite sequence of disjoint non-empty finite sets, and let G be an infinite graph on their union. Assume that every vertex v in V_n, for $n \geq 1$, has a neighbor $f(v)$ in V_{n-1}. Then G contains a ray, that is a one-way-infinite path, $v_0 v_1 \ldots$ with $v_n \in V_n$ for all n.

Proof.

Let $\mathcal{P}$ be the set of all finite paths of the form $v f(v) f(f(v)) \ldots$. Since V_0 is finite, but $\mathcal{P}$ is infinite, infinitely many of the paths in $\mathcal{P}$ begin at the same vertex v_0. On these infinitely many paths, infinitely many agree on $v_1 \in V_1$, because V_1 is finite. This gives rise to the inductive definition of v_n for every n so that $v_0 v_1 \ldots$ form a ray. $\square$

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Proof: To simplify notation, we will also use n denote the set $\{0, 1, \ldots, n-1\}$.

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Proof: To simplify notation, we will also use n denote the set $\{0, 1, \ldots, n-1\}$. Suppose the theorem fails for some k, c, and r.

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Proof: To simplify notation, we will also use n denote the set $\{0, 1, \ldots, n-1\}$.

Suppose the theorem fails for some k, c, and r. Then for every $n \geq k$ there is a c-coloring of $[n]^k$ such that n contains no monochromatic r-element subset.

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Proof: To simplify notation, we will also use n denote the set $\{0, 1, \ldots, n-1\}$.

Suppose the theorem fails for some k, c, and r. Then for every $n \geq k$ there is a c-coloring of $[n]^k$ such that n contains no monochromatic r-element subset. We will call such colorings bad.

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Proof: To simplify notation, we will also use n denote the set $\{0, 1, \ldots, n-1\}$.

Suppose the theorem fails for some k, c, and r. Then for every $n \geq k$ there is a c-coloring of $[n]^k$ such that n contains no monochromatic r-element subset. We will call such colorings bad.

For every $n \geq k$, let V_n be the (nonempty) set of bad colorings of $[n]^k$.

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Proof: To simplify notation, we will also use n denote the set $\{0, 1, \ldots, n-1\}$.
Suppose the theorem fails for some k, c, and r. Then for every $n \geq k$ there is a c-coloring of $[n]^k$ such that n contains no monochromatic r-element subset. We will call such colorings bad.
For every $n \geq k$, let V_n be the (nonempty) set of bad colorings of $[n]^k$. For $n > k$, the restriction $f(g)$ of any $g \in V_n$ to $[n-1]^k$ is still bad, and so lies in V_{n-1}.

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Proof: To simplify notation, we will also use n denote the set $\{0, 1, \ldots, n-1\}$.

Suppose the theorem fails for some k, c, and r. Then for every $n \geq k$ there is a c-coloring of $[n]^k$ such that n contains no monochromatic r-element subset. We will call such colorings bad.

For every $n \geq k$, let V_n be the (nonempty) set of bad colorings of $[n]^k$. For $n > k$, the restriction $f(g)$ of any $g \in V_n$ to $[n-1]^k$ is still bad, and so lies in V_{n-1}. By König Infinity Lemma 11.10, there is an infinite sequence g_k, g_{k+1}, ... of bad colorings $g_n \in V_n$ such that $f(g_n) = g_{n-1}$ for all $n > k$.

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Proof: To simplify notation, we will also use n denote the set $\{0, 1, \ldots, n-1\}$.
Suppose the theorem fails for some k, c, and r. Then for every $n \geq k$ there is a c-coloring of $[n]^k$ such that n contains no monochromatic r-element subset. We will call such colorings bad.
For every $n \geq k$, let V_n be the (nonempty) set of bad colorings of $[n]^k$. For $n > k$, the restriction $f(g)$ of any $g \in V_n$ to $[n-1]^k$ is still bad, and so lies in V_{n-1}. By König Infinity Lemma 11.10, there is an infinite sequence g_k, $g_{k+1}, \ldots$ of bad colorings $g_n \in V_n$ such that $f(g_n) = g_{n-1}$ for all $n > k$. For every $m \geq k$, all colorings g_n with $n \geq m$ agree on $[m]^k$, so for each $Y \in [\mathbb{N}]^k$ the value $g_n(Y)$ coincides for all $n > \max Y$. We define $g(Y)$ as this common value $g_n(Y)$.

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Proof: To simplify notation, we will also use n denote the set $\{0, 1, \ldots, n-1\}$.

Suppose the theorem fails for some k, c, and r. Then for every $n \geq k$ there is a c-coloring of $[n]^k$ such that n contains no monochromatic r-element subset. We will call such colorings bad.

For every $n \geq k$, let V_n be the (nonempty) set of bad colorings of $[n]^k$. For $n > k$, the restriction $f(g)$ of any $g \in V_n$ to $[n-1]^k$ is still bad, and so lies in V_{n-1}. By König Infinity Lemma 11.10, there is an infinite sequence g_k, $g_{k+1}, \ldots$ of bad colorings $g_n \in V_n$ such that $f(g_n) = g_{n-1}$ for all $n > k$. For every $m \geq k$, all colorings g_n with $n \geq m$ agree on $[m]^k$, so for each $Y \in [\mathbb{N}]^k$ the value $g_n(Y)$ coincides for all $n > \max Y$. We define $g(Y)$ as this common value $g_n(Y)$.

Then g is a bad coloring of $[\mathbb{N}]$ since every r-element subset S of $\mathbb{N}$ is contained in some sufficiently large $[n]$, and so S cannot be monochromatic since g coincides on $[n]^k$ with the bad coloring g_n.

Theorem 11.11

For all positive integers k, c, and r there is an integer $n \geq k$ such that every n-element set X has a monochromatic r-element subset with respect to any c-coloring of $[X]^k$.

Proof: To simplify notation, we will also use n denote the set $\{0, 1, \ldots, n-1\}$.
Suppose the theorem fails for some k, c, and r. Then for every $n \geq k$ there is a c-coloring of $[n]^k$ such that n contains no monochromatic r-element subset. We will call such colorings bad.
For every $n \geq k$, let V_n be the (nonempty) set of bad colorings of $[n]^k$. For $n > k$, the restriction $f(g)$ of any $g \in V_n$ to $[n-1]^k$ is still bad, and so lies in V_{n-1}. By König Infinity Lemma 11.10, there is an infinite sequence g_k, $g_{k+1}, \ldots$ of bad colorings $g_n \in V_n$ such that $f(g_n) = g_{n-1}$ for all $n > k$. For every $m \geq k$, all colorings g_n with $n \geq m$ agree on $[m]^k$, so for each $Y \in [\mathbb{N}]^k$ the value $g_n(Y)$ coincides for all $n > \max Y$. We define $g(Y)$ as this common value $g_n(Y)$.
Then g is a bad coloring of $[\mathbb{N}]$ since every r-element subset S of $\mathbb{N}$ is contained in some sufficiently large $[n]$, and so S cannot be monochromatic since g coincides on $[n]^k$ with the bad coloring g_n. This contradicts 11.9.

Ramsey numbers

Definition 11.12

The least integer n associated with k, c, and r as in Theorem 11.11 is the Ramsey number for k, c and r, and is denoted by $R(k, c, r)$.

Definition 11.12

The least integer n associated with k, c, and r as in Theorem 11.11 is the Ramsey number for k, c and r, and is denoted by $R(k, c, r)$. We will also use the notation $R(H_1, H_2)$ to denote the least order n such that H_1 is a subgraph of G or H_2 is a subgraph of $\overline{G}$ for every graph G of order n.

Definition 11.12

The least integer n associated with k, c, and r as in Theorem 11.11 is the Ramsey number for k, c and r, and is denoted by $R(k, c, r)$. We will also use the notation $R(H_1, H_2)$ to denote the least order n such that H_1 is a subgraph of G or H_2 is a subgraph of $\overline{G}$ for every graph G of order n. If $H_1 = H_2$, then $R(H_1, H_2)$ may be written as $R(H_1)$.

We proved before that $R(2, 2, 3) = 6$,

Ramsey numbers

Definition 11.12

The least integer n associated with k, c, and r as in Theorem 11.11 is the Ramsey number for k, c and r, and is denoted by $R(k, c, r)$. We will also use the notation $R(H_1, H_2)$ to denote the least order n such that H_1 is a subgraph of G or H_2 is a subgraph of $\overline{G}$ for every graph G of order n. If $H_1 = H_2$, then $R(H_1, H_2)$ may be written as $R(H_1)$.

We proved before that $R(2, 2, 3) = 6$, and that $R(K_3, K_3) = 6$.

Ramsey numbers

Definition 11.12

The least integer n associated with k, c, and r as in Theorem 11.11 is the Ramsey number for k, c and r, and is denoted by $R(k, c, r)$. We will also use the notation $R(H_1, H_2)$ to denote the least order n such that H_1 is a subgraph of G or H_2 is a subgraph of $\overline{G}$ for every graph G of order n. If $H_1 = H_2$, then $R(H_1, H_2)$ may be written as $R(H_1)$.

We proved before that $R(2, 2, 3) = 6$, and that $R(K_3, K_3) = 6$. In most cases the exact Ramsey numbers are not known. Most known values and bounds are listed at http://mathworld.wolfram.com/RamseyNumber.html

Theorem 11.13

Let s and t be positive integers, and let T be a tree of order t. Then $R(T, K_s) = (s-1)(t-1)+1$.

Theorem 11.13

Let s and t be positive integers, and let T be a tree of order t. Then $R(T, K_s) = (s-1)(t-1)+1$.

Proof.

The disjoint union of $s-1$ copies of K_{t-1} contains no copy of T,

$\square$

Theorem 11.13

Let s and t be positive integers, and let T be a tree of order t. Then $R(T, K_s) = (s-1)(t-1)+1$.

Proof.

The disjoint union of $s-1$ copies of K_{t-1} contains no copy of T, while the complement of this graph, the complete $s-1$-partite graph K_{s-1}^{t-1}, does not contain K_s.

$\square$

Theorem 11.13

Let s and t be positive integers, and let T be a tree of order t. Then $R(T, K_s) = (s-1)(t-1)+1$.

Proof.

The disjoint union of $s-1$ copies of K_{t-1} contains no copy of T, while the complement of this graph, the complete $s-1$-partite graph K_{s-1}^{t-1}, does not contain K_s. Thus $R(T, K_s) \geq (s-1)(t-1)+1$.

□

Theorem 11.13

Let s and t be positive integers, and let T be a tree of order t. Then $R(T, K_s) = (s-1)(t-1)+1$.

Proof.

The disjoint union of $s-1$ copies of K_{t-1} contains no copy of T, while the complement of this graph, the complete $s-1$-partite graph K_{s-1}^{t-1}, does not contain K_s. Thus $R(T, K_s) \geq (s-1)(t-1)+1$.
Conversely, suppose that G is any graph of order $n = (s-1)(t-1)+1$ whose complement contains no copy of K_s.

$\square$

Theorem 11.13

Let s and t be positive integers, and let T be a tree of order t. Then $R(T, K_s) = (s-1)(t-1)+1$.

Proof.

The disjoint union of $s-1$ copies of K_{t-1} contains no copy of T, while the complement of this graph, the complete $s-1$-partite graph K_{s-1}^{t-1}, does not contain K_s. Thus $R(T, K_s) \geq (s-1)(t-1)+1$.
Conversely, suppose that G is any graph of order $n = (s-1)(t-1)+1$ whose complement contains no copy of K_s. Then $s > 1$ and in any proper vertex coloring of G, at most $s-1$ vertices can get the same color.

□

Theorem 11.13

Let s and t be positive integers, and let T be a tree of order t. Then $R(T, K_s) = (s-1)(t-1)+1$.

Proof.

The disjoint union of $s-1$ copies of K_{t-1} contains no copy of T, while the complement of this graph, the complete $s-1$-partite graph K_{s-1}^{t-1}, does not contain K_s. Thus $R(T, K_s) \geq (s-1)(t-1)+1$.
Conversely, suppose that G is any graph of order $n = (s-1)(t-1)+1$ whose complement contains no copy of K_s. Then $s > 1$ and in any proper vertex coloring of G, at most $s-1$ vertices can get the same color. Hence $\chi(G) \geq \lceil n/(s-1) \rceil = t$,

□

Theorem 11.13

Let s and t be positive integers, and let T be a tree of order t. Then $R(T, K_s) = (s-1)(t-1) + 1$.

Proof.

The disjoint union of $s-1$ copies of K_{t-1} contains no copy of T, while the complement of this graph, the complete $s-1$-partite graph K_{s-1}^{t-1}, does not contain K_s. Thus $R(T, K_s) \geq (s-1)(t-1) + 1$.
Conversely, suppose that G is any graph of order $n = (s-1)(t-1) + 1$ whose complement contains no copy of K_s. Then $s > 1$ and in any proper vertex coloring of G, at most $s-1$ vertices can get the same color. Hence $\chi(G) \geq \lceil n/(s-1) \rceil = t$, and so G has a subgraph H with $\delta(H) \geq t-1$ (greedy coloring). □

Theorem 11.13

Let s and t be positive integers, and let T be a tree of order t. Then $R(T, K_s) = (s-1)(t-1)+1$.

Proof.

The disjoint union of $s-1$ copies of K_{t-1} contains no copy of T, while the complement of this graph, the complete $s-1$-partite graph K_{s-1}^{t-1}, does not contain K_s. Thus $R(T, K_s) \geq (s-1)(t-1)+1$.
Conversely, suppose that G is any graph of order $n = (s-1)(t-1)+1$ whose complement contains no copy of K_s. Then $s > 1$ and in any proper vertex coloring of G, at most $s-1$ vertices can get the same color. Hence $\chi(G) \geq \lceil n/(s-1) \rceil = t$, and so G has a subgraph H with $\delta(H) \geq t-1$ (greedy coloring). Then H contains T as a subgraph. $\square$

Theorem 11.14 (Chvatál, Rödl, Szemerédi and Trotter 1983)

For every positive integer Δ there is a constant c such that $R(H) \leq c|H|$ for all graphs H with $\Delta(H) \leq \Delta$.

Theorem 11.14 (Chvatál, Rödl, Szemerédi and Trotter 1983)

For every positive integer Δ there is a constant c such that $R(H) \leq c|H|$ for all graphs H with $\Delta(H) \leq \Delta$.

Proof omitted—uses the Regularity Lemma.

Ramsey Graphs

Ramsey's Theorem can be restated as follows: For every graph $H = K_r$ there is a graph G such that every 2-coloring of the edges of G gives an induced monochromatic subgraph isomorphic to H.

Ramsey Graphs

Ramsey's Theorem can be restated as follows: For every graph $H = K_r$ there is a graph G such that every 2-coloring of the edges of G gives an induced monochromatic subgraph isomorphic to H. (In fact, a sufficiently large complete graph will work for G.)

Ramsey Graphs

Ramsey's Theorem can be restated as follows: For every graph $H = K_r$ there is a graph G such that every 2-coloring of the edges of G gives an induced monochromatic subgraph isomorphic to H. (In fact, a sufficiently large complete graph will work for G.)
Question: Given an arbitrary graph H, is there a graph G such that every 2-coloring of G gives an induced monochromatic subgraph isomorphic to H?

Ramsey Graphs

Ramsey's Theorem can be restated as follows: For every graph $H = K_r$ there is a graph G such that every 2-coloring of the edges of G gives an induced monochromatic subgraph isomorphic to H. (In fact, a sufficiently large complete graph will work for G.)

Question: Given an arbitrary graph H, is there a graph G such that every 2-coloring of G gives an induced monochromatic subgraph isomorphic to H?

Theorem 11.15 (Deuber; Erdős, Hajnal, Pósa; Rödl 1973)

Every graph has a Ramsey graph.

Ramsey's Theorem can be restated as follows: For every graph $H = K_r$ there is a graph G such that every 2-coloring of the edges of G gives an induced monochromatic subgraph isomorphic to H. (In fact, a sufficiently large complete graph will work for G.)
Question: Given an arbitrary graph H, is there a graph G such that every 2-coloring of G gives an induced monochromatic subgraph isomorphic to H?

Theorem 11.15 (Deuber; Erdős, Hajnal, Pósa; Rödl 1973)

Every graph has a Ramsey graph. For every graph H there is a graph G such that, for every partition $\{E_1, E_2\}$ of $E(G)$, has an induced subgraph H with $E(H) \subseteq E_1$ or $E(H) \subseteq E_2$.

Proof

Given two graphs $G = (V, E)$ and H, and $U \subseteq V$, we write $G[U \to H]$ to denote the graph obtained from G by replacing each vertex u in U by a copy $H(u)$ of H,

Proof

Given two graphs $G = (V, E)$ and H, and $U \subseteq V$, we write $G[U \to H]$ to denote the graph obtained from G by replacing each vertex u in U by a copy $H(u)$ of H, joining $H(u)$ completely to $H(u')$ whenever $uu' \in E$,

Proof

Given two graphs $G = (V, E)$ and H, and $U \subseteq V$, we write $G[U \to H]$ to denote the graph obtained from G by replacing each vertex u in U by a copy $H(u)$ of H, joining $H(u)$ completely to $H(u')$ whenever $uu' \in E$, and joining each $H(u)$ to v whenever $uv \in E$ and $v \in V \setminus U$.

Proof

Given two graphs $G = (V, E)$ and H, and $U \subseteq V$, we write $G[U \to H]$ to denote the graph obtained from G by replacing each vertex u in U by a copy $H(u)$ of H, joining $H(u)$ completely to $H(u')$ whenever $uu' \in E$, and joining each $H(u)$ to v whenever $uv \in E$ and $v \in V \setminus U$. We will prove the following strengthening of the theorem

(*) For any two graphs H_1 and H_2, there is a graph $G = G(H_1, H_2)$ such that every edge-coloring of G with colors 1 and 2 yields either an induced $H_1 \subseteq G$ with all edges colored 1, or an induced $H_2 \subseteq G$ with all edges colored 2.

Proof

Given two graphs $G = (V, E)$ and H, and $U \subseteq V$, we write $G[U \to H]$ to denote the graph obtained from G by replacing each vertex u in U by a copy $H(u)$ of H, joining $H(u)$ completely to $H(u')$ whenever $uu' \in E$, and joining each $H(u)$ to v whenever $uv \in E$ and $v \in V \setminus U$. We will prove the following strengthening of the theorem
(*) For any two graphs H_1 and H_2, there is a graph $G = G(H_1, H_2)$ such that every edge-coloring of G with colors 1 and 2 yields either an induced $H_1 \subseteq G$ with all edges colored 1, or an induced $H_2 \subseteq G$ with all edges colored 2.
We proceed by induction on $|H_1| + |H_2|$.

Proof

Given two graphs $G = (V, E)$ and H, and $U \subseteq V$, we write $G[U \to H]$ to denote the graph obtained from G by replacing each vertex u in U by a copy $H(u)$ of H, joining $H(u)$ completely to $H(u')$ whenever $uu' \in E$, and joining each $H(u)$ to v whenever $uv \in E$ and $v \in V \setminus U$. We will prove the following strengthening of the theorem

(*) For any two graphs H_1 and H_2, there is a graph $G = G(H_1, H_2)$ such that every edge-coloring of G with colors 1 and 2 yields either an induced $H_1 \subseteq G$ with all edges colored 1, or an induced $H_2 \subseteq G$ with all edges colored 2.

We proceed by induction on $|H_1| + |H_2|$. If either H_1 or H_2 has no edges, in particular, $|H_1| + |H_2| \leq 1$, then (*) holds with $G = \overline{K_n}$ for sufficiently large n.

Proof

Given two graphs $G = (V, E)$ and H, and $U \subseteq V$, we write $G[U \to H]$ to denote the graph obtained from G by replacing each vertex u in U by a copy $H(u)$ of H, joining $H(u)$ completely to $H(u')$ whenever $uu' \in E$, and joining each $H(u)$ to v whenever $uv \in E$ and $v \in V \setminus U$. We will prove the following strengthening of the theorem

(*) For any two graphs H_1 and H_2, there is a graph $G = G(H_1, H_2)$ such that every edge-coloring of G with colors 1 and 2 yields either an induced $H_1 \subseteq G$ with all edges colored 1, or an induced $H_2 \subseteq G$ with all edges colored 2.

We proceed by induction on $|H_1| + |H_2|$. If either H_1 or H_2 has no edges, in particular, $|H_1| + |H_2| \leq 1$, then (*) holds with $G = \overline{K_n}$ for sufficiently large n. For the induction step, assume that both H_1 and H_2 have at least one edge, and that (*) holds for all pairs (H_1', H_2') with smaller $|H_1'| + |H_2'|$.

For each $i \in \{1, 2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H_i' = H_i - x_i$, and let H_i'' be the subgraph of H_i induced by the neighbors of x_i.

For each $i \in \{1, 2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H_i' = H_i - x_i$, and let H_i'' be the subgraph of H_i induced by the neighbors of x_i. We will construct a sequence G^0, G^1, ..., G^n of disjoint graphs with G^n being the desired Ramsey graph $G(H_1, H_2)$.

For each $i \in \{1, 2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H_i' = H_i - x_i$, and let H_i'' be the subgraph of H_i induced by the neighbors of x_i. We will construct a sequence G^0, G^1, ..., G^n of disjoint graphs with G^n being the desired Ramsey graph $G(H_1, H_2)$. We will also define subsets $V_i \subseteq V(G^i)$ and a map $f : V^1 \cup V^2 \cup \cdots \cup V^n \to V^0 \cup V^1 \cup \cdots \cup V^{n-1}$ such that $f(V^i) = V^{i-1}$ for all $i \geq 1$.

For each $i \in \{1, 2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H_i' = H_i - x_i$, and let H_i'' be the subgraph of H_i induced by the neighbors of x_i. We will construct a sequence G^0, G^1, ..., G^n of disjoint graphs with G^n being the desired Ramsey graph $G(H_1, H_2)$. We will also define subsets $V_i \subseteq V(G^i)$ and a map $f : V^1 \cup V^2 \cup \cdots \cup V^n \to V^0 \cup V^1 \cup \cdots \cup V^{n-1}$ such that $f(V^i) = V^{i-1}$ for all $i \geq 1$. We will also write f^i for composing f with itself i times, with f^0 understood as the identity map on $V(G^0)$.

For each $i \in \{1, 2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H_i' = H_i - x_i$, and let H_i'' be the subgraph of H_i induced by the neighbors of x_i. We will construct a sequence G^0, G^1, ..., G^n of disjoint graphs with G^n being the desired Ramsey graph $G(H_1, H_2)$. We will also define subsets $V_i \subseteq V(G^i)$ and a map $f : V^1 \cup V^2 \cup \cdots \cup V^n \to V^0 \cup V^1 \cup \cdots \cup V^{n-1}$ such that $f(V^i) = V^{i-1}$ for all $i \geq 1$. We will also write f^i for composing f with itself i times, with f^0 understood as the identity map on $V(G^0)$. Note that $f^i(v) \in V^0$ for all $v \in V^i$; we call $f^i(v)$ the origin of v.

For each $i \in \{1, 2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H_i' = H_i - x_i$, and let H_i'' be the subgraph of H_i induced by the neighbors of x_i. We will construct a sequence G^0, G^1, ..., G^n of disjoint graphs with G^n being the desired Ramsey graph $G(H_1, H_2)$. We will also define subsets $V_i \subseteq V(G^i)$ and a map $f : V^1 \cup V^2 \cup \cdots \cup V^n \to V^0 \cup V^1 \cup \cdots \cup V^{n-1}$ such that $f(V^i) = V^{i-1}$ for all $i \geq 1$. We will also write f^i for composing f with itself i times, with f^0 understood as the identity map on $V(G^0)$. Note that $f^i(v) \in V^0$ for all $v \in V^i$; we call $f^i(v)$ the origin of v.
Vertices in V^i with different origins are adjacent in G^i if and only if their origins are adjacent in G^0.

For each $i \in \{1, 2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H'_i = H_i - x_i$, and let H''_i be the subgraph of H_i induced by the neighbors of x_i. We will construct a sequence G^0, G^1, ..., G^n of disjoint graphs with G^n being the desired Ramsey graph $G(H_1, H_2)$. We will also define subsets $V_i \subseteq V(G^i)$ and a map $f : V^1 \cup V^2 \cup \cdots \cup V^n \to V^0 \cup V^1 \cup \cdots \cup V^{n-1}$ such that $f(V^i) = V^{i-1}$ for all $i \geq 1$. We will also write f^i for composing f with itself i times, with f^0 understood as the identity map on $V(G^0)$. Note that $f^i(v) \in V^0$ for all $v \in V^i$; we call $f^i(v)$ the origin of v.

Vertices in V^i with different origins are adjacent in G^i if and only if their origins are adjacent in G^0.

By the induction hypothesis, there are Ramsey graphs $G_1 = G(H_1, H'_2)$ and $G_2 = G(H'_1, H_2)$.

For each $i \in \{1, 2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H_i' = H_i - x_i$, and let H_i'' be the subgraph of H_i induced by the neighbors of x_i. We will construct a sequence G^0, G^1, ..., G^n of disjoint graphs with G^n being the desired Ramsey graph $G(H_1, H_2)$. We will also define subsets $V_i \subseteq V(G^i)$ and a map $f : V^1 \cup V^2 \cup \cdots \cup V^n \to V^0 \cup V^1 \cup \cdots \cup V^{n-1}$ such that $f(V^i) = V^{i-1}$ for all $i \geq 1$. We will also write f^i for composing f with itself i times, with f^0 understood as the identity map on $V(G^0)$. Note that $f^i(v) \in V^0$ for all $v \in V^i$; we call $f^i(v)$ the origin of v.

Vertices in V^i with different origins are adjacent in G^i if and only if their origins are adjacent in G^0.

By the induction hypothesis, there are Ramsey graphs $G_1 = G(H_1, H_2')$ and $G_2 = G(H_1', H_2)$. Let G^0 be a copy of G_1, and let $V^0 = V(G^0)$.

For each $i \in \{1, 2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H_i' = H_i - x_i$, and let H_i'' be the subgraph of H_i induced by the neighbors of x_i. We will construct a sequence G^0, G^1, ..., G^n of disjoint graphs with G^n being the desired Ramsey graph $G(H_1, H_2)$. We will also define subsets $V_i \subseteq V(G^i)$ and a map $f : V^1 \cup V^2 \cup \cdots \cup V^n \to V^0 \cup V^1 \cup \cdots \cup V^{n-1}$ such that $f(V^i) = V^{i-1}$ for all $i \geq 1$. We will also write f^i for composing f with itself i times, with f^0 understood as the identity map on $V(G^0)$. Note that $f^i(v) \in V^0$ for all $v \in V^i$; we call $f^i(v)$ the origin of v.

Vertices in V^i with different origins are adjacent in G^i if and only if their origins are adjacent in G^0.

By the induction hypothesis, there are Ramsey graphs $G_1 = G(H_1, H_2')$ and $G_2 = G(H_1', H_2)$. Let G^0 be a copy of G_1, and let $V^0 = V(G^0)$. Let W_0', W_1', ..., W_{n-1}' be the subsets of V^0 spanning an H_2' in G^0.

For each $i \in \{1, 2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H_i' = H_i - x_i$, and let H_i'' be the subgraph of H_i induced by the neighbors of x_i. We will construct a sequence G^0, G^1, ..., G^n of disjoint graphs with G^n being the desired Ramsey graph $G(H_1, H_2)$. We will also define subsets $V_i \subseteq V(G^i)$ and a map $f : V^1 \cup V^2 \cup \cdots \cup V^n \to V^0 \cup V^1 \cup \cdots \cup V^{n-1}$ such that $f(V^i) = V^{i-1}$ for all $i \geq 1$. We will also write f^i for composing f with itself i times, with f^0 understood as the identity map on $V(G^0)$. Note that $f^i(v) \in V^0$ for all $v \in V^i$; we call $f^i(v)$ the origin of v.
Vertices in V^i with different origins are adjacent in G^i if and only if their origins are adjacent in G^0.
By the induction hypothesis, there are Ramsey graphs $G_1 = G(H_1, H_2')$ and $G_2 = G(H_1', H_2)$. Let G^0 be a copy of G_1, and let $V^0 = V(G^0)$. Let W_0', W_1', ..., W_{n-1}' be the subsets of V^0 spanning an H_2' in G^0. Thus n is defined as the number of induced copies of H_2' in G^0.

For each $i \in \{1,2\}$, pick a vertex $x_i \in H_i$ that is incident with an edge, let $H_i' = H_i - x_i$, and let H_i'' be the subgraph of H_i induced by the neighbors of x_i. We will construct a sequence G^0, G^1, ..., G^n of disjoint graphs with G^n being the desired Ramsey graph $G(H_1, H_2)$. We will also define subsets $V_i \subseteq V(G^i)$ and a map $f : V^1 \cup V^2 \cup \cdots \cup V^n \to V^0 \cup V^1 \cup \cdots \cup V^{n-1}$ such that $f(V^i) = V^{i-1}$ for all $i \geq 1$. We will also write f^i for composing f with itself i times, with f^0 understood as the identity map on $V(G^0)$. Note that $f^i(v) \in V^0$ for all $v \in V^i$; we call $f^i(v)$ the origin of v.

Vertices in V^i with different origins are adjacent in G^i if and only if their origins are adjacent in G^0.

By the induction hypothesis, there are Ramsey graphs $G_1 = G(H_1, H_2')$ and $G_2 = G(H_1', H_2)$. Let G^0 be a copy of G_1, and let $V^0 = V(G^0)$. Let W_0', W_1', ..., W_{n-1}' be the subsets of V^0 spanning an H_2' in G^0. Thus n is defined as the number of induced copies of H_2' in G^0. For $i \in \{0, 1, \ldots, n-1\}$, let W_i'' be the image of $V(H_2'')$ under some isomorphism $H_2' \to G^0[W_i']$.

Assume now that G^0, G^1, ..., G^{i-1} and V^0, V^1, ..., V^{i-1} have been defined for $i \geq 1$ and that f has been defined on $V^1 \cup \ldots \cup V^{i-1}$ as described above.

Assume now that G^0, G^1, ..., G^{i-1} and V^0, V^1, ..., V^{i-1} have been defined for $i \geq 1$ and that f has been defined on $V^1 \cup \ldots \cup V^{i-1}$ as described above. We construct G^i from G^{i-1} in two steps.

Assume now that G^0, G^1, ..., G^{i-1} and V^0, V^1, ..., V^{i-1} have been defined for $i \geq 1$ and that f has been defined on $V^1 \cup \ldots \cup V^{i-1}$ as described above. We construct G^i from G^{i-1} in two steps. For the first step, consider the set U^i of all the vertices $v \in V^{i-1}$ whose origin $f^{i-1}(v)$ lies in W''_{i-1}. ($U^0 = W''_0$)

Assume now that G^0, G^1, ..., G^{i-1} and V^0, V^1, ..., V^{i-1} have been defined for $i \geq 1$ and that f has been defined on $V^1 \cup \ldots \cup V^{i-1}$ as described above. We construct G^i from G^{i-1} in two steps. For the first step, consider the set U^i of all the vertices $v \in V^{i-1}$ whose origin $f^{i-1}(v)$ lies in W''_{i-1}. ($U^0 = W''_0$) Expand G^{i-1} to a new graph $\tilde{G}^{i-1}$ by replacing every vertex $u \in U^{i-1}$ with a copy of $G_2(u)$ of G_2, that is, let $\tilde{G}^{i-1} = G^{i-1}[U^{i-1} \to G_2]$.

Assume now that G^0, G^1, ..., G^{i-1} and V^0, V^1, ..., V^{i-1} have been defined for $i \geq 1$ and that f has been defined on $V^1 \cup \ldots \cup V^{i-1}$ as described above. We construct G^i from G^{i-1} in two steps. For the first step, consider the set U^i of all the vertices $v \in V^{i-1}$ whose origin $f^{i-1}(v)$ lies in W''_{i-1}. ($U^0 = W''_0$) Expand G^{i-1} to a new graph $\tilde{G}^{i-1}$ by replacing every vertex $u \in U^{i-1}$ with a copy of $G_2(u)$ of G_2, that is, let $\tilde{G}^{i-1} = G^{i-1}[U^{i-1} \rightarrow G_2]$. Set $f(u') = u$ for all $u \in U^{i-1}$ and $u' \in G_2(u)$, and $f(v') = v$ for all $v' = (v, \emptyset)$ with $v \in V^{i-1} \setminus U^{i-1}$. ($(v, \emptyset)$ is the unexpanded copy of a vertex $v \in G^{i-1}$ in $\tilde{G}^{i-1}$)

Assume now that G^0, G^1, ..., G^{i-1} and V^0, V^1, ..., V^{i-1} have been defined for $i \geq 1$ and that f has been defined on $V^1 \cup \ldots \cup V^{i-1}$ as described above. We construct G^i from G^{i-1} in two steps. For the first step, consider the set U^i of all the vertices $v \in V^{i-1}$ whose origin $f^{i-1}(v)$ lies in W''_{i-1}. ($U^0 = W''_0$) Expand G^{i-1} to a new graph $\tilde{G}^{i-1}$ by replacing every vertex $u \in U^{i-1}$ with a copy of $G_2(u)$ of G_2, that is, let $\tilde{G}^{i-1} = G^{i-1}[U^{i-1} \to G_2]$. Set $f(u') = u$ for all $u \in U^{i-1}$ and $u' \in G_2(u)$, and $f(v') = v$ for all $v' = (v, \emptyset)$ with $v \in V^{i-1} \setminus U^{i-1}$. ($(v, \emptyset)$ is the unexpanded copy of a vertex $v \in G^{i-1}$ in $\tilde{G}^{i-1}$) For the second step, let $\mathcal{F}$ denote the set of all families F of the form $F = (H'(u) | u \in U^{i-1})$, where each $H'_1(u)$ is an induced subgraph of $G_2(u)$.

Assume now that G^0, G^1, ..., G^{i-1} and V^0, V^1, ..., V^{i-1} have been defined for $i \geq 1$ and that f has been defined on $V^1 \cup \ldots \cup V^{i-1}$ as described above. We construct G^i from G^{i-1} in two steps. For the first step, consider the set U^i of all the vertices $v \in V^{i-1}$ whose origin $f^{i-1}(v)$ lies in W''_{i-1}. ($U^0 = W''_0$) Expand G^{i-1} to a new graph $\tilde{G}^{i-1}$ by replacing every vertex $u \in U^{i-1}$ with a copy of $G_2(u)$ of G_2, that is, let $\tilde{G}^{i-1} = G^{i-1}[U^{i-1} \to G_2]$. Set $f(u') = u$ for all $u \in U^{i-1}$ and $u' \in G_2(u)$, and $f(v') = v$ for all $v' = (v, \emptyset)$ with $v \in V^{i-1} \setminus U^{i-1}$. ($(v, \emptyset)$ is the unexpanded copy of a vertex $v \in G^{i-1}$ in $\tilde{G}^{i-1}$) For the second step, let $\mathcal{F}$ denote the set of all families F of the form $F = (H'(u) | u \in U^{i-1})$, where each $H'_1(u)$ is an induced subgraph of $G_2(u)$. For each F in $\mathcal{F}$, add new vertex $x(F)$ to $\tilde{G}^{i-1}$ and join it, for every $u \in U^{i-1}$, to all the vertices in the image $H''_1(u) \subseteq H'_1(u)$ under some isomorphism from H'_1 o the $H'_1(u) \subseteq G_2(u)$ selected by F.

Assume now that G^0, G^1, ..., G^{i-1} and V^0, V^1, ..., V^{i-1} have been defined for $i \geq 1$ and that f has been defined on $V^1 \cup \ldots \cup V^{i-1}$ as described above. We construct G^i from G^{i-1} in two steps. For the first step, consider the set U^i of all the vertices $v \in V^{i-1}$ whose origin $f^{i-1}(v)$ lies in W''_{i-1}. ($U^0 = W''_0$) Expand G^{i-1} to a new graph $\tilde{G}^{i-1}$ by replacing every vertex $u \in U^{i-1}$ with a copy of $G_2(u)$ of G_2, that is, let $\tilde{G}^{i-1} = G^{i-1}[U^{i-1} \to G_2]$. Set $f(u') = u$ for all $u \in U^{i-1}$ and $u' \in G_2(u)$, and $f(v') = v$ for all $v' = (v, \emptyset)$ with $v \in V^{i-1} \setminus U^{i-1}$. ($(v, \emptyset)$ is the unexpanded copy of a vertex $v \in G^{i-1}$ in $\tilde{G}^{i-1}$) For the second step, let $\mathcal{F}$ denote the set of all families F of the form $F = (H'(u) | u \in U^{i-1})$, where each $H'_1(u)$ is an induced subgraph of $G_2(u)$. For each F in $\mathcal{F}$, add new vertex $x(F)$ to $\tilde{G}^{i-1}$ and join it, for every $u \in U^{i-1}$, to all the vertices in the image $H''_1(u) \subseteq H'_1(u)$ under some isomorphism from H'_1 o the $H'_1(u) \subseteq G_2(u)$ selected by F. Denote the resulting graph by G^i.

Now we show that G^n satisfies (*).

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1,

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2,

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2, or an induced graph colored 2 such that $V(H) \subseteq V^i$ and the restriction of f^i to $V(H)$ is an isomorphism between H and $G^0[W_k']$ for some $k \in \{i, i+1, \ldots, n-1\}$.

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2, or an induced graph colored 2 such that $V(H) \subseteq V^i$ and the restriction of f^i to $V(H)$ is an isomorphism between H and $G^0[W'_k]$ for some $k \in \{i, i+1, \ldots, n-1\}$.
For $i = 0$, (**) follows from the choice of G^0 as a copy of $G_1 = G(H_1, H'_2)$.

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2, or an induced graph colored 2 such that $V(H) \subseteq V^i$ and the restriction of f^i to $V(H)$ is an isomorphism between H and $G^0[W'_k]$ for some $k \in \{i, i+1, \ldots, n-1\}$.
For $i = 0$, (**) follows from the choice of G^0 as a copy of $G_1 = G(H_1, H'_2)$.

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2, or an induced graph colored 2 such that $V(H) \subseteq V^i$ and the restriction of f^i to $V(H)$ is an isomorphism between H and $G^0[W'_k]$ for some $k \in \{i, i+1, \ldots, n-1\}$.
For $i = 0$, (**) follows from the choice of G^0 as a copy of $G_1 = G(H_1, H'_2)$.
Now let $1 \leq i \leq n$, and assume that (**) holds for all smaller values of i.

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2, or an induced graph colored 2 such that $V(H) \subseteq V^i$ and the restriction of f^i to $V(H)$ is an isomorphism between H and $G^0[W_k']$ for some $k \in \{i, i+1, \ldots, n-1\}$.
For $i = 0$, (**) follows from the choice of G^0 as a copy of $G_1 = G(H_1, H_2')$.
Now let $1 \leq i \leq n$, and assume that (**) holds for all smaller values of i.
Let an edge coloring of G^i be given.

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2, or an induced graph colored 2 such that $V(H) \subseteq V^i$ and the restriction of f^i to $V(H)$ is an isomorphism between H and $G^0[W'_k]$ for some $k \in \{i, i+1, \dots, n-1\}$.
For $i = 0$, (**) follows from the choice of G^0 as a copy of $G_1 = G(H_1, H'_2)$.
Now let $1 \leq i \leq n$, and assume that (**) holds for all smaller values of i.
Let an edge coloring of G^i be given. For each $u \in U^{i-1}$ there is a copy of G_2 in G^i:

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2, or an induced graph colored 2 such that $V(H) \subseteq V^i$ and the restriction of f^i to $V(H)$ is an isomorphism between H and $G^0[W'_k]$ for some $k \in \{i, i+1, \dots, n-1\}$.
For $i = 0$, (**) follows from the choice of G^0 as a copy of $G_1 = G(H_1, H'_2)$.
Now let $1 \le i \le n$, and assume that (**) holds for all smaller values of i.
Let an edge coloring of G^i be given. For each $u \in U^{i-1}$ there is a copy of G_2 in G^i:

$$G^i \supseteq G_2(u) \cong G(H'_1, H_2).$$

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2, or an induced graph colored 2 such that $V(H) \subseteq V^i$ and the restriction of f^i to $V(H)$ is an isomorphism between H and $G^0[W'_k]$ for some $k \in \{i, i+1, \ldots, n-1\}$.
For $i = 0$, (**) follows from the choice of G^0 as a copy of $G_1 = G(H_1, H'_2)$.
Now let $1 \leq i \leq n$, and assume that (**) holds for all smaller values of i.
Let an edge coloring of G^i be given. For each $u \in U^{i-1}$ there is a copy of G_2 in G^i:

$$G^i \supseteq G_2(u) \cong G(H'_1, H_2).$$

If some $G_2(u)$ contains an induced H_2 colored 2, then the conclusion holds.

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2, or an induced graph colored 2 such that $V(H) \subseteq V^i$ and the restriction of f^i to $V(H)$ is an isomorphism between H and $G^0[W'_k]$ for some $k \in \{i, i+1, \ldots, n-1\}$.
For $i = 0$, (**) follows from the choice of G^0 as a copy of $G_1 = G(H_1, H'_2)$.
Now let $1 \le i \le n$, and assume that (**) holds for all smaller values of i.
Let an edge coloring of G^i be given. For each $u \in U^{i-1}$ there is a copy of G_2 in G^i:

$$G^i \supseteq G_2(u) \cong G(H'_1, H_2).$$

If some $G_2(u)$ contains an induced H_2 colored 2, then the conclusion holds. If not, then every $G_2(u)$ has an induced subgraph $H'_1(u) \cong H'_1$ colored 1.

Now we show that G^n satisfies (*). We prove the following:
(**) For every edge coloring with colors 1 and 2, the graph G^i contains either and induced H_1 colored 1, or an induced H_2 colored 2, or an induced graph colored 2 such that $V(H) \subseteq V^i$ and the restriction of f^i to $V(H)$ is an isomorphism between H and $G^0[W'_k]$ for some $k \in \{i, i+1, \dots, n-1\}$.
For $i = 0$, (**) follows from the choice of G^0 as a copy of $G_1 = G(H_1, H'_2)$.
Now let $1 \leq i \leq n$, and assume that (**) holds for all smaller values of i.
Let an edge coloring of G^i be given. For each $u \in U^{i-1}$ there is a copy of G_2 in G^i:

$$G^i \supseteq G_2(u) \cong G(H'_1, H_2).$$

If some $G_2(u)$ contains an induced H_2 colored 2, then the conclusion holds. If not, then every $G_2(u)$ has an induced subgraph $H'_1(u) \cong H'_1$ colored 1. Let F be the family of these graphs $H'_1(u)$, one for each $u \in U^{i-1}$ and let $x = x(F)$.

If, for some $u \in U^{i-1}$, all the $x - H_1''(u)$ edges in G^i are also colored 1, then we have an induced copy of H_1 in G^i and again the conclusion holds.

If, for some $u \in U^{i-1}$, all the $x - H_1''(u)$ edges in G^i are also colored 1, then we have an induced copy of H_1 in G^i and again the conclusion holds. So we may assume that each $H_1''(u)$ has a vertex y_u for which the edge xy_u is colored 2.

If, for some $u \in U^{i-1}$, all the $x - H_1''(u)$ edges in G^i are also colored 1, then we have an induced copy of H_1 in G^i and again the conclusion holds. So we may assume that each $H_1''(u)$ has a vertex y_u for which the edge xy_u is colored 2. The restriction $y_u \mapsto u$ of f to $\hat{U}^{i-1} = \{y_u | u \in U^{i-1}\} \subseteq V^i$ extends by $(v, \emptyset) \mapsto v$ to an isomorphism from

$$\hat{G}^{i-1} = G^i[\hat{U}^{i-1} \cup \{(v, \emptyset) | v \in V(G^{i-1}) \setminus U^{i-1}\}]$$

to G^{i-1},

If, for some $u \in U^{i-1}$, all the $x - H_1''(u)$ edges in G^i are also colored 1, then we have an induced copy of H_1 in G^i and again the conclusion holds. So we may assume that each $H_1''(u)$ has a vertex y_u for which the edge xy_u is colored 2. The restriction $y_u \mapsto u$ of f to $\hat{U}^{i-1} = \{y_u | u \in U^{i-1}\} \subseteq V^i$ extends by $(v, \emptyset) \mapsto v$ to an isomorphism from

$$\hat{G}^{i-1} = G^i[\hat{U}^{i-1} \cup \{(v, \emptyset) | v \in V(G^{i-1}) \setminus U^{i-1}\}]$$

to G^{i-1}, and so our edge-coloring of G^i induces an edge-coloring of G^{i-1}.

If, for some $u \in U^{i-1}$, all the $x - H_1''(u)$ edges in G^i are also colored 1, then we have an induced copy of H_1 in G^i and again the conclusion holds. So we may assume that each $H_1''(u)$ has a vertex y_u for which the edge xy_u is colored 2. The restriction $y_u \mapsto u$ of f to $\hat{U}^{i-1} = \{y_u | u \in U^{i-1}\} \subseteq V^i$ extends by $(v, \emptyset) \mapsto v$ to an isomorphism from

$$\hat{G}^{i-1} = G^i[\hat{U}^{i-1} \cup \{(v, \emptyset) | v \in V(G^{i-1}) \setminus U^{i-1}\}]$$

to G^{i-1}, and so our edge-coloring of G^i induces an edge-coloring of G^{i-1}. If this coloring gives an induced $H_1 \subseteq G^{i-1}$ colored 1, or an induced $H_2 \subseteq G^{i-1}$ colored 2, we have these also in $\hat{G}^{i-1} \subseteq G^i$ and again the conclusion holds.

If, for some $u \in U^{i-1}$, all the $x - H_1''(u)$ edges in G^i are also colored 1, then we have an induced copy of H_1 in G^i and again the conclusion holds. So we may assume that each $H_1''(u)$ has a vertex y_u for which the edge xy_u is colored 2. The restriction $y_u \mapsto u$ of f to $\hat{U}^{i-1} = \{y_u | u \in U^{i-1}\} \subseteq V^i$ extends by $(v, \emptyset) \mapsto v$ to an isomorphism from

$$\hat{G}^{i-1} = G^i[\hat{U}^{i-1} \cup \{(v, \emptyset) | v \in V(G^{i-1}) \setminus U^{i-1}\}]$$

to G^{i-1}, and so our edge-coloring of G^i induces an edge-coloring of G^{i-1}. If this coloring gives an induced $H_1 \subseteq G^{i-1}$ colored 1, or an induced $H_2 \subseteq G^{i-1}$ colored 2, we have these also in $\hat{G}^{i-1} \subseteq G^i$ and again the conclusion holds.
By (**) for $i-1$ we may then assume that G^{i-1} has an induced H' colored 2 with $V(H') \subseteq V^{i-1}$ and such that the restriction of f^{i-1} to $V(H')$ is an isomorphism from H' to $G^0[W_k'] \cong H_2'$ for some $k \in \{i-1, \ldots, n-1\}$.

If, for some $u \in U^{i-1}$, all the $x - H_1''(u)$ edges in G^i are also colored 1, then we have an induced copy of H_1 in G^i and again the conclusion holds. So we may assume that each $H_1''(u)$ has a vertex y_u for which the edge xy_u is colored 2. The restriction $y_u \mapsto u$ of f to $\hat{U}^{i-1} = \{y_u | u \in U^{i-1}\} \subseteq V^i$ extends by $(v, \emptyset) \mapsto v$ to an isomorphism from

$$\hat{G}^{i-1} = G^i[\hat{U}^{i-1} \cup \{(v, \emptyset) | v \in V(G^{i-1}) \setminus U^{i-1}\}]$$

to G^{i-1}, and so our edge-coloring of G^i induces an edge-coloring of G^{i-1}. If this coloring gives an induced $H_1 \subseteq G^{i-1}$ colored 1, or an induced $H_2 \subseteq G^{i-1}$ colored 2, we have these also in $\hat{G}^{i-1} \subseteq G^i$ and again the conclusion holds.
By (**) for $i-1$ we may then assume that G^{i-1} has an induced H' colored 2 with $V(H') \subseteq V^{i-1}$ and such that the restriction of f^{i-1} to $V(H')$ is an isomorphism from H' to $G^0[W_k'] \cong H_2'$ for some $k \in \{i-1, \ldots, n-1\}$. Let $\hat{H}'$ be the corresponding induced subgraph of $\hat{G}^{i-1} \subseteq G^i$ (also colored 2).

If, for some $u \in U^{i-1}$, all the $x - H_1''(u)$ edges in G^i are also colored 1, then we have an induced copy of H_1 in G^i and again the conclusion holds. So we may assume that each $H_1''(u)$ has a vertex y_u for which the edge xy_u is colored 2. The restriction $y_u \mapsto u$ of f to $\hat{U}^{i-1} = \{y_u | u \in U^{i-1}\} \subseteq V^i$ extends by $(v, \emptyset) \mapsto v$ to an isomorphism from

$$\hat{G}^{i-1} = G^i[\hat{U}^{i-1} \cup \{(v, \emptyset) | v \in V(G^{i-1}) \setminus U^{i-1}\}]$$

to G^{i-1}, and so our edge-coloring of G^i induces an edge-coloring of G^{i-1}. If this coloring gives an induced $H_1 \subseteq G^{i-1}$ colored 1, or an induced $H_2 \subseteq G^{i-1}$ colored 2, we have these also in $\hat{G}^{i-1} \subseteq G^i$ and again the conclusion holds.

By (**) for $i-1$ we may then assume that G^{i-1} has an induced H' colored 2 with $V(H') \subseteq V^{i-1}$ and such that the restriction of f^{i-1} to $V(H')$ is an isomorphism from H' to $G^0[W_k'] \cong H_2'$ for some $k \in \{i-1, \ldots, n-1\}$. Let $\hat{H}'$ be the corresponding induced subgraph of $\hat{G}^{i-1} \subseteq G^i$ (also colored 2). then $V(\hat{H}') \subseteq V_i$,

$$f^i(V(\hat{H}')) = f^{i-1}(V(H')) = W_k',$$

and $F^i : \hat{H}' \to G^0[W_k']$ is an isomorphism.

If, for some $u \in U^{i-1}$, all the $x - H_1''(u)$ edges in G^i are also colored 1, then we have an induced copy of H_1 in G^i and again the conclusion holds. So we may assume that each $H_1''(u)$ has a vertex y_u for which the edge xy_u is colored 2. The restriction $y_u \mapsto u$ of f to $\hat{U}^{i-1} = \{y_u | u \in U^{i-1}\} \subseteq V^i$ extends by $(v, \emptyset) \mapsto v$ to an isomorphism from

$$\hat{G}^{i-1} = G^i[\hat{U}^{i-1} \cup \{(v, \emptyset) | v \in V(G^{i-1}) \setminus U^{i-1}\}]$$

to G^{i-1}, and so our edge-coloring of G^i induces an edge-coloring of G^{i-1}. If this coloring gives an induced $H_1 \subseteq G^{i-1}$ colored 1, or an induced $H_2 \subseteq G^{i-1}$ colored 2, we have these also in $\hat{G}^{i-1} \subseteq G^i$ and again the conclusion holds.

By (**) for $i-1$ we may then assume that G^{i-1} has an induced H' colored 2 with $V(H') \subseteq V^{i-1}$ and such that the restriction of f^{i-1} to $V(H')$ is an isomorphism from H' to $G^0[W_k'] \cong H_2'$ for some $k \in \{i-1, \dots, n-1\}$. Let $\hat{H}'$ be the corresponding induced subgraph of $\hat{G}^{i-1} \subseteq G^i$ (also colored 2). then $V(\hat{H}') \subseteq V_i$,

$$f^i(V(\hat{H}')) = f^{i-1}(V(H')) = W_k',$$

and $F^i : \hat{H}' \to G^0[W_k']$ is an isomorphism.

If $k \geq i$, then the proof of (**) is complete with $H = \hat{H}'$.

We thus assume that $k < i$, and so $k = i - 1$.

We thus assume that $k < i$, and so $k = i - 1$. By definition of U^{i-1} and $\hat{G}^{i-1}$, the inverse image of W''_{i-1} under isomorphism $f^i : \hat{H}' \to G^0[W'_{i-1}]$ is a subset of U^{i-1}.

We thus assume that $k < i$, and so $k = i - 1$. By definition of U^{i-1} and $\hat{G}^{i-1}$, the inverse image of W''_{i-1} under isomorphism $f^i : \hat{H}' \to G^0[W'_{i-1}]$ is a subset of U^{i-1}. Since x is adjacent to those vertices that lie in $\hat{U}^{i-1}$ and all those edges are colored 2, the graph $\hat{H}'$ and x together induce in G^i a copy of H_2 colored 2.

Problem 22

Prove that for every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G has a subdivision of C_k or $K_{2,k}$. Find an upper bound on N it terms of k.

Problem 23

Find a Ramsey graph for C_4, that is, find a graph G such that if the edges of G are partitioned into $\{E_1, E_2\}$, then G has a induced subgraph isomorphic to C_4 all of whose edges belong to one of E_1 or E_2.

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Theorem 11.17

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains C_k or $K_{2,k}$ as topological minors.

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Theorem 11.17

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains C_k or $K_{2,k}$ as topological minors.

Theorem 11.18

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains a topological minor

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Theorem 11.17

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains C_k or $K_{2,k}$ as topological minors.

Theorem 11.18

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains a topological minor W_k (wheel) or

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Theorem 11.17

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains C_k or $K_{2,k}$ as topological minors.

Theorem 11.18

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains a topological minor W_k (wheel) or V_k (wheel with center replaced by a path) or

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Theorem 11.17

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains C_k or $K_{2,k}$ as topological minors.

Theorem 11.18

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains a topological minor W_k (wheel) or V_k (wheel with center replaced by a path) or $K_{3,k}$.

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Theorem 11.17

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains C_k or $K_{2,k}$ as topological minors.

Theorem 11.18

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains a topological minor W_k (wheel) or V_k (wheel with center replaced by a path) or $K_{3,k}$.

Theorem 11.19

For every positive integer k there is an integer N such that if G is an 4-connected graph of order at least N, then G contains as a topological minor

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Theorem 11.17

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains C_k or $K_{2,k}$ as topological minors.

Theorem 11.18

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains a topological minor W_k (wheel) or V_k (wheel with center replaced by a path) or $K_{3,k}$.

Theorem 11.19

For every positive integer k there is an integer N such that if G is an 4-connected graph of order at least N, then G contains as a topological minor DW_k (double wheel) or

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Theorem 11.17

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains C_k or $K_{2,k}$ as topological minors.

Theorem 11.18

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains a topological minor W_k (wheel) or V_k (wheel with center replaced by a path) or $K_{3,k}$.

Theorem 11.19

For every positive integer k there is an integer N such that if G is an 4-connected graph of order at least N, then G contains as a topological minor DW_k (double wheel) or Z_k (zig-zag ladder) or

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Theorem 11.17

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains C_k or $K_{2,k}$ as topological minors.

Theorem 11.18

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains a topological minor W_k (wheel) or V_k (wheel with center replaced by a path) or $K_{3,k}$.

Theorem 11.19

For every positive integer k there is an integer N such that if G is an 4-connected graph of order at least N, then G contains as a topological minor DW_k (double wheel) or Z_k (zig-zag ladder) or M_k (Möbius ladder) or

Ramsey Theorem for Connected Graphs

Theorem 11.16

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains P_k or $K_{1,k}$ as a subgraph.

Theorem 11.17

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains C_k or $K_{2,k}$ as topological minors.

Theorem 11.18

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains a topological minor W_k (wheel) or V_k (wheel with center replaced by a path) or $K_{3,k}$.

Theorem 11.19

For every positive integer k there is an integer N such that if G is an 4-connected graph of order at least N, then G contains as a topological minor DW_k (double wheel) or Z_k (zig-zag ladder) or M_k (Möbius ladder) or $K_{4,k}$.

Parallel Minors

Recall that a graph is a topological minor of another if it can be obtained by

- deleting edges or isolated vertices
- contracting edges in series that are in series with another edge

Parallel Minors

Recall that a graph is a topological minor of another if it can be obtained by

- deleting edges or isolated vertices
- contracting edges in series that are in series with another edge

Topological minors are also called series minors

Parallel Minors

Recall that a graph is a topological minor of another if it can be obtained by

- deleting edges or isolated vertices
- contracting edges in series that are in series with another edge

Topological minors are also called series minors

Parallel Minors

Recall that a graph is a topological minor of another if it can be obtained by

- deleting edges or isolated vertices
- contracting edges in series that are in series with another edge

Topological minors are also called series minors
A graph is a parallel minor of another if it can be obtained by

- contracting edges
- deleting edges that are in parallel with other edges (simplifying)

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or K_k or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or K_k or W_k.

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or K_k or W_k.

Theorem 11.23

For every positive integer k there is an integer N such that if G is a 4-connected graph of order at least N, then G contains as a parallel minor

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or K_k or W_k.

Theorem 11.23

For every positive integer k there is an integer N such that if G is a 4-connected graph of order at least N, then G contains as a parallel minor $K'_{4,k}$ or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or K_k or W_k.

Theorem 11.23

For every positive integer k there is an integer N such that if G is a 4-connected graph of order at least N, then G contains as a parallel minor $K'_{4,k}$ or DW_k or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or K_k or W_k.

Theorem 11.23

For every positive integer k there is an integer N such that if G is a 4-connected graph of order at least N, then G contains as a parallel minor $K'_{4,k}$ or DW_k or DW'_k (double wheel with an axle) or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or K_k or W_k.

Theorem 11.23

For every positive integer k there is an integer N such that if G is a 4-connected graph of order at least N, then G contains as a parallel minor $K'_{4,k}$ or DW_k or DW'_k (double wheel with an axle) or TF_k (triple fan) or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or K_k or W_k.

Theorem 11.23

For every positive integer k there is an integer N such that if G is a 4-connected graph of order at least N, then G contains as a parallel minor $K'_{4,k}$ or DW_k or DW'_k (double wheel with an axle) or TF_k (triple fan) or Z_k or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or K_k or W_k.

Theorem 11.23

For every positive integer k there is an integer N such that if G is a 4-connected graph of order at least N, then G contains as a parallel minor $K'_{4,k}$ or DW_k or DW'_k (double wheel with an axle) or TF_k (triple fan) or Z_k or M_k or

Unavoidable Parallel Minors

Theorem 11.20

For every positive integer k there is an integer N such that if G is a connected graph of order at least N, then G contains as a parallel minor P_k or $K_{1,k}$ or K_k or C_k.

Theorem 11.21

For every positive integer k there is an integer N such that if G is a 2-connected graph of order at least N, then G contains as a parallel minor $K'_{2,k}$ ($K_{2,k}$ plus an edge) or F_k (fan) or K_k or C_k.

Theorem 11.22

For every positive integer k there is an integer N such that if G is a 3-connected graph of order at least N, then G contains as a parallel minor $K'_{3,k}$ ($K_{3,k}$ plus a triangle) or DF_k (double fan) or K_k or W_k.

Theorem 11.23

For every positive integer k there is an integer N such that if G is a 4-connected graph of order at least N, then G contains as a parallel minor $K'_{4,k}$ or DW_k or DW'_k (double wheel with an axle) or TF_k (triple fan) or Z_k or M_k or K_k.

Conjecture 11.24 (Chartrand, Geller, Hedetniemi)

Every planar graph is a union of two outerplanar graphs.

Partitioning Planar Graphs

Conjecture 11.24 (Chartrand, Geller, Hedetniemi)

Every planar graph is a union of two outerplanar graphs.

"Proof":

- WLOG G is a triangulation

Partitioning Planar Graphs

Conjecture 11.24 (Chartrand, Geller, Hedetniemi)

Every planar graph is a union of two outerplanar graphs.

"Proof":

- WLOG G is a triangulation
- G is 4-connected
  $\Rightarrow$ Hamilton cycle C
  $\Rightarrow$ ($C+$ edges inside C) $\cup$ ($C+$ edges outside C)

Partitioning Planar Graphs

Conjecture 11.24 (Chartrand, Geller, Hedetniemi)

Every planar graph is a union of two outerplanar graphs.

"Proof":

- WLOG G is a triangulation
- G is 4-connected
  $\Rightarrow$ Hamilton cycle C
  $\Rightarrow$ ($C+$ edges inside C) $\cup$ ($C+$ edges outside C)
- min counter-example G not 4-connected
  $\Rightarrow$ G is 0-, 1-, 2-, or 3-sum of A and B
  $\Rightarrow$ decompose each A and B and make parts "fit together"

Partitioning Planar Graphs

Conjecture 11.24 (Chartrand, Geller, Hedetniemi)

Every planar graph is a union of two outerplanar graphs.

~~“Proof”~~:

- WLOG G is a triangulation
- G is 4-connected
  $\Rightarrow$ Hamilton cycle C
  $\Rightarrow$ ($C+$ edges inside C) $\cup$ ($C+$ edges outside C)
- min counter-example G not 4-connected
  $\Rightarrow$ G is 0-, 1-, 2-, or 3-sum of A and B
  $\Rightarrow$ decompose each A and B and make parts “fit together” FAIL!

Partitioning Planar Graphs

Conjecture 11.24 (Chartrand, Geller, Hedetniemi)

Every planar graph is a union of two outerplanar graphs.

~~"Proof"~~:

- WLOG G is a triangulation
- G is 4-connected
  $\Rightarrow$ Hamilton cycle C
  $\Rightarrow$ $(C+$ edges inside $C) \cup (C+$ edges outside $C)$
- min counter-example G not 4-connected
  $\Rightarrow$ G is 0-, 1-, 2-, or 3-sum of A and B
  $\Rightarrow$ decompose each A and B and make parts "fit together" FAIL!
- later, proved by true by Gonçalves

Partitioning Planar Graphs

Conjecture 11.24 (Chartrand, Geller, Hedetniemi)

Every planar graph is a union of two outerplanar graphs.

~~"Proof"~~:

- WLOG G is a triangulation
- G is 4-connected
  $\Rightarrow$ Hamilton cycle C
  $\Rightarrow$ ($C+$ edges inside C) $\cup$ ($C+$ edges outside C)
- min counter-example G not 4-connected
  $\Rightarrow$ G is 0-, 1-, 2-, or 3-sum of A and B
  $\Rightarrow$ decompose each A and B and make parts "fit together" FAIL!
- later, proved by true by Gonçalves

Theorem 11.25 (Ding, O., Sanders, Vertigan; Kedlaya)

Every planar graph is a union of two series-parallel graphs.

Partitioning Planar Graphs

Conjecture 11.24 (Chartrand, Geller, Hedetniemi)

Every planar graph is a union of two outerplanar graphs.

~~"Proof"~~:

- WLOG G is a triangulation
- G is 4-connected
  $\Rightarrow$ Hamilton cycle C
  $\Rightarrow$ ($C+$ edges inside C) $\cup$ ($C+$ edges outside C)
- min counter-example G not 4-connected
  $\Rightarrow$ G is 0-, 1-, 2-, or 3-sum of A and B
  $\Rightarrow$ decompose each A and B and make parts "fit together" FAIL!
- later, proved by true by Gonçalves

Theorem 11.25 (Ding, O., Sanders, Vertigan; Kedlaya)

Every planar has an edge-partition into two graphs of tree-width ≤ 2.

Partitioning Graphs on Surfaces

Theorem 11.26 (Ding, O., Sanders, Vertigan)

Every projective graph has a vertex-partition into two graphs of tw ≤ 2.

Partitioning Graphs on Surfaces

Theorem 11.26 (Ding, O., Sanders, Vertigan)

Every projective graph has a vertex-partition into two graphs of tw ≤ 2.

Theorem 11.27 (DOSV)

Every graph of non-negative Euler characteristic has a vertex-partition and an edge-partition into two graphs of tw ≤ 3.

Partitioning Graphs on Surfaces

Theorem 11.26 (Ding, O., Sanders, Vertigan)

Every projective graph has a vertex-partition into two graphs of tw ≤ 2.

Theorem 11.27 (DOSV)

Every graph of non-negative Euler characteristic has a vertex-partition and an edge-partition into two graphs of tw ≤ 3.

Note 11.28

This is best possible for toroidal graphs.

Theorem 11.26 (Ding, O., Sanders, Vertigan)

Every projective graph has a vertex-partition into two graphs of tw ≤ 2.

Theorem 11.27 (DOSV)

Every graph of non-negative Euler characteristic has a vertex-partition and an edge-partition into two graphs of tw ≤ 3.

Note 11.28

This is best possible for toroidal graphs.

Theorem 11.29 (DOSV)

Every graph G *has*

- *vertex-partition into two graphs of tw* $\leq 6 - 2\chi(G)$

Partitioning Graphs on Surfaces

Theorem 11.26 (Ding, O., Sanders, Vertigan)

Every projective graph has a vertex-partition into two graphs of tw ≤ 2.

Theorem 11.27 (DOSV)

Every graph of non-negative Euler characteristic has a vertex-partition and an edge-partition into two graphs of tw ≤ 3.

Note 11.28

This is best possible for toroidal graphs.

Theorem 11.29 (DOSV)

Every graph G has

- *vertex-partition into two graphs of tw $\leq 6 - 2\chi(G)$*
- *edge-partition into two graphs of tw $\leq 9 - 3\chi(G)$*

Proofs

- Set $v \in V(G)$ and V_k = set of vertices distance k from v.

Proofs

- Set $v \in V(G)$ and $V_k =$ set of vertices distance k from v.
- Vertex-partitions: graphs induced by $\bigcup_{k \text{ even}} V_k$ and $\bigcup_{k \text{ odd}} V_k$

Proofs

- Set $v \in V(G)$ and $V_k =$ set of vertices distance k from v.
- Vertex-partitions: graphs induced by $\bigcup_{k \text{ even}} V_k$ and $\bigcup_{k \text{ odd}} V_k$
- Edge-partitions: let $H_k =$ induced by edges $[V_k, V_k]$ and $[V_k, V_{k+1}]$ $\bigcup_{k \text{ even}} H_k$ and $\bigcup_{k \text{ odd}} H_k$

Minor-Closed Classes

Conjecture 11.30 (Thomas)

For every G there is an integer k such that every graph with no G-minor has a vertex-partition and edge-partition into two graphs of tw $\leq k$.

Minor-Closed Classes

Conjecture 11.30 (Thomas)

For every G there is an integer k such that every graph with no G-minor has a vertex-partition and edge-partition into two graphs of tw $\leq k$.

A graph is a minor of another if it can be obtained by

- deleting edges and/or vertices
- contracting edges

Minor-Closed Classes

Conjecture 11.30 (Thomas)

For every G there is an integer k such that every graph with no G-minor has a vertex-partition and edge-partition into two graphs of tw $\leq k$.

Theorem 11.31 (DOSV, DeVos, Reed, Seymour)

For every minor-closed class of graphs other than the class of all graphs there is a number k such that every member of the class has a vertex-partition and edge-partition into two graphs of tw $\leq k$.

A graph is a minor of another if it can be obtained by

- deleting edges and/or vertices
- contracting edges

Minor-Closed Classes

Conjecture 11.30 (Thomas)

For every G there is an integer k such that every graph with no G-minor has a vertex-partition and edge-partition into two graphs of tw $\leq k$.

Theorem 11.31 (DOSV, DeVos, Reed, Seymour)

For every minor-closed class of graphs other than the class of all graphs there is a number k such that every member of the class has a vertex-partition and edge-partition into two graphs of tw $\leq k$.

A graph is a minor of another if it can be obtained by

- deleting edges and/or vertices
- contracting edges

Minor-closed class of graphs: if G is in the class, then so are all of its minors.

Minor-Closed Classes

Conjecture 11.30 (Thomas)

For every G there is an integer k such that every graph with no G-minor has a vertex-partition and edge-partition into two graphs of tw $\leq k$.

Theorem 11.31 (DOSV, DeVos, Reed, Seymour)

For every minor-closed class of graphs other than the class of all graphs there is a number k such that every member of the class has a vertex-partition and edge-partition into two graphs of tw $\leq k$.

Theorem 11.32 (Robertson and Seymour)

All members of any minor-closed class of graphs other than the class of all graphs are clique-sums of graphs that can "almost" be embedded on surfaces of bounded genus.

A graph is a minor of another if it can be obtained by

- deleting edges and/or vertices
- contracting edges

Minor-closed class of graphs: if G is in the class, then so are all of its minors.

Minor-Closed Classes

Conjecture 11.30 (Thomas)

For every G there is an integer k such that every graph with no G-minor has a vertex-partition and edge-partition into two graphs of tw $\leq k$.

Theorem 11.31 (DOSV, DeVos, Reed, Seymour)

For every minor-closed class of graphs other than the class of all graphs there is a number k such that every member of the class has a vertex-partition and edge-partition into two graphs of tw $\leq k$.

Theorem 11.32 (Robertson and Seymour)

All members of any minor-closed class of graphs other than the class of all graphs are clique-sums of graphs that can "almost" be embedded on surfaces of bounded genus.

A graph is a minor of another if it can be obtained by

- deleting edges and/or vertices
- contracting edges

Minor-closed class of graphs: if G is in the class, then so are all of its minors.

Theorem 11.33 (R&S)

Every minor-closed class of graphs can be characterized by excluding finitely many graphs as minors.

Partitions and Contractions

Question 11.34 (Oxley)

Can every co-graphic matroid be partitioned into two series-parallel matroids?

Partitions and Contractions

Question 11.34 (Oxley)

Can the edges of every graph be partitioned into E_1 and E_2 such that each of G/E_1 and G/E_2 is series parallel?

Partitions and Contractions

Question 11.34 (Oxley)

Can the edges of every graph be partitioned into E_1 and E_2 such that each of G/E_1 and G/E_2 is series parallel?

- yes if planar: dualize our theorem

Partitions and Contractions

Question 11.34 (Oxley)

Can the edges of every graph be partitioned into E_1 and E_2 such that each of G/E_1 and G/E_2 is series parallel?

- yes if planar: dualize our theorem
- yes if 4-connected: 2 edge-disjoint spanning trees by Nash-Williams

Partitions and Contractions

Question 11.34 (Oxley)

Can the edges of every graph be partitioned into E_1 and E_2 such that each of G/E_1 and G/E_2 is series parallel?

- yes if planar: dualize our theorem
- yes if 4-connected: 2 edge-disjoint spanning trees by Nash-Williams

Theorem 11.35 (Morgan, O.)

The edges of every projective graph can be partitioned into E_1 and E_2 such that each of G/E_1 and G/E_2 has tw ≤ 3.

Partitions and Contractions

Question 11.34 (Oxley)

Can the edges of every graph be partitioned into E_1 and E_2 such that each of G/E_1 and G/E_2 is series parallel?

- yes if planar: dualize our theorem
- yes if 4-connected: 2 edge-disjoint spanning trees by Nash-Williams

Theorem 11.35 (Morgan, O.)

The edges of every projective graph can be partitioned into E_1 and E_2 such that each of G/E_1 and G/E_2 has tw ≤ 3.

Theorem 11.36 (MO)

The edges of every toroidal graph can be partitioned into E_1 and E_2 such that of tw$(G/E_1) \leq 3$ and tw$(G/E_2) \leq 4$.

Partitions and Contractions

Question 11.34 (Oxley)

Can the edges of every graph be partitioned into E_1 and E_2 such that each of G/E_1 and G/E_2 is series parallel?

- yes if planar: dualize our theorem
- yes if 4-connected: 2 edge-disjoint spanning trees by Nash-Williams

Theorem 11.35 (Morgan, O.)

The edges of every projective graph can be partitioned into E_1 and E_2 such that each of G/E_1 and G/E_2 has tw ≤ 3.

Theorem 11.36 (MO)

The edges of every toroidal graph can be partitioned into E_1 and E_2 such that of tw$(G/E_1) \leq 3$ and tw$(G/E_2) \leq 4$.

Theorem 11.37 (Demaine, Hajiaghayi, Mohar)

The edges of a graph of genus g can be partitioned into E_1 and E_2 such that each of G/E_1 and G/E_2 has tw $\leq O(g^2)$.

Definition of $T(k, l, r)$

- Start with K_k, and assign all of its vertices level 0

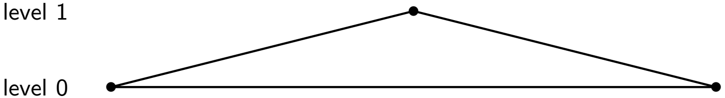
Definition of $T(k, l, r)$

- Start with K_k, and assign all of its vertices level 0
- Inductively, for each K_k subgraph H of level $n-1$, add r new vertices, join each of them to all vertices of H and declare all newly created vertices and K_k subgraphs to have level n.

Definition of $T(k, l, r)$

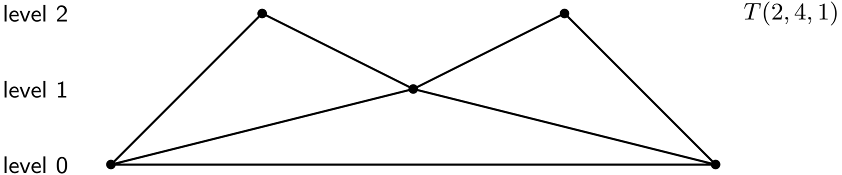
- Start with K_k, and assign all of its vertices level 0
- Inductively, for each K_k subgraph H of level $n-1$, add r new vertices, join each of them to all vertices of H and declare all newly created vertices and K_k subgraphs to have level n.

$T(2,4,1)$



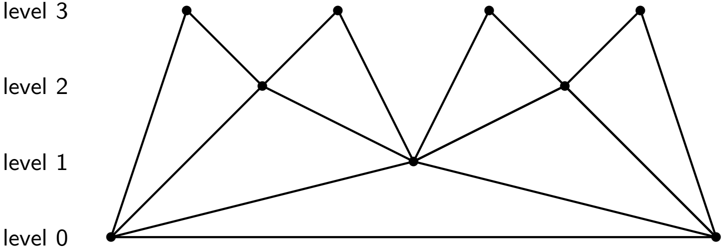
Definition of $T(k,l,r)$

- Start with K_k, and assign all of its vertices level 0
- Inductively, for each K_k subgraph H of level $n-1$, add r new vertices, join each of them to all vertices of H and declare all newly created vertices and K_k subgraphs to have level n.



Definition of $T(k, l, r)$

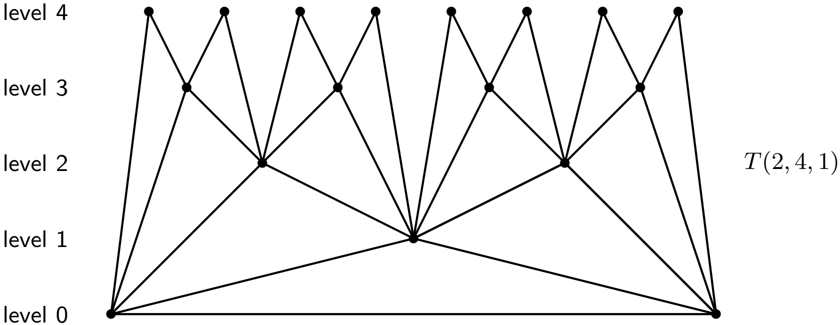
- Start with K_k, and assign all of its vertices level 0
- Inductively, for each K_k subgraph H of level $n - 1$, add r new vertices, join each of them to all vertices of H and declare all newly created vertices and K_k subgraphs to have level n.



$T(2, 4, 1)$

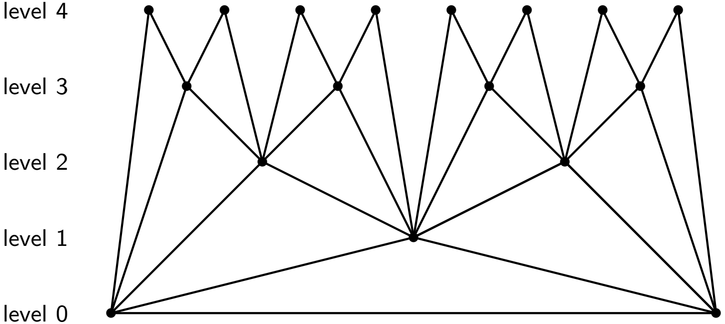
Definition of $T(k, l, r)$

- Start with K_k, and assign all of its vertices level 0
- Inductively, for each K_k subgraph H of level $n - 1$, add r new vertices, join each of them to all vertices of H and declare all newly created vertices and K_k subgraphs to have level n.



Definition of $T(k, l, r)$

- Start with K_k, and assign all of its vertices level 0
- Inductively, for each K_k subgraph H of level $n-1$, add r new vertices, join each of them to all vertices of H and declare all newly created vertices and K_k subgraphs to have level n.
- Stop after having created all level-l subgraphs.



$T(2, 4, 1)$

Partitioning k-Trees

Definition 11.38

k-tree: $T(k, l, r)$ where l is arbitrary, and r can very arbitrarily at every stage.

Partitioning k-Trees

Definition 11.38

k-tree: $T(k, l, r)$ where l is arbitrary, and r can very arbitrarily at every stage.

Theorem 11.39 (DOSV)

Every $(k_1 + k_2 + 1)$-tree has a vertex-partition into a k_1-tree and a k_2-tree.

Partitioning k-Trees

Definition 11.38

k-tree: $T(k, l, r)$ where l is arbitrary, and r can very arbitrarily at every stage.

Theorem 11.39 (DOSV)

Every $(k_1 + k_2 + 1)$-tree has a vertex-partition into a k_1-tree and a k_2-tree.

Theorem 11.40 (DOSV)

Every $(k_1 + k_2)$-tree has an edge-partition into a k_1-tree and a k_2-tree.

Ramsey-Type Results

Theorem 11.41 (DOSV)

For every k_1, k_2, l, and r there is L such that

Ramsey-Type Results

Theorem 11.41 (DOSV)

For every k_1, k_2, l, *and* r *there is* L *such that*
for every vertex-partition $\{G_1, G_2\}$ *of* $T(k_1 + k_2, L, r)$:
$T(k_1, l, r) \subseteq G_1$ *or* $T(k_2, l, r) \subseteq G_2$.

Ramsey-Type Results

Theorem 11.41 (DOSV)

For every k_1, k_2, l, and r there is L such that for every vertex-partition $\{G_1, G_2\}$ of $T(k_1 + k_2, L, r)$: $T(k_1, l, r) \subseteq G_1$ or $T(k_2, l, r) \subseteq G_2$.

Corollary 11.42

For every k, l, and r there is L such that

Ramsey-Type Results

Theorem 11.41 (DOSV)

For every k_1, k_2, l, and r there is L such that for every vertex-partition $\{G_1, G_2\}$ of $T(k_1 + k_2, L, r)$: $T(k_1, l, r) \subseteq G_1$ or $T(k_2, l, r) \subseteq G_2$.

Corollary 11.42

For every k, l, and r there is L such that for every vertex-partition $\{G_1, \ldots, G_k\}$ of $T(k, L, r)$:

Ramsey-Type Results

Theorem 11.41 (DOSV)

For every k_1, k_2, l, and r there is L such that for every vertex-partition $\{G_1, G_2\}$ of $T(k_1 + k_2, L, r)$: $T(k_1, l, r) \subseteq G_1$ or $T(k_2, l, r) \subseteq G_2$.

Corollary 11.42

For every k, l, and r there is L such that for every vertex-partition $\{G_1, \dots, G_k\}$ of $T(k, L, r)$: at least one G_i contains $T(1, l, r)$.

Ramsey-Type Results

Theorem 11.41 (DOSV)

For every k_1, k_2, l, and r there is L such that for every vertex-partition $\{G_1, G_2\}$ of $T(k_1 + k_2, L, r)$: $T(k_1, l, r) \subseteq G_1$ or $T(k_2, l, r) \subseteq G_2$.

Corollary 11.42

For every k, l, and r there is L such that for every vertex-partition $\{G_1, \ldots, G_k\}$ of $T(k, L, r)$: at least one G_i contains $T(1, l, r)$.

Conjecture 11.43 (DOSV)

For every k, l, and r there is L such that

Ramsey-Type Results

Theorem 11.41 (DOSV)

For every k_1, k_2, l, and r there is L such that for every vertex-partition $\{G_1, G_2\}$ of $T(k_1 + k_2, L, r)$: $T(k_1, l, r) \subseteq G_1$ or $T(k_2, l, r) \subseteq G_2$.

Corollary 11.42

For every k, l, and r there is L such that for every vertex-partition $\{G_1, \ldots, G_k\}$ of $T(k, L, r)$: at least one G_i contains $T(1, l, r)$.

Conjecture 11.43 (DOSV)

For every k, l, and r there is L such that for every edge-partition $\{G_1, \ldots, G_k\}$ of $T(k, L, r)$:

Ramsey-Type Results

Theorem 11.41 (DOSV)

For every k_1, k_2, l, and r there is L such that for every vertex-partition $\{G_1, G_2\}$ of $T(k_1 + k_2, L, r)$: $T(k_1, l, r) \subseteq G_1$ or $T(k_2, l, r) \subseteq G_2$.

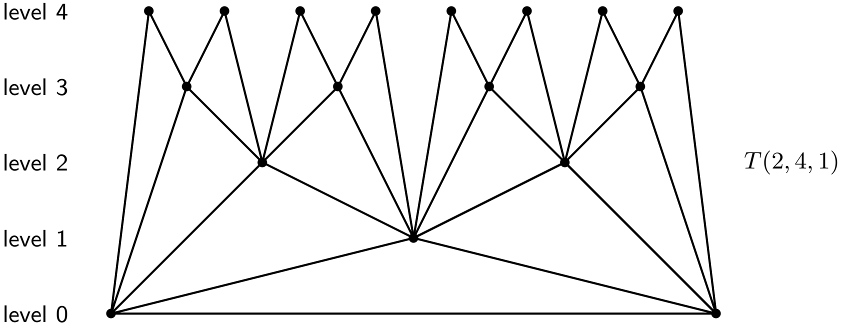
Corollary 11.42

For every k, l, and r there is L such that for every vertex-partition $\{G_1, \ldots, G_k\}$ of $T(k, L, r)$: at least one G_i contains $T(1, l, r)$.

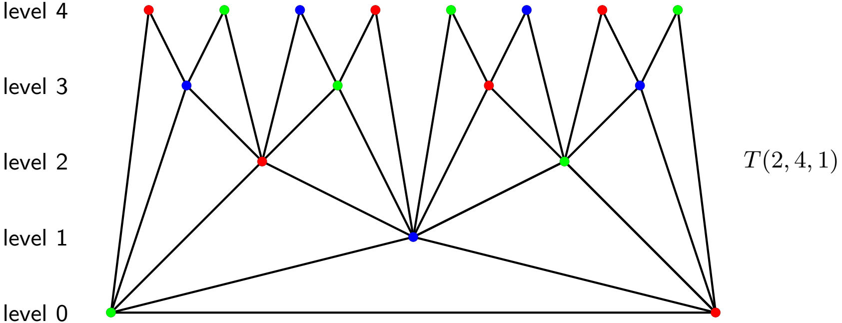
Conjecture 11.43 (DOSV)

For every k, l, and r there is L such that for every edge-partition $\{G_1, \ldots, G_k\}$ of $T(k, L, r)$: at least one G_i contains a subdivision of $T(1, l, r)$.

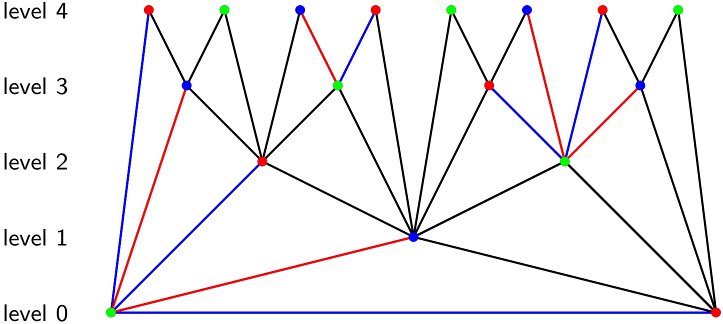
Large k-Trees and Edge-Partitions



Large k-Trees and Edge-Partitions

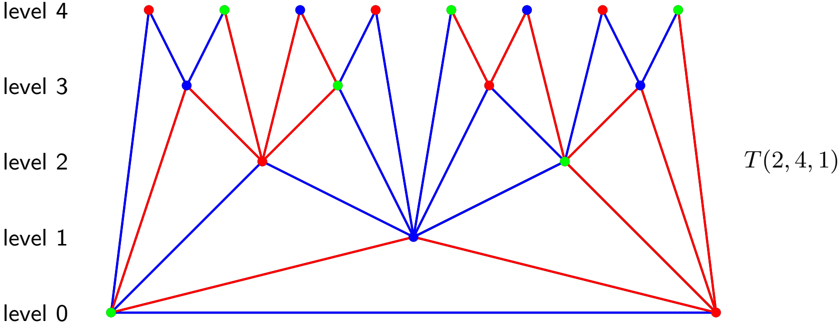


Large k-Trees and Edge-Partitions

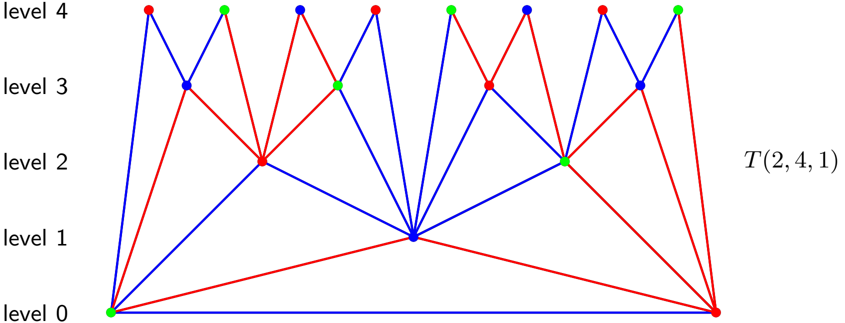


$T(2,4,1)$

Large k-Trees and Edge-Partitions



Large k-Trees and Edge-Partitions



Theorem 11.44 (DOSV)

For every l and r there are L and R such that if $T(2, L, R)$ has its edges colored red and blue, then it contains a red $T(1, l, r)$ or a blue subdivision of $T(1, l, r)$.

Partitioning Into Graphs With Only Small Components

Partitioning Into Graphs With Only Small Components

Theorem 11.45 (Alon, Ding, O, Vertigan)

If k is tree-width and Δ is maximum degree of G, then

Partitioning Into Graphs With Only Small Components

Theorem 11.45 (Alon, Ding, O, Vertigan)

If k is tree-width and Δ is maximum degree of G, then

- *there is a vertex-partition of G into 2 graphs with components on only $\leq 24k\Delta$ vertices;*

Partitioning Into Graphs With Only Small Components

Theorem 11.45 (Alon, Ding, O, Vertigan)

If k is tree-width and Δ is maximum degree of G, then

- *there is a vertex-partition of G into 2 graphs with components on only $\leq 24k\Delta$ vertices;*
- *there is an edge-partition of G into 2 graphs with components on only $\leq 24k\Delta(\Delta + 1)$ vertices.*

Partitioning Into Graphs With Only Small Components

Theorem 11.45 (Alon, Ding, O, Vertigan)

If k is tree-width and Δ is maximum degree of G, then

- *there is a vertex-partition of G into 2 graphs with components on only $\leq 24k\Delta$ vertices;*
- *there is an edge-partition of G into 2 graphs with components on only $\leq 24k\Delta(\Delta+1)$ vertices.*

Q: Is it enough to bound just the tree-width?

Partitioning Into Graphs With Only Small Components

Theorem 11.45 (Alon, Ding, O, Vertigan)

If k is tree-width and Δ is maximum degree of G, then

- *there is a vertex-partition of G into 2 graphs with components on only $\leq 24k\Delta$ vertices;*
- *there is an edge-partition of G into 2 graphs with components on only $\leq 24k\Delta(\Delta + 1)$ vertices.*

Q: Is it enough to bound just the tree-width?

A: No, consider

- large star for edge-partitions,

Partitioning Into Graphs With Only Small Components

Theorem 11.45 (Alon, Ding, O, Vertigan)

If k is tree-width and Δ is maximum degree of G, then

- *there is a vertex-partition of G into 2 graphs with components on only $\leq 24k\Delta$ vertices;*
- *there is an edge-partition of G into 2 graphs with components on only $\leq 24k\Delta(\Delta + 1)$ vertices.*

Q: Is it enough to bound just the tree-width?

A: No, consider

- large star for edge-partitions,
- and a large fan for vertex-partitions.

Partitioning Into Graphs With Only Small Components

Theorem 11.45 (Alon, Ding, O, Vertigan)

If k is tree-width and Δ is maximum degree of G, then

- *there is a vertex-partition of G into 2 graphs with components on only $\leq 24k\Delta$ vertices;*
- *there is an edge-partition of G into 2 graphs with components on only $\leq 24k\Delta(\Delta + 1)$ vertices.*

Q: Is it enough to bound just the tree-width?

A: No, consider

- large star for edge-partitions,
- and a large fan for vertex-partitions.

Q: Is it enough to bound just the vertex degree?

Partitioning Into Graphs With Only Small Components

Theorem 11.45 (Alon, Ding, O, Vertigan)

If k is tree-width and Δ is maximum degree of G, then

- *there is a vertex-partition of G into 2 graphs with components on only $\leq 24k\Delta$ vertices;*
- *there is an edge-partition of G into 2 graphs with components on only $\leq 24k\Delta(\Delta+1)$ vertices.*

Q: Is it enough to bound just the tree-width?

A: No, consider

- large star for edge-partitions,
- and a large fan for vertex-partitions.

Q: Is it enough to bound just the vertex degree?

A: No, there are 4-regular graphs of arbitrarily large girth (Erdős, Sachs)

Partitioning Into Graphs With Only Small Components

Theorem 11.45 (Alon, Ding, O, Vertigan)

If k is tree-width and Δ is maximum degree of G, then

- *there is a vertex-partition of G into 2 graphs with components on only $\leq 24k\Delta$ vertices;*
- *there is an edge-partition of G into 2 graphs with components on only $\leq 24k\Delta(\Delta+1)$ vertices.*

Q: Is it enough to bound just the tree-width?

A: No, consider

- large star for edge-partitions,
- and a large fan for vertex-partitions.

Q: Is it enough to bound just the vertex degree?

A: No, there are 4-regular graphs of arbitrarily large girth (Erdős, Sachs)

- one part of an edge-partition will contain a cycle

Partitioning Into Graphs With Only Small Components

Theorem 11.45 (Alon, Ding, O, Vertigan)

If k is tree-width and Δ is maximum degree of G, then

- *there is a vertex-partition of G into 2 graphs with components on only $\leq 24k\Delta$ vertices;*
- *there is an edge-partition of G into 2 graphs with components on only $\leq 24k\Delta(\Delta+1)$ vertices.*

Q: Is it enough to bound just the tree-width?
A: No, consider

- large star for edge-partitions,
- and a large fan for vertex-partitions.

Q: Is it enough to bound just the vertex degree?
A: No, there are 4-regular graphs of arbitrarily large girth (Erdős, Sachs)

- one part of an edge-partition will contain a cycle
- for vertex-partitions, consider line graphs of those graphs

$\Delta \leq 4$ and Vertex-Partiions

Theorem 11.46 (ADOV)

If $\Delta(G) \leq 4$, then G has a vertex-partition into two graphs on components with at most

$\Delta \leq 4$ and Vertex-Partiions

Theorem 11.46 (ADOV)

If $\Delta(G) \leq 4$, then G has a vertex-partition into two graphs on components with at most 57 vertices.

$\Delta \leq 4$ and Vertex-Partiions

Theorem 11.46 (ADOV)

If $\Delta(G) \leq 4$, then G has a vertex-partition into two graphs on components with at most 57 vertices.

Theorem 11.47 (Haxell, Szabó, Tardos)

57 can be reduced to 6.

$\Delta \leq 4$ and Vertex-Partiions

Theorem 11.46 (ADOV)

If $\Delta(G) \leq 4$, then G has a vertex-partition into two graphs on components with at most 57 vertices.

Theorem 11.47 (Haxell, Szabó, Tardos)

57 can be reduced to 6.

Note: 5 is a lower bound.

Theorem 11.48 (Haxell, Szabó, Tardos)

If $\Delta(G) \leq 5$, then G has a vertex-partition into two graphs on components with at most

$\Delta \leq 4$ and Vertex-Partiions

Theorem 11.46 (ADOV)

If $\Delta(G) \leq 4$, then G has a vertex-partition into two graphs on components with at most 57 vertices.

Theorem 11.47 (Haxell, Szabó, Tardos)

57 can be reduced to 6.

Note: 5 is a lower bound.

Theorem 11.48 (Haxell, Szabó, Tardos)

If $\Delta(G) \leq 5$, then G has a vertex-partition into two graphs on components with at most 6,053,628,175 vertices.

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No!

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No! For a positive integer n, take n disjoint copies of a fan on $n^2 + n + 1$ vertices.

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No! For a positive integer n, take n disjoint copies of a fan on $n^2 + n + 1$ vertices. Then add one more vertex v_0 joining it to all vertices of all the fans; name the graph U_n.

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No! For a positive integer n, take n disjoint copies of a fan on $n^2 + n + 1$ vertices. Then add one more vertex v_0 joining it to all vertices of all the fans; name the graph U_n.

Theorem 11.50

In every vertex 3-coloring of U_n, there is a monochromatic component on more than n vertices.

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No! For a positive integer n, take n disjoint copies of a fan on $n^2 + n + 1$ vertices. Then add one more vertex v_0 joining it to all vertices of all the fans; name the graph U_n.

Theorem 11.50

In every vertex 3-coloring of U_n, there is a monochromatic component on more than n vertices.

Proof.

Without loss of generality, the color of v_0 is red.

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No! For a positive integer n, take n disjoint copies of a fan on $n^2 + n + 1$ vertices. Then add one more vertex v_0 joining it to all vertices of all the fans; name the graph U_n.

Theorem 11.50

In every vertex 3-coloring of U_n, there is a monochromatic component on more than n vertices.

Proof.

Without loss of generality, the color of v_0 is red. If each fan has a vertex colored red, then the conclusion follows.

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No! For a positive integer n, take n disjoint copies of a fan on $n^2 + n + 1$ vertices. Then add one more vertex v_0 joining it to all vertices of all the fans; name the graph U_n.

Theorem 11.50

In every vertex 3-coloring of U_n, there is a monochromatic component on more than n vertices.

Proof.

Without loss of generality, the color of v_0 is red. If each fan has a vertex colored red, then the conclusion follows. So suppose that one of the fans F has its vertices colored with only two colors,

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No! For a positive integer n, take n disjoint copies of a fan on $n^2 + n + 1$ vertices. Then add one more vertex v_0 joining it to all vertices of all the fans; name the graph U_n.

Theorem 11.50

In every vertex 3-coloring of U_n, there is a monochromatic component on more than n vertices.

Proof.

Without loss of generality, the color of v_0 is red. If each fan has a vertex colored red, then the conclusion follows. So suppose that one of the fans F has its vertices colored with only two colors, and suppose the tip v of F is blue.

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No! For a positive integer n, take n disjoint copies of a fan on $n^2 + n + 1$ vertices. Then add one more vertex v_0 joining it to all vertices of all the fans; name the graph U_n.

Theorem 11.50

In every vertex 3-coloring of U_n, there is a monochromatic component on more than n vertices.

Proof.

Without loss of generality, the color of v_0 is red. If each fan has a vertex colored red, then the conclusion follows. So suppose that one of the fans F has its vertices colored with only two colors, and suppose the tip v of F is blue. If F has n other vertices colored blue, then the conclusion follows.

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No! For a positive integer n, take n disjoint copies of a fan on $n^2 + n + 1$ vertices. Then add one more vertex v_0 joining it to all vertices of all the fans; name the graph U_n.

Theorem 11.50

In every vertex 3-coloring of U_n, there is a monochromatic component on more than n vertices.

Proof.

Without loss of generality, the color of v_0 is red. If each fan has a vertex colored red, then the conclusion follows. So suppose that one of the fans F has its vertices colored with only two colors, and suppose the tip v of F is blue. If F has n other vertices colored blue, then the conclusion follows. In the remaining case, the blue vertices cut the path $F - v$ into at most n monochromatic segments,

Three-Color Theorem for Planar Graphs?

Question 11.49

Is there a number c such that the every planar graph can have its vertices colored with 3-colors so that each monochromatic component has at most c vertices?

Answer: No! For a positive integer n, take n disjoint copies of a fan on $n^2 + n + 1$ vertices. Then add one more vertex v_0 joining it to all vertices of all the fans; name the graph U_n.

Theorem 11.50

In every vertex 3-coloring of U_n, there is a monochromatic component on more than n vertices.

Proof.

Without loss of generality, the color of v_0 is red. If each fan has a vertex colored red, then the conclusion follows. So suppose that one of the fans F has its vertices colored with only two colors, and suppose the tip v of F is blue. If F has n other vertices colored blue, then the conclusion follows. In the remaining case, the blue vertices cut the path $F - v$ into at most n monochromatic segments, and so at least one of those segments must have more than n vertices. □

Four-Color Theorem for Minor-Closed Classes of Graphs

Theorem 11.51 (ADOV+S)

Let $\mathcal{G}$ be a minor-closed class of graphs other than the class of all graphs, and pick Δ.
There is a number $c(\mathcal{G}, \Delta)$ such that every member of $\mathcal{G}$ whose max degree is $\leq \Delta$ can be vertex 4-colored so that all monochromatic components have at most c vertices.

Four-Color Theorem for Minor-Closed Classes of Graphs

Theorem 11.51 (ADOV+S)

Let $\mathcal{G}$ be a minor-closed class of graphs other than the class of all graphs, and pick Δ.
There is a number $c(\mathcal{G}, \Delta)$ such that every member of $\mathcal{G}$ whose max degree is $\leq \Delta$ can be vertex 4-colored so that all monochromatic components have at most c vertices.

Proof.

- Every graph G in $\mathcal{G}$ has a vertex-partition into two graphs of tw $\leq w(\mathcal{G})$.

$\square$

Four-Color Theorem for Minor-Closed Classes of Graphs

Theorem 11.51 (ADOV+S)

Let $\mathcal{G}$ be a minor-closed class of graphs other than the class of all graphs, and pick Δ.
There is a number $c(\mathcal{G}, \Delta)$ such that every member of $\mathcal{G}$ whose max degree is $\leq \Delta$ can be vertex 4-colored so that all monochromatic components have at most c vertices.

Proof.

- Every graph G in $\mathcal{G}$ has a vertex-partition into two graphs of tw $\leq w(\mathcal{G})$.
- If $\Delta(G) \leq \Delta$, then each of those can be 2-colored with components on at most c vertices.

$\square$

Four-Color Theorem for Minor-Closed Classes of Graphs

Theorem 11.51 (ADOV+S)

Let $\mathcal{G}$ be a minor-closed class of graphs other than the class of all graphs, and pick Δ.
There is a number $c(\mathcal{G}, \Delta)$ such that every member of $\mathcal{G}$ whose max degree is $\leq \Delta$ can be vertex 4-colored so that all monochromatic components have at most c vertices.

Proof.

- Every graph G in $\mathcal{G}$ has a vertex-partition into two graphs of tw $\leq w(\mathcal{G})$.
- If $\Delta(G) \leq \Delta$, then each of those can be 2-colored with components on at most c vertices.
- This gives a 4-coloring of G with components on at most c vertices.

$\square$