Journal of Computational Physics 230 (2011) 8669–8682

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

An adaptive high-order minimum action method

Xiaoliang Wan

Department of Mathematics and Center for Computation and Technology, Louisiana State University, Baton Rouge, LA 70803, United States

ARTICLE INFO

Article history:
Received 21 March 2011
Received in revised form 19 July 2011
Accepted 11 August 2011
Available online 24 August 2011

Keywords:
Random perturbation
Dynamical system
White noise
Minimum action method
Rare events
Spectral elements

ABSTRACT

In this work, we present an adaptive high-order minimum action method for dynamical systems perturbed by small noise. We use the *hp* finite element method to approximate the minimal action path and nonlinear conjugate gradient method to solve the optimization problem given by the Freidlin–Wentzell least action principle. The gradient of the discrete action functional is obtained through the functional derivative and the moving mesh technique is employed to enhance the approximation accuracy. Numerical examples are given to demonstrate the efficiency and accuracy of the proposed numerical method.

1. Introduction

Dynamical systems are often subject to random perturbations since noise is ubiquitous in nature. Even when these random perturbations have a small amplitude, they can produce a profound effect on the long time dynamics by inducing rare but important events. A large number of interesting phenomena in physics, chemistry and biology such as phase transitions, biological switches and chemical reactions, etc., are examples of such noise-induced rare events [13].

When the random perturbations are small, the Freidlin–Wentzell theory of large deviations provides a rigorous mathematical framework for us to understand how the transitions occur and how frequent they are. The transition pathways between metastable sets in a dynamical system often have a rather deterministic nature. As the noise amplitude decreases to zero, the events for successful transitions between metastable sets have a sharply peaked probability around a certain deterministic path that is least unlikely. Special features of such a path tell us crucial information about the mechanism of the transition. One class of examples that have been well studied for a long time are the gradient systems, for which the vector field is the gradient of a potential function. In gradient systems, the most probable transition path is the minimum energy path (MEP), which passes through the basin boundary between the stable states at some saddle points with one dimensional unstable manifold [16,20]. For non-gradient systems we need to consider the action functional instead of the energy, which is the central object to the Freidlin–Wentzell theory. The minimizer of the action functional provides the most probable transition path; the minimum of the action functional provides an estimate of the probability and the rate of occurrence of the transition. Thus an important practical task is to compute the minimum and minimizer of the action functional.

A large number of numerical algorithms have been designed for gradient systems. Some popular algorithms include the string method [2,4], nudged elastic band method [12], eigenvector-following-type method (e.g. [1]) as well as the dimer method [9], which usually take advantage of the fact that in gradient systems the transition paths are always parallel to the drift term of the stochastic differential equation. For general (non-gradient) systems, we need to minimize directly

E-mail address: xlwan@math.lsu.edu

0021-9991/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2011.08.006

the Freidlin–Wentzell action functional and available algorithms include the minimum action method [3], the adaptive minimum action method [18] and the geometric minimum action method [8]. Compared to gradient systems, the transition mechanism in non-gradient systems is usually much more subtle, since the phase space may display a very complicated structure, where invariant sets can be fixed points, as well as limit cycles, tori or even chaotic strange attractors. We refer to [19,21] for the study of the Lorenz system and the Kuramoto–Sivashinsky equation, where it is demonstrated that the minimum action method can be a valuable tool to explore the phase space and study the transition mechanism in non-gradient systems.

In this work, we develop an adaptive high-order minimum action method by coupling the *hp* finite element approximation and a preconditioned nonlinear conjugate gradient optimization solver. In the finite element framework, the gradient of the action functional is formulated straightforwardly with respect to the functional derivative. To enhance the accuracy, we employ the moving meshing technique to adjust the temporal discretization adaptively. The methodology is general and can be easily applied to both gradient and non-gradient dynamical systems.

This paper is organized as follows. In Section 2 we briefly describe the problem and the theoretical background. We present the developed numerical method in Section 3. In Section 4, we examine the accuracy and efficiency of the method using dynamical systems given by an ordinary differential equation and a partial differential equation, respectively. Some discussions are given in Section 5.

2. Problem description and theoretical background

We consider random perturbations of dynamical systems. Let the random process $X_t = X(t) : \mathbb{R}_+ \to \mathbb{R}^n$ defined by the following stochastic ordinary differential equation (SODE):

$$dX_t = b(X_t)dt + \sqrt{\varepsilon}\,dW_t, \tag{1}$$

where W_t is a standard Wiener process in $\mathbb{R}^n$ and ε is a small positive parameter. Let $\phi(t) \in \mathbb{R}^n$ be an absolutely continuous function defined for $t \in [0,T]$. The Wentzell–Freidlin theory tells us that the probability of $X(t)$ passing through the δ-tube about ϕ on $[0,T]$ is

$$\Pr(\rho(X,\phi) < \delta) \approx \exp\left(-\frac{1}{\varepsilon}S_T(\phi)\right) \tag{2}$$

with $\rho(\phi,\varphi) = \sup_{t\in[0,T]}|\phi(t) - \varphi(t)|$, $|\cdot|$ indicates the ℓ_2 norm in $\mathbb{R}^n$, and $S_T(\phi)$ is the action functional of ϕ on $[0,T]$, defined as

$$S_T(\phi) = \frac{1}{2}\int_0^T L(\dot{\phi},\phi)\,dt, \tag{3}$$

where $L\left(\dot{\phi},\phi\right) = \left|\dot{\phi} - b(\phi)\right|^2$. In general, we have the following large deviation principle

$$\lim_{\varepsilon\to 0}\varepsilon\log\Pr(X\in A) = -\min_{\phi\in A}S_T(\phi), \tag{4}$$

where A is a particular set of random events. Thus, in analogy with the Laplace's method, the basic contribution to $\Pr(X \in A)$ is given by the neighborhood of the minimum of $S_T(\phi)$ when ε is small enough. The minimizer ϕ^*, which satisfies $S_T(\phi^*) = \min_{\phi\in A}S_T(\phi)$ is also called the "minimal action path" (MAP).

Different definitions of the set A in Eq. (4) correspond to many important phenomena that occur in dynamical systems. For example,

- If we are interested in the probability of $X(t)$ connecting one point a_1 and the other point a_2 in the phase space due to the random perturbations, A can be defined as

$$A = \{X(0) = a_1, X(T) = a_2\}.$$

The MAP will be the most probable path for the transition from a_1 to a_2 in the sense that the probability of the system taking all the other paths decays exponentially with respect to the noise amplitude ε according to the large deviation principle. Note that when a_1 and a_2 are attractors, it is more appropriate to define the set A as

$$A = \{X(-\infty) = a_1, X(\infty) = a_2\}.$$

We keep a finite time interval here mainly due to the numerical approximation discussed later.

- If a_1 and a_2 are two adjacent stable states in gradient systems, the MAP will be consistent with the minimum energy path (MEP), which passes through the basin boundary between a_1 and a_2 at a certain saddle point with one-dimensional unstable manifold.
- If there exists dynamics between a_1 and a_2, the MAP will be the path given by the dynamics corresponding to a zero action functional, which implies that the MAP is also helpful for us to study the structure of the phase space. For instance, if a_1 and a_2 are two unstable fixed points and the MAP has a zero action functional, we can conclude that there exists a heteroclinic orbit between a_1 and a_2, which is given exactly by the MAP.

Although the Wentzell–Freidlin theory shows that it is very important to find out the MAP when we consider random perturbations of dynamical systems, it is usually very difficult to obtain it analytically for a general dynamical system. We need to consider numerical approximation in practice.

2.1. Available numerical approaches

We here only focus on numerical methods for general (non-gradient) dynamical systems, which are usually called minimum action methods (MAM).

- **The original MAM**. The original MAM was proposed in [3] coupling the finite difference discretization in time and an L-BFGS optimization solver. The optimization problem given by Eq. (4) is solved on a finite time interval $[0,T]$ although the real time interval should be $[0,\infty)$ if the end of the path is an attractor.
- **The adaptive MAM**. The adaptive MAM was proposed in [18]. The authors observed that a sufficiently large time interval can resolve the MAP defined on an infinite time interval and the main reason that the original MAM may converge poorly is due to the slow dynamics in the transition regions which makes the uniform temporal discretization dramatically skewed with respect to the arc length. A moving mesh technique is then used to redistribute the grid points to make them more uniform according to the arc length, which improves significantly both accuracy and efficiency.
- **The geometric MAM**. The geometric MAM was proposed in [8]. The authors considered discretization with respect to a parametrization variable (the arc length) instead of time. Thus the aforementioned issues given by the temporal discretization are avoid. However, the problem needs to be reformulated with respect to the parametrization variable, i.e., in the space of curves.

3. An adaptive high-order minimum action method

A common feature of the minimum action methods summarized in Section 2.1 is to couple the finite difference discretization with respect to time or arc length and an appropriate optimization solver. It is well known that the efficiency of almost all optimization algorithms requires the computation of the gradient. However, the gradient of the action functional based on a finite difference discretization is not easy to generalize especially when the dynamical system is complex and a high-order finite difference discretization is considered. Based such an observation, we consider the minimum action method in the framework of finite element approximation. In particular, we look at the *hp* finite element discretization of the action functional. For convenience, we study the discretization with respect to time instead of arc length such that we do not have to reformulate the original problem.

3.1. A variational approach based on the hp finite element approximation

We first consider the optimization problem given by the large deviation principle in an abstract form:

$$\min_{\phi\in A} S_T(\phi) = \min_{\phi\in A} \int_0^T L(\dot{\phi},\phi)\,dt, \tag{5}$$

where the constraints are

$$\phi(0) = a_1, \quad \phi(T) = a_2. \tag{6}$$

Let $\frac{\delta S_T}{\delta \phi}$ be the functional derivative of S_T with respect to ϕ, which satisfies

$$\delta S_T(\phi) = \left\langle \frac{\delta S_T(\phi)}{\delta \phi}, \delta\phi \right\rangle_t = \lim_{\epsilon\to 0} \frac{S_T(\phi+\epsilon\delta\phi) - S_T(\phi)}{\epsilon}, \tag{7}$$

with $\delta\phi$ being an arbitrary perturbation testing function. Here $\langle f(t),g(t)\rangle_t$ indicates the inner product of functions $f(t),\ g(t)\in\mathbb{R}^n$. It is well known that the Euler–Lagrange equation is given by

$$\begin{cases} \frac{\delta S_T(\phi)}{\delta\phi} = 0, & t\in(0,T), \\ \phi(t) = a_1, & t = 0, \\ \phi(t) = a_2, & t = T, \end{cases} \tag{8}$$

whose solution is a local or global minimizer of $S_T(\phi)$. However, for a general dynamical system, the Euler–Lagrange equation of $S_T(\phi)$ is usually a high-order nonlinear PDE, which is difficult to be discretized directly. Instead of considering the Euler–Lagrange equation (8), we solve the optimization problem (5) using a variational approach. Suppose that we choose a finite dimensional approximation space spanned by $\{\psi_i(t)\}_{i=1}^M$ such that

$$\phi(t) = \sum_{i=1}^M \phi_i\psi_i(t), \quad \delta\phi(t) = \sum_{i=1}^M \delta\phi_i\psi_i(t), \tag{9}$$

where $\phi_i, \delta\phi_i \in \mathbb{R}^n$. We then have

$$\delta S_T(\phi) = \left\langle \frac{\delta S_T}{\delta \phi}, \delta\phi \right\rangle_t = \sum_{i=1}^{M} \left\langle \frac{\delta S_T}{\delta \phi}, \delta\phi_i \psi_i(t) \right\rangle_t. \tag{10}$$

Consider the particular choice of $\delta\phi$, whose coefficients are equal to zero except the jth component $\delta\phi_{ij}$ of $\delta\phi_i$. We then obtain

$$\delta S_T(\phi) = \left\langle \frac{\delta S_T}{\delta \phi}, \psi_i(t)\mathbf{e}_j \right\rangle_t \delta\phi_{ij}, \tag{11}$$

which implies that

$$(\nabla S_T(\phi))_{k(i,j)} = \frac{\partial S_T}{\partial \phi_{ij}} = \left\langle \frac{\delta S_T}{\delta \phi}, \psi_i(t)\mathbf{e}_j \right\rangle_t, \tag{12}$$

where $\mathbf{e}_j \in \mathbb{R}^n$ is the unit Euclidean vector such that its jth component is 1 and the rest components are zero, $k(i,j)$ is a global index uniquely determined by $i = 1,\ldots,M$ and $j = 1,\ldots,n$. Note that ∇S_T is usually required by an efficient optimization algorithm, such as nonlinear conjugate gradient method, L-BFGS, etc.

We subsequently present a choice of the approximation space of $\phi(t)$, which allows high-order approximation. Consider a (nonuniform) partitioning $\mathcal{T}_h$ of the time interval $[0,T]$:

$$\mathcal{T}_h : 0 = t_0 < t_1 < \cdots < t_{N+1} = T.$$

We define the finite element approximation space as

$$V_h^K = \{v : v \circ F_k^{-1} \in \mathcal{P}_p(R)\},$$
$$V_h = \left\{ v \in H_0^1(T) : v|_K \in V_h^K, K \in \mathcal{T}_h \right\},$$

where F_K is the mapping function for the element $K = [t_i, t_{i+1}]$, $i = 0,1,\ldots,N$, which maps the reference element $R = [-1,1]$ to element K, and $\mathcal{P}_p(R)$ denotes the set of polynomials of degree up to p over R. On the reference element R, we assume that $\mathcal{P}_p(R)$ consists of linear combinations of the following basis functions [14]:

$$\hat{\psi}_i(\tau) = \begin{cases} \frac{1-\tau}{2} & i = 0, \\ \frac{1-\tau}{2}\frac{1+\tau}{2} P_{i-1}^{1,1}(\tau), & 0 < i < p, \\ \frac{1+\tau}{2} & i = p, \end{cases} \tag{13}$$

where $P_i^{1,1}$ denote orthogonal Jacobi polynomials of degree i with respect to the weight function $(1-\tau)(1+\tau)$. $\hat{\psi}_0(\tau)$, and $\hat{\psi}_p(\tau)$ are consistent with linear finite element basis, and $\hat{\psi}_i(\tau),\ 0 < i < p$, are introduced for high-order approximation. Note that $\hat{\psi}_i(\pm 1) = 0$ for $0 < i < p$. We call $\hat{\psi}_0(\tau)$ and $\hat{\psi}_p(\tau)$ boundary modes, and $\hat{\psi}_i(\tau)$, $0 < i < p$, interior modes. Note here that $V_h \subset H_0^1([0,T])$ due to the facts that $\phi(0)$ and $\phi(T)$ are specified and the action functional $S_T(\phi)$ takes the form

$$S_T(\phi) = \frac{1}{2}\left\langle \dot{\phi} - b(\phi), \dot{\phi} - b(\phi) \right\rangle_t. \tag{14}$$

Let $\{\psi_i(t)\}_{i=1}^M$ expand the finite element approximation space V_h, where M is the total number of degrees of freedom. The absolutely continuous function $\phi(t) \in \mathbb{R}^n$ can be approximated as

$$\phi(t) \approx \phi_h(t) = \sum_{i=1}^{M} \phi_i \psi_i(t), \quad \phi_i \in \mathbb{R}^n. \tag{15}$$

Then the action functional takes an approximate form

$$S_T(\phi_h) = \frac{1}{2}\int_0^T L\left(\sum_{i=1}^{M} \phi_i \dot{\psi}_i(t), \sum_{i=1}^{M} \phi_i \psi_i(t) \right) dt. \tag{16}$$

3.2. Computation of the gradient of $S_T(\phi_h)$

Let $\phi_h^* = \sum_{i=1}^M \phi_i^* \psi_i(t)$ be the finite element expansion of the MAP ϕ^*. Then the finite element coefficients $\phi_{ij}^* \in \mathbb{R}$ of the MAP ϕ^* will satisfy the discrete optimization problem

$$S_T(\phi_h^*) = \min_{\phi_h \in A} S_T(\phi_h) = \min_{\phi_h \in A} S_T(\phi_{ij}), \tag{17}$$

where $i = 1,\ldots,M$ and $j = 1,\ldots,n$.

We now derive the gradient $\partial S_T(\phi_h)/\partial\phi_{ij}$. For clarity, we rewrite the action functional as

$$S_T(\phi_h) = \frac{1}{2}\left\langle \dot{\phi}_h - b(\phi_h), \dot{\phi}_h - b(\phi_h)\right\rangle_t. \tag{18}$$

We assume that the linear part of the perturbation of $b(\phi)$ is given as

$$b(\phi + \delta\phi) = b(\phi) + \hat{b}(\phi)\delta\phi + O(\delta^2\phi), \tag{19}$$

where $\delta\phi$ is an arbitrary small perturbation of $\phi(t)$. We then have

$$\begin{aligned} S_T(\phi_h + \delta\phi_h) - S_T(\phi_h) &= \frac{1}{2}\left\langle \dot{\phi}_M + \delta\dot{\phi}_M - b(\phi_h) - \hat{b}(\phi_h)\delta\phi_h, \dot{\phi}_M + \delta\dot{\phi}_M - b(\phi_h) - \hat{b}(\phi_h)\delta\phi_h\right\rangle_t \\ &\quad - \frac{1}{2}\left\langle \dot{\phi}_M - b(\phi_h), \dot{\phi}_M - b(\phi_h)\right\rangle_t + O(\delta^2\phi_h) + O\left(\delta\phi_h\delta\dot{\phi}_M\right) \\ &= \left\langle \dot{\phi}_M - b(\phi_h), \delta\dot{\phi}_M - \hat{b}(\phi_h)\delta\phi_h\right\rangle_t + O(\delta^2\phi_h) + O\left(\delta\phi_h\delta\dot{\phi}_M\right). \end{aligned}$$

Taking the linear part of $S_T(\phi_h + \delta\phi_h) - S_T(\phi_h)$, we have the functional derivative

$$\left\langle \frac{\delta S_T(\phi_h)}{\delta\phi_h}, \delta\phi_h\right\rangle_t = \left\langle \dot{\phi}_M - b(\phi_h), \delta\dot{\phi}_M - \hat{b}(\phi_h)\delta\phi_h\right\rangle_t. \tag{20}$$

Let $\delta\phi_h = \delta\phi_{i,j}\psi_i(t)\mathbf{e}_j$. Using Eq. (12), we obtain the gradient of $S_T(\phi_h)$ as

$$(\nabla S_T(\phi_h))_{k(i,j)} = \frac{\partial S_T(\phi_h)}{\partial\phi_{i,j}} = \left\langle \dot{\phi}_h - b(\phi_h), \dot{\psi}_i(t)\mathbf{e}_j - \hat{b}(\phi_h)\psi_i(t)\mathbf{e}_j\right\rangle_t. \tag{21}$$

It is observed that the computation of $\partial S_T/\partial\,\phi_{i,j}$ is an integration problem, which can be easily dealt with by using Gauss-type quadrature formulas.

Once the gradient of the action functional is obtained, we use the nonlinear conjugate gradient (CG) method to solve the optimization problem to get the MAP ϕ_h^*. Let $\Phi \in \mathbb{R}^{Mn}$ be a global vector whose components are $\phi_{i,j}$. The nonlinear CG method can be summarized as

$$\begin{cases} \Phi_{k+1} = \Phi_k + \alpha_k\mathbf{d}_k, \\ \mathbf{d}_{k+1} = -\mathbf{g}_{k+1} + \beta_k^{HZ}\mathbf{d}_k, \quad \mathbf{d}_0 = -\mathbf{g}_0, \end{cases} \tag{22}$$

where the subscript k indicates the iteration step, the positive step size α_k is obtained by a line search algorithm, $\mathbf{g}_k = \nabla S_T(\Phi_k)$, and β_k^{HZ} is the CG update parameter. We define $\beta_k^{HZ} = \max\{\beta_k, \eta_k\}$ as in [7]:

$$\beta_k = \left(\mathbf{y}_k - 2\mathbf{d}_k\frac{|\mathbf{y}_k|^2}{\mathbf{d}_k^{\mathrm{T}}\mathbf{y}_k}\right)^{\mathrm{T}}\frac{\mathbf{g}_{k+1}}{\mathbf{d}_k^{\mathrm{T}}\mathbf{y}_k}, \quad \eta_k = \frac{-1}{|\mathbf{d}_k|\min\{0.01, |\mathbf{g}_k|\}} \tag{23}$$

with $\mathbf{y}_k = \mathbf{g}_{k+1} - \mathbf{g}_k$.

When the preconditioning is desired, we consider a new variable $\Phi = \mathbf{S}\widehat{\Phi}$, where $\mathbf{S}$ is an invertible matrix chosen to speed-up the convergence. Writing the nonlinear CG method with respect to $\widehat{\Phi}$ and converting it back to Φ we obtain the preconditioned nonlinear CG method:

$$\begin{cases} \Phi_{k+1} = \Phi_k + \alpha_k\mathbf{d}_k, \\ \mathbf{d}_{k+1} = -\mathbf{P}\mathbf{g}_{k+1} + \bar{\beta}_k^{HZ}\mathbf{d}_k, \quad \mathbf{d}_0 = -\mathbf{P}\mathbf{g}_0, \end{cases} \tag{24}$$

where $\mathbf{P} = \mathbf{S}\mathbf{S}^{\mathrm{T}}$. The parameter $\bar{\beta}_k^{HZ}$ is the same as β_k^{HZ} except that $\mathbf{g}_k$ and $\mathbf{d}_k$ are replaced by $\mathbf{S}^{\mathrm{T}}\mathbf{g}_k$ and $\mathbf{S}^{-1}\mathbf{d}_k$, respectively. However, we do not need to know $\mathbf{S}$ explicitly by observing that $(\mathbf{S}^{\mathrm{T}}\mathbf{g}_k)^{\mathrm{T}}(\mathbf{S}^{\mathrm{T}}\mathbf{g}_k) = \mathbf{g}_k^{\mathrm{T}}\mathbf{S}\mathbf{S}^{\mathrm{T}}\mathbf{g}_k = \mathbf{g}_k^{\mathrm{T}}\mathbf{P}\mathbf{g}_k$ and $\left(\mathbf{S}^{-1}\mathbf{d}_k\right)^{\mathrm{T}}(\mathbf{S}^{\mathrm{T}}\mathbf{y}_k) = \mathbf{d}_k^{\mathrm{T}}\mathbf{S}^{-\mathrm{T}}\mathbf{S}^{\mathrm{T}}\mathbf{y}_k = \mathbf{d}_k^{\mathrm{T}}\mathbf{y}_k$. Thus we only need to know the matrix $\mathbf{P}$. An effective preconditioner $\mathbf{P}$ is, in general, an approximation of the Hessian. Unfortunately, it is usually difficult to find an effective preconditioner for a general nonlinear function, which is heavily problem dependent. In this work we use the inverse of the linear part of the Euler–Lagrange equation of the action functional as a preconditioner whenever possible, see Section 4.2 for an application of such a strategy.

3.3. Some computational issues

We subsequently discuss some important computation issues with respect to the numerical efficiency and accuracy.

3.3.1. Parallel computing

Parallel computing is the most straightforward means to deal with large scale simulations, for which the scalability of the algorithm is essential. For simplicity, we here only consider the parallelization with respect to the temporal discretization. The key observation is that both S_T and $\partial S_T/\partial\phi_{i,j}$ are expressed as a time integration, which can be further decomposed into location time integrations in each finite element. Suppose that the same polynomial order is employed in each finite element, then all the finite elements can be uniformly distributed onto the CPU processors. Due to the choice of the finite

element basis, i.e., $\hat{\psi}_i(\pm 1) = 0$ for $0 < i < p$, only the coefficients of $\hat{\psi}_0$ or $\hat{\psi}_p$ need to be exchanged between two adjacent CPU processors. In this sense, the proposed numerical method is highly scalable for parallel computing.

3.3.2. Complexity

With respect to complexity, the finite element approximation of the MAP has the following two main advantages: (1) When we add more interior basis functions $\psi_i(t)$ (p-refinement) or more elements (h-refinement) to refine the approximation, the computation of the gradient of the action functional requires no extra reformulations; (2) Since interior basis functions vanish at element boundaries, the information exchanged between CPU processors will remain the same for both h- and p-refinement.

If we employ finite difference methods to approximate the action functional, the complexity for the aforementioned two issues will increase. First, it is not easy to derive a uniform formula of gradient for finite difference discretizations of an arbitrary order and the formula can become much more complicated when a high-order finite difference scheme is employed; Second, when a larger stencil is employed, more information need to be exchanged between CPU processors and decreases the scalability of the algorithm for parallel computation.

3.3.3. Adaptivity

One difficulty of approximating the MAP is that the dynamics can significantly affect the quality of temporal discretization. Since we are looking for a curve in the phase space, we can also describe it by the arc length, i.e., the temporal discretization corresponds to an arc length discretization of the MAP. Specifically, the time element $[t_i, t_{i+1}]$ corresponds to the arc length element $\left[\int_0^{t_i} \sqrt{|\phi_t|^2}\, dt, \int_0^{t_{i+1}} \sqrt{|\phi_t|^2}\, dt\right]$. However, due to the nonlinear relation between time and arc length, a uniform discretization with respect to time may correspond to a highly nonuniform discretization with respect to arc length. For example, the time element $[t_i, t_{i+1}]$ has an element size $t_{i+1} - t_i$ while the corresponding arc length element has an element size $\int_{t_i}^{t_{i+1}} \sqrt{|\phi_t|^2}\, dt$, which is determined by ϕ_t. In the transition region close to fixed points, the dynamics will become much slower, i.e., ϕ_t is close to zero, the arc length elements become very small and do not contribute to the approximation of the MAP. To improve the accuracy, we employ the moving mesh technique proposed in [18].

Let $s \in [0,1]$ indicate a scaled arc length such that the total length of the MAP is equal to 1. We need to find a mapping from a temporal discretization to a (nearly) uniform discretization with respect to s. A variational approach was used in [18], which minimizes the following functional

$$E(s) = \int_0^T w^{-1}(t) \left(\frac{ds}{dt}\right)^2 dt, \tag{25}$$

where $w(t)$ is a monitor function chosen as $w(t) = \sqrt{1 + C|\phi_t|^2}$ with C being a positive constant. Note when C goes to infinity, $w(t) \sim |\phi_t|$. The Euler–Lagrange equation of the functional (25) is

$$\begin{cases} \frac{d}{dt}\left(w^{-1}(t)\frac{ds}{dt}\right) = 0, & t \in (0,T), \\ s(0) = 0, & s(T) = 1. \end{cases} \tag{26}$$

For mesh adjustment, we first map the current time mesh to a discretization of $[0,1]$ with respect to s by solving Eq. (26). A quadratic finite element approximation is employed. Specifically, we separate the boundary modes from the interior second-order modes. It is easy to see that the Schur complement for the boundary modes is a tridiagonal matrix which can be inversed with a linear cost. Second, we map a uniform discretization of $[0,1]$ with respect to s to a discretization of $[0,T]$ by computing $t^{-1}(s)$. This will be our new time mesh. Third, we project the current path $\phi_h(t)$ onto the new time mesh. Instead of projecting the current path onto the approximation space given by the whole new time mesh, we implement the projection element-wisely, which corresponds to inverse the mass matrix in each element. We use this strategy mainly for numerical efficiency. The drawback of the element-wise projection is that the coefficients of boundary modes given by two adjacent elements may be different. We currently use the coefficient given by the left element for a certain boundary mode although a certain type of average may yield a better result from the approximation point of view. Numerical experiments show that the nonlinear CG iteration is not sensitive to such a discontinuity since the discontinuity only comes from the element boundaries and the action functional is an integration. To this end, we can continue the nonlinear CG iteration.

We use a low-order finite element approximation to solve Eq. (26) for two reasons: (1) we just need a nearly uniform arc length discretization; (2) $t^{-1}(s)$ can be computed explicitly for a second-order piecewise polynomial. The cost for computing the new mesh is $O(N)$, where N is the number of elements. Since the Cholesky factorization of the local mass matrix in each element can be pre-computed, we only need to inverse N small lower-triangular matrices for the projection. Thus the cost for projection is $O(p^2N)$, where p is the polynomial order in each element. Since $M \approx pN$, the overall cost is $O(pM)$, which is linear with respect to the total number of degrees of freedom.

Remark 1. Due to the choice of finite element approximation, the projection from the old time mesh to the new time mesh is natural. When finite difference discretization is used, an interpolation algorithm, such as spline interpolation, is needed. Our experience shows that the spline interpolation usually introduces extra fluctuations, which can result in a large increase

of the action functional and deteriorate the efficiency of the nonlinear CG iteration. The projection strategy usually does not affect the value of the current action functional even for a low-order finite element approximation.

3.4. Generalization for SPDEs

The proposed MAM can be easily generalized to study random perturbations of stochastic partial differential equations

$$\frac{\partial u(\mathbf{x},t)}{\partial t} = Gu(\mathbf{x},t) + \sqrt{\varepsilon}\dot{W}(\mathbf{x},t), \tag{27}$$

where $\mathbf{x} \in D \subset \mathbb{R}^d$, $d = 1,2,3$, G indicates a differential operator in the physical space and $\dot{W}(\mathbf{x},t)$ is space-time white noise. The action functional for the SPDE (27) is defined as [5]

$$S_T(u) = \frac{1}{2}\int_0^T \int_D (\partial_t u - Gu(\mathbf{x},t))^2 \, d\mathbf{x}\, dt = \frac{1}{2}\langle \partial_t u - Gu, \partial_t u - Gu\rangle_{\mathbf{x},t}, \tag{28}$$

where $\langle f,g\rangle_{\mathbf{x},t}$ indicates the inner product of f and g with respect to both $\mathbf{x}$ and t.

Let $\{h_i(\mathbf{x})\}_{i=1}^{N_\mathbf{x}}$ and $\{\psi_i(t)\}_{i=1}^{N_t}$ be the approximation bases for space and time, respectively. Then $u(\mathbf{x},t)$ has the following approximation

$$u(\mathbf{x},t) \approx u_h(\mathbf{x},t) = \sum_{i=1}^{N_\mathbf{x}} \sum_{j=1}^{N_t} u_{ij} h_i(\mathbf{x})\psi_j(t).$$

We define the perturbation operator $\widehat{G}$ as

$$G(u+\delta u) = Gu + \widehat{G}\delta u + O(\delta^2 u).$$

Then we have

$$S_T(u+\delta u) - S_T(u) = \langle \partial_t u - Gu, \partial_t \delta u - \widehat{G}\delta u\rangle_{\mathbf{x},t},$$

which implies that the gradient with respect to $u_{i,j}$ can be expressed as

$$(\nabla S_T)_{k(i,j)} = \frac{\partial S_T(u_h)}{\partial u_{i,j}} = \langle (\partial_t - G)u_h, (\partial_t - \widehat{G})(h_i(\mathbf{x})\psi_j(t))\rangle_{\mathbf{x},t}, \tag{29}$$

where $i = 1,2,\ldots,N_\mathbf{x}$ and $j = 1,2,\ldots,N_t$. Once the gradient ∇S_T is obtained, it is straightforward to apply the (preconditioned) nonlinear CG optimization solver, described in Section 3.2.

Remark 2. Although the main numerical strategy of MAM for SPDEs is similar with that for SODEs, the large number of degrees of freedom from spatial discretization can make the problem computationally challenging. Note the Euler–Lagrange equation for the action functional of SPDEs takes the form

$$\frac{\delta S_T}{\delta u} = \left(\partial_t - \widehat{G}\right)^* (\partial_t - G)u = 0,$$

where $(\partial_t - \widehat{G})^*$ is the adjoint operator of $\partial_t - G$, which is a high-order nonlinear $(d+1)$-dimensional partial differential equation. Obviously, it is not enough to only consider the parallelization in the time direction, since the number of CPU processors is limited by N_t while $N_\mathbf{x}$ can be large. The study of such an issue will be reported elsewhere.

Algorithm 1: Adaptive high-order MAM

Project the initial path $\phi(t)$ onto a uniform time mesh $\mathcal{T}_h$ of $[0,T]$ and define $\mathbf{\Phi}_0$ which is a global vector containing all unknown coefficients of the finite element approximation;

Start the iteration of nonlinear CG solver (22)

$$\Phi_{k+1} = \Phi_k + \alpha_k \mathbf{d}_k.$$

Check the mesh quality very m iteration steps.

- Compute the arc length for each element according to the monitor function $w(t)$.
- If the ratio r_s between the largest arc length and the smallest one is larger than a prescribed threshold, solve Eq. (26) to obtain a new time mesh.
- Project the current path onto the new time mesh and update $\mathbf{\Phi}_k$.

Stop the CG iteration when error tolerance or the maximum iteration number is achieved.

4. Numerical examples

In this section, we apply the adaptive high-order MAM to two dynamical systems: one is an SODE example and the other one is an SPDE example. For the SODE example we focus on the examination of the accuracy and efficiency according to the exact solution and for the SPDE example we demonstrate a practical application of the method to a complex dynamical system.

4.1. An SODE example

We consider the following example, for which the MAP can be obtained explicitly:

$$\begin{cases} dx = -\partial_x V(x,y)dt + \sqrt{\varepsilon}dW_t^x, \\ dy = -\partial_y V(x,y)dt + \sqrt{\varepsilon}dW_t^y, \end{cases} \tag{30}$$

where the potential $V(x,y)$ is

$$V(x,y) = (1-x^2-y^2)^2 + y^2/(x^2+y^2). \tag{31}$$

The dynamical system has two stable fixed points $a_1 = (-1,0)$ and $a_2 = (1,0)$, which are local minima of the potential $V(x,y)$. We consider the MAP in the upper half-plane connecting a_1 and a_2 through the saddle point $a_3 = (0,1)$. Then the explicit form of this MAP is the upper branch of the unit circle: $x^2 + y^2 = 1$. The exact action functional is $2 \times (V(a_3) - V(a_1)) = 2$. After some simple algebra, we obtain the linear part $\hat{b}(x,y) \in \mathbb{R}^{2\times 2}$ as

$$\begin{aligned}
\hat{b}_{11}(x,y) &= 4(1-3x^2-y^2) + \frac{2y^2}{(x^2+y^2)^2} - \frac{8x^2y^2}{(x^2+y^2)^3}, \\
\hat{b}_{12}(x,y) &= -8xy + \frac{4xy}{(x^2+y^2)^2} - \frac{8xy^3}{(x^2+y^2)^3}, \\
\hat{b}_{21}(x,y) &= -8xy - \frac{4xy}{(x^2+y^2)^2} + \frac{8x^3y}{(x^2+y^2)^3}, \\
\hat{b}_{22}(x,y) &= 4(1-x^2-3y^2) - \frac{2x^2}{(x^2+y^2)^2} + \frac{8x^2y^2}{(x^2+y^2)^3}.
\end{aligned}$$

We first examine the convergence of the proposed method. This example was intensively studied in [18] using an adaptive minimum action method based on the finite difference temporal discretization and an L-BFGS optimization solver. According to the study in [18] we choose the time interval as $[0,T] = [0,50]$ to replace the time interval $[-\infty,\infty]$, which is enough to resolve the problem accurately. The same initial path is used here for the nonlinear CG solver:

$$x(t) = -1 + 2t/T, \quad t \in [0,T], \quad y(t) = \begin{cases} 2.4t/T, & t \in [0,T/4], \\ 0.6, & t \in [T/4, 3T/4], \\ 2.4t/T - 1.2, & t \in [3T/4, T]. \end{cases} \tag{32}$$

Some important parameters are chosen as follows. The mesh quality is checked every 10 iteration steps. The constant C in the monitor function $w(t)$ is set to be 1000, and the threshold for the ratio r_s between the largest arc length and the smallest one is set to be 3.

We define the L_∞ error of the approximate MAP as the maximum value of $\left|\sqrt{x^2(t)+y^2(t)} - 1\right|$ on some Gauss-type quadrature points. In Fig. 1 we plot the h-convergence where we use linear finite elements for approximation and refine the solution by increasing the element number. It is seen that the adaptive minimum action method provides a second-order accuracy which cannot be achieved without the reparametrization of the time mesh [18]. In Fig. 2 we show the p-convergence, i.e., spectral convergence, where we fixed the element number as 10 and refine the approximation by increasing the polynomial order. We obtain an overall p-convergence for both the L_∞ error of the approximate MAP and the error of the approximate action functional. In Fig. 3 we compare the h-convergence and p-convergence with respect to the total number of degrees of freedom using the error of the action functional. It is seen that when the MAP is smooth enough the p-convergence can be much more efficient than the h-convergence.

We subsequently discuss the effect of constant C in the monitor function $w(t)$. We choose polynomial order $p = 5$, element number $N = 50$ and the skewness parameter $r_s = 3$. We test the following two cases: $C = 10^3$ and $C = 10^9$. In Fig. 4 we plot the two final adaptive meshes. It is seen that the time interval can be roughly decomposed as $[0,50] = [0,10] \cup [10,12.5] \cup [12.5,37.5] \cup [37.5,50]$. In the region $[0,10] \cup [12.5,37.5] \cup [37.5,50]$, the particle stays around the fixed points and has a very small velocity. Thus a longer time is needed to make the arc length close to $L/50$, where L is the arc length of the MAP. Since when C goes to infinity, $w(t) \sim |\phi_t|$, we then have a more uniform mesh with respect to arc length when C is larger. It is observed that fewer points are required in the transition regions for a larger C. Although such a strategy can improve the accuracy significantly, this does not mean that the accuracy is a monotonically decreasing function of C. Actually, the

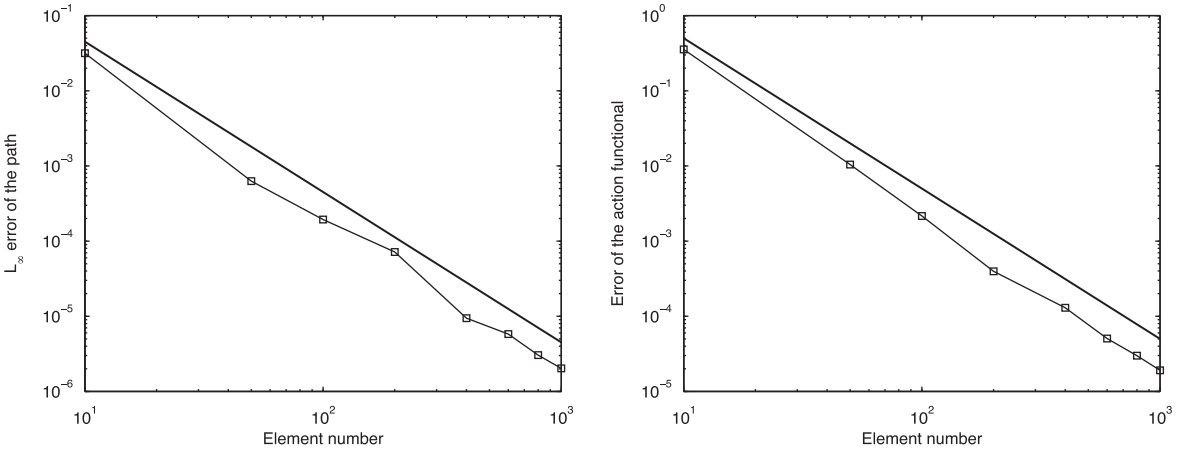

Fig. 1. *h*-convergence for the SODE example. Linear finite elements are used for approximation. The straight lines are reference lines with slopes −2. Left: the L_∞ error of the MAP versus the element number. Right: the error of the action functional versus the element number.

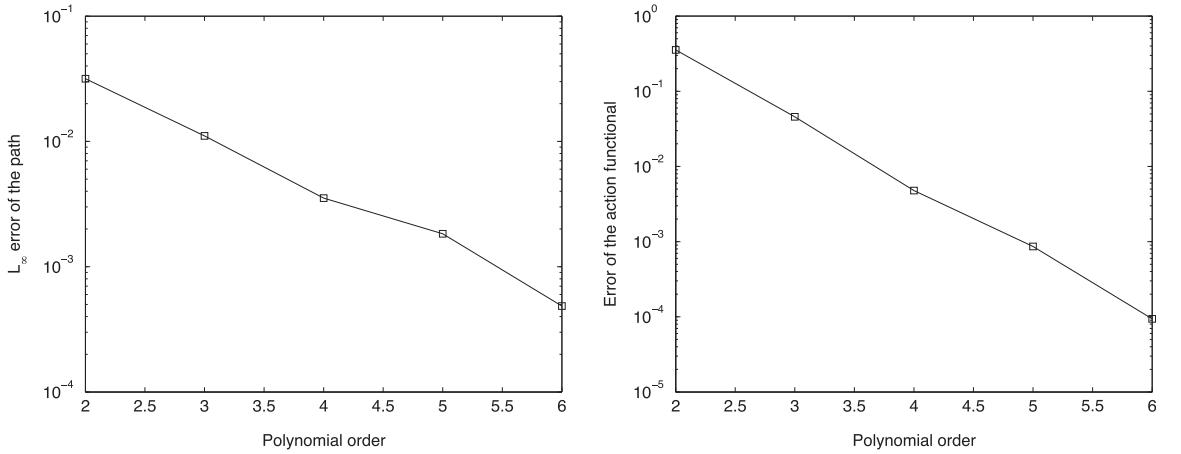

Fig. 2. *p*-convergence for the SODE example. 10 finite elements are used for approximation. Left: the L_∞ error of the MAP versus the polynomial order. Right: the error of the action functional versus the polynomial order.

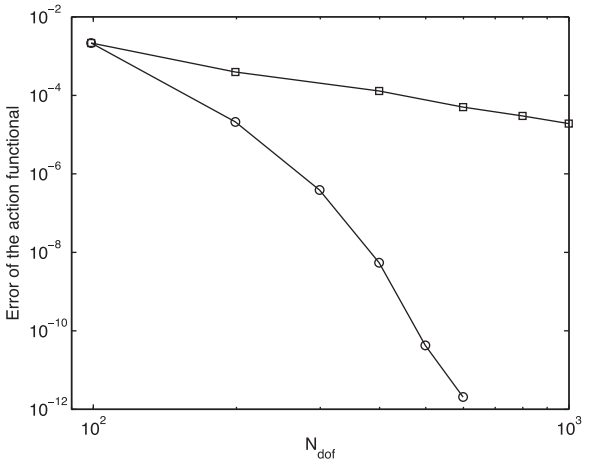

Fig. 3. Convergence with respect to the number of degrees of freedom. The line with squares is given by *h*-convergence where linear finite elements are used; the line with circles is given by *p*-convergence where 100 elements are used.

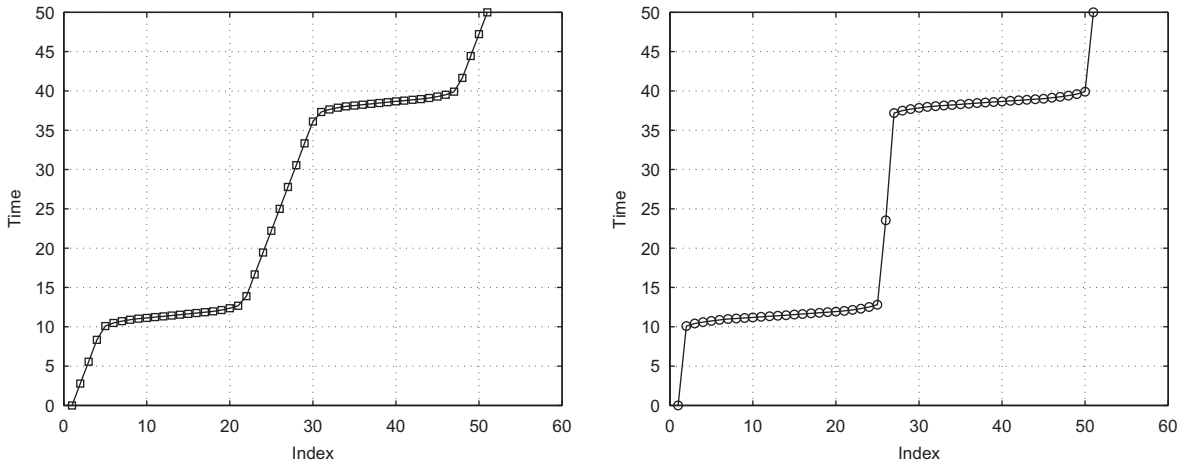

Fig. 4. Two final adaptive time meshes. $p = 5$ and $N = 50$. Left: $C = 10^3$; Right: $C = 10^9$.

errors of the action functional are $O(10^{-6})$ and $O(10^{-3})$ for $C = 10^3$ and $C = 10^9$, respectively. This is mainly due to the fact that our approximation is with respect to time. It is obvious that for a certain polynomial order, the approximation error will become larger as the element size increases. Thus, the optimal choice of the constant C is problem dependent. A more efficient and robust monitor function deserves further study to balance the mesh adjustment and the approximation errors in each element.

4.2. An SPDE example

We now apply the developed method to random perturbations of the normalized Kuramoto–Sivashinsky (K–S) equation defined as

$$\begin{cases} u_t + 4u_{xxxx} + \alpha\left[u_{xx} + \frac{1}{2}(u_x)^2\right] = 0, \quad (x,t) \in (0,2\pi) \times \mathbb{R}_+, \\ u(x,0) = u_0(x), \quad u(x,t) = u(x+2\pi,t), \end{cases} \tag{33}$$

where α is a bifurcation parameter. This is one of the simplest spatially extended nonlinear systems of physical interest. It was derived in 1970s by Kuramoto [15] and Sivashinsky [17]. Holmes et al. [11] provided a delightful discussion of why this system is a good model problem for studying turbulence in the full-fledged Navier–Stokes boundary shear flows. As the bifurcation parameter α increases, the solution of K–S equation undergoes a complex bifurcation sequence including fixed points, traveling waves, invariant tori and homoclinic orbits [10] on the route of transition to chaos.

Eq. (33) is characterized by a second order *negative diffusion* (due to the sign of the Laplacian), a fourth-order stabilizing term and a quadratic nonlinear coupling term. For example, the trivial solution $u \equiv 0$ is a global attractor when $\alpha \leqslant 4$, and becomes unstable as soon as the bifurcation parameter α is larger than 4. This is reflected in the spectrum of (33) linearized at $u = 0$, where the eigen-functions are the $\exp(\mathrm{i}kx)$, $k \in \mathbb{Z}$, and the sequence of eigenvalues is

$$\Lambda_k = \alpha k^2 - 4k^4.$$

When α is large enough, the K–S equation will exhibit temporal chaos. On the other hand, the existence of a unique compact inertial manifold was establish in [6]. Thus, the solution of the K–S equation is characterized by the coexistence of coherent spatial structures with complex temporal dynamics. Since the mean mode

$$\bar{u}(t) = \frac{1}{2\pi}\int_0^{2\pi} u(x,t)\,dx$$

is unbounded, we consider the equation for $v(x,t) = u(x,t) - \bar{u}(t)$ for convenience [10]:

$$v_t + 4v_{xxxx} + \alpha\left[v_{xx} + \frac{1}{2}(v_x)^2\right] + \dot{\bar{u}}(t) = \sqrt{\varepsilon}\dot{W}(x,t), \tag{34}$$

where

$$\dot{\bar{u}}(t) = \frac{-\alpha}{4\pi}\int_0^{2\pi}(u_x)^2\,dx = \frac{-\alpha}{4\pi}\int_0^{2\pi}(v_x)^2\,dx.$$

Define the operator G in the action functional (28) as

$$Gv(x,t) = -4v_{xxxx} - \alpha\left[v_{xx} + \frac{1}{2}(v_x)^2\right] + \frac{\alpha}{4\pi}\int_0^{2\pi}(v_x)^2\,dx. \tag{35}$$

Then the perturbation operator $\widehat{G}$ is

$$\widehat{G}\delta v(x,t) = -4\delta v_{xxxx} - \alpha[\delta v_{xx} + v_x\delta v_x] + \frac{\alpha}{2\pi}\int_0^{2\pi} v_x\delta v_x\,dx. \tag{36}$$

Due to the periodic boundary conditions in space, we choose the Fourier spectral method for the spatial discretization to take advantage of the Fast Fourier Transform (FFT). In other words, the approximation space for x is defined as

$$B_{N_x} = \mathrm{span}\{\cos(mx), 0 \leqslant m \leqslant N_x/2\} \cup \{\sin(mx),\ 1 \leqslant m \leqslant N_x/2 - 1\}, \tag{37}$$

where N_x is an even integer and B_{N_x} has dimension N_x. We order the basis functions as follows

$$1, \cos(x), \cos(2x), \ldots, \cos\left(\frac{N_x}{2}x\right), \sin\left(\left(\frac{N_x}{2} - 1\right)x\right), \ldots, \sin(x)$$

and denote them as $h_m(x)$, $1 \leqslant m \leqslant N_x$. Note here that since the mean $\bar{u}(t)$ is subtracted, the coefficient for the mean mode is always zero. Then the MAP has an expansion

$$v(x,t) = \sum_{m=1}^{N_x}\sum_{j=1}^{N_t} v_{m,j}h_m(x)\psi_j(t).$$

To evaluate $S_T(v)$ and $\partial S_T/\partial v_{m,j}$ exactly in the x-direction, $2N_x$ points are needed for the FFT due to the nonlinearity.

4.2.1. A practical preconditioner

The Euler–Lagrange equation of the action functional

$$\left(-\partial_t^2 + \alpha^2\partial_x^4 + 8\alpha\partial_x^6 + 16\partial_x^8\right)v + \mathcal{N}(v) = 0, \tag{38}$$

where $\mathcal{N}(v)$ is the non-linear part. From the linear part of the Euler–Lagrange equation (38), we see that the ratio between the largest and smallest eigenvalues is $C_k = (\alpha k^2 - 4k^4)^2/(\alpha - 4)^2$, where $k \in \mathbb{Z}$ and the smallest absolute value of k is equal to 1 since the mean mode is taken out. When 32 Fourier modes are employed, $C_{16} = O(10^7)$, which implies that the condition number of the Hessian can be very large. Thus an efficient preconditioner is necessary to improve the convergence of the nonlinear CG iteration.

We use the inverse of the linear part of the Euler–Lagrange equation, i.e. $\left(-\partial_t^2 + \alpha^2\partial_x^4 + 8\alpha\partial_x^6 + 16\partial_x^8\right)^{-1}$, as the preconditioner for the nonlinear CG method. In other words, for a certain $v(x,t)$, we need to solve the following equation

$$\begin{cases} \partial_t^2\hat{v} = \alpha^2\partial_x^4\hat{v} + 8\alpha\partial_x^6\hat{v} + 16\partial_x^8\hat{v} - \frac{\delta S_T}{\delta v}(v), \quad (x,t) \in (0,2\pi)\times\mathbb{R}_+, \\ \hat{v}(x,0) = \hat{v}(x,T) = 0, \quad \hat{v}(x,t) = \hat{v}(x+2\pi,t), \end{cases} \tag{39}$$

where $\delta S_T/\delta v$ is the functional derivative of $v(x,t)$. Homogeneous boundary conditions are used such that they will not affect the force term. Let

$$\hat{v} = \sum_{m=1}^{N_x/2}\hat{v}_{a,m}(t)\cos(mx) + \sum_{m=1}^{N_x/2-1}\hat{v}_{b,m}(t)\sin(mx). \tag{40}$$

Taking Galerkin projection of Eq. (39) with respect to $\cos(nx)$ or $\sin(nx)$, we obtain a system of decoupled one-dimensional wellposed elliptic equations

$$\frac{d^2}{dt^2}\hat{v}_{a,m} = (\alpha^2m^4 - 8\alpha m^6 + 16m^8)\hat{v}_{a,m} - \left\langle\frac{\delta S_T}{\delta v}, \cos(mx)\right\rangle_x, \quad 1 \leqslant m \leqslant N_x/2, \tag{41}$$

$$\frac{d^2}{dt^2}\hat{v}_{b,m} = (\alpha^2m^4 - 8\alpha m^6 + 16m^8)\hat{v}_{b,m} - \left\langle\frac{\delta S_T}{\delta v}, \sin(mx)\right\rangle_x, \quad 1 \leqslant m < N_x/2, \tag{42}$$

by noting that $\alpha^2m^4 - 8\alpha m^6 + 16m^8 = m^4(\alpha - 4m^2)^2 \geqslant 0$. Then Eqs. (41) and (42) can be solved one by one.

Let

$$\hat{v}(x,t) = \sum_{m=1}^{N_x}\hat{v}_m(t)h_m(x) = \sum_{m=1}^{N_x}\sum_{j=1}^{N_t}\hat{v}_{mj}h_m(x)\psi_j(t).$$

With respect to $h_m(x)$, we rewrite Eqs. (41) and (42) as

$$\frac{d^2}{dt^2}\hat{v}_m(t) = c_m\hat{v}_m(t) - \left\langle\frac{\delta S_T}{\delta v}, h_m(x)\right\rangle_x. \tag{43}$$

We then use the finite element space $\{\psi_j(t)\}_{j=1}^{N_t}$ to approximate Eq. (43) corresponding to the weak form

$$\sum_{j=1}^{N_t} \hat{v}_{m,j} \int_0^T \left(\frac{d\psi_j}{dt} \frac{d\psi_l}{dt} + c_m \psi_j \psi_l \right) dt = \frac{\partial S_T}{\partial v_{m,j}}, \quad 1 \leqslant l \leqslant N_t, \tag{44}$$

by noting that

$$\left\langle \frac{\delta S_T}{\delta v}, h_m(x)\psi_j(t) \right\rangle_{x,t} = \frac{\partial S_T}{\partial v_{m,j}}.$$

Using a similar technique as we did for the mesh adjustment, we separate the boundary modes from the interior modes. The Schur complement for the boundary modes is a tridiagonal matrix and the matrix for the interior modes is block-wisely diagonal. The cost for the preconditioning step is thus $O((N + p^2 N)N_x) \sim O(pN_tN_x)$, which is linear with respect to the total number of degrees of freedom.

4.2.2. Boundaries of the MAP

We choose $\alpha = 52$, for which there exist two types of attractors: one is a fixed point $v_F(x)$ and the other one is a traveling wave $v_{TW}(x + ct)$, shown in Fig. 5. We consider the transition from the fixed point $v_F(x)$ to the traveling wave $v_{TW}(x + ct)$ using the proposed method. Due to the even-order spatial derivatives and quadratic nonlinearity, the K–S equation is invariant with respect to translation and reflection. We refer to [21] for discussions about the relation between the boundaries of the MAP and the equivalent solutions of K–S equation. We here focus on the performance of the proposed numerical method and the boundaries of the MAP are chosen as snapshots of the two attractors.

The transition from $v_F(x)$ to $v_{TW}(x + ct)$ was studied in [21] based on the following ODE system

$$\frac{d\mathbf{v}}{dt} = \mathcal{P}_{N_x}\mathbf{L}(\mathbf{v}) + \mathcal{P}_{N_x}\mathbf{N}(\mathbf{v}, t) + \sqrt{\varepsilon}\dot{W}(t), \tag{45}$$

where $\mathbf{v} \in \mathbb{R}^{N_x - 1}$ indicates the Fourier coefficients of $v(x,t)$, $\mathcal{P}_{N_x}\mathbf{L}(\mathbf{v})$ is the projection of the linear operator $\mathbf{L}(v) = -4v_{xxxx} - \alpha v_{xx}$ onto B_{N_s}, and $\mathcal{P}_{N_x}\mathbf{N}(\mathbf{v}, t)$ the similar projection of the nonlinear operator $\mathbf{N}(v) = -\frac{\alpha}{2} v_x^2 - \dot{u}(t)$. Eq. (45) corresponds to the action functional

$$\widehat{S}_T(\mathbf{v}) = \frac{1}{2} \int_0^T |\dot{\mathbf{v}} - \mathcal{P}_{N_x}\mathbf{L}(\mathbf{v}) - \mathcal{P}_{N_x}\mathbf{N}(\mathbf{v}, t)|^2 \, dt. \tag{46}$$

In this work, we consider the action functional (28) directly without implementing the projection of the original stochastic PDE. Note that when we implement the Galerkin projection of the original stochastic PDE, the higher-order modes given by the nonlinear operator will be eliminated. Thus for a stable fixed point $\hat{v}(x)$ of Eq. (45), $\|\partial_t \hat{v} - G\hat{v}\|^2$ may be not exactly equal to zero, where $\|\cdot\|$ indicates the L_2 norm in space. However since

$$\|\partial_t \hat{v}(x) - G\hat{v}(x)\|^2 = \pi |\dot{\hat{\mathbf{v}}} - \mathcal{P}_{N_x}\mathbf{L}(\hat{\mathbf{v}}) + \mathcal{P}_{N_x}\mathbf{N}(\hat{\mathbf{v}}, t)|^2 + \epsilon,$$

where ϵ is the contribution from the higher-order modes given by the nonlinear operator $\mathbf{N}(\hat{v})$, which is positive and small, the MAP given by a direct approximation of the action functional (28) should be close to the MAP of the ODE system (45) when the number of Fourier modes is large enough. Furthermore, if $\|\partial_t \hat{v}(x) - G\hat{v}(x)\| = 0$, $\hat{\mathbf{v}}$ must be a fixed point of Eq. (45).

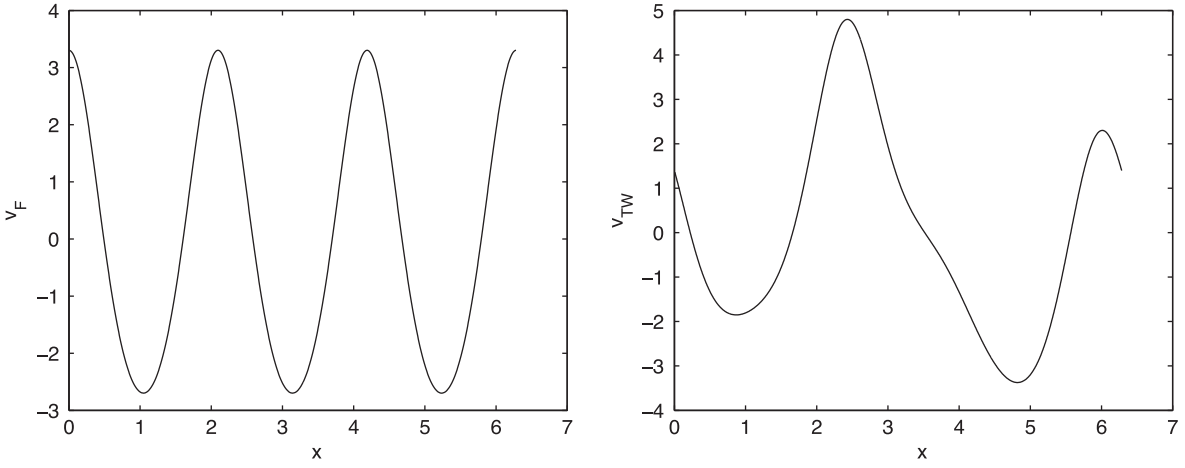

Fig. 5. Profiles of the attractors. Left: fixed point, $v_F(x)$; Right: traveling wave, $v_{TW}(x + ct)$.

4.2.3. Numerical results

We first look at the effectiveness of the proposed preconditioner (39). We use 500 linear finite elements for temporal discretization and 32 Fourier modes for spatial discretization. The mesh quality is checked every 50 iteration steps. The constant C in the monitor function $w(t)$ is set to be 1000, and the threshold for the ratio r_s between the largest arc length and the smallest one is set to be 10. In Fig. 6, we plot the convergence behavior of the nonlinear CG iterations. The solid line corresponds to the nonlinear CG solver without preconditioner, and the dash line is given by the preconditioned nonlinear CG solver. We first notice that the value of the action functional keeps decreasing, which means that the projection of the MAP from the old time mesh to the new time mesh does not result in any significant increase of the action functional. Second, we see that the preconditioner significantly improve the efficiency of the nonlinear CG iteration, which implies that the preconditioner is an effective approximation of the Hessian.

We subsequently look at the transition from the fixed point $v_F(x)$ to the traveling wave $v_{TW}(x+ct)$. In this work we use the stable solutions given by the ODE system (45), see Fig. 5, as the boundaries of the MAP, since the boundaries of the MAP can be two arbitrary points or sets in the phase space and are not restricted to stable solutions. In Fig. 7 we compare the MAPs given by the action functional (28) and the projected ODE system (45), respectively. On the time interval [0,30], 3000 grid points are used for discretization. For the action functional (28), the linear finite element discretization is used; for the action

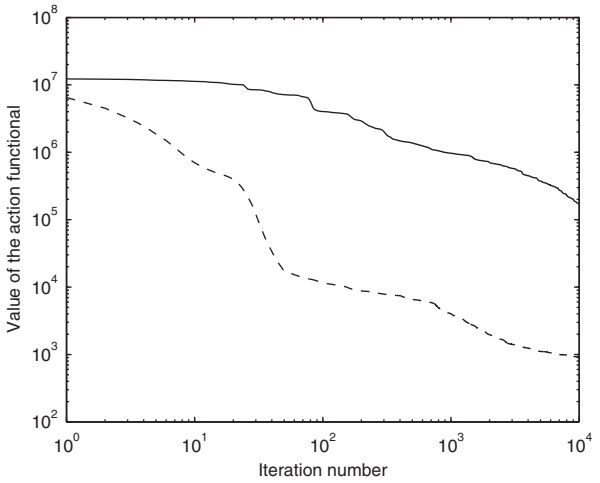

Fig. 6. Convergence behavior of the nonlinear CG iterations. The solid line is given by a nonlinear CG solver without preconditioner; The dash line is given by the preconditioned nonlinear CG solver.

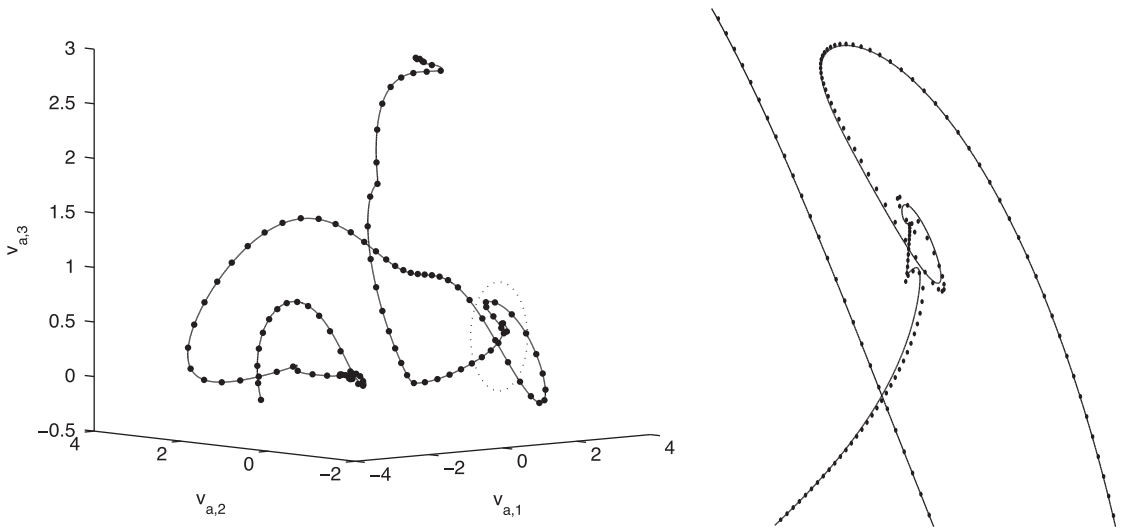

Fig. 7. MAP of the K–S equation between $v_F(x)$ and $v_{TW}(x+ct)$ discretized by 3000 points in the time interval [0,30]. The solid line is given by the minimum action method based on the linear finite element discretization of action functional (28), and the dotted line is given by the minimum action method based on the finite difference discretization of action functional (46). Left: MAP in the phase space with respect to the coefficients of $\cos(x)$, $\cos(2x)$ and $\cos(3x)$; Right: A close-up view of the region marked by the dotted ellipse in the left figure.

functional (46) given by the projected ODE system (45), the finite difference discretization is used. 32 Fourier modes are used for the physical discretization. It is seen that the local minimizer of the action functional (28) agrees very well with that of the action functional (46). Although the positions of fixed points are slightly different due to the Galerkin projection, the dynamical properties remain the same. In other words, if we want to study the phase space of the ODE equation (45), we can also use the MAP given by the action functional (28). We refer to [21] for a detailed study of the configuration space of the K–S equation based on the minimal action path.

We now look at the numerical efficiency. We know that the MAP approaches saddle points from their stable manifold and leaves from their unstable manifold, which implies that the MAP may just have C^0 regularity at saddle points with respect to the arc length. From the numerical point of view, we need to rely on *h*-convergence, i.e., small elements, around the saddle points to avoid the Gibbs phenomena in spectral methods. In [21], we identified five zones along the MAP, where saddle points are found, which is the reason that we only consider linear finite element discretization for this problem. However, we also observe that between saddle points the MAP usually has a much better regularity, which means that higher-order polynomial approximation can be more efficient in such a region. In other words, *hp*-adaptivity can be implemented to further improve the numerical efficiency. In this work, we focus on the general methodology and only consider a uniform polynomial order in each element. The *hp*-adaptivity will remain as a research issue for future study.

5. Summary and discussions

In this work we formulate the minimum action method in the framework of finite element method. This version of the minimum action method has the following main features: (1) An *hp* finite element space is used for the approximation of the MAP, which allows both *h*- and *p*-refinement. (2) Formulas for the computation of the gradient of the action functional are simple and can be easily coupled with finite element approximations or spectral approximations in the physical space for a general partial differential equation. (3) In the mesh adjustment, the mapping of the MAP from the old time mesh to the new one based on projection appears to introduce no significant increase of the action functional.

Overall the current version of the minimum action method seems more flexible and robust than the minimum action methods based on the finite difference discretization and interpolation, especially for the study of complex dynamical systems modeled by partial differential equations. We currently use the same polynomial order in each finite element. However, the study of the configuration space of the Kuramoto–Sivashinsky equation shows that the *hp* adaptivity is necessary to further improve the efficiency of the method. If the minimum action method is formulated with respect to time, it was shown that the time mesh adjustment is important for the numerical accuracy and efficiency [18]. However, more understanding about the balance between the uniformness of arc length elements and the approximation errors of the MAP with respect to time is still required to further improve the robustness of the monitor function for the time mesh adjustment.

Acknowledgements

This work was supported by DOE Grant SC0002324 and by NSF Grant DMS-1115632. The author would like to thank Weinan E and Xiang Zhou for helpful discussions.

References

[1] C. Cerjan, W. Miller, On finding transition states, J. Chem. Phys. 75 (6) (1981) 2800–2806.
[2] W. E, W. Ren, E. Vanden-Eijnden, String method for the study of rare events, Phys. Rev. B 66 (2002) 052301.
[3] W. E, W. Ren, E. Vanden-Eijnden, Minimum action method for the study of rare events, Commun. Pure Appl. Math. 57 (2004) 637–656.
[4] W. E, W. Ren, E. Vanden-Eijnden, Simplified and improved string method for computing the minimum energy paths in barrier-crossing events, J. Chem. Phys. 126 (2007) 164103.
[5] W. Faris, G. Jona-Lasinio, Large fluctuations for a nonlinear hear equation with noise, J. Phys. A: Math. Gen. 15 (1982) 3025–3055.
[6] C. Foias, B. Nicolaenko, G. Sell, R. Temam, Inertial manifolds for the Kuramoto–Sivashinsky equation and an estimate of their lowest dimensions, J. Math. Pure Appl. 67 (1988) 197–226.
[7] W. Hager, H. Zhang, A new conjugate gradient method with guaranteed descent and an efficient line search, SIAM J. Optim. 16 (1) (2005) 170–192.
[8] M. Heymann, E. Vanden-Eijnden, The geometric minimum action method: a least action principle on the space of curves, Commun. Pure Appl. Math. 61 (2008) 1052–1117.
[9] G. Henkelman, H. Jónsson, A dimer method for finding saddle points on high dimensional potential surfaces using only first derivatives, J. Chem. Phys. 111 (15) (1999) 7010–7022.
[10] J. Hyman, B. Nicolaenko, The Kuramoto–Sivashinsky equation: a bridge between PDE's and dynamical systems, Physica D 18 (1–3) (1986) 113–126.
[11] P. Holmes, J. Lumley, G. Berkooz, Turbulence, Coherent Structures, Dynamical Systems and Symmetry, Cambridge University Press, 1998.
[12] H. Jònsson, G. Mills, K. Jacobsen, Nudged elastic band method for finding minimum energy paths of transitions, in: B. Berne, G. Ciccotti, D. Coker (Eds.), Classical and Quantum Dynamics in Condensed Phase Simulations, 1998.
[13] N. van Kampen, Stochastic Processes in Physics and Chemistry, North-Holland, 1981.
[14] G. Karniadakis, S. Sherwin, Spectral/hp Element Methods for Computational Fluid Dynamics, second ed., Oxford University Press, 2005.
[15] Y. Kuramoto, Diffusion-induced chaos in reactions systems, Suppl. Prog. Theor. Phys. 64 (1978) 346–367.
[16] L. Onsager, S. Machlup, Fluctuations and irreversible processes, Phys. Rev. 91 (1953) 1505–1512.
[17] G. Sivashinsky, Nonlinear analysis of hydrodynamic instability in laminar flames. Par I: deviation of basic equations, Acta Astron. 4 (1977) 1177–1206.
[18] X. Zhou, W. Ren, W. E, Adaptive minimum action method for the study of rare events, J. Chem. Phys. 128 (2008) 104111.
[19] X. Zhou, W. E, Study of noise-induced transitions in the Lorenz system using the minimum action method, Comm. Math. Sci. 8 (2) (2010) 341–355.
[20] W. Ren, Numerical methods for the study of energy landscape and rare events, Ph.D. Thesis, New York University, 2002.
[21] X. Wan, X. Zhou, W. E, Study of the noise-induced transition and the exploration of the configuration space for the Kuramoto–Sivashinsky equation using the minimum action method, Nonlinearity 23 (2010) 475–493.