A trust region-CG algorithm for deblurring problem in atmospheric image reconstruction

WANG Yanfei , YUAN Yaxiang & ZHANG Hongchao

State Key Laboratory of Scientific and Engineering Computing, Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and System Sciences, Chinese Academy of Sciences, Beijing 100080, China
Correspondence should be addressed to Wang Yanfei (email: wyf@lsec.cc.ac.cn)

Received October 22, 2001

Abstract In this paper we solve large scale ill-posed problems, particularly the image restoration problem in atmospheric imaging sciences, by a trust region-CG algorithm. Image restoration involves the removal or minimization of degradation (blur, clutter, noise, etc.) in an image using *a priori* knowledge about the degradation phenomena. Our basic technique is the so-called trust region method, while the subproblem is solved by the truncated conjugate gradient method, which has been well developed for well-posed problems. The trust region method, due to its robustness in global convergence, seems to be a promising way to deal with ill-posed problems.

Keywords: **image reconstruction, ill-posed problems, truncated CG, trust region.**

1 Preliminary

Image restoration involves the removal or minimization of degradation (blur, clutter, noise, etc.) in an image using *a* knowledge about the degradation phenomena. This kind of problem receives much attention recently[1−3].

Astronomical images obtained, for example, from ground-based telescopes are usually corrupted or distorted by blurring due to atmospheric turbulence and noise[1]. Blind restoration is the process of estimating both the true image and the blur from the degraded image characteristics, using only partial information about degradation sources and the imaging system. Our main interest concerns optical image enhancement, where the degradation involves a convolution process.

Scientists and engineers are actively attempting to overcome the degradation of astronomical image quality caused by the effects of atmospheric turbulence and other image degradation processes.

In principle, if a sophisticated model of the scattering process is available, the true image can be reconstructed from the photo by solving the associated inverse problem. Such models, however, are very difficult to derive, because atmospheric turbulences are hard to predict and currently can only be accessed via stochastical processes[4].

The usual approach we applied is as follows: First we formulate a mathematical model (called the forward model) relating the image data to the object and the atmospheric distortion and then

solve the inverse problem, i.e. we estimate the object and certain features of the atmosphere given observed image data and the forward model.

A simple forward model is the first kind of convolution integral equation

$$\begin{aligned} h(x,y) &= \int\int k(x-\xi, y-\eta) f(\xi,\eta) d\xi d\eta + e(x,y) \\ &= (k \star f)(x,y) + e(x,y). \end{aligned} \tag{1}$$

Here h represents image data, f the object (the true image), e noise in the data, and k is known as the point spread function (PSF). The PSF is the image that results from an idealized point object-guided star, characterizing atmospheric blurring effects. According to the space-invariance of the imaging process, a guided star image is essentially the convolution k with a delta distribution, and therefore provides an approximation of the values of k.

Problem (1) is ill-posed. Assume that the PSF k is known. Then the process of estimating f from h is called deconvolution. In ideal case, i. e. the error $e = 0$, one can utilize convolution theorem to obtain the Fourier transform of the object at a given spatial frequency vector $\omega = (\omega_1, \omega_2)$,

$$F(f)(\omega) = \frac{F(h)(\omega)}{F(k)(\omega)}. \tag{2}$$

The object can then be restored by applying the inverse Fourier transform (IFFT) to the r.h.s. of (2). However, if the PSF k is smooth, then its Fourier transform tends to zero at high spatial frequencies. Morerover, if the data h contains error, then its Fourier transform does not decay rapidly as ω is large, hence the r.h.s. of (2) becomes large and instability occurs. Now a question arises: how to overcome this problem?

Regularization is such a technique that can maintain stability when solving an approximate model of the original problem. However, as is well known, this kind of method involves a parameter, known as regularization parameter, which quantifies the trade-off between error amplification due to instability and truncation due to regularization. The choice of an appropriate regularization parameter is a subtle thing. Unless the parameter is appropriately chosen, the solution will not be a good approximation to the true solution of the original problem.

Facing these difficulties, in this context, we will solve the problem in a different way. First we transform this problem into an optimization problem:

$$\min M[f] = \|Kf - h\|^2, \tag{3}$$

and then solve f by a trust region method. The trust region method, due to its robustness in global convergence, seems a suitable method for solving such problems. We are free from the careful choice of the regularization parameter by using this method.

2 Tikhonov regularization method

If we let K denote the blurring operator and e the noise process, then the image restoration problem with additive noise can be expressed as a linear operator equation

$$h = Kf + e, \tag{4}$$

where h and f denote functions containing the information of the recorded and original images respectively.

As a rule, the numerical solution of the fundamental problem would be impossible without the use of computers. Therefore we may assume that a discrete model of (4) is obtained (note that there are a lot of quadrature rules to discretize (4)), which we denote by

$$h = \mathcal{K}f + e. \tag{5}$$

For notation purpose we assume that the image is n-by-n, where n is the number of pixels in each direction, and thus contains n^2 pixels, and accordingly $\mathcal{K}$ is $n^2 \times n^2$. Typically n is chosen to be a power of 2, such as 256 or larger, so the number of unknowns grows to at least 65536. The vectors $f \in R^{n^2}$ and $h \in R^{n^2}$ represent the true and observed image pixel values, respectively, in a row-wise ordering. Due to the special structure of Problem (1), the matrix $\mathcal{K}$ is a block Toeplitz matrix with Toeplitz blocks (abbreviated to BTTB), each subblock of $\mathcal{K}$ is an $n \times n$ Toeplitz matrix. Thus the fast Fourier transform (FFT) can be used in computations involving $\mathcal{K}$.

It has been verified that the eigenvalues of $\mathcal{K}$ cluster at the origin so that $\mathcal{K}$ has a really large condition number[4]. As already mentioned in sec. 1, the solution f of the discrete linear system (5) is very sensitive to the measurement errors e resulting from imaging process. To overcome this ill-conditioning, regularization technique has to be employed. Certainly, Tikhonov regularization[5,6] is the most popular one. In this method, one solves the minimization problem

$$\min \|\mathcal{K}f - h\|^2 + \alpha\|Lx\|^2, \tag{6}$$

where $\alpha > 0$ is the so-called regularization parameter, L is the penalty matrix (often a positive definite matrix). For example, L can be chosen as the identity matrix I in L_2 or as the discrete negative Laplacian matrix in Sobolev space H^1. In Tikhonov regularization, the choice of regularization parameter α is a critical and delicate thing. It cannot be too large or too small. If α is too large, its solution may be far from the noise-free solution, since the new problem is a poor approximation to the original problem; if α is too small, the influence of the data errors may cause instabilities. The optimal balance between these two extremes is so hard to obtain that in practice α is chosen in an *a* way, on the basis of preliminary experiments and experience.

Clearly the minimization problem (6) is equivalent to solving the following Euler equation:

$$(\mathcal{K}^*\mathcal{K} + \alpha L^*L)f = \mathcal{K}^*h. \tag{7}$$

Because of the large size of matrix $\mathcal{K}$, direct methods should be avoided. Usually, in order to utilize the special structure of the Toeplitz matrix $\mathcal{K}$, the matrix $\mathcal{K}$ is replaced by a block-circulant-circulant-block (BCCB) matrix $\mathcal{K}$, which coincides with $\mathcal{K}$ in all central block diagonals and the central diagonals of all blocks (see ref. [3] for details). Note that the inverse of a BCCB matrix is also BCCB, hence the new system can be solved by using a two-dimensional fast Fourier transform (2-D FFT).

3 A trust region-CG algorithm

Trust region methods are a group of methods for ensuring global convergence while retaining fast local convergence in optimization algorithms. For example, we consider the minimization problem

$$\min_{x \in R^n} f(x). \tag{8}$$

In trust region methods, we first choose a trial step length $\quad_c$, and then use the quadratic model to select the best step of (at most) this length for the quadratic model by solving

$$\min \psi(x_c + \xi) = f(x_c) + (g(x_c), \xi) + \frac{1}{2}(H_c\xi, \xi), \tag{9}$$

$$s.\ t.\ \|\xi\| \leqslant \quad_c. \tag{10}$$

The trial step length $\quad_c$ is considered as an estimate of how far we trust the quadratic model, hence it is called a trust radius and the resultant method is called a trust region method.

In this section, we will consider the approximation minimization problem (3) by utilizing the trust region technique mentioned above.

In discrete case, (3) can be rewritten as

$$\min \mathcal{M}[f] = \|\mathcal{K}f - h\|^2. \tag{11}$$

The gradient and Hessian of the objective functional $\mathcal{M}[f]$ can be computed explicitly as

$$\mathrm{grad}(\mathcal{M}[f]) = \mathcal{K}^*\mathcal{K}f - \mathcal{K}^*h, \quad \mathrm{Hess}(\mathcal{M}[f]) = \mathcal{K}^*\mathcal{K}.$$

Our trust region algorithm for (11) involves solving the following subproblem (TRS):

$$\min \phi(s) = (\mathrm{grad}(\mathcal{M}[f]), s) + \frac{1}{2}(\mathrm{Hess}(\mathcal{M}[f])s, s), \tag{12}$$

$$s.t. \|s\| \leqslant \quad . \tag{13}$$

At each iteration of a trust region algorithm, a problem in the form of (12)-(13) has to be solved exactly or inexactly to obtain a trial step. The trial step, often called the trust region step, will either be accepted or rejected after testing some test condition based on the predicted reduction and the actual reduction of the objective function. Let s_k be a solution of (12)-(13). The predicted reduction is defined by the reduction in the approximate model,

$$\mathrm{Pred}_k = \phi_k(0) - \phi_k(s_k) = -\phi_k(s_k). \tag{14}$$

Unless the current point s_k is a stationary point and $\mathrm{Hess}(\mathcal{M}[f])$ is positive semi-definite, the predicted reduction is always positive. The actual reduction is the reduction in the objective function

$$\mathrm{Ared}_k = \mathcal{M}[f_k] - \mathcal{M}[f_k + s_k]. \tag{15}$$

And we define the ratio between the actual and the predicted reduction by

$$r_k = \frac{\mathrm{Ared}_k}{\mathrm{Pred}_k}, \tag{16}$$

which is used to decide whether the trial step is acceptable and to adjust the new trust region radius.

With the above analysis, we construct the trust region algorithm for solving image restoration problem as follows.

Algorithm 3.1 (Trust region algorithm for image restoration problem).

Step 1. Given the initial guess value $f_1 \in R^{n^2}$, $\quad_1 > 0$, $0 < \tau_3 < \tau_4 < 1 < \tau_1$, $0 \leqslant \tau_0 \leqslant \tau_2 < 1$, $\tau_2 > 0$, $k := 1$.

Step 2. If the stopping rule is satisfied, then STOP; Else, solve (12)—(13) giving s_k.

Step 3. Compute r_k;

$$f_{k+1} = \begin{cases} f_k & \text{if} \quad r_k \leqslant \tau_0, \\ f_k + s_k & \text{otherwise;} \end{cases}$$

Choose $\quad_{k+1}$ that satisfies

$$_{k+1} = \begin{cases} [\tau_3\|s_k\|, \tau_4 \quad_k] & \text{if} \quad r_k < \tau_2, \\ [\quad_k, \tau_1 \quad_k] & \text{otherwise.} \end{cases}$$

Step 4. Evaluate $\mathrm{grad}(\mathcal{M}[f_k])$; $k := k+1$; GOTO Step 2.

The constant τ_i $(i = 0, \cdots, 4)$ can be chosen by users. Typical values are $\tau_0 = 0$, $\tau_1 = 2$, $\tau_2 = \tau_3 = 0.25$, $\tau_4 = 0.5$. For other choices of those constants, please see refs. [7—10], etc. The parameter τ_0 is usually zero[7,11], or a small positive constant[12,13]. The advantage of using zero τ_0 is that a trial step is accepted whenever the objective function is reduced. Hence it would not throw away a "good point" which is a desirable property especially when the function evaluations are very expensive.

In Step 2, the stopping rule is based on the predicted reduction Pred_k. In our algorithm, if $\mathrm{Pred}_k \leqslant \epsilon$, then the iteration is stopped, where ϵ can be chosen by users.

The following lemma is well known[14,15]:

Lemma 3.1. A vector $s^* \in R^{n^2}$ is a solution of (12)-(13) if and only if there exists $\lambda^* \geqslant 0$ such that

$$(\mathrm{Hess}(\mathcal{M}[f]) + \lambda^* I)s^* = -\mathrm{grad}(\mathcal{M}[f]) \tag{17}$$

and that $\mathrm{Hess}(\mathcal{M}[f]) + \lambda^* I$ is positive semi-definite, $\|s^*\| \leqslant \quad$ and

$$\lambda^*(\quad - \|s^*\|) = 0. \tag{18}$$

It is shown by Powell[11] that trust region algorithms for (11) is convergent if the trust region step satisfies

$$\mathrm{Pred}(s) \geqslant c\|\mathrm{grad}(\mathcal{M}[f])\| \min\{\quad, \|\mathrm{grad}(\mathcal{M}[f])\|/\|\mathrm{Hess}(\mathcal{M}[f])\} \tag{19}$$

and some other conditions on $\mathrm{Hess}(\mathcal{M}[f])$ are satisfied. It is easy to see that

$$\begin{aligned} &\phi(0) - \min_{\|s\| \leqslant \Delta, s \in \mathrm{span}\{\mathrm{grad}(\mathcal{M}[f])\}} \phi(s) \\ &\quad \geqslant \frac{1}{2}\|\mathrm{grad}(\mathcal{M}[f])\| \min\{\quad, \mathrm{grad}(\mathcal{M}[f])/\|\mathrm{Hess}(\mathcal{M}[f])\}. \end{aligned} \tag{20}$$

Therefore it is quite common that in practice the trial step at each iteration of a trust region method is computed by solving the TRS (12)-(13) inexactly. One way to compute an inexact solution of (12)-(13) was the truncated conjugate gradient method proposed by Toint[16] and Steihaug[17] and analyzed by Yuan[18].

The conjugate gradient method for (12) yields a sequence as follows:

$$s_{k+1} = s_k + \alpha_k d_k, \tag{21}$$

$$d_{k+1} = -g_k + \beta_k d_k, \tag{22}$$

where $g_k = \nabla\phi(s_k) = \mathrm{Hess}(\mathcal{M}[f])s_k + \mathrm{grad}(\mathcal{M}[f])$ and

$$\alpha_k = -g_k^{\mathrm{T}} d_k / d_k^{\mathrm{T}} \mathrm{Hess}(\mathcal{M}[f]) d_k, \quad \beta_k = \|g_{k+1}\|^2 / \|g_k\|^2, \tag{23}$$

with the initial values

$$s_1 = 0, \quad d_1 = -g_1 = -\mathrm{grad}(\mathcal{M}[f]). \tag{24}$$

Toint[16] and Steihaug[17] were the first to use the conjugate gradient method to solve the general trust region subproblem (12)-(13). Even without assuming the positive definite of $\mathrm{Hess}(\mathcal{M}[f])$,

we can adopt the conjugate gradient method provided that $d_k^{\mathrm{T}}\mathrm{Hess}(\mathcal{M}[f])d_k$ is positive. If the iterate $s_k+\alpha_k d_k$ computed is in the trust region ball, it can be accepted, and the conjugate gradient iterates can be continued to the next iteration. Whenever $d_k^{\mathrm{T}}\mathrm{Hess}(\mathcal{M}[f])d_k$ is not positive or $s_k+\alpha_k d_k$ is outside the trust region, we can take the longest step along d_k within the trust region and terminate the calculation.

Algorithm 3.2 (truncated conjugate gradient method for TRS).

Step 1. Given $s_1=0$, ϵ(tolerance) >0 and compute $g_1=g(s_1)$, and set $k=1$, $d_1=-g_1=-\mathrm{grad}(\mathcal{M}[f])$.

Step 2. If $\|g_k\|\leqslant\epsilon$, stop, output $s^*=s_k$;
compute $d_k^{\mathrm{T}}\mathrm{Hess}(\mathcal{M}[f])d_k$: if $d_k^{\mathrm{T}}\mathrm{Hess}(\mathcal{M}[f])d_k\leqslant 0$ then goto step 4;
calculate α_k by (23).

Step 3. If $\|s_k+\alpha_k d_k\|\geqslant$ then goto step 4;
set s_{k+1} by (21) and $g_{k+1}=g_k+\alpha_k\mathrm{Hess}(\mathcal{M}[f])d_k$;
compute β_k by (23) and set d_{k+1} by (22);
$k:=k+1$, goto step 2.

Step 4. Compute $\alpha_k^*\geqslant 0$ satisfying $\|s_k+\alpha_k^* d_k\|=$;
set $s^*=s_k+\alpha_k^* d_k$, and stop.

Note that α_k^* can be computed by choosing the positive root of the quadratic equation in α:

$$\|d_k\|^2\alpha^2+2(s_k,d_k)\alpha+\|s_k\|^2- \quad ^2=0. \tag{25}$$

Let s^* be the inexact solution of (12)-(13) obtained by the above truncated CG method and let $\hat{s}$ be the exact solution of (12)-(13). If $n=2$, Yuan[18] shows that

$$\frac{\phi(0)-\phi(s^*)}{\phi(0)-\phi(\hat{s})}\geqslant\frac{1}{2}. \tag{26}$$

Recently Yuan[18] also proved that (26) is true for all n, which can be written as the following theorem.

Theorem 3.1. For any >0, $g\in R^n$ and any positive definite matrix $\mathrm{Hess}(\mathcal{M}[f])\in R^{n\times n}$, let $\hat{s}$ be the global solution of the trust region subproblem (12)-(13), and let s^* be the solution obtained by the truncated CG method. Then

$$\phi(s^*)\leqslant\frac{1}{2}\phi(\hat{s}). \tag{27}$$

This theorem tells us that the reduction in the approximate model is at least half at each iteration if we use truncated conjugate gradient method for solving subproblem (12)-(13). Noticing this fact, we know from algorithms 3.1 and 3.2 that there will not be many iterations to generate convergence.

4 Numerical simulation

In this section, we give a numerical test of a two-dimensional image restoration problem. We perform our numerical test on SGI workstation.

From algorithms 3.1 and 3.2, we can see clearly that there are two stage iterations: the inner loop and the outer loop. The inner loop is purely truncated conjugate gradient method, the outer loop is the trust region method. Since we only want to get an approximation solution of the TRS,

it needs not be solved so exactly. Noticing this fact, we add to our algorithm 3.2 another stopping rule, i. e. the maximal iterations: itermax. If the inner loop numbers are greater than itermax, then we stop the inner iteration and turn to the outer loop. In fact, this stopping rule is activated in the first two iterations of our numerical example.

It should be also pointed out that in algorithms 3.1 and 3.2, the Toeplitz structure of the model problem is not destroyed. Therefore we can still use this special structure to code, e.g. the 2-D FFT and 2-D IFFT can be used. In this way, the cost of computation is greatly reduced.

We consider a 128×128 image with irregular boundaries. The reconstructions are obtained from simulated satellite data generated from a model (1). This model has been used by several authors[3,4,1,19,20]. The data, which includes the point spread function (PSF), true object and observed image, is shown in figs. 1—3.

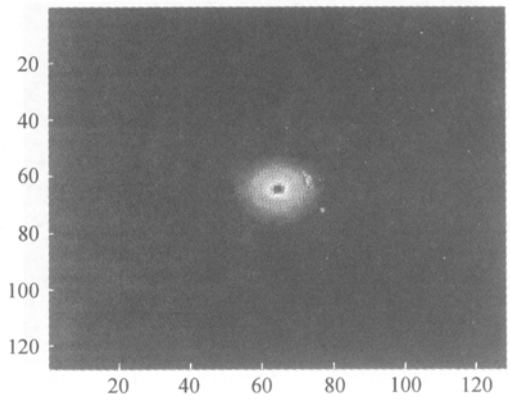

Fig. 1. Point spread function.

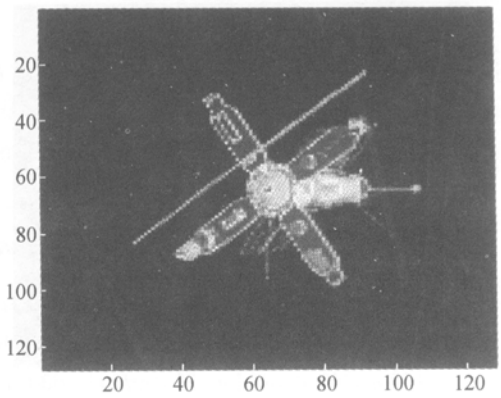

Fig. 2. The true image.

We first implement the trust region-CG algorithm, i. e. algorithm 3.1 and algorithm 3.2. The numerical results are shown in figs. 4 and 5.

In fig. 4, we choose the initial guess values as follows: $f_1 = 0$, $\quad_1 = 1.0$, $\tau_0 = 0$, $\tau_1 = 2$, $\tau_2 = 0.1$, $\tau_3 = 0.25$, $\tau_4 = 0.5$, itermax $= 50$, $\epsilon = 1.0e{-}8$. From this setting, the truncation is never activated, because the first few TRS solutions are within the trust region bound. In this case, only two iterations are needed to generate convergence. However in fig. 5, the truncation is activated. Since we choose the trust region radius as $\quad_1 = 0.01$, the first few TRS solutions are outside the trust region bound, hence the truncation is activated. Other notations are the same as in fig. 4. It requires three iterations to generate convergence.

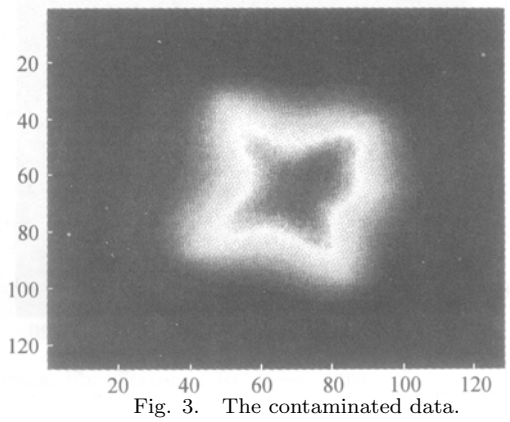

Fig. 3. The contaminated data.

We also perform Tikhonov regularization to this problem. The numerical algorithm used here is also based on conjugate gradient method. The results are shown in figs. 6—10.

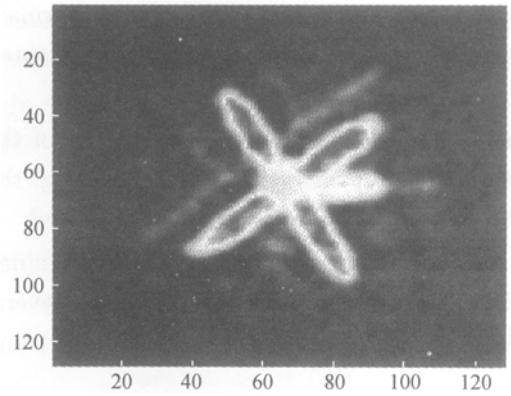

Fig. 4. The approximate image.

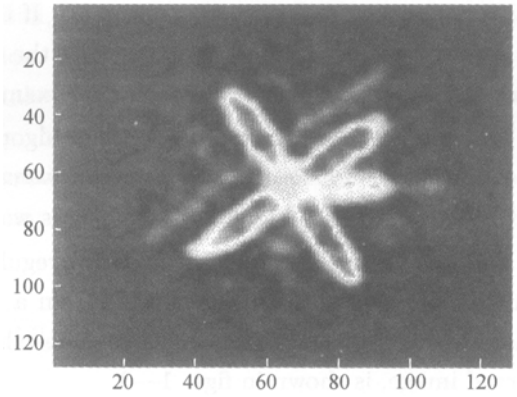

Fig. 5. The approximate image.

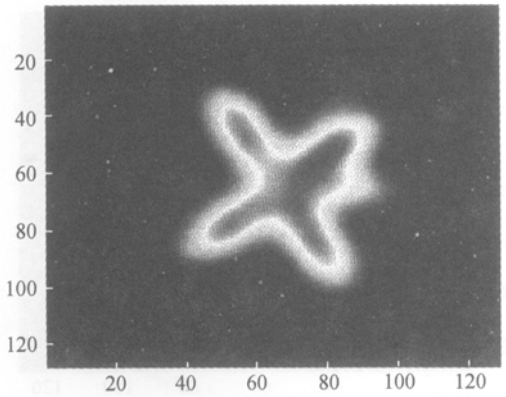

Fig. 6. The approximate image.

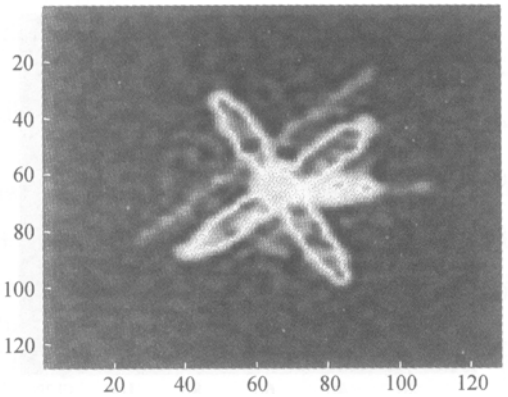

Fig. 7. The approximate image.

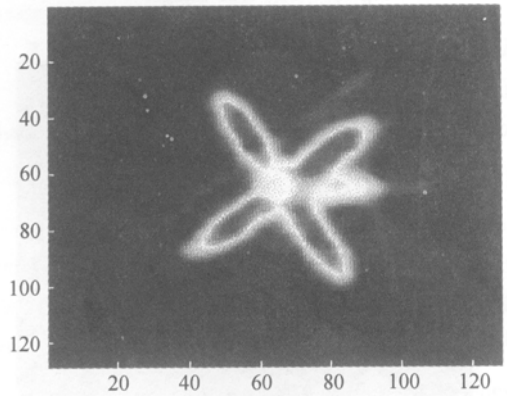

Fig. 8. The approximate image.

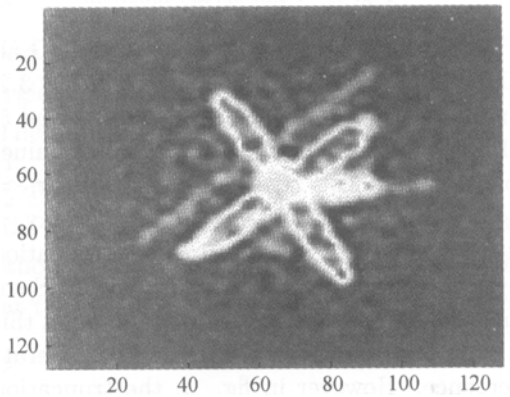

Fig. 9. The approximate image.

For Tikhonov regularization, we first choose the regularization parameter as $\alpha = 0.1$. The restoration image is shown in fig. 6 after 100 iterations. Then we choose $\alpha = 0.01$. The restoration image is shown in fig. 7 after 80 iterations. Fig. 8 gives a plot of the restoration image as we choose $\alpha = 0.001$ after 30 iterations. For small α, say, $\alpha = 1.0 \times 10^{-10}$, the restoration results are shown in figs. 9 and 10. In fig. 9, we take 50 iterations; in fig. 10, we take 100 iterations. This demonstrates

that the choice of the regularization parameter α is a subtle thing. We have to use our experience or the a information about the model to choose the regularization parameter. Besides, we observe that the iteration k also serves as a regularization parameter. Figs. 7 and 9 illustrate this fact. Therefore, how to choose a reasonable stopping rule for Tikhonov regularization is another important matter.

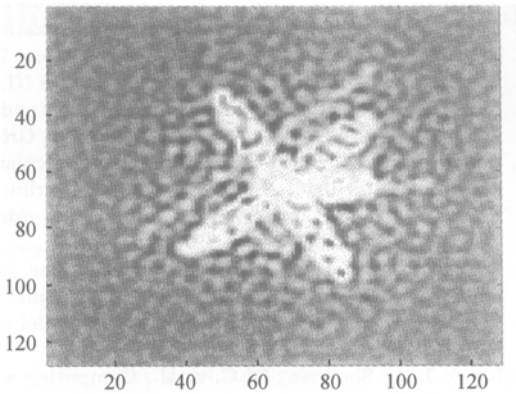

Fig. 10. The approximate image.

Remark. For Tikhonov regularization, the choice of regularization parameter α is crucial: If α is too large, say 0.1, the reconstruction result is bad (fig. 6) even with many iterations. If α is too small, say 1.0×10^{-10}, and if we give many iterations, the result is also bad (fig. 10). The best result is fig. 7 with $\alpha = 0.01$ and 80 iterations.

For trust region-CG algorithms, the results are a little better. According to Theorem 3.1, the reduction in the approximate model is at least half at each iteration. Hence, using Pred $< \epsilon$ as the stopping rule is reasonable. We can give an appropriate termination to obtain satisfactory results.

5 Conclusion and future research work

The numerical experiment shows that the trust region method is stable for ill-posed problems, at least for atmospheric image restoration problem concerned in this paper. We do not claim that the trust region-CG is better than Tikhonov regularization, which has been developed for about 40 years starting with the basic works by Tikhonov. But at least it can give comparative results. Numerical results also show that our trust region algorithm performs more easily than the Tikhonov regularization method.

It should be noted that we did not apply preconditioned technique to the present version of this paper. With appropriate preconditioner, the cost of computation can be significantly reduced. However, the choice of the preconditioner is another important problem for ill-posed image restoration problems. This is our future work.

Acknowledgements This work was partially supported by the National Natural Science Foundation of China (Grant No. 19731010) and the Knowledge Innovation Program of CAS.

References

1. Roggemann, M., Welsh, B., Imaging Through Turbulence, Boca Raton: CRC Press, 1996.
2. Vogel, C. R., Oman, M. E. et al., Fast, robust total variation-based reconstruction of noisy, blurred images, IEEE Transactions on Image Processing, 1998, 7: 813—824.
3. Michael, K. N., Plemmons, R. J., Qiao San-zheng et al., Regularized blind deconvolution using recursive inverse filtering, in Proc. HK97 Conference on Scientific Computation (eds. Golub, G., Liu, S., Luk, F. et al.), Berlin: Springer-Verlag, 1997.
4. Hanke, M., Iterative regularization techniques in image reconstruction, in Proceedings of the Conference Mathematical Methods in Inverse Problems for Partial Differential Equations, Mt. Holyoke: Springer-Verlag,

1998.
5. Tikhonov, A. N., Arsenin, V. Y. et al., Solutions of Ill-posed Problems, New York: Wiley, 1977.
6. Engl, H. W., Neubauer, H. M. et al., Regularization of Inverse Problems, Dordrecht: Kluwer, 1996.
7. Fletcher, R., Practical Methods of Optimization, 2nd ed. Chichester: John Wiley and Sons, 1987.
8. Fletcher, R., A model algorithm for composite NDO problem, Math. Prog. Study, 1982, 17: 67—76.
9. Moré, J. J., Recent developments in algorithms and software for trust region methods, Mathematical Programming: The State of the Art (eds. Bachem, A., Grötschel, K. B.), Berlin: Springer-Verlag, 1983: 258—287.
10. Powell, M. J. D., Nonconvex minimization calculations and the conjugate gradient method, in Lecture Notes in Mathematics 1066: Numerical Analysis, Berlin: Springer-Verlag, 1984, 122—141.
11. Powell, M. J. D., Convergence properties of a class of minimization algorithms, in Nonlinear Programming, New York: Academic Press, 1975, 2: 1—27.
12. Duff, I. S., Nocedal, J., Reid, J. K. et al., The use of linear programming for the solution of sparse sets of nonlinear equations, SIAM J. Sci. Stat. Comput., 1987, 8: 99—108.
13. Sorensen, D. C., Newton's method with a model trust region modification, SIAM J. Numer. Anal., 1982, 19: 409—426.
14. Moré, J. J., Sorensen, D. C. et al., Computing a trust region step, SIAM J. Sci. Stat. Comput., 1983, 4: 553—572.
15. Gay, D. M., Computing optimal local constrained step, SIAM J. Sci. Stat. Comp., 1981, 2: 186—197.
16. Toint, P. L., Towards an efficient sparsity exploiting Newton method for minimization, in Sparse Matrices and Their Uses (ed. Duff, I.), Berlin: Academic Press, 1981, 57—88.
17. Steihaug, T., The conjugate gradient method and trust regions in large scale optimization, SIAM J. Numer. Anal., 1983, 20: 626—637.
18. Yuan, Y., On the truncated conjugate gradient method, Math. Prog., 2000, 87: 561—571.
19. Vogel, C. R., A limited memory BFGS method for an inverse problem in atmospheric imaging, in Lecture Notes in Earth Sciences: Methods and Applications of Inversion (eds. Hansen, P. C., Jacobsen, B. H., Mosegaard, K.), Berlin: Springer-Verlag, 2000, 92: 292—304.
20. Vogel, C. R., Computational Methods for Inverse Problems, Frontiers in Applied Mathematics, Philadelphia: SIAM, 2002, in press.