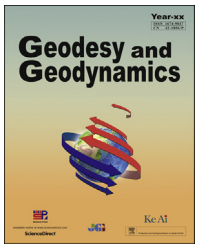
GEODESY AND GEODYNAMICS 2015, VOL 6 NO 6, 453–459

Available online at www.sciencedirect.com
ScienceDirect
journal homepage: www.keaipublishing.com/en/journals/geog;
http://www.jgg09.com/jweb_ddcl_en/EN/volumn/home.shtml



An iterative algorithm for solving ill-conditioned linear least squares problems

Deng Xingsheng[*], *Yin Liangbo*, *Peng Sichun*, *Ding Meiqing*

Engineering Laboratory of Spatial Information Technology of Highway Geological Disaster Early Warning in Hunan Province, Changsha University of Science & Technology, Changsha 410004, China

ARTICLE INFO

Article history:
Received 18 May 2015
Accepted 23 June 2015
Available online 22 December 2015

Keywords:
Severe ill-conditioned matrix
Linear least squares problems
Self-adaptive
Iterative scheme
Cholesky decomposition
Regularization parameter
Tikhonov solution
Truncated SVD solution

ABSTRACT

Linear Least Squares (LLS) problems are particularly difficult to solve because they are frequently ill-conditioned, and involve large quantities of data. Ill-conditioned LLS problems are commonly seen in mathematics and geosciences, where regularization algorithms are employed to seek optimal solutions. For many problems, even with the use of regularization algorithms it may be impossible to obtain an accurate solution. Riley and Golub suggested an iterative scheme for solving LLS problems. For the early iteration algorithm, it is difficult to improve the well-conditioned perturbed matrix and accelerate the convergence at the same time. Aiming at this problem, self-adaptive iteration algorithm (SAIA) is proposed in this paper for solving severe ill-conditioned LLS problems. The algorithm is different from other popular algorithms proposed in recent references. It avoids matrix inverse by using Cholesky decomposition, and tunes the perturbation parameter according to the rate of residual error decline in the iterative process. Example shows that the algorithm can greatly reduce iteration times, accelerate the convergence, and also greatly enhance the computation accuracy.

1. Introduction

A linear or linearized model is expressed as

$$\boldsymbol{L} = \boldsymbol{AX} - \boldsymbol{V}, \quad \mathrm{cov}(\boldsymbol{V}) = \sigma_0^2\boldsymbol{Q}, \quad \boldsymbol{P} = \boldsymbol{Q}^{-1} \tag{1}$$

where $\boldsymbol{L}\in\boldsymbol{R}^n$ is an observation vector contaminated by an error vector $\boldsymbol{V}\in\boldsymbol{R}^n$ with normal distribution of mean zero and covariance matrix $\sigma_0^2\boldsymbol{Q}$; $\boldsymbol{P}$ is a positive-definite weight matrix; $\boldsymbol{A}\in\boldsymbol{R}^{n\times m}$ is a matrix with full column rank connected to an unknown vector $\boldsymbol{X}\in\boldsymbol{R}^m$ and generally $n > m$. We are concerned with the solution of least-squares problems:

$$\min_{\boldsymbol{X}\in\boldsymbol{R}^m} \|\boldsymbol{AX} - \boldsymbol{L}\| \tag{2}$$

where $\|\cdot\|$ denotes the Euclidean vector norm, $\boldsymbol{X}$ is the unknown vector to be solved. If the matrix $\boldsymbol{A}$ is well-conditioned,

* *Corresponding author*. Tel.: +86 731 85258575; fax: +86 731 85258375.
E-mail address: whudxs@163.com (Deng X.).
Peer review under responsibility of Institute of Seismology, China Earthquake Administration.

Production and Hosting by Elsevier on behalf of KeAi

http://dx.doi.org/10.1016/j.geog.2015.06.004

the least-squares solution has the best unbiased estimation to this over determined system of equation (1) which is given as

$$\begin{cases} (\mathbf{A}^{\mathrm{T}}\mathbf{PA})\mathbf{X} = \mathbf{A}^{\mathrm{T}}\mathbf{PL} \\ \mathbf{X} = (\mathbf{A}^{\mathrm{T}}\mathbf{PA})^{-1}(\mathbf{A}^{\mathrm{T}}\mathbf{PL}) \end{cases} \tag{3}$$

However, $\mathbf{A}^{\mathrm{T}}\mathbf{PA}$ may be a severely ill-conditioned matrix, thus it cannot be inverted. Problems of this kind are referred to ill-posed problems. Due to the ill-conditioning of $\mathbf{A}^{\mathrm{T}}\mathbf{PA}$, these problems are difficult to solve accurately [1]. Inverting ill-conditioned large matrices is a challenging problem involved in a wide range of applications, including inverse problems and partial differential equations [2]. Global Navigation Satellite System (GNSS) is a fast, dynamic, high precision positioning technique that has been attracting more and more attention in modern geodesy. In the static positioning of GNSS, the carrier phase ambiguity and other parameters are set as unknown for solving. A linear observational equation system for real-time GNSS carrier phase ambiguity resolution is often severely ill-posed, in the case of poor satellite geometry [3]. Generally, in order to improve the precision and reliability of the solution, a long time for GNSS observing is usually needed. GNSS satellites belong to high orbit satellite, and the angle velocity is small. If the observation period is not long enough, the directions of the receivers to the satellites will see little change, and the distances between stations and satellites vary little in the whole observing session. Thus observation equations of the same satellites and different epochs are almost similar, so to rapidly determine phase ambiguity is a typical ill-condition problem.

Linear discrete ill-posed problems are difficult to solve numerically, because their solution is very sensitive to perturbations which may stem from errors in the data, round-off errors and discretization errors during which introduced the solution process [4,5]. Severely ill-conditioned matrix inverting problems abound in the geosciences, especially in the data processing of modern survey. In the numerical algorithm, all the cases of inappropriate function model or inappropriate calculating method, a morbid or singular iteration matrix and so on, will lead to inaccurate solutions. For singular matrix and ill-posed problems, there are a large number of research results, such as regularization methods. Among all regularization methods, perhaps the best known and most commonly used is the Tikhonov–Phillips method, which was originally proposed by Tikhonov and Phillips in 1962 and 1963 [6]. It's possible that the best understood regularization method is due to Tikhonov [7]. The Tikhonov regularization method is one of the most popular approaches to determine an approximation of X. This method replaces the linear system of equation (2) by a penalized least-squares problem of the form [8–12]:

$$\min_{\mathbf{X}\in\mathbf{R}^m}\left\{\|\mathbf{AX}-\mathbf{L}\|^2+\mu\|\mathbf{TX}\|^2\right\} \tag{4}$$

where $\mu > 0$ is known as the regularization parameter, $\mathbf{T}$ is some suitably chosen Tikhonov matrix. Ill-posed problems must be first regularized if one wants to successfully attack the task of numerically approximating their solutions. It is often said that the art of applying regularization methods consist always in maintaining an adequate balance between accuracy and stability [13]. As to regularization methods, there are three drawbacks: (1) these methods destroy the equivalence relation of the equation (3); (2) a regularized solution is well-known to be biased [14]; and (3) to determine the optimal regularization parameter is rather difficult.

Riley [15] and Golub [16] suggested an iterative scheme for solving LLS problems, which has advantages as follows: (1) it makes the perturbed matrix well-conditioned, and improves the condition number of matrix in the normal equation; (2) it keeps the equivalence relation of the equation unchanged; and (3) the iteration can always converge to the optimal solution theoretically. For these reasons, it has attracted attention from geodesists in data processing widely. However, a few problems are found in its practical application in recent years [17]. The choice of perturbation parameter will greatly affect the rate of convergence of the iterative method, and thus one must choose it with great care [16]. The perturbation parameter chosen should be large enough to make the perturbed matrix well-conditioned, yet small enough to ensure that the error $\|\mathbf{X}-\widetilde{\mathbf{X}}\|$ is small [18]. If the perturbation parameter increases, the convergence rates turn out to be low; but if decreased, the ill-posed matrix cannot be improved to be well-conditioned. For this reason, based on theoretical analysis and a large number of experiments, a new self-adaptive iteration algorithm is proposed in this paper.

The contributions of this paper are as follows: (1) a formula to determine the initial perturbation parameter is given; (2) a self-adaptive strategy is proposed to determine the tunable perturbation parameter dynamically; (3) an optimal termination point is found to stop the iteration. Comparison results of some experiments indicate that the algorithm can accelerate the convergence and improve computation accuracy. The rest of the paper is organized as follows. Section 2 introduces the algorithm in detail for severe ill-posed problems. Section 3 gives several experiments to demonstrate the superior performance of the proposed algorithm. The concluding remarks are outlined in Section 4.

2. Self-adaptive iteration algorithms

2.1. *Implementations of regularization*

The ill-posed matrix is generally measured by the condition number of the matrix. If the condition number of $\mathbf{A}^{\mathrm{T}}\mathbf{PA}$ is very large, that means the matrix is usually ill-posed. In this case, finding the inverse matrix of $\mathbf{A}^{\mathrm{T}}\mathbf{PA}$ in equation (3) may have no stable solution. To solve the problem, many references [8,18–21] employ an algorithm like this

$$\mathbf{X}_{\mu} = (\mathbf{A}^{\mathrm{T}}\mathbf{PA}+\mu\mathbf{I})^{-1}(\mathbf{A}^{\mathrm{T}}\mathbf{PL}) \tag{5}$$

where μ is an arbitrary regularization parameter, $\mathbf{I}$ denotes identity matrix. It is obvious that adding $\mu\mathbf{I}$ to the right side of equation (5) will destroy the equivalence relation in equation (3). The solution $\mathbf{X}_{\mu}$ solved by equation (5) is no longer the same $\mathbf{X}$ in equation (3). Another drawback is that the condition number of $\mathbf{A}^{\mathrm{T}}\mathbf{PA}$ is much more than that of $\mathbf{A}$, which requires μ to be large enough to control the condition of the matrix [18]. Moreover, it is difficulty to determine an

optimal value of μ. Study results show that if the regularization parameter does not grow too fast (not faster than a geometric sequence), then the scheme converges with optimal convergence rates [22].

In order to stabilize the solution in such an ill-posed model, an iteration algorithm is suggested as following:

$$\begin{cases} (\mathbf{A}^{\mathrm{T}}\mathbf{PA}+I)\mathbf{X}=\mathbf{A}^{\mathrm{T}}\mathbf{PL}+\mathbf{X} \\ \mathbf{X}^{(k+1)}=(\mathbf{A}^{\mathrm{T}}\mathbf{PA}+I)^{-1}\left[\mathbf{A}^{\mathrm{T}}\mathbf{PL}+\mathbf{X}^{(k)}\right] \end{cases} \tag{6}$$

where k denotes kth iteration times. In most cases, equation (6) can solve many general-conditions ill-posed problems [17]. But in some extremely ill-posed problems, the iteration is difficult to converge. And even if iteration converges, the convergence rates will also be rather low, millions of iteration times are always required to get higher accuracy.

2.2. Perturbation parameter

In order to accelerate the convergence, a perturbation parameter is added to equation (6):

$$\mathbf{X}^{(k+1)}=(\mathbf{A}^{\mathrm{T}}\mathbf{PA}+a\mathbf{I})^{-1}\left[\mathbf{A}^{\mathrm{T}}\mathbf{PL}+a\mathbf{X}^{(k)}\right] \tag{7}$$

where a is the perturbation parameter. In equation (7), $0 \le a \le 1$. When $a = 0$, equations (7) and (3) are equivalent; when $a = 1$, equations (7) and (6) are equivalent. So equation (7) contains both scenarios of (3) and (6). Experiments have shown that decreasing the value of a may increase the iteration speed. But if a is too small, then the equations will remain ill-conditioned. Therefore, how to balance the convergence rates and matrix condition is a dilemmatic task. For this reason, to determine perturbation parameter a is rather difficult. An optimal selection of a must improve both the matrix condition and accelerate the convergence of equation (7).

2.3. Cholesky decomposition

To invert a severely ill-conditioned matrix, most algorithms will fail to get an optimal solution. Moreover, the arbitrary selection of a is not appropriate in most cases. Aiming at the problems, we use a tunable self-adaptive perturbation parameter a and avoid inverting an ill-conditioned matrix by Cholesky decomposition. The proposed algorithm based on equation (8), gives $\mathbf{X}$ and a an initial value, then adjusts the value of a in next iteration.

$$\begin{cases} (\mathbf{A}^{\mathrm{T}}\mathbf{PA}+a\mathbf{I})\mathbf{X}^{(k+1)}=\mathbf{A}^{\mathrm{T}}\mathbf{PL}+a\mathbf{X}^{(k)} \\ \mathbf{X}^{(0)}=\underset{n\times 1}{0} \end{cases} \tag{8}$$

In equation (8), the value of a is determined by a self-adaptive way. If a is reduced to a small value, the condition number of the coefficient matrix will increase. In order to avoid inverting $\mathbf{A}^{\mathrm{T}}\mathbf{PA} + a\mathbf{I}$ matrix, Cholesky decomposition is adopted to solve equation (8). In the early version of our algorithm [23], we use LU decomposition in the process. It is reported in many references that Cholesky decomposition would be better than LU decomposition. Then we revised our algorithm to use Cholesky decomposition, and don't do any more comparison between these two methods. The merits of Cholesky decomposition can be found in many references.

In linear algebra, Cholesky triangle is a decomposition of a Hermitian, positive-definite matrix into the product of a lower triangular matrix and its conjugate transpose. It was discovered by André-Louis Cholesky for real matrices. When it is applicable, the Cholesky decomposition is roughly twice as efficient as the LU decomposition for solving systems of linear equations [24]. Cholesky decomposition is much simpler than the eigenvalues and SVD methods. It needs fewer arithmetic operations and less computational time. Cholesky decomposition is a direct decomposition method without inversion. At the same time, a large matrix can be computed in parallel, thus it has advantages in solving a large normal equation. Especially targeted to solve an ill-posed problem, Cholesky decomposition has the significant advantage of its simplicity and does not need to invert the matrix comparing with other inverse methods.

2.4. Initial value of the perturbation parameter

If δ is a lower bound of the smallest non-zero singular value, Golub suggested that a should be chosen as

$$\frac{a}{a+\delta^2}<0.1 \tag{9}$$

This means at each stage, there will be at least one more place of accuracy in the solution [16]. At the beginning of iteration, we give the initial value of a according to equation (10)

$$\begin{cases} \lambda=\min(|eig(\mathbf{A}^{\mathrm{T}}\mathbf{PA})|) \\ a=\lambda 10^{0.5|\log(\lambda)|+1} \end{cases} \tag{10}$$

In equation (10), *eig* is the multiple eigenvalue of the matrix $\mathbf{A}^{\mathrm{T}}\mathbf{PA}$; min means the minimal value; log is logarithm at the base 10; λ is the smallest eigenvalue of the matrix $\mathbf{A}^{\mathrm{T}}\mathbf{PA}$; $|\cdot|$ is the absolute value, it is necessary for the *eig* may be a negative number. The initial value of a is determined by λ_{min}, when $\lambda_{min} = 10^{-10}$, $a = 10^{-4}$; when $\lambda_{min} = 0.01$, $a = 1$. Equation (9) is suitable for the matrix with $\lambda_{min} \ll 0.01$.

The purpose we set the initial value of a as equation (10) is to balance the matrix condition and the convergence. The condition number of M is $cond(M) = \lambda_{max}/\lambda_{min}$. As to a severely ill-conditioned matrix M, $\lambda_{min} \approx 0$, so $cond(M)$ is usually a very big number. Assuming λ_i is the multiple eigenvalue of the matrix $\mathbf{A}^{\mathrm{T}}\mathbf{PA}$, then the multiple eigenvalue of the matrix $\mathbf{A}^{\mathrm{T}}\mathbf{PA} + a\mathbf{I}$ is $\lambda_i + a$. The role of the perturbation parameter a is to change the condition number from $\lambda_{max}/\lambda_{min}$ to $\lambda_{max} + a/\lambda_{min} + a$. A proper a can decrease the condition number and improve the matrix condition.

2.5. Iteration algorithm

Now we give the steps of the proposed algorithm, which are as follows:

(1) Setting $\mathbf{NX} = \mathbf{W}$. Assuming $\mathbf{N}$, $\mathbf{W}$ are:

$$\mathbf{N}=\mathbf{A}^{\mathrm{T}}\mathbf{PA}+a\mathbf{I} \tag{11}$$

$$\mathbf{W}=\mathbf{A}^{\mathrm{T}}\mathbf{PL}+a\mathbf{X}^{(k)} \tag{12}$$

N can be decomposed into the product of a lower triangular matrix $\mathbf{C}$ and its conjugate transpose $\mathbf{C}^{\mathrm{T}}$, i.e., $\boldsymbol{N} = \boldsymbol{C}\boldsymbol{C}^{\mathrm{T}}$. This will yield an equation $(\boldsymbol{C}\boldsymbol{C}^{\mathrm{T}})\boldsymbol{X} = \boldsymbol{W}$.

(2) Assuming the initial value of $\boldsymbol{C}$ is:

$$\boldsymbol{C} = \underset{n\times n}{0} \tag{13}$$

where 0 is zero matrix; $n \times n$ is the size of $\mathbf{N}$.

(3) According to Cholesky decomposition, computing matrix element of $\boldsymbol{C}$ by following equations (14) and (15):

$$\begin{cases} \boldsymbol{C}(1,1) = \sqrt{\boldsymbol{N}(1,1)} \\ \boldsymbol{C}(i,1) = \boldsymbol{N}(i,1)/\boldsymbol{C}(1,1) \end{cases}, \quad i = 2 \sim n \tag{14}$$

$$\begin{cases} \boldsymbol{C}(j,j) = \sqrt{\boldsymbol{N}(j,j) - \sum_{k=1}^{j-1} \boldsymbol{C}(j,k)^2} \\ \boldsymbol{C}(i,j) = \left[\boldsymbol{N}(i,j) - \sum_{k=1}^{j-1} \boldsymbol{C}(i,k) \times \boldsymbol{C}(j,k)\right] \Big/ \boldsymbol{C}(j,j) \\ i = j+1 \sim n \end{cases}, \quad j = 2 \sim n, \tag{15}$$

(4) Letting $\boldsymbol{Y} = \boldsymbol{C}^{\mathrm{T}}\boldsymbol{X}$. Then solving $\boldsymbol{Y}$ by the equation $\boldsymbol{C}\boldsymbol{Y} = \boldsymbol{W}$:

$$\begin{cases} \boldsymbol{Y}(i) = \boldsymbol{W}(i)/\boldsymbol{C}(i,i), \quad i = 1 \sim n \\ \boldsymbol{Y}(i) = \boldsymbol{Y}(i) - \sum_{j=1}^{i-1} \boldsymbol{C}(i,j)/\boldsymbol{C}(i,i) \times \boldsymbol{Y}(j), \quad i = 2 \sim n \end{cases} \tag{16}$$

(5) Letting $\boldsymbol{U} = \boldsymbol{C}^{\mathrm{T}}$, taking the values of $\boldsymbol{Y}$ and solving $\boldsymbol{X}$ by the equation $\boldsymbol{U}\boldsymbol{X} = \boldsymbol{Y}$. This will give the solution $\boldsymbol{X}$ in the equation $\boldsymbol{N}\boldsymbol{X} = \boldsymbol{W}$.

$$\begin{cases} \boldsymbol{X}(i) = \boldsymbol{Y}(i)/\boldsymbol{U}(i,i), \quad i = n \sim 1 \\ \boldsymbol{X}(i) = \boldsymbol{X}(i) - \sum_{j=n}^{i+1} \boldsymbol{U}(i,j)/\boldsymbol{U}(i,i) \times \boldsymbol{X}(j), \quad i = n-1 \sim 1 \end{cases} \tag{17}$$

(6) Adjusting the value of a, doing steps (1)–(5) again, and stopping the iteration when the termination criterion $err^{(k)}/err^{(k-1)} > 1$ is satisfied, where $err^{(k)} = \|\boldsymbol{N}\boldsymbol{X}^{(k)} - \boldsymbol{W}\|$.

2.6. *Adjusting* a *self-adaptively*

In the iterative process, the value of a is not fixed. It is adjusted automatically by a "double or half" strategy according to the rate of residual reduction [25]. If the ratio of kth iteration residuals to $(k-1)$th is over 0.75, which means the iteration residuals cannot be effectively reduced, i.e., $err^{(k)}/err^{(k-1)} > 0.75$. If the iteration satisfies the condition, then reduced the value of a to its half, i.e., $a^{(k+1)} = a^{(k)}/2$.

But in some cases, we also find that the iteration residuals decrease quickly, i.e., $err^{(k)}/err^{(k-1)} < 0.25$. Then we double the value of a in the next iteration, i.e., $a^{(k+1)} = a^{(k)} \times 2$.

If an appropriate value of a can make the iteration residual error decrease steadily, then a remains unchanged and iteration continues.

2.7. *Terminating the iteration*

The iteration will be terminated as soon as the iteration meets the precision requirements, or satisfies a stopping rule related to the discrepancy principle. Let

$$\boldsymbol{e}^{(k)} = \boldsymbol{X}^{(k+1)} - \boldsymbol{X}^{(k)} \tag{18}$$

It is easy to see that

$$\boldsymbol{e}^{(k+1)} = a\left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{A} + a\boldsymbol{I}\right)^{-1}\boldsymbol{e}^{(k)} \tag{19}$$

It is obvious that $\left\|\boldsymbol{e}^{(k+1)}\right\|$ should be less than $\left\|\boldsymbol{e}^{(k)}\right\|$. Golub suggested that a good termination procedure is to stop iterating as soon as $\left\|\boldsymbol{e}^{(k)}\right\|$ increases or does not change [16].

The discrepancy principle prescribes that the iterations should be terminated as soon as an iteration $\boldsymbol{x}_k$ that satisfies the stopping criterion $\|\boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b}\| \le \eta\delta$ has been found, where $\eta > 1$ is a user-specified constant independent of δ [1,26]. In order to terminate the iteration at an idea point, we have traced the process of iteration residual errors at different iteration times in many experiments. In most cases, we can find that the processes of iteration residual errors include two parts: one is the monotonically decreasing process, the other is the monotonically increasing process, as shown in Fig. 1.

Obviously, the idea terminating point satisfies $err^{(k)}/err^{(k-1)} > 1$, which will be adopted as the iteration termination criterion. So the $(k-1)$th iteration results can be accepted as the final solutions.

3. Example and comparative analyse

In this section, a twenty order Hilbert ill-conditioned matrix is solved, to demonstrate the performance of the proposed algorithm SAIA, and comparing it with other solutions, which include least-squares solution, Tikhonov solution, and truncated SVD solution.

The typical Hilbert matrix is defined as

$$\boldsymbol{H}_n = \left(h_{ij}\right)_{n\times n}, \quad h_{ij} = \frac{1}{i+j-1} \tag{20}$$

Hilbert is a symmetric positive definite matrix. With the increase of its order, it becomes more seriously ill-conditioned. A twenty order Hilbert matrix $\boldsymbol{H}_{20}$, its determinant value is -9.9312×10^{-197}, and its condition number is 2.0383×10^{18}. Thus $\boldsymbol{H}_{20}$ is a severely ill-conditioned matrix. By increasing the order of Hilbert matrices respectively, more experiments are made for the proposed algorithm. SARA is still a successful method. It can obtain precise solutions after a few iterations, and solutions are reliable.

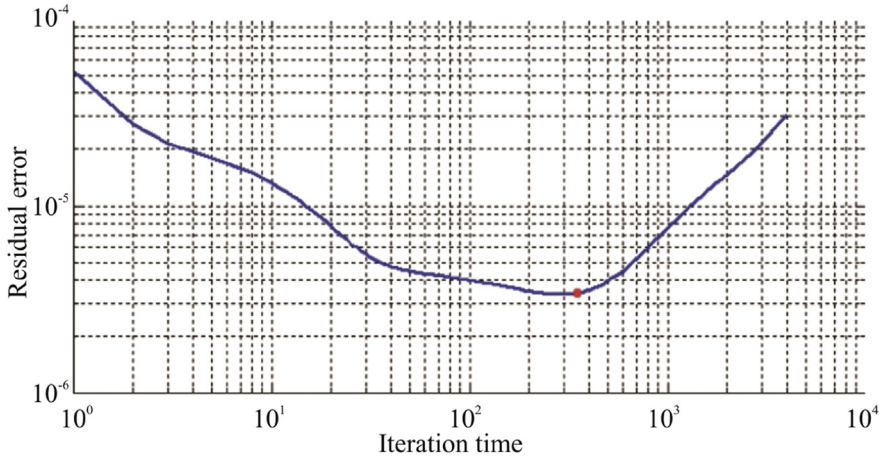


Fig. 1 – Process chart of iteration residual error. The red dot, where $err^{(k)} > err^{(k-1)}$, denotes the optimal iteration terminating point.

Let $\boldsymbol{X}_{20\times 1} = [11\ldots1]^{T}$, we can get $\boldsymbol{B}_{20\times1} = \boldsymbol{H}_{20}\cdot\boldsymbol{X}_{20\times1}$. $\boldsymbol{X}$ is a known 20×1 matrix, whose elements all are 1. Using $\boldsymbol{H}$ and $\boldsymbol{B}$ to solve $\boldsymbol{X}$, the solutions are shown in Table 1, and the iteration process is shown in Fig. 2.

Table 1 shows that the severely ill-conditioned Hilbert matrices cannot be solved directly, for the computing accuracy of the direct least-squares solution is rather low. Nearly all of the solutions differ greatly from their true values, and partial results are distorted severely. The residual error of Tikhonov solutions (μ=3 $\times$ 10^{-4}) is 2.616 $\times$ 10^{-6}. The residual error of the truncated SVD (truncation parameter $K = 4$) is 2.262 $\times$ 10^{-5}. Among four methods, solutions of the proposed SAIA are relatively better. It only iterates thirteen times, and the residual error is 2.488 $\times$ 10^{-6}, which is the minimum among four methods. If we fix a as 10^{-6}, after iterating 100,000 times, the residual error is 3.699 $\times$ 10^{-6}. These data suggest that iteration times of SAIA are greatly reduced; in addition, convergence rates and calculation accuracy are greatly improved.

4. Conclusions

(1) The ill-posed matrix is a basic problem in mathematics and geosciences. Although there have been many regularization methods, they destroy the equivalence relation of the normal equation, and the estimation results are known to be biased. Furthermore, the optimal regularization parameter is difficult to determine. Self-adaptive regularization iteration algorithm can get an unbiased estimation. It does not change the equivalence relation of the normal equation. However,

Table 1 – Solutions of Hilbert matrix.

	Matrix inverse $\boldsymbol{X} = \boldsymbol{H}^{-1}\boldsymbol{B}$	Tikhonov solution $\mu = 3 \times 10^{-4}$	Truncated SVD $K = 4$	Proposed SAIA
X1	−3.47749698162079	0.999906132035295	0.993484916005128	0.999999999822808
X2	15.6925048828125	0.998085504035077	1.04427686168369	1.0000000121112
X3	0.989501953125	1.00925970365237	0.973558996862218	0.999999802248412
X4	1.0625	0.999951891233500	0.960144502783415	1.00000128005709
X5	0.65625	0.992605432679234	0.971565746097472	0.999996282824641
X6	1.3125	0.990707371106231	0.989348404555657	1.00000407253213
X7	1.4375	0.992767213589953	1.00590053450885	1.00000154275461
X8	2	0.996795187253857	1.01856243381075	0.99999477735415
X9	10	1.00128915092934	1.02675428986424	1.00000044986221
X10	−18	1.00529583238094	1.03072750808467	1.00000127598629
X11	32	1.00827842719852	1.03102389280142	0.9999999974566
X12	−10	1.00997946007215	1.02824441853018	1.00000293440795
X13	12	1.01031715334737	1.02295497280410	0.999998635600596
X14	−56	1.00931470778990	1.01565302674189	1.000000968559
X15	−4	1.00705427788116	1.00676103038321	0.999997484394717
X16	24	1.00364787791722	0.996630926933501	0.999998792618754
X17	39.5	0.999219440634574	0.985552653225642	0.999997483156963
X18	−144	0.993894068787960	0.973763427637823	1.00000458486704
X19	32	0.987791862511157	0.961456471138547	1.00000242073192
X20	−10	0.981024622513384	0.948788669242438	0.999997202450518
Residual error	38.2587	2.616×10^{-6}	2.262×10^{-5}	2.488×10^{-6}

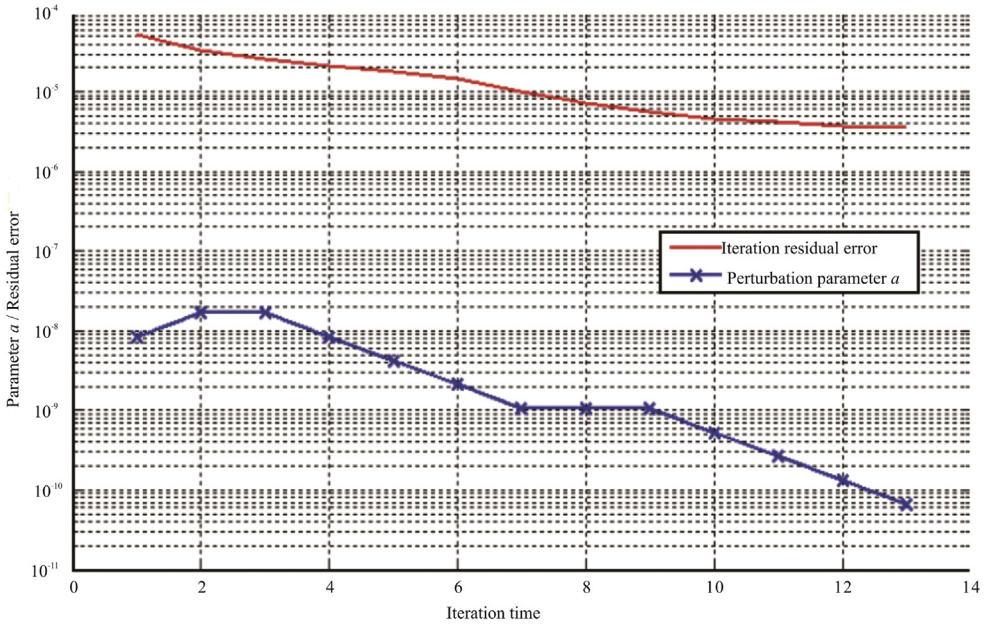


Fig. 2 – Process chart shows the iteration residual error and the value of perturbation parameter *a*. The iteration residual error is decreased steadily while perturbation parameter *a* decreased in thirteen iteration times.

its convergence rates are rather low, and the time for iteration is quite long, even failing to converge in a limited iteration times. It is rather difficult to improve the computation precision. The iteration can be speeded up by adding a perturbation parameter *a*, yet how to determine an optimal *a* is a hard task. It is contradictory to improve the well-conditioned matrix and speed up the convergence at the same time.

(2) The proposed self-adaptive regularization algorithm SAIA is a new algorithm to treat the ill-posed problem. It is different from other popular algorithms in some recent references. The algorithm adopts Cholesky decomposition to avoid matrix inverting. A formula is given to determine the initial value of the perturbation parameter. In the iteration, the perturbation parameter is adjusted self-adaptively according to the residual error descent rate. It balances the iteration convergence rates and well-conditioned matrix simultaneously. The performance of the proposed algorithm is demonstrated in the Hilbert example. It can greatly reduce the iteration times, and also enhance the convergence rates and computation accuracy greatly.

Acknowledgements

This research was supported by Open Fund of Engineering Laboratory of Spatial Information Technology of Highway Geological Disaster Early Warning in Hunan Province (Changsha University of Science & Technology, kfj150602); Hunan Province Science and Technology Program Funded Projects, China (2015NK3035); the Land and Resources Department Scientific Research Project of Hunan Province, China (2013-27); and the Education Department Scientific Research Project of Hunan Province, China (13C1011). Many friends and colleagues whose previous work contributed indirectly to this paper are greatly appreciated.

REFERENCES

[1] Neuman A, Reichel L, Sadok H. Implementations of range restricted iterative methods for linear discrete ill-posed problems. Linear Algebra Appl 2012;436:3974–90.

[2] Maréchal Pierre, Rondepierre Aude. A proximal approach to the inversion of ill-conditioned matrices. C R Acad Sci Paris Ser I 2009;347:1435–8.

[3] Bofeng Li, Yunzhong Shen, Yanming Feng. Fast GNSS ambiguity resolution as an ill-posed problem. J Geod 2010;84:683–98.

[4] Reichel Lothar, Rodriguez Giuseppe. Old and new parameter choice rules for discrete ill-posed problems. Numer Algor 2013;63:65–87.

[5] Morigi S, Plemmons R, Reichel L, Sgallari F. A hybrid multilevel-active set method for large box-constrained linear discrete ill-posed problems. Calcolo 2011;48:89–105.

[6] Mazzieri Gisela L, Spies Ruben D. Regularization methods for ill-posed problems in multiple Hilbert scales. Inverse Probl 2012;28(5):1–31.

[7] Klann E, Ramlau R, Reichel L. Wavelet-based multilevel methods for linear ill-posed problems. BIT Numer Math 2011;51:669–94.

[8] Donatelli M, Neuman A, Reichel L. Square regularization matrices for large linear discrete ill-posed problems. Numer Linear Algebra Appl 2012;19:896–913.

[9] Hochstenbach Michiel E, Reichel Lothar. Fractional Tikhonov regularization for linear discrete ill-posed problems. BIT Numer Math 2011;51:197–215.

[10] Eggermont PPB, LaRiccia VN, Nashed MZ. Moment discretization for ill-posed problems with discrete weakly bounded noise. Int J Geomath 2012;3:155–78.

[11] Neuman A, Reichel L, Sadok H. Algorithms for range restricted iterative methods for linear discrete ill-posed problems. Numer Algor 2012;59:325–31.
[12] Zhang Jianjun, Mammadov Musa. A new method for solving linear ill-posed problems [J]. Appl Math Comput 2012;218:10180–7.
[13] Herdman Terry, Spies Ruben D, Temperini Karina G. Global saturation of regularization methods for inverse ill-posed problems. J Optim Theory Appl 2011;148:164–96.
[14] Shen Yunzhong, Xu Peiliang, Li Bofeng. Bias-corrected regularized solution to inverse ill-posed models. J Geod 2012;86(8):597–608.
[15] Riley JD. Solving systems of linear equations with a positive definite, symmetric, but possibly ill-conditioned matrix. Math Tables Aids Comput 1956;9:96–101.
[16] Golub G. Numerical methods for solving linear least squares problems. Numer Math 1965;7:206–16.
[17] Deng Xingsheng, Hua Xianghong, You Yangsheng. Transfer of height datum across seas using GPS leveling, gravimetric geoid and corrections based on a polynomial surface. Comput Geosci 2013;51:135–42.
[18] Wu Yan. Parametric inverse of severely ill-conditioned Hermitian matrices in signal processing. J Frankl Inst 2012;349:1048–60.
[19] Martin David R, Reichal Lothar. Minimization of functionals on the solution of a large-scale discrete ill-posed problem. BIT Numer Math 2013;53:153–73.
[20] Martin David R, Reichel Lothar. Projected Tikhonov Regularization of Large-Scale Discrete Ill-Posed Problems. J Sci Comput 2013. http://dx.doi.org/10.1007/s10915-013-9685-0.
[21] Fermín S, Bazán Viloche, Borges Leonardo S. GKB-FP: an algorithm for large-scale discrete ill-posed problems. BIT Numer Math 2010;50:481–507.
[22] Hochbruck M, Hönig M. On the convergence of a regularizing Levenberg– Marquardt scheme for nonlinear ill-posed problems. Numer Math 2010;115:71–9.
[23] Deng Xingsheng, Sun Honghong. Self-adaptive spectrum correction LU decomposition algorithm for solving a normal equation with severely ill-conditioned matrix. J Geod Geodyn 2014;34(6):135–9 [in Chinese].
[24] Press William H, Teukolsky Saul A, Vetterling William T, Brian P. Flannery numerical recipes in C: the art of scientific computing. 2nd ed. Cambridge: Cambridge University Press; 1992. p. 994.
[25] Deng Xingsheng, Wang Xinzhou. Incremental learning of dynamic fuzzy neural networks for accurate system modeling. Fuzzy Sets Syst 2009;160:972–87.
[26] Morigi S, Reichel L, Sgallari F. Noise-reducing cascadic multilevel methods for linear discrete ill-posed problems. Numer Algor 2010;53:1–22.

Deng Xingsheng, graduated from Wuhan University and obtained a doctoral degree in geodesy in 2007. He is an associate professor at the Department of Surveying Engineering, College of Traffic and Transportation Engineering, Changsha University of Science & Technology. His main research fields are geodetic data processing, and its applications in satellite positioning. He has hosted for over sixteen projects in surveying engineering fields and has published more than fifty papers, including over four papers in international journals.