INSTITUTE OF PHYSICS PUBLISHING INVERSE PROBLEMS

Inverse Problems **17** (2001) 281–291 www.iop.org/Journals/ip PII: S0266-5611(01)13706-8

Fast realization algorithms for determining regularization parameters in linear inverse problems

Yan-fei Wang[1] **and Ting-yan Xiao**[2]

[1] 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, PO Box 2719, Beijing, 100080, People's Republic of China
[2] Department of Applied Mathematics, Hebei University of Technology, Tianjin, 300130, People's Republic of China

E-mail: wyf@lsec.cc.ac.cn

Received 27 April 2000, in final form 11 December 2000

Abstract

In this paper we propose a cubically convergent algorithm. Our basic tool is the Tikhonov regularization and Morozov's and damped Morozov's discrepancy principles. Numerical experiments for integral equations of the first kind are presented to compare the efficiency of the proposed algorithms.

1. Introduction

It is well known that inverse problems are encountered in many fields of application ranging from science to engineering [3,5,11]. These problems often lead to solving operator equations of the first kind, which, by their nature, are ill-posed in the sense of Hadamard [3, 11]. This means that the required solution is extremely sensitive to the perturbations in the observation data. Thus some kind of regularization method must be utilized to obtain the stable resolution of the problems.

So far, a great amount of research work has focused on the development of appropriate strategies for selecting the regularization parameter (see [3, 4, 9, 12] and references therein). However, Kunisch and Zou [8] have pointed out that much less work has been carried out on the numerical realization of such strategies, and in fact it appears that very few of the strategies are utilized for practical applications. One of the causes may be the huge amount of computation required in the iterative process of choosing a reasonable regularization parameter.

Some existing efficient methods for solving the discrepancy principle are the Newton method with quadratical convergence and the quasi-Newton method with superlinear convergence (see [8]). This paper presents a cubic convergence algorithm. Like the Newton method and the quasi-Newton method, this new method could be used for most of the *posteriori* parameter choice strategies.

Let us consider a linear ill-posed inverse problem of the form

$$Az = u \tag{1}$$

0266-5611/01/020281+11$30.00

where $A : F \longrightarrow U$ is a bounded linear operator with domain $D(A)$ in a Hilbert space F and with its range $R(A)$ in a Hilbert space U and $u \in U$. Here we call problem (1) ill-posed in the sense that the solution of (1) does not depend continuously on the right-hand side data which are often obtained by measurement and hence contain errors. Let us assume that u_δ are observation data of u, and

$$\|u_\delta - u\| \leqslant \delta \tag{2}$$

with a given noise level $\delta > 0$. Now the computation of solution (1) from the observation data u_δ becomes an important topic.

Among the methods developed to solve linear ill-posed problems, Tikhonov regularization is the most well known one. In this method, the solution z_δ^α of the minimization problem

$$M^\alpha[z, u_\delta] = \{\|Az - u_\delta\|^2 + \alpha\|z - z^*\|^2\} \tag{3}$$

is used to approximate the solution of (1), where $\alpha > 0$ is the regularization parameter and $z^* \in D(A)$ is an *a priori* guess of solution (1). Under appropriate conditions on A, the stability of z_δ^α with respect to u_δ can be guaranteed, and with a suitable choice of α, z_δ^α can be guaranteed to converge to a z^*-minimum-norm-solution (z^*-MNS) $z^\dagger$ of (1), i.e. converge to an element $z^\dagger \in F$ with the property

$$Az^\dagger = u \qquad \text{and} \qquad \|z^\dagger - z^*\| = \min_{z \in D(A)} \{\|z - z^*\| : Az = u\}. \tag{4}$$

Now the regularization parameter α affects not only the convergence of z_δ^α but also the rates of convergence, and hence the choice of regularization parameter is vital.

Obviously (3) is equivalent to the following so-called Euler equation:

$$(A^*A + \alpha I)(z - z^*) = A^*(u_\delta - Az^*) \tag{5}$$

where A^* denotes the Hilbert-adjoint operator of A, I denotes the unit operator. The following theorem is verified by Kunisch and Zou in [8].

Theorem 1. *The solution z_δ^α of the Euler equation (5) is infinitely differentiable at every $\alpha > 0$, which satisfies the following equations:*

$$(A^*A + \alpha I)\frac{\mathrm{d}z_\delta^\alpha}{\mathrm{d}\alpha} = -(z_\delta^\alpha - z^*), \tag{6}$$

$$(A^*A + \alpha I)\frac{\mathrm{d}^k z_\delta^\alpha}{\mathrm{d}\alpha^k} = -k\frac{\mathrm{d}^{k-1} z_\delta^\alpha}{\mathrm{d}\alpha^{k-1}}, \qquad k = 2, 3, \ldots. \tag{7}$$

2. A cubic convergence algorithm

Morozov's discrepancy principle has been used for linear ill-posed problems to choose the regularization parameter and α is determined from the following nonlinear equation:

$$\|Az_\delta^\alpha - u_\delta\| = \delta. \tag{8}$$

In some applications, the Morozov principle may not be so satisfactory. For example, if the exact solution $z^\dagger$ satisfies $z^\dagger - z^* \in R((A^*A)^\nu)$ for some $\nu > \frac{1}{2}$, the optimal convergence of the regularized solutions is not obtained [3]. We therefore consider a more general class of the damped Morozov principle [7, 9] given by

$$\|Az_\delta^\alpha - u_\delta\|^2 + \alpha^\gamma \|z_\delta^\alpha\|^2 = \delta^2 \tag{9}$$

where $\gamma \in [1, \infty]$. Obviously, the exact Morozov principle (8) is a special case of the damped case with $\gamma = \infty$.

Throughout this section we assume that $u_\delta \notin \ker A^*$. We observe that equations (8) or (9) can be expressed in terms of α as

$$\phi(\alpha) := \|Az_\delta^\alpha - u_\delta\|^2 - \delta^2 \tag{10}$$

or

$$\phi(\alpha) := \|Az_\delta^\alpha - u_\delta\|^2 + \alpha^\gamma \|z_\delta^\alpha\|^2 - \delta^2. \tag{11}$$

The following lemma (also in [8]) tells us that the first-order derivative of $\phi(\alpha)$ is positive.

Lemma 2. *Let $\phi(\alpha)$ be defined as (10) or (11) and suppose that $u_\delta \notin \ker A^*$. Then for equation (10), $\phi'(\alpha) > 0$, $\forall \alpha \in (0, \infty)$; for equation (11), $\phi'(\alpha) > 0$, $\forall \alpha \in (0, 1]$.*

For practical purposes, it certainly suffices to restrict α to $(0, 1]$.

From theorem 1 we know $\phi(\alpha)$ is infinitely differentiable with respect to α, so by truncating the Taylor expansion after the term $(\alpha - \alpha_n)^3$, we have

$$\phi(\alpha) = \phi(\alpha_n) + \phi'(\alpha_n)(\alpha - \alpha_n) + \tfrac{1}{2}(\alpha - \alpha_n)^2 \phi''(\xi_n) \tag{12}$$

where ξ_n is between α and α_n.

From (12), an iterative formula can be immediately obtained as follows:

$$\alpha_{n+1} = \alpha_n - \frac{2\phi(\alpha_n)}{\phi'(\alpha_n) + (\phi'(\alpha_n)^2 - 2\phi(\alpha_n)\phi''(\alpha_n))^{\frac{1}{2}}}. \tag{13}$$

Without loss of generality, take (10) as an example. Let $\beta(\alpha) = \|z_\delta^\alpha\|^2$, $\phi'(\alpha)$, $\phi''(\alpha)$ can be computed as follows:

$$\phi'(\alpha) = -\alpha\beta'(\alpha) \tag{14}$$

$$\phi''(\alpha) = -\beta'(\alpha) - 2\alpha \left[\left\| \frac{\mathrm{d}z_\delta^\alpha}{\mathrm{d}\alpha} \right\|^2 + \left(z_\delta^\alpha, \frac{\mathrm{d}^2 z_\delta^\alpha}{\mathrm{d}\alpha^2} \right) \right] \tag{15}$$

where $\beta'(\alpha) = 2(\frac{\mathrm{d}z_\delta^\alpha}{\mathrm{d}\alpha}, z_\delta^\alpha)$. Finding z_δ^α, $\mathrm{d}z_\delta^\alpha/\mathrm{d}\alpha$, $\mathrm{d}^2 z_\delta^\alpha/\mathrm{d}\alpha^2$ at the kth step will lead to the solutions of the following equations:

$$(A^*A + \alpha I)(z_{\alpha_k} - z^*) = A^*(u_\delta - Az^*) \tag{16}$$

$$(A^*A + \alpha I) z'_{\alpha_k} = -(z_{\alpha_k} - z^*) \tag{17}$$

$$(A^*A + \alpha I) z''_{\alpha_k} = -2z'_{\alpha_k}. \tag{18}$$

From equations (16)–(18) we observe that their differences only occur in their right-hand sides. This suggests that, by utilizing Cholesky factorization (we can choose $\alpha > 0$ such that $A^*A + \alpha I$ is positive definite) once only with back substitution three times, we can obtain the vectors z_{α_k}, z'_{α_k}, z''_{α_k}. Now we compute the number of operations for our algorithm: first to form A^*A (only once), the cost is n^3; then in each step of α, the cost of the Cholesky factorization is $n^3/6$; back substitution three times is $3n^2$; computing $\phi(\alpha)$, $\phi'(\alpha)$ and $\phi''(\alpha)$ the costs are $n^2 + 2n$, n and $2n$ respectively. So after A^*A is formed, the number of operations is $\frac{1}{6}n^3 + 4n^2 + 5n$ in each iteration. Since the cost of computing $\phi''(\alpha)$ is $2n$ and an additional back substitution is n^2, thus there is only a small amount of computation added compared to the Newton method, but the convergence rate is greatly improved so that the CPU time is also greatly saved. It should be pointed out here that Eldén [1, 2] gave us algorithms for the regularization of ill-conditioned least squares problems. In his approach (see [1] for details), he first performed a bidiagonalization of the matrix A once in $\mathrm{O}(\frac{4}{3}n^3)$ operations and then there were only $\mathrm{O}(n)$ operations for each iteration. If the number of iterations is larger than 5 to 7 then his approach greatly reduces the number of operations and the number of iterations

is no longer so essential. However, in our opinion, our method is easier to code and perform compared to his approach and it is more efficient than the traditional Newton method and quasi-Newton method.

Now we give the following algorithm.

Algorithm 1 (Cubic convergence algorithm).

Step 1. Input $\alpha_0 > 0$, $\delta > 0$, ϵ(tolerance) > 0, A, $k_{\max}$, u_δ, z^*, set $k := 0$;
Step 2. Solve equations (16)–(18);
Step 3. Compute $\phi(\alpha_k)$, $\phi'(\alpha_k)$ and $\phi''(\alpha_k)$;
Step 4. Solve for α_{k+1} from iterative formula (13).
Step 5. If $\mid \alpha_{k+1} - \alpha_k \mid \leqslant \epsilon$ or $k = k_{\max}$, STOP; otherwise, set $k := k + 1$, GOTO step 2.

Notice that (13) can be written as

$$\alpha_{k+1} = \alpha_k - \frac{\phi(\alpha_k)}{\phi'(\alpha_k)} \frac{2}{1 + (1 - 2t(\alpha_k))^{\frac{1}{2}}}, \tag{19}$$

where, $t(\alpha_k) = \frac{\phi''(\alpha_k)\phi(\alpha_k)}{\phi'(\alpha_k)^2}$. Define the iteration function:

$$F(\alpha) = \alpha - \frac{\phi(\alpha)}{\phi'(\alpha)} G(\alpha), \tag{20}$$

here, $G(\alpha) = \frac{2}{1+(1-2t(\alpha))^{\frac{1}{2}}}$. Then (19) is equivalent to

$$\alpha_{k+1} = F(\alpha_k). \tag{21}$$

We know the functions $\phi(\alpha)$, $G(\alpha)$ are both infinitely differentiable, and if we let α^* be the single root of $\phi(\alpha) \in (0, 1)$, then $t(\alpha^*) = 0$, $G(\alpha^*) = 1$. It should be pointed out here that it can be ensured that the term $1-2t(\alpha)$ is positive. Since $\alpha \in U(\alpha^*, \epsilon)$ (i.e., the ϵ-neighbourhood of α^*), and notice that $\phi''(\alpha)$ is bounded in $(0, 1)$, so $1-2t(\alpha) = 1-2\frac{\phi''(\alpha)\phi(\alpha)}{\phi'(\alpha)^2}$ can be ensured to be positive in $U(\alpha^*, \epsilon)$ if ϵ is sufficiently small. Therefore the iteration formula makes sense. Now we introduce a well known general pth order convergence result in the following theorem.

Theorem 3. *Let $\alpha_{k+1} = F(\alpha_k)$ be an iteration process, if $F^{(p)}(\alpha)$ $(p = 1, 2, \ldots)$ are continuous at $U(\alpha^*, \epsilon)$, and*

$$F'(\alpha^*) = F''(\alpha^*) = \cdots = F^{(p-1)}(\alpha^*) = 0, \qquad F^{(p)}(\alpha^*) \neq 0,$$

then the iteration process is pth order convergent at $U(\alpha^*, \epsilon)$.

Now we give a short proof of the following theorem.

Theorem 4. *Assume that equations (10) or (11) have a root at $\alpha = \alpha^*$, i.e. $\phi(\alpha^*) = 0$, then $\exists\ U(\alpha^*, \epsilon)$ and $\epsilon > 0$, such that $\forall \alpha_0 \in U(\alpha^*, \epsilon)$, the sequence $\{\alpha_k\}_{k=1}^{\infty}$ generated by the above algorithm is locally cubically convergent.*

Proof. Obviously $F(\alpha)$ in (20) is infinitely differentiable with respect to α at $U(\alpha^*, \epsilon)$. Defining $s(\alpha) = \frac{\phi(\alpha)}{\phi'(\alpha)}$, we compute that

$$s'(\alpha) = 1 - t(\alpha),$$
$$s''(\alpha) = -t'(\alpha) = -\frac{\phi''(\alpha)}{\phi'(\alpha)} + \phi(\alpha)\left(\frac{2\phi''(\alpha)^2}{\phi'(\alpha)^3} - \frac{\phi'''(\alpha)}{\phi'(\alpha)^2}\right).$$

Therefore,

$$F'(\alpha^*) = 1 - s'(\alpha^*)G(\alpha^*) - s(\alpha^*)G'(\alpha^*) = 1 - G(\alpha^*) = 1 - 1 = 0,$$

$$F''(\alpha^*) = -s''(\alpha^*)G(\alpha^*) - 2s'(\alpha^*)G'(\alpha^*) - s(\alpha^*)G''(\alpha^*) = \frac{\phi''(\alpha^*)}{\phi'(\alpha^*)} - t'(\alpha^*) = 0.$$

In the same way we can verify that $F'''(\alpha^*) \neq 0$, so by theorem 3 above, theorem 4 holds true. □

Remark 1. Algorithm 1 is locally cubically convergent. But in the numerical experiment, the initial value α_0 need not be chosen too strictly. Algorithm 1 is almost convergent for arbitrary $\alpha_0 \in (0, 1)$. In [2, p 234], an approach was shown for Newton's method that guarantees global convergence under some conditions by applying Newton's method to $\psi(\beta) = \phi(1/\beta)$, i.e. α is replaced by $1/\beta$. For our algorithm, we must ensure the term $\phi'(\alpha)^2 - 2\phi(\alpha)\phi''(\alpha)$ is non-negative in each step, so the convergence is locally cubic.

3. Hybrid algorithms

A two-parameter algorithm has been suggested in [9], which is efficient for choosing an initial value α, but not for the whole iterative process.

Let $f(\alpha)$ denote the minimal value function of $\frac{1}{2}M^{\alpha}[z, u_\delta]$, i.e.

$$f(\alpha) = \tfrac{1}{2}M^{\alpha}[z(\alpha), \delta] = \tfrac{1}{2}\|Az(\alpha) - u_\delta\|^2 + \tfrac{1}{2}\alpha\|z(\alpha) - z^*\|^2 \tag{22}$$

we have

$$f'(\alpha) = \tfrac{1}{2}\|z(\alpha) - z^*\|^2 \qquad \text{and} \qquad f''(\alpha) = (z(\alpha) - z^*, z'(\alpha)) \qquad \forall\alpha > 0. \tag{23}$$

Throughout this section we assume that $u_\delta \notin \ker A^*$. We observe that equations (10) and (11) can be expressed in terms of $f(\alpha)$ as

$$f(\alpha) - \alpha f'(\alpha) = \delta^2 \tag{24}$$

and

$$f(\alpha) + (\alpha^\gamma - \alpha) f'(\alpha) = \delta^2 \tag{25}$$

respectively, where $\gamma \in [1, \infty]$. Morozov [9] has verified that the regularization parameter α satisfies a model function

$$m(\alpha) = C\left(1 - \frac{T}{T + \alpha}\right) \tag{26}$$

where T, C are two parameters. To update the two parameters C and T in the above model function and consequently solve the Morozov equations (10) or (11) approximately, we have the following algorithm [9].

Algorithm 2 (Two-parameter algorithm). Set $k := 0$ and choose $\alpha_0 > 0$, $\epsilon > 0$.

Step 1. Compute $f'(\alpha_k)$ and $f(\alpha_k)$ using (22), (23). Compute T_k and C_k from

$$m(\alpha_k) = C_k\left(1 - \frac{T_k}{T_k + \alpha_k}\right) = f(\alpha_k), \tag{27}$$

$$m'(\alpha_k) = \frac{C_k T_k}{(T_k + \alpha_k)^2} = f'(\alpha_k). \tag{28}$$

Step 2. Set

$$m(\alpha) = C_k\left(1 - \frac{T_k}{T_k + \alpha}\right).$$

Step 3. Solve for α_{k+1} the Morozov equation

$$m(\alpha) - \alpha m'(\alpha) = \delta^2. \tag{29}$$

Step 4. If $| \alpha_{k+1} - \alpha_k | \leqslant \epsilon$, STOP; otherwise set $k := k + 1$ GOTO (1).

In step 1 of algorithm 2, one needs to compute T_k and C_k from (27) and (28). Combining (27) and (28) we have

$$T_k = \frac{\alpha_k^2 f'(\alpha_k)}{f(\alpha_k) - \alpha_k f'(\alpha_k)}, \qquad C_k = \frac{f^2(\alpha_k)}{f(\alpha_k) - \alpha_k f'(\alpha_k)}. \tag{30}$$

Since $u_\delta \notin \ker A^*$, the denominators in (30) do not vanish.

Kunisch and Zou [8] noticed that the two-parameter algorithm is only useful during the first few iterations (one to three iterations), so combining algorithm 1 with 2, we give a hybrid algorithm.

Algorithm 3 (Hybrid algorithm).

Step 1. Select α from algorithm 2 after two iterates;
Step 2. Take the above α as the initial value α_0 in algorithm 1 and implement algorithm 1.

We can also consider combining the Newton method with the two-parameter algorithm (TPA) (see [8]), since the TPA is useful for giving a good initial guess value. Another hybrid algorithm is given in [8], i.e. combining the quasi-Newton method with the TPA.

Remark 2. Since the two-parameter strategy can provide us with a good initial guess α_0 value, the hybrid algorithms will be much better.

4. Numerical experiments

The purpose of this final section is to illustrate the theory from the previous sections with two numerical examples. The numerical experiments are completed with MATLAB 5.1 on an SGI workstation.

Throughout this section TPA denotes the two-parameter algorithm [8], Newton + TPA denotes Newton's method with the TPA, QN + TPA denotes the quasi-Newton method [8] with the TPA, CCA denotes algorithm 1 and HA denotes the hybrid algorithm.

The first example is a one-dimensional model problem in image reconstruction from [10] (also in [6]), which solves the Fredholm integral equations of the first kind:

$$(Az)(s) = \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} k(s,t)z(t)\,\mathrm{d}t = u(s), \qquad s, t \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \tag{31}$$

with kernel

$$k(s,t) = (\cos s + \cos t)\left(\frac{\sin r}{r}\right), \qquad r = \pi(\sin s + \sin t).$$

For the solution z we choose a simple function with two 'humps':

$$z(t) = 2\exp(-6(t - 0.8)^2) + \exp(-2(t + 0.5)^2) \tag{32}$$

as the true solution z_{T}, and we have chosen the values of the right-hand side $u(s)$ of (31) on the grid$\{s_i\}_{i=1}^m$ in $[-\frac{\pi}{2}, \frac{\pi}{2}]$ in accordance with the following rule.

As vector u at the right-hand side we have used the vector obtained by multiplication of the $(m \times n)$-dimensional matrix A, approximating the operator in (31), by the column vector z_{T} of values of the exact solution on the grid $\{t_i\}_{i=1}^n$ in the interval $[-\frac{\pi}{2}, \frac{\pi}{2}]$:

$$u_i = \sum_{j=1}^{n} A_{ij} z_{\mathrm{T}}(t_j). \tag{33}$$

Table 1. The comparison of efficiency for five algorithms for Morozov's discrepancy principle.

Algorithms	Iter.	α^*	$\|z_\delta^{\alpha^*} - z_T\|$	$\|Az_\delta^{\alpha^*} - u_\delta\|/\|u_\delta\|$
Newton + TPA	5	1.1176e−6	1.77e−2	6.2527e−6
CCA	9	1.1176e−6	1.77e−2	6.2528e−6
TPA	7	1.1176e−6	1.77e−2	6.2526e−6
HA	5	1.1176e−6	1.77e−2	6.2526e−6
QN + TPA	13	1.1176e−6	1.77e−2	6.2527e−6

This way of choosing the right-hand side guarantees that the minimum of the discrepancy functional $\|Az - u_\delta\|^2$ on this set of vectors will be zero. This property of the solution is essential when using properties of iteration algorithms for solving ill-posed problems. Tikhonov *et al* [12] has pointed out that a common property of the majority of iteration algorithms is the rapid decline of the functional. Therefore an important characteristic of iteration algorithms is the actual minimal level of the discrepancy functional up to which the minimization process runs in real time. This parameter makes it possible to estimate beforehand the error in specifying the initial information for which it makes sense, then apply the given method. Or, conversely, to choose on the basis of the error in specifying the initial information an algorithm that is most suitable for solving the given problem. So, if in model problems we can minimize in real time the discrepancy functional to the 1% level (in relation to the norm of the right-hand side), then it is clear that when using this algorithm we can, in general, successfully solve problems in which the error of specifying the initial information is 0.1%.

For this reason, we study the level up to which we can minimize the discrepancy and use the right-hand side of (31) computed in accordance with (33).

To evaluate the integrals involved, we choose $m = n = 100$ to divide the interval into 100 subintervals, and on each subinterval the rectangular quadrature rule is used. The accuracy of specifying the right-hand side is assumed to be equal to $\delta = 1.0 \times 10^{-4}$.

In general, when solving (31) we always have to keep track of the fact that the error of approximating the integral in (31) is substantially smaller than the error δ in specifying the right-hand side. For this it is necessary either to choose sufficiently dense grids (thus increasing the dimension of the problem and bringing about a substantial increase of computational time expenditure) or to use more exact quadrature formula.

We should point out here that in (13) the term $\phi'(\alpha)^2 - 2\phi(\alpha)\phi''(\alpha)$ may be negative if the initial α_0 value is chosen too approximately. We therefore add a technique in our numerical test. That is, we denote $\Delta = \phi'(\alpha)^2 - 2\phi(\alpha)\phi''(\alpha)$, if $\Delta < 0$, and we take $|\ \mathrm{Re}\ (\sqrt{\Delta})\ |$; otherwise, we take $\sqrt{\Delta}$, where Re represents the real part of $\sqrt{\Delta}$. But in our numerical example, it was never activated.

The comparison of the results of the above five algorithms are shown in tables 1–4. For tables 1–3, the error level is chosen as $\delta = 1.0 \times 10^{-4}$; table 4 gives us a comparison of different error level δ values for the five algorithms. We now give some notations: z_δ^α is defined as in the previous section, α^* stands for the final chosen regularization parameter in the iteration process; iter. stands for iteration steps; $k_{\max} = 200$, $z^* = 0$, α_0 is chosen as 0.1.

From table 1 we see that the parameters and the absolute errors obtained by the above five methods are the same. Newton + TPA and HA are the two fastest algorithms.

Table 2 compares the CPU time and the iterations of the five algorithms. With an increase of the discrete matrix dimension, the HA and the Newton + TPA will save a lot of time compared to the other algorithms.

Table 2. The comparison of CPU time for five algorithms for Morozov's discrepancy principle, where [·] denotes iterates.

m, n	Newton + TPA	CCA	TPA	HA	QN + TPA
100	0.05[5]	0.12[9]	0.08[7]	0.06[5]	0.12[13]
200	0.45[7]	0.77[10]	0.73[11]	0.44[6]	0.84[15]
300	1.31[6]	2.75[11]	2.02[9]	1.18[5]	3.17[16]
400	3.38[6]	6.79[11]	5.14[9]	2.99[5]	8.39[16]
600	13.08[7]	21.89[11]	17.12[9]	9.70[5]	28.45[16]

Table 3. The impact of the γ value on the choices of regularization parameters by utilizing HA for damped Morozov's discrepancy principle.

γ	Iter.	α^*	$\|z_\delta^{\alpha^*} - z_{\mathrm{T}}\|$	$\|Az_\delta^{\alpha^*} - u_\delta\|/\|u_\delta\|$
1.0	6	1.0036e−10	1.6206e−4	1.9322e−8
1.2	10	4.6559e−9	2.4234e−4	3.0637e−8
1.3	12	2.2031e−8	6.7713e−4	1.1813e−7
1.5	11	2.1053e−7	3.9000e−3	1.2117e−6
2.0	10	1.1105e−6	1.7600e−2	6.2141e−6
4.0	10	1.1176e−6	1.7700e−2	6.2527e−6
∞	10	1.1176e−6	1.7700e−2	6.2528e−6

Table 4. Comparison of the five algorithms for Morozov's discrepancy principle with different error level δ.

Error level	$\delta = 10\%$		$\delta = 1\%$		$\delta = 0.1\%$	
Algorithms	Iter.	$\|z_\delta^{\alpha^*} - z_{\mathrm{T}}\|$	Iter.	$\|z_\delta^{\alpha^*} - z_{\mathrm{T}}\|$	Iter.	$\|z_\delta^{\alpha^*} - z_{\mathrm{T}}\|$
Newton + TPA	5	3.8740e−1	4	1.2910e−1	3	4.46e−2
CCA	3	3.8740e−1	4	1.2920e−1	5	4.03e−2
TPA	8	3.8740e−1	6	1.2900e−2	5	3.91e−2
HA	3	3.8740e−1	3	1.2900e−2	2	4.46e−2
QN + TPA	7	3.8740e−1	8	1.2900e−2	5	4.94e−2

Table 3 compares the impact of the γ value on the choice of regularization parameter for damped Morozov's discrepancy principle. With the appropriate choice of γ value, we can improve the accuracy of the solution. Obviously, as $\gamma = 1.0, 1.2, 1.3$, the absolute error $\|z_\delta^{\alpha^*} - z_{\mathrm{T}}\|$ is much smaller than using Morozov's discrepancy principle.

Table 4 gives us a comparison of the five algorithms for different error levels. From the table we see that the HA is the most efficient algorithm to obtain the same accuracy of the solution.

Figure 1 gives us a plot of the exact solution (dotted curve) and the approximate solution (open circles) by utilizing the HA algorithm. We use the damped Morozov discrepancy principle for this example, in which γ is chosen as $1.0, \alpha_0 = 0.1, \delta = 1.0 \times 10^{-4}, m = n = 50$.

We close this section with a second example from [13]. Varah [13] presented an example for inverse Laplace transform. In the notation of equation (1) the problem he considered is defined by

$$(Af)(s) = \int_{t_{\min}}^{t_{\max}} k(s,t) f(t)\,\mathrm{d}t = g(s), \qquad s \in [s_{\min}, s_{\max}], \tag{34}$$

where, the integral kernel is $k(s,t) = \exp(-st)$, the exact right-hand side is $g(s) = \frac{1}{(s+1)^2}$, $[t_{\min}, t_{\max}) = [0, \infty)$, $(s_{\min}, s_{\max}) = (-1, \infty)$. Through simple manipulation, we compute

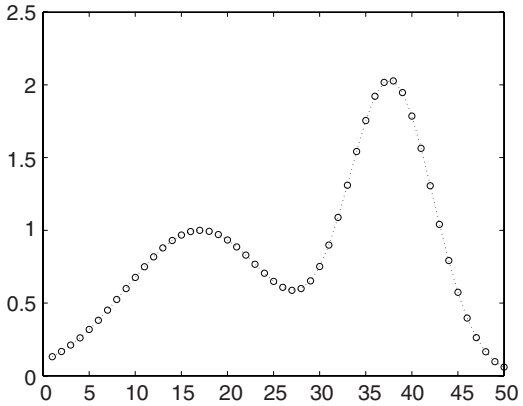

Figure 1. Reconstruction results: approximate solution, open circles; exact solution, dotted curve.

Table 5. The comparison of efficiency for five algorithms with $\alpha_0 = 0.1$, $\hat{\delta} = 1\%$ for Morozov's discrepancy principle.

Algorithms	Iter.	α^*	$\|f_\delta^{\alpha^*} - f_T\|$	$\|Af_\delta^{\alpha^*} - g_\delta\|/\|g_\delta\|$
Newton + TPA	6	3.4279e−4	3.46e−2	5.50e−3
CCA	5	3.4270e−4	3.30e−2	5.50e−3
TPA	69	3.6851e−4	3.61e−2	5.50e−3
HA	4	3.4270e−4	3.30e−2	5.50e−3
QN + TPA	10	3.7455e−4	3.67e−2	5.50e−3

Table 6. Convergence of the HA with $\alpha_0 = 0.1$, $\gamma = 1.4$ for damped Morozov's discrepancy principle.

$\tilde{\delta}$	1%	2%	4%	7%	10%
Iter.	6	6	6	5	5
α^*	7.1807e−5	1.8287e−4	4.6172e−4	9.7207e−4	1.6e−3
$\|f_\delta^{\alpha^*} - f_T\|$	2.59e−2	3.06e−2	3.82e−2	5.24e−2	6.53e−2
$\dfrac{\|Af_\delta^{\alpha^*} - g_\delta\|}{\|g_\delta\|}$	5.40e−3	1.08e−2	2.15e−2	3.76e−2	5.36e−2

that the exact solution is $f_T(t) = t\exp(-t)$. In order to simulate measurement inaccuracies we add noise to the right-hand side g as follows:

$$g_\delta(s) = g + \hat{\delta} \times \mathrm{rand}(s),$$

where rand(s) is random perturbation to the right-hand side and $\hat{\delta}$ is the noise level. Our purpose is to numerically reconstruct the exact solution f from g_δ.

In our simulations we restrict $[t_{\min}, t_{\max}] = [s_{\min}, s_{\max}] = [0, 10]$. We choose $m = n = 100$ and use the mid-point rule to get a discrete equation. Let the other notations be the same as in the first example; the numerical results are shown in tables 5–8.

From tables 5–7 we see that the HA algorithm works really well even with Gaussian white noise. The absolute error $\|f_\delta^{\alpha^*} - f_T\|$ can keep up to $O(10^{-2})$.

Table 8 gives us a comparison of the impact of the γ value on the choice of regularization parameter for the damped Morozov discrepancy principle. The noise level is $\hat{\delta} = 5\%$. We see that the absolute error $\|f_\delta^{\alpha^*} - f_T\|$ for $\gamma = 1.5$ is smaller than for the other γ value.

Table 7. Convergence of the HA with $\alpha_0 = 0.1$ for Morozov's discrepancy principle.

$\tilde{\delta}$	1%	2%	4%	7%	10%
Iter.	4	4	4	5	5
α^*	3.4270e−4	7.7413e−4	1.7000e−3	3.3000e−3	4.9000e−3
$\|f_\delta^{\alpha^*} - f_{\mathrm{T}}\|$	3.30e−2	5.21e−2	7.12e−2	8.65e−2	9.87e−2
$\frac{\|Af_\delta^{\alpha^*} - g_\delta\|}{\|g_\delta\|}$	5.50e−3	1.10e−2	2.19e−2	3.82e−2	5.44e−2

Table 8. The impact of γ's value on the choices of regularization parameters by utilizing HA for the damped Morozov discrepancy principle, $\hat{\delta} = 5\%$, $\alpha_0 = 0.1$.

γ	Iter.	α^*	$\|f_\delta^{\alpha^*} - f_{\mathrm{T}}\|$	$\|Af_\delta^{\alpha^*} - g_\delta\|/\|g_\delta\|$
1.0	8	4.1139e−5	3.075e−1	2.67e−2
1.2	6	2.1140e−4	1.20e−1	2.68e−2
1.3	6	6.2148e−4	4.28e−2	2.69e−2
1.5	5	9.1251e−4	3.80e−2	2.70e−2
2.0	5	2.10e−3	7.28e−2	2.73e−2
4.0	5	2.20e−3	7.71e−2	2.73e−2
∞	5	2.20e−3	7.71e−2	2.73e−2

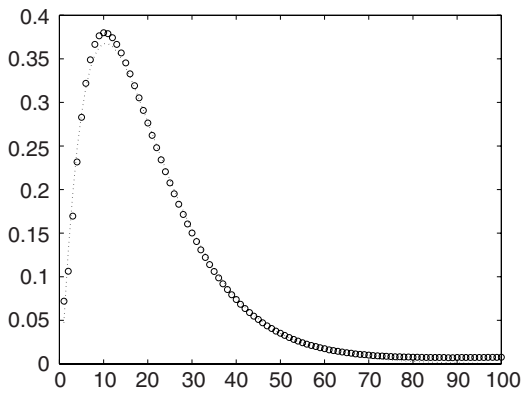

Figure 2. Reconstructions after five iterations.

Therefore the choice of γ is crucial. For actual computations one will have to use experience with synthetic data to choose γ.

Figure 2 gives us a plot of the exact solution (dotted curve) and the approximate solution (open circles) by utilizing the HA algorithm. We use the damped Morozov discrepancy principle for this example; where γ is chosen as 1.4, $\alpha_0 = 0.1$, $\hat{\delta} = 10\%$, $m = n = 100$.

5. Conclusion

From the above two examples we conclude that the HA is the fastest algorithm. This is not by chance. Because the TPA can ensure the selection of a reasonable initial α_0 value, with this α_0 value the CCA can realize the fast numerical implementation of selecting the regularization parameter so as to obtain the stable regularization solution.

However, we should point out that all the algorithms presented in this paper use direct

solvers to solve equations (16), or (17) and (18). As a consequence of this the time cost is still high, especially for large-scale problems (say, two- and three-dimensional problems). We suggest the user applies either our cubic convergent method or Newton's method or hybrid algorithms for problems of small size or for one-dimensional problems. For large problems, one should use the iterative solvers (such as the conjugate gradient method) instead.

Acknowledgments

This paper was completed with financial support from the Natural Science Foundation of Hebei Province grant no 698047. The first author was also partially supported by Chinese NSF grant 19731010 and the Knowledge Innovation Program of CAS.

References

[1] Eldén L 1977 Algorithms for the regularization of ill-conditioned least squares problems *BIT* **17** 134–45
[2] Eldén L 1984 A note on the computation of the generalized cross-validation function for ill-conditioned least squares problems *BIT* **24** 467–72
[3] Engl H W, Hanke M and Neubauer A 1996 *Regularization of Inverse Problems* (Dordrecht: Kluwer)
[4] Groetsch C W 1984 *The Theory of Tikhonov Regularization for Fredholm Equations of the First Kind* (Boston, MA: Pitman)
[5] Groetsch C W 1993 *Inverse Problems in the Mathematical Sciences* (Braunschweig: Vieweg)
[6] Hansen P C 1992 Numerical tools for analysis and solution of Fredholm integral equations of the first kind *Inverse Problems* **8** 849–72
[7] Kunisch K 1993 On a class of damped Morozov principles *Computing* **50** 185–98
[8] Kunisch K and Zou Jun 1998 Iterative choices of regularization parameter in linear inverse problems *Inverse Problems* **14** 1247–64
[9] Morozov V A 1984 *Methods for Solving Incorrectly Posed Problems* (New York: Springer)
[10] Shaw C B Jr 1972 Improvement of the resolution of an instrument by numerical solution of an integral equation *J. Math. Anal. Appl.* **37** 83–112
[11] Tikhonov A N and Arsenin V Y 1977 *Solutions of Ill-Posed Problems* (New York: Wiley)
[12] Tikhonov A N, Goncharsky A V, Stepanov V V and Yagola A G 1995 *Numerical Methods for the Solution of Ill-Posed Problems* (Dordrecht: Kluwer)
[13] Varah J M 1973 On the numerical solution of ill-conditioned linear systems with application to ill-posed problems *SIAM J. Numer. Anal.* **10** 257–67