Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2011, Article ID 467017, 12 pages
doi:10.1155/2011/467017

Research Article

A Low Memory Solver for Integral Equations of Chandrasekhar Type in the Radiative Transfer Problems

Mohammed Yusuf Waziri, Wah June Leong, Malik Abu Hassan, and Mansor Monsi

Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, 43400 Serdang, Malaysia

Correspondence should be addressed to Mohammed Yusuf Waziri, mywaziri@gmail.com

Received 4 February 2011; Revised 14 July 2011; Accepted 8 August 2011

Academic Editor: Alexei Mailybaev

The problems of radiative transfer give rise to interesting integral equations that must be faced with efficient numerical solver. Very often the integral equations are discretized to large-scale nonlinear equations and solved by Newton's-like methods. Generally, these kind of methods require the computation and storage of the Jacobian matrix or its approximation. In this paper, we present a new approach that was based on approximating the Jacobian inverse into a diagonal matrix by means of variational technique. Numerical results on well-known benchmarks integral equations involved in the radiative transfer authenticate the reliability and efficiency of the approach. The fact that the proposed method can solve the integral equations without function derivative and matrix storage can be considered as a clear advantage over some other variants of Newton's method.

1. Introduction

The study of Chandrasekhar's integral equation involved in the radiative transfer problem has been a foremost subject of much investigations and was first formulated by Chandrasekhar [1] in 1960. It arose originally in connection with scattering through a homogeneous semi-infinite plane atmosphere and since it has been used to model diverse forms of scattering via the H-function of Chandrasekhar [2], defined by

$$H(x) = 1 + H(x)\int_0^1 \frac{x}{x+t}\psi(t)H(t)dt. \tag{1.1}$$

Chandrasekhar H-function plays a crucial role in radiative transfer and transport theory [3, 4]. Since then, there have been diverse solvers of (1.1). It is well known that the numerical solution of Chandrasekhar integral equation is difficult to be obtained [5], and thus it is convenient to have a reliable and efficient solver. The problem of finding approximate solution of such integral equations is still popular today, and various methods of solving these integral equations have been established [5–7]. The common approach for the approximate solution of (1.1) is at first discretizing (1.1) by a vector $\overline{x} \in R^n$, then replacing the integrals by quadrature sums and the derivatives by difference quotients involving only the components of $\overline{x} \in R^n$ (see [8], e.g.). By doing so, (1.1) becomes a problem of finding the solution of system of n nonlinear equations with n unknowns

$$F(\overline{x}) = 0, \tag{1.2}$$

where $F : R^n \rightarrow R^n$ is a nonlinear mapping. Often, the mapping F is assumed to satisfy the following assumptions:

(A1) there exists $\overline{x}^* \in R^n$ s.t. $F(\overline{x}^*) = 0$,

(A2) F is a continuously differentiable mapping in an open neighborhood of $\overline{x}^*$,

(A3) $F'(\overline{x}^*)$ is invertible.

The famous iterative method for solving (1.2) is the classical Newton's method, where the Newtonian iteration is given by

$$x_{k+1} = x_k - \left(F'(x_k)\right)^{-1} F(x_k), \quad k = 0, 1, 2, \ldots. \tag{1.3}$$

The convergence rate for the Newton's method is quadratic from any initial point x_0 in the neighborhood of $\overline{x}^*$ [9]. However, an iteration of (1.3) turns to be expensive, because it requires to compute and store the Jacobian matrix, as well as solving Newton's system which is a linear system in each iteration. The major difficulty of Newton's type method is the matrix storage requirements especially when handling large systems of nonlinear equations [5, 6, 9]. There are quite a number of revised Newton's type methods being introduced, which include fixed Newton's and quasi-Newton's, to diminish the weakness of (1.3). Fixed Newton method [10] for the determination of solution x^* is given by

$$x_{k+1} = x_k - \left(F'(x_0)\right)^{-1} F(x_k), \quad k = 0, 1, 2, \ldots. \tag{1.4}$$

The method avoids computation and storing the Jacobian in each iteration (except at $k = 0$). However, it still requires solving the systems of n linear equations and may consume more CPU time as the system's dimension increases [10].

A quasi-Newton's method is another variant of Newton's type methods, and it replaces the Jacobian or its inverse with an approximation which can be updated at each iteration [11], and its updating scheme is given by

$$x_{k+1} = x_k - B_k^{-1} F(x_k), \tag{1.5}$$

where the matrix B_k is the approximation of the Jacobian at x_k. The main idea behind quasi-Newton's method is to eliminate the evaluation cost of the Jacobian matrix, in which if function evaluations are very expensive, the cost of finding a solution by quasi-Newton's methods could be much smaller than some other Newton's-like methods [7, 12, 13]. Various Jacobian approximations matrices such as the Broyden's method [11, 14] are proposed. However, the most critical part of such solvers is that they need the storage of full matrix of the approximate Jacobian, which can be a very expensive task as the dimension of systems increases [15]. In this paper, we propose an alternative approximation to the Jacobian inverse into a diagonal matrix by means of variational techniques. It is worth mentioning that the suggested method can be applied to solve (1.2) without the cost of computing or storing the true Jacobian. Hence, it can reduce computational cost, storage requirement, processing time (CPU time) and also eliminates the need for solving n linear equations in each iteration. The proposed method works efficiently, and the results so far are very encouraging. This paper is arranged as follows; we present our proposed method in Section 2; numerical results are reported in Section 3; finally conclusion is given in Section 4.

2. Chandrasekhar H-Equation

In this section, we present the detailed process of discretizing the Chandrasekhar-type integral equations in the radiative transfer problem. Chandrasekhar and Breen [16] compute H-equation as the solution of the nonlinear integral equation

$$H(x) - c\frac{x}{2}H(x)\int_0^1 \frac{H(y)}{x+y}dy = 1, \tag{2.1}$$

where $c \in [0,1]$ and $H : [0,1] \rightarrow \mathbb{R}$ is an unknown continuous function. From (2.1), we obtain

$$H(x)\left[1 - \frac{c}{2}\int_0^1 \frac{xH(y)}{x+y}dy\right] = 1. \tag{2.2}$$

Let us partition $[0,1]$ into n subinterval, $0 < x_1 < \cdots < x_j = j/n < \cdots < 1$. Denote H_k as $H(x_k)$, then the evaluation of (2.1) at every x_i yields the equation

$$H_i\left[1 - \frac{c}{2}\int_0^1 \frac{x_i H(y)}{x_i+y}dy\right] = 1, \quad i = 1,2,\ldots,n. \tag{2.3}$$

After multiplying both sides of (2.2) by $[1-(c/2)\int_0^1 (xH(y)/(x+y))dy]^{-1}$ and performing some algebra, we arrive at (2.4), which is known as the Chandrasekhar H-equation [15]

$$F(H)(x) = H(x) - \left(1 - \frac{c}{2}\int_0^1 \frac{xH(y)dy}{x+y}\right)^{-1} = 0. \tag{2.4}$$

If (2.4) is discretized by using the midpoint quadrature formula

$$\int_0^1 f(t)dt = \frac{1}{n}\sum_{j=0}^{n} f(t_j), \tag{2.5}$$

for $t_j = (j - 0.5)h$, $0 \leq j \leq 1$, $i = 2, \dots, n$, $h = 1/n$, $c \in (0, 1)$, then we have the following:

$$F_i = x_i - \left(1 - \frac{c}{2n}\sum_{j=1}^{n}\frac{t_i x_j}{t_i + t_j}\right)^{-1}. \tag{2.6}$$

Function (2.6) is called the discretized Chandrasekhar H-equation which can be solved by some iterative methods.

Nevertheless, the most difficult part in solving (2.6) arises dramatically as c approaches 1, since its Jacobian is singular at $c = 1$. Due to this disadvantage, we aim to derive a method that hopefully will not be affected by this difficulty.

3. Derivation of the Method (LMSI)

Firstly, note that by the mean value theorem, we have

$$\overline{F'}(x_k)(x_{k+1} - x_k) = F(x_{k+1}) - F(x_k), \tag{3.1}$$

where $\overline{F'}(x_k) = \int_0^1 F'(x_k + \theta(x_{k+1} - x_k))d\theta$.

Let us denote $\Delta x_k = x_{k+1} - x_k$ and $\Delta F_k = F(x_{k+1}) - F(x_k)$, then (3.1) becomes

$$\overline{F'}(x_k)\Delta x_k = \Delta F_k. \tag{3.2}$$

Equation (3.2) is always regarded as the secant equation. Alternatively, we can rearrange (3.2) to obtain

$$\Delta x_k = \left(\overline{F'}(x_k)\right)^{-1}\Delta F_k. \tag{3.3}$$

Here, we propose to use a diagonal matrix, say D, to approximate $(\overline{F'}(x_k))^{-1}$, that is,

$$\left(\overline{F'}(x_k)\right)^{-1} \approx D_k. \tag{3.4}$$

Let us consider an updating scheme for D, in which we should update D by adding a correction M which is also a diagonal matrix at every iteration

$$D_{k+1} = D_k + M_k. \tag{3.5}$$

In order to incorporate correct information of the Jacobian inverse into the updating matrix, D_{k+1}, we require that D_{k+1} satisfies the secant equation (3.2), that is,

$$\Delta x_k = (D_k + M_k)\Delta F_k. \tag{3.6}$$

However, since it is difficult to have a diagonal matrix that satisfies the secant equation, in particular, because Jacobian approximations are not usually done in element wise, we consider the use of the weak secant condition [17] instead,

$$\Delta F_k^T \Delta x_k = \Delta F_k^T (D_k + M_k)\Delta F_k. \tag{3.7}$$

To encourage good condition number as well as numerical stability in the approximation, we attempt to control the growth error of the correction by minimizing its magnitude under some norms (here, we consider the Frobenius norm), such that (3.7) holds. To this end, we consider the following problem:

$$\begin{aligned} &\min \frac{1}{2}\|D_{k+1} - D_k\|_F^2 \\ &\text{s.t.} \quad \Delta F_k^T D_{k+1} \Delta F_k = \Delta F_k^T \Delta x_k, \end{aligned} \tag{3.8}$$

where $\|\cdot\|_F$ is the Frobenius norm. If we let $D_{k+1} - D_k = M_k = \operatorname{diag}(\beta_1, \beta_2, \ldots, \beta_n)$ and $\Delta F_k = (\Delta F_k^{(1)}, \Delta F_k^{(2)}, \ldots, \Delta F_k^{(n)})$, the above problem can be expressed as follows:

$$\begin{aligned} &\min \frac{1}{2}\left(\beta_1^2 + \beta_2^2 + \cdots + \beta_n^2\right) \\ &\text{s.t.} \quad \sum_{i=1}^{n} \Delta F_k^{(i)^2} \beta_i - \Delta F_k^T \Delta x_k + \Delta F_k^T D_k \Delta F_k = 0. \end{aligned} \tag{3.9}$$

Since the objective function and the constraint are convex, we will have unique solution for (3.9). The solution can be obtained by considering the Lagrangian function of problem (3.9)

$$L(\beta_i, \lambda) = \frac{1}{2}\left(\beta_1^2 + \beta_2^2 + \cdots + \beta_n^2\right) + \lambda\left(\sum_{i=1}^{n} \Delta F_k^{(i)^2} \beta_i - \Delta F_k^T \Delta x_k + \Delta F_k^T D_k \Delta F_k\right), \tag{3.10}$$

where λ is the corresponding Lagrangian multiplier.

Taking the partial derivatives of (3.10) with respect to each β_i and λ, respectively, and setting them equal to zero, we have

$$\frac{\partial L}{\partial \beta_i} = \beta_i + \lambda\left(\Delta F_k^{(i)}\right)^2 = 0, \quad i = 0, 1, 2, \ldots, n, \tag{3.11}$$

$$\frac{\partial L}{\partial \lambda} = \sum_{i=1}^{n}\left(\Delta F_k^{(i)}\right)^2 \beta_i - \Delta F_k^T \Delta x_k + \Delta F_k^T D_k \Delta F_k = 0. \tag{3.12}$$

Premultiplying both sides of (3.11) by $\Delta F_k^{(i)^2}$ and summing them all yield

$$\sum_{i=1}^{n}\left(\Delta F_k^{(i)}\right)^2\beta_i+\lambda\sum_{i=1}^{n}\left(\Delta F_k^{(i)}\right)^4=0. \tag{3.13}$$

It follows from (3.13) that

$$\sum_{i=1}^{n}\left(\Delta F_k^{(i)}\right)^2\beta_i=-\lambda\sum_{i=1}^{n}\left(\Delta F_k^{(i)}\right)^4. \tag{3.14}$$

Invoking the constraint (3.12), we have

$$\sum_{i=1}^{n}\left(\Delta F_k^{(i)}\right)^2\beta_i=\Delta F_k^T\Delta x_k-\Delta F_k^T D_k\Delta F_k. \tag{3.15}$$

Equating (3.14) with (3.15) gives

$$\lambda=-\frac{\Delta F_k^T\Delta x_k-\Delta F_k^T D_k\Delta F_k}{\sum_{i=1}^{n}\left(\Delta F_k^{(i)}\right)^4}. \tag{3.16}$$

Substituting (3.16) into (3.14) and after some simplifications, we obtain

$$\beta_i=\frac{\left(\Delta F_k^T\Delta x_k-\Delta F_k^T D_k\Delta F_k\right)}{\sum_{i=1}^{n}\left(\Delta F_k^{(i)}\right)^4}\left(\Delta F_k^{(i)}\right)^2,\quad i=1,2,\dots,n. \tag{3.17}$$

Denoting $G_k=\operatorname{diag}((\Delta F_k^{(1)})^2,(\Delta F_k^{(2)})^2,\dots,(\Delta F_k^{(n)})^2)$ and $\sum_{i=1}^{n}(\Delta F_k^{(i)})^4=\operatorname{Tr}(G_k^2)$ where Tr is the trace operation, we obtain, therefore,

$$M_k=\frac{\left(\Delta F_k^T\Delta x_k-\Delta F_k^T D_k\Delta F_k\right)}{\operatorname{Tr}\left(G_k^2\right)}G_k. \tag{3.18}$$

Finally, the proposed updating formula for the approximation of the Jacobian inverse is given as follows:

$$D_{k+1}=D_k+\frac{\left(\Delta F_k^T\Delta x_k-\Delta F_k^T D_k\Delta F_k\right)}{\operatorname{Tr}\left(G_k^2\right)}G_k. \tag{3.19}$$

To safeguard possibly very small ΔF_k and $\operatorname{Tr}(G_k^2)$, we require that $\|\Delta F_k\|\geq\epsilon_1$ for some chosen small $\epsilon_1>0$. Else, we will skip the update by setting $D_{k+1}=D_k$.

Now, we can describe the algorithm for our proposed method (LMSI) as follows.

Algorithm LMSI

Steps are the following.

Step 1. Given x_0 and D_0, set $k = 0$.

Step 2. Compute $F(x_k)$ and $x_{k+1} = x_k - D_k F(x_k)$.

Step 3. If $\|\Delta x_k\|_2 + \|F(x_k)\|_2 \leq 10^{-4}$, stop. Else, go to Step 4.

Step 4. If $\|\Delta F_k\|_2 \geq \epsilon_1$ where $\epsilon_1 = 10^{-4}$, compute D_{k+1}, if not, $D_{k+1} = D_k$. Set $k := k + 1$ and go to Step 2.

4. Local Convergence Results

In this section, we will give some convergence properties of LMSI method. Before we proceed further, we will make the following standard assumptions on nonlinear systems F.

Assumption 4.1. We have the following.

(i) F is differentiable in an open-convex set E in $\Re^n$.

(ii) There exists $x^* \in E$ such that $F(x^*) = 0$, and $F'(x)$ is continuous for all x.

(iii) $F'(x)$ satisfies Lipschitz condition of order one, that is, there exists a positive constant μ such that

$$\|F'(x) - F'(y)\| \leq \mu\|x - y\|, \tag{4.1}$$

for all $x, y \in \Re^n$.

(iv) There exists constants $c_1 \leq c_2$ such that $c_1\|\omega\|^2 \leq \omega^T F'(x)\omega \leq c_2\|\omega\|^2$ for all $x \in E$ and $\omega \in \Re^n$.

We will also need the following result which is a special case of a more general theorem of [15].

Theorem 4.2. *Assume that Assumption 4.1 holds. If there exists $K_B > 0$, $\delta > 0$, and $\delta_1 > 0$, such that for $x_0 \in B(\delta)$ and the matrix-valued function $B(x)$ satisfies $\|I - B(x)F'(x^*)\| = \rho(x) < \delta_1$ for all $x \in B(\delta)$, then the iteration*

$$x_{k+1} = x_k - B(x_k)F(x_k) \tag{4.2}$$

converges linearly to x^*.

For the proof of Theorem 4.2, see [15].

Using Assumption 4.1 and Theorem 4.2, one has the following result.

Theorem 4.3. *Assume that Assumption 4.1 holds. There exist $\beta > 0$, $\delta > 0$, $\alpha > 0$, and $\gamma > 0$, such that if $x_0 \in E$ and D_0 satisfies $\|I - D_0 F'(x^*)\|_F < \delta$ for all $x_k \in E$, then for iteration*

$$x_{k+1} = x_k - D_k F(x_k), \tag{4.3}$$

D_k defined by (3.19),

$$\left\| I - D_k F'(x^*) \right\|_F < \delta_k, \tag{4.4}$$

holds for some constant $\delta_k > 0$, $k \geq 0$.

Proof. Since $\|D_{k+1}\|_F = \|D_k + M_k\|_F$, it follows that

$$\|D_{k+1}\|_F \leq \|D_k\|_F + \|M_k\|_F. \tag{4.5}$$

For $k = 0$ and assuming $D_0 = I$, we have

$$\left| M_0^{(i)} \right| = \left| \frac{\Delta F_0^T \Delta x_0 - \Delta F_0^T D_0 \Delta F_0}{\mathrm{Tr}(G^2)} \Delta F_0^{(i)^2} \right| \leq \frac{\left| \Delta F_0^T \Delta x_0 - \Delta F_0^T D_0 \Delta F_0 \right|}{\mathrm{Tr}\left(G_0^2\right)} \Delta F_0^{(\max)^2}, \tag{4.6}$$

where $(\Delta F_0^{(\max)})^2$ is the largest element among $(\Delta F_0^{(i)})$, $i = 1, 2, \ldots, n$. After multiplying (4.6) by $(\Delta F_0^{(\max)})^2 / (\Delta F_0^{(\max)})^2$ and substituting $\mathrm{Tr}(G_0^2) = \sum_{i=1}^n (\Delta F_0^{(i)})^4$, we have

$$\left| M_0^{(i)} \right| \leq \frac{\left| \Delta F_0^T \Delta x_0 - \Delta F_0^T D_0 \Delta F_0 \right|}{\left(\Delta F_0^{(\max)}\right)^2 \sum_{i=1}^n \left(\Delta F^{(i)}\right)^4} \left(\Delta F_0^{(\max)}\right)^4. \tag{4.7}$$

Since $(\Delta F_0^{(\max)})^4 / \sum_{i=1}^n (\Delta F_0^{(i)})^4 \leq 1$, then (4.7) turns into

$$\left| M_0^{(i)} \right| \leq \frac{\left| \Delta F_0^T F'(x) \Delta F_0 - \Delta F_0^T D_0 \Delta F_0 \right|}{\left(\Delta F_0^{(\max)}\right)^2}. \tag{4.8}$$

From Assumption 4.1(iv) and $D_0 = I$, (4.8) becomes

$$\left| M_0^{(i)} \right| \leq \frac{|c-1| \left(\Delta F_0^T \Delta F_0\right)}{\left(\Delta F_0^{(\max)}\right)^2}, \tag{4.9}$$

where $c = \max\{|c_1|, |c_2|\}$.

Since $\Delta(F_0^{(i)})^2 \leq (\Delta F_0^{(\max)})^2$ for $i = 1, \ldots, n$, it follows that

$$\left| M_0^{(i)} \right| \leq \frac{|c-1| n \left(\Delta F_0^{(\max)}\right)^2}{\left(\Delta F_0^{(\max)}\right)^2}. \tag{4.10}$$

Hence, we obtain

$$\|M_0\|_F \leq n^{3/2} |c-1|. \tag{4.11}$$

Table 1: Results of Chandrasekhar H-equation (number of iteration/CPU time).

	n	NM	FN	BM	LMSI
	200	4/6.9134	7/5.1673	6/4.7112	5/0.0312
	500	4/14.0945	7/11.0418	8/6.9813	5/0.0624
	1000	4/64.5813	9/27.2974	8/9.7416	5/0.1716
$c = 0.9$	2000	4/95.6180	9/118.2094	8/11.1690	5/1.6410
	5000	—	—	—	5/2.3556
	10000	—	—	—	5/2.8209
	20000	—	—	—	5/3.7915
	200	6/8.1834	8/6.4282	7/5.1946	6/0.0780
	500	6/18.2457	8/14.6132	7/7.6137	6/0.0824
	1000	6/83.0569	9/30.0542	8/11.0432	6/0.2184
$c = 0.99$	2000	6/121.5309	9/153.0351	8/15.9724	4/1.3104
	5000	—	—	—	4/1.4976
	10000	—	—	—	5/1.7910
	20000	—	—	—	5/2.0371
	200	8/15.1792	—	9/5.0388	5/0.3312
	500	8/36.0330	—	10/7.1412	5/0.6024
	1000	8/134.9045	—	10/12.0644	6/0.7803
$c = 0.9999$	2000	8/175.0521	—	10/14.9064	6/1.2537
	5000	—	—	—	6/1.5132
	10000	—	—	—	6/1.5808
	20000	—	—	—	6/1.6301

Suppose that $\alpha = n^{3/2}|c-1|$, then

$$\|M_0\|_F \leq \alpha. \tag{4.12}$$

From the fact that $\|D_0\|_F = \sqrt{n}$, it follows that

$$\|D_1\|_F \leq \beta, \tag{4.13}$$

where $\beta = \sqrt{n} + \alpha > 0$.

Therefore, if we assume that $\|I - D_0 F'(x^*)\|_F < \delta$, then

$$\begin{aligned}\left\|I - D_1 F'(x^*)\right\|_F &= \left\|I - (D_0 + M_0) F'(x^*)\right\|_F \\ &\leq \left\|I - D_0 F'(x^*)\right\|_F + \left\|M_0 F'(x^*)\right\|_F \\ &\leq \left\|I - D_0 F'(x^*)\right\|_F + \|M_0\|_F \left\|F'(x^*)\right\|_F,\end{aligned} \tag{4.14}$$

hence $\|I - D_1 F'(x^*)\|_F < \delta + \alpha\phi = \delta_1$.

And hence, by induction, $\|I - D_k F'(x^*)\|_F < \delta_k$ for all k. □

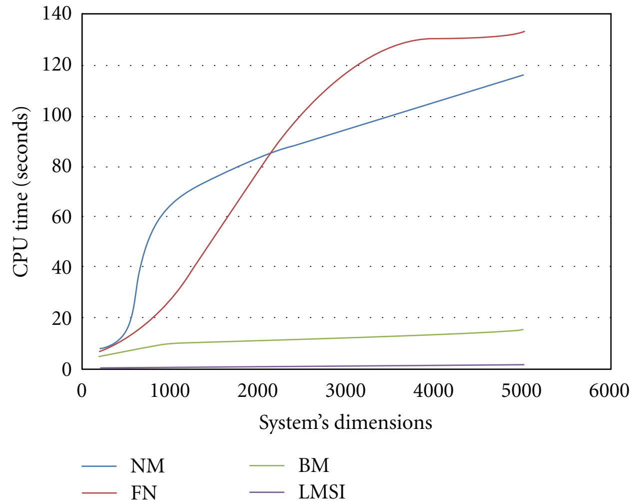

Figure 1: Comparison of NM, FN, BM, and LMSI methods when $c = 0.9$, in terms of CPU time.

5. Numerical Results

In this section, we compare the performance of LMSI method with that of the Newton's method (NM), fixed Newton's method (FN), and Broyden's method (BM). We apply the algorithms to the well-known benchmarks integral equations involved in radiative transfer. The comparison is based upon the following criterion: number of iterations, CPU time in seconds, and storage requirement. The computations are done in MATLAB 7.0 using double-precision computer. The stopping criterion used is

$$\|\Delta x_k\| + \|F(x_k)\| \leq 10^{-4}. \tag{5.1}$$

The starting point x_0 is given by $(1, 1, \ldots 1)^T$.

The symbol "–" is used to indicate a failure due to the following:

(1) The number of iteration is at least 200, but no point of x_k satisfying (5.1) is obtained,

(2) CPU time in second reaches 200,

(3) insufficient memory to initial the run.

The numerical results of the methods when solving Chandrasekhar H-Equation in different parameter are reported in Table 1. The first column of the table contains the parameter of problem. Generally, with our choice of c, the corresponding Jacobian is not diagonally dominate; however, when $c \to 1$, the Jacobian is nearly singular. From Table 1, it was shown that only LMSI is able to solve problems where $n > 2000$. This is due to the fact that LMSI requires very low-storage requirement in building the approximation of the Jacobian inverse. Indeed, the size of the updating matrix increases in $O(n)$ as the dimension of the system increases, as opposed to NM, FN, and BM methods that increase in $O(n^2)$.

Moreover, we observe that LMSI method has a 100% of success rate (convergence to the solution) when compared with NM method having 57%, FN method with 39% and BM with 71%, respectively. In addition, it is worth mentioning that the result of LMSI in solving

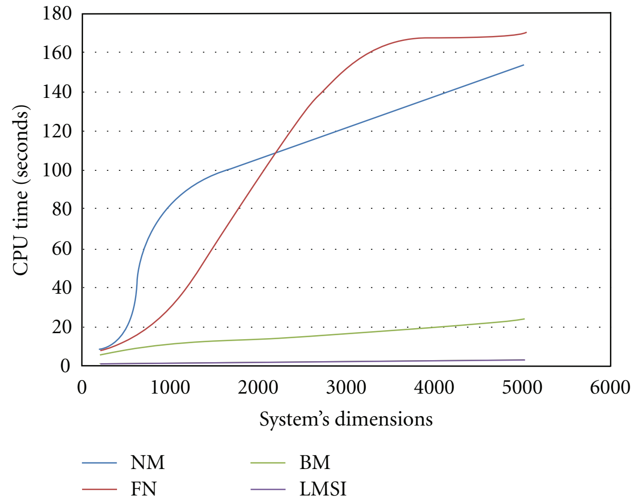

Figure 2: Comparison of NM, FN, BM, and LMSI methods when $c = 0.99$, in terms of CPU time.

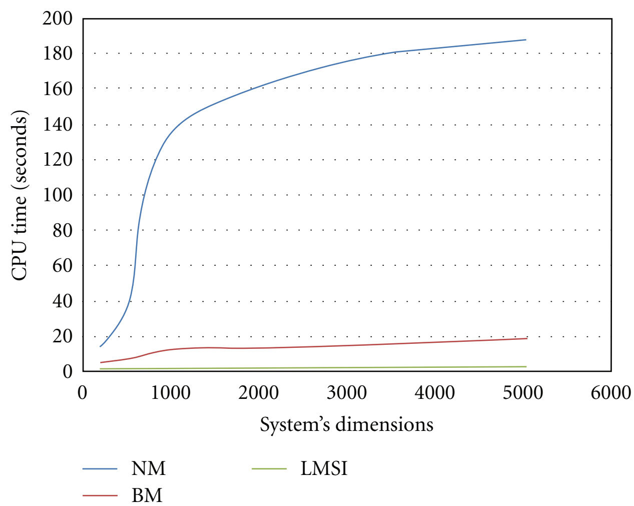

Figure 3: Comparison of NM, FN, BM, and LMSI methods when $c = 0.9999$, in terms of CPU time.

problem 1 when $c = 0.9999$ shows that the method could be a good solver even when the Jacobian is nearly singular. Figures 1, 2, and 3 reveal that the CPU time of LMSI method increases linearly as the dimension of the systems increases, whereas for NM, FN, and BM, the rates grow exponentially. This also suggests that our solver is a good alternative when the dimension of the problem is very high.

6. Conclusion

In this paper, we present a low memory solver for integral equation of Chandrasekhar type in the radiative transfer problems. Our approach is based on approximating the Jacobian inverse into a diagonal matrix. The fact that the LMSI method can solve the discretized integral equations without computing and storing the Jacobian makes clear the advantage over NM and FN methods. It is also worth mentioning that the method is capable of significantly reducing the execution time (CPU time), as compared to NM, FN, and BM methods while maintaining good accuracy of the numerical solution to some extend. Another fact that makes the LMSI method appealing is that throughout the numerical experiments it never fails to converge. Finally, we conclude that our method (LMSI) is a good alternative to Newton-type methods for solving large-scale nonlinear equations with nearly singular Jacobian.

References

[1] S. Chandrasekhar, *Radiative Transfer*, Oxford University Press, Londo, UK, 1950.

[2] J. Caballero, A. B. Mingrarelli, and K. Sadarangani, "Existence of solution of an integral equation of Chandrasekhar type in the radiative transfer," *Electronic Journal of Differential Equations*, vol. 27, pp. 1–11, 2006.

[3] G. A. Hively, "On a class of nonlinear integral equations arising in transport theory," *SIAM Journal on Mathematical Analysis*, vol. 9, no. 5, pp. 787–792, 1978.

[4] W. Kalkofen, *Numerical Radiative Transfer*, Cambridge University Press, Cambridge, UK, 1987.

[5] I. W. Busbridge, "On solutions of Chandrasekhar's integral equation," *Transactions of the American Mathematical Society*, vol. 105, pp. 112–117, 1962.

[6] R. W. Leggett, "A new approach to the *H*-equation of Chandrasekhar," *SIAM Journal on Mathematical Analysis*, vol. 7, no. 4, pp. 542–550, 1976.

[7] C. T. Kelley and J. I. Northrup, "A pointwise quasi-Newton method for integral equations," *SIAM Journal on Numerical Analysis*, vol. 25, no. 5, pp. 1138–1155, 1988.

[8] M. O. James and C. R. Werner, "On discretization and differentiation of operators with application to Newton's method," *SIAM Journal on Numerical Analysis*, vol. 3, no. 1, pp. 143–156, 1966.

[9] J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice Hall Series in Computational Mathematics, Prentice Hall , Englewood Cliffs, NJ, USA, 1983.

[10] K. Natasa and L. Zorna, "Newton-like method with modification of the right-hand vector," *Mathematics of Computation*, vol. 71, pp. 237–250, 2001.

[11] C. G. Broyden, "A class of methods for solving nonlinear simultaneous equations," *Mathematics of Computation*, vol. 19, pp. 577–593, 1965.

[12] M. Farid, W. J. Leong, and M. A. Hassan, "A new two-step gradient-type method for large-scale unconstrained optimization," *Computers & Mathematics with Applications*, vol. 59, no. 10, pp. 3301–3307, 2010.

[13] M. A. Hassan, W. J. Leong, and M. Farid, "A new gradient method via quasi-Cauchy relation which guarantees descent," *Journal of Computational and Applied Mathematics*, vol. 230, no. 1, pp. 300–305, 2009.

[14] C. G. Broyden, "Quasi-Newton methods and their application to function minimisation," *Mathematics of Computation*, vol. 21, pp. 368–381, 1967.

[15] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, vol. 16 of *Frontiers in Applied Mathematics*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, Pa, USA, 1995.

[16] S. Chandrasekhar and F. H. Breen, "On the radiative equilibrium of a stellar atmosphere. XIX," *The Astrophysical Journal*, vol. 106, pp. 143–144, 1947.

[17] J. E. Dennis and H. Wolkowicz, "Sizing and least-change secant methods," *SIAM Journal on Numerical Analysis*, vol. 30, no. 5, pp. 1291–1314, 1993.