A&A 470, 1185–1191 (2007)
DOI: 10.1051/0004-6361:20065231
© ESO 2007

Astronomy & Astrophysics

The GMRES method applied to the BEM extrapolation of solar force-free magnetic fields

I. Constant α force-free field

Y. Li[1,2,5], Y. Yan[2], M. Devel[3], R. Langlet[4], and G. Song[1]

[1] School of Science, Xidian University, Xi'an, Shaanxi, PR China
e-mail: ywli27@163.com
[2] National Astronomical Observatories, Chinese Academy of Sciences, Beijing, PR China
e-mail: yyh@bao.ac.cn
[3] Institut UTINAM, UMR CNRS 6213, 16 route de GRAY, 25030 Besançon Cedex, France
e-mail: michel.devel@univ-fcomte.fr
[4] Laboratoire de Physique du Solide, FUNDP, Rue de Bruxelles 61, 5000 Namur, Belgium
e-mail: rachel.langlet@fundp.ac.be
[5] School of Applied Science, Taiyuan University of Science and Technology, Taiyuan, Shanxi, PR China

Received 18 March 2006 / Accepted 17 April 2007

ABSTRACT

Context. Since the 1990's, Yan and colleagues have formulated a kind of boundary integral formulation for the linear or non-linear solar force-free magnetic fields with finite energy in semi-infinite space, and developed a computational procedure by virtue of the boundary element method (BEM) to extrapolate the magnetic fields above the photosphere.

Aims. In this paper, the generalized minimal residual method (GMRES) is introduced into the BEM extrapolation of the solar force-free magnetic fields, in order to efficiently solve the associated BEM system of linear equations, which was previously solved by the Gauss elimination method with full pivoting.

Methods. Being a modern iterative method for non-symmetric linear systems, the GMRES method reduces the computational cost for the BEM system from $O(N^3)$ to $O(N^2)$, where N is the number of unknowns in the linear system. Intensive numerical experiments are conducted on a well-known analytical model of the force-free magnetic field to reveal the convergence behaviour of the GMRES method subjected to the BEM systems. The impacts of the relevant parameters on the convergence speed are investigated in detail.

Results. Taking the Krylov dimension to be 50 and the relative residual bound to be 10^{-6} (or 10^{-2}), the GMRES method is at least 1000 (or 9000) times faster than the full pivoting Gauss elimination method used in the original BEM extrapolation code, when N is greater than 12 321, according to the CPU timing information measured on a common desktop computer (CPU speed 2.8 GHz; RAM 1 GB) for the model problem.

Key words. Sun: magnetic fields – Sun: corona – methods: numerical

1. Introduction

Magnetic fields play an important role in many physical events occurring in the solar atmosphere. So far, reliable magnetic field measurements have been realized at the photosphere or chromosphere levels, which is however not the case for the coronal observations (Yan 2003; Valori et al. 2005). For many years, various extrapolation methods based on the force-free magnetic field assumption have been developed as a major tool to reconstruct the structures and configurations of the magnetic fields in the corona, using the magnetogram data as the boundary condition (see, e.g., Schmidt 1964; Altschuler & Newkirk 1969; Nakagawa et al. 1973; Chiu & Hilton 1977; Seehafer 1978; Alissandrakis 1981; Sakurai 1981; Wu et al. 1985, 1990; Yan 1995; Amari et al. 1998, 1999, 2006; Yan & Sakurai 1997, 2000; Wheatland 1999, 2004; Valori et al. 2005; Yan & Li 2006). The force-free magnetic field satisfies (See, e.g., Amari et al. 1998)

$$\nabla \times \boldsymbol{B} = \alpha(\boldsymbol{r})\boldsymbol{B} \quad \text{in} \quad \Omega, \tag{1}$$

$$\nabla \cdot \boldsymbol{B} = 0 \quad \text{in} \quad \Omega, \tag{2}$$

where both $\alpha(\boldsymbol{r})$ and $\boldsymbol{B}$ are unknowns in general. Although the force-free magnetic field equations take a disarmingly simple form, the nature of their solutions has not yet been understood (Priest 1994). Nevertheless, significant progress has been made in tackling this problem during the past half century (see review papers by, e.g., Aly 1989; Gary 1989; Sakurai 1989; McClymont et al. 1997; Wang 1999; Yan 2003; Aschwanden 2004; see also the brief review on the linear force-free field in Démoulin & Priest 1992; Schmieder & Aulanier 2003; see Seehafer 1978 and Aly 1992, for some important properties of the linear force-free field associated to the different geometry of the photosphere). Besides theoretical approaches, computational methods play an important role among these studies.

Since the 1990's, Yan and colleagues have formulated a kind of boundary integral representations for the linear or non-linear solar force-free magnetic field with finite energy content in the semi-infinite space above the Sun, and applied a well established numerical method – boundary element method (BEM) – to compute the magnetic field above the active regions on the Sun (Yan et al. 1991; Yan 1995; Yan & Sakurai 1997, 2000). The BEM extrapolation has found wide applications in the study of the

Article published by EDP Sciences and available at http://www.aanda.org or http://dx.doi.org/10.1051/0004-6361:20065231

coronal magnetic fields since its advent (see, e.g., Yan & Wang 1995; Yan et al. 1995, 2001a,b; Wang et al. 2001, 2002; Liu et al. 2002; Zhang & Wang 2002).

In the BEM extrapolation, however, one needs to compute the normal derivative of the magnetic field at the boundary region as an intermediate step, which amounts to solving a non-symmetric, dense linear system of equations. In the original BEM extrapolation code, a Gauss solver with full pivoting, found in Liu et al. (1980), was used for the solution of the BEM system, which is a fairly time-consuming process for a large scale problem. This well known difficulty has made the routine use of the BEM extrapolation code prohibitive on common computers. Although the difficulty can be reduced by exploiting a supercomputer, it is more desirable to seek faster substitutive algorithms in the long run.

In this paper, we use the generalized minimal residual method (usually called GMRES method), originally presented by Saad & Schult (1986), to efficiently solve the linear systems arising from the BEM extrapolation. Compared with other iterative algorithms for non-symmetric linear systems such as the generalized conjugate residual method, the GMRES method is more economical in memory requirements and arithmetic operations, and can terminate in at most N steps (where N is the number of unknowns of the linear system under consideration), which is desirable for large scale problems (Huang & Vorst 1989, 1990). There are some practical variants of the original GMRES algorithm, among which the *restarted* GMRES algorithm, called GMRES(m) in short, and the *incomplete* GMRES algorithm are the most popular ones (Saad & Wu 1996; Quarteroni & Valli 1997). Recent applications of the GMRES method and its relatives alike can be found, e.g., in Patton & Holloway (2002); Ravindra & Muddun (2004), Spyropoulos et al. (2004) and Ronald (2005). Like other iterative methods for linear systems, the efficiency of the GMRES method is dependent on the nature of the coefficient matrix of the linear system under consideration. Fortunately, the GMRES solver we used converges very fast to the solution of our problem with a desired accuracy. This is understandable since the linear systems arising from integral equations are usually well-conditioned.

The paper is divided as follows. In Sect. 2, we briefly recall the boundary integral representation of the constant α force-free magnetic field proposed by Yan (1995), and make some notation conventions. In Sect. 3, we introduce the main ideas of the GMRES method while the numerical results are reported in Sect. 4. Finally, we discuss some important points and draw conclusions in Sect. 5.

2. The boundary integral representation of the constant α force-free magnetic field

We recall the boundary integral representation of the constant α force-free magnetic field proposed by Yan (1995) as follows. Let Ω be the outer space above the Sun, Γ the photosphere surface, and Γ_D the magnetogram area, i.e. $\Omega = \{(x,y,z)|z > 0\}$, $\Gamma = \{(x,y,z)|z = 0\}$, and $\Gamma_D = \{(x,y,z)|z = 0, a \le x \le b, c \le y \le d\}$. The governing Eqs. (1) and (2) together with the boundary condition

$$\boldsymbol{B}(x,y,z) = \begin{cases} \boldsymbol{B}_0, & (x,y,z) \in \Gamma_D, \\ \boldsymbol{0}, & (x,y,z) \in \Gamma_\infty := \Gamma - \Gamma_D, \end{cases} \tag{3}$$

and the asymptotic condition

$$\boldsymbol{B} = O(r^{-2}), r \to +\infty, \tag{4}$$

form an exterior boundary value problem, where $\boldsymbol{B}_0$ can be obtained from the observed vector magnetograms while its three components are assumed to satisfy the governing equations consistently. Note that, $\boldsymbol{B}(x,y,z)$ has been artificially imposed to be $\boldsymbol{0}$ on Γ_∞, beginning from Schmidt (1964), due to the realistic restriction of the observational instruments. The asymptotic condition is applied to make the solution unique and have a finite energy content.

By applying the Green's second identity to the above exterior boundary value problem, one can obtain an integral representation of the constant α force-free magnetic field as follows:

$$c_i\boldsymbol{B}_i = \int\limits_\Gamma \left(Y\frac{\partial \boldsymbol{B}}{\partial n} - \boldsymbol{B}\frac{\partial Y}{\partial n} \right) \mathrm{d}\Gamma, \tag{5}$$

where c_i is a constant determined by the location of the point i, of coordinates (x',y',z'), i.e., $c_i = 1/2$ for $(x',y',z') \in \Gamma$ or $c_i = 1$ for $(x',y',z') \in \Omega$; Y is the fundamental solution of the Helmholtz's equation, chosen as

$$Y := Y(x,y,z;x',y',z') = \frac{\cos(\alpha r)}{4\pi r}, \tag{6}$$

where α is the force-free parameter and $r = \sqrt{(x-x')^2 + (y-y')^2 + (z-z')^2}$.

One may observe from (5) that the value of $\boldsymbol{B}$ at any point $i \in \Omega$ can be determined by the boundary value of $\boldsymbol{B}$ and Y together with the boundary value of their normal derivatives. However, the normal derivative of the magnetic field at the boundary, i.e. $\frac{\partial \boldsymbol{B}(x,y,z)}{\partial n}|_{z=0}$, is unknown.

To evaluate the normal derivative of the magnetic field at the boundary, we let $i \in \Gamma$. Then, one can verify that $\frac{\partial Y}{\partial n} = 0$ always holds at the boundary, which makes (5) take the form of

$$\frac{1}{2}\boldsymbol{B}_i = \iint\limits_{\Gamma_D} Y(x,y,0;x',y',0)\frac{\partial \boldsymbol{B}(x,y,0)}{\partial n}\,\mathrm{d}x\,\mathrm{d}y, \tag{7}$$

where values of $\boldsymbol{B}_i$ and $Y(x,y,0;x',y',0)$ are known, and $\frac{\partial \boldsymbol{B}(x,y,0)}{\partial n}$ is an unknown function of two variables; (x',y') is the coordinate of the point i on Γ_D. In Yan (1995), Eq. (7) is discretized by the boundary element method, which amounts to solving a dense linear system with three right-hand sides (called 3D right-hand side hereafter), whose solutions provide an approximation of $\frac{\partial \boldsymbol{B}(x,y,0)}{\partial n}$ on Γ_D. Then, with the computed $\frac{\partial \boldsymbol{B}(x,y,0)}{\partial n}$ and the boundary condition (3), we compute the magnetic field over Γ_D with Eq. (5). The details of the boundary element method can be found in Yan (1995) and the reference therein.

3. The GMRES method

Most of the iterative methods for linear equations are subjected to a general framework of *projection methods*. Here, we summarize the basic ideas of the GMRES method from this point of view. The details of the method can be found in (Saad & Schultz 1986; Saad 1996).

Consider the linear equations

$$\boldsymbol{Ax} = \boldsymbol{b}, \tag{8}$$

where $\boldsymbol{A}$ is an $N \times N$ real matrix. The idea of projection methods is to extract an approximate solution to the above problem from a subspace of R^N, called projection subspace. In particular, the Krylov subspace $K_l := K_l(\boldsymbol{A}, \boldsymbol{r}_0) := span\{r_0, Ar_0, A^2r_0, \ldots, A^{l-1}r_0\}$ can be taken as the projection subspace, where $r_0 := b - Ax_0$ is the initial residual while x_0 is an initial guess solution. The GMRES method is a kind of Krylov subspace method. Two problems are involved in the projection methods: one is how to compute the basis of the projection space and another is how to obtain the approximate solution. In the GMRES method, Arnoldi's iteration algorithm (Arnoldi 1951) is used to obtain an orthonormal basis of the Krylov projection subspace. In addition, the following relation can be derived from Arnoldi's algorithm:

$$AV_l = V_{l+1}\overline{H}_l, \tag{9}$$

where V_l is the matrix formed by the orthonormal basis of the Krylov subspace K_l and $\overline{H}_l$ is a Hessenberg matrix of order $(l+1) \times l$ (Saad & Schultz 1986). The approximate solution $\tilde{x}$ is then sought in $x_0 + K_l$, such that the corresponding residual is minimized. More precisely, one can express the approximate solution $\tilde{x}$ in $x_0 + K_l$ as

$$\tilde{x} = x_0 + V_l\tilde{y}, \tag{10}$$

where $\tilde{y}$ is an l dimensional vector to be determined. By virtue of (9) and (10), the residual can be expressed as

$$b - A\tilde{x} = V_{l+1}(\|r_0\|e_1 - \overline{H}_l\tilde{y}), \tag{11}$$

where e_1 denotes the first column of the unit matrix, and r_0 is the initial residual. As V_{l+1} is orthonormal, the norm of the residual simply is:

$$\|\beta e_1 - \overline{H}_l\tilde{y}\|, \ (\beta := \|r_0\|). \tag{12}$$

Thus, the components of $\tilde{y}$ can be determined by minimizing $\|\beta e_1 - \overline{H}_l\tilde{y}\|$.

In the GMRES algorithm, one needs to store the matrices V_l and $\overline{H}_l$ in the memory. Since the Krylov dimension l keeps increasing with iterations until an approximate solution with a prescribed accuracy is obtained (or until reaching N), the GMRES algorithm may encounter storage issues for large scale problems, in case the algorithm can not satisfy the accuracy criterion in a small number of iterations for the specific problem. To overcome this difficulty, a variant algorithm called GMRES(m), also proposed by Saad & Schultz (1986), is widely used in practical applications. Here, the Krylov subspace dimension m is a fixed number that should be properly determined to obtain a good performance (the algorithm is restarted every m steps, hence the name). Numerical results for $m = 2, 3, 4, 5, 20$ have been reported in Saad & Schultz (1986) on a pure mathematical test problem, but usually no theoretical conclusion is available on the optimal choice of m for a specific problem at hand (Barrett et al. 1994). For this reason, experimental efforts for determining a proper m for the given problem are in principle needed (see, e.g., Spyropoulos et al. 2004).

The GMRES algorithm and a few of its variants, together with some other popular Krylov-type iterative solvers, have been implemented in several well-known packages, such as SPARSKIT (Saad 1994), Templates (Barrett et al. 1994), and Aztec (Tuminaro et al. 1999). We have adapted most of the SPARSKIT package to our recently rebuilt BEM extrapolation code. Considerable efforts have been made in our new code to hide as many technical details as possible from the users, so they just need to configure a few external parameter files for their own problems, escaping from the potential dangers of modifying a huge source code or providing inconsistent inputs.

4. Numerical experiments

In this section, we investigate the asymptotic convergence behaviour of the GMRES(m) solver in the context of the BEM extrapolation of the solar force-free magnetic fields. In particular, we study the impact of some relevant parameters on the convergence speed of the GMRES(m) solver. For a given system of linear equations, this convergence speed is mainly determined by the dimension m of the Krylov subspace and the stop criterion. A common stop criterion for iterative solvers is to verify whether the relative residual error $\|b - A\tilde{x}\|/\|b\|$ is less than a prescribed positive number ε, called *relative residual bound* hereafter, which at the same time controls the accuracy of the solution. Therefore, for a given test problem, we sought a near optimal configuration of these two parameters such that the amount of computation is minimized, while keeping the accuracy of the solution satisfactory in the context of the field extrapolation.

4.1. Setting up the test problem

Our numerical experiments are conducted by using the well-known analytical force-free magnetic field constructed by Low (1982), which takes a form of:

$$B_x = -\frac{B_0 a}{r}\cos\varphi(r), \tag{13}$$

$$B_y = \frac{B_0 axy}{r[y^2 + (z+a)^2]}\cos\varphi(r) - \frac{B_0 a(z+a)}{y^2 + (z+a)^2}\sin\varphi(r), \tag{14}$$

$$B_z = \frac{B_0 ax(z+a)}{r[y^2 + (z+a)^2]}\cos\varphi(r) + \frac{B_0 ay}{y^2 + (z+a)^2}\sin\varphi(r), \tag{15}$$

where $r = \sqrt{x^2 + y^2 + (z+a)^2}$; $\varphi(r)$ is a free function called generating function; B_0 and $a(>0)$ are constants. The force-free parameter α of this analytical force-free magnetic field satisfies

$$\alpha = \frac{\mathrm{d}\varphi(r)}{\mathrm{d}r}\cdot \tag{16}$$

In particular, we take a generating function $\varphi(r)$ of the form (cf. Low 1982)

$$\varphi(r) = \varphi_0 + (\varphi_1 - \varphi_0)(r - a)/(r_0 - a), \tag{17}$$

where φ_0, φ_1 and r_0 are constants. Unlike Low's configuration of the function $\varphi(r)$, which generates a ball of constant α force-free magnetic field centred at $(0, 0, -a)$ with radius r_0, surrounded by a potential field for the purpose there, our configuration of the function $\varphi(r)$ generates a constant α force-free magnetic field in the whole space, which is more suitable for the purpose of this paper (see also Yan et al. 1993; Amari et al. 1998). Specifically, we set $\varphi_0 = \pi/4$, $\varphi_1 = \pi/2$, $a = 1.0$, $B_0 = 1000$, and $r_0 = 9.2736$. These parameters are taken from Low (1982) and Yan et al. (1993), except for r_0, which is chosen to let α small. Here, we have $\alpha = (\varphi_1 - \varphi_0)/(r_0 - a) \approx 0.0949$. We note that small α is not an intrinsic requirement of the BEM extrapolation, but a condition to maintain a faster decay of the analytical solution (see Amari et al. 1998, 1999, for discussions on the problems of slow decay of the analytical solution when simulating a finite magnetogram).

Now, we take a square region $\Gamma_D = \{(x, y) : |x| \le L_x, |y| \le L_y\}$ with $L_x = L_y = 5$ on the XOY plane, acting as the magnetogram region, on which the "magnetogram data" is supplied by the analytical force-free magnetic field. The scheme is first to discretize the region with a uniform grid, then the BEM method is applied

Table 1. CPU timing information for assembling the BEM matrices, for solving the linear systems of equations with the Gauss solver, the LU solver, or the GMRES(50) solver with various settings of ε, and for extrapolating the fields for 15 layers (CPU time in seconds).

Degrees of freedom (N)	1023	1443	1935	2499	3135	3843	4623	5475	6399	7395
Gauss	12.53	34.64	163.67	850.44	2524.63	5938.27	12 783.19	21 886.44	38 032.91	61 239.19
LU	0.86	3.66	10.34	23.22	46.44	85.94	149.66	248.36	395.78	611.64
GMRES(50, 10^{-16})	2.97	6.39	12.48	22.95	37.41	59.41	88.83	137.97	191.08	271.03
GMRES(50, 10^{-6})	0.89	1.86	3.50	6.06	10.08	16.06	24.41	35.44	49.54	69.69
GMRES(50, 10^{-2})	0.19	0.41	0.73	1.23	1.84	2.80	4.03	5.66	7.70	10.75
Assembling	0.88	1.78	3.27	5.50	9.33	15.48	22.88	32.39	44.55	59.14
Extrapolation	15.72	31.59	56.81	95.47	151.05	227.31	330.59	478.05	644.13	863.23

to compute the boundary normal derivative of the magnetic field on the grid, and finally the extrapolation is conducted on several levels of height, with a vertical step $z_h = 0.2$. In the intermediate step of computing the boundary normal derivative by the BEM, one normally just needs to discretize the region only once with a properly determined resolution, resulting in a linear system with a 3D right-hand side. Here, in order to investigate the asymptotic convergence behaviour of the GMRES(m) solver, we take 10 grids with higher and higher resolution, i.e. grids defined by $G_k = \{(x_i, y_j) : x_i = -L_x + 2L_x(i-1)/(m_k-1), y_j = -L_y + 2L_y(j-1)/(n_k-1); i = 1, \ldots, m_k; j = 1, \ldots, n_k\}$, where $m_k = 33 + 6(k-1)$, $n_k = m_k - 2$, $k = 1, 2, \ldots, 10$. Thus, we obtain 10 linear systems with 3D right-hand sides, whose sizes are $N_k = m_k \times n_k$, $k = 1, 2, \ldots, 10$. Note that these sizes are designed such that they increase in a "near" linear manner, which is desirable for proper representation of the results.

4.2. Efficiency and accuracy

These 10 linear systems of equations with 3D right-hand sides are solved by the GMRES(m) solver with various settings of m and ε on an Intel Pentium IV computer (CPU speed 2.8 GHz; RAM 1 GB). The same set of problems is also solved by the Gauss solver found in the original BEM extrapolation code for the constant α force-free magnetic field and an LU solver found in the Linear Algebra Package, i.e. LAPACK (Anderson et al. 1999) for comparative purpose. The respective CPU times of the Gauss solver, LU solver, and GMRES(m) solver with $m = 50$ and $\varepsilon = 10^{-16}$, $\varepsilon = 10^{-6}$ or $\varepsilon = 10^{-2}$ are listed in Table 1 (CPU times for assembling the BEM matrices and for extrapolating the fields for 15 layers are also included in the table for reference purpose; CPU times for these two stages of the BEM extrapolation are independent of the choice of the linear system solvers). Here, the setting of $\varepsilon = 10^{-6}$ is quite typical for the GMRES(m) solver or other iterative solvers (see, e.g., Saad & Schultz 1986; Spyropoulos et al. 2004). This is understandable since too much accuracy is not necessary in their context of applications. Actually, it is quite safe to increase ε to be 10^{-2} in our case, as will be discussed below. Nevertheless, we also included $\varepsilon = 10^{-16}$ for the GMRES(m) solver in our experiments to maintain a *fair* comparison against the Gauss solver and the LU solver, since the accuracy level of the solutions obtained by the two direct solvers stays around 10^{-14} while the actual level of accuracy of the solutions obtained by the GMRES(m) solver is about one or two orders lower than the prescribed ε. The Krylov dimension m is set to be 50 (see discussion below).

One may observe in Table 1. that the CPU time for the Gauss solver increases very quickly, ranging from about 12.5 s to about 17 h, when N increases from 1023 to 7395, and that it is almost always dominant compared to the assembling and extrapolation times (even much bigger for the bigger sizes). One can also see that the ratio of the CPU time taken by the LU solver to the CPU time taken by the Gauss solver is considerably reduced when N increases from 1023 to 7395, i.e. by a factor ranging from 9 to 100! Still, we have verified that both of the solvers are $O(N^3)$ procedures. The large disparities of the CPU time concerning the two direct solvers should be due to the fact that the Gauss solver we tested is implemented using the Gaussian elimination method with full pivoting while the LU solver we tested is implemented using Crout' s method with partial pivoting and heavily optimized for all target processor architectures including the one we used. The fact that both of the direct solvers provide solutions with almost the same accuracy makes it a pity that the slowest solver has been in use for so many years. We note that the Gauss solver with full pivoting was chosen during the design of the original BEM extrapolation code because it was believed more accurate (see Liu et al. 1980), but no systematic numerical tests were conducted then to reveal the asymptotic performance behaviour of the two kinds of direct solvers applied to the BEM extrapolation code, since it was not intended for large scale problems at that moment.

Next, we see in Table 1 and Fig. 1a that the GMRES(50) solver with $\varepsilon = 10^{-16}$ becomes faster than the LU solver when N is greater than 2499, but the relative difference remains small over the range of N tested here. As the iterative solver is an $O(N^2)$ procedure, it may have advantages over the LU solver for even bigger problems. However, in both cases, it can be seen that the CPU time needed to solve the linear systems and extrapolate the fields becomes *relatively* dominant compared to the assembly time so that differences will not be very noticeable.

By lifting ε to be 10^{-6} (or 10^{-2}), the GMRES(50) solver is accelerated by a factor ranging from 3 (or 15) to 4 (or 25) when N increases from 1023 to 7395, relatively to the GMRES(m) solver with $\varepsilon = 10^{-16}$. We see that the extrapolating stage becomes relatively dominant in the BEM extrapolation if the GMRES(50) solver with $\varepsilon = 10^{-6}$ or $\varepsilon = 10^{-2}$ is used for solving the BEM systems. In addition, we observe that the CPU time needed by the assembling stage is just slightly less than that needed by the GMRES(50) solver with $\varepsilon = 10^{-6}$.

The above timing data are further illustrated in Fig. 1a. Note that, we have taken the natural logarithm values of the timing data to represent them properly. One can observe there are intersections between the performance curves of the LU solver and the GMRES(50) solver with $\varepsilon = 10^{-6}$ or 10^{-16} around $N = 1023$ or $N = 2499$, which demonstrate that the investigation on the asymptotic performance behaviour of these solvers is indeed necessary for acquiring a more complete knowledge on their performance. Since the trends of variation of these performance curves are rather regular in the current coordinates, we can extrapolate them to predict the solution time of the corresponding solvers for problems with more degrees of freedom (see also Bucher et al. 2002). We estimate that, for a problem with $N = 20\,000$ degrees of freedom, the Gauss solver requires about 22.9 days to get the solution, and the LU solver needs

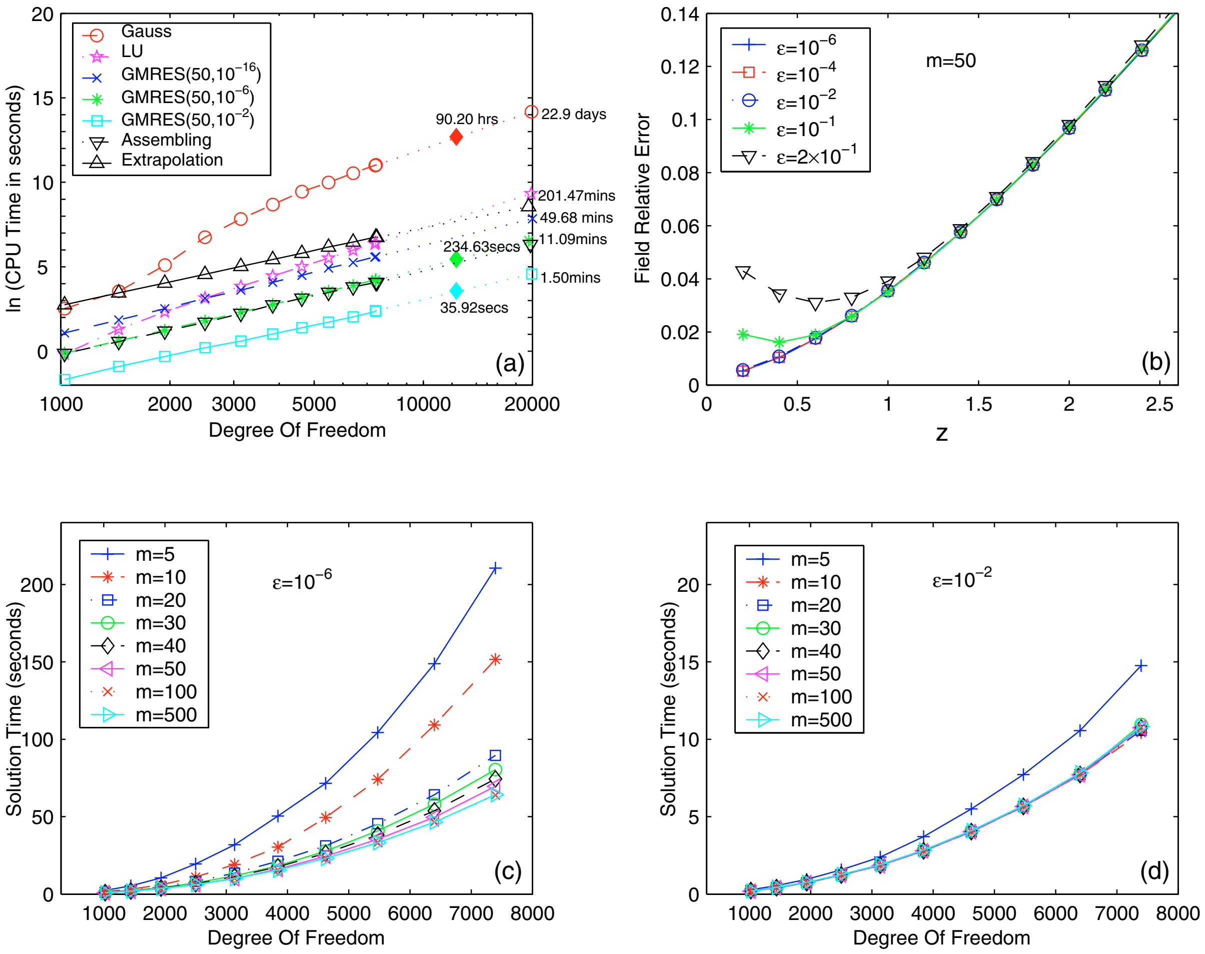

Fig. 1. a) Performance of the Gauss solver, LU solver and GMRES(50) solver with $\varepsilon = 10^{-16}$, 10^{-6} or 10^{-2}; dotted lines are extended from the actual timing data for prediction purpose; diamonds indicate the extra experiments for confirming and correcting the prediction lines; CPU times for assembling the BEM matrices and extrapolating the fields for 15 layers are also plotted for reference purpose. **b)** Impact of the setting of ε of the GMRES(50) solver on the accuracy of the extrapolated field. **c)** Impact of the Krylov dimension m on the convergence behaviour of the GMRES(m) solver for $\varepsilon = 10^{-6}$. **d)** Impact of the Krylov dimension m on the convergence behaviour of the GMRES(m) solver for $\varepsilon = 10^{-2}$.

Table 2. Iteration information of the GMRES(m) solver with $m = 50$ and various settings of ε ($\varepsilon_1 = 10^{-16}$, $\varepsilon_2 = 10^{-6}$, $\varepsilon_3 = 10^{-2}$).

DOF (N)	GMRES(50, ε_1)	GMRES(50, ε_2)	GMRES(50, ε_3)
1023	(129, 134, 137)	(35, 41, 42)	(6, 11, 11)
1443	(140, 145, 148)	(38, 43, 45)	(6, 11, 11)
1935	(149, 151, 173)	(40, 45, 47)	(6, 11, 11)
2499	(172, 175, 176)	(42, 47, 49)	(6, 11, 11)
3135	(178, 183, 182)	(43, 50, 53)	(5, 11, 11)
3843	(186, 194, 194)	(45, 51, 59)	(5, 11, 11)
4623	(193, 200, 201)	(46, 56, 61)	(5, 11, 11)
5475	(200, 228, 230)	(47, 59, 63)	(5, 11, 11)
6399	(204, 230, 233)	(48, 60, 65)	(5, 11, 11)
7395	(230, 236, 238)	(49, 64, 67)	(5, 11, 11)

about 3.36 h for the same thing, while the GMRES(50) solver can bring out its solution in about 50 min for $\varepsilon = 10^{-16}$, about 11 min for $\varepsilon = 10^{-6}$, or less than 2 min for $\varepsilon = 10^{-2}$. To verify our prediction, we have executed an additional experiment by solving a problem with a degree of freedom $N = 12\,321$ with the Gauss solver and the GMRES(50) solver with $\varepsilon = 10^{-6}$ or $\varepsilon = 10^{-2}$ respectively, as individually indicated by the symbols "♦" in Fig. 1a.

The above timing data, of course, is associated to the machine architectures and compiler options, but the relative relations among these data should be less dependent on the computational environment. To this point, we have collected in Table 2 the iteration information of the GMRES(50) solver (which is independent on the computing environment) for the three concerned values of ε. Note that, we have represented the iteration information with 3-component vectors since each of the linear systems has three right-hand sides. We see that, for each of the cases of ε, the iteration steps for the second and third right-hand sides for each degree of freedom are close to (or same) each other, and greater than those of the first ones. This is understandable since there is a symmetrical relation between the later two components of the boundary data (see (14) and (15)), while the first component of the boundary data is rather different from the other two. In addition, we point out that, for the cases of $\varepsilon = 10^{-16}$ or $\varepsilon = 10^{-6}$, the number of iteration steps for each of the right-hand sides increases with N. Furthermore, the larger ε is, the slower the convergence rate of the GMRES(50) solver. On the contrary, for $\varepsilon = 10^{-2}$ the number of iteration steps for each of the right-hand sides remains constant or even decreases when N increases. Finally, we verified, for each problem size, that the ratios of CPU times corresponding to the three variants of the iterative solver in Table 1 are nearly exactly proportional to the corresponding ratios of numbers of iterations from Table 2.

The above analysis shows that the GMRES(50) solver holds great promise in efficiency compared with the Gauss solver or the LU solver, especially when the relative residual bound ε is increased up to 10^{-2}. One may ask, however, whether it is reasonable to increase ε up to a value as big as 10^{-2}! To answer this question, we fix $m = 50$ and discretize the problem with a grid of 75×73, and solve the resulting linear system of size 5475 by the GMRES(m) solver with $\varepsilon = 10^{-6}$, $\varepsilon = 10^{-4}$, $\varepsilon = 10^{-2}$, $\varepsilon = 10^{-1}$, and $\varepsilon = 2\times10^{-1}$ respectively. Thus, we obtain five sets of 3D boundary derivatives of the magnetic fields of differing accuracies, resulting in five sets of numerical 3D fields of differing accuracies after the extrapolation process. Note that, all five sets of numerical 3D fields are computed on a cubic grid, so that we can use the method in Amari et al. (1998) to measure the accuracy of these computed fields against the analytical one. More precisely, one computes the Euclidean *field* relative error of the extrapolated magnetic fields vs the analytical magnetic field on each extrapolation level (or height) by the formula (cf. Amari et al. 1998):

$$RE^j_{\text{field}} = \left(\frac{\sum\limits_{k=1}^{N_j} \|\widetilde{\boldsymbol{B}}^j_k - \boldsymbol{B}^j_k\|^2}{\sum\limits_{k=1}^{N_j} \|\boldsymbol{B}^j_k\|^2} \right)^{1/2}, \qquad (18)$$

where $\widetilde{\boldsymbol{B}}^j_k$ denotes the extrapolated magnetic field at the kth node of the jth level ($j = 1, \ldots, N_z$); $\boldsymbol{B}^j_k$ denotes the analytical magnetic field at the kth node of the jth level; $\|\cdot\|$ denotes the Euclidean 2-norm; N_j is the number of *valid nodes* on the jth level. Here, the "valid nodes" refer to those whose transverse coordinates belong to a subregion Γ^σ_D of Γ_D, called *valid region* (or *concerned region*) hereafter. Such a treatment is necessary to exclude the errors due to the truncation of the boundary data outside the magnetogram region Γ_D (cf. Amari et al. 1998, p. 256). Here, we take $\Gamma^\sigma_D = \{(x, y) : |x| \le \sigma L_x, |y| \le \sigma L_y\}$ with $\sigma = 0.33$, a value commonly accepted. Clearly, the field relative error defined by (18) is a function of height z and can be represented by a curve, called *FRE − z curve* for short hereafter. This curve was used as a *quantitative* overall evaluation on the quality of the extrapolated 3D magnetic fields by different approaches for comparison purpose in Amari et al. (1998). Figure 1b shows $FRE - z$ curves for the five sets of BEM-extrapolated 3D magnetic fields corresponding to the five different settings of the relative residual bound ε for the GMRES(m) solver. One can see that the field relative errors at a given height are almost the same when ε is set to be 10^{-6}, 10^{-4} or 10^{-2}, but become bigger at the lower heights (say $z < 1$) when ε is increased up to 10^{-1} or 2×10^{-1}. Therefore, we may take $\varepsilon = 10^{-2}$ as a near optimal choice for the relative residual bound of the GMRES(m) solver, in order to obtain a near optimal balance between the convergence speed and the solution accuracy. Still, we do not exclude the case of $\varepsilon = 10^{-6}$ in the subsequent analysis for comparative purpose.

Now, let us consider the choice of the Krylov dimension m of the GMRES(m) solver. The scheme is to solve each of the 10 linear systems of 3D right-hand sides above by the GMRES(m) solver, taking $m = 5, 10, 20, 30, 40, 50, 100,$ and 500 respectively while fixing $\varepsilon = 10^{-6}$ or $\varepsilon = 10^{-2}$. Figure 1c shows the impact of the value of m on the convergence speed of the GMRES(m) solver for $\varepsilon = 10^{-6}$. We see that the GMRES(m) solver is accelerated by a factor of 2 or 3 when the value of m is increased from 5 to 20 and upwards. Meanwhile, the speed-up becomes less significant (or even null) when m is greater than 20 (or 100). It appears to us that $m = 50$ can be an optimal choice for the Krylov dimension of the GMRES(m) solver while fixing $\varepsilon = 10^{-6}$, since it does not increase too much workspace compared with $m = 100$, and may save more time compared with $m = 20$ when N becomes even bigger. The situation is somewhat changed when ε is increased to 10^{-2}, as shown in Fig. 1d (note that we have zoomed in 10 times in Fig. 1d compared to Fig. 1c for proper representation of the results). We see that the convergence speed of the GMRES(m) solver is almost not changed when m is set to be 10 and upwards. In addition, we observe some subtle phenomena of mathematical interest when further zooming in the figure. For example, for all N, the setting of $m = 500$ is even slightly worse than that of $m = 10$, concerning the convergence speed of the GMRES(m) solver for $\varepsilon = 10^{-2}$. However, detailed analysis of these phenomena is out of the scope of this paper. Similar phenomena for "DGMRES algorithm" have recently been reported in (Zhou & Wei 2006). Generally speaking, we may take around 20, 30, 40, or 50 as a proper choice of the Krylov dimension m of the GMRES(m) solver for both cases of $\varepsilon = 10^{-6}$ and $\varepsilon = 10^{-2}$, while $m = 50$ is our preference since relatively bigger m can make the iterative solver more stable, and it appears more fair to set $m = 50$ when comparing the convergence speed of GMRES(m, 10^{-6}) and GMRES(m, 10^{-2}).

5. Conclusion and discussion

We have successfully applied the GMRES(m) method to the BEM extrapolation of the solar force-free magnetic field on a well known test problem, and made an effort on the optimal choice of the Krylov dimension m and the relative residual bound ε[1]. By investigating the asymptotic convergence behaviour of the GMRES(m) solver applied to the BEM systems of linear equations, we reveal that the GMRES(m) solver, with properly determined m and ε, is much more efficient than the Gauss solver or the LU solver when the size of the problem becomes very large, while the accuracy of the extrapolated field can still be satisfactory within some height. Thus, the BEM extrapolation has taken a significant step forward towards the routine task of extrapolating the daily observed photospheric magnetogram data. In the future, we will apply the GMRES(m) algorithm to the non-linear case of the BEM extrapolation proposed by Yan & Sakurai (2000). Finally, we note that it is possible to apply the GMRES(m) algorithm to other field extrapolation methods containing large scale linear system of equations, such as the method for "a better linear force-free field" proposed by Wheatland (1999) or the calculation method for the non-linear force-free field proposed by Sakurai (1981).

Despite the encouraging progress on the efficiency, there are still problems to be dealt with when applying the rebuilt BEM extrapolation code to the real data. (i) The BEM matrix requires considerable amount of memory for storage when N becomes very large (see, e.g., Wheatland 1999; Sakurai 1981 for similar problems). For example, for a magnetogram with 1024 × 1024 pixels, that amount could be as large as 8192 GB, which would be challenging even for a supercomputer. Fortunately, this problem has been widely recognized in many research fields, and some promising methods, such as multipole expansions (Greengard & Rokhlin 1987), panel clustering (Hackbusch & Nowak 1989), and wavelet compression method (Beylkin et al. 1991) have been proposed. We have already begun to study the adaptation of the wavelet approach to the solution of integral equation (Li et al. 2005) and will apply it to tackle the

[1] We mention that the optimum ranges for the parameters m and ε of the GMRES(m) solver may differ somewhat from the values obtained here if different test fields and higher values of α are considered.

storage problem in the BEM extrapolation in a forthcoming paper (Li et al. in preparation). (ii) Since two dimensional numerical quadrature is involved both in the procedure of generating the BEM matrix and in the stage of computing the magnetic field in Ω in the BEM extrapolation, computational cost of numerical quadrature can be considerable in the case that high space resolution is desirable. Therefore, it is necessary to seek more effective quadrature methods. We have also started to work on this aspect. (iii) The convergence behaviour of the GMRES(m) method subjected to the BEM extrapolation with real magnetogram data remains to be investigated. However, it can be expected that the noise in the observed data, the distribution of α, the size and resolution of the magnetogram region may affect the convergence behaviour of the GMRES(m) method so that slow convergence may be encountered due to these factors. In that case, we suggest trying a bigger m, but the ε should not be bigger than 10^{-2}, according to the numerical experiments done in this paper. In addition, since we have implemented most of SPARSKIT (which contains, among other things, 10 iterative solvers) in our rebuilt BEM extrapolation code, the users of our code can consider using other solvers of SPARSKIT, such as DQGMRES, BCG, etc., in case the GMRES(m) solver converges slowly for the specific problem under consideration or to check the accuracy by comparing results provided by different kinds of (fast) solvers. Furthermore, we have implemented our new code in a modular way such that the users usually need not modify the source code but simply change parameters in input files to change algorithms or set parameters and may adopt new standard packages if more advanced method, known as "preconditioning", have to be considered. Alternatively the users may just use the LU solver from LAPACK for comparison purposes on small size problems. Finally, we note that further numerical experiments will be done in Li (in preparation) to compare the convergence behaviour of some other iterative solvers in the SPARSKIT, in the context of the BEM extrapolation.

Acknowledgements. The authors are deeply indebted to the anonymous referee for her/his critical and constructive comments on the first version of the paper. This work was supported by NSFC (10225313), MOST (G2000078403), and CAS grants. One of the authors (Yiwei Li) is also supported by Taiyuan University of Science and Technology, Taiyuan, China, while another (R. Langlet) was supported by the region of Franche-Comté, France. Professors Qinshao Zhang, Gening Xu, Zhiqing Li, Xingjian Yang, Xiyun Wang and Junlin Li of TYUST are thanked for their persistent encouragement and support to Y. Li. Professor Jingxiu Wang (NAOC) is thanked for a general discussion on the solar magnetic fields and for providing some useful references at the beginning of this work. Professor Huaning Wang (NAOC), Professor Xiangchu Feng (XDU) and Dr. Zhuheng Li (XDU) are thanked for some useful discussions. Drs. Chijie Xiao, Jiangtao Su, Guiping Zhou, Chengming Tan, and Yuzong Zhang of NAOC are thanked for their warm help and concern on this work. Professor Qijun Fu, Dr. Shujuan Wang, Sr. Engineers Yuying Liu, Zhijun Chen, Huirong Ji and Guoshu Ji, Ms. Hongxia Nie, and Ms. Hongrong Du are thanked for their warm help to Y. Li during his stay at NAOC.

References

Alissandrakis, C. E. 1981, A&A, 100, 197
Altschuler, M. D., & Newkirk, G. Jr. 1969, Sol. Phys., 9, 131
Aly, J. J. 1989, Sol. Phys., 120, 19
Aly, J. J. 1992, Sol. Phys., 138, 133
Amari, T., Boulmezaoud, T. Z., & Maday, Y. 1998, A&A, 339, 252
Amari, T., Boulmezaoud, T. Z., & Mikic, Z. 1999, A&A, 350, 1051
Amari, T., Boulmezaoud, T. Z., & Aly, J. J. 2006, A&A, 446, 691
Anderson, E., Bai, Z., Bischof, C. et al 1999, LAPACK Users' Guide, 3rd Ed., SIAM, Philadelphia, PA
Arnoldi, W. E. 1951, Q. Appl. Math., 9, 17
Aschwanden, M. J. 2004, in Physics of The Solar Corona (Springer-Verlag), Chap. 5, 175
Barrett, R., Berry, M., Chan, T. F. et al. 1994, Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, SIAM, Philadelphia, PA
Beylkin, G., Coifman, R., & Rokhlin, V. 1991, Comm. Pure Appl. Math., 44, 141
Bucher, H. F., Wrobel, L. C., Mansur, W. J. et al. 2002, Electronic Journal of Boundary Elements, BETEQ 2001, 187
Chiu, Y. T., & Hilton, H. H. 1977, ApJ, 212, 873
Démoulin, P., & Priest, E. R. 1992, A&A, 258, 535
Gary, G. A. 1989, ApJS, 69, 323
Greengard, L., & Rokhlin, V. 1987, Comput. Phys., 73, 325
Hackbusch, W., & Nowak, Z. P. 1989, Numer. Math., 54, 463
Huang, Y., & Vorst, H. A. 1989, Natural Sci. J. Xiangtan University, 11, 103
Huang, Y., & Vorst, H. A. 1990, Natural Sci. J. Xiangtan University, 12, 135
Li, Y., Yan, Y., Devel, M., et al. 2005, in Wavelet Analysis and Active Media Technology, Proceedings of the 6th international progress, ed. J. P. Li et al. (World Scientific Publishing), 3, 1138
Liu, D., Fei, J., Yu, Y. et al. 1980, Fortran algorithm collection (I) (Press of industry of national defence), 176
Liu, Y., Zhao, X. P., Hoeksema, J. T. et al. 2002, Sol. Phys., 206, 333
Low, B. C. 1982, Sol. Phys., 77, 43
McClymont, A. N., Jiao, L., & Mikic, Z. 1997, Sol. Phys., 174, 191
Nakagawa, Y., Raadu, M. A., & Harvey, J. W. 1973, Sol. Phys., 30, 421
Patton, B. W., & Holloway, J. P. 2002, Ann. Nuclear Energy, 29, 109
Priest, E. R. 1994, Magnetohydrodynamics. In Saas-Fee Advanced Course 24, Plasma Astrophysics, Lecture Notes of the Swiss Society for Astronomy and Astrophysics (SSAA), held in 1994 in Les Diablerets, Switzerland, ed. Benz, A. O., & Courvoisier, T. J.-L., Berlin (New York: Springer)
Quarteroni A., & Valli A. 1997, Numerical approximation of partial differential equations, (New York: Springer-Verlag)
Ravindra, B., & Muddun B. 2004, Future Generation Computer Systems, 20, 389
Ronald, B. M. 2005, Appl. Numer. Math., 54, 222
Saad, Y. 1994, Sparskit: a basic toolkit of sparse matrix computations, version 2. Tech. rep., Research Institute for Advance Computer Science, NASA Ames Research Center, Moffet Field, CA
Saad, Y. 1996, Iterative Methods for Sparse Linear Systems (Boston: PWS Publishing Company)
Saad, Y., & Schultz, M. H. 1986, SIAM J. Sci Statist. Comput., 7, 856
Saad, Y., & Wu, K. 1996, Numer. Linear Algebra Appl., 3, 329
Sakurai, T. 1981, Sol. Phys., 69, 343
Sakurai, T. 1989, Space Sci. Rev., 51, 11
Schmidt, H. U. 1964, in Physics of Solar Flares, NASA SP 50, 107
Schmieder, B., & Aulanier, G. 2003, Adv. Space Res., 32, 1875
Seehafer, N. 1978, Sol. Phys., 58, 215
Spyropoulos, A. N., Palyvos, J. A., & Boudouvis, A. G. 2004, Computer Methods in Applied Mechanics and Engineering, 193, 4707
Tuminaro, R. S., Heroux, M., Hutchinson, S. A., et al. 1999, Official Aztec User's Guide: Version 2.1, `http://www.cs.sandia.gov/CRF/pspapers/Aztec_ug_2.1.ps`
Valori, G., Kliem, B., & Keppens, R. 2005, A&A, 433, 335
Wang, J. 1999, Fund. Cosmic Phys., 20, 251
Wang, H., Yan, Y., & Sakurai, T. 2001, Sol. Phys., 201, 323
Wang, T., Yan, Y., Wang, J., et al. 2002, ApJ, 572, 580
Wheatland, M. S. 1999, ApJ, 518, 948
Wheatland, M. S. 2004, Sol. Phys., 222, 247
Wu S. T., Chang H. M., & Hagyard, M. J. 1985, in Measurement of Solar Vector Magnetic Fields, NASA Conf. Pub. 2374, ed. M. J. Hagyard, Marshall Space Flight Center Workshop, 17
Wu S. T., Sun M. T., Chang H. M. et al. 1990, ApJ, 362, 698
Yan, Y. 1995, Sol. Phys., 159, 97
Yan, Y. 2003, Space Sci. Rev., 107, 119
Yan, Y., & Wang, J. 1995, A&A, 298, 277
Yan, Y., & Sakurai, T. 1997, Sol. Phys., 174, 65
Yan, Y., & Sakurai, T. 2000, Sol. Phys., 195, 89
Yan, Y., & Li, Z. 2006, ApJ, 638, 1162
Yan, Y., Yu, Q., & Kang, F. 1991, Sol. Phys., 136, 195
Yan, Y., Yu, Q., & Kang, F. 1993, In The Magnetic and Velocity Fields of Solar Active Regions, ed. Zirin, H., et al., ASP Conf. Ser., 46, 127
Yan, Y., Yu, Q., & Wang, T. J. 1995, Sol. Phys., 159, 115
Yan, Y., Auschwanden, M. J., Wang, S. et al. 2001a, Sol. Phys., 204, 27
Yan, Y., Deng, Y., Karlicky, M. et al. 2001b, ApJ, 551, L115
Zhang, C., & Wang, J. 2002, Sol. Phys., 205, 303
Zhou, J., & Wei, Y. 2006, Appl. Math. Comput., 176, 293