Princeton Plasma Physics Laboratory

PPPL- 4476

PPPL- 4476

Pseudo-transient Continuation Based Variable Relaxation Solve in Nonlinear Magnetohydrodynamic Simulations

Jin Chen

December, 2009

PPPL PRINCETON PLASMA PHYSICS LABORATORY

Prepared for the U.S. Department of Energy under Contract DE-AC02-09CH11466.

Princeton Plasma Physics Laboratory
Report Disclaimers

Full Legal Disclaimer

Trademark Disclaimer

PPPL Report Availability

Princeton Plasma Physics Laboratory:

http://www.pppl.gov/techreports.cfm

Office of Scientific and Technical Information (OSTI):

http://www.osti.gov/bridge

Related Links:

U.S. Department of Energy

Office of Scientific and Technical Information

Fusion Links

International Journal of Computer Mathematics

Vol. 00, No. 00, Month 200x, 1–19

Pseudo-transient continuation based Variable Relaxation Solve in Nonlinear Magnetohydrodynamic Simulations †

JIN CHEN*

Princeton Plasma Physics Laboratory, Princeton, NJ 08543, USA

(*Received 00 Month 200x; revised 00 Month 200x; in final form 00 Month 200x*)

Efficient and robust Variable Relaxation Solver, based on pseudo-transient continuation, is developed to solve nonlinear anisotropic thermal conduction arising from fusion plasma simulations. By adding first and/or second order artificial time derivatives to the system, this type of method advances the resulting time-dependent nonlinear PDEs to steady state, which is the solution to be sought. In this process, only the stiffness matrix itself is involved so that the numerical complexity and errors can be greatly reduced. In fact, this work is an extension of integrating efficient linear elliptic solvers for fusion simulation on Cray X1E.

Two schemes are derived in this work, first and second order Variable Relaxations. Four factors are observed to be critical for efficiency and preservation of solution's symmetric structure arising from periodic boundary condition: refining meshes in different coordinate directions, initializing nonlinear process, varying time steps in both temporal and spatial directions, and accurately generating nonlinear stiffness matrix. First finer mesh scale should be taken in strong transport direction; Next the system is carefully initialized by the solution with linear conductivity; Third, time step and relaxation factor are vertex-based varied and optimized at each time step; Finally, the nonlinear stiffness matrix is updated by just scaling corresponding linear one with the vector generated from nonlinear thermal conductivity.

Keywords: nonlinear anisotropic transport; pseudo-transient continuation; Variable Relaxation; Jacobian free Newton Krylov iteration; nonlinear magnetohydrodynamic simulation;

AMS Subject Classifications: 65F10; 81T80; 82C80; 74F15; F.2.1

†This work is supported by DOE Contract DE-AC02-76CH03073.
*Corresponding author. Email: jchen@pppl.gov

ISSN 0020-7160 print/ISSN 1029-0265 online

http://www.tandf.co.uk/journals

1. Introduction

In fusion plasma physics simulation [1], the steady state of nonlinear anisotropic thermal conduction can be modeled by the following nonlinear elliptic equation

$$\frac{\partial}{\partial x}(\kappa_x \frac{\partial T}{\partial x}) + \frac{\partial}{\partial y}(\kappa_y \frac{\partial T}{\partial y}) = s \tag{1}$$

on a 2D rectangular domain ABCD: $[0, L_x] \times [0, L_y]$ with four vertexes at A$(0,0)$, B$(0, L_x)$, C(L_x, L_y), and D$(0, L_y)$. The coordinate is given in Cartesian (x, y) system and discretized as mesh (x_i, y_j). The magnetic field is directed in the y direction, and accordingly we can set $\kappa_x = 1$ and κ_y as an nonlinear function of the temperature T, parallel to magnetic field line. Therefore we can omit κ_x and denote κ_y by $\kappa_{||}$ to make its meaning more clear. The periodic boundary condition is set on edges AD and BC, and Dirichlet boundary conditions are set on edges AB and CD. This setup allows us to separate the effects of grid misalignment from the boundary effects. The upper boundary, CD, represent the material surface where the temperature is low, and the boundary condition there is $T_{CD} = 1$. At the lower boundary, AB, the inflow boundary condition is $T_{AB}(x) = 10 + 40e^{(-|x - L_x/2|)}$.

Finite element discretization [2] generates the following nonlinear system

$$(\mathbf{S}_{xx} + \mathbf{S}_{yy}(\boldsymbol{T}))\boldsymbol{T} = \mathbf{M}\boldsymbol{s}. \tag{2}$$

$\mathbf{M}$ is the mass matrix. $\mathbf{S}_{xx}$ and $\mathbf{S}_{yy}(T)$ are the stiffness matrices contributed by operator $\frac{\partial^2 T}{\partial x^2}$ and $\frac{\partial}{\partial y}(\kappa_{||}\frac{\partial T}{\partial y})$, respectively. $\boldsymbol{T}$ is the temperature profile to be solved. When $\kappa_{||}$ is linear, $\mathbf{S}_{yy}(\boldsymbol{T})$ reduced to $\kappa_{||}\mathbf{S}_{yy}$. Newton-Krylov method can be used to solve system (2). But usually it is quite expensive to update Jacobian at each iteration. Although the Jacobian-free variation [3, 4] is more efficient, information of the Jacobian is still needed to form the preconditioner.

In this work we present an alternative way, Variable Relaxation [5, 6], to solve the nonlinear system (1). This is a class of iterative methods which solve the elliptic equations by adding first and/or second

order time derivative terms to eq.(1) to convert it to nonlinear parabolic or hyperbolic equation and then marching the system to steady state. In this marching process, only the nonlinear stiffness matrix $\mathbf{S}_{yy}(\boldsymbol{T})$ itself is involved and needs to be updated regularly.

We have been using this type of idea on Cray X1E to design efficient linear elliptic solvers for M3D code [6]. Although It takes longer to converge, each iteration is much cheaper than other iterative solvers [7] so that it still wins on vector architectured machines.

Two versions of Variable Relaxation are implemented. one is first order variable relaxation by adding first order time derivative term; the other is second order variable relaxation by adding one more second order time derivative term. The nonlinear iteration can be completed in two steps:

Step 1: solve eq.(1) with linear conductivity $10^0 \leq \kappa_{||} \leq 10^9$.

Step 2: solve eq.(1) with nonlinear conductivity $\kappa_{||} = T^{5/2}$.

The solution from '**Step 1**' is used as an initial guess for '**Step 2**'. Experiments will show that this is a very powerful strategy to accelerate convergence. We will also demonstrate how to choose artificial time step from CFL condition and relaxation factor from dispersion relation to achieve optimization. These parameters differ in each timestep and individual grid point as well. An efficient way to generate the stiffness matrix is also to be discussed in order to preserve the symmetry structure of the solution as a result of periodic boundary condition.

2. First order relaxation and numerical schemes

The so called first order relaxation is obtained by adding a first order time derivative term to eq. (1)

$$\frac{\partial u}{\partial t} = \frac{\partial^2 T}{\partial x^2} + \frac{\partial}{\partial y}(\kappa_{||}\frac{\partial T}{\partial y}). \tag{3}$$

Discretizing it in temporal direction by finite difference and spatial directions as in system (2), we have

$$(\frac{1}{\delta t}\mathbf{M} - \theta \mathbf{S}_{non})\boldsymbol{T}^{k+1} = [\frac{1}{\delta t}\mathbf{M} + (1-\theta)\mathbf{S}_{non})]\boldsymbol{T}^{k} - \mathbf{M}s. \tag{4}$$

$0 \le \theta \le 1$. When $\theta = 0$, the system is fully explicit; when $\theta = 1$, the system is fully implicit; when $\theta = \frac{1}{2}$, the system is stable and has smallest truncation error as well. $\mathbf{S}_{non} = \mathbf{S}_{xx} + \mathbf{S}_{yy}(\boldsymbol{T})$. The system marches from kth timestep to next $(k+1)$th timestep. δt is the artificial time step which should be chosen to be small enough to make the scheme stable and big enough to allow the system approach steady state quickly. According to CFL condition, δt is related to mesh scales δx in x direction and δy in y direction by

$$\delta t \le \frac{1}{2}\frac{1}{\frac{1}{\delta x^2} + \kappa_{\|}\frac{1}{\delta y^2}} = \frac{\delta x \delta y}{4}\frac{2}{\frac{\delta y}{\delta x} + \kappa_{\|}\frac{\delta x}{\delta y}} \equiv \frac{\delta x \delta y}{4}\bar{\delta t} \tag{5}$$

for explicit case ($\theta = 0$). Obviously, when $\kappa_{\|} = 1$ and δx and δy are taken to be identical, the above formulae is reduced to the very familiar expression for 2D heat conduction equation. More can be derived if we different $\bar{\delta t}$ with respect to δx and δy

$$\frac{\partial \bar{\delta t}}{\partial \delta x} = -2\frac{-\frac{\delta y}{\delta x^2} + \kappa_{\|}\frac{1}{\delta y}}{(\frac{\delta y}{\delta x} + \kappa_{\|}\frac{\delta x}{\delta y})^2} = 2\frac{1}{\delta y}\frac{\frac{\delta y^2}{\delta x^2} - \kappa_{\|}}{(\frac{\delta y}{\delta x} + \kappa_{\|}\frac{\delta x}{\delta y})^2},$$

$$\frac{\partial \bar{\delta t}}{\partial \delta y} = -2\frac{\frac{1}{\delta x} - \kappa_{\|}\frac{\delta x}{\delta y^2}}{(\frac{\delta y}{\delta x} + \kappa_{\|}\frac{\delta x}{\delta y})^2} = 2\frac{\delta x}{\delta y^2}\frac{\kappa_{\|} - \frac{\delta y^2}{\delta x^2}}{(\frac{\delta y}{\delta x} + \kappa_{\|}\frac{\delta x}{\delta y})^2}.$$

When $\kappa_{\|} > 1$, most likely we will have $\frac{\partial \bar{\delta t}}{\partial \delta x} < 0$ and $\frac{\partial \bar{\delta t}}{\partial \delta y} > 0$. This suggests that δx should be taken as large as possible, while δy as small as possible.

The convergence of scheme (4) can be analyzed in the following way. Given the form of transient solution of eq.(3) as $\tilde{u} = e^{-\gamma t}\sin\frac{m\pi x}{L_x}\sin\frac{n\pi y}{L_y}$, the operator $\frac{\partial^2}{\partial x^2} + \frac{\partial}{\partial y}(\kappa_{\|}\frac{\partial}{\partial y})$ has eigenvalues $\lambda_{mn} = \pi^2(\frac{m^2}{L_x^2} + \kappa_{\|}\frac{n^2}{L_y^2})$. m and n are the mode numbers in x and y directions, respectively. Then the decaying rate is $-\lambda_{11}$ and the corresponding decaying time can be found by

$$t = \frac{1}{\lambda_{11}} = \frac{1}{\pi^2}\frac{1}{\frac{1}{L_x^2} + \kappa_{\|}\frac{1}{L_y^2}}.$$

The number of iterations needed for convergence can be predicted by

$$N_{its} \equiv \frac{t}{\delta t} = \frac{2}{\pi^2}\frac{\frac{1}{\delta x^2} + \kappa_{\|}\frac{1}{\delta y^2}}{\frac{1}{L_x^2} + \kappa_{\|}\frac{1}{L_y^2}} = \frac{2}{\pi^2}\frac{\frac{N_x^2}{L_x^2} + \kappa_{\|}\frac{N_y^2}{L_y^2}}{\frac{1}{L_x^2} + \kappa_{\|}\frac{1}{L_y^2}}.$$

When $\kappa_{\|} \to \infty$

$$N_{its} \to \frac{2}{\pi^2}N_y^2 \approx \frac{1}{5}\frac{N_y}{N_x}(N_xN_y) \equiv c(N_xN_y).$$

(N_xN_y) is the number of unknowns. After some experiments, we found the optimized coefficient should be $c = 0.64$ for the problem we are studying. Also from the following expression we found the number of iterations increases as $\kappa_{\|}$ gets larger

$$\frac{dN_{its}}{d\kappa_{\|}} = \frac{2}{\pi^2}\frac{(N_y^2 - N_x^2)}{(\frac{L_y}{L_x} + \kappa_{\|}\frac{L_x}{L_y})^2} > 0$$

as long as $\delta y \leq \delta x$. This is illustrated in the red curve of Fig.1.

'[Insert figure 1 about here]'

3. Second order relaxation and numerical schemes

In addition to supply a first order derivative term in eq. (3), the second order relaxation is obtained by adding a relaxation factor, τ, and a second order time derivative term to eq. (1)

$$\frac{\partial^2 u}{\partial t^2} + \frac{2}{\tau}\frac{\partial u}{\partial t} = \frac{\partial^2 T}{\partial x^2} + \frac{\partial}{\partial y}(\kappa_{\|}\frac{\partial T}{\partial y}). \tag{6}$$

Again it can be discretized

$$\mathbf{M}\frac{\boldsymbol{T}^{k+1}-2\boldsymbol{T}^{k}+\boldsymbol{T}^{k-1}}{\delta t^2}+\mathbf{M}\frac{1}{\tau}\frac{\boldsymbol{T}^{k+1}-\boldsymbol{T}^{k-1}}{\delta t}=\theta\mathbf{S}_{non}\boldsymbol{T}^{k+1}+(1-\theta)\mathbf{S}_{non}\boldsymbol{T}^{k}-\mathbf{M}s,$$

or rearranged as

$$\begin{aligned}&[(1+\tfrac{\delta t}{\tau})\mathbf{M}-\theta\mathbf{S}_{non}]\boldsymbol{T}^{k+1}=\\&-(1-\tfrac{\delta t}{\tau})\mathbf{M}\boldsymbol{T}^{k-1}+[2\mathbf{M}+\delta t^2(1-\theta)\mathbf{S}_{non}]\boldsymbol{T}^{k}-\delta t^2\mathbf{M}s.\end{aligned}\tag{7}$$

We need the solutions at both kth and $(k-1)$th time steps to update the solution at $(k+1)$th time step.

The CFL condition can be expressed as $\delta t^2(\frac{1}{\delta_x^2}+\kappa_{\|}\frac{1}{\delta_y^2})\leq 1$. Therefore,

$$\delta t\leq\frac{1}{\sqrt{\frac{1}{\delta x^2}+\kappa_{\|}\frac{1}{\delta_y^2}}}=\frac{\sqrt{\delta x\delta y}}{\sqrt{\frac{\delta y}{\delta x}+\kappa_{\|}\frac{\delta x}{\delta y}}}=\frac{\sqrt{\delta x\delta y}}{\sqrt{2}}\frac{\sqrt{2}}{\sqrt{\frac{\delta y}{\delta x}+\kappa_{\|}\frac{\delta x}{\delta y}}}\tag{8}$$

for explicit case ($\theta=0$). Again, the above expression is reduced to the familiar stability condition for 2D convection equation when $\kappa_{\|}=1$ and δx and δy are taken to be identical, The relaxation factor can be found by looking for the transient solution of eq.(6) as well. The decay rates satisfy $\gamma^2-\frac{2}{\tau}\gamma+\lambda^{mn}=0$, or $\gamma=\frac{1}{\tau}\pm(\frac{1}{\tau^2}-\lambda^{mn})^{1/2}$. For optimal damping, we choose $\tau^2=\frac{1}{\lambda^{11}}=1/[(\frac{L_y^2}{L_x^2}+\kappa_{\|})\frac{\pi^2}{L_y^2}]$, i.e.,

$$\tau=\frac{1}{\pi\sqrt{\frac{1}{L_x^2}+\kappa_{\|}\frac{1}{L_y^2}}}\tau=\frac{\sqrt{L_xL_y}}{\sqrt{2}\pi}\frac{\sqrt{2}}{\sqrt{\frac{L_y}{L_x}+\kappa_{\|}\frac{L_x}{L_y}}}\tag{9}$$

and the number of iterations for convergence can be predicted by

$$\begin{aligned}&N_{its}\equiv\tfrac{\tau}{\delta t}\\&=\tfrac{1}{\pi}\frac{\sqrt{\frac{1}{\delta x^2}+\kappa_{\|}\frac{1}{\delta_y^2}}}{\sqrt{\frac{1}{L_x^2}+\kappa_{\|}\frac{1}{L_y^2}}}=\tfrac{1}{\pi}\frac{\sqrt{\frac{N_x^2}{L_x^2}+\kappa_{\|}\frac{N_y^2}{L_y^2}}}{\sqrt{\frac{1}{L_x^2}+\kappa_{\|}\frac{1}{L_y^2}}}=\tfrac{1}{\pi}\frac{\sqrt{\frac{25}{9}N_x^2+\kappa_{\|}N_y^2}}{\sqrt{\frac{25}{9}+\kappa_{\|}}}=\tfrac{1}{\pi}\frac{\sqrt{\frac{25}{9}\frac{N_x}{N_y}+\kappa_{\|}\frac{N_y}{N_x}}}{\sqrt{\frac{25}{9}+\kappa_{\|}}}\sqrt{N_xN_y}\end{aligned}$$

When $\kappa_{||} \to \infty$

$$N_{its} \to (\frac{1}{\pi}\sqrt{\frac{N_y}{N_x}})\sqrt{N_x N_y} \equiv c\sqrt{N_x N_y}.$$

Experiments show that the optimal coefficient would be $c = 0.6$. The number of iteration increases as the conductivity $\kappa_{||}$ increases. This can be understood from the following expression and the black curve of Fig.1:

$$\frac{dN_{its}}{d\kappa_{||}} = \frac{1}{\pi}\frac{\sqrt{3N_y^2 + \kappa_{||}N_y^2} - \sqrt{3N_x^2 + \kappa_{||}N_y^2}}{3 + \kappa_{||}} > 0.$$

4. Variable relaxations

When $\kappa_{||}$ is an nonlinear function of T, $\kappa_{||}$ changes as T_{ij}^k changes at every grid point (i, j) and every time step k. Therefore, time step and relaxation factor changes as well. This is why the name "Variable" is given. From now on, schemes (4) is called VR(4), scheme (7) VR(7), and $\kappa_{||}$ is rewritten as κ_{ij}^k in nonlinear case. From the analysis given in the previous two sections, we have

$$\delta t_{ij}^k \leq \frac{1}{2}\frac{1}{\frac{1}{\delta x^2} + \kappa_{ij}^k\frac{1}{\delta y^2}} = \frac{\delta x \delta y}{4}\frac{2}{\frac{\delta y}{\delta x} + \kappa_{ij}^k\frac{\delta x}{\delta y}} \tag{10}$$

for VR(4) and

$$\delta t_{ij}^k \leq \frac{1}{\sqrt{\frac{1}{\delta x^2} + \kappa_{ij}^n\frac{1}{\delta_y^2}}} = \frac{\sqrt{\delta x \delta y}}{\sqrt{\frac{\delta y}{\delta x} + \kappa_{ij}^k\frac{\delta x}{\delta y}}} = \frac{\sqrt{\delta x \delta y}}{\sqrt{2}}\frac{\sqrt{2}}{\sqrt{\frac{\delta y}{\delta x} + \kappa_{ij}^k\frac{\delta x}{\delta y}}} \tag{11}$$

$$\tau_{ij}^k = \frac{\sqrt{L_x L_y}}{\sqrt{2}\pi}\frac{\sqrt{2}}{\sqrt{\frac{L_y}{L_x} + \kappa_{ij}^k\frac{L_x}{L_y}}} \tag{12}$$

for VR(7). The characteristics of $\delta\bar{t}_{ij}^k \equiv \frac{2}{\frac{\delta y}{\delta x}+\kappa_{ij}^k\frac{\delta x}{\delta y}}$ for VR(4) and $\delta\bar{t}_{ij}^k \equiv \frac{\sqrt{2}}{\sqrt{\frac{\delta y}{\delta x}+\kappa_{ij}^k\frac{\delta x}{\delta y}}}$ for VR(7) with respect to δx and δy are given as contour plots in Fig. 2. For the case when $\kappa_{\|} = 1$, $\delta\bar{t}_{ij}^k$ is symmetric with respect to $(\delta x, \delta y)$ and get maximized at $\delta x = \delta y$. This is obvious in the upper two plots. For the case with large $\kappa_{\|}$, such as $\kappa_{\|} = 10^4$ in the bottom two plots, T changes more rapidly in y direction other than x direction. Therefore large δx and relatively small δy will optimize the convergence.

'[Insert figure 2 about here]'

A little extra work will help to understand VR(4) and VR(7) better. In fact, if we define residual

$$r^k = S_{non}T^k - Ms$$

at k the timestep, these two schemes can be rewritten equivalently to

$$\begin{aligned}
\boldsymbol{T}^{k+1} &= \boldsymbol{T}^k + \delta t \mathbf{M}^{-1} r^k \\
&\equiv \boldsymbol{T}^k + \alpha \mathbf{M}^{-1} r^k \\
\boldsymbol{T}^{k+1} &= \tfrac{2}{1+\frac{2}{\tau}\delta t}\boldsymbol{T}^k - \tfrac{1+\frac{2}{\tau}\delta t}{1+\frac{2}{\tau}\delta t}\boldsymbol{T}^{k-1} + \tfrac{\delta t^2}{1+\frac{2}{\tau}\delta t}\mathbf{M}^{-1} r^k \\
&\equiv \alpha \boldsymbol{T}^k + (1-\alpha)\boldsymbol{T}^{k-1} + \beta \mathbf{M}^{-1} r^k.
\end{aligned}$$

Depending on how to choose α and β, they converge differently.

For linear case ($\kappa_{\|}$ = constant), δt and τ are constant so that α and β are constant. They are equivalent to first and second order stationary iterations. For optimal choices of δt and τ, second order method converges with an order of magnitude faster than the first order method.

For the nonlinear case discussed here, δt and τ vary as iterations proceed. Therefore, they correspond to nonstationary first and second order methods. Different choices of optimal sequences of δt^k and τ^k can reproduce Chebyshev iteration, Conjugate Gradient, Richardson iteration, or the one shown in this work. Nevertheless, the asymptotic rate of convergence are the same as for the stationary ones [8].

5. Numerical issues

In practical application we are going to take the equal sign (=) in the two expressions given for δt by (10) and (11). Due to nonuniform meshes and nonlinearity of the problem, δt and the damping factor τ are modified by scaling factors t_{scale} and τ_{scale}. The optimal δt and τ in both cases can be found by tuning these two parameters. This is summarized in table 1.

'[Insert table 1 about here]'

$\frac{\delta x \delta y}{4}$ is the stability criterion for VR(4) when $\kappa_{||} = 1$. $\frac{2}{\frac{\delta y}{\delta x}+\kappa_{||}\frac{\delta x}{\delta y}}$ or $\frac{2}{\frac{\delta y}{\delta x}+\kappa^k_{ij}\frac{\delta x}{\delta y}}$ is the extra term if $\kappa_{||}$ is nonlinear or a scalar which is larger than one. $\frac{\sqrt{\delta x \delta y}}{\sqrt{2}}$ is the stability criterion for VR(7) when $\kappa_{||} = 1$. $\frac{\sqrt{2}}{\sqrt{\frac{\delta y}{\delta x}+\kappa_{||}\frac{\delta x}{\delta y}}}$ or $\frac{\sqrt{2}}{\sqrt{\frac{\delta y}{\delta x}+\kappa^k_{ij}\frac{\delta x}{\delta y}}}$ is the extra term if $\kappa_{||}$ is nonlinear or a scalar which is larger than one. For the relaxation factor τ, $\frac{\sqrt{L_x L_y}}{\sqrt{2}\pi}$ is the criterion for VR(7) when $\kappa_{||} = 1$ and $\frac{\sqrt{2}}{\sqrt{\frac{L_y}{L_x}+\kappa_{||}\frac{L_x}{L_y}}}$ or $\frac{\sqrt{2}}{\sqrt{\frac{L_y}{L_x}+\kappa^k_{ij}\frac{L_x}{L_y}}}$ is the extra term when $\kappa_{||}$ is larger than one or nonlinear.

In order to maximize δt, $\theta = 1$ is chosen in this work. Mesh scales δx and δy are chosen based on the guidelines discussed in the previous sections. As an example, we have $N_x = (16-1)\times 2+1$ and $N_y = (51-1)\times 2+1$ so that δx is 3 times greater than δy. N_x and N_y are the number of corresponding grid points in x and y directions. In this case VR(4) converged in $29,708$ number of iterations at optimal $t_{scale} = 0.174$; while VR(7) converged in $1,308$ number of iterations at optimal $t_{scale} = 0.41, \tau_{scale} = 0.87$. From here we can say that VR(7) is much faster (here 20 times) than VR(4). This is always true for the choices of δt and τ sequences in this work if we check the expressions of N_{its} for both cases. Actually it is easy to understand that the difference of their asympotic convergence rates is an order of magnitude. Hence from now on we will only use VR(7). Although iteration numbers seems to be large, each iteration is very cheap even compared to JFNK which requires preconditioning.

Next let's study the impact of initializing on convergence. As mentioned before, the nonlinear process can be initialized by the solution from the linear system with constant $\kappa_{||}$. This solution has a symmetric structure as demonstrated in Fig.3 in (x, y) coordinate system.

'[Insert figure 3 about here]'

The size of $\kappa_{||}$ changes from 10^0 to 10^9. Given the linear solution with different size of $\kappa_{||}$, the number of iterations for the nonlinear system to reach steady state is given in the table 2. We found the nonlinear convergence doesn't have much difference as long as the linear solution has $\kappa_{||} \geq 2$. It only diverges when a guess is given by $\kappa_{||} = 1$.

'[Insert table 2 about here]'

The marching process is even accelerated by varying δt and τ at each grid point (i, j) and every time step k. We found the iteration won't even converge if δt and τ are chosen uniformly in either temporal or spatial directions.

Finally we give an efficient approach to update the nonlinear stiffness matrix $S_{yy}(T)$ at each time step. The numerical integration has to be carefully chosen in order to keep the symmetric structure as a result of periodic boundary condition. Generally there exists

$$\mathbf{S}_{yy}(\boldsymbol{T}) = -\int\int \kappa_{||} \frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y} d\sigma$$

with N_i and N_j the ith and jth base functions in finite element space. On each triangle, assuming n is the index running through all of the collocation points, then one way to formulate $\mathbf{S}_{yy}(\boldsymbol{T})$ at kth time step would be

$$\mathbf{S}^{ij}_{yy}(\boldsymbol{T}) = \sum_n w(n)\kappa^k(n)\frac{\partial N_i}{\partial y}(n)\frac{\partial N_j}{\partial y}(n)J(n)$$

where $w(n)$, $\kappa^k(n)$, and $J(n)$ are the corresponding weight, conductivity, and Jacobian at nth point. $\frac{\partial N_i}{\partial y}(n)$ and $\frac{\partial N_j}{\partial y}(n)$ are valued at these points as well. As a function of T, $\kappa^k(n)$ can be found by

$$\sum_l (T^k_l)^{5/2} N_l(n) \quad \text{or} \quad \sum_l [T^k_l N_l(n)]^{5/2}$$

where l is the index running through all of the Cartesian grid points on each triangle. But experiments

show that the symmetric structure is destroyed by the above two formulations. Then we designed the following formula

$$\mathbf{S}_{yy}^{ij}(\boldsymbol{T}) = \kappa_{ij}^{k} \sum_{n} w_n \frac{\partial N_i}{\partial y}(n) \frac{\partial N_j}{\partial y}(n) J(n)$$

which leads to

$$\mathbf{S}_{non} = \mathbf{S}_{xx} + \boldsymbol{B}^{k} \mathbf{S}_{yy}$$

where $\boldsymbol{B}^k$ is a vector with component $B_{ij} = \kappa_{ij}^k$ at each grid point given by (i, j). Therefore, we conclude that the nonlinear stiffness matrix $\mathbf{S}_{yy}$ can be updated by just scaling the linear stiffness matrix $\mathbf{S}_{yy}$ using nonlinear vector $\boldsymbol{B}$. This approach not only significantly reduces the computation complexity, but also preserves the symmetric structure of the periodic solution. The nonlinear solution is shown in Fig. 4 again in (x, y) coordinate system. The linear initial guess with $\kappa_{||} = 2 \times 10^4$ given in the left plot is applied.

'[Insert figure 4 about here]'

6. Conclusions

As an extension of developing efficient elliptic solvers for fusion simulation on Cray X1E, Variable Relaxation is constructed by by adding first and/or second order time derivative to the nonlinear elliptic equation and marching the resulting time-dependent PDEs to steady state. Instead of Jacobian, Only the stiffness matrix itself is involved and needs to be updated at each iteration.

Two schemes has been given, first and second order Variable Relaxations. Four numerical issues has been discussed: The mesh scale ratio, nonlinear process initialization, variable time step and relaxation factor, efficient calculation of the nonlinear stiffness matrix. In summary, the mesh needs to be finer in direction with strong conductivity; convergence can be sped up by using the solution from corresponding

linear system as an initial guess; time step and relaxation factor has to be varied at each grid point and every time step as well; only the nonlinear vector, used to update the nonlinear stiffness matrix, needs to be updated regularly. Therefore, the only computation consists of renewing δt_{ij}^k, τ_{ij}^k, and B^k at each iteration, and apparently these approaches results in a flexible, efficient, and robust algorithm to solve nonlinear systems. The second order scheme is an order of magnitude faster than the first order one due to the speedup of slowest mode decaying.

Acknowledgment

The author would like to give cordial thanks to M3D group for their useful discussions and constant supports.

References

[1] W Park et al., 2003, Nonlinear simulation studies of tokamaks and STs. *Nucl. Fusion,* **43**, 483.

[2] J Chen, S C Jardin, H R Strauss, 2005, Solving anisotropic transport equation on misaligned grids. *LNCS,* **3516**, 1076–1079.

[3] D A Knoll, D E Keyes, 2004, Jacobian-free Newton-Krylov methods: a survey of approaches and applications. *J comp. Phys.,* **193**, 357–397.

[4] A Ern, V Giovangigli, D E Keyes, M D Smooke, 1994, Towards polyalgorithmic linear system solvers for nonlinear elliptic problems. *SIAM J Sci. Comput.,* **15**, 681–703.

[5] Y T Feng, 2006, On the discrete dynamic nature of the conjugate gradient method. *J comp. Phys.,* **211**, 91–98.

[6] J Chen, J Breslau, G Fu, S Jardin, W Park, 2006, New applications of dynamic relaxation in advanced scientific computing. Paper presented at ISICS'06, Dalian, China, 15–18 August.

[7] Y Saad, 1996, *shape Iterative methods for sparse linearsystems*, PWS Publishing Company.

[8] Owe Axelsson, 1996, *Iterative Solution Methods*, Cambridge University Press.

Table 1. nonlinear iterations

VR(4) for linear problem	VR(7) for linear problem
$\delta t = \frac{\delta x \delta y}{4} \frac{2}{\frac{\delta y}{\delta x} + \kappa_{\parallel} \frac{\delta x}{\delta y}} \cdot t_{scale}$	$\delta t = \frac{\sqrt{\delta x \delta y}}{\sqrt{2}} \frac{\sqrt{2}}{\sqrt{\frac{\delta y}{\delta x} + \kappa_{\parallel} \frac{\delta x}{\delta y}}} \cdot t_{scale}$ $\tau = \frac{\sqrt{L_x L_y}}{\sqrt{2}\pi} \frac{\sqrt{2}}{\sqrt{\frac{L_y}{L_x} + \kappa_{\parallel} \frac{L_x}{L_y}}} \cdot \tau_{scale}$
VR(4) for nonlinear problem	VR(7) for nonlinear problem
$\delta t_{ij}^k = \frac{\delta x \delta y}{4} \frac{2}{\frac{\delta y}{\delta x} + \kappa_{ij}^k \frac{\delta x}{\delta y}} \cdot t_{scale}$	$\delta t_{ij}^k = \frac{\sqrt{\delta x \delta y}}{\sqrt{2}} \frac{\sqrt{2}}{\sqrt{\frac{\delta y}{\delta x} + \kappa_{ij}^k \frac{\delta x}{\delta y}}} \cdot t_{scale}$ $\tau_{ij}^k = \frac{\sqrt{L_x L_y}}{\sqrt{2}\pi} \frac{\sqrt{2}}{\sqrt{\frac{L_y}{L_x} + \kappa_{ij}^k \frac{L_x}{L_y}}} \cdot \tau_{scale}$

Table 2. nonlinear iterations

$\kappa_{\|}$	1	2	3	4	5	6	7,8,9,$10^1 - -10^9$
N_{its}	diverge	1313	1310	1309	1309	1309	1308

Figure captions

Figure 1. The coefficient c is bounded and related to $\kappa_{||}$.

Figure 2. δt is related to δx and δy.

Figure 3. Linear solution at Nx=31, Ny=101.

Figure 4. Nonlinear solution at Nx=31, Ny=101, $t_{scale} = 0.41, \tau_{scale} = 0.87$.

VR(7) is stable when $t_{scale} \leq 0.41$; VR(4) is stable when $t_{scale} \leq 0.174$.

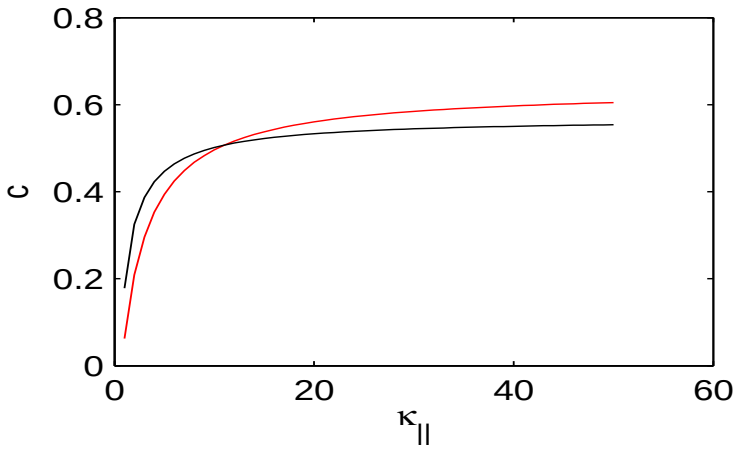

Figure 1. The coefficient c is bounded and related to $\kappa_{||}$.

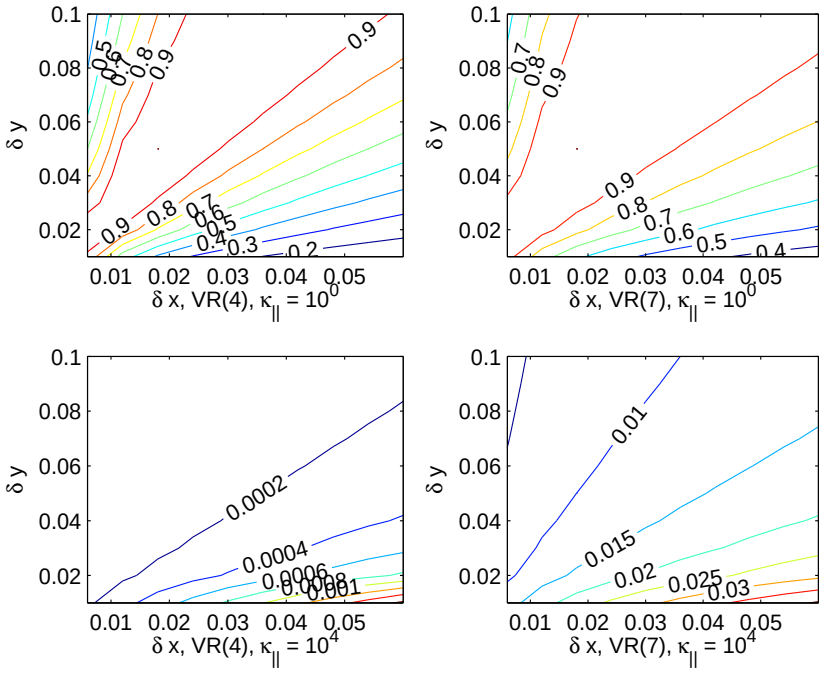

Figure 2. δt is related to δx and δy.

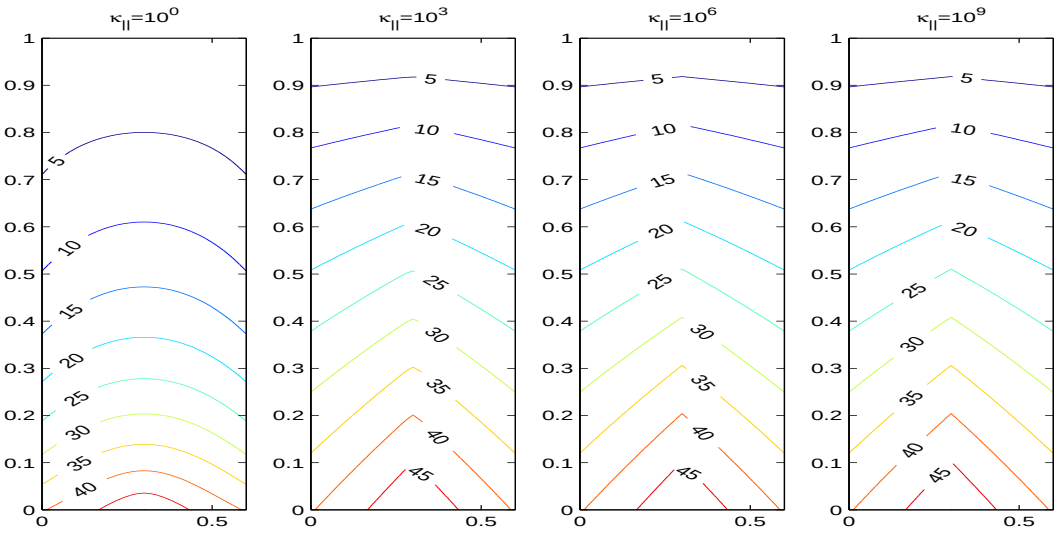

Figure 3. Linear solution at Nx=31, Ny=101.

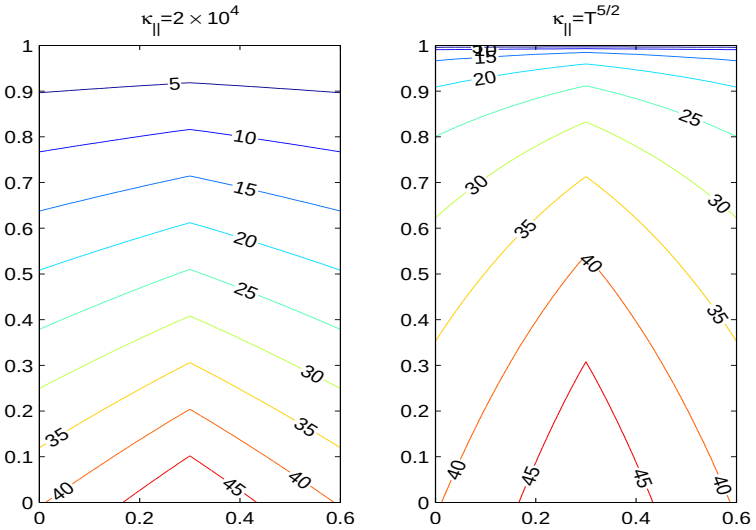

Figure 4. Nonlinear solution at Nx=31, Ny=101, $t_{scale}=0.41, \tau_{scale}=0.87$. VR(7) is stable when $t_{scale}\leq 0.41$; VR(4) is stable when $t_{scale}\leq 0.174$.

The Princeton Plasma Physics Laboratory is operated
by Princeton University under contract
with the U.S. Department of Energy.

Information Services
Princeton Plasma Physics Laboratory
P.O. Box 451
Princeton, NJ 08543

Phone: 609-243-2245
Fax: 609-243-2751
e-mail: pppl_info@pppl.gov
Internet Address: http://www.pppl.gov