Multi-CPU simulation of the tearing mode and \((m = 1, n = 1)\) internal resistive kink mode

S. Chatterjee\(^1\), M. P. Bora\(^1\), A. Sen\(^2\), and D. Chandra\(^2\)
\(^1\)Physics Department, Gauhati University, Guwahati, India
\(^2\)Institute for Plasma Research, Bhat, Gandhinagar, India
E-mail: samiran@gauhati.ac.in

Abstract. In this paper, we report the preliminary results of parallel simulation of the tearing mode and the \((m = 1, n = 1)\) internal resistive kink mode, which is believed to be the cause of the sawtooth oscillations observed in current carrying plasmas (tokamak plasmas). The simulation is carried with the help of a initial-value pure-spectral code NEAR involving the generalised reduced MHD equations (GRMHD) [4]. The parallelisation is done using MPI in the MPICH environment and carried out in two phases (i) with trivial parallelisation using a serial block-tridiagonal solver [6] and (ii) with a parallel block-tridiagonal solver [8]. We also report the simulation of the sawtooth crash using these algorithms.

1. Introduction
Neoclassical tearing modes (NTMs) are resistive tearing mode islands sustained by the helically perturbed bootstrap current [1]. In a toroidal plasma, poloidal asymmetry of the magnetic field causes two kinds of particle to exist — the passing particles and the trapped particles in the "banana" orbits, the diffusion and momentum exchange between which give rise to the bootstrap current. The NTM both degrade the plasma energy and momentum and can lead disruption of the plasma confinement. NTMs can put a serious constraint on the plasma beta which often provides the practical limit to the long-pulse (steady state), high confinement tokamak operation [2]. The NTM is linearly stable but nonlinearly unstable and requires a “seed” to develop to a tearing instability. The NEAR code [3] is an initial value code based on a set of reduced MHD (RMHD) equations which are derived using the (the ratio of parallel to perpendicular wave numbers) as a small expansion parameter [4]. The resultant equations are a set of RMHD equations known as the generalised RMHD (GRMHD) equations, valid at any aspect ratio. Besides, these equations also permit the incorporation of sub-Alfvénic equilibrium shear flows and neoclassical closures in a straightforward manner and are therefore well suited for studying the nonlinear evolution of NTMs in the presence of flows [5].

In a recent work involving the studies of NTMs using these GRMHD equations [5], the effect of equilibrium shear flow on the NTM is investigated, where the neoclassical effects are incorporated by using a bootstrap current term in the parallel Ohm’s law, based on a heuristic neoclassical closure [6], known as poloidal flow damping closure. Unsolved issues with the present NEAR code are (a) definitive demonstration of nonlinear saturation of the NTMs either with only the bootstrap current or with a full neoclassical closure for an otherwise classically stable tearing mode, (b) capability to include truly arbitrary equilibrium flow, (c) a proper
interface for various existing equilibrium solver codes such as JSOLVER from Princeton Plasma Physics Laboratory (PPPL) [7], (d) comparison of the scaling of the NTMs with existing theory, namely the Fitzpatrick threshold [8], and (e) a realistic on-line calculation of the classical tearing mode stability parameter from the equilibrium profile which can also serve as a discriminating parameter for the chosen equilibrium profile.

2. The generalized RMHD equations

Our numerical simulations are based on the solutions of a set of reduced MHD equations originally proposed by Kruger et al in 1998 [4]. These equations, which are valid at any aspect ratio, are derived using \( k/k_\perp \) (the ratio of parallel to perpendicular wave numbers) as a small expansion parameter and by employing a multiple scale analysis that respects equilibrium constraints and also permits elimination of fast time scales associated with perpendicular wave motion. The model equations thus evolve scalar potential quantities on a time scale associated with the parallel wave vector (shear-Alfvén wave time scale), which is the time scale of interest for resistive MHD instabilities like tearing modes. In addition, the model equations also permit the incorporation of sub-Alfvénic equilibrium shear flows and neoclassical closures in a straightforward manner and are therefore well suited for studying the nonlinear evolution of NTMs in the presence of flows. In the limit of \( \beta = \delta^{1/2} (\delta \ll 1) \), a simplified set of the evolution equations can be written as,

\[
\frac{\partial \Psi}{\partial t} - (b_0 + b_1) \cdot \nabla \phi_1 - b_1 \cdot \nabla \phi_0 = \eta J_\parallel - \frac{1}{ne} b_0 \cdot \nabla \cdot \Pi, \tag{1}
\]

\[
\nabla \cdot \left( \frac{\rho}{B_0} \frac{d}{dt} \frac{\nabla \phi_1}{B_0} \right) + (V_1 \cdot \nabla) \left[ \nabla \cdot \left( \frac{\rho}{B_0} \frac{\nabla \phi_0}{B_0} \right) \right] = (B_0 \cdot \nabla) \frac{J_\parallel}{B_0} + (B_1 \cdot \nabla) \frac{J_{T\parallel}}{B_0} \tag{2}
\]

\[
+ \nabla \cdot \left( \frac{B_0 \times \nabla p_1}{B_0^2} \right) + \nabla \cdot \left( \frac{B_0}{B_0^2} \times \nabla \cdot \Pi \right),
\]

\[
\frac{dp_1}{dt} + (V_1 \cdot \nabla) p_0 + \Gamma_{PT} \nabla \cdot \nabla V_1 = (\Gamma - 1) \left[ \eta J_{T\parallel}^2 \right] - \Pi : \nabla V + \Pi \cdot \nabla \left( \frac{J}{ne} \right) - \nabla \cdot q \tag{3}
\]

\[
\frac{dV_1}{dt} + (V_1 \cdot \nabla) V_{\parallel0} = -b_0 \cdot \nabla p_1 - b_1 \cdot \nabla p_T - b_0 \cdot \nabla \cdot \Pi \tag{4}
\]

\[
\nabla \cdot q = -\chi_{\parallel} \nabla^2 p_1 - (\chi_{\parallel} - \chi_{\perp}) (b_1 \cdot \nabla (b_0 \cdot \nabla p_0)) + b_0 \cdot \nabla (b_0 \cdot \nabla p_1) + b_1 \cdot \nabla (b_0 \cdot \nabla p_0)
\]

\[
+ b_0 \cdot \nabla (b_1 \cdot \nabla p_0) + b_1 \cdot \nabla (b_0 \cdot \nabla p_1),
\]

where

\[
\frac{d}{dt} \equiv \frac{\partial}{\partial t} + \nabla \cdot V, \quad b_T = b_0 + b_1 \quad = \frac{B_0 + B_1}{B_0} \tag{6}
\]

\[
J_{T\parallel} = J_{\parallel0} + J_{\parallel} \quad = b_0 \cdot \nabla \times B_0 + \nabla^2 \Psi, \tag{7}
\]

\[
V = \Omega(\psi) R^2 \nabla \zeta + V_1 \quad = \frac{B_0 \times \nabla \phi_0}{B_0^2} + V_{\parallel0} b_0
\]

\[
+ \frac{B_0 \times \nabla \phi_1}{B_0^2} + V_1 b_T.
\]

The Ohm’s law is represented by Eq.(1) where \( \Psi \) is the magnetic flux, \( \phi \) is the electrostatic potential, \( J \) is the current density, \( \eta \) is the resistivity, \( B \) is the magnetic field strength, \( n \) is the
number density, and $\Pi_e$ is the electron stress tensor. The vorticity and pressure equations are given by Eq.(2) and (3), where $V, \rho, q$, and $\Pi$ are the flow velocity, mass density, heat flux, and total stress tensor, respectively. Equation (4) is the parallel velocity equation. The subscripts 0, 1, and $T$ represent the equilibrium, perturbed, and total quantities. The equilibrium toroidal velocity is expressed through the function $\Omega(\psi)$ and is ordered so that $V_0/V_A \sim \epsilon \ll 1$, so that the flows are restricted to the sub-Alfvénic range.

3. The linear solver

The NEAR code is a toroidal initial-value code which uses a Fourier decomposition in the poloidal and toroidal directions and a finite-difference scheme in the radius-like flux coordinate. The spatial discretisation for the flux coordinate uses a three-point centered difference scheme with a variable mesh size for more grid resolution around the mode-rational surface (though this has not been utilised till date with all the NEAR-based codes). This discretisation can be briefly expressed as,

$$\vec{L} \cdot \frac{\partial \vec{x}}{\partial t} = \vec{R} \cdot \vec{x} + \vec{N}(\vec{x}), \quad (9)$$

where $\vec{L}$ and $\vec{R}$ are linear operators, $\vec{N}$ represents the nonlinear terms, and $\vec{x}$ are the field variables. The linear operators are a series of block tridiagonal matrices. The time difference scheme is given by,

$$\left( \vec{L} - \frac{1}{2} \Delta t \vec{R} \right) \cdot \vec{x}^{t+\Delta t} = \left( \vec{L} + \frac{1}{2} \Delta t \vec{R} \right) \cdot \vec{x}^t + \Delta t \vec{N} \cdot \vec{x}. \quad (10)$$

The linear portion is solved implicitly with a block tridiagonal solver and nonlinear terms are solved explicitly. Though the implicit nature of the linear solver allows large time steps (the characteristic time step is the shear-Alfvén time scale), the method is a pure spectral method where each of the convolution terms are evaluated with direct multiplications in the Fourier space [7], for which the computing time scales as $O(N^2)$ for the convolution of two $N$-point sequences. And the resultant block tridiagonal matrix requires inversion at every time step, which consumes most of the run-time (see Fig.1). As number of helical harmonics $h = m \theta + n \phi$ increases, which often is the case, the inversion process becomes less and less economic.

4. Parallel simulation

4.1. Requirement for parallelisation

A quick look at the run-time of the CPU utilisation of the present NEAR code is sufficient to highlight the need for parallelisation. The following figures (Fig.1) shows the utilisation of the CPU time in seconds for one time step (the resistive time scale) for a number of helical harmonics $h$.

As can be seen from the above figures, the convolution time with direct multiplication is almost linear with the number of harmonics. However, the time taken by the tridiagonal solver goes as square with the increased equilibrium (and evolved) harmonics but scales linearly with the perturbed harmonics. Both the convolution and solution time grows linearly with the radial mesh, which however can not be improved upon. These figures tell us that for a modest number of equilibrium harmonics of about 10 and an equally large number of perturbed harmonics with radial grid size of about 5000 mesh points, the CPU time is prohibitively large, which calls for a better equilibrium solver with a pseudo-spectral scheme and massive parallelisation.

4.2. Parallel tearing mode calculation

4.2.1. Trivial parallelisation

In this section we present the trivial parallelisation resulting using the classical block-tridiagonal inversion method [6]. The results of a tearing mode, parallel
Figure 1. The computing cost (CPU time) for a number helical harmonics and radial grid points. The red colour represents the convolution time and the green colour represents the time taken by the tridiagonal solver. The radial grid size is 500 for both the cases. These runs are made in an Intel Pentium-4 CPU (2.6 Ghz) with 256 MB RAM.

Figure 2. The nonlinear saturation of the energy of the 2/1 tearing mode and width of the 2/1 tearing mode.

simulation using four processors are shown in Fig.2. In Fig.2 we show the nonlinear saturation of the tearing mode in terms of energy and in terms of island width. All times are in the units of resistive time In these calculation, we have used the FORTRAN 90 compiler, ifort from Intel. The compilation options are -ftz -parallel -02 -132 -r8. We have used the MPI run-time environment through MPICH2.

In Fig.3, the growth rate, $\gamma$, of the tearing mode is shown, which falls to zero as the tearing mode saturates after the Rutherford regime due to nonlinear coupling modes. The (2/1) island at saturation is shown in the second panel of Fig.3. The parallel benchmarking is represented in Fig.4. It shows the relative times of calculation through serial and 4-threads parallel calculation as we increase the radial grid points. As can be seen from the figure that the efficiency of parallel calculation increases as we increase the grid points. In Fig.4, we show the relative times as the number of helical modes in the simulation increases.

4.2.2. Block-tridiagonal parallelisation Now, we present the results from adapting a parallel block-tridiagonal solver due to Garaud and Garaud [8]. The results are summarised in Fig.5. This solver has already been solved for scalability and stability.
Figure 3. The nonlinear saturation of the energy of the 2/1 tearing mode.

Figure 4. Relative run-time of 4-thread parallel calculation versus serial, plotted with radial grid points and against the number helical modes.

Figure 5. Benchmarking results from the parallel block-tridiagonal solver.
5. Simulation of the sawtooth
In this section we report the preliminary results of the triggering of the sawtooth through the 
\((m = 1, n = 1)\) internal resistive kink mode. We chose an equilibrium \(q\)-profile which starts 
below unity making the \((2/1)\) mode unstable to sawtooth perturbation. The study of sawtooth 
in the framework of NEAR is important as the sawtooth mechanisms is believed to be the one 
of the causes which may trigger an eventually unstable neoclassical mode. In Fig.6, we show 
the \(q\)-profile used for the sawtooth simulation and the evolution of the central pressure. The 
straight line in the second panel shows the evolution of the central tokamak pressure when 
the \(q\)-profile has been raised above unity. As expected, the pressure (and so the temperature) 
increases linearly with time. We note that more works are necessary for realistic simulation of 
the sawtooth.

6. Conclusion
In this work, we have reported the preliminary results of the multi-CPU parallel simulation of the 
tearing mode with the help of the spectral code NEAR. We have shown the results from trivial 
parallelisation with the classic block-tridiagonal solver and the parallelisation results from the 
parallel block-tridiagonal solver. The computing time required for the parallel block-tridiagonal 
solver scales almost linearly.

Acknowledgment
SC and MPB acknowledge the support through BRFST research project NFP/TH/SIM/01.

References
[1] La Haye R J 2006 Phys. Plasmas 13 55501
[2] La Haye R J et. al. 2000 Phys. Plasmas 7 3349
[3] Kruger S E 1999 Generalized Reduced Magnetohydrodynamic Equations (University of Wisconsin Center for 
Plasma Theory and Computation Report UW-CPTC 99-2, 1999)
[4] Kruger S E, Hegna C C, and Callen J D 1988 Phys. Plasmas 5 4169
[5] Chandra D, Sen A, Kaw P, Bora M P, and Kruger S 2005 Nucl. Fus. 45 524
[6] Hindmarsh A C 1977 Solution of Block-Tridiagonal Systems of Algebraic Equations (UCID- 30150, Lawrence 
Livermore National Laboratory, Livermore, California, 1977)
[7] Tajima T 2000 Computational Plasma Physics With Applications to Fusion and Astrophysics (Frontiers in 
Physics Series, Springer Verlag)
[8] Garaud P and Garaud J-D 2008 private communication