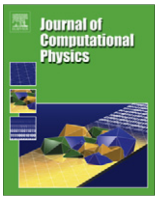
Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp



BCYCLIC: A parallel block tridiagonal matrix cyclic solver ☆

S.P. Hirshman *, K.S. Perumalla, V.E. Lynch, R. Sanchez

Oak Ridge National Laboratory, Oak Ridge, Tennessee 37830, USA

ARTICLE INFO

Article history:
Received 30 December 2009
Received in revised form 27 April 2010
Accepted 30 April 2010
Available online 13 May 2010

Keywords:
Cyclic reduction
Block matrix
Dense blocks
Tridiagonal matrix
Thomas algorithm
Parallel computing

ABSTRACT

A block tridiagonal matrix is factored with minimal fill-in using a cyclic reduction algorithm that is easily parallelized. Storage of the factored blocks allows the application of the inverse to multiple right-hand sides which may not be known at factorization time. Scalability with the number of block rows is achieved with cyclic reduction, while scalability with the block size is achieved using multithreaded routines (OpenMP, GotoBLAS) for block matrix manipulation. This dual scalability is a noteworthy feature of this new solver, as well as its ability to efficiently handle arbitrary (non-powers-of-2) block row and processor numbers. Comparison with a state-of-the art parallel sparse solver is presented. It is expected that this new solver will allow many physical applications to optimally use the parallel resources on current supercomputers. Example usage of the solver in magneto-hydrodynamic (MHD), three-dimensional equilibrium solvers for high-temperature fusion plasmas is cited.

1. Introduction

1.1. Motivation and scope

The solution of the matrix equation $Ax = b$, with possible multiple right-hand sides b, arises routinely in the solution of linear and nonlinear partial differential equations (PDEs). In three-dimensional systems where two coordinates are angular and one is radial, the Fourier-transformed PDEs result in A being block tridiagonal when the underlying equations involve at most second order derivatives. The block size M is the additive Fourier extents in the two angular directions, and the number of block rows N is the number of discrete radial nodes. Such matrix structure emerges naturally in numerical simulations of hot thermonuclear plasmas confined by a magnetic toroidal field, such as those in a tokamak or stellarator. The nonlinear fluid equations that describe the equilibrium, stability and evolution of these plasmas are often discretized spectrally in the two periodic directions, while finite differences are used in the radial direction. In this case the highest derivative in the radial direction is second order, the resulting linear sub-problems that emerge when solving the nonlinear equations via iterative Newton–Krylov methods exhibit a block tridiagonal structure.

Examples of physics codes which require efficient block solvers include the ideal magnetohydrodynamic (MHD) nonlinear solvers VMEC [1] and SIESTA [2], and the linear full-wave ion cyclotron radio frequency (ICRF) code TORIC [3]. All of these codes solve equations of the form $Ax = b$, with the block tridiagonal matrix A consisting of large, dense blocks. Usually

☆ This submission was sponsored by a contractor of the United States Government under contract DE-AC05-00OR22725 with the United States Department of Energy. The United States Government retains, and the publisher, by accepting this submission for publication, acknowledges that the United States Government retains, a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this submission, or allow others to do so, for United States Government purposes.

* Corresponding author. Tel.: +1 865 574 1289.
E-mail address: hirshmansp@ornl.gov (S.P. Hirshman).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.04.049

multiple right sides b occur. Such a block solver is essential for the numerical efficiency of the SIESTA [2] code which computes high-resolution MHD equilibria in the presence of magnetic islands. Most of the computational time in SIESTA is spent in the inversion of large block matrices ($N > 100$, $M > 300$) which are repeatedly applied as preconditioners to accelerate the convergence of the linearized MHD equations toward an equilibrium state. Eventual simulations for plasmas in the International Thermonuclear Experimental Reactor (ITER, with $T \sim 15$ keV, $a \sim 2$ m, $B \sim 5$ T) will require even larger spatial resolution resulting in greater block row numbers and sizes. Therefore the efficient parallel inversion and storage of the factors of A is essential in SIESTA and was the primary motivation for the present code development.

Fig. 1 shows a SIESTA equilibrium calculation for pressure contours (magnetic surfaces) in a tokamak plasma with an internal $q = 1$ surface near the center of the plasma. The block size for this problem was $M = 273$ and there were $N = 101$ block rows. For this small sized problem, the serial calculation can be done in under 5 min on a desktop computer using Compaq Visual Fortran. Using ScaLAPACK (see below) to factor and invert the blocks improves the performance for this problem by about a factor of 5 before communication bottlenecks saturate the performance gain. As we discuss in Section 4, the new cyclic reduction routine BCYCLIC significantly extends the scalability for this, and larger, problem sizes.

1.2. Related work in parallel block tridiagonal matrix solvers

There has been considerable work in developing solution algorithms for tridiagonal matrices. Here, we briefly review work related to code development for block tridiagonal solvers. For a block tridiagonal matrix A, it is possible to obtain an exact inverse (direct solution) with no fill-in by using the well-known Thomas [4] *serial*algorithm, which is easily generalized for block sizes $M \gg 1$. While this is the fastest algorithm on a serial computer [5], it is not parallelizable since each solution step in the algorithm depends on the preceding one.

Many authors have considered efficient parallel block solvers for scalar block ($M = 1$) matrices based on cyclic reduction [6–8]. Cyclic reduction was first described by Heller [9] for block tridiagonal systems, although an efficient parallel code was not described. The new code BCYCLIC described here fills a software gap in the available codes for solving tridiagonal systems

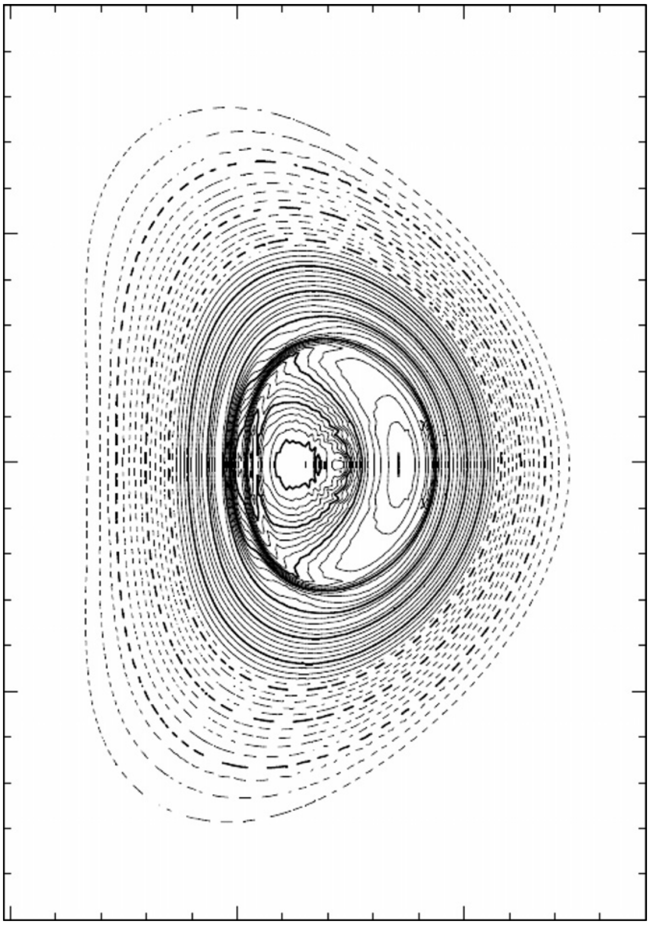


Fig. 1. Example of a tokamak equilibrium plasma with a $q = 1$ tearing mode creating an $m = 1$ island.