63

Capacitance Matrix Preconditioning

Karina Terekhova

1 Introduction

Iterative methods, widely used for solutions of large linear systems require preconditioning as an essential part. The advent of parallel computers motivates a search for preconditioners suitable for parallel processing.

A domain decomposition approach can satisfy this demand. The domain of definition of the problem is partitioned into subdomains, and the original problem is substituted by an equivalent one, defined on the internal boundaries (interfaces) separating the subdomains. This smaller problem is solved by an iterative method, usually with the help of preconditioning to accelerate the convergence. A preconditioner in this case must be an easily invertible approximation to the interface operator, also called the capacitance matrix, or the Schur complement.

A good approximation to the Schur complement of a linear system can be constructed algebraically by investigating its numerical structure. This idea was introduced by M. Dryja [Dry82] and developed in a paper by G. Golub and D. Mayers [GM83] that referred to the symmetric 2D case. This paper shows how the underlying reasoning can be extended to design a similar preconditioner for other elliptic problems.

2 Problems in Two Dimensions

Let us consider a symmetric model problem $\Delta u = f$ defined on a rectangular region subdivided into two rectangular parts with homogeneous Dirichlet conditions imposed on the outer boundary. The Schur complement S is then symmetric and positive definite. To solve the equation for S efficiently we need the preconditioning matrix M to be close to S and especially for the eigenvalues of $M^{-1}S$ to be clustered as closely as possible.

Examination of the Schur complements in some particular cases shows that the elements of S are dependent mainly on the distance from the diagonal, $\mid i - j \mid$, with

Ninth International Conference on Domain Decomposition Methods
Editor Petter E. Bjørstad, Magne S. Espedal and David E. Keyes

the largest element on the diagonal, the elements decreasing quite rapidly as $\mid i-j \mid$ increases.

This suggests that a useful approximation to S may be found by letting the boundaries of the two subdomains move to infinity. Then the setting is: find the solutions of Laplace's equation in the two half-planes, the solution being required to vanish at infinity and also at all points on the dividing axis, except at the origin, where it is equal to one.

Denoting by r and s the Cartesian indices along and normal to the interface in a uniform two-dimensional grid we have

$$\begin{cases} u_{r,s-1} + u_{r,s+1} + u_{r+1,s} + u_{r-1,s} - 4\,u_{r,s} = 0 \\ u_{r,s} \to 0 \text{ as } r \to \pm\infty, s \to \infty \\ u_{r,0} = 0 \ (r \neq 0) \\ u_{0,0} = 1 \end{cases}$$

Defining the generating function

$$\phi_s(t) = \sum_{r=-\infty}^{\infty} t^r u_{r,s}$$

we obtain the solution from the characteristic equation

$$\phi_s(t) = \left[2 - \frac{1}{2}\left(t + \frac{1}{t}\right) - \left(\left\{ 2 - \frac{1}{2}\left(t + \frac{1}{t}\right) \right\}^2 - 1 \right)^{1/2} \right]^s$$

The residuals at the grid points on the axis are given by

$$\rho_r = u_{r-1,0} + u_{r+1,0} + 2\,u_{r,1} - 4\,u_{r,0}$$

for which the generating function is

$$\psi(t) = \left(t + \frac{1}{t} - 4\right)\ \phi_0 + 2\,\phi_1 = -2\ \left\{ \left[2 - \frac{1}{2}\left(t + \frac{1}{t}\right) \right]^2 - 1 \right\}^{1/2}$$

We then expand ψ in positive and negative powers of t to obtain ρ_r which is the coefficient of t^r.

$$\begin{aligned} \rho_r &= \frac{1}{2\pi} \int_{-\pi}^{\pi} -2\cos r\theta\,[(2-\cos\theta)^2 - 1]^{1/2} d\theta \\ &= -\frac{4}{\pi} \int_0^{\pi} \cos 2k\alpha\, \sin\alpha\,[1 + \sin^2\alpha]^{1/2} d\alpha \end{aligned}$$

A possible preconditioner then is $M_{ij}^{(1)} = \rho_{|i-j|}$. Full details can be found in [GM83].

The described method can be applied with some changes to a convection-diffusion equation of the form

$$-\varepsilon\,\Delta\,u + a\,u_x + b\,u_y = f,$$

where ε, a and b are constants with $a \geq 0, b \geq 0, \varepsilon > 0$, and where ε may be small compared with a and b. As before, the problem is defined on a rectangular region subdivided into two rectangular parts. The Schur complement is, of course, unsymmetric, but the dependence of its elements on the distance from the diagonal is still quite clear. This justifies approximation of the problem with the boundaries of the region moving away to infinity, just as in the symmetric case. Upon upwind finite differencing the system to solve is

$$\begin{cases} u_{r,s+1} + u_{r+1,s} + A\, u_{r-1,s} + B\, u_{r,s-1} - C\, u_{r,s} = 0 \\ u_{r,s} \to 0 \text{ as } r \to \pm\infty, s \to \infty \\ u_{r,0} = 0 \ (r \neq 0) \\ u_{0,0} = 1 \end{cases}$$

where $A = 1 + ah/\varepsilon$, $B = 1 + bh/\varepsilon$, $C = 2 + A + B$.

Again, ρ_r is the coefficient of t^r and the residuals on the axis are finally

$$\rho_r = \frac{1}{2\pi} \int_{-\pi}^{\pi} \cos r\theta \left[\left(2\sqrt{A} \cos\theta - C \right)^2 - 4B \right]^{1/2} d\theta$$

so the preconditioner should be defined as

$$M_{ij} = A^{-\frac{i-j}{2}} \rho_{|i-j|}.$$

Note that the preconditioner is unsymmetric and its elements grow rapidly above the diagonal and decrease under it.

As the preconditioner contains an exponential quantity $A^{-\frac{i-j}{2}}$ which depends on the parameter a of the problem, evaluation of the solution far from the interface may cause floating point precision loss. A sufficient number of interfaces and a suitable choice of the grid size h should be used in order to avoid this problem.

3 Problems in Three Dimensions

Let us consider a model problem $\Delta u = f$ defined on a cube subdivided into two parts by a plane.

The reasoning for the two-dimensional model problem applies to the three-dimensional one if the relevant changes are made to the formulation of the discretised infinite problem. Thus, after the outer boundaries have moved to infinity, we have

$$\begin{cases} u_{r+1,s,t} + u_{r-1,s,t} + u_{r,s-1,t} + u_{r,s+1,t} + \\ u_{r,s,t+1} + u_{r,s,t-1} - 6\, u_{r,s,t} = 0 \\ u_{r,s,t} \to 0 \text{ as } r \to \pm\infty, s \to \pm\infty, t \to \infty \\ u_{r,s,0} = 0 \ (r, s \neq 0) \\ u_{0,0,0} = 1 \end{cases}$$

We solve the discretised problem for the generating functions, and then separate the desired residuals as coefficients of double Fourier series. The residuals obtained are

$$\rho_{r,s} = -\frac{8}{\pi^2} \int_0^{\pi} \int_0^{\pi} \cos r\alpha \, \cos s\beta \, ([3 - \cos\alpha - \cos\beta]^2 - 1)^{1/2} d\alpha d\beta$$

These residuals, ordered with respect to the ordering of variables in the original problem, form the preconditioner. It can be dense if all residuals are used, or it can take block-diagonal or banded forms if we substitute the residuals which are close to zero in some sense by zeroes. This does not usually cause the loss of convergence properties and gives the obvious advantage of easy inversion.

4 Results

In this section we discuss the practical aspects of applying the capacitance matrix preconditioner, serially and in parallel, to model and real-life test problems in two and three dimensions. All industrial examples were supplied by Elf Geoscience Research Centre. They were obtained from convection-diffusion equations modelling the process of oil recovery.

Methods

The particular iterative method used in the numerical experiments is BiCGSTAB, proposed by H. van der Vorst (see [vdV92]). In it two solves of the subproblems and two applications of the preconditioner are required per iteration. The preconditioner is calculated and inverted in advance, so its application is computationally cheap. The most time-consuming operation is, therefore, solving the subproblems. Exact solvers (Gaussian elimination), direct solvers (for example, fast Fourier transform solver) and various iterative techniques are proposed in the literature for this purpose.

We used the ORTHOMIN algorithm with the nested factorisation preconditioner as a solver for subproblems in three dimensions. ORTHOMIN is an optimal and minimal conjugate-gradient-like algorithm showing fast reliable convergence at the expense of relatively high storage requirements.

Many problems relevant for the industrial applications take block diagonal form after discretisation. This particular structure of their matrices can be exploited to achieve efficient solution of subproblems. The algorithm of nested factorisation preconditioning, although not easily adapted to deal with general sparse matrices, is particularly good for block tridiagonal ones. The algorithms of ORTHOMIN and of recursive evaluation of the nested factorisation preconditioner is given in [ACP81].

The nested factorisation preconditioner can hardly be parallelised without considerable loss of efficiency because of data interdependence. However, ORTHOMIN with the nested factorisation preconditioning make a fast and predictable serial solver of the 3D subproblems in a parallel iterative solution process preconditioned by the capacitance matrix preconditioner.

Parallel Model

The parallel program was written following the bulk-synchronous parallel (BSP) paradigm.

The BSP model, introduced by L. Valiant in 1990 ([Val89]), implements the idea of portable parallel software. A BSP computation consists of a number of asynchronous supersteps during which the processors can issue requests for non-local read or write

Table 1 2D: Nonsymmetric model problem

		Direct solver	Neumann preconditioner	New preconditioner
32×32	iterations	n/a	22	3
	time	75	126	105
	Mflops	2.4	7.9	4.0
64×64	iterations	n/a	39	3
	time	1259	1727	877
	Mflops	40.1	114.2	35.8

Table 2 3D: Nonsymmetric model problem of size $44 \times 17 \times 14$ solved serially

Subdomains	time	iterations
2	80	5
4	82	7

operations. Each superstep is followed by a synchronisation session which ensures that all information exchange is completed.

The total cost of a BSP computation can be expressed in terms of separate computation, communication and synchronisation costs, combined with the parameters of the computer reflecting its performance in computation, communication and synchronisation. Variation of these costs with the change in the number of processors is predictable, as well as the performance of a particular parallel computer running a given algorithm.

We have obtained the portable cost estimates for the capacitance matrix preconditioner in combination with BiCGSTAB. The computational cost of one iteration of the proposed algorithm is

$$\begin{array}{ll} \text{3D:} & O(\frac{n}{p} + gn^{2/3} + l) \\ \text{2D:} & O(\frac{n}{p} + gn^{1/2} + l) \end{array}$$

where n is the number of grid points, p is the number of processors, g characterises the communication throughput and l is the synchronisation latency of the parallel computer.

The cost of an iteration of unpreconditioned BiCGSTAB is of the same order of magnitude. This means that the capacitance matrix preconditioner increases the cost of a BiSGCTAB iteration only by a constant factor. The great reduction in the number of iterations justifies the small extra cost of preconditioning.

Test Cases

The preconditioner was tried on model and real-life examples, both in two and three dimensions. Two-dimensional examples were solved serially, examples in three

Table 3 3D: Model problem of size $46 \times 19 \times 5$ solved in parallel

Processors	Serial (time)	BSP (time)	iterations
2	59	31	3
4	61	18	3

dimensions serially and in parallel.

The serial programs were run on a SUN SPARC workstation; the parallel program was run on several workstations connected via a network. All programs were written in Fortran 77 using double precision arithmetic.

The model problems in two dimensions were derived from the convection-diffusion equations of the form described in Section 2. The coefficients a and b varied in the range $[0, 5]$ with ε a constant varying between 10^{-7} and 1 in different runs. The results presented in Table 1 are for an typical case with $a = 2$, $b = 1$, $\varepsilon = 10^{-6}$. Reduction in residuals by nine orders of magnitude was achieved in each run.

Table 2 contains the number of iterations and runtimes in seconds for a three-dimensional model problem solved sequentially.

Table 3 contains the results of solving a typical model problem in parallel.

5 Conclusions

The capacitance matrix preconditioner for nonsymmetric matrices described in this paper has shown encouraging results in comparison with the established way of preconditioning. It possesses good scalability and convergence properties and can be applied to the problems with variable coefficients as well as constant ones.

The new method of preconditioning was designed and tested on problems in two and three dimensions discretised with the five-point upwinding on a regular grid. It is specifically intended for use in parallel computation and it has proved to possess two important qualities — naturally decoupled structure and closeness to the interface operator, resulting in accelerated convergence in many important industrial problems.

REFERENCES

[ACP81] Appleyard J. R., Cheshire I. M., and Pollard R. K. (1981) Special techniques for fully-implicit simulators. In *Proc. European Symposium on Enhanced Oil Recovery Bournemouth, UK*, pages 395–408.

[Dry82] Dryja M. (1982) A capacitance matrix method for Dirichlet problem on polygon region. *Numer. Math.* 39: 51–64.

[GM83] Golub G. H. and Mayers D. (1983) The use of preconditioning over irregular regions. In *Proc. 6th Internl. Conf. Comput. Meth. Sci. Engng. Versailles, France.*

[Val89] Valiant L. G. (August 1989) A bridging model for parallel computation. *Communications of the ACM* 33(8): 103–111.

[vdV92] van der Vorst H. A. (March 1992) BiCGStab: A fast and smoothly converging variant of BiCG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Stat. Comp.* 13(2): 631–644.