The Schur complement method and solution of large-scale geophysical problems

Vladislav Aleksandrov, Henri Samuel

BGI Bayerisches Geoinstitut

Motivation

The main purpose of our work is to develop a program for solution of large sparse systems, arising in elliptic PDE treated by finite differences. The presented method proves to be a simple and efficient approach to the problem while pertaining many advantages. One of these is the parallelism of the algorithm and its robustness (compared to iterative methods). Another advantage is the possibility to solve the problem with less allocated memory while still relatively fast compared to the best direct solvers.

Introduction

The Schur complement domain decomposition is a method used for solution of large linear systems, arising in boundary-value problems for PDE solved by finite differences or finite elements. The algorithm is based on subdivision of the domain into smaller ones and reordering of the nodes within the domains. It achieves parallelism with large granularity and almost equal load balance. The method is introduced by Przemieniecki [1] and currently it is used to solve applied problems from fluid mechanics, magnetohydrodynamics, and other areas.

Algorithm

Consider the Laplace equation in a rectangular domain.

$$\Delta u = 0$$

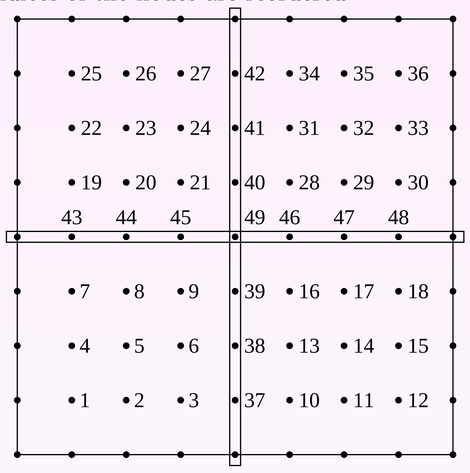
One can discretize the above equation using finite differences: five point stencil in two dimensions

$$u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j} = 0$$

or seven point stencil in three dimensions

$$u_{i+1,j,k} + u_{i-1,j,k} + u_{i,j+1,k} + u_{i,j-1,k} + u_{i,j,k+1} + u_{i,j,k-1} - 6u_{i,j,k} = 0$$

This implies that only the nearest neighbor nodes are related to each other. The domain is divided into non-overlapping subdomains and interface. The indices of the nodes are reordered



Using this ordering, the linear system is transformed to the form

$$\begin{bmatrix} A_{0,0} & 0 & 0 & 0 & \dots & 0 & A_{0,p} \\ 0 & A_{1,1} & 0 & 0 & \dots & 0 & A_{1,p} \\ 0 & 0 & A_{2,2} & 0 & \dots & 0 & A_{2,p} \\ \vdots & \vdots & \vdots & \ddots & & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & A_{p-1,p-1} & A_{p-1,p} \\ A_{p,0} & A_{p,1} & A_{p,2} & A_{p,3} & \dots & A_{p,p-1} & A_{p,p} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{p-1} \\ x_p \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_{p-1} \\ b_p \end{bmatrix}$$

where p is the number of subdomains, $A_{0,0}, A_{1,1}, \dots, A_{p-1,p-1}$ are square matrices of order equal to the number of nodes of the respective domain and $A_{p,p}$ is a square matrix of order equal to the number of nodes of the interface.

Block Gaussian elimination

$$A'_{p,p} = A_{p,p} - \sum_{i=0}^{p-1} A_{p,i} A_{i,i}^{-1} A_{i,p}$$

$$b'_p = b_p - \sum_{i=0}^{p-1} A_{p,i} A_{i,i}^{-1} b_i$$

Equation for the interface unknowns x_p

$$A'_{p,p} x_p = b'_p$$

$A'_{p,p}$ is called Schur matrix.

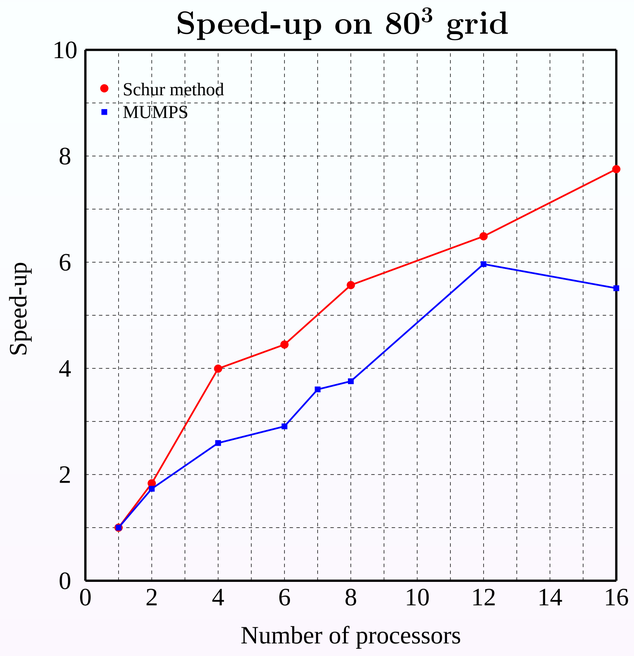
Back-substitution for $i = 0, 1, 2, \dots, p-1$

$$A_{i,i} x_i = b_i - A_{i,p} x_p$$

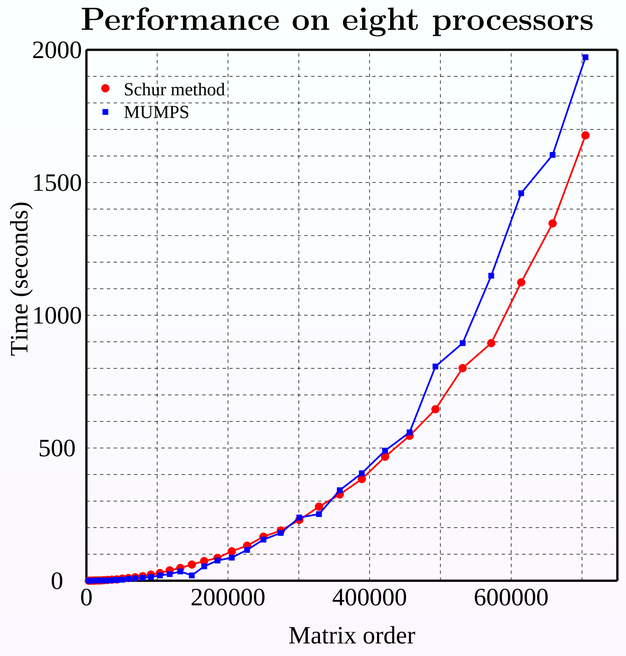
Numerical results

The three dimensional Laplace equation is used to test the performance of the Schur domain decomposition method. The Schur complement for each subdomain is computed with the MUMPS library on distributed memory computer. The Schur interface equation is solved by ScaLAPACK and its benchmark is presented. The solution time for the problem with our program is compared to the time required for MUMPS to solve the global problem. Four subdomains are used to demonstrate results for the speed-up and the timing on 80^3 domain. Test cases with two and three subdomains were also performed but their timings are worse.

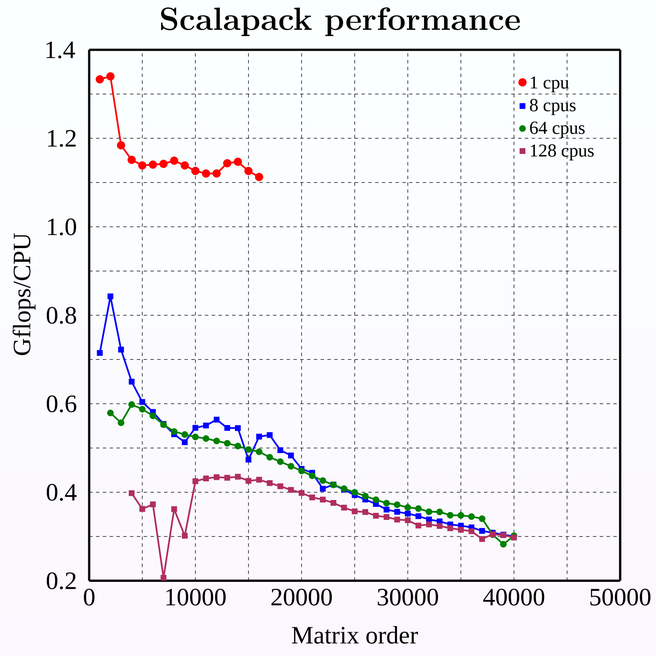
Speed-up on 80^3 grid



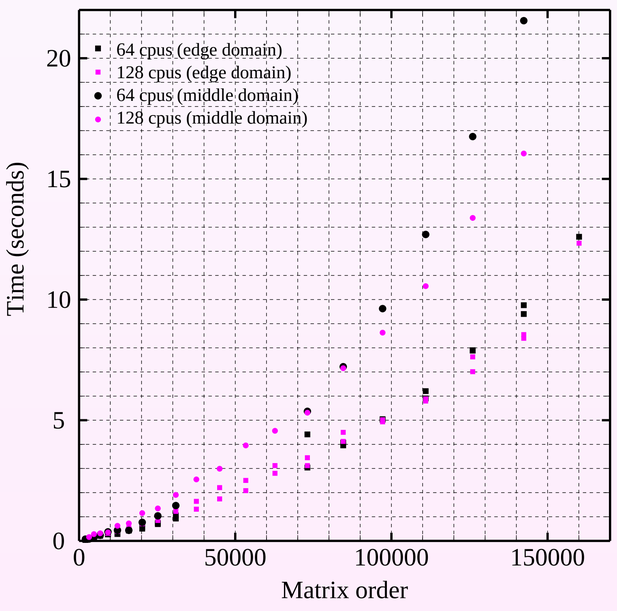
Performance on eight processors



Scalapack performance



Schur complement on three subdomains



The advantage of dividing the global domain into large number of subdomains is that the sparsity of the Schur matrix increases. When the domain is divided into $\frac{N}{2}$ subdomains (N is an even number equal to the resolution in each direction) then the order of the Schur matrix $A_{p,p}$ is equal to $\left(\frac{N}{2} - 1\right)(N-1)^2$.

Conclusion

The Schur complement domain decomposition method demonstates good performance compared to the direct multifrontal solver MUMPS. The algorithm consists of three steps which are straightforward to implement. The advantage of the method is the possibility to estimate the necessary memory for each step and the time required for its execution under different test cases. The solution time is minimal when four subdomains are used but the required memory is optimal when the number of the subdomains is increased. For example, the use of eight subdomains will decrease the time and the allocated memory during the calculation phase of the Schur matrix but the time for its inversion will increase about 4 times when executed in parallel. The overall performance depends also on the communication between the processes. Because of the optimized memory usage the method is a reliable technique for solution of large problems.

Future work

The program can be improved by the use of iterative methods to solve the equation for the interface unknowns. Simple test shows that on one processor the conjugate gradient method solves the Schur system 3 times faster than Scalapack on 80^3 grid. Reordering with eight subdomains will allow to solve larger problems.

References

[1] J.S. Przemieniecki, "Matrix structural analysis of substructures", *AIAA J.* 1, (1963) 138-147.

[2] S. Kocak, H.U. Akay "Parallel Schur complement method for large-scale systems on distributed memory computers", *Appl. Math. Modelling* 25, (2001) 873-886.

[3] Vanderstraeten, D. and Keunings, R, "A parallel solver based on the dual Schur decomposition of general finite element matrices", *International Journal for Numerical Methods in Fluids* 28, (1998) 23-46

[4] L. Mansfield, "On the Conjugate Gradient Solution of the Schur Complement System Obtained from Domain Decomposition", *SIAM on Num. Analysis* 27-6, (1990) 1612-1620

[5] , *ScaLAPACK Users' Guide*, Society for Industrial and Applied Mathematics, 1977.