Numer. Math. Theor. Meth. Appl.
doi: 10.4208/nmtma.2010.33.5

Vol. **3**, No. 3, pp. 352-366
August 2010

Preconditioning Schur Complement Systems of Highly-Indefinite Linear Systems for a Parallel Hybrid Solver[†]

I. Yamazaki[*], X. S. Li and E. G. Ng

Lawrence Berkeley National Laboratory, Berkeley, California, USA.

Received 30 September 2009; Accepted (in revised version) 27 January 2010

Available online 2 July 2010

Abstract. A parallel hybrid linear solver based on the Schur complement method has the potential to balance the robustness of direct solvers with the efficiency of preconditioned iterative solvers. However, when solving large-scale highly-indefinite linear systems, this hybrid solver often suffers from either slow convergence or large memory requirements to solve the Schur complement systems. To overcome this challenge, we in this paper discuss techniques to preprocess the Schur complement systems in parallel. Numerical results of solving large-scale highly-indefinite linear systems from various applications demonstrate that these techniques improve the reliability and performance of the hybrid solver and enable efficient solutions of these linear systems on hundreds of processors, which was previously infeasible using existing state-of-the-art solvers.

AMS subject classifications: 65F10, 15A12, 65N55

Key words: Schur complement method, preconditioning, matrix preprocessing.

1. Introduction

A number of parallel linear solvers have been implemented based on a domain decomposition idea called the Schur complement method [7, 8]. In this method, the unknowns in interior subdomains are first eliminated using a direct solver. Then, the remaining Schur complement system is solved using a preconditioned iterative method. This method often exhibits great parallel performance because interior subdomains can be solved in parallel. Furthermore, this *hybrid* approach has the potential to balance the robustness of direct solvers with the efficiency of iterative solvers because the unknowns in the relatively-small interior subdomains can be eliminated efficiently using a direct solver, while the sparsity can be enforced for solving the Schur complement system, where most of the fill occurs. In

[†]This paper is presented at the International Conference on Preconditioning Techniques for Scientific and Industrial Applications, August 24-26, 2009, Hong Kong.

[*]Corresponding author. *Email addresses:* `ic.yamazaki@gmail.com` (I. Yamazaki), `XSLi@lbl.gov` (X. S. Li), `EGNg@lbl.gov` (E. G. Ng)

addition, for a symmetric positive definite system, the Schur complement has a smaller condition number than the original coefficient matrix [20, Section 4.2], and fewer iterations are often required for solving the Schur complement system. Unfortunately, for a highly-indefinite linear system, the preconditioned iterative method often suffers from either slow convergence or large memory requirement to solve the Schur complement system.

To address this challenge, we discuss in this paper techniques to preprocess the Schur complement systems in parallel. Effective preprocessing techniques have been already developed to solve highly-indefinite linear systems of equations. For example, a matrix permutation to preserve the sparsity of a preconditioner significantly reduces the computational and memory requirements [11, 13]. Furthermore, unsymmetric scaling and permutations that place large entries on the diagonal often improve the reliability and performance of the preconditioned iterative solver [3, 6]. However, the effectiveness of these preprocessing techniques on the performance of a *parallel* hybrid solver has not been well studied. There are software packages which compute matrix permutations in order to preserve the sparsity of the preconditioners on a distributed memory system [4, 12]. However, these packages are designed primarily for sparse matrices, and their performance suffers on the Schur complements, which are relatively dense. Furthermore, a robust parallel implementation to place large entries on the diagonal has not yet been developed [5, 9, 18]. The primary purpose of this paper is to fill this gap.

The rest of this paper is organized as follows: In Section 2 we review the Schur complement method. In Section 3 we discuss the techniques to preprocess the Schur complements in parallel. Then, in Section 4 we present numerical results to demonstrate the effectiveness of the preprocessing techniques for solving the Schur complement systems of large-scale highly-indefinite linear systems of equations. We also present numerical results to demonstrate that our new parallel hybrid linear solver incorporates these preprocessing techniques and efficiently solves these linear systems on a large number of processors. Finally, in Section 5 we conclude with final remarks.

2. Schur complement method

The Schur complement method is a non-overlapping domain decomposition method, which is also referred to as iterative substructuring. Specifically, the original linear system is first reordered into a 2×2 block system of the following form:

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \tag{2.1}$$

where A_{11} is a block-diagonal matrix, each of whose diagonal blocks represents an *interior subdomain*, A_{22} represents *separators*, and A_{12} and A_{21} are the *interfaces* between A_{11} and A_{22}. After one step of the block Gaussian elimination, the 2×2 block system (2.1) becomes

$$\begin{pmatrix} A_{11} & A_{12} \\ 0 & S \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ \widehat{b}_2 \end{pmatrix}, \tag{2.2}$$