Computers and Mathematics with Applications 70 (2015) 2331–2341

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

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

Impact of element-level static condensation on iterative solver performance

D. Pardo [a,b,c,*], J. Álvarez-Aramberri [a,b,d], M. Paszynski [e], L. Dalcin [f,g], V.M. Calo [g,h]

[a] *Department of Applied Mathematics, Statistics, and Operational Research, University of the Basque Country (UPV/EHU), Bilbao, Spain*
[b] *Basque Center for Applied Mathematics (BCAM), Bilbao, Spain*
[c] *IKERBASQUE, Basque Foundation for Science, Bilbao, Spain*
[d] *University of Pau (UPPA), France*
[e] *AGH University of Science and Technology, Krakow, Poland*
[f] *National Scientific and Technical Research Council, Argentina*
[g] *Center for Numerical Porous Media, King Abdullah University of Science and Technology, Saudi Arabia*
[h] *Applied Mathematics & Computational Science and Earth Science & Engineering, King Abdullah University of Science and Technology, Saudi Arabia*

ARTICLE INFO

Article history:
Received 30 October 2014
Received in revised form 24 August 2015
Accepted 5 September 2015
Available online 2 October 2015

Keywords:
Static condensation
Finite element method
p-FEM
Iterative solvers

ABSTRACT

This paper provides theoretical estimates that quantify and clarify the savings associated to the use of element-level static condensation as a first step of an iterative solver. These estimates are verified numerically. The numerical evidence shows that static condensation at the element level is beneficial for higher-order methods. For lower-order methods or when the number of iterations required for convergence is low, the setup cost of the elimination as well as its implementation may offset the benefits obtained during the iteration process. However, as the iteration count (e.g., above 50) or the polynomial order (e.g., above cubics) grows, the benefits of element-level static condensation are significant.

1. Introduction

Within the Finite Element (FE) community, the term static condensation of interior degrees of freedom refers to the Gaussian elimination of the element interior *bubble functions* arising from high-order discretizations [1]. Other terms such as Guyan condensation (reduction) can also be found in the literature to refer to the same set of linear algebra operations [2]. Static condensation can also be interpreted as a partial LU factorization of the interior degrees of freedom, as a first step of a specific substructuring technique, or as a partial orthogonalization of basis functions [3].

Interpreted in any of these forms, static condensation constitutes a fundamental building block for direct solvers and delivers significant performance improvements [4–6]. In high-order methods such as the *p*- and *hp*-FE methods, interior degrees of freedom are eliminated first, leading to a reduced system (called Schur complement) that is subsequently LU factorized. This static condensation step ensures the elimination of interior degrees of freedom before starting the LU factorization of the skeleton problem, thereby providing often better performance than that achieved with traditional ordering techniques, as shown in [7]. It also explains why those matrices lacking a structure that enables static condensation

* Corresponding author at: Department of Applied Mathematics, Statistics, and Operational Research, University of the Basque Country (UPV/EHU), Bilbao, Spain.
E-mail address: dzubiaur@gmail.com (D. Pardo).

http://dx.doi.org/10.1016/j.camwa.2015.09.005
0898-1221/

(e.g., higher-continuous basis functions as those used in meshless methods [8], reconstructing kernel element methods [9], and isogeometric analysis [10]) typically require a significantly larger number of floating point operations in order to be factorized [5].

While the use of static condensation in direct solvers is always beneficial [11], its use with iterative solvers is more controversial. Some authors postulate that static condensation should always be the starting point of any iterative solver, while others refrain from doing so, since it adds some complexity to the implementation. Even when static condensation is unused, most iterative solvers for *p*- and *hp*-FE methods still perform some type of elimination (or a spectrally equivalent operation) of local interior bubble functions (cf., [12–17]).

The key point, however, is to determine how profitable it is to explicitly build the Schur complement of element bubble functions (as performed in static condensation) and eliminate the corresponding unknowns from the global system before performing iterations with respect to keeping the local LU-factorized matrices as part of the preconditioner without ever computing the Schur complement. In other words, the distinguishing feature between iterative solvers that employ partial LU factorizations versus those that perform static condensation before executing an iterative solver is that the latter explicitly build the Schur complement and eliminate interior bubble functions from the global matrix rather than only evaluating their action over a given residual.

This paper provides quantitative estimates about the profitability of using static condensation before employing an iterative solver. We corroborate these estimates with numerical experiments in two and three spatial dimensions. Numerical experimentation also enlightens the behavior on the pre-asymptotic regime. As a result, we describe those situations in which the use of static condensation is most beneficial. To quantitatively compare both methods, we present floating point operations (FLOPs) estimates that also provide interesting clues for the design of optimal hybrid solvers [18].

In order to make this analysis tractable and easy to follow, we make several assumptions, which are described in Section 2 along with our model problem. Section 3 presents precise theoretical complexity estimates illustrating the advantages and limitations of using static condensation for each particular discretization. We describe the implementation details in Section 4 and we present numerical results confirming the estimates in Section 5. Section 6 describes the conclusions of our study and suggests future research lines in the topic.

2. Model problem and assumptions

Our starting point is the following algebraic system of linear equations:

$$Ax = b, \tag{1}$$

where A is a non-singular real-valued $N \times N$ sparse matrix, b is the right-hand side, and x is the solution vector.

In this work, we assume that the system matrix A is associated to a regular quadrilateral or hexahedral grid coming from a finite element discretization with uniform order of approximation p and with the same number of elements in each spatial direction. When the number of elements in each direction is substantially different, then the problem complexity reduces to that given by a lower dimensional problem.

In our estimates and computations, we avoid taking advantage of orthogonal basis functions, i.e., we consider all contributions originating from a trial and a test function with shared support as nonzero (a.k.a. "logical nonzero entry"), despite the fact that the actual values could indeed be zero. In arbitrarily mapped elements (non-affine) and/or in complex bilinear forms, logical nonzero entries are indeed different from zero.

We assume that the number of iterations needed to solve a given problem before static condensation is of the same order as that needed after static condensation. A large family of iterative solvers complies with this assumption, as shown in the Appendix.

We further assume that the cost of building the preconditioner associated to the skeleton problem is negligible, since the number of unknowns in the skeleton problem is $\mathcal{O}(p)$ times smaller than those in the interiors of the elements. In the case of a multigrid solver, we also assume that the coarse-grid correction has a negligible cost, since it consists of solving a smaller-size problem.

For simplicity, we restrict our attention to boundary value problems (with Dirichlet boundary conditions) that are governed by second order partial differential equations (PDEs) of the form:

$$\begin{aligned} -\nabla \cdot (c_1 \nabla u) + c_2 \nabla u + c_3 u &= f \qquad \text{in } \Omega, \\ u &= u_0 \qquad \text{on } \Gamma = \partial\Omega, \end{aligned} \tag{2}$$

where c_1 is a symmetric positive definite tensor, c_2 a vector, and c_3 a scalar function, f is the right-hand side, u is the solution, u_0 is the Dirichlet data, and ∇, $\nabla\cdot$ are the gradient and divergence operators, respectively. c_1, c_2, and c_3 are bounded and spatially varying so they may also incorporate the Jacobian of a transformation from the reference elements to a deformed geometry [19,20]. We also assume that the coefficients are such that the above problem has a unique solution.

The variational formulation of problem (2) is given by (see e.g., [21]):

$$\begin{cases} \text{Find } u \in u_0 + V \text{ such that,} \\ b(u, v) = l(v) \quad \forall v \in V, \end{cases} \tag{3}$$