Contemporary Mathematics
Volume 157, 1994

The Dual Schur Complement Method With Well-Posed Local Neumann Problems

CHARBEL FARHAT
FRANCOIS-XAVIER ROUX

ABSTRACT. The Dual Schur Complement (DSC) Domain Decomposition (DD) method introduced by Farhat and Roux is an efficient and practical algorithm for the parallel solution of self-adjoint elliptic partial differential equations. A given spatial domain is partitioned into *disconnected* subdomains where an incomplete solution for the primary field is first evaluated using a direct method. Next, intersubdomain field continuity is enforced via a combination of discrete, polynomial, and/or piece-wise polynomial Lagrange multipliers, applied at the subdomain interfaces. This leads to a smaller size symmetric *dual* problem where the unknowns are the "gluing" Lagrange multipliers, and which is best solved with a preconditioned conjugate gradient (PCG) algorithm. However for time independent elasticity problems, every floating subdomain is associated with a singular stiffness matrix, so that the dual interface operator is in general indefinite. Previously, we have dealt with this issue by filtering out at each iteration of the PCG algorithm the contributions of the local null spaces. We have shown that for a small number of subdomains, say less than 32, this approach is computationally feasible. Unfortunately, the filtering phase couples the subdomain computations, increases the numerical complexity of the overall solution algorithm and limits its *parallel implementation scalability*, and therefore is inappropriate for a large number of subdomains. In this paper, we regularize the DSC method with a perturbed Lagrangian formulation which restores the positiveness of the dual interface operator, reduces the computational complexity of the overall methodology, and improves its parallel implementation scalability. This regularization procedure corresponds to a novel splitting method of the interface operator which entails well posed local discrete Neumann problems, even in the presence of floating subdomains. Therefore, it can be also interesting for other DD algorithms such as those considered by Bjordstad and Widlund, Marini and Quarteroni, De Roeck and Le Tallec, and recently by Mandel.

1991 Mathematics Subject Classification Primary 65N20, 65N30, 65W05
Supported by NASA Langley Research Center Grant NAG-1536427
and National Science Foundation Grant ASC-8717773.
This paper is in final form and no version of it will be submitted for publication elsewhere.

1. Introduction.

Recently, Farhat and Roux have introduced a Dual Schur Complement (DSC) Domain Decomposition (DD) method for the efficient solution of static [1-4] and transient [5] finite element structural problems on parallel processors. The method was shown to outperform direct solvers on both serial and coarse-grained multiprocessors such as the CRAY Y-MP system, and to compare favorably with other domain decomposition algorithms on a 32 processor hypercube [2]. However, for large number of subdomains and processors and for time independent problems, the DSC method may lose some of its efficiency because of its special treatment of floating subdomains. The objective of this paper is to present a regularization procedure that is based on a "balanced" perturbed Lagrangian formulation and which improves the overall computational efficiency of the DSC method. This regularization procedure corresponds to a novel splitting method of the interface operator which entails well posed local discrete Neumann problems, even in the presence of floating subdomains. It provides a practical mean to regularize several other DD algorithms using both Dirichlet and Neumann local solvers and which can suffer from the presence of floating subdomains [6, 7, 8, 9].

The resulting Regularized Dual Schur Complement (RDSC) method is a two-field alternative to the related three-field hybrid domain decomposition method introduced by Glowinski and Le Tallec [10]. Because of space limitations, we refer the reader to [2-3] for a background on the DSC method and for a discussion on the effect of floating subdomains on the algebraic properties and computational requirements of the resulting interface problem.

2. A regularized DSC method

2.1. The two-subdomain problem. The variational form of the three-dimensional boundary-value problem to be solved is as follows. Given f and h, find the displacement function u which is a stationary point of the energy functional:

(1)

$$J(v) = \frac{1}{2}a(v,v) - (v,f) - (v,h)_\Gamma$$

where

$$a(v,w) = \int_\Omega v_{(i,j)} c_{ijkl} w_{(k,l)}\, d\Omega; \qquad (v,f) = \int_\Omega v_i f_i\, d\Omega; \qquad (v,h)_\Gamma = \int_{\Gamma_h} v_i h_i\, d\Gamma$$

In the above, the indices i, j, k take the value 1 to 3, $v_{(i,j)} = (v_{i,j} + v_{j,i})/2$ and $v_{i,j}$ denotes the partial derivative of the $i-th$ component of v with respect to the $j-th$ spatial variable, c_{ijkl} are the elastic coefficients, Ω denotes the volume of the elastostatic body, Γ its piecewise smooth boundary, and Γ_h the piece of Γ where the tractions h_i are prescribed. If Ω is subdivided into two subdomains Ω_1 and Ω_2, solving the above elastostatic problem is equivalent to finding the displacement functions u_1 and u_2 which are stationary points of the perturbed Lagrangian functional:

$$H(v_1, v_2, \lambda) = J_1(v_1) + J_2(v_2) + \int_{\Gamma_I} \lambda(v_1 - v_2)\, d\Gamma \cdot \tag{2}$$
$$- \frac{1}{2} \int_{\overline{\Gamma}_I} \mathcal{L}(v_1|_{\overline{\Gamma}_I})^2 - \mathcal{L}(v_2|_{\overline{\Gamma}_I})^2 \, d\Gamma$$

where $\overline{\Gamma}_I \subset \Gamma_I$ is a subset of the interface boundary, and $\mathcal{L}$ is a linear operator that acts on the traces of v_1 and v_2 on $\overline{\Gamma}_I$. The proper selection of $\overline{\Gamma}_I$ and $\mathcal{L}$ is discussed later. In the remainder of this section, we assume that only Ω_2 is a floating subdomain — that is, a subdomain without sufficient Dirichlet boundary conditions to guarantee a non singular stiffness matrix. The finite element equations associated with (2) are given by:

$$(3)\quad (\mathbf{K}_1 - \boldsymbol{\Delta})\mathbf{u}_1 = \mathbf{f}_1 - \mathbf{B}_1^T\lambda; \quad (\mathbf{K}_2 + \boldsymbol{\Delta})\mathbf{u}_2 = \mathbf{f}_2 - \mathbf{B}_2^T\lambda; \quad \mathbf{B}_1\mathbf{u}_1 + \mathbf{B}_2\mathbf{u}_2 = 0$$

where $\mathbf{K}_1$ is positive definite, $\mathbf{K}_2$ is positive semi-definite, and $\boldsymbol{\Delta}$ is zero everywhere except on a subset of the interface boundary degrees of freedom. If discrete Lagrange multipliers are used, $\mathbf{B}_1$ and $\mathbf{B}_2$ are signed boolean matrices: otherwise, they are standard finite element matrices with full column rank. Clearly, $\boldsymbol{\Delta}$ serves the purpose of regularizing $\mathbf{K}_2$. However, $\boldsymbol{\Delta}$ is useful only if it is sparse and computationally inexpensive, and if it restores the positive definiteness nature of the stiffness in Ω_2, without destroying this algebraic property in Ω_1. In other words, the problem now is to find an economical matrix $\boldsymbol{\Delta}$ — or its corresponding operator $\mathcal{L}$ — which can stiffen Ω_2 enough so that $\mathbf{K}_2 + \boldsymbol{\Delta}$ is symmetric positive definite, without softening Ω_1 enough so that $\mathbf{K}_1 - \boldsymbol{\Delta}$ is singular or indefinite. We refer to this problem as a *balancing* problem, and we refer to the corresponding functional $H(v_1, v_2, \lambda)$ as a *balanced* perturbed Lagrangian.

REMARK 2.1.1. Given that $\int_{\overline{\Gamma}_I} \mathcal{L}(v_1|_{\overline{\Gamma}_I})^2 - \mathcal{L}(v_2|_{\overline{\Gamma}_I})^2 \, d\Gamma$ depends only on the traces of v_1 and v_2 on the subset $\overline{\Gamma}_I$ of the interface boundary Γ_I where the Lagrange multipliers λ enforce the continuity equation $(v_1 - v_2)|_{\Gamma_I} = 0$, the solution $(\mathbf{u}_1, \mathbf{u}_2)$ of the system of Eqs. (3) is independent of the operator $\mathcal{L}$ and its corresponding matrix $\boldsymbol{\Delta}$.

2.2. Balancing the subdomains. Here we consider the problem of constructing $\boldsymbol{\Delta}$ such that both $\mathbf{K}_1 - \boldsymbol{\Delta}$ and $\mathbf{K}_2 + \boldsymbol{\Delta}$ are symmetric positive definite.

Let n_I and n_r denote respectively the total number of unknowns on Γ_I, and the exact number of rigid body modes associated with Ω_2 — that is, the dimension of the null space of $\mathbf{K}_2$. These rigid body modes can be eliminated, for example, by fixing n_c degrees of freedom on Γ_I, where $n_r \leq n_c \leq n_I$. We

define $\overline{\Gamma}_I$ as the spatial support of these n_c degrees of freedom (FIG. 1).

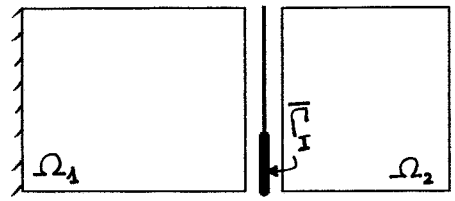

FIG. 1. $\overline{\Gamma}_I$ *for the two-subdomain problem with a local singularity*

Next, we partition the subdomain stiffness matrices as:

$$\mathbf{K}_1 = \begin{bmatrix} \mathbf{K}_1^{ff} & \mathbf{K}_1^{fc} \\ \mathbf{K}_1^{fc^T} & \mathbf{K}_1^{cc} \end{bmatrix} \quad \text{and} \quad \mathbf{K}_2 = \begin{bmatrix} \mathbf{K}_2^{ff} & \mathbf{K}_2^{fc} \\ \mathbf{K}_2^{fc^T} & \mathbf{K}_2^{cc} \end{bmatrix} \tag{4}$$

where the superscript ^{c} denotes those degrees of freedom on $\overline{\Gamma}_I$, and the superscript ^{f} denotes all of the other degrees of freedom. The above partitioning is characterized by:

1. $\mathbf{K}_1^{ff}$ and $\mathbf{K}_2^{ff}$ are symmetric positive definite,
2. $\mathbf{K}_1^{fc}$ and $\mathbf{K}_2^{fc}$ have full column rank,
3. $\mathbf{K}_1^{cc} - \mathbf{K}_1^{fc^T}\mathbf{K}_1^{ff^{-1}}\mathbf{K}_1^{fc}$ is symmetric positive definite, and:
4. $\mathbf{K}_2^{cc} - \mathbf{K}_2^{fc^T}\mathbf{K}_2^{ff^{-1}}\mathbf{K}_2^{fc}$ is symmetric positive semi-definite.

After all but the last n_c equations in each subdomain are reduced with a Gaussian elimination or a Choleski decomposition, $\mathbf{K}_1$ and $\mathbf{K}_2$ are overwritten with:

$$\begin{aligned} \mathbf{K}_1^R &= \begin{bmatrix} \mathbf{K}_1^{ff^R} & \mathbf{K}_1^{fc^R} \\ \cdots & \mathbf{K}_1^{cc} - \mathbf{K}_1^{fc^T}\mathbf{K}_1^{ff^{-1}}\mathbf{K}_1^{fc} \end{bmatrix} \\ \mathbf{K}_2^R &= \begin{bmatrix} \mathbf{K}_2^{ff^R} & \mathbf{K}_2^{fc^R} \\ \cdots & \mathbf{K}_2^{cc} - \mathbf{K}_2^{fc^T}\mathbf{K}_2^{ff^{-1}}\mathbf{K}_2^{fc} \end{bmatrix} \end{aligned} \tag{5}$$

where the superscript ^{R} indicates a reduced matrix. Now, if $\boldsymbol{\Delta}$ is constructed as:

$$\boldsymbol{\Delta} = \boldsymbol{\Delta}^* = \frac{1}{2}(\mathbf{K}_1^{cc} - \mathbf{K}_1^{fc^T}\mathbf{K}_1^{ff^{-1}}\mathbf{K}_1^{fc}) \tag{6}$$

then it follows that:

(7)

$$(\mathbf{K}_1 - \boldsymbol{\Delta}^*)^R = \begin{bmatrix} \mathbf{K}_1^{ff^R} & \mathbf{K}_1^{fc^R} \\ \cdots & \frac{1}{2}(\mathbf{K}_1^{cc} - \mathbf{K}_1^{fc^T}\mathbf{K}_1^{ff^{-1}}\mathbf{K}_1^{fc})^R \end{bmatrix}$$

$$(\mathbf{K}_2 + \boldsymbol{\Delta}^*)^R = \begin{bmatrix} \mathbf{K}_2^{ff^R} & \mathbf{K}_2^{fc^R} \\ \cdots & [\mathbf{K}_2^{cc} - \mathbf{K}_2^{fc^T}\mathbf{K}_2^{ff^{-1}}\mathbf{K}_2^{fc} + \frac{1}{2}(\mathbf{K}_1^{cc} - \mathbf{K}_1^{fc^T}\mathbf{K}_1^{ff^{-1}}\mathbf{K}_1^{fc})]^R \end{bmatrix}$$

which clearly demonstrates that both of $\mathbf{K}_1 - \mathbf{\Delta}^*$ and $\mathbf{K}_2 + \mathbf{\Delta}^*$ are symmetric positive definite matrices.

The extreme values of n_c correspond to two extreme regularization strategies. For $n_c = n_I$, the local problems are better conditioned than for $n_c = n_r$, but a lot of fill-in may be introduced during the factorization of $\mathbf{K}_2 + \mathbf{\Delta}^*$. On the other hand, for $n_c = n_r$, the local problems are not as well conditioned as for $n_c = n_I$, but $\mathbf{K}_2 + \mathbf{\Delta}^*$ and $\mathbf{K}_2$ have in general the same sparsity pattern, and therefore the latter regularization strategy is cheaper to implement. The case $n_r \leq n_c \leq n_I$ corresponds to a compromise between the two extreme options. In the remainder of this paper, we consider only the case where $n_c = n_r$, and therefore we have:

$$\mathbf{K}_2^{cc} - \mathbf{K}_2^{fc^T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc} = 0 \tag{8}$$

2.3. The multiple subdomain problem. For the sake of clarity, we first consider a strip-wise decomposed three-subdomain problem where Ω_1 is clamped at one end, and Ω_2 and Ω_3 are floating subdomains (FIG. 2). $\overline{\Gamma}_I^{(1,2)}$ is defined as the spatial support of $n_r^{(1,2)}$ degrees of freedom on the interface between Ω_1 and Ω_2 that must be constrained in order to remove the rigid body modes in Ω_2. Similarly, $\overline{\Gamma}_I^{(2,3)}$ is defined as the spatial support of $n_r^{(2,3)}$ degrees of freedom on the interface between Ω_2 and Ω_3 which must be constrained in order to eliminate the rigid body modes in Ω_3, *after* Ω_2 has been regularized. For subdomain Ω_2, we refer to the degrees of freedom on $\overline{\Gamma}_I^{(1,2)}$ as the *receiving* degrees of freedom, and to those on $\overline{\Gamma}_I^{(2,3)}$ as the *emitting* degrees of freedom.

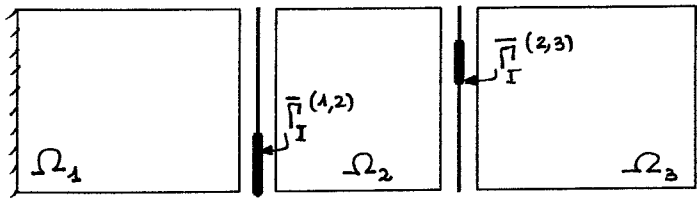

FIG. 2. *The strip-wise decomposed three-subdomain problem with local singularities*

We partition the subdomain stiffness matrices as:

$$\mathbf{K}_1 = \begin{bmatrix} \mathbf{K}_1^{ff} & \mathbf{K}_1^{fc} \\ \mathbf{K}_1^{fc^T} & \mathbf{K}_1^{cc} \end{bmatrix} \qquad \mathbf{K}_3 = \begin{bmatrix} \mathbf{K}_3^{ff} & \mathbf{K}_3^{fc} \\ \mathbf{K}_3^{fc^T} & \mathbf{K}_3^{cc} \end{bmatrix}$$

$$\text{and} \quad \mathbf{K}_2 = \begin{bmatrix} \mathbf{K}_2^{ff} & \mathbf{K}_2^{fc^r} & \mathbf{K}_2^{fc^e} \\ \mathbf{K}_2^{fc^{r}T} & \mathbf{K}_2^{c^r c^r} & \mathbf{K}_2^{c^r c^e} \\ \mathbf{K}_2^{fc^{e}T} & \mathbf{K}_2^{c^r c^{e}T} & \mathbf{K}_2^{c^e c^e} \end{bmatrix} \tag{9}$$

where the superscript $^{c^r}$ and $^{c^e}$ indicate the receiving and emitting degrees of freedom in Ω_2, respectively. After all but the last $n_r^{(1,2)}$ equations in Ω_1 and Ω_2,

and all but the last $n_r^{(2,3)}$ equations in Ω_3 are reduced, the subdomain stiffness matrices are overwritten with:

(10)

$$\mathbf{K}_1^R = \begin{bmatrix} \mathbf{K}_1^{ff^R} & \mathbf{K}_1^{fc^R} \\ \cdots & \mathbf{K}_1^{cc} - \mathbf{K}_1^{fc^T} \mathbf{K}_1^{ff^{-1}} \mathbf{K}_1^{fc} \end{bmatrix} \qquad \mathbf{K}_3 = \begin{bmatrix} \mathbf{K}_3^{ff^R} & \mathbf{K}_3^{fc^R} \\ \cdots & \mathbf{O} \end{bmatrix}$$

and

$$\mathbf{K}_2^R = \begin{bmatrix} \mathbf{K}_2^{ff^R} & \mathbf{K}_2^{fc^{r}R} & \mathbf{K}_2^{fc^{e}R} \\ \cdots & \mathbf{K}_2^{c^r c^r} - \mathbf{K}_2^{fc^{r}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fe^r} & \mathbf{K}_2^{c^r c^e} - \mathbf{K}_2^{fc^{r}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^e} \\ \cdots & [\mathbf{K}_2^{c^r c^e} - \mathbf{K}_2^{fc^{r}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^e}]^T & \mathbf{K}_2^{c^e c^e} - \mathbf{K}_2^{fc^{e}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^e} \end{bmatrix}$$

First, Ω_2 is regularized with the $n_r^{(1,2)} \times n_r^{(1,2)}$ matrix $\mathbf{\Delta}^{(1,2)} = \frac{1}{2}(\mathbf{K}_1^{cc} - \mathbf{K}_1^{fc^T} \mathbf{K}_1^{ff^{-1}} \mathbf{K}_1^{fc})$ after which the factored stiffnesses in Ω_1 and Ω_2 become:

$$(\mathbf{K}_1 - \mathbf{\Delta}^{(1,2)})^R = \begin{bmatrix} \mathbf{K}_1^{ff^R} & \mathbf{K}_1^{fc^R} \\ \cdots & \mathbf{\Delta}^{(1,2)^R} \end{bmatrix} \tag{11}$$

and

$$(\mathbf{K}_2 + \mathbf{\Delta}^{(1,2)})^R = \begin{bmatrix} \mathbf{K}_2^{ff^R} & \mathbf{K}_2^{fc^{r}R} & \mathbf{K}_2^{fc^{e}R} \\ \cdots & [\mathbf{K}_2^{c^r c^r} - \mathbf{K}_2^{fc^{r}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^r} + \mathbf{\Delta}^{(1,2)}]^R & [\mathbf{K}_2^{c^r c^e} - \mathbf{K}_2^{fc^{r}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^e}]^R \\ \cdots & \cdots & \mathbf{T}_2 \end{bmatrix}$$

where

$$\begin{aligned} \mathbf{T}_2 = {} & \mathbf{K}_2^{c^e c^e} - \mathbf{K}_2^{fc^{e}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^e} - [\mathbf{K}_2^{c^r c^e} - \mathbf{K}_2^{fc^{r}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^e}]^T \\ & [\mathbf{K}_2^{c^r c^r} - \mathbf{K}_2^{fc^{r}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^r} + \mathbf{\Delta}^{(1,2)}]^{-1} [\mathbf{K}_2^{c^r c^e} - \mathbf{K}_2^{fc^{r}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^e}] \end{aligned} \tag{12}$$

Next, Ω_3 is regularized with the $n_r^{(2,3)} \times n_r^{(2,3)}$ matrix $\mathbf{\Delta}^{(2,3)} = \frac{1}{2}\mathbf{T}_2$ and finally the factored stiffnesses in Ω_2 and Ω_3 become:

(13)

$$(\mathbf{K}_2 - \mathbf{\Delta}^{(2,3)})^R = \begin{bmatrix} \mathbf{K}_2^{ff^R} & \mathbf{K}_2^{fc^{r}R} & \mathbf{K}_2^{fc^{e}R} \\ \cdots & [\mathbf{K}_2^{c^r c^r} - \mathbf{K}_2^{fc^{r}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^r} + \mathbf{\Delta}^{(1,2)}]^R & [\mathbf{K}_2^{c^r c^e} - \mathbf{K}_2^{fc^{r}T} \mathbf{K}_2^{ff^{-1}} \mathbf{K}_2^{fc^e}]^R \\ \cdots & \cdots & \mathbf{\Delta}^{(2,3)^R} \end{bmatrix}$$

and

$$(\mathbf{K}_3 + \mathbf{\Delta}^{(2,3)})^R = \begin{bmatrix} \mathbf{K}_3^{ff^R} & \mathbf{K}_3^{fc^R} \\ \cdots & \mathbf{\Delta}^{(2,3)^R} \end{bmatrix} \tag{14}$$

For arbitrary mesh decompositions, the crucial point is to detect for a given subdomain Ω_s and a neighboring Ω_q, whether $\overline{\Gamma}_I^{(s,q)}$ is a "receiving" or "emitting" subset of Γ_I. If V_s denote the number of neighboring subdomains to Ω_s, the overall regularization algorithm can be implemented as follows:

Step 1. For every subdomain Ω_s, reduce all but the last $n_r^{(s)}$ equations of the corresponding stiffness matrix $\mathbf{K}_s$. (Note that for the minimal strategy $n_r^{(s)} \leq 3 \times V_s$ in two-dimensional problems, and $n_r^{(s)} \leq 6 \times V_s$ in three-dimensional ones.)

Step 2. Regularization proceeds from the non-floating subdomains towards the floating ones. Floating subdomains which contain both "receiving" and "emitting" interface subsets are regularized first, and the "receiving" degrees of freedom are treated before the "emitting" ones.

Clearly *Step 1* is a parallel step but *Step 2* is a sequential one. However, for most large-scale problems, the CPU time corresponding to *Step 2* is an extremely small fraction of the CPU time corresponding to *Step 1*, as the first step may involve millions of floating-point operations while the second one involves at most a few hundreds. Interprocessor communication is required only in *Step 2* and is limited to neighboring processors.

3. The regularized interface problem.

If the floating subdomains are not regularized, the interface problem associated with Eqs. (3) can be written in the case of N_s arbitrary subdomains as [1]:

$$\begin{bmatrix} \sum_{s=1}^{s=N_s} \mathbf{B}_s \mathbf{K}_s^* \mathbf{B}_s^T & -\mathbf{G}_I \\ -\mathbf{G}_I{}^T & \mathbf{O} \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} \mathbf{d} \\ -\mathbf{e} \end{bmatrix} \tag{15}$$

where

$$\mathbf{K}_s^* = \mathbf{K}_s^{-1} \quad \text{if } \Omega_s \text{ is not a floating subdomain}$$
$$\mathbf{K}_s^* = \mathbf{K}_s^{+} \quad \text{if } \Omega_s \text{ is a floating subdomain}$$

where $\mathbf{K}_s^+$ is a generalized inverse of $\mathbf{K}_s$, $\mathbf{G}_I$ is a full column rank matrix that stores the traces on the interface boundary of the null spaces corresponding to the floating subdomains, and $\boldsymbol{\alpha}$ stores the contributions of these null spaces to the local solutions. The above interface problem (15) is clearly indefinite and its solution requires special handling [1-3]. On the other hand, the regularized interface problem associated with Eqs. (3) can be written for N_s arbitrary subdomains as:

$$[\sum_{s=1}^{s=N_s} \mathbf{B}_s (\mathbf{K}_s + \delta_s \boldsymbol{\Delta}_s)^{-1} \mathbf{B}_s^T] \boldsymbol{\lambda} = \sum_{s=1}^{s=N_s} \mathbf{B}_s (\mathbf{K}_s + \delta_s \boldsymbol{\Delta}_s)^{-1} \mathbf{f}_s \tag{16}$$

where δ_s is +1,-1, or 0 if Ω_s is a receiving, sending, or non floating subdomain, respectively. Clearly, the above regularized system (16) is symmetric positive definite. The PCG algorithm can be applied to the solution of (16) at little additional cost and using only subdomain-by-subdomain scalable and intrinsically

parallel computations. As in the case of non floating subdomains, the above interface problem is preconditioned with $\mathbf{P}$ derived in [1] as:

$$\mathbf{P}^{-1} = \sum_{s=1}^{s=N_s} \mathbf{B}_s \mathbf{K}_s \mathbf{B}_s^T \tag{17}$$

4. Performance improvement.

Here, we consider the static analysis of the cabin of a launch vehicle subjected to external aerodynamic loading and internal pressurization (FIG. 3). All computations are performed on an iPSC/860 Touchstone machine. The structure is discretized with triangular shell elements. Several meshes of different sizes are constructed, each corresponding to a different number of processors and therefore to a different total memory size. In order to highlight the improvement in performance induced by the regularization approach described in this paper, both the "filtered" DSC method [1] and the RDSC method are applied to the solution of the same problem. Also, the Jacobi Preconditioned (diagonal scaling) Conjugate Gradient (JPCG) algorithm is used as a reference for CPU timings.

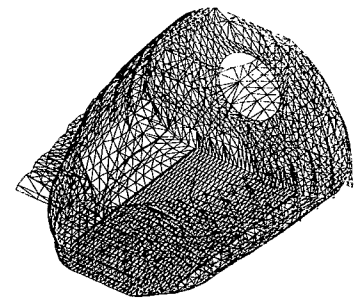

FIG. 3. *Finite element discretization of the cabin*

All measured performance results are summarized in TABLE 1 where **NP**, **NDOF**, and $\mathbf{T_p}$ denote respectively the number of processors, the number of degrees of freedom, and the solution parallel time measured in seconds. The number of iterations is indicated in between parenthesis. For all solution algorithms, convergence is established by requiring that the relative global residual be less than 10^{-3}:

$$\frac{||\mathbf{K}\mathbf{u} - \mathbf{f}||_2}{||\mathbf{f}||_2} \leq 10^{-3} \tag{18}$$

TABLE 1

Performance results - iPSC/860

NP	NDOF	T_p (JPCG)	T_p (DSC)	T_p (RDSC)
4	4,024	46 s. (700 it.)	23 s. (34 it.)	22 s. (38 it.)
16	16,456	110 s. (2308 it.)	54 s. (66 it.)	49 s. (62 it.)
32	31,928	168 s. (3980 it.)	79 s. (89 it.)	62 s. (92 it.)
64	59,064	295 s. (6127 it.)	189 s. (142 it.)	121 s. (144 it.)

The performance results reported above clearly indicate that:

1. the regularization process does not seem to negatively affect the performance of the solution of the interface problem as the DSC and RDSC method appear to converge after an almost identical number of iterations,
2. as expected, the performance improvement due to regularization is most important for an increasing number of processors,
3. both DSC methods are shown to outperform the JPCG algorithm by a factor of three.

REFERENCES

1. C. Farhat and F. X. Roux, A Method of Finite Element Tearing and Interconnecting and its Parallel Solution Algorithm, Int.J. Num. Meth. Eng., **32** (1991), 1205-1227.

2. C. Farhat and F. X. Roux, An Unconventional Domain Decomposition Method for an Efficient Parallel Solution of Large-Scale Finite Element Systems, SIAM J. Sc. Stat. Comp., **13** No. 1 (1992), 379-396.

3. C. Farhat, A Lagrange Multiplier Based Divide and Conquer Finite Element Algorithm, J. Comp. Systems in Eng., **2** No. 2/3 (1991), 149-156.

4. C. Farhat, A Saddle-Point Principle Domain Decomposition Method for the Solution of Solid Mechanics Problems, Proceedings of the Fifth SIAM Conference on Domain Decomposition Methods for Partial Differential Equations, 1991.

5. C. Farhat, Parallel Processing in Structural Mechanics: Blending Mathematical, Implementational, and Technological Advances, it Computing Methods in Applied Sciences and Engineering, ed. by R. Glowinski, Nova Science Publishers, Inc., New York, 1992, 89-106.

6. P. E. Bjordstad and O. B. Widlund, Iterative Methods for Solving Elliptic Problems on Regions Partitioned into Substructures, SIAM J. of Num. Anal., **23** No. 6 (1986).

7. P. L. D. Marini and A. Quarteroni, An Iterative Procedure for Domain Decomposition Methods : a Finite Element Approach, Proceedings of the First International Symposium on Domain Decomposition Methods for Partial Differential Equations, SIAM, 1988.

8. Y. H. De Roeck and P. Le Tallec, Analysis and Test of a Local Domain Decomposition Preconditioner, Proceedings of the Fourth SIAM International Conference on Domain Decomposition Methods, SIAM, 1991.

9. J. Mandel, Balancing Domain Decomposition, Comm. Appl. Num. Meth., (to appear)

10. R. Glowinski and P. Le Tallec, Augmented Lagrangian Interpretation of the Nonoverlapping Schwarz Alternating Method, Proceedings of the Third International Symposium on Domain Decomposition Methods for Partial Differential Equations, SIAM, 1989, 224-231.

DEPARTMENT OF AEROSPACE ENGINEERING SCIENCES, CENTER FOR SPACE STRUCTURES AND CONTROLS, UNIVERSITY OF COLORADO, BOULDER, COLORADO 80309-0429, USA

O.N.E.R.A. GROUPE CALCUL PARALLELE, *Current address*: 29 Av. de la Division Leclerc, BP72 92322 Chatillon Cedex, FRANCE