COMMUNICATIONS IN NUMERICAL METHODS IN ENGINEERING, Vol. 9, 233–241 (1993)

BALANCING DOMAIN DECOMPOSITION

JAN MANDEL

Computational Mathematics Group, University of Colorado at Denver, Denver, CO 80217-3364, U.S.A.

SUMMARY

The Neumann–Neumann algorithm is known to be an efficient domain decomposition preconditioner with unstructured subdomains for iterative solution of finite-element discretizations of difficult problems with strongly discontinuous coefficients (De Roeck and Le Tallec, 1991). However, this algorithm suffers from the need to solve in each iteration an inconsistent singular problem for every subdomain, and its convergence deteriorates with increasing number of subdomains due to the lack of a coarse problem to propagate the error globally. We show that the equilibrium conditions for the singular problems on subdomains lead to a simple and natural construction of a coarse problem. The construction is purely algebraic and applies also to systems such as those that arise in elasticity. A convergence bound independent of the number of subdomains is proved and results of computational tests are reported.

1. INTRODUCTION

Consider a system of linear algebraic equations

$$\mathbf{A}\mathbf{x} = \mathbf{f} \tag{1}$$

arising from a finite-element discretization of a linear, elliptic, self-adjoint boundary-value problem on a domain Ω. Throughout this paper, assume that the stiffness matrix $\mathbf{A}$ is an $m \times m$, symmetric positive-definite matrix. The domain Ω is split into non-overlapping subdomains $\Omega_1, \ldots, \Omega_k$, each of which is the union of some of the elements. Let $\mathbf{x}_i$ be the vector of degrees of freedom corresponding to all elements in subdomain Ω_i, and let $\mathbf{N}_i$ denote the 0–1 matrix that maps the degrees of freedom $\mathbf{x}_i$ into global degrees of freedom; then $\mathbf{x}_i = \mathbf{N}_i^T\mathbf{x}$ and, by the standard subassembly process,

$$\mathbf{A} = \sum_{i=1}^{k} \mathbf{N}_i\mathbf{A}_i\mathbf{N}_i^T$$

where $\mathbf{A}_i$ is the local stiffness matrix corresponding to subdomain Ω_i. Each $\mathbf{x}_i$ is split into degrees of freedom $\bar{x}_i$, that correspond to the interface of the subdomain Ω_i with other subdomains, and the remaining degrees of freedom $\dot{x}_i$, which are associated with the subdomain Ω_i only. The subdomain stiffness matrices and the 0–1 matrices $\mathbf{N}_i$ are then split accordingly:

$$\mathbf{x}_i = \begin{pmatrix} \bar{x}_i \\ \dot{x}_i \end{pmatrix}, \qquad \mathbf{A}_i = \begin{pmatrix} \bar{\mathbf{A}}_i & \mathbf{B}_i \\ \mathbf{B}_i^T & \dot{\mathbf{A}}_i \end{pmatrix}, \qquad \mathbf{N}_i = (\bar{N}_i, \dot{N}_i)$$

After eliminating the $\dot{x}_i$, the system (1) becomes

$$\mathbf{S}\mathbf{u} = \mathbf{g} \tag{2}$$

0748–8025/93/030233–09$09.50

Received 18 November 1991
Revised 16 March 1992

where $\mathbf{S}$ is the assembly of the Schur complements,

$$\mathbf{S} = \sum_{i=1}^{k} \bar{\mathbf{N}}_i \mathbf{S}_i \bar{\mathbf{N}}_i^{\mathrm{T}}, \qquad \mathbf{S}_i = \bar{\mathbf{A}}_i - \mathbf{B}_i \mathring{\mathbf{A}}_i^{-1} \mathbf{B}_i^{\mathrm{T}} \tag{3}$$

We assume that the subdomain matrices $\mathbf{A}_i$ are symmetric and positive semi-definite, with the submatrices $\mathring{\mathbf{A}}_i$ non-singular. Then the Schur complements $\mathbf{S}_i$ are also positive semi-definite. The reduced problem (2) is posed in the space of the interface degrees of freedom, which we denote by V. Similarly, let V_i be the space of the interface degrees of freedom for the subdomain Ω_i. Then, interpreting matrices as mappings, we have

$$\mathbf{S}\colon \mathrm{V} \rightarrow \mathrm{V}, \qquad \mathbf{S}_i\colon \mathrm{V}_i \rightarrow \mathrm{V}_i, \qquad \bar{\mathbf{N}}_i\colon \mathrm{V}_i \rightarrow \mathrm{V} \tag{4}$$

Denote $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u}^{\mathrm{T}}\mathbf{v}$ and, for symmetric positive semi-definite $\mathbf{B}$, $\langle \mathbf{u}, \mathbf{v} \rangle_{\mathrm{B}} = \langle \mathbf{Bu}, \mathbf{v} \rangle$ and $\| \mathbf{u} \|_{\mathrm{B}} = (\langle \mathbf{u}, \mathbf{u} \rangle_{\mathrm{B}})^{1/2}$.

A large number of domain decomposition (or substructuring) methods consist of solving the reduced system (2) iteratively; see, for example References 1 and 2. Since $\mathbf{S}$ is symmetric positive definite, the preconditioned conjugate gradient method[3,4] is the standard choice for iterative methods. This method requires at each step the solution of an auxiliary problem

$$\mathbf{Mz} = \mathbf{r} \tag{5}$$

with a symmetric positive-definite matrix $\mathbf{M}$, called a preconditioner. In l steps, preconditioned conjugate gradients yield an approximate solution $\mathbf{u}^{(l)}$ with the error bounded according to

$$\| \mathbf{u}^{(l)} - \mathbf{S}^{-1}\mathbf{g} \|_{\mathrm{S}} \leqq 2\left(\frac{\sqrt{\varkappa} - 1}{\sqrt{\varkappa} + 1}\right)^{l} \| \mathbf{u}^{(0)} - \mathbf{S}^{-1}\mathbf{g} \|_{\mathrm{S}}$$

where $\varkappa = \mathrm{cond}(\mathbf{M}, \mathbf{S}) = \lambda_{\max}(\mathbf{M}^{-1}\mathbf{S})/\lambda_{\min}(\mathbf{M}^{-1}\mathbf{S})$ is the relative condition number of $\mathbf{M}$ and $\mathbf{S}$.[4] The basic problem here is, of course, to achieve low $\varkappa$ while keeping the cost of the solution of the auxiliary system (5) low.

The preconditioner proposed in this paper is a variation of the following preconditioner due to De Roeck and Le Tallec,[5] based on an earlier work for the case of two subdomains[6] and a closely related method of Glowinski and Wheeler for mixed problems.[7] It is also called the Neumann–Neumann preconditioner because of an interpretation in terms of boundary conditions on interfaces of the subdomains.[2,8] The method uses a collection of matrices $\mathbf{D}_i$ that form a decomposition of unity on the space V,

$$\sum_{i=1}^{k} \bar{\mathbf{N}}_i \mathbf{D}_i \bar{\mathbf{N}}_i^{\mathrm{T}} = \mathbf{I} \tag{6}$$

Remark 1

The simplest choice for $\mathbf{D}_i$ is the diagonal matrix with diagonal elements equal to the reciprocal of the number of subdomains with which the degree of freedom is associated.[5]

Algorithm 1. Given $\mathbf{r} \in \mathrm{V}$, compute $\mathbf{M}^{-1}\mathbf{r}$ as follows. Distribute $\mathbf{r}$ to the subdomains Ω_i by

$$\mathbf{r}_i = \mathbf{D}_i^{\mathrm{T}} \bar{\mathbf{N}}_i^{\mathrm{T}} \mathbf{r}, \qquad i = 1, \ldots, k$$

Solve for subdomains the local problems

$$\mathbf{S}_i \mathbf{z}_i = \mathbf{r}_i, \qquad i = 1, \ldots, k \tag{7}$$

and average the local solutions on the interfaces

$$\mathbf{z} = \sum_{i=1}^{k} \bar{\mathbf{N}}_i \mathbf{D}_i \mathbf{z}_i$$

It is easy to see that the solution $\mathbf{z}_i$ of (7) can be found, without the explicit creation of the Schur complement, from the system

$$\begin{pmatrix} \bar{\mathbf{A}}_i & \mathbf{B}_i \\ \mathbf{B}_i^T & \dot{\mathbf{A}}_i \end{pmatrix} \begin{pmatrix} \mathbf{z}_i \\ \cdot \end{pmatrix} = \begin{pmatrix} \mathbf{r}_i \\ 0 \end{pmatrix} \tag{8}$$

Unfortunately, $\mathbf{A}_i$ and thus $\mathbf{S}_i$ are typically singular. In this case, De Roeck and Le Tallec[5] suggest modifying the Gaussian elimination algorithm for solving (8) to avoid zero pivots and thus solve (8) only approximately. A different approximate solver was proposed by Dryja and Widlund.[8]

Another drawback of Algorithm 1 is the lack of a mechanism to exchange information between all subdomains in the preconditioning step and thus to prevent the growth of the condition number with the number of subdomains. Indeed, it was observed experimentally that the condition number of Algorithm 1 deteriorates with the number of subdomains, and that 16 subdomains are a practical limit.[9] In other domain decomposition methods, the needed propagation of errors has been accomplished by solving a 'course problem' with few degrees of freedom per subdomain in each iteration.[1,10–13] Such methods are close in spirit to multigrid methods and especially to two-level methods such as in References 14–16. Several such coarse problems for Algorithm 1 have been suggested in Reference 8, based on earlier coarse problems in different contexts. A related method was introduced and tested in a parallel environment by Farhat and Roux.[17,18] Their method enforces continuity on subdomain interfaces by Lagrange multipliers and uses a coarse problem to obtain consistency of the singular problems associated with subdomains.

This paper presents a new construction of a coarse problem for the method of Reference 5, which is very simple and easy to program. Unlike the coarse problems in Reference 8, no part of our algorithm needs to know what are the faces, edges or vertices of the subdomains. The subdomains can be completely unstructured, and the algorithm is formulated in purely algebraic terms. In fact, the abstract results here apply more generally to block iterations for arbitrary symmetric, positive-definite systems.

2. FORMULATION OF THE NEW BALANCING PRECONDITIONER

The problem (7) has a solution if and only if $\mathbf{r}_i$ satisfies the equilibrium condition

$$\langle \mathbf{v}, \mathbf{r}_i \rangle = 0, \qquad \text{for all } \mathbf{v} \in \text{Null } \mathbf{S}_i \tag{9}$$

If a solution $\mathbf{z}_i$ exists, then it is determined only modulo Null $\mathbf{S}_i$. Our construction of the coarse problem follows naturally from the requirement that the systems (7) should be solvable, and that the result of the preconditioner should not depend on the choice of a specific solution.

Let $n_i = \dim \mathrm{V}_i$ and $\mathbf{Z}_i$ be $n_i \times m_i$ matrices of full column rank such that

$$\text{Null } \mathbf{S}_i \subset \text{Range } \mathbf{Z}_i, \qquad i = 1, \ldots, k \tag{10}$$

and let W be the subspace of V defined by

$$\mathrm{W} = \{\mathbf{v} \in \mathrm{V} \colon \mathbf{v} = \sum_{i=1}^{k} \bar{\mathbf{N}}_i \mathbf{D}_i \mathbf{u}_i,\ \mathbf{u}_i \in \text{Range } \mathbf{Z}_i\}$$

We say that $\mathbf{s} \in \mathrm{V}$ is balanced if

$$\mathbf{Z}_i^T \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \mathbf{s} = 0, \qquad i = 1, \ldots, k \tag{11}$$

and note that (11) and (10) imply (9). The process of replacing $\mathbf{r}$ by a balanced $\mathbf{s} = \mathbf{r} - \mathbf{S}\mathbf{w}, \mathbf{w} \in \mathrm{W}$, will be called balancing.

Algorithm 2

Given $\mathbf{r} \in \mathrm{V}$, compute $\mathbf{z} = \mathbf{M}^{-1}\mathbf{r}$ as follows. Balance the original residual by solving the auxiliary problem for unknown vectors $\boldsymbol{\lambda}_i \in \Re^{m_i}$,

$$\mathbf{Z}_i^T \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \left(\mathbf{r} - \mathbf{S} \sum_{j=1}^{k} \overline{\mathbf{N}}_j \mathbf{D}_j \mathbf{Z}_j \boldsymbol{\lambda}_j \right) = 0, \qquad i = 1, \ldots, k \tag{12}$$

and set

$$\mathbf{s} = \mathbf{r} - \mathbf{S} \sum_{j=1}^{k} \overline{\mathbf{N}}_j \mathbf{D}_j \mathbf{Z}_j \boldsymbol{\lambda}_j, \qquad \mathbf{s}_i = \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \mathbf{s}, \qquad i = 1, \ldots, k \tag{13}$$

Find any solution $\mathbf{u}_i$ for each of the local problems

$$\mathbf{S}_i \mathbf{u}_i = \mathbf{s}_i, \qquad i = 1, \ldots, k \tag{14}$$

balance the residual by solving the auxiliary problem for $\mu_i \in \Re^{m_i}$,

$$\mathbf{Z}_i^T \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \left(\mathbf{r} - \mathbf{S} \sum_{j=1}^{k} \overline{\mathbf{N}}_j \mathbf{D}_j (\mathbf{u}_j + \mathbf{Z}_j \mu_j) \right) = 0, \qquad i = 1, \ldots, k \tag{15}$$

and average the result on the interfaces according to

$$\mathbf{z} = \sum_{i=1}^{k} \overline{\mathbf{N}}_i \mathbf{D}_i (\mathbf{u}_i + \mathbf{Z}_i \mu_i) \tag{16}$$

We allow the case that some $m_i = 0$; then $\mathbf{Z}_i$ as well as the block unknowns μ_i and λ_i are void, and the ith block equation is taken out of (12) and (15).

In an implementation, it is advantageous to balance the residual of the initial approximation as in (15); then the first balancing step (12) in every iteration can be omitted since the residual $\mathbf{r}$ received from the conjugate gradients algorithm will be always balanced and so always $\boldsymbol{\lambda}_i = 0, i = 1, \ldots, k$.

The local solves (14) can be done completely in parallel. Communication between subdomains occurs only in creating the local right-hand sides $\mathbf{s}_i$ in (13), in the solution of the balancing problem (15), and in the averaging (16). Matrix–vector multiply $\mathbf{S}\mathbf{u}$ also requires communication between subdomains.

The matrix of the problem (15) is size $m = \Sigma_{i=1}^k m_i$ and sparse, with zero blocks corresponding to subdomains that have no common neighbouring subdomain. It can be assembled using const. m inner products and m matrix–vector multiplications of the form $\mathbf{S}\overline{\mathbf{N}}_i\mathbf{D}_i(\mathbf{Z}_i)_j$, where $(\mathbf{Z}_i)_j$ denotes the jth column of $\mathbf{Z}_i$. In addition, for a given i, these matrix–vector products can be calculated using only the matrices $\mathbf{S}_k$ for subdomains Ω_k adjacent to Ω_i. The inner products can be restricted to subvectors in a similar fashion. As a result, the cost of computing the matrix of (15) grows only as $\Sigma_{i=1}^k m_i^2$.

3. THEORY

Our analysis is based on the observation that Algorithm 2 can be interpreted as one iteration with zero initial solution of a two-level algorithm of variational multigrid type[19] for the problem $\mathbf{Sz}=\mathbf{r}$, with coarse grid space W. Let $\mathbf{P}$ be the S-orthogonal projection onto W and

$$\mathbf{T}=\sum_{i=1}^{k}\overline{\mathbf{N}}_i\mathbf{D}_i\mathbf{S}_i^{+}\mathbf{D}_i^T\overline{\mathbf{N}}_i^T \tag{17}$$

where $\mathbf{S}_i^+$ is the Moore–Penrose pseudoinverse of $\mathbf{S}_i$ (cf., for example, Reference 4). In particular, $\mathbf{S}_i^+$ is symmetric and $\mathbf{S}_i\mathbf{S}_i^+\mathbf{s}_i=\mathbf{s}_i$ for all $\mathbf{s}_i\in$ Range $\mathbf{S}_i$.

The following Lemma gives an explicit form of the preconditioner $\mathbf{M}$, implemented by Algorithm 2.

Lemma 3.1. Algorithm 2 returns $\mathbf{z}=\mathbf{M}^{-1}\mathbf{r}$, where

$$\mathbf{M}^{-1}=((\mathbf{I}-\mathbf{P})\mathbf{TS}(\mathbf{I}-\mathbf{P})+\mathbf{P})\mathbf{S}^{-1}$$

In particular, $\mathbf{z}$ does not depend on the choice of the local solutions $\mathbf{u}_i$ in (14).

Proof. Denote $\mathbf{z}^*=\mathbf{S}^{-1}\mathbf{r}$ and $\mathbf{w}=\Sigma_{j=1}^{k}\overline{\mathbf{N}}_j\mathbf{D}_j\mathbf{Z}_j\boldsymbol{\lambda}_j$ from (12). Then

$$\langle\mathbf{z}^*-\mathbf{w},\mathbf{v}\rangle_{\mathrm{S}}=0,\qquad \text{for all } \mathbf{v}\in\mathrm{W}$$

so $\mathbf{w}=\mathbf{Pz}^*$. Denote $\mathbf{u}=\Sigma_{j=1}^{k}\overline{\mathbf{N}}_j\mathbf{D}_j\mathbf{u}_j$. From (13) and (14), it follows that

$$\mathbf{u}=\mathbf{Ts}+\mathbf{w}'$$

where $\mathbf{w}'\in\mathrm{W}$, and

$$\mathbf{s}=\mathbf{r}-\mathbf{Sw}=(\mathbf{Sz}^*-\mathbf{Sw})=\mathbf{S}(\mathbf{z}^*-\mathbf{w})=\mathbf{S}(\mathbf{I}-\mathbf{P})\mathbf{z}^* \tag{18}$$

Finally, using (15) and denoting $\mathbf{y}=\Sigma_{j=1}^{k}\overline{\mathbf{N}}_j\mathbf{D}_j\mathbf{Z}_j\boldsymbol{\mu}_j\in\mathrm{W}$, we obtain

$$\langle\mathbf{Sz}^*-\mathbf{Su}-\mathbf{Sy},\mathbf{v}\rangle=\langle\mathbf{Sz}^*-\mathbf{STs}-\mathbf{S}(\mathbf{y}+\mathbf{w}'),\mathbf{v}\rangle=0,\qquad \text{for all } \mathbf{v}\in\mathrm{W}$$

Consequently, $\mathbf{y}+\mathbf{w}'=\mathbf{P}(\mathbf{z}^*-\mathbf{Ts})$, so

$$\mathbf{z}=\mathbf{u}+\mathbf{y}=\mathbf{Ts}+\mathbf{y}+\mathbf{w}'=\mathbf{Ts}+\mathbf{P}(\mathbf{z}^*-\mathbf{Ts})=(\mathbf{I}-\mathbf{P})\mathbf{Ts}+\mathbf{Pz}^*$$

and the conclusion follows using (18). □

We are now in the position to formulate and prove a general bound on the condition number.

Theorem 3.2

Algorithm 2 returns $\mathbf{z}=\mathbf{M}^{-1}\mathbf{r}$, where $\mathbf{M}$ is symmetric positive definite and cond$(\mathbf{M},\mathbf{S})\leqslant C$, where

$$C=\sup\left\{\frac{\Sigma_{j=1}^{k}\|\overline{\mathbf{N}}_j^T\,\Sigma_{i=1}^{k}\overline{\mathbf{N}}_i\mathbf{D}_i\mathbf{u}_i\|_{\mathbf{S}_j}^2}{\Sigma_{i=1}^{k}\|\mathbf{u}_i\|_{\mathbf{S}_i}^2}:\ \mathbf{u}_i\in\mathrm{V}_i.\ \langle\mathbf{v}_i,\mathbf{u}_i\rangle=0 \text{ for all } \mathbf{v}_i\in\text{Range }\mathbf{Z}_i\right\} \tag{19}$$

Proof. By S-orthogonality of $\mathbf{P}$, for all $\mathbf{u},\mathbf{v}\in\mathrm{V}$,

$$\langle\mathbf{M}^{-1}\mathbf{Su},\mathbf{v}\rangle_{\mathrm{S}}=\langle\mathbf{TS}(\mathbf{I}-\mathbf{P})\mathbf{u},(\mathbf{I}-\mathbf{P})\mathbf{v}\rangle_{\mathrm{S}}+\langle\mathbf{Pu},\mathbf{Pv}\rangle_{\mathrm{S}} \tag{20}$$

Since all $\mathbf{S}_i^+$ are symmetric, $\mathbf{T}$ is symmetric, and so $\mathbf{M}$ is also symmetric by (20).

Let $\mathbf{u}\in\mathrm{V}$ and $\mathbf{y}=(\mathbf{I}-\mathbf{P})\mathbf{u}$. Then $\mathbf{Sy}$ is balanced and, using (10), we get that there exist $\mathbf{u}_i\in\mathrm{V}_i$, $i=1,\ldots,k$, such that

$$\mathbf{S}_i\mathbf{u}_i=\mathbf{D}_i^T\overline{\mathbf{N}}_i^T\mathbf{S}y,\qquad \langle\mathbf{v}_i,\mathbf{u}_i\rangle=0 \text{ for all } \mathbf{v}_i\in\text{Null }\mathbf{S}_i \tag{21}$$

To estimate $\langle \mathbf{TSy}, \mathbf{y} \rangle_{\mathbf{S}}$, we proceed as in Reference 5. From (3) and (17),

$$\begin{aligned}\langle \mathbf{TSy}, \mathbf{y} \rangle_{\mathbf{S}} = \langle \mathbf{Sy}, \mathbf{TSy} \rangle &= \sum_{j=1}^{k} \langle \mathbf{S}_j \overline{\mathbf{N}}_j^T \mathbf{y}, \overline{\mathbf{N}}_j^T \mathbf{TSy} \rangle \\ &= \sum_{j=1}^{k} \langle \mathbf{S}_j \overline{\mathbf{N}}_j^T \mathbf{y}, \overline{\mathbf{N}}_j^T \sum_{i=1}^{k} \overline{\mathbf{N}}_i \mathbf{D}_i \mathbf{S}_i^{+} \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \mathbf{Sy} \rangle \\ &= \sum_{i,j=1}^{k} \langle \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \overline{\mathbf{N}}_j \mathbf{S}_j \overline{\mathbf{N}}_j^T \mathbf{y}, \mathbf{S}_i^{+} \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \mathbf{Sy} \rangle \\ &= \sum_{i=1}^{k} \langle \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \mathbf{Sy}, \mathbf{u}_i \rangle\end{aligned}$$

which gives

$$\langle \mathbf{TSy}, \mathbf{y} \rangle_{\mathbf{S}} = \sum_{i=1}^{k} \langle \mathbf{S}_i \mathbf{u}_i, \mathbf{u}_i \rangle \tag{22}$$

Further,

$$\begin{aligned}\langle \mathbf{y}, \mathbf{y} \rangle_{\mathbf{S}} &= \sum_{j=1}^{k} \langle \mathbf{S}_j \overline{\mathbf{N}}_j^T \mathbf{y}, \overline{\mathbf{N}}_j^T \mathbf{y} \rangle = \sum_{j=1}^{k} \langle \overline{\mathbf{N}}_j \mathbf{S}_j \overline{\mathbf{N}}_j^T \mathbf{y}, \sum_{i=1}^{k} \overline{\mathbf{N}}_i \mathbf{D}_i \overline{\mathbf{N}}_i^T \mathbf{y} \rangle = \sum_{i=1}^{k} \langle \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \mathbf{Sy}, \overline{\mathbf{N}}_i^T \mathbf{y} \rangle \\ &= \sum_{i=1}^{k} \langle \mathbf{S}_i \mathbf{u}_i, \overline{\mathbf{N}}_i^T \mathbf{y} \rangle \leqslant \left(\sum_{i=1}^{k} \langle \mathbf{S}_i \mathbf{u}_i, \mathbf{u}_i \rangle \right)^{1/2} \left(\sum_{i=1}^{k} \langle \mathbf{S}_i \overline{\mathbf{N}}_i^T \mathbf{y}, \overline{\mathbf{N}}_i^T \mathbf{y} \rangle \right)^{1/2} \\ &= \left(\sum_{i=1}^{k} \langle \mathbf{S}_i \mathbf{u}_i, \mathbf{u}_i \rangle \right)^{1/2} (\langle \mathbf{Sy}, \mathbf{y} \rangle)^{1/2}\end{aligned}$$

thus

$$\langle \mathbf{y}, \mathbf{y} \rangle_{\mathbf{S}} \leqslant \sum_{i=1}^{k} \langle \mathbf{S}_i \mathbf{u}_i, \mathbf{u}_i \rangle \tag{23}$$

Finally,

$$\begin{aligned}\sum_{i=1}^{k} \langle \mathbf{S}_i \mathbf{u}_i, \mathbf{u}_i \rangle = \sum_{i=1}^{k} \langle \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \mathbf{Sy}, \mathbf{u}_i \rangle &= \sum_{i=1}^{k} \langle \mathbf{D}_i^T \overline{\mathbf{N}}_i^T \sum_{j=1}^{k} \overline{\mathbf{N}}_j \mathbf{S}_j \overline{\mathbf{N}}_j^T \mathbf{y}, \mathbf{u}_i \rangle \\ &= \sum_{j=1}^{k} \langle \mathbf{S}_j \overline{\mathbf{N}}_j^T \mathbf{y}, \overline{\mathbf{N}}_j^T \sum_{i=1}^{k} \overline{\mathbf{N}}_i \mathbf{D}_i \mathbf{u}_i \rangle \\ &\leqslant \left(\sum_{j=1}^{k} \langle \mathbf{S}_j \overline{\mathbf{N}}_j^T \mathbf{y}, \overline{\mathbf{N}}_j^T \mathbf{y} \rangle \right)^{1/2} \left(\sum_{j=1}^{k} \| \overline{\mathbf{N}}_j^T \sum_{i=1}^{k} \overline{\mathbf{N}}_i \mathbf{D}_i \mathbf{u}_i \|_{\mathbf{S}_i}^2 \right)^{1/2}\end{aligned}$$

which yields

$$\sum_{i=1}^{k} \langle \mathbf{S}_i \mathbf{u}_i, \mathbf{u}_i \rangle \leqslant C \langle \mathbf{y}, \mathbf{y} \rangle_{\mathbf{S}} \tag{24}$$

From (22), (23) and (24), $\langle \mathbf{y}, \mathbf{y} \rangle_{\mathbf{S}} \leqslant \langle \mathbf{M}^{-1} \mathbf{Sy}, \mathbf{y} \rangle_{\mathbf{S}} \leqslant C \langle \mathbf{y}, \mathbf{y} \rangle_{\mathbf{S}}$, which together with (20) and the Pythagorean theorem $\langle \mathbf{u}, \mathbf{u} \rangle_{\mathbf{S}} = \langle \mathbf{y}, \mathbf{y} \rangle_{\mathbf{S}} + \langle \mathbf{Pu}, \mathbf{Pu} \rangle_{\mathbf{S}}$ gives

$$\langle \mathbf{u}, \mathbf{u} \rangle_{\mathbf{S}} \leqslant \langle \mathbf{M}^{-1} \mathbf{Su}, \mathbf{u} \rangle_{\mathbf{S}} \leqslant C \langle \mathbf{u}, \mathbf{u} \rangle_{\mathbf{S}}$$

concluding the proof. □

Theorem 3.2 can now be used to obtain asymptotic bounds on the condition number in three dimensions by a simple application of an estimate from Reference 5. An analogous bound for the two-dimensional case will be published elsewhere. For ease of exposition and to avoid tedious notation, we restrict ourselves to the following simple problem. Generalization to more complicated problems, such as general boundary conditions, non-constant coefficients, or linear elasticity, is straightforward.

Theorem 3.3.

Let (1) be a discretization by quasiuniform, conforming linear elements with characteristic mesh size h, of the Laplace equation on a domain $\Omega \subset \mathfrak{R}^3$, with Dirichlet boundary condition on $\Gamma \subset \partial\Omega$ and Neumann boundary conditions on $\partial\Omega\backslash\Gamma$. Assume that $\Omega_i = \mathrm{T}_i\hat{\Omega}$, where $\hat{\Omega}$ is a reference square or cube, and the mappings T_i satisfy the Lipschitz conditions:

$$|\mathrm{T}_i\mathbf{x} - \mathrm{T}_i\mathbf{y}| \leqslant \text{const.}\ H|\mathbf{x}-\mathbf{y}|, \qquad \text{for all } \mathbf{x}, \mathbf{y} \in \hat{\Omega}$$
$$|\mathrm{T}_i^{-1}\mathbf{x} - \mathrm{T}_i^{-1}\mathbf{y}| \leqslant \text{const.}\ H^{-1}|\mathbf{x}-\mathbf{y}|, \qquad \text{for all } \mathbf{x}, \mathbf{y} \in \hat{\Omega}_i$$

where $H = \max \operatorname{diam} \Omega_i$. All boundary segments $\partial\Omega \cap \partial\Omega_i$ are assumed to be either empty or of positive boundary measure, and, in the latter case, the type of the boundary condition is assumed not to change on $\partial\Omega \cap \partial\Omega_i$. The matrix $\mathbf{Z}_i$ is a column constant vector if $\partial\Omega_i \cap \Gamma = \emptyset$, void otherwise, and the weight matrices $\mathbf{D}_i$ are chosen as in Remark 1.1.

Then $\operatorname{cond}(\mathbf{M}, \mathrm{S}) \leqslant \text{const.}\ (1 + \log^2 H/h)$, where the constant does not depend on the number of the subdomains.

Proof. For $v_i \in \mathrm{V}_i$, denote by v_i^{h} the discrete harmonic extension of v, that is,

$$v_i^{\mathrm{h}} = \begin{pmatrix} v_i \\ \dot{v}_i \end{pmatrix}, \qquad \| v_i^{\mathrm{h}} \|_{\mathrm{A}_i} \rightarrow \min$$

Then

$$\| \mathbf{v}_i \|_{\mathrm{S}_i} = \| \mathbf{v}_i^{\mathrm{h}} \|_{\mathrm{A}_i} = | \mathrm{h}(\mathbf{v}_i) |_{1,2,\Omega_i} \tag{25}$$

where $\mathrm{h}(\mathbf{v}_i)$ denotes the continuous, piece-wise linear function corresponding to the coefficient vector $\mathbf{v}_i^{\mathrm{h}}$, and $|\cdot|_{1,2,\Omega_i}$ is the Sobolev semi-norm in the space $\mathrm{W}^{1,2}(\Omega_i)$. Because of the construction of $\mathbf{Z}_i$ and assumptions on the boundary conditions, the semi-norms (25) are norms on the space

$$\mathbf{U}_i = \{\mathbf{v}_i \in \mathrm{V}_i\colon\ \mathbf{Z}_i^{\mathrm{T}}\mathbf{v}_i = 0\}$$

Note that if the constraint $\mathbf{Z}_i^{\mathrm{T}}\mathbf{v}_i = 0$ is void, our assumptions imply that Dirichlet boundary conditions are imposed on a whole face of Ω_i, and then $\mathrm{h}(\mathbf{v}_i)$ must satisfy corresponding homogeneous boundary conditions. Because the $\mathrm{W}^{1,2}$ semi-norm is homogeneous with respect to stretching of the domain, we may assume without loss of generality that $H = 1$. Because the mappings T_i preserve the $\mathrm{W}^{1,2}$ semi-norm with uniform equivalence constants, we obtain equivalence of norms

$$c_1 \| \mathbf{v}_i \|_{\mathrm{S}_i} \leqslant \| \mathrm{h}(\mathbf{v}_i) \|_{1,2,\Omega_i} \leqslant c_2 \| \mathbf{v}_i \|_{\mathrm{S}_i}, \qquad \text{for all } v_i \in \mathrm{U}_i \tag{26}$$

with constants $c_1 > 0, c_2 < \infty$ independent of h.

To estimate C in (19), let $\mathbf{u}_i \in \mathrm{U}_i$. It follows from Reference 5, Theorem 2 that

$$\frac{\Sigma_{j=1}^{k} \| \mathrm{h}(\overline{\mathbf{N}}_j^{\mathrm{T}}\overline{\mathbf{N}}_i\mathbf{D}_i\mathbf{u}_i) \|_{1,2,\Omega_j}^2}{\| \mathrm{h}(\mathbf{u}_i) \|_{1,2,\Omega_i}^2} \leqslant \frac{\text{const.}}{l^2}\left(1 + \ln \frac{l}{h}\right)^2 \tag{27}$$

where l is the number of subdomains Ω_j on which $h(\bar{\mathbf{N}}_j^T\mathbf{D}_i\mathbf{u}_i) \neq 0$. But these are exactly all subdomains neighbouring Ω_i, and their number is bounded as a result of our assumptions on T_j. This fact, together with the bound (27) and the equivalence of norms (26), concludes the proof. □

Note that the key estimate (27) could be also proved using the techniques developed in References 1 and 10.

4. COMPUTATIONAL RESULTS

Consider the model problem

$$-\Delta\mathbf{u} = \mathbf{f} \text{ in } \Omega, \qquad \mathbf{u} = 0 \text{ on } \Gamma, \qquad \frac{\partial \mathbf{u}}{\partial n} = 0 \text{ on } \partial\Omega\backslash\Gamma$$

The subdomains Ω_i are unit squares and the domain Ω is the rectangle $\Omega = (0, n_1) \times (0, n_2)$, consisting of $n_1 \times n_2$ subdomains. Γ is one of the sides of length n_1. Discretization by linear finite elements on a uniform mesh, equivalent to the five-point difference formula was used. The right-hand side was generated randomly. The results are summarized in Table I. The condition number was calculated from the relation between conjugate gradients and the Lanczos algorithm, cf. Reference 4.

The purpose of our computational tests was to demonstrate the fast convergence of the present method on simple problems. For this reason, our prototype implementation was written in the matrix package CLAM,[20] and we have calculated the Schur complements and calculated their pseudoinverses explicitly. Consequently, we do not report CPU times.

It should be noted that, in a more practical implementation, the Schur complements may not be formed explicitly; rather, a Neumann and a Dirichlet problem is to be solved for each subdomain in each step, cf. Reference 5. Experience with efficient implementations will be reported elsewhere, along with further developments of the method and the theory.

Fortran 77 code that implements the method is available as BDD from MGNET by anonymous ftp from casper.cs.yale.edu. The code invokes user-supplied subroutines that implement the matrix–vector multiplications $\mathbf{S}_i\mathbf{x}_i$ and solution of the possibly singular systems $\mathbf{S}_i\mathbf{z}_i = \mathbf{r}_i$.

Table I. Condition numbers for the balancing preconditioner

$n_1 \times n_2$	2×2	2×4	2×8	4×2	4×4	4×8	8×8	8×2	16×2	32×2
$h = 1/10$	1·30	1·42	1·44	2·64	2·74	2·74	3·04	2·99	3·10	3·11
$h = 1/20$	1·51	1·67	1·79	3·48	3·60	3·60	3·97	3·90	4·02	4·02
$h = 1/40$	1·76	1·98	2·03	4·49	4·62	4·62	5·05	4·98	5·12	5·15

5. ACKNOWLEDGEMENTS

Sincere thanks are due to Patrick Le Tallec, Olof Widlund and Barry Smith for bringing this problem to the author's attention and for discussions at the Conference on Domain Decomposition in Lambrecht, Germany, in September 1991, and to Steve McCormick for reading an early version of this paper. The author would also like to thank Marian Brezina for his assistance with programming the CLAM computations, and for proof-reading. This research was supported by NSF grant DMS-9015259.

REFERENCES

1. J. H. Bramble, J. E. Pasciak and A. H. Schatz, 'The construction of preconditioners for elliptic problems by substructuring, IV, Math. Comput., 53, 1–24 (1989).
2. M. Dryja and O. B. Widlund, 'Towards a unified theory of domain decomposition algorithms for elliptic problems', in Third International Symposium on Domain Decomposition Methods for Partial Differential Equations, held in Houston, Texas, 20–22 March 1989, T. Chan, R. Glowinski, J. Périaux and O. Widlund (Eds.), SIAM, Philidelphia, PA, 1990.
3. P. Concus, G. H. Golub and D. P. O'Leary, 'A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations', in *Sparse Matrix Computations*, J. R. Bunch and D. J. Rose (Eds.), Academic Press, 1976, pp. 309–332.
4. G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, (2 edn.), 1989.
5. Y.-H. De Roeck and P. Le Tallec, 'Analysis and test of a local domain decomposition preconditioner', in Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations, R. Glowinski, Y. Kuznetsov, G. Meurant, J. Périaux and O. Widlund (Eds.), SIAM, Philadelphia, PA, 1991.
6. J.-F. Bourgat, R. Glowinski, P. Le Tallec and M. Vidrascu, 'Variational formulation and algorithm for trace operator in domain decomposition calculations', in T. Chan, R. Glowinski, J. Périaux and O. Widlund (Eds.), *Domain Decomposition Methods*, Philadelphia, PA, 1989, SIAM.
7. R. Glowinski and M. F. Wheeler, 'Domain decomposition and mixed finite element methods for elliptic problems', in First International Symposium on Domain Decomposition Methods for Partial Differential Equations, R. Glowinski, G. H. Golub, G. A. Meurant and J. Périaux (Eds.), Philadelphia, PA, 1988, SIAM.
8. M. Dryja and O. B. Widlund, 'Additive Schwarz methods for elliptic finite element problems in three dimensions', Computer Science Technical Report 570, Courant Institute of Mathematical Sciences, 1991.
9. P. Le Tallec, Y.-H. De Roeck and M. Vidrascu, 'Domain decomposition methods for large linearly elliptic three dimensional problems', Tech. Report TR/PA/90/20, Centre Européen de Recherche et de Formation Avancée en Calcul Scientifique, Toulouse, France, 1990.
10. J. H. Bramble, J. E. Pasciak and A. H. Schatz, 'The construction of preconditioners for elliptic problems by substructuring', I, *Math. Comput.*, **47**, 103–134 (1986).
11. J. Mandel, 'Two-level domain decomposition preconditioning for the p-version finite element method in three dimensions', *Int. j. numer. methods eng.*, **29**, 1095–1108 (1990).
12. B. F. Smith, 'A domain decomposition algorithm for elliptic problems in three dimensions,' Tech. Report 519, Department of Computer Science, Courant Institute, October 1990, to be published in *Numer. Math.*
13. B. F. Smith, 'An optimal domain decomposition preconditioner for the finite element solution of linear elasticity problems', *SIAM J. Sci. Stat. Comput.*, **13** (1991).
14. O. Axelsson and I. Gustafsson, 'Preconditioning and two-level multigrid methods of arbitrary degree of approximation', *Math. Comput.*, **40**, 219–242 (1983).
15. C. Farhat, 'A multigrid-like semi-iterative algorithm for the massively parallel solution of large scale finite element systems', in Proceedings of the Fourth Copper Mountain Conference on Multigrid Methods, J. Mandel, S. F. McCormick, J. E. Dendy, Jr., C. Farhat, G. Lonsdale, S. V. Parter, J. W. Ruge and K. Stüben (Eds.), Philadelphia, 1989, SIAM, pp. 171–180.
16. M. Kočvara and J. Mandel, 'A multigrid method for three-dimensional elasticity and algebraic convergence estimates', *Appl. Math. Comput.*, **23**, 121–135 (1987).
17. C. Farhat, 'A saddle-point principle domain decomposition method for the solution of solid mechanics problems', in Domain Decomposition Methods, Proceedings, Norfolk 1991, T. Chan, R. Glowinski, J. Périaux and O. Widlund (Eds.), to be published.
18. C. Farhat and F.-X. Roux, 'An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems,' *SIAM J. Sci. Stat. Comput.*, **13**, 379–396 (1992).
19. J. Mandel, S. McCormick and R. Bank, 'Variational multigrid theory', in *Multigrid Methods*, S. F. McCormick (Ed.), SIAM, Philadelphia, 1987, Chap. 5, pp. 131–177.
20. Scientific Computing Associates, *CLAM User's Guide; The Computational Linear Algebra Machine, Version 2.00*, New Haven, CT, 1990.