NASA Contractor Report 178122

ICASE REPORT NO. 86-30

ICASE

ON SUBSTRUCTURING ALGORITHMS AND SOLUTION TECHNIQUES FOR THE NUMERICAL APPROXIMATION OF PARTIAL DIFFERENTIAL EQUATIONS

M. D. Gunzburger
R. A. Nicolaides

Contract No. NAS1-18107
May 1986

INSTITUTE FOR COMPUTER APPLICATIONS IN SCIENCE AND ENGINEERING
NASA Langley Research Center, Hampton, Virginia 23665

Operated by the Universities Space Research Association

National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665

ON SUBSTRUCTURING ALGORITHMS AND SOLUTION TECHNIQUES FOR THE NUMERICAL APPROXIMATION OF PARTIAL DIFFERENTIAL EQUATIONS

M. D. Gunzburger
Carnegie-Mellon University

R. A. Nicolaides
Carnegie-Mellon University

Dedicated to Milton E. Rose
on Occasion of his 60th Birthday

ABSTRACT

Substructuring methods are in common use in structural mechanics problems where typically the associated linear systems of algebraic equations are positive definite. Here these methods are extended to problems which lead to nonpositive definite, nonsymmetric matrices. The extension is based on an algorithm which carries out the block Gauss elimination procedure without the need for interchanges even when a pivot matrix is singular. Examples are provided wherein the method is used in connection with finite element solutions of the stationary Stokes equations and the Helmholtz equation, and dual methods for second-order elliptic equations.

Support for the first author was provided by the Air Force Office of Scientific Research under Grant No. AFOSR-83-0101 and by the Army Research Office under Contract No. DAAG-29-83-4-0084. The second author was supported under AFOSR Grant No. AFOSR-83-0231. Additional support was provided by the National Aeronautics and Space Administration under NASA Contract No. NAS1-18107.

1. THE SUBSTRUCTURING ALGORITHM IN THE POSITIVE DEFINITE CASE

The use of substructuring techniques in the numerical solution of problems governed by positive definite partial differential equations is in widespread use. The most notable case is found in structural mechanics, especially in connection with the equations of linear elasticity. For the sake of simplicity, here we describe the technique for the Dirichlet problem for the Poisson equation. Specifically, suppose we want to solve

$$\begin{aligned} -\Delta u &= f \quad \text{in} \quad \Omega \\ u &= 0 \quad \text{on} \quad \partial\Omega \end{aligned} \tag{1}$$

where Ω is, say, an open bounded region in $\mathbb{R}^2$ with boundary $\partial\Omega$. We subdivide the region Ω into open subregions Ω_i, $i = 1,\cdots,m$, such that $\overline{\Omega} = \bigcup_{i=1}^{m} \overline{\Omega}_i$ and $\Omega_i \cap \Omega_j = 0$ for $i \neq j$. We denote by Γ_{ij}, $1 \leq i < j \leq m$ the interfaces between regions Ω_i and Ω_j, i.e., $\Gamma_{ij} = \overline{\Omega}_i \cap \overline{\Omega}_j$. Of course, for particular choices of i and j in a given subdivision, Γ_{ij} may be empty. A sketch of a particular example with $m = 5$ is given in Figure 1.

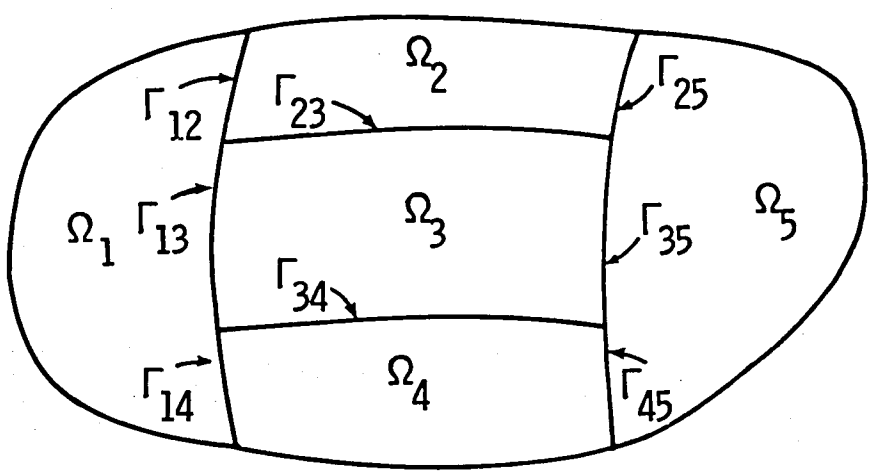

Figure 1. A subdivision of a region into five subregions.

We also subdivide Ω into a finite difference or finite element grid which in practice is much finer than the above subdivision of Ω into m subregions. We choose the two subdivisions so that the interfaces Γ_{ij} coincide with edges of the finite difference or finite element cells. The discretization of (1) proceeds in the usual manner. The essence of the substructuring algorithm is found in the particular choice for the ordering of the unknowns and equations, i.e., columns and rows, in the linear system resulting from the discretization of (1). Specifically, all unknowns and equations associated with the interior of a substructure Ω_i are numbered sequentially, one substructure at a time, and unknowns and equations associated with the interfaces Γ_{ij} are grouped together and numbered last. For example, in a typical finite difference discretization of (1), one associates equations and unknowns with nodes in the grid. In this case, we would group together all the unknowns in subregion Ω_1 together and number them first, then proceed to Ω_2, etc., and finally to Ω_m. Then we would number all the unknowns along the interfaces Γ_{ij}, $1 \leq i, j \leq m$. The equations would be numbered in the same way.[1] Likewise, in a finite element discretization of (1), some unknowns (trial functions) and equations (test functions) are associated with nodes or edges and these are

---

[1]The subdivision and numbering method described here applies to difference methods with stencils involving only nearest neighbors. The method may be extended in an obvious manner, e.g., by defining the interfaces to be more than one grid point in thickness, to methods having stencils with a greater degree of connectivity.

numbered in the same manner as in the finite difference case above.[2] In addition there may be test and trial functions more naturally associated with the finite elements themselves, and the equations and unknowns associated with these functions are grouped together with the other ones associated with the interior of the corresponding subregion Ω_i.

The net result of the above numbering schemes is that the linear system resulting from the discretization of (1) has the form

$$\begin{pmatrix} A_1 & & & & B_1 \\ & A_2 & & & B_2 \\ & & \ddots & & \vdots \\ & & & A_m & B_m \\ C_1 & C_2 & \cdots & C_m & A_0 \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ \vdots \\ U_m \\ U_0 \end{pmatrix} = \begin{pmatrix} F_1 \\ F_2 \\ \vdots \\ F_m \\ F_0 \end{pmatrix}. \qquad (2)$$

In (2), the matrices A_i, $i = 1,\cdots,m$, in the finite element case, result in the case of both the test and trial functions being associated with the interior of the subregion Ω_i, $i = 1,\cdots,m$, respectively, while the matrix A_0 results from test and trial functions associated with the interfaces Γ_{ij}, $1 \leq i < j \leq m$. The matrices C_i, and B_i, represent trial, respectively test,

---

[2]Again, the method described here applies to the case where the test and trial functions vanish outside the elements which contain the associated node or edge. However, by defining the interfaces to be one or more elements thick, the method may be easily extended to other cases, e.g., cubic B-spline test and trial functions.

functions associated with the interior of Ω_i and test, respectively trial, functions associated with the interfaces. The vectors U_i, $i = 1,\cdots,m$, respectively denote the unknowns associated with the interior of Ω_i, $i = 1,\cdots,m$, while U_0 denotes the unknowns associated with the interfaces. All of these associations can also be made in the finite difference case.

It is well-known that the coefficient matrix of the linear system (2), resulting from a discretization of (1), is symmetric and positive definite. Indeed, $A_i = A_i^T$, $B_i = C_i^T$ for $i = 1,\cdots,m$ and $A_0 = A_0^T$. It is also easy to see that the matrices A_i, $i = 1,\cdots,m$, are themselves positive definite. In fact, these matrices are exactly the ones which would result from the analogous discretization of the problems

$$\begin{aligned} \Delta u &= f \quad \text{in} \quad \Omega_i \\ u &= 0 \quad \text{on} \quad \partial\Omega_i \end{aligned} \tag{3}$$

for $i = 1,\cdots,m$, where $\partial\Omega_i$ denotes the boundary of Ω_i. Note that this boundary may consist of both interfaces and a portion of the boundary $\partial\Omega$ of Ω, as is the case for Ω_1, Ω_2, Ω_4, and Ω_5 in Figure 1, or may consist wholly of interfaces as is the case for Ω_3 in that figure. Discretization of (3) results in a linear system with a coefficient matrix A_i, and thus A_i is clearly symmetric and positive definite. We note that even in the case of the Neumann problem, i.e., the boundary condition in (1) is replaced by $\partial u/\partial n = 0$ on $\partial\Omega$, the matrices A_i in (2) would still be, at least in the finite element case, symmetric and positive definite.[3] This is so because the problem (3) associated with the matrix A_i is now given by

$$\Delta u = f \quad \text{in} \quad \Omega_i$$

$$\frac{\partial u}{\partial n} = 0 \quad \text{on} \quad \partial\Omega_i \cap \partial\Omega \tag{4}$$

$$u = 0 \quad \text{on} \quad \partial\Omega_i \cap \Gamma_{ij}, \; j = 1,\cdots,m,$$

where we have set $\Gamma_{ij} = \Gamma_{ji}$. Since $\partial\Omega_i \cap \Gamma_{ij}$ is never empty, the matrix A_i associated with (4) is symmetric and positive definite.[4]

With the matrices A_i, $i = 1,\cdots,m$, being positive definite, one may proceed to solve (2) by a block elimination procedure. Symbolically, we may express the first m stages of this procedure by the relations

$$U_i = A_i^{-1}(F_i - B_i U_0), \qquad i = 1,\cdots,m, \tag{5}$$

which uniquely express U_i in terms of data and the interface unknowns U_0. The last stage of the process requires the solution of the linear system

$$DU_0 = G \tag{6}$$

where

$$D = A_0 - \sum_{i=1}^{m} C_i A_i^{-1} B_i \quad \text{and} \quad G = F_0 - \sum_{i=1}^{m} C_i A_i^{-1} F_i. \tag{7}$$

---

[3] If on $\partial\Omega_i \cap \partial\Omega$ something other than Dirichlet data is specified, then the matrix A_i also contains rows and columns associated with test and trial functions associated with nodes or edges on that portion of the boundary.

[4] Of course, the fact that A_i, $i = 0,\ldots,m$, are positive definite may be deduced directly from the fact the coefficient matrix of (2) is positive definite, i.e., the former is a necessary condition for the latter.

Of course, in (5) and (7) the inverses are not explicitly computed, but rather appropriate linear systems are solved. The solvability of the system (6) follows whenever the system (2) is solvable. In fact, if the system (2) is positive definite, so is the matrix D [1]. Once (6) is solved for U_0, (5) yields U_i, $i = 1,\cdots,m$.

Although we have described the substructuring algorithm in the context of the Poisson equation, the method can be applied in a similar manner to any positive definite problem. As noted above, the method has encountered great success in structural mechanics problems. However, in other fields where the governing equations are not positive definite or symmetric one may still order the equations and unknowns to produce linear systems such as (2), but these may not always be solved by a standard block elimination procedure. In the next two sections we describe a procedure to solve (2) even in the case of the matrices A_i being singular and show how the method may be implemented through an elimination procedure. In Section 4 we describe examples which lead to singular matrices A_i. Finally, in Section 5 we give some concluding remarks.

Incidentally, in almost all situations the use of a properly implemented substructuring algorithm will result in savings in computational costs when compared to a banded elimination procedure. For example, consider a discretization of Poisson's equation on a unit square. Suppose we have M subregions in each direction so that $m = M^2$ and suppose that each subregion is further subdivided by introducing an $n \times n$ grid. Thus, there are a total of Mm points in each direction. Banded elimination requires $O(M^4 n^4)$ operations, while the above substructuring algorithm can be implemented in, at most, $O(Mn^4 + M^4 n^3)$ operations. We note that this particular problem is not

particularly well-suited for substructuring methods. Also, the relative advantage of substructuring is greater when one considers three-dimensional problems or systems of partial differential equations.

We also note that substructuring ideas in connection with preconditioning techniques have been discussed in [2].

2. THE SOLUTION ALGORITHM IN THE GENERAL CASE

We begin by describing a method for solving (2) in the case where the matrices A_i are singular. The algorithm described here is a special case of a more general algorithm which applies to arbitrary matrices with arbitrary subdivisions into blocks, e.g., the matrix has no special structure and the matrices A_i may not only be singular, but may even be rectangular. The more general algorithm is described in [3]. We will describe the algorithm as applied to (2) and we will make use of pseudo-inverses in order to simplify the initial presentation. However, we emphasize that the algorithm may be implemented without the need for the explicit calculation of any pseudo-inverses; such an implementation is discussed in the next section. This is similar to the observation that the algorithm contained in (5)-(7) may be implemented without explicitly computing any inverses, e.g., by solving linear systems.

The system (2) is equivalent to

$$A_i U_i + B_i U_0 = F_i. \qquad i = 1,\cdots,m, \tag{8}$$

$$\sum_{i=1}^{m} C_i U_i + A_0 U_0 = F_0. \tag{9}$$

Now, U_i may be orthogonally decomposed in the form

$$U_i = Y_i + Z_i, \qquad i = 1,\cdots,m, \tag{10}$$

where

$$A_i Z_i = 0, \qquad i = 1,\cdots,m, \tag{11}$$

and Y_i is orthogonal to all vectors satisfying (11). In particular,

$$Y_i^T Z_i = 0, \qquad i = 1,\cdots,m. \tag{12}$$

Substitution of (10)-(11) into (8) yields that

$$A_i Y_i = F_i - B_i U_0, \qquad i = 1,\cdots,m. \tag{13}$$

Since Y_i is orthogonal to the null space of A_i, (13) yields that

$$Y_i = A_i^+(F_i - B_i U_0), \qquad i = 1,\cdots,m, \tag{14}$$

where A_i^+ denotes the pseudo-inverse of A_i. This relation states that Y_i is uniquely determined from the data and U_0. Note that (8) yields no information concerning Z_i as is to be expected since $A_i Z_i = 0$. Substituting (10) and (14) into (9) yields that

$$DU_0 = G - \sum_{i=1}^{m} C_i Z_i \tag{15}$$

where

$$D = A_0 - \sum_{i=1}^{m} C_i A_i^+ B_i \quad \text{and} \quad G = F_0 - \sum_{i=1}^{m} C_i A_i^+ F_i. \tag{16}$$

We may also decompose U_0 in the form

$$U_0 = Y_0 + Z_0 \tag{17}$$

where

$$DZ_0 = 0 \tag{18}$$

and Y_0 is orthogonal to all vectors satisfying (18). In particular,

$$Y_0^T Z_0 = 0. \tag{19}$$

Substitution of (17)-(18) into (15) yields that

$$DY_0 = G - \sum_{i=1}^{m} C_i Z_i \tag{20}$$

and, since Y_0 is orthogonal to the null space of D, (20) yields that

$$Y_0 = D^+(G - \sum_{i=1}^{m} C_i Z_i). \tag{21}$$

Again, it is not surprising that (15) yields no information concerning Z_0. Substitution of (17) and (21) into (14) then yields that

$$Y_i = A_i^+[F_i - B_i D^+(G - \sum_{j=1}^{m} C_j Z_j)] - A_i^+ B_i Z_0 \tag{22}$$

for $i = 1,\cdots,m$.

At this point we have shown that Y_i, $i = 0,\cdots,m$, may be uniquely expressed in terms of Z_i, $i = 0,\cdots,m$, by (21) and (22). It remains to show how to find the latter. The first step is to multiply (13) by $(I - A_i A_i^+)$. Since $A_i A_i^+ A_i = A_i$, we have that

$$(I - A_i A_i^+)(F_i - B_i U_0) = 0, \qquad i = 1,\cdots,m,$$

or substituting (17) and (21),

$$(I - A_i A_i^+)[F_i - B_i Z_0 - B_i D^+(G - \sum_{j=1}^{m} C_j Z_j)] = 0, \qquad i = 1,\cdots,m. \tag{23}$$

Now suppose we are able to determine bases for the null spaces of A_i, $i = 1,\cdots,m$, and D. We collect each of these basis sets into matrices N_i, $i = 0,\cdots,m$, i.e., N_i, $i = 0,\cdots,m$, have linearly independent columns,

$$DN_0 = 0 \quad \text{and} \quad A_i N_i = 0, \qquad i = 1,\cdots,m, \tag{24}$$

and the columns of N_0, respectively N_i, span the null space of D, respectively A_i, $i = 1,\cdots,m$. The number of columns in N_i is, of course, the dimension of the corresponding null spaces. Now, we may write that

$$Z_i = N_i \Lambda_i, \qquad i = 0,\cdots,m, \tag{25}$$

for some vectors Λ_i. Substituting (25) into (23) then yields that

$$\sum_{j=1}^{m} R_{ij} \Lambda_j = H_i, \qquad i = 1,\cdots,m, \tag{26}$$

where

$$R_{i0} = (I - A_i A_i^+)B_i N_0, \qquad H_i = (I - A_i A_i^+)[F_i - B_i D^+ G]$$

and (27)

$$R_{ij} = (I - A_i A_i^+)B_i D^+ C_j N_j, \qquad j = 1,\cdots,m.$$

Now letting

$$R = \begin{pmatrix} R_{10} & R_{11} & \cdots & R_{1m} \\ R_{20} & R_{21} & \cdots & R_{2m} \\ \vdots & \vdots & & \vdots \\ R_{m0} & R_{m1} & & R_{mm} \end{pmatrix}, \quad H = \begin{pmatrix} H_1 \\ H_2 \\ \vdots \\ H_m \end{pmatrix} \text{ and } \Lambda = \begin{pmatrix} \Lambda_0 \\ \Lambda_1 \\ \vdots \\ \Lambda_m \end{pmatrix}, \qquad (28)$$

(26) may be expressed in the form

$$R\Lambda = H. \qquad (29)$$

In general, R is a rectangular matrix. The number of rows in R is equal to the sum of the number of rows of the matrices A_i, $i = 1,\cdots,m$, and the number of columns of R is equal to the sum of the dimensions of the null spaces of A_i, $i = 1,\cdots,m$; and D. It can be shown [3] that the system (29) is a consistent system, and we may find its solution, for example, by forming

$$(R^T R)\Lambda = R^T H. \qquad (30)$$

Suppose we can solve (30) for Λ. Then (28) yields Λ_i, $i = 0,\cdots,m$, (25) then yields Z_i, $i = 0,\cdots,m$, (21) and (22) yields Y_i, $i = 0,\cdots,m$, and finally (10) and (17) yield the solution U_i, $i = 0,\cdots,m$, of (2).

The algorithm described here is related in the following manner to the block elimination algorithm in Section 1. Suppose that the matrix of (2) and all the A_i's and D are nonsingular. Then, the algorithm of this section reduces to the standard block Gauss elimination procedure. Indeed, in this case, $A_i^+ = A_i^{-1}$, $D^+ = D^{-1}$ and $Z_i = 0$ so that $U_i = Y_i$ and the latter are determined uniquely by (14) and (21). Note the correspondence, in this case, between (14)-(15) and (5)-(6).

In the more general case, i.e., some or all of the A_i's and D being singular, it can be shown [3] that the rank deficiency of (30) is exactly that of the original coefficient matrix in (2). Therefore, if the latter is nonsingular, then so is R^TR and then Λ in (30) is uniquely determined. Since the Z_i's and Y_i's are uniquely determined from Λ, the algorithm produces the unique solution of (2). If the matrix of (2) is singular, so is R^TR and (30) does not have a unique solution. However, (30) may be solved anyway, either in terms of arbitrary parameters or by adding constraints. The number of parameters or constraints is equal to the dimension of the null space of R^TR which in turn is the same as the dimension of the null space of the coefficient matrix in (2). In any case, once a particular Λ is determined, then Z_i and Y_i are also determined.

In particular applications to the solution of partial differential equations, the dimension of the system (30) is small compared to that of the system (2). Indeed, typically $\dim(R^TR) = 0(m)$, the number of subregions. For example, the dimension of the null spaces of the matrices A_i and D may be one or zero, in which case $\dim(R^TR) \leq m + 1$.

3. AN ELIMINATION IMPLEMENTATION

We begin by restating the algorithm of the previous section. Given the matrices $A_0,\cdots,A_m$, $B_1,\cdots,B_m$, $C_1,\cdots,C_m$ and the vectors $F_0,\cdots,F_m$, we find vectors $U_0,\cdots,U_m$ satisfying (2) by the following procedure.

1. Compute $A_i^+ F_i$ and $A_i^+ B_i$ for $i = 1,\cdots,m$.
2. Compute N_i, $i = 1,\cdots,m$, whose columns constitute a basis for the null space of A_i, $i = 1,\cdots,m$, respectively.
3. Compute $C_i(A_i^+ B_i)$, $C_i(A_i^+ F_i)$ and $C_i N_i$ for $i = 1,\cdots,m$.
4. Compute $D = A_0 - \sum_{i=1}^{m} C_i(A_i^+ B_i)$ and $G = F_0 - \sum_{i=1}^{m} C_i(A_i^+ F_i)$.
5. Compute $D^+ G$.
6. Compute N_0 whose columns constitute a basis for the null space of D.
7. Compute $D^+ C_i N_i$ for $i = 1,\cdots,m$.
8. Compute the matrices

$$R_{i0} = B_i N_0 - A_i (A_i^+ B_i)N_0 \quad \text{for} \quad i = 1,\cdots,m,$$

$$R_{ij} = B_i(D^+ C_j N_j) - A_i(A_i^+ B_i)(D^+ C_j N_j) \quad \text{for} \quad i,j = 1,\cdots,m$$

and the vectors

$$H_i = F_i - B_i(D^+ G) - A_i(A_i^+ F) + A_i(A_i^+ B_i)(D^+ G) \quad \text{for} \quad i = 1,\cdots,m.$$

9. Assemble the results of step 6 into the matrix R and vector H according to (28) and then compute $R^T R$ and $R^T H$.
10. Solve the linear system $R^T R\Lambda = R^T H$ for Λ and then compute Λ_i, $i = 0,\cdots,m$, according to the partition of (28).
11. Compute $Z_i = N_i \Lambda_i$ for $i = 0,\cdots,m$.

12. Compute $Y_0 = (D^+ G) - \sum_{i=1}^{m} C_i N_i \Lambda_i = (D^+ G) - \sum_{i=1}^{m} C_i Z_i$.

13. Compute $U_0 = Y_0 + Z_0$.

14. Compute $Y_i = (A_i^+ F_i) - (A_i^+ B_i)U_0$ for $i = 1,\cdots,m$.

15. Compute $U_i = Y_i + Z_i$ for $i = 1,\cdots,m$.

Other than steps 1, 2, 5, 6, and 10, the above algorithm requries only matrix and matrix-vector multiplications. In this section we show how to carry out the other operations required by the algorithm through an elimination procedure. In particular, we will not need to explicitly calculate any pseudo-inverses of matrices.

We first describe how to carry out steps 1 and 2. Consider the linear system.

$$A_i Q = S = (B_i, F_i, 0) \tag{31}$$

where the right-hand side matrix S consists of the matrix B_i, the vector F_i, and some additional columns of zeroes. The number of these additional columns should be greater or equal to the dimension of the null space of A_i. This dimension will actually be determined during the elimination procedure.[5] We now proceed to solve (31) by Gauss elimination with partial pivoting. If the matrix A_i is singular, then one or more times during the elimination procedure we will not be able to locate a nonzero pivot element. In fact, the number of times this occurs is exactly the dimension of the null space of A_i. However, at such an occurrence, the corresponding column is

---

[5]See Section 5 concerning the effects that roundoff errors may have on the determination of this dimension.

already in the eliminated form so that we may skip over to the next column and continue the elimination process. At the end of the process, (31) has been reduced to the form

$$\bar{A}_i \, Q = J = (\bar{B}_i, \bar{F}_i, 0) \tag{32}$$

where $\bar{A}_i$ is upper triangular and in row echelon form. When A_i is singular, $\bar{A}_i$ will have zeros at the pivot location for exactly those columns for which no nonvanishing pivot element was found.

We now proceed to backsolve (32). No difficulty is encountered until a row is reached for which the pivot entry of $\bar{A}_i$ is zero. For the columns of Q corresponding to B_i and F_i, we may arbitrarily set (to something other than zero) the entry in the row corresponding to the zero pivot of $\bar{A}_i$. Then the backsolve procedure may continue until we reach another zero pivot entry, at which time we again arbitrarily specify an entry in the columns of Q corresponding to the columns B_i and F_i of S. While all this is going on we are also solving (32) for the columns corresponding to the zero columns of S. For these columns, whenever a zero pivot entry is encountered in $\bar{A}_i$, one of the elements in the corresponding row is set to one while the rest are set to zero. Each time a zero pivot entry is encountered, a different column is chosen for which one sets the arbitrary element to one. At the end of this backsolve procedure, (32) yields that

$$Q = (\hat{L}, \hat{K}, N_i).$$

Here the columns of N_i form a basis for the null space of A_i and $\hat{L}$ and $\hat{K}$ are particular solutions of the systems.

$$A_i L = B_i \quad \text{and} \quad A_i K = F_i. \tag{33}$$

The final step is to orthogonalize the columns of $\hat{L}$ and $\hat{K}$ with respect to the columns of N_i to yield

$$\tilde{Q} = (\tilde{L},\tilde{K},N_i).$$

Since $A_iN_i = 0$, $\tilde{L}$ and $\tilde{K}$ are still solutions of (33). Moreover, the columns of $\tilde{L}$ and $\tilde{K}$ are orthogonal to the null space of A_i and, therefore, are minimum norm solutions. By the uniqueness of the minimum norm solution, we have that

$$\tilde{L} = A_i^+ B_i \quad \text{and} \quad \tilde{K} = A_i^+ F_i.$$

Thus the above elimination procedure has accomplished the tasks of steps 1 and 2 of the algorithm.

The tasks of steps 5, 6, and 7 can be accomplished in an analogous manner. Also, if the matrix R^TR is nonsingular, then it may be easily solved by an ordinary Gauss elimination procedure. If it is singular then a solution in terms of arbitrary parameters may be determined in a manner similar to the above procedure for the system (31). We note that any sparsity or structure inherent in the matrices A_i may be exploited in the above procedure. However, in general, the matrix D will be dense. We will return to this point in the concluding section.

4. EXAMPLES

The Stationary Stokes Equation

Consider the stationary Stokes equations for the slow flow of a viscous fluid in a bounded region in $\mathbb{R}^2$. These are given by

$$\begin{aligned} \Delta \underline{u} - \text{grad } p &= \underline{f} \quad \text{in} \quad \Omega \\ \text{div } \underline{u} &= 0 \quad \text{in} \quad \Omega \\ \underline{u} &= 0 \quad \text{on} \quad \partial\Omega. \end{aligned} \tag{34}$$

Here $\underline{u}$ denotes the velocity, p the pressure, $\underline{f}$ the given body force and the viscosity coefficient has been absorbed into p and $\underline{f}$. Clearly, the pressure cannot be determined uniquely since we may add an arbitrary constant to the pressure and still satisfy (34).

A finite element approximation of the solution $(\underline{u},p)$ of (34) may be defined as follows. Given finite-dimensional spaces V^h and S^h for the discrete velocity and pressure fields, we seek $\underline{u}^h \in V^h$ and $p^h \in S^h$ such that

$$\begin{aligned} &\int_\Omega (\text{grad } \underline{u}^h : \text{grad } \underline{v}^h - p^h \text{ div } \underline{v}^h) d\Omega = -\int_\Omega \underline{f} \cdot \underline{v}^h \, d\Omega \quad \text{for all} \quad \underline{v}^h \in V^h \\ &\int_\Omega q^h \text{ div } \underline{u}^h \, d\Omega = 0 \quad \text{for all} \quad q^h \in S^h. \end{aligned} \tag{35}$$

Here we assume that the elements of V^h satisfy the boundary condition in (34). By choosing bases for the spaces V^h and S^h, (35) can be expressed as a linear algebraic system for the coefficients in the basis function expansions of $\underline{u}^h$ and p^h.

Now it is well-known that arbitrary choices of spaces V^h and S^h may not yield stable or accurate solutions. However, there are now known many element pairs for which (35) yields optimally accurate solutions [4], [5], [6]. One such pair is described as follows. Suppose S_h denotes a triangulation of the region Ω. We denote by V_h a finer triangulation derived from S_h by subdividing each triangle in S_h into four congruent triangles by joining the midsides. See Figure 2. We define S^h to consist of piecewise constant functions over the triangulation S_h

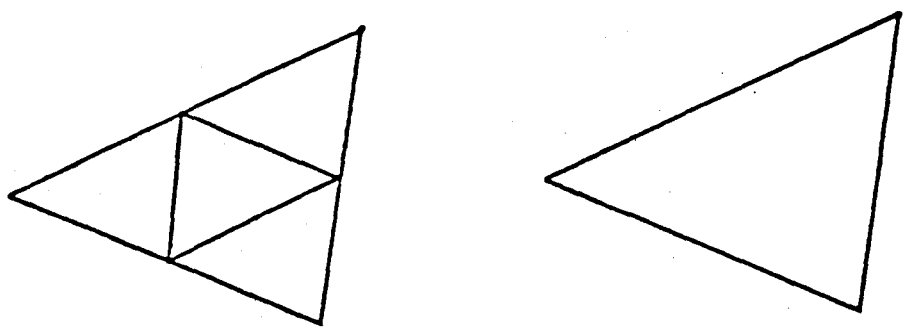

Figure 2. A triangle in S_h and the corresponding triangles in V_h.

and V^h to consist of piecewise linear functions over the triangulation V_h which are continuous over Ω and vanish on $\partial\Omega$. This combination is known to be stable and be optimally accurate [6].[6] The basis functions for V^h are easily associated with the vertices of the triangulation V_h while the basis functions for S^h are associated with the triangles in the triangulation S_h.

---

[6]See below for the necessary restriction on the pressure which yields this result.

Now let us consider a substructuring technique for the solution of (35). We assume that the interfaces Γ_{ij} between subregions are made up of edges of the triangulation S_h so that these interfaces do not cut across pressure triangles. One may easily arrange a numbering scheme for the unknowns and equations which yields a linear system of the form (2). For example, U_i consists of all velocity unknowns associated with vertices of V_h located in the interior of the subregion Ω_i and all pressure unknowns associated with the triangles of S_h which are also in Ω_i. Note that U_0 contains only velocity unknowns, namely those associated with vertices V_h which lie on the interfaces Γ_{ij} but not on $\partial\Omega$.

We have not constrained the pressure space and therefore the system (2) corresponding to this discretization of (34) is singular. In fact, its rank defficiency is one, and the null vector corresponds to the pressure function which is constant over Ω. On the other hand, the velocity approximation is uniquely determined by (2) [6]. Furthermore, it is easy to see that the submatrices $A_1,\cdots,A_m$ are singular. In fact, these matrices are exactly those which arise from the analogous discretization of the problem.

$$\Delta \underline{u} - \text{grad } p = \underline{f} \quad \text{in} \quad \Omega_i$$

$$\text{div } \underline{u} = f \quad \text{in} \quad \Omega_i$$

$$u = 0 \quad \text{on} \quad \partial\Omega_i .$$

Thus each of the matrices A_i has a single local pressure null vector, i.e., the dimension of N_i is one and N_i corresponds to the pressure function which is constant over Ω_i. On the other hand, since the velocity field can

be uniquely determined from (2) and since U_0 consists of only velocity unknowns, the matrix D in the linear system (15) is nonsingular, i.e., $N_0 = 0$. Thus, in this case, the system (30) has dimension m and has a one-dimensional null space, the latter following from the fact that the system (2) itself has a one-dimensional null space.

If we choose the pressure space S^h to consist of piecewise linear functions over the triangulation S_h which are continuous over Ω, while retaining the same velocity space, the situation changes drastically. For example, now the basis functions for S^h are more easily associated with the vertices of S_h. Now U_i contains pressure unknowns corresponding to vertices in S_h which are in the interior of Ω_i or lie on $\partial\Omega_i \cap \partial\Omega$. More important, U_0 now contains pressure unknowns associated with vertices of S_h which lie on Γ_{ij} but not on $\partial\Omega$. In this case the matrices A_i are nonsingular and the matrix D is singular with a one-dimensional null space.

The Helmholtz Equation

Now consider the problem

$$\begin{aligned} \Delta u + \lambda u &= f \quad \text{in} \quad \Omega \\ u &= 0 \quad \text{on} \quad \partial\Omega \end{aligned} \tag{36}$$

where λ is not near an eigenvalue of the operator $-\Delta$. Standard finite element or finite difference discretizations of (36) yield linear algebraic systems with coefficient matrices which are symmetric and indefinite, but which certainly may, by using a partial pivoting strategy, be stably inverted. Now consider the following specific situation. Let Ω be the

square $(0,\pi) \times (0,\pi)$ and let $\lambda = 13/4$. Since the eigenvalues of $-\Delta$ for this region are given by $(n^2 + m^2)$, $m,n = 1,2,\cdots$, we see that $\lambda = 13/4$ is not an eigenvalue and therefore the problem (36) leads to nonsingular coefficient matrices. Now, suppose we consider solving (36) by using the substructuring algorithm with the two subregions $\Omega_1 = (0,2\pi/3) \times (0,\pi)$ and $\Omega_2 = (2\pi/3,\pi) \Omega (0,\pi)$. Then the matrices A_i in (2) correspond to the coefficient matrix for the analogous discretization of the problem

$$\begin{aligned} \Delta u + \lambda u &= f \quad \text{in} \quad \Omega_i \\ u &= 0 \quad \text{on} \quad \partial\Omega_i. \end{aligned} \tag{37}$$

But the eigenvalues of $-\Delta$ for the region Ω_1 are given by $(n^2 + 9m^2/4)$, $m,n = 1,2,3,\cdots$, so that $\lambda = (13/4)$ is an eigenvalue of $-\Delta$ for the region Ω_1 and therefore the matrix A_1 is singular even though the system (2) is not.

Admittedly, this example is somewhat pathological in the sense that for random choices of regions, subregions, and parameters λ, the probability is zero that the matrices A_i in (2) will be singular. However, for particular choices of λ, Ω and Ω_i, one or more of the matrices A_i may be singular; after all, the above example is not really all that far-fetched. Of course, if any of the A_i's are singluar, the situation may be remedied by choosing a different subdivision of the region Ω; this in turn implies a complete reassembly of the coefficient matrix in (2). On the other hand, the algorithm of Sections 2 and 3 may be used whether or not any of the matrices A_i are singluar.

There is a small but nonvanishing probability that for some of the problems (37) λ, although not an eigenvalue of $-\Delta$ for the region Ω_i, is close to such an eigenvalue. If λ is close enough to such an eigenvalue, the matrix A_i, in finite precision arithmetic, may be mistakenly determined to be singular by the algorithm of Section 3. However, this will be the case only when the difference between λ and an eigenvalue is much smaller than the discretization error, i.e., of the order of the unit roundoff error of the machine, and no serious effect on the accuracy of the solution should result.

Dual Methods for Second-Order Elliptic Equations

For a third example, we consider dual methods for second-order elliptic partial differential equations. An example of these are methods based on the complementary energy principle in linear elasticity. For simplicity, we here consider the problem

$$\begin{aligned} u &= \nabla\phi \quad \text{in} \quad \Omega \\ \operatorname{div} \underline{u} &= f \quad \text{in} \quad \Omega \\ \underline{u}\cdot\underline{n} &= 0 \quad \text{on} \quad \Gamma_1 \end{aligned} \tag{38}$$

and

$$\phi = g \quad \text{on} \quad \Gamma_2$$

where again $\Gamma_1 \cap \Gamma_2 = \partial\Omega$ denotes the boundary of the bounded region $\Omega \subset \mathbb{R}^2$ and n denotes the unit outer normal to $\partial\Omega$. A finite element approximation of (38) may be obtained by choosing finite-dimensional spaces V^h and S^h and then seeking $\underline{u}^h \in V^h$ and $\phi^h \in S^h$ such that

$$\int_{\Omega}(\underline{u}^h\cdot\underline{v}^h + \phi^h \operatorname{div} \underline{v}^h)d\Omega = \int_{\Gamma_2} g\underline{v}^h\cdot\underline{n}d\Omega \quad \forall \underline{v}^h\in V^h$$

$$\int_{\Omega} \psi^h \operatorname{div} \underline{u}^h \, d\Omega = \int_{\Omega} f\psi^h \quad \forall \psi^h\in S^h.$$

We assume that the elements of V^h satisfy the boundary condition on Γ_1 in (38). The boundary condition on ϕ is natural in this formulation, which is one of its advantages.

In [7], the following choice of V^h and S^h was shown to yield stable and optimally accurate approximations, at least for polygonal domains. First, we subdivide Ω into quadrilaterals, and then subdivide each quadrilateral into four triangles by drawing the diagonals. For V^h we take all continuous piecewise linear vector fields with respect to the resulting triangulation and then define $S^h = \operatorname{div} V^h$. The resulting space S^h can be shown to be a subspace of all piecewise constants over the triangulation. See [7] for details.

In the implementation of the substructuring algorithm, we assume that the interfaces Γ_{ij} coincide with some of the edges of the quadrilaterals which initially defined our finite element triangulation of Ω, i.e., the interfaces do not cut through any of these quadrilaterals. The test and trial functions from V^h are associated with nodes while those from S^h are associated with the interior of the quadrilaterals. The matrices A_i in (2) now correspond to the discretization of the problem

$$\underline{u} = \nabla\phi \quad \text{and} \quad \operatorname{div} \underline{u} = f \quad \text{in} \quad \Omega_i$$

$$\underline{u}\cdot\underline{n} = 0 \quad \text{on} \quad \Gamma_1\cap\partial\Omega_i, \quad \phi = g \quad \text{on} \quad \Gamma_2\cap\partial\Omega_i \tag{39}$$

and

$$\underline{u} = 0 \quad \text{on} \quad \Gamma_{ij} \cap \partial\Omega_i .$$

Because of the last boundary condition, the problem (39) is over constrained insofar as the variable $\underline{u}$ us concerned. Nevertheless, if $\Gamma_2 \cap \partial\Omega_i = 0$, i.e., a given subregion does not have part of its boundary coincide with that part of $\partial\Omega$ on which data for ϕ are given, then the problem (39) can only determine ϕ to an additive constant. This, for example, would be the case for subregion Ω_3 in Figure 1, i.e., an interior subregion. For such situations, i.e., $\Gamma_2 \cap \partial\Omega_i = 0$, the matrix A_i in (2) will again be singular, with a one-dimensional null space. Since (38) always uniquely determines $\underline{u}$, the matrix D of (16) will be nonsingular. The rank deficiency of the system (30) will be one or zero, depending on whether or not Γ_2 has vanishing measure, i.e., whether or not the problem (38) uniquely determines ϕ.

5. CONCLUDING REMARKS

Determination fo Zero Pivot Elements

A crucial step in the elimination algorithm presented in Section 3 is the determination of when all the elements in a column to be eliminated are already zero. This is necessary for the determination of the null spaces of the matrices A_i and D. In practice one would declare an element to vanish whenever its magnitude is less than some prescribed tolerance which should be proportional to the unit roundoff error of the machine. This naturally leaves open the possiblity of a very small but nonzero element being mistaken for a vanishing element. This situation can be avoided, at least when one is

solving partial differential equations, by first using high enough precision arithmetic, e.g., 60 or 64 bit floating point arithmetic, and by making sure that the algorithms used are stable. The former is easily arranged, while the latter points out the importance of rigorous mathematics. Indeed, if an algorithm is stable, as are the ones discussed in Section 4, and the machine precision is high enough, one should not encounter nonzero elements which are comparable in magnitude to the unit roundoff error unless the matrix in hand is singular or very nearly singular.

An alternative to the use of elimination type procedures is, of course, to employ methods based on orthogonal transformations. At the price of greater computational expense, such methods are less susceptible to ill effects due to roundoff error.

Parallelism

One of the attractions of substructuring algorithms is the obvious inherent parallelism both in the assembly and solution stages. The sets of matrices and vectors (A_i,B_i,C_i,F_i), $i = 1,\cdots,m$, can each be assembled independently. Furthermore, at least in the finite element case, we may write the matrix A_0 and the vector F_0 in the form

$$A_0 = \sum_{i=1}^{m} A_{0i}, \qquad F_0 = \sum_{i=1}^{m} F_{0i} \tag{40}$$

where the matrix A_{0i} and the vector F_{0i} represent the contribution to the matrix A_0 and vector F_0 coming from region Ω_i. Each of the sets (A_{0i}, F_{0i}), $i = 1,\cdots,m$, may be assembled in parallel. Thus, in the assembly stage, the sets $(A_i,B_i,C_i,F_i,A_{0i},F_{0i})$, $i = 1,\cdots,m$, may be assembled in parallel.

For example, each of the above sets may be assembled on separate processors, with no need for interprocessor communications. At the end of the assembly process, the concatenations of (40) must be performed. This step is not parallelizable, but represents a minor portion of the assembly process.

There is also a large degree of parallelism in the solution algorithm described at the beginning of Section 3. Steps 1, 2, and 3 are completely parallelizable, again with no interprocessor communications necessary. Furthermore, if the appropriate information can be transferred to the processors, steps 7, 8, 11, 14, and 15 and a portion of step 12 can also be computed in parallel. The only relatively major steps which are not parallelizable are steps 5 and 6.

The issue of parallelism in connection with substructuring algorithms has been studied in [8] in the context of a specific three-dimensional positive definite problem. That paper contains a discussion of operation counts which, for the most part, is also relevant in the present context.

Three-Dimensional Problems

As pointed out above, the major nonparallel steps in the computation are embodied in steps 5 and 6 in the algorithm of Section 3. Even on a serial machine these steps may be costly since, in general, they involve dense matrices. In two-dimensional problems, by keeping the number of subregions relatively small compared to the total number of elements in the triangulation, the size of these dense calculations can be kept small, i.e., the size of D can be of the order of the square root of the size of the A_i's. The latter usually are sparse, e.g., banded. A similar arrangement in three-dimensional problems would, in general, lead to a matrix D whose size

is of the order of the two-thirds power of the size of the A_i's, which may be unacceptably large. Furthermore, in steps 1 and 2 of the algorithm, the number of right-hand sides would be approximately equal to the number of columns of D and the size of the A_i's may be too large, when relatively few subregions are used. Therefore, for three-dimensional problems one must be especially careful to implement the algorithm in an efficient manner as possible.

These potential difficulties can be mitigated in a variety of ways. For example, many of the right-hand sides in the computations of step 1 of the algorithm are zero because any column of B_i which corresponds to an interface unknown which is not associated with $\partial\Omega_i$ would vanish. The corresponding row of C_i is also zero. Thus, one can avoid computations involving linear systems with zero right-hand sides and multiplications by zero vectors. The savings possible, in storage and computing time, by accounting for these features are relatively higher for three-dimensional problems.

Although, in general, the number of interface variables may be large for three-dimensional problems, in practice it is often the case that specific features of the domain Ω lead to a small number of such unknowns. For instance, in a wing-fuselage configuration, it is natural to consider the wing and fuselage to be different subregions and the interface between these two substructures is relatively small in extent. Indeed, it was exactly in this type of application that the terminology "substructuring" arose.

Finally we consider the most serious problem, namely that of the size of the matrix D. However, even here a judicious implementation can effect great savings. As a simple illustration consider the subregion structure of Figure 3 where we have now labeled the interface boundaries by Γ_i, $i = 1,\cdots,m-1$.

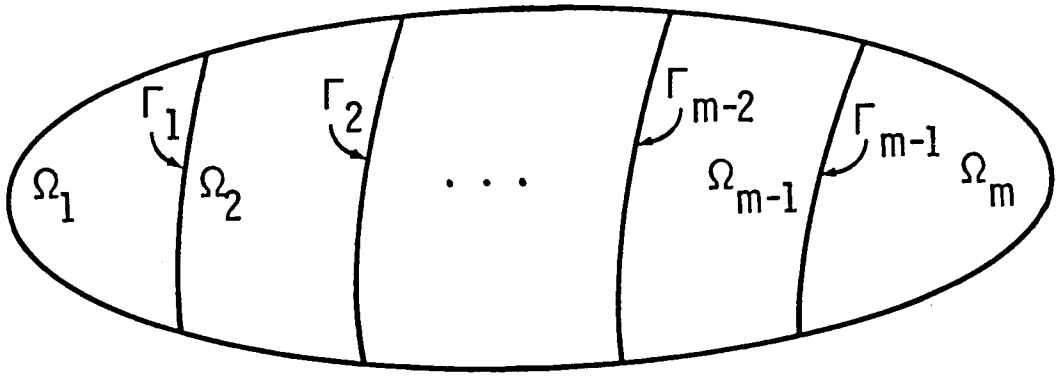

Figure 3. An example subdivision of the region Ω.

It is natural to order the interface unknowns U_0 one interface at a time, e.g., first those on Γ_1, then those on Γ_2, etc. It is not hard to see that the matrix D for this example is block tridiagonal, i.e., the unknowns corresponding to the interface Γ_i are connected only to the unknowns on the interfaces Γ_{i-1}, Γ_i, and Γ_{i+1}. By taking advantage of features such as this, the cost of step 5 and 6 of the algorithm can be greatly reduced, especially in three-dimensional settings. We note that these ideas are similar to those connected with one-way direction algorithms for positive definite problems [9].

REFERENCES

[1] F. Gantmacher, The Theory of Matrices, Chelsea, New York, 1960.

[2] G. Golub and D. Mayers, "Use of preconditioning over irregular regions," in Computer Methods in Applied Science and Engineering, VI, (R. Glowinski and J. Lions, Eds.), 1983, pp. 3-14.

[3] M. Gunzburger and R. Nicolaides, "Elimination with noninvertible pivots," Linear Algebra Appl., 64, 1985, pp. 183-189.

[4] V. Girault and P.-A. Raviart, Finite Element Approximation of the Navier-Stokes Equations, Springer, Berlin, 1979.

[5] J. Boland and R. Nicolaides, "Stability of finite elements under divergence constraints," SIAM J. Numer. Anal., 20, 1983, pp. 722-731.

[6] J. Boland and R. Nicolaides, "Stable and semistable low order finite elements for viscous flows," SIAM J. Numer. Anal., 22, 1985, pp. 474-492.

[7] G. Fix, M. Gunzburger, and R. Nicolaides, "On mixed finite element methods for first-order elliptic systems," Numer. Math., 37, 1981, pp. 29-48.

[8] L. Adams and R. Voigt, "A methodology for exploiting parallelism in the finite element process," in Proceedings of the NATO Workshop on High Speed Computations, (J. Kowolik, Ed.), Springer-Verlag, Berlin, 1984, pp. 373-392.

[9] A. George and J. Liu, Computer Solution of Large Sparse Positive Definite Systems, Prentice Hall, Englewood Cliffs, New Jersey, 1981.

Standard Bibliographic Page

1. Report No. NASA CR-178122 ICASE Report No. 86-30	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle ON STRUCTURING ALGORITHMS AND SOLUTION TECHNIQUES FOR THE NUMERICAL APPROXIMATION OF PARTIAL DIFFERENTIAL EQUATIONS		5. Report Date May 1986 6. Performing Organization Code
7. Author(s) M. D. Gunzburger, R. A. Nicolaides		8. Performing Organization Report No. 86-30
9. Performing Organization Name and Address Institute for Computer Applications in Science and Engineering Mail Stop 132C, NASA Langley Research Center Hampton, VA 23665-5225		10. Work Unit No. 11. Contract or Grant No. NAS1-18107
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		13. Type of Report and Period Covered Contractor Report 14. Sponsoring Agency Code 505-31-83-01

15. Supplementary Notes

Langley Technical Monitor:
J. C. South

Submitted to Applied Numerical Mathematics

Final Report

16. Abstract

Substructuring methods are in common use in mechanics problems where typically the associated linear systems of algebraic equations are positive definite. Here these methods ae extended to problems which lead to nonpositive definite, nonsymmetric matrices. The extension is based on an algorithm which carries out the block Gauss elimination procedure without the need for interchanges even when a pivot matrix is singular. Examples are provided wherein the method is used in connection with finite element solutions of the stationary Stokes equations and the Helmholtz equation, and dual methods for second-order elliptic equations.

17. Key Words (Suggested by Authors(s))	18. Distribution Statement
substructuring, parallel computing	64 - Numerical Analysis Unclassified - unlimited

19. Security Classif.(of this report)	20. Security Classif.(of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	32	A03

For sale by the National Technical Information Service, Springfield, Virginia 22161

End of Document