Lawrence Berkeley National Laboratory
Recent Work

Title
Finite element capacitance matrix methods

Permalink
https://escholarship.org/uc/item/5xn9q3rf

Author
Widlund, Olof B.

Publication Date
1986-12-01

eScholarship.org Powered by the California Digital Library
University of California

LBL-22583

FINITE ELEMENT CAPACITANCE MATRIX METHODS[1]

Christoph Börgers[2]

Lawrence Berkeley Laboratory
and
Department of Mathematics
University of California
Berkeley, California 94720

and

Olof B. Widlund[3]

Department of Computer Science
Courant Institute of Mathematical Sciences
251 Mercer Street
New York, New York 10012

December 1986

[1]Also issued as Technical Report 261 at the Computer Science Department of New York University.

[2]Supported in part by the Applied Mathematical Sciences Subprogram of the Office of Energy Research, U.S. Department of Energy under contract DE-AC03-76SF00098.

[3]Supported in part by the National Science Foundation under grant NSF-DCR-8405506 and by the U.S. Department of Energy under contract DE-AC02-76ER03077-V at the Courant Mathematics and Computing Laboratory.

Finite Element Capacitance Matrix Methods

1. Introduction

In this paper, we will discuss finite element imbedding methods, in particular capacitance matrix methods, for the solution of Neumann and Dirichlet problems on general bounded domains in the plane. We will give a survey of such methods, discuss details of implementation, and present numerical results. We have implemented these methods for the Helmholtz equation. We note that they are also applicable to certain other elliptic equations without any significant change.

In section 3, we describe a new triangulation algorithm particularly useful in connection with finite element imbedding methods, where all the triangles away from the boundary should have their vertices on a regular mesh. For the performance of finite element imbedding methods, it is important to avoid small triangles; see Proskurowski and Widlund (1980). We give a quantitative definition of non-degeneracy of a triangulation, motivated by the convergence theory of imbedding methods, and prove a bound on the degeneracy of the triangulations generated by our algorithm which is uniform in h and in the region Ω.

In section 4, we describe imbedding methods from a linear algebra point of view. These considerations also apply to domain decomposition methods; see, e.g., Bjørstad and Widlund (1984), Widlund (1986).

For Neumann problems, an efficient finite element imbedding method was introduced by Proskurowski and Widlund (1980). Related finite difference methods had previously been studied by Astrakhantsev (1978), Kuznetsov and Matsokin (1974), O'Leary and Widlund (1979), Proskurowski (1979), Proskurowski and Widlund (1976) and Shieh (1978). A similar finite element method was also proposed by Korneev (1977). In section 5, we describe an improved implementation of the method proposed by Proskurowski and Widlund (1980) and study several variations, including versions using quadratic isoparametric finite elements.

The construction of good imbedding techniques is significantly harder for Dirichlet problems; see sections 6 and 7. One known method makes use of the connection between interior Dirichlet problems and exterior Neumann problems; see Dryja (1983), Widlund (1984). A second method is motivated by the Fredholm integral equation of the second kind used to establish existence for Dirichlet problems; see, e.g., Garabedian (1964). In this construction, the solution is obtained as the potential generated by a dipole layer located on the boundary of the domain. The solution of discrete Dirichlet problems can similarly be obtained as a discrete dipole layer potential. Imbedding methods of this kind have been studied in a finite difference context by Astrakhantsev (1977), O'Leary and Widlund (1979), Proskurowski and Widlund (1976), Shieh (1979). A finite element dipole method is described in section 6. A third possibility is to use a single layer Ansatz and precondition the capacitance matrix with an appropriate operator, e.g., with the square root of a discrete Helmholtz operator on the boundary. We have not carried out numerical experiments with this method. See, e.g., Bjørstad and Widlund (1984) for a closely related domain decomposition algorithm.

2. Notation and the form of the finite element systems

Let Ω be a domain in R^2 such that $\overline{\Omega}$ is contained in $(0;1)^2$. We will consider the Neumann problem

$$\begin{aligned} -\Delta u + cu &= f \quad \text{on } \Omega \\ \frac{\partial u}{\partial n} &= g \quad \text{on } \partial\Omega, \end{aligned} \tag{2.1}$$

and the Dirichlet problem

$$\begin{aligned} -\Delta u + cu &= f \quad \text{on } \Omega \\ u &= g \quad \text{on } \partial\Omega. \end{aligned} \tag{2.2}$$

$\frac{\partial}{\partial n}$ denotes the exterior normal derivative, and c is a real constant. We restrict c to values such that the problem is, at least, positive semi-definite.

We use finite element discretizations based on triangles $\tau_1, \cdots, \tau_k$ and $\tau_{k+1}, \cdots, \tau_{2N^2}$ such that $(\tau_\nu)_{1 \le \nu \le 2N^2}$ is a triangulation of $(0;1)^2$ and

$$\Omega^k := \bigcup_{\nu=1}^{k} \tau_\nu \tag{2.3}$$

approximates Ω; see section 3 for our triangulation algorithm. We use linear elements, i.e. piecewise linear Lagrangian finite elements, and quadratic isoparametric triangles of type (2); see Ciarlet (1978), p. 228. In the isoparametric quadratic case, the edges of the triangles which intersect the boundary can be parabolic curves.

The degrees of freedom are the values of the finite element functions at the vertices of the triangles, and, in the case of quadratic elements, in addition the values in the midpoints of the sides of those triangles. We shall make use of an auxiliary boundary value problem on the entire square $(0;1)^2$, with boundary conditions on $\partial(0;1)^2$ specified later.

The finite element discretization of the problems on $(0;1)^2$ results in a system of linear equations

$$K(c)\underline{x} = \underline{r}, \tag{2.4}$$

with

$$K(c) = K + cM, \tag{2.5}$$

where K is the stiffness matrix and M is the mass matrix. The entries in K and M are of the form

$$\int_{[0;1]^2} \nabla\phi^T \cdot \nabla\psi \; d\underline{x},$$

$$\int_{[0;1]^2} \phi\psi \; d\underline{x},$$

respectively, where ϕ,ψ are canonical basis functions of the finite element space.

We order the unknowns such that K and M take the following form.

$$K = \begin{pmatrix} K_{11} & 0 & K_{13} \\ 0 & K_{22} & K_{23} \\ K_{13}^T & K_{23}^T & K_{33} \end{pmatrix}, \tag{2.6}$$

$$M = \begin{pmatrix} M_{11} & 0 & M_{13} \\ 0 & M_{22} & M_{23} \\ M_{13}^T & M_{23}^T & M_{33} \end{pmatrix}, \tag{2.7}$$

where the subscripts 1,2,3 correspond to nodes in the interior of Ω^h, the exterior and on the boundary, respectively. We split the matrices K_{33} and M_{33} as follows.

$$K_{33} = K_{33}^{(1)} + K_{33}^{(2)}, \tag{2.8}$$

$$M_{33} = M_{33}^{(1)} + M_{33}^{(2)}. \tag{2.9}$$

Here $K_{33}^{(1)}$ and $M_{33}^{(1)}$ are constructed from the contributions of Ω^h to the integrals defining the elements of K_{33}, M_{33}, and $K_{33}^{(2)} = K_{33} - K_{33}^{(1)}$ and $M_{33}^{(2)} = M_{33} - M_{33}^{(1)}$ are the corresponding contributions from the exterior.

We shall use the notation

$$G(c) = \begin{pmatrix} G_{11}(c) & G_{12}(c) & G_{13}(c) \\ G_{12}(c)^T & G_{22}(c) & G_{23}(c) \\ G_{13}(c)^T & G_{23}(c)^T & G_{33}(c) \end{pmatrix} := K(c)^{-1} \tag{2.10}$$

and

$$G := G(0), \quad G_{ij} := G_{ij}(0). \tag{2.11}$$

Vectors such as $\underline{x}$ and $\underline{r}$ in (2.4) will often be called mesh functions in the following sections. Similarly, matrices such as $K(c)$ will be called operators.

3. A triangulation algorithm

Let Ω be an open set in R^2 whose closure $\overline{\Omega}$ is contained in $(0;1)^2$. Let P be a finite set of points in $(0;1)^2$. In our numerical tests, Ω has been a curvilinear polygon, and P the set of corners of $\partial\Omega$. Let $N \geq 1$ be an integer and $h = \frac{1}{N}$. Define

$$\hat{\Gamma}^h := \{(i_1 h, i_2 h): 0 \leq i_1, i_2 \leq N\}. \tag{3.1}$$

Assumptions on h: h is assumed to be so small that the following two conditions are satisfied.

(i) If $(i_1,i_2) \in \{0;\ldots;N\}^2$, $\underline{x} \in P$, $\underline{y} \in P$, $\underline{x} \neq \underline{y}$, and if

$$\underline{x} \in [(i_1-\tfrac{1}{2})h\,;(i_1+\tfrac{1}{2})h] \times [(i_2-\tfrac{1}{2})h\,;(i_2+\tfrac{1}{2})h], \text{ then} \tag{3.2}$$

$$\underline{y} \notin [(i_1-\tfrac{3}{2})h\,;(i_1+\tfrac{3}{2})h] \times [(i_2-\tfrac{3}{2})h\,;(i_2+\tfrac{3}{2})h]. \tag{3.3}$$

(ii) $\overline{\Omega}$ is contained in $[2h\,;1-2h]^2$.

We shall construct triangles $\tau_1, \cdots, \tau_k$ and $\tau_{k+1}, \cdots, \tau_{2N^2}$ such that $(\tau_\nu)_{1 \leq \nu \leq 2N^2}$ is a triangulation of $(0;1)^2$ and

$$\Omega^h := \bigcup_{\nu=1}^{k} \tau_\nu \tag{3.4}$$

approximates Ω. Fig. 1 shows the triangulation of a region Ω^h, and Fig. 2 the corresponding triangulation of $(0;1)^2$. The algorithm will attempt to construct the triangulation in such a way that all points in P become vertices of Ω^h. P can be prescribed in an arbitrary way; in particular, some of its points can be far from Ω. Such points are detected and removed from P by our code.

We first define a function

$$\hat{\phi}: \hat{\Gamma}^h \to \{-1;0;1\} \tag{3.5}$$

by the following algorithm.

Algorithm 3.1:

For $\hat{\underline{x}} \in \hat{\Gamma}^h$, define initial values $\hat{\phi}(\hat{\underline{x}})$ as follows.

$\hat{\phi}(\hat{\underline{x}})=0$ if there is a point in P contained in the square with side length h centered at $\hat{\underline{x}}$. Otherwise, $\hat{\phi}(\hat{\underline{x}})=1$ if $\hat{\underline{x}} \in \Omega$, $\hat{\phi}(\hat{\underline{x}})=-1$ if $\hat{\underline{x}} \notin \Omega$.

Modify the values $\hat{\phi}(\hat{\underline{x}})$ in the following way.

For $i_1=0,\ldots,N-1$:

For $i_2=0,\ldots,N-1$:

If $\hat{\phi}(i_1h, i_2h)\cdot\hat{\phi}((i_1+1)h, i_2h)=-1$: Determine $x_1 \in [i_1h; (i_1+1)h]$ with $(x_1, i_2h) \in \partial\Omega$; if $x_1 \geq (i_1+\frac{1}{2})h$, set $\hat{\phi}((i_1+1)h, i_2h)=0$, otherwise set $\hat{\phi}(i_1h, i_2h)=0$.

If $\hat{\phi}(i_1h, i_2h)\cdot\hat{\phi}(i_1h, (i_2+1)h)=-1$: Determine $x_2 \in [i_2h; (i_2+1)h]$ with $(i_1h, x_2) \in \partial\Omega$; if $x_2 \geq (i_2+\frac{1}{2})h$, set $\hat{\phi}(i_1h, (i_2+1)h)=0$, otherwise set $\hat{\phi}(i_1h, i_2h)=0$.

Notice that the triangulation depends on the order in which we inspect the points (i_1h, i_2h).

We define

$$\hat{I}^h := \{\hat{\underline{x}} \in \hat{\Gamma}^h : \hat{\phi}(\hat{\underline{x}})=1\}, \tag{3.6}$$

$$\hat{B}^h := \{\hat{\underline{x}} \in \hat{\Gamma}^h : \hat{\phi}(\hat{\underline{x}})=0\}, \tag{3.7}$$

$$\hat{E}^h := \{\hat{\underline{x}} \in \hat{\Gamma}^h : \hat{\phi}(\hat{\underline{x}})=-1\}. \tag{3.8}$$

Points in $\hat{I}^h$, $\hat{B}^h$ and $\hat{E}^h$ are called interior, boundary and exterior points respectively. Each point $\hat{\underline{x}}=(\hat{x}_1, \hat{x}_2) \in \hat{\Gamma}^h$ is associated in an obvious way with a point

$$\underline{x} \in [\hat{x}_1-\frac{h}{2}; \hat{x}_1+\frac{h}{2}] \times [\hat{x}_2-\frac{h}{2}; \hat{x}_2+\frac{h}{2}] \tag{3.9}$$

belonging to

$$\Gamma^h = \hat{I}^h \cup B^h \cup \hat{E}^h. \tag{3.10}$$

The set B^h is the image of $\hat{B}^h$ and contains P as well as the points constructed above by intersecting the boundary $\partial\Omega$ with certain mesh lines.

By construction, the set of boundary points $\hat{B}^h$ has no holes in the following sense.

Proposition 3.1: If $\hat{\underline{x}}=(\hat{x}_1,\hat{x}_2)$, $\hat{\underline{y}}=(\hat{y}_1,\hat{y}_2)$ are points in $\hat{\Gamma}^h$, and if $|\hat{x}_1-\hat{y}_1| + |\hat{x}_2-\hat{y}_2| = h$, then $\hat{\phi}(\hat{\underline{x}})\cdot\hat{\phi}(\hat{\underline{y}}) \neq -1$.

If one regards $\hat{\Gamma}^h$ as an undirected graph, proposition 3.1 can be expressed as follows. Each connected component of $\hat{\Gamma}^h - \hat{B}^h$ is entirely contained in $\hat{I}^h$ or entirely contained in $\hat{E}^h$.

We are now prepared to define the triangles τ_ν. Let $(i_1,i_2)\in\{0;\cdots;N-1\}^2$, $\hat{\underline{x}}^{(1)} := (i_1h, i_2h)$, $\hat{\underline{x}}^{(2)} := ((i_1+1)h, i_2h)$, $\hat{\underline{x}}^{(3)} := ((i_1+1)h, (i_2+1)h)$, $\hat{\underline{x}}^{(4)} := (i_1h, (i_2+1)h)$, and let Q be the quadrilateral formed by $\underline{x}^{(1)}$, $\underline{x}^{(2)}$, $\underline{x}^{(3)}$, $\underline{x}^{(4)}$, the points in Γ^h associated with the $\hat{\underline{x}}^{(i)}$. Q is cut into two triangles along one of its diagonals, and a decision is made for each of them if it belongs to Ω^h or to $(0;1)^2-\overline{\Omega}^h$.

To decide how to cut Q, we need a way of measuring the degeneracy of the resulting triangles. We assign a number μ to a pair (Q,d), where d is a diagonal of Q, as follows. Cutting Q along d gives two triangles $\tau^{(1)},\tau^{(2)}$. Let ψ_τ be an affine mapping from the reference triangle

$$\{\underline{x}=(x_1,x_2) : 0\leq x_1\leq 1,\ 0\leq x_2\leq 1-x_1\} \tag{3.11}$$

onto $\tau=\tau^{(1)}$ or $\tau^{(2)}$. There are several such affine mappings. To specify our choice, let $\hat{\psi}_\tau(0,0)$, $\hat{\psi}_\tau(1,0)$, $\hat{\psi}_\tau(0,1)$ denote the points in $\hat{\Gamma}^h$ corresponding to the points $\psi_\tau(0,0),\psi_\tau(1,0),\psi_\tau(0,1)$ in Γ^h, and require that $\hat{\psi}_\tau(1,0)-\hat{\psi}_\tau(0,0)$, $\hat{\psi}_\tau(0,1)-\hat{\psi}_\tau(0,0)$ be a positively oriented pair of orthogonal vectors.

We define then

$$\mu := \min_{\tau=\tau^{(1)},\tau^{(2)}} \frac{\det D\,\psi_\tau}{\lambda_{\max}((D\,\psi_\tau)^T\,(D\,\psi_\tau))}. \tag{3.12}$$

Here $\lambda_{\max}(\cdots)$ denotes the larger of the two eigenvalues. The two values of μ, corresponding to the two diagonals of Q, are used below to decide how Q is to be divided. We note that one of these values will always be positive. The larger μ, the less degenerate is the resulting configuration. Proskurowski and Widlund (1980) use the simpler degeneracy measure

$$\frac{1}{\text{length of } d}. \tag{3.13}$$

In most cases, this criterion leads to triangulations which are very similar to those obtained with (3.12). However, consider the case

$$\begin{aligned}
\underline{x}_1 &= ((i_1+\tfrac{1}{2})h\ ,i_2h\ ) \\
\underline{x}_2 &= ((i_1+1)h\ ,(i_2+\tfrac{1}{2})h\ ) \\
\underline{x}_3 &= ((i_1+\tfrac{3}{2})h\ ,(i_2+1)h\ ) \\
\underline{x}_4 &= (i_1h\ ,(i_2+\tfrac{3}{2})h\ ).
\end{aligned}$$

Q has a 180°-angle. The criterion (3.13) does not detect the difference between the two possible ways of cutting Q, and this can result in a triangle of zero area.

Formula (3.12) can be motivated as follows. Let K be the finite element stiffness matrix for the Dirichlet problem

$$\begin{aligned}
-\Delta\phi &= f \quad \text{on } (0;1)^2 \\
\phi &= 0 \quad \text{on } \partial(0;1)^2
\end{aligned} \tag{3.14}$$

obtained with piecewise linear finite elements, using the triangulation of $(0;1)^2$ generated by our algorithm. Let $\hat{K}$ be the analogous matrix obtained using a regular triangulation obtained by cutting the squares

$$[i_1h\ ;(i_1+1)h\ ]\times[i_2h\ ;(i_2+1)h\ ] \tag{3.15}$$

into pairs of triangles. As is well-known, $\hat{K}$ is the usual five point Laplacian. The following estimate holds,

$$\text{cond}[\hat{K}^{-1/2}K\hat{K}^{-1/2}]\ \leq\ \max_{\tau=\tau_1,\cdots,\tau_{2N^2}}\ \text{cond}[(D\,\psi_\tau)^T\,(D\,\psi_\tau)]. \tag{3.16}$$

This condition number occurs naturally in bounds of the rate of convergence of the conjugate gradient methods discussed in later sections.

If $\mu>0$, then

$$\mu = \min_{\tau=\tau^{(1)},\tau^{(2)}} \frac{1}{\sqrt{\text{cond}\ (D\,\psi_\tau)^T\,(D\,\psi_\tau)}}. \tag{3.17}$$

Proof of (3.16): Let

$$(\tau_\nu)_{1\le\nu\le 2N^2} \tag{3.18}$$

be the triangulation of $(0;1)^2$ obtained by our algorithm, and let

$$(\hat{\tau}_\nu)_{1\le\nu\le 2N^2} \tag{3.19}$$

denote the regular triangulation in the logical plane defined by the mapping relating $\hat{\Gamma}^h$ to Γ^h. Let ϕ be a linear function on $\tau=\tau_\nu$, and let $\hat{\phi}$ be a linear function on $\hat{\tau}=\hat{\tau}_\nu$ with $\hat{\phi}(\hat{\underline{x}}^{(i)})=\phi(\underline{x}^{(i)})$, $i=1,2,3$. The desired estimate,

$$\frac{1}{\sqrt{\text{cond}\,(D\,\psi_\tau)^T\,(D\,\psi_\tau)}} \le \frac{\int_\tau |\,\nabla\phi\,|^2 d\underline{x}}{\int_{\hat{\tau}} |\,\nabla\hat{\phi}\,|^2 d\hat{\underline{x}}} \le \sqrt{\text{cond}\,(D\,\psi_\tau)^T\,(D\,\psi_\tau)}, \tag{3.20}$$

is obtained immediately by a change of variables. □

μ may be ≤ 0. This occurs exactly when d does not lie in the interior of Q. By making μ as large as possible, d will lie within Q.

If Q is a square, then $\mu=1$. Somewhat arbitrarily, we say that configuration is non-degenerate if $\mu\ge\frac{1}{4}$; see cases d and e below.

Let n_I be the number of points in $\hat{I}^h\cap\{\hat{\underline{x}}^{(1)};\hat{\underline{x}}^{(2)};\hat{\underline{x}}^{(3)};\hat{\underline{x}}^{(4)}\}$, and let n_B and n_E be defined analogously.

Case a: $n_E\ge 2$. Then $n_I=0$. If $n_E=4$, Q is a square and is cut along the diagonal joining the left upper and right lower vertex. In all other cases, we maximize μ. Both triangles belong to $(0;1)^2-\overline{\Omega}^h$.

Case b: $n_I\ge 2$. This case is analogous to case a.

Case c: $n_E=1, n_I=1$. Q is convex in this case. To see this, assume that the vertices of Q satisfy

$$\underline{x}^{(1)}=(i_1h\,,i_2h\,),$$
$$\underline{x}^{(2)}\in[(i_1+\tfrac{1}{2})h\,;(i_1+\tfrac{3}{2})h\,]\times[(i_2-\tfrac{1}{2})h\,;(i_2+\tfrac{1}{2})h\,],\ \ \underline{x}^{(2)}\in B^h\,,$$
$$\underline{x}^{(3)}=((i_1+1)h\,(i_2+1)h\,),$$

$$\underline{x}^{(4)} \in [(i_1 - \frac{1}{2})h\,;(i_1 + \frac{1}{2})h\,] \times [(i_2 + \frac{1}{2})h\,;(i_2 + \frac{3}{2})h\,],\quad \underline{x}^{(4)} \in B^h\,.$$

By proposition 3.1, these assumptions may be made without loss of generality. The straight line joining $\underline{x}^{(1)}$ and $\underline{x}^{(3)}$ divides the plane into two half planes, one containing $\underline{x}^{(2)}$ and one containing $\underline{x}^{(4)}$, and Q is convex.

We can therefore cut along the diagonal joining the two vertices of Q belonging to B^h. The triangle whose third vertex lies in $\hat{I}^h$ belongs to Ω^h, the other one to $(0;1)^2 - \overline{\Omega}^h$. A detailed argument is required to establish that the resulting triangles are not degenerate; see proposition 3.3 below.

Case d: $n_E = 1$, $n_I = 0$, $n_B = 3$. We attempt to cut such that all three vertices of one of the two resulting triangles lie in B^h. If the resulting configuration is non-degenerate, and if the centroid of the triangle whose three vertices all lie in B^h lies in Ω, we include this triangle in Ω^h and the remaining one in $(0;1)^2 - \overline{\Omega}^h$. In all other cases, we proceed as in case a.

Case e: $n_I = 1$, $n_E = 0$, $n_B = 3$. This case is analogous to case d.

Case f: $n_E = n_I = 0$, $n_B = 4$. We cut Q in the way which leads to the least degenerate configuration. Each resulting triangle is included in Ω^h if and only if its centroid belongs to Ω.

For later reference, we introduce the following terminology. We call the quadrilateral Q *regular* if $n_I = 4$ or $n_E = 4$, otherwise Q is *irregular*. A triangle τ_ν is *regular* if it is obtained by cutting a regular cell, otherwise it is *irregular*. An irregular triangle is *interior* if it belongs to Ω^h, otherwise it is *exterior*.

For the implementation of a finite element discretization of a boundary value problem on Ω, it is useful to know the set $\tilde{B}^h$ of all nodes on $\partial\Omega^h$. A node belongs to $\tilde{B}^h$ if it belongs to an interior irregular triangle as well as to an exterior irregular triangle.

Proposition 3.2: $\tilde{B}^h$ is contained in B^h, but may be smaller than B^h.

Proof: If $\underline{x} \in \tilde{B}^h$, then $\underline{x}$ belongs to some interior triangle. Therefore $\underline{x} \notin \hat{E}^h$. Similarly, one concludes that $\underline{x} \notin \hat{I}^h$. Therefore $\underline{x} \in B^h$.

If P contains a point with a positive distance to the region Ω, then this point will, for sufficiently small h, belong to B^h but not to $\tilde{B}^h$. Thus the two sets need not be equal. Points in B^h which do not belong to $\tilde{B}^h$ may also occur if there are several very close intersections of a mesh line with $\partial\Omega$. □

A node is *interior irregular* if it lies in the interior of Ω^h and the stiffness matrix K couples it to a point in $\tilde{B}^h$. Similarly, a node is *exterior irregular* if it lies in the interior of the complement of Ω^h and if the stiffness matrix K couples it to a point in $\tilde{B}^h$.

By examining the six cases, the following proposition is seen to hold.

Proposition 3.3: There is a number γ independent of Ω and h such that

$$\text{cond}((D\,\psi_\tau)^T\,(D\,\psi_\tau)) \leq \gamma \tag{3.21}$$

for all triangles τ in the triangulation generated by our algorithm.

A calculation shows that

$$\gamma = (3+\sqrt{8})^2 \approx 34 \quad \text{if } P \text{ is empty,} \tag{3.22}$$

$$\gamma = (15+\sqrt{221})^2/4 \approx 223 \quad \text{if } P \text{ is not empty.} \tag{3.23}$$

The condition number of $\hat{K}^{-1/2}K\hat{K}^{-1/2}$ is therefore bounded independently of Ω and h. From the theory of capacitance matrix methods discussed below, it then follows that the rate of convergence depends on Ω alone. (3.22), (3.33) are the best possible estimates for γ, but it is important to note that (3.16) is often far from sharp.

We have made no specific assumptions on how the region is represented in the computer. We only assume that a subroutine is available to decide if a given point $\underline{x} \in (0;1)^2$ lies in Ω or not. If this decision can be made in $O(1)$ operations, the total number of operations required by our algorithm is $O(N^2+n_B \log\frac{1}{\epsilon})$, where $\epsilon>0$ is the error tolerance for x_1 and x_2 in algorithm 3.1. We assume here that the bisection method is used in algorithm 3.1 to determine x_1 and x_2. We note that in principle it seems possible to design an $O(n_B)$-algorithm if the boundary is given in parametric form. However, our approach may be preferable if a parametrization of the boundary cannot easily be obtained.

Ω^h is a poor approximation of Ω in some cases. This is unavoidable, since we want Ω^h to be a union of non-degenerate triangles whose areas are approximately $\frac{1}{2}h^2$. In particular, difficulties can occur near corners and if $\partial\Omega$ has two intersections with a regular grid line which are closer than h to each other. In such a case, it is possible that neither of the two intersections is detected. The last observation leads to the following improvement of algorithm 3.1.

Algorithm 3.1 (improved version): Replace the conditions

(i) $\hat{\phi}(i_1h, i_2h)\cdot\hat{\phi}((i_1+1)h, i_2h) = -1$,

(ii) $\hat{\phi}(i_1h, i_2h)\cdot\hat{\phi}(i_1h, (i_2+1)h) = -1$ by

(i)' $\hat{\phi}(i_1h, i_2h)\cdot\hat{\phi}((i_1+1)h, i_2h) \neq 0$ and $[(i_1h; (i_1+1)h] \times \{i_2h\} \cap \partial\Omega$ is not empty,

(ii)' $\hat{\phi}(i_1h, i_2h)\cdot\hat{\phi}(i_1h, (i_2+1)h) \neq 0$ and $\{ih\} \times [i_1h; (i_2+1)h] \cap \partial\Omega$ is not empty.

In our code, we search for intersections of $\partial\Omega$ with $[(i_1h; (i_1+1)h] \times \{i_2h\}$ by testing whether the points $(i_1 + \frac{\nu}{5})h, i_2h)$, $\nu = 1, \cdots, 5$, lie in Ω. In a similar way, we search for intersections of $\partial\Omega$ with $\{i_1h\} \times [i_2h; (i_2+1)h]$. Notice, however, that this version can be much more expensive than the original algorithm.

In spite of this improvement, corners are frequently cut off by our algorithm. It is possible that the corner points in P could be characterized differently and that further improvements of the algorithm are possible. In the case of a relatively pointed corner, it might be useful to provide not just the location of the corner, but also a half-ray, the initial section of which lies in Ω. With this additional information, it should be possible to avoid assigning triangles close to the corner to the wrong set.

No region Ω is rejected unless h is too large, and in our experience, Ω^h is a good approximation of Ω whenever $\partial\Omega$ is smooth. Our algorithm is therefore an improvement over the algorithm in Proskurowski and Widlund (1980), which may break down if Ω is not convex.

4. Algebraic description of imbedding methods

In this section, we describe imbedding algorithms from the point of view of linear algebra. We assume that we wish to solve a given system of linear equations, and that we know an easy way of satisfying most, but not all of the equations. In our applications, these are the equations corresponding to mesh points away from $\partial\Omega$.

We consider a linear system of equations

$$A\underline{x} = \underline{b}, \text{ with } A \in R^{n\times n}, \quad \underline{x}, \underline{b} \in R^n. \tag{4.1}$$

Let $\tilde{A} \in R^{n\times n}$ be a non-singular matrix such that $\tilde{A}^{-1}$ is an approximate inverse of A in the following sense.

$$A\tilde{A}^{-1} = \begin{pmatrix} I & 0 \\ Q & C \end{pmatrix}, \quad I \in R^{p\times p}, \quad C \in R^{q\times q}, \quad p+q=n. \tag{4.2}$$

Thus an application of $\tilde{A}^{-1}$ to $\underline{b}$ satisfies the first p equations in (4.1), but not necessarily the remaining ones.

The matrix C is called the capacitance matrix. If C is known, $A\underline{x}=\underline{b}$ can be solved as follows.

Algorithm 4.1a: Solve $C\underline{y}_2 = \underline{b}_2 - Q\underline{b}_1$. Set $\underline{y} := \begin{pmatrix} \underline{b}_1 \\ \underline{y}_2 \end{pmatrix}$. Then $\underline{x} := \tilde{A}^{-1}\underline{y}$ solves $A\underline{x}=\underline{b}$.

Note that $Q\underline{b}_1$ can be computed by applying $A\tilde{A}^{-1}$ to $\begin{pmatrix} \underline{b}_1 \\ \underline{0} \end{pmatrix}$.

This algorithm is called the *direct* imbedding method. It is sometimes presented in the following way:

Algorithm 4.1b: Compute $\underline{b} - A\tilde{A}^{-1}\underline{b} = \begin{pmatrix} \underline{0} \\ \underline{b}_2 - Q\underline{b}_1 - C\underline{b}_2 \end{pmatrix}$. Solve $C\underline{w} = \underline{b}_2 - Q\underline{b}_1 - C\underline{b}_2$.

Compute $\underline{x} = \tilde{A}^{-1}\begin{pmatrix} \underline{b}_1 \\ \underline{b}_2 + \underline{w} \end{pmatrix}$. $\underline{x}$ solves $A\underline{x}=\underline{b}$.

The fact that C may be singular poses no problems here. If $A\underline{x}=\underline{b}$ has a solution, then the systems involving C in algorithms 4.1a, 4.1b have solutions. We remark that algo-

rithm 4.1a is slightly more efficient than algorithm 4.1b.

The direct imbedding method can be an efficient technique if $q \ll p$ and if a sequence of problems $A\underline{x} = \underline{b}$ with different right-hand sides $\underline{b}$ is to be solved.

Next we describe *iterative* imbedding methods. Here the system involving the matrix C is solved iteratively. For this purpose, we use the conjugate gradient algorithm, written in the following form.

Consider a system of linear equations of the form

$$M\underline{u} = \underline{b}, \tag{4.3}$$

where M is a symmetric, positive semi-definite $n \times n$-matrix. We assume that $\underline{b}$ lies in the range of M. Let N be a symmetric, positive definite $n \times n$-matrix, the preconditioner. In the special case $N = I$, one obtains the conjugate gradient algorithm without preconditioning.

Algorithm 4.2a (conjugate gradient algorithm, first form):

Choose $\underline{z}^{(0)} \in R^n$.

$\underline{g}^{(0)} := \underline{b} - MN^{-1}\underline{z}^{(0)}$

Replace $\underline{g}^{(0)}$ by its orthogonal projection onto the range of M.

$\underline{d}^{(0)} := \underline{g}^{(0)}$

$\tilde{\underline{g}}^{(0)} := N^{-1}\underline{g}^{(0)}$

$\tilde{\underline{d}}^{(0)} := \tilde{\underline{g}}^{(0)}$

For $j = 0,1,2,\ldots$:

$$\alpha^{(j)} := \frac{\underline{g}^{(j)T}\tilde{\underline{g}}^{(j)}}{\tilde{\underline{d}}^{(j)T} M \tilde{\underline{d}}^{(j)}}$$

$\underline{z}^{(j+1)} := \underline{z}^{(j)} + \alpha^{(j)}\underline{d}^{(j)}$

$\underline{g}^{(j+1)} := \underline{g}^{(j)} - \alpha^{(j)} M \tilde{\underline{d}}^{(j)}$

Replace $\underline{g}^{(j+1)}$ by its orthogonal projection onto the range of M.

$$\tilde{\underline{g}}^{(j+1)} := N^{-1}\underline{g}^{(j+1)}$$

$$\beta^{(j)} := \frac{\tilde{\underline{g}}^{(j+1)T}\underline{g}^{(j+1)}}{\tilde{\underline{g}}^{(j)T}\underline{g}^{(j)}}$$

$$\underline{d}^{(j+1)} := \underline{g}^{(j+1)} + \beta^{(j)}\underline{d}^{(j)}$$

$$\tilde{\underline{d}}^{(j+1)} := \tilde{\underline{g}}^{(j+1)} + \beta^{(j)}\tilde{\underline{d}}^{(j)}.$$

The sequence $\underline{u}^{(j)} = N^{-1}\underline{z}^{(j)}$ converges to a solution of $M\underline{u} = \underline{b}$.

The projections onto the range of M are without any effect in exact arithmetic. In floating point arithmetic, however, the algorithm may diverge if the kernel of M is non-trivial and if one omits the projections.

It was pointed out by Proskurowski and Widlund (1980) that algorithm 4.2a is a particularly efficient way of writing the conjugate gradient method in the context of iterative imbedding methods. If N^{-1} is an approximate inverse of M in the sense specified above, then multiplications by $M-N$ are very cheap, since the first p rows of $M-N$ are zero. This is a useful fact because of

Remark 4.1: In each iteration, algorithm 4.2a requires one multiplication of a vector by M, and one multiplication of a vector by N^{-1}. Alternatively, the algorithm can be carried out such that it requires one multiplication of a vector by $M-N$ and one multiplication of a vector by N^{-1} in each iteration.

If $\underline{z}^{(0)}$ coincides with $\underline{b}$ in the first p components, then the corresponding components of $\underline{g}^{(j)}$ and $\underline{d}^{(j)}$ are zero for all j. In algorithm 4.2a, this can be exploited in several ways. It is especially important that only the last q components of the vectors $\tilde{g}^{(j)} = N^{-1}g^{(j)}$ need to be computed. When N^{-1} is a fast Poisson solver, the method developed by Banegas (1978) can be used; see also Proskurowski (1979).

Notice that the computation of $\underline{u}^{(j)}$, starting from a nonzero $\underline{z}^{(0)}$, requires $j+2$ applications of N^{-1}. If $\underline{z}^{(0)}=\underline{0}$, this number is reduced to $j+1$. However, $\underline{z}^{(0)}=\underline{b}$ is normally a better initial guess than $\underline{z}^{(0)}=\underline{0}$.

For later reference, we also state the second commonly used form of the algorithm.

Algorithm 4.2b (conjugate gradient algorithm, second form):

Choose $\underline{u}^{(0)} \in R^n$.

$\underline{g}^{(0)} := \underline{b} - M\underline{u}^{(0)}$

Replace $\underline{g}^{(0)}$ by its orthogonal projection onto the range of M.

$\tilde{\underline{g}}^{(0)} := N^{-1}\underline{g}^{(0)}$

$\tilde{\underline{d}}^{(0)} := \tilde{\underline{g}}^{(0)}$

For $j = 0,1,2,\ldots$:

$$\alpha^{(j)} := \frac{\underline{g}^{(j)T}\tilde{\underline{g}}^{(j)}}{\tilde{\underline{d}}^{(j)T}M\tilde{\underline{d}}^{(j)}}$$

$\underline{u}^{(j+1)} := \underline{u}^{(j)} + \alpha^{(j)}\tilde{\underline{d}}^{(j)}$

$\underline{g}^{(j+1)} := \underline{g}^{(j)} - \alpha^{(j)}M\tilde{\underline{d}}^{(j)}$

Replace $\underline{g}^{(j+1)}$ by its orthogonal projection onto the range of M.

$\tilde{\underline{g}}^{(j+1)} := N^{-1}\underline{g}^{(j+1)}$

$$\beta^{(j)} := \frac{\tilde{\underline{g}}^{(j+1)T}\underline{g}^{(j+1)}}{\tilde{\underline{g}}^{(j)T}\underline{g}^{(j)}}$$

$\tilde{\underline{d}}^{(j+1)} := \tilde{\underline{g}}^{(j+1)} + \beta^{(j)}\tilde{\underline{d}}^{(j)}$

The sequence $\underline{u}^{(j)}$ converges to a solution of $M\underline{u} = \underline{b}$.

Notice that the vectors $d^{(j)}$ are not needed here. One easily sees that the two algorithms are equivalent. Here the computation of $\underline{u}^{(j)}$, starting with any $\underline{u}^{(0)}$, requires only j applications of N^{-1}.

We now consider using the conjugate gradient method for the equation

$$C\underline{y}_2 = \underline{b}_2 - Q\underline{b}_1.$$

The difficulty is that C is virtually never symmetric and positive definite with respect to the euclidean inner product. To see this, note that C is a principal minor of $A\tilde{A}^{-1}$, which is non-symmetric in general even if A and $\tilde{A}$ are symmetric and positive definite. However, C

is often symmetric and positive definite with respect to an appropriately chosen inner product. We shall now demonstrate this fact and show how to exploit it. We note that a similar result holds in the continuous case; see Proskurowski and Widlund (1980).

We use the notation

$$\tilde{A} = \begin{pmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ \tilde{A}_{21} & \tilde{A}_{22} \end{pmatrix}, \quad \tilde{A}^{-1} =: \tilde{B} = \begin{pmatrix} \tilde{B}_{11} & \tilde{B}_{12} \\ \tilde{B}_{21} & \tilde{B}_{22} \end{pmatrix}, \quad \tilde{A}_{11}, \tilde{B}_{11} \in R^{p \times p}, \ \tilde{A}_{22}, \tilde{B}_{22} \in R^{q \times q}. \tag{4.4}$$

We make no assumptions of positive definiteness yet, but assume from here on that not only $\tilde{A}$, but also the block $\tilde{A}_{11}$ is invertible. The matrix

$$S := \tilde{A}_{22} - \tilde{A}_{21}\tilde{A}_{11}^{-1}\tilde{A}_{12}, \tag{4.5}$$

is then well-defined. It is called the Schur complement of $\tilde{A}$ with respect to $\tilde{A}_{11}$; see Cottle (1974). It is easy to prove the following proposition.

Proposition 4.1: S is invertible, with

$$S^{-1} = \tilde{B}_{22}. \tag{4.6}$$

We collect some statements about C, S and A which we shall use later:

Proposition 4.2: (i) $\underline{x} \in \ker(CS)$ if and only if $\begin{pmatrix} \tilde{B}_{12} S\underline{x} \\ \underline{x} \end{pmatrix} \in \ker(A)$. In particular, $\dim \ker(A) = \dim \ker(C) = \dim \ker(CS)$.

(ii) If A is symmetric, then CS is symmetric.

(iii) If A is positive (semi-)definite, then CS is positive (semi-)definite.

Proof: (i) $\underline{x} \in \ker(CS) <=> S\underline{x} \in \ker(C) <=> \begin{pmatrix} 0 \\ S\underline{x} \end{pmatrix} \in \ker(A\tilde{A}^{-1})$. The last equivalence follows from (4.2). Thus, we have $\underline{x} \in \ker(CS) <=> \tilde{A}^{-1}\begin{pmatrix} 0 \\ S\underline{x} \end{pmatrix} \in \ker(A)$, and the assertion follows from (4.4) and (4.6).

(ii) A is symmetric if and only if $\tilde{A}^{-T} A \tilde{A}^{-1}$ is symmetric. Since

$$\tilde{A}^{-T} A \tilde{A}^{-1} = \tilde{A}^{-T} \begin{pmatrix} I & 0 \\ Q & C \end{pmatrix} = \begin{pmatrix} \tilde{B}_{11}^T & \tilde{B}_{21}^T \\ \tilde{B}_{12}^T & \tilde{B}_{22}^T \end{pmatrix} \begin{pmatrix} I & 0 \\ Q & C \end{pmatrix} = \begin{pmatrix} * & * \\ * & S^{-T} C \end{pmatrix}, \tag{4.7}$$

(ii) follows.

(iii) The foregoing computation also proves (iii). For (iii), we need not assume that A and CS are symmetric. □

We remark that the factorization $C = CS \cdot S^{-1}$ of the capacitance matrix can also be written in the following way. Assume, as before, that $\tilde{A}$ and $\tilde{A}_{11}$ are invertible. As mentioned above, we have

$$S = \tilde{B}_{22}^{-1}. \tag{4.8}$$

If A is also invertible, with $A^{-1} = B$, then we have

$$CS = B_{22}^{-1}. \tag{4.9}$$

(In particular, our assumptions imply the invertibility of B_{22}.) (4.9) is proved by the following computation.

$$B = \tilde{B}(A\tilde{B})^{-1} = \begin{pmatrix} \tilde{B}_{11} & \tilde{B}_{12} \\ \tilde{B}_{21} & \tilde{B}_{22} \end{pmatrix} \begin{pmatrix} I & 0 \\ Q & C \end{pmatrix}^{-1} = \begin{pmatrix} \tilde{B}_{11} & \tilde{B}_{12} \\ \tilde{B}_{21} & \tilde{B}_{22} \end{pmatrix} \begin{pmatrix} I & 0 \\ -C^{-1}Q & C^{-1} \end{pmatrix} = \begin{pmatrix} * & * \\ * & S^{-1}C^{-1} \end{pmatrix}.$$

(4.9) shows that, if A_{11} is invertible, CS is the Schur complement of A with respect to A_{11}. Proposition 4.2 could, of course, be derived from this. Notice, however, that our proof of proposition 4.2 assumes neither A nor A_{11} to be invertible.

We shall now present two different versions of the iterative imbedding method, algorithms 4.3 and 4.4.

Algorithm 4.3a: Assume that CS is symmetric and positive semi-definite, and that S is symmetric and positive definite. To solve $C\underline{z} = \underline{b}_2 - Q\underline{b}_1$, apply the conjugate gradient method (algorithm 4.2a) to $CS\underline{u} = \underline{b}_2 - Q\underline{b}_1$, using S as a preconditioner for CS.

This method can be implemented in an efficient way. To see this, it is useful to observe its connection with the following variant.

Algorithm 4.3b: Assume that A is symmetric and positive semi-definite, and that $\tilde{A}$ is symmetric and positive definite. Solve $A\underline{x} = \underline{b}$ with algorithm 4.2a, using $\tilde{A}$ as a preconditioner.

Algorithms 4.3a and 4.3b are essentially the same. More precisely:

Proposition 4.3: If the assumptions of algorithm 4.3a are satisfied, then algorithm 4.3b can be carried out, even if the assumptions of algorithm 4.3b are violated, provided that the initial guess is of the form

$$\begin{pmatrix} \underline{b}_1 \\ \underline{z}_2^{(0)} \end{pmatrix}. \tag{4.10}$$

The method generates a sequence of vectors

$$\begin{pmatrix} \underline{b}_1 \\ \underline{z}_2^{(j)} \end{pmatrix}, \tag{4.11}$$

where $\underline{z}_2^{(j)}$ is the sequence obtained with algorithm 4.3a, starting at $\underline{z}_2^{(0)}$.

Proof: Straightforward induction on j. □

If CS is positive semi-definite but not invertible, the conjugate gradient algorithm for a system with the matrix CS requires the numerical computation of orthogonal projections onto the orthogonal complement of the kernel of CS. For this purpose, it is important to give a simple description of $\ker(CS)$. The following proposition pertains to a special case.

Proposition 4.4: If $\ker(A)$ is spanned by the constant vector

$$\begin{pmatrix} 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{pmatrix} \in R^n,$$

then $\ker(CS)$ is spanned by

$$\begin{pmatrix} 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{pmatrix} \in R^q.$$

Proof: By proposition 4.2 (i), $\underline{x} \in \ker(CS)$ implies that $\begin{pmatrix} \tilde{B}_{12} S \underline{x} \\ \underline{x} \end{pmatrix} \in \ker(A)$, and therefore, in our case, that $\underline{x}$ is a constant vector. Since $\dim \ker(CS) = \dim \ker(A) = 1$, the assertion follows. □

If algorithm 4.3b is used, then the projections onto the orthogonal complement of

$\ker(CS)$ can be carried out easily if a basis of $\ker(CS)$ is known. The following observation, a consequence of the proof of proposition 4.3, is useful: The sequence of residuals obtained using algorithm 4.3b is

$$\begin{pmatrix} 0 \\ \underline{q}_2^{(j)} \end{pmatrix},$$

where $\underline{q}_2^{(j)}$ are the residuals obtained with algorithm 4.3a.

If algorithm 4.3 is not applicable, one may, for example, use the system of normal equations:

Algorithm 4.4: To solve $C\underline{z}=\underline{r}$, apply the conjugate gradient method without preconditioning to the system $C^T C\underline{z}=C^T\underline{r}$.

The use of the normal equations is the simplest way of treating non-symmetric problems with the conjugate gradient method. There are, however, more sophisticated possibilities; see Eisenstat, Elman and Schultz (1983), Elman, Saad and Saylor (1986), and Saad and Schultz (1985).

5. Neumann problems

5.1. Description of the algorithms

The finite element discretization of the Neumann problem (2.1) leads to the symmetric system

$$\begin{pmatrix} K_{11} & K_{13} \\ K_{13}^T & K_{33}^{(1)} \end{pmatrix} \begin{pmatrix} \underline{x}_1 \\ \underline{x}_3 \end{pmatrix} + c \begin{pmatrix} M_{11} & M_{13} \\ M_{13}^T & M_{33}^{(1)} \end{pmatrix} \begin{pmatrix} \underline{x}_1 \\ \underline{x}_3 \end{pmatrix} = \begin{pmatrix} \underline{b}_1 \\ \underline{b}_3 \end{pmatrix}, \tag{5.1}$$

which is positive semi-definite if $c = 0$ and positive definite if $c > 0$. The notation is as in section 1.

As a preconditioner for the matrix

$$\begin{pmatrix} K_{11} & K_{13} \\ K_{13}^T & K_{33}^{(1)} \end{pmatrix} + c \begin{pmatrix} M_{11} & M_{13} \\ M_{13}^T & M_{33}^{(1)} \end{pmatrix}, \tag{5.2}$$

we first consider

$$\begin{aligned} \left[\begin{pmatrix} I & 0 & 0 \\ 0 & 0 & I \end{pmatrix} \left(K + cM\right)^{-1} \begin{pmatrix} I & 0 \\ 0 & 0 \\ 0 & I \end{pmatrix} \right]^{-1} &= \begin{pmatrix} G_{11}(c) & G_{13}(c) \\ G_{13}(c)T & G_{33}(c) \end{pmatrix}^{-1} \\ &= \begin{pmatrix} K_{11}(c) & K_{13}(c) \\ K_{13}(c)^T & K_{33}^{(1)}(c) + S^{(2)} \end{pmatrix}, \end{aligned} \tag{5.3}$$

where $S^{(2)} := K_{33}(c)^{(2)} - K_{23}(c)^T K_{22}(c)^{-1} K_{23}(c)$. The last identity in (5.3) is obtained by a straightforward computation.

The resulting method is known to be optimal; see Widlund (1986). Related results for finite difference methods have been proved by Astrakhantsev (1978), Shieh (1978) by using quite different techniques. The bounds given by Astrakhantsev (1978) and Widlund (1986) rely on an extension theorem for finite element functions which parallels well-known results for Sobolev spaces. If any finite element function on Ω^h can be extended to a finite element function on $(0;1)^2$ with an increase in energy by at most a factor C, then the condition number of the preconditioned matrix equals C. For general conforming finite elements, the existence of an h-independent bound C of this kind has been proved by Widlund (1986).

Because of the triangles near $\partial\Omega$, M does not have the same stencil everywhere, and $K+cM$ cannot be inverted using a fast solver on $(0;1)^2$. In (5.3), we therefore replace this matrix by a more convenient spectrally equivalent matrix. A first possibility is

$$\hat{K} + ch^2 I. \tag{5.4}$$

As in section 3, $\hat{K}$ is the matrix defined in the same way as K, but on the regular triangulation $(\hat{\tau}_\nu)_{1\leq\nu\leq 2N^2}$; see (3.19).

In the case of linear elements, the matrix (5.4) is described by the 5-point difference star

$$\begin{array}{ccc} & -1 & \\ -1 & 4+ch^2 & -1 \\ & -1 & \end{array} \tag{5.5}$$

Triangles of type (2), give the following.

$$\begin{array}{ccc} & -\frac{4}{3} & \\ -\frac{4}{3} & \frac{16}{3}+ch^2 & -\frac{4}{3} \\ & -\frac{4}{3} & \end{array} \tag{5.6}$$

in points $(i_1\frac{h}{2}, i_2\frac{h}{2})$ with odd i_1 or odd i_2, and

$$\begin{array}{ccccc} & & \frac{1}{3} & & \\ & & -\frac{4}{3} & & \\ \frac{1}{3} & -\frac{4}{3} & 4+ch^2 & -\frac{4}{3} & \frac{1}{3} \\ & & -\frac{4}{3} & & \\ & & \frac{1}{3} & & \end{array} \tag{5.7}$$

in points $(i_1\frac{h}{2}, i_2\frac{h}{2})$ with even i_1 and even i_2. (5.6) and (5.7) are difference stars with the mesh size $\frac{h}{2}$. For $c=0$, (5.5) and (5.6-7) describe $\hat{K}$ based on a triangulation of $(0;1)^2$ obtained by cutting each of the cells $[i_1h\,;(i_1+1)h\,]\times[j_1h\,;(j_1+1)h\,]$ along one of its diagonals. This result is independent of the choice of the diagonals. This direction independence is,

however, accidental. For triangles of type (3), the direction in which the cells are cut will effect the stencils.

As an alternative to (5.6-7) we could also use the operator (5.5) on the grid $\hat{\Gamma}^{h/2}$ as a preconditioner in the case of quadratic elements. It can easily be shown that (5.5) and (5.6-7) are spectrally equivalent operators, by comparing the corresponding quadratic forms, element by element.

In addition to (5.5) or (5.6-7), we must specify boundary conditions for $x_1=0$, $x_1=1$, $x_2=0$, $x_2=1$. They should be chosen such that the resulting discrete Helmholtz problems can be treated by a fast solver. The choice of the boundary conditions on $\partial(0;1)^2$ will be discussed further in section 5.2.

We report now on numerical experiments illustrating the performance of several different versions of the method for the following test regions; compare Figures 1-6.

$$\{(x_1,x_2) : [(x_1-0.5)^2+(x_2-0.5)^2]^{1/2} < 0.4\} \tag{5.8}$$

$$\{(x_1,x_2) : [(x_1-0.5)^2+(x_2-0.5)^2]^{1/2} \in (0.1;0.4)\} \tag{5.9}$$

$$\{(x_1,x_2) : [(x_1-0.5)^2+(x_2-0.5)^2]^{1/2} < 0.4,\ x_1<0.5 \text{ or } x_2<0.5\} \tag{5.10}$$

$$(0.2;0.8)^2 \tag{5.11}$$

$$(0.2;0.8)^2 - [0.475;0.525]\times[0.2;0.5] \tag{5.12}$$

5.2. The choice of boundary conditions for the auxiliary Helmholtz problems

We shall present numerical comparisons of the following boundary conditions on $\partial(0;1)^2$:

(i) Periodicity conditions at $x_1=0$, $x_1=1$ and homogeneous Dirichlet conditions at $x_2=0$, $x_2=1$.

(ii) Homogeneous Dirichlet conditions on the entire boundary $\partial(0;1)^2$.

(iii) Homogeneous Neumann conditions on the entire boundary $\partial(0;1)^2$.

The condition number of the preconditioned matrix is minimized when homogeneous Neumann boundary conditions are chosen, since the minimum energy extension to $(0;1)^2$of a finite element function on Ω^h satisfies, in the discrete sense, homogeneous Neumann boundary conditions on $\partial(0;1)^2$. If the constant c in eq. (2.1) is zero, the solution of the auxiliary problem is unique only up to an additive constant. We then require the solution to have zero average in $\overline{\Omega}^h$.

We use algorithm 2.2a and count the number of calls to the fast solver on $\partial(0;1)^2$ required to reduce the euclidean norm of the residual by a factor $<10^{-6}$. Here the word residual refers to the quantity which is denoted by $\underline{g}$ in algorithm 2.2a. In the notation of algorithm 2.2a, the initial approximation $\underline{z}_0$ is taken to be the right-hand side $\underline{b}$. One could also use $\underline{z}_0=0$. The resulting condition number estimates are equal to those obtained with $\underline{z}_0=\underline{b}$, but the number of iterations needed to reach the desired accuracy is often by 1 or 2 larger.

We apply the method to the problem

$$-\Delta u + cu = \text{const.} + \sin(x_1+x_2) \text{ on } \Omega \tag{5.13}$$

$$\frac{\partial u}{\partial n} = 0 \text{ on } \partial\Omega, \tag{5.14}$$

where the constant is chosen such that the discrete compatibility condition is satisfied.

Some numerical results for $c=0$, using linear elements, are shown in Table I. Note that the number of calls to the fast solver is the number of iterations plus two. Our results confirm that Neumann conditions on $\partial\Omega$ are the best choice, but also suggest that the choice of boundary conditions on $\partial(0;1)^2$ is of no great importance.

5.3. The choice of the discrete Helmholtz operator in the case of quadratic elements

Table II shows results analogous to those in Table I, but using piecewise quadratic elements. We use (i) (5.6-7) with periodicity conditions in the x_1-direction and homogeneous

Dirichlet conditions in the x_2-direction, (ii) (5.5) with mesh width $\frac{h}{2}$ and homogeneous Neumann conditions on $\partial(0;1)^2$, and (iii) (5.5) with mesh width $\frac{h}{2}$ and homogeneous Dirichlet conditions on $\partial(0;1)^2$. We conclude that the use of (5.5) instead of (5.6-7) leads to a slight increase in the number of calls to the fast solver. (5.5) may, however, be preferable, since the implementation of a fast solver is more difficult for (5.6-7) than for (5.5).

We briefly indicate a way of constructing an FFT-based fast solver for problems involving the operator (5.6-7). We confine ourselves to the case of periodicity conditions at $x_1=0$, $x_1=1$ and homogeneous Dirichlet conditions at $x_2=0$, $x_2=1$, and assume that N is even.

Any grid function $u^h(x_1,x_2)$ on

$$\{(x_1,x_2)=(i_1\frac{h}{2},i_2\frac{h}{2}) : 0\leq i_1\leq 2N-1,\ 1\leq i_2\leq 2N-1\} \tag{5.15}$$

has a unique expansion of the form

$$u^h(x_1,x_2) = \frac{1}{2}A_0(x_2) + \sum_{k=1}^{N-1} A_k(x_2)\cos(2\pi k x_1) + \sum_{k=1}^{N-1} B_k(x_2)\sin(2\pi k x_1) + \frac{1}{2}A_N(x_2)\cos(2\pi N x_1). \tag{5.16}$$

Using the Fast Fourier Transform, such expansions can be computed and evaluated in $O(N^2\log N)$ operations.

To develop a fast solver for (5.6-7) based on (5.16), we must consider the result $r^h(x_1,x_2)$ of applying (5.6-7) to a function of the form (5.16). $r^h(x_1,x_2)$ has the expansion

$$r^h(x_1,x_2) = \frac{1}{2}C_0(x_2) + \sum_{k=1}^{N-1} C_k(x_2)\cos(2\pi k x_1) + \sum_{k=1}^{N-1} D_k(x_2)\sin(2\pi k x_1) + \frac{1}{2}C_N(x_2)\cos(2\pi N x_1). \tag{5.17}$$

A straightforward computation shows that the coefficient $C_k(x_2)$, for a fixed index k and a given x_2, depends only on $A_k(y)$ and $A_{k'}(y)$, where

$$k' = N-k \tag{5.18}$$

and $|y-x_2|\leq 2h$. Similarly, $D_k(x_2)$ depends only on $B_k(y)$ and $B_{k'}(y)$. The derivation of these results, and of the systems of linear equations which describe the relations between A_k, B_k, C_k and D_k, use the formulae

$$\cos(2\pi k' \; i\frac{h}{2}) = \cos(2\pi k i\frac{h}{2}) \text{ if } i \text{ is even,} \tag{5.19}$$

$$\cos(2\pi k' \; i\frac{h}{2}) = -\cos(2\pi k i\frac{h}{2}) \text{ if } i \text{ is odd,} \tag{5.20}$$

and similar formulae for the sine.

For each pair (k,k'), one obtains a system of linear equations relating the $A_k, A_{k'}$ to the $C_k, C_{k'}$, and a system relating the $B_k, B_{k'}$ to the $D_k, D_{k'}$. These systems are block pentadiagonal, with blocks of size 2×2. Within the nonzero blocks, there is considerable additional sparsity.

An different solver can be derived as follows. Consider the four variables associated with the points $(2i_1\frac{h}{2},2i_2\frac{h}{2})$, $((2i_1+1)\frac{h}{2},2i_2\frac{h}{2})$, $(2i_1\frac{h}{2},(2i_2-1)\frac{h}{2})$, $((2i_1+1)\frac{h}{2},(2i_2-1)\frac{h}{2})$ as a four-vector-valued variable. Then the operator (5.6-7) is a five-point operator, operating on four-vector-valued grid functions. We can then use the Fast Fourier Transform with respect to one variable to reduce the linear system to block tridiagonal systems, with blocks of size 4×4. Alternatively, if the problem is doubly periodic on $[0;1]^2$, then the Fast Fourier Transform with respect to both variables can be used. This results in N^2 4×4 linear systems of equations. We note that this systematic technique has been used by Bjørstad and Widlund (1981) to develop a fast solver for a conforming finite element approximation of the biharmonic equation on regular hexagonal meshes.

5.4. Inexact solution of the auxiliary problems on the square

Instead of using a direct fast solver, one can attempt to use an approximate solver to increase the efficiency of the method. We use (5.5) with homogeneous Dirichlet conditions on $\partial(0;1)^2$. As an approximate solver, we use a multigrid cycle constructed such that the effective preconditioner is symmetric and positive definite and is spectrally equivalent with the exact preconditioner. A cycle satisfying these conditions can easily be found. We use a V-cycle (see Stüben and Trottenberg (1982)), with red-black Gauss-Seidel sweeps arranged in a symmetric way relative to the coarse grid correction step. The total amount of work

required for the cycle corresponds to 5-6 Gauss-Seidel iteration steps. Results with this method are shown in Table III.

5.5. The case $c > 0$

If Ω, h and f ,g are fixed, the number of iterations required to reduce the residual by a prescribed factor increases as $c \to 0$, and it is significantly larger for $c > 0$, $c \approx 0$ than for $c = 0$. This fact will prove useful in the following section, where exterior Neumann problems for the Helmholtz equation will be used as auxiliary problems in a Dirichlet solver.

If $c = 0$, then $G(c)^{1/2}K(c)G(c)^{1/2}$ is uniformly well-conditioned in the sense that the quotient of the largest and the smallest *nonzero* eigenvalue is bounded uniformly in h. Notice, however, that $G(c)^{1/2}K(c)G(c)^{1/2}$ has a simple zero eigenvalue. If $c > 0$, $c \approx 0$, then the condition number of $G(c)^{1/2}K(c)G(c)^{1/2}$ is large, by the continuity of the eigenvalues. There is only one outlying eigenvalue, namely one near 0. For the conjugate gradient method, a small outlying eigenvalue is more harmful than a large one; see Jennings (1977). Denote the eigenvalues of $G(c)^{1/2}K(c)G(c)^{1/2}$ by $0 < \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_n$. Following Jennings, the number of conjugate gradient iterations needed to achieve a fixed accuracy ϵ may increase by at most

$$\sqrt{\frac{\lambda_n}{\lambda_2}}\left(1+\frac{1}{2}\log\left(\frac{\pi\lambda_2}{4\lambda_1}\right)\right). \tag{5.21}$$

in comparison with the case of $c = 0$. This estimate is independent of ϵ.

Table IV contains results with $c > 0$. For $c = 0.1$, the table also contains the prediction based on Jennings' formula (5.21), i.e. the (rounded) sum of (5.21) and the number of calls to the fast solver needed for $c = 0$. The results show that (5.21) is a good, even though not quite sharp, prediction of the true behaviour.

6. Dirichlet problems on grid-aligned regions

In this section, we assume that all boundary nodes lie on the regular grid. For regions which are not grid-aligned, additional considerations are needed; see section 7. We shall restrict our discussion to the case $c = 0$, i.e. to the Poisson equation. This is, in a certain respect, the hardest case, since the difficulty with a related Neumann problem described in section 6.2.1 and resolved in section 6.2.2 does not occur if $c > 0$ or if the Neumann problem otherwise is known to be non-singular. We only use homogeneous Dirichlet boundary conditions on $\partial(0;1)^2$ in this section.

6.1. A non-optimal method

The simplest approach to the Dirichlet problem is to treat it as if it were a Neumann problem, i.e. to solve a problem of the form

$$K_{11}\underline{x}_1 = \underline{b}_1 \tag{6.1}$$

using the conjugate gradient method with the preconditioner G_{11}^{-1}. This method is non-optimal, i.e. the condition number of the preconditioned matrix is not bounded as $h \to 0$, but grows like $O(\frac{1}{h})$. To see this, we consider, without loss of generality, the Dirichlet problem on a region enlarged by one layer of mesh points, i.e. the problem described by

$$\begin{pmatrix} K_{11} & K_{13} \\ K_{13}^T & K_{33} \end{pmatrix}. \tag{6.2}$$

We use the preconditioner

$$\begin{pmatrix} G_{11} & G_{13} \\ G_{13}^T & G_{33} \end{pmatrix}^{-1}. \tag{6.3}$$

As is shown in section 5, (6.3) is spectrally equivalent with

$$\begin{pmatrix} K_{11} & K_{13} \\ K_{13}^T & K_{33}^{(1)} \end{pmatrix}. \tag{6.4}$$

We can therefore compare (6.4) with (6.2). The generalized Rayleigh quotient can be written as

$$\frac{\begin{pmatrix} \underline{x}_1 \\ \underline{x}_3 \end{pmatrix}^T \begin{pmatrix} K_{11} & K_{13} \\ K_{13}^T & K_{33}^{(1)} \end{pmatrix} \begin{pmatrix} \underline{x}_1 \\ \underline{x}_3 \end{pmatrix} + \underline{x}_3^T K_{33}^{(2)} \underline{x}_3}{\begin{pmatrix} \underline{x}_1 \\ \underline{x}_3 \end{pmatrix}^T \begin{pmatrix} K_{11} & K_{13} \\ K_{13}^T & K_{33}^{(1)} \end{pmatrix} \begin{pmatrix} \underline{x}_1 \\ \underline{x}_3 \end{pmatrix}} . \tag{6.5}$$

Our problem can easily be reduced to one of the form

$$K_{11}\underline{x}_1 + K_{13}\underline{x}_3 = \underline{0}. \tag{6.6}$$

The Rayleigh quotient then takes the form

$$1 + \frac{\underline{x}_3^T K_{33}^{(1)} \underline{x}_3}{\underline{x}_3^T S^{(1)} \underline{x}_3}, \tag{6.7}$$

where

$$S^{(1)} = K_{33}^{(1)} - K_{13}^T K_{11}^{-1} K_{13}. \tag{6.8}$$

It can easily be shown that $K_{33}^{(1)}$ is diagonally dominant and that all its eigenvalues are in a fixed interval bounded away from zero. $S^{(1)}$, on the other hand, has some eigenvalues on the order of h. Therefore the condition number grows linearly with $\frac{1}{h}$.

Nevertheless, our experiments have lead to the conclusion that the method is more efficient than one might expect. Unlike the methods of sections 6.2 and 6.3, it requires no modifications on regions which are not grid-aligned. In addition, it has the advantage of permitting the incomplete solution of the auxiliary problems on $(0;1)^2$, while the method of section 6.2 requires the exact solution of those problems.

Table V shows some of our numerical results, using the grid-aligned L-shaped region

$$(0.2;0.8)^2\text{–}[0.5;0.8]^2 \tag{6.9}$$

as well as regions (5.8), (5.9), which are not grid-aligned. The right-hand side is $\sin(x_1+x_2)$, the boundary condition is homogeneous.

6.2. The method using exterior Neumann problems

6.2.1. The method using exterior Poisson problems with Neumann boundary conditions on $\partial\Omega^h$

In this subsection, we assume that Ω^h is simply connected.

Proposition 6.1: The exterior Neumann problem is non-singular, i.e. the matrix

$$E_N = \begin{pmatrix} K_{22} & K_{23} \\ K_{23}^T & K_{33}^{(2)} \end{pmatrix} \tag{6.10}$$

is invertible, if and only if Ω^h is simply connected.

Proof: The kernel of (6.10) consists of those finite element functions on the complement of Ω^h which are zero on $\partial(0;1)^2$ and which are constant on each triangle belonging to the complement of Ω^h. All such functions are zero if and only if the complement of Ω^h is connected, i.e. if and only if Ω^h is simply connected. □

We remark that proposition 6.1 depends on our assumption that homogeneous Dirichlet conditions are used on $\partial(0;1)^2$. If they are replaced by Neumann conditions, then the nullspace will have dimension one or larger, depending on whether Ω^h is simply connected or not.

Without any significant loss, our problem may be assumed to be of the form

$$\begin{pmatrix} K_{11} & K_{13} \\ 0 & I \end{pmatrix} \begin{pmatrix} \underline{x}_1 \\ \underline{x}_3 \end{pmatrix} = \begin{pmatrix} \underline{0} \\ \underline{b}_3 \end{pmatrix}. \tag{6.11}$$

Consider the exterior Neumann problem

$$\begin{pmatrix} K_{22} & K_{23} \\ K_{23}^T & K_{33}^{(2)} \end{pmatrix} \begin{pmatrix} \underline{x}_2 \\ \underline{x}_3 \end{pmatrix} = \begin{pmatrix} K_{23}\underline{b}_3 \\ K_{33}^{(2)}\underline{b}_3 \end{pmatrix}. \tag{6.12}$$

The solution of this problem is $(\underline{0},\underline{b}_3)^T$. Solving it with the preconditioned conjugate gradient algorithm, using

$$\begin{pmatrix} G_{22} & G_{23} \\ G_{23}^T & G_{33} \end{pmatrix}^{-1} \tag{6.13}$$

as the preconditioner, one obtains the solution in the form

$$\begin{pmatrix} \underline{0} \\ \underline{b}_3 \end{pmatrix} = \begin{pmatrix} G_{22} & G_{23} \\ G_{23}^T & G_{33} \end{pmatrix} \begin{pmatrix} \underline{z}_2 \\ \underline{z}_3 \end{pmatrix}. \tag{6.14}$$

Setting

$$\begin{pmatrix} \underline{x}_1 \\ \underline{x}_2 \\ \underline{x}_3 \end{pmatrix} := G \begin{pmatrix} \underline{0} \\ \underline{z}_2 \\ \underline{z}_3 \end{pmatrix}, \tag{6.15}$$

the solution $(\underline{x}_1,\underline{x}_3)^T$ of (6.11) is obtained.

From a slightly different viewpoint, the method can be described as follows. Let

$$A := \begin{pmatrix} K_{11} & 0 & K_{13} \\ 0 & K_{22} & K_{23} \\ 0 & K_{23}^T & K_{33}^{(2)} \end{pmatrix}. \tag{6.16}$$

Then

$$AG = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ * & * & C_N \end{pmatrix}, \tag{6.17}$$

where

$$C_N = K_{23}^T G_{23} + K_{33}^{(2)} G_{33} \tag{6.18}$$

is the capacitance matrix for the exterior Neumann problem. Algorithm 2.3a is applicable. We therefore have a method for solving problems with the matrix A. The discrete Dirichlet problem

$$\begin{pmatrix} K_{11} & K_{13} \\ 0 & I \end{pmatrix} \begin{pmatrix} \underline{x}_1 \\ \underline{x}_3 \end{pmatrix} = \begin{pmatrix} \underline{b}_1 \\ \underline{b}_3 \end{pmatrix} \tag{6.19}$$

can be reformulated as

$$A \begin{pmatrix} \underline{x}_1 \\ \underline{x}_2 \\ \underline{x}_3 \end{pmatrix} = \begin{pmatrix} \underline{b}_1 \\ K_{23}\underline{b}_3 \\ K_{33}^{(2)}\underline{b}_3 \end{pmatrix}. \tag{6.20}$$

We have therefore obtained another method for solving discrete Dirichlet problems.

From proposition 4.3, we immediately conclude that the two methods described above

are essentially equivalent. Notice that equations (6.16)-(6.20) describe the more efficient implementation, avoiding the reduction to the case $\underline{b}_1=\underline{0}$, which costs one call to the Helmholtz solver on $(0;1)^2$.

The method described so far fails if Ω^h is multiply connected, a fact which has apparently been overlooked in the previous literature on finite element imbedding methods. Convergence occurs and is as rapid as on simply connected regions, but the limit is usually not the solution of the problem which we want to solve.

To understand this difficulty, first consider the formulation (6.16)-(6.20). It is then easy to see that one must assume that Ω^h is simply connected: (6.20) is not equivalent with (6.19) if Ω^h is multiply connected.

The difficulty is also present in the formulation (6.11)-(6.15). To see this, suppose that we use the initial guess

$$\begin{pmatrix} \underline{z}_2^{(0)} \\ \underline{z}_3^{(0)} \end{pmatrix} = \begin{pmatrix} K_{23}\underline{b}_3 \\ K_{33}^{(2)}\underline{b}_3 \end{pmatrix} \tag{6.21}$$

for the conjugate gradient iteration. It can easily be shown that the conjugate gradient iteration then converges to a limit

$$\begin{pmatrix} \underline{z}_2^* \\ \underline{z}_3^* \end{pmatrix} \tag{6.22}$$

such that the difference

$$\begin{pmatrix} \underline{z}_2^* \\ \underline{z}_3^* \end{pmatrix} - \begin{pmatrix} \underline{z}_2^{(0)} \\ \underline{z}_3^{(0)} \end{pmatrix} \tag{6.23}$$

is orthogonal to the kernel of E_N (see (6.10)) with respect to the euclidean inner product. Now observe that (6.21) is orthogonal to $\ker(E_N)$, since it lies in the range of E_N. Therefore

$$\begin{pmatrix} G_{22} & G_{23} \\ G_{23}^T & G_{33} \end{pmatrix} \begin{pmatrix} \underline{z}_2^* \\ \underline{z}_3^* \end{pmatrix} = \begin{pmatrix} \underline{0} \\ \underline{b}_3 \end{pmatrix} \tag{6.24}$$

is only possible if

$$\begin{pmatrix} G_{22} & G_{23} \\ G_{23}^T & G_{33} \end{pmatrix}^{-1} \begin{pmatrix} \underline{0} \\ \underline{b}_3 \end{pmatrix} \tag{6.25}$$

is orthogonal to $\ker(E_N)$.

Proposition 6.2: (6.25) is orthogonal to $\ker(E_N)$ for every $\underline{b}_3$ if and only if C_N is invertible, i.e. if and only if Ω^h is simply connected.

Proof: (6.25) is orthogonal to $\ker(E_N)$ if and only if there are $\underline{x}_2, \underline{x}_3$ such that

$$\begin{pmatrix} K_{22} & K_{23} \\ K_{23}^T & K_{33}^{(2)} \end{pmatrix} \begin{pmatrix} \underline{x}_2 \\ \underline{x}_3 \end{pmatrix} = \begin{pmatrix} G_{22} & G_{23} \\ G_{23}^T & G_{33} \end{pmatrix}^{-1} \begin{pmatrix} \underline{0} \\ \underline{b}_3 \end{pmatrix}. \tag{6.26}$$

Using

$$\begin{pmatrix} G_{22} & G_{23} \\ G_{23}^T & G_{33} \end{pmatrix} \begin{pmatrix} K_{22} & K_{23} \\ K_{23}^T & K_{33}^{(2)} \end{pmatrix} = \begin{pmatrix} I & * \\ 0 & C_N^T \end{pmatrix}, \tag{6.27}$$

the assertion follows. □

Table VI illustrates the performance of the method, using the test regions (5.8), (5.10) and (6.9). Since (5.8) and (5.10) are not grid-aligned, we approximate them by grid-aligned regions here, i.e. we move the boundary nodes back onto the associated regular grid points; see section 3.

In implementing the method described in this subsection, it is useful to recall remark 4.1, which implies that the only pieces of the stiffness matrix K which are needed are K_{23}, K_{23}^T and $K_{33}^{(2)}$. This observation leads to a reduction in storage and computational work.

6.2.2. The method using exterior Helmholtz problems with Neumann boundary conditions on $\partial\Omega^h$

There is a straightforward way of overcoming the difficulty in the case when Ω^h is multiply connected while preserving the optimality of the method: Replace (6.16) by

$$A := \begin{pmatrix} K_{11} & 0 & K_{13} \\ 0 & K_{22}+ch^2 I_{22} & K_{23} \\ 0 & K_{23}^T & K_{33}^{(2)}+ch^2 I_{33} \end{pmatrix}, \tag{6.16)'}$$

where $c > 0$ and I_{22}, I_{33} are identity matrices. With the obvious modifications in (6.20), one obtains a method for the Dirichlet problem which is applicable to any region Ω^h, and which

is optimal for any fixed $c > 0$. To see this, we have to show that

$$\begin{pmatrix} G_{22} & G_{23} \\ G_{23}^T & G_{33} \end{pmatrix}^{-1} \tag{6.28}$$

is an optimal preconditioner for the exterior Neumann problem for the Helmholtz equation, i.e. for the matrix

$$\begin{pmatrix} K_{22}(c) & K_{23}(c) \\ K_{23}(c)^T & K_{33}^{(2)}(c) \end{pmatrix}. \tag{6.29}$$

This follows from the well-known fact that

$$\begin{pmatrix} G_{22}(c) & G_{23}(c) \\ G_{23}(c)^T & G_{33}(c) \end{pmatrix}^{-1} \tag{6.30}$$

is an optimal preconditioner for (6.29); see, e.g., Dryja (1983). From Poincare's inequality follows that (6.28) and (6.30) are spectrally equivalent. From section 4.5, we conclude that c should not be chosen too small. Experiments suggest that the precise value of c is not important. We have always chosen $c = 10$. Table VII contains numerical results obtained with this method.

We remark that in contrast with the method in section 6.2.1 for simply connected regions, the method described in the present subsection does require applications of K_{22}.

6.3. The discrete dipole method

The discrete Dirichlet problems is a linear systems of equations with the matrix

$$A := \begin{pmatrix} K_{11} & K_{13} \\ 0 & I \end{pmatrix}. \tag{6.31}$$

The matrix

$$\tilde{B} := \begin{pmatrix} \tilde{B}_{11} & \tilde{B}_{13} \\ \tilde{B}_{31} & \tilde{B}_{33} \end{pmatrix} := \begin{pmatrix} I & 0 & 0 \\ 0 & 0 & I \end{pmatrix} G \begin{pmatrix} I & 0 \\ 0 & K_{23} \\ 0 & K_{33}^{(2)} \end{pmatrix} \tag{6.32}$$

is an approximate inverse of A in the sense of section 2:

$$A\tilde{B} = \begin{pmatrix} I & 0 \\ G_{13}^T & C_D \end{pmatrix} \tag{6.33}$$

with

$$C_D = G_{23}^T K_{23} + G_{33} K_{33}^{(2)}. \tag{6.34}$$

In order to motivate the last factor in the product in (6.32), let us consider

$$\begin{pmatrix} 0 \\ K_{23} \\ K_{33}^{(2)} \end{pmatrix}. \tag{6.35}$$

This matrix is the transpose of the part of the stiffness matrix for the Neumann problem which corresponds to the discretization of the normal derivative. Applying it to a mesh function u^h defined on the boundary nodes, one obtains a mesh function v^h which is nonzero only on boundary nodes and on exterior nodes adjacent to the boundary. v^h resembles the discrete dipole layers used by Astrakhantsev (1977 and 1985), O'Leary and Widlund (1979), Proskurowski and Widlund (1976), and Shieh (1979). We refer to these papers for further discussion of the use of dipole layers and the relation to classical potential theory. In the methods proposed by these authors, the interior layer is located within the region, at distance $O(h)$ from the boundary, while it is located on the boundary in the method studied here.

The capacitance matrix C_D is the transpose of the matrix C_N which arose in our discussion of the exterior Neumann problem; see (6.18). That problem could be solved with a preconditioned conjugate gradient method in spite of the fact that C_N is, in general, nonsymmetric; see sections 4 and 5. It comes as something of a surprise that we have been unable to find a similar device for the problem at hand. We therefore have used algorithm 4.4. As previously shown, $S = G_{33}^{-1}$ is spectrally equivalent with $C_N S$, and C_N is therefore uniformly well-conditioned in the S^{-1}-norm. From this follows that C_D is uniformly well-conditioned in the S-norm, a fact which also could be verified directly. The condition number relevant for our algorithm is, however, $\mathrm{cond}(C_D^T C_D) = \mathrm{cond}(C_N)^2$, where "cond" denotes the condition number with respect to the *euclidean* norm. We have not been able to prove a uniform bound for this number. The numerical results presented below suggest that there should be such a bound but are not quite conclusive. We note that if the boundary

curve is sufficiently smooth, then one can establish the corresponding bound in $L^2(\partial\Omega)$ for the Fredholm integral operators of the second kind which are the continuous counterparts of C_N and C_D. However, in the discrete case, much fewer technical tools are available.

Table VIII contains results for the disk (5.8). Since (5.8) is not grid-aligned, we have moved the boundary nodes onto the nearby points on the regular grid. Table VIII shows $\text{cond}(C_D^T C_D)$, for several values of h. The fourth column of Table VIII contains the number of calls to the fast solver required to reduce the euclidean norm of $\underline{r}-C_D\,\underline{w}$ by 10^{-6} or less, when solving $C_D^T C_D\,\underline{w}=C_D^T\underline{r}$ with the right-hand side $\underline{r}$ arising from the Dirichlet problem

$$\begin{aligned} -\Delta u &= \sin(x_1+x_2) \quad \text{on } \Omega \\ u &= 0 \quad \text{on } \partial\Omega. \end{aligned}$$

To justify this stopping criterion, notice that the residual in $A\underline{x}=\underline{b}$ caused by a residual $\underline{\rho}=\underline{r}-C_D\,\underline{w}$ is $\begin{pmatrix}0\\ \underline{\rho}\end{pmatrix}$. The computation of $\underline{r}$ requires two calls to the fast solver, and an additional call is required to determine the solution after $\underline{w}$ has been computed; compare algorithm 4.1a. These calls are counted in Table VIII. Thus, the numbers in Table VIII are equal to

$$2\cdot\#\text{ of conjugate gradient iterations } + \text{ 3.}$$

Table VIII also contains results for the L-shaped region (6.9) and the triangle

$$\{(x_1,x_2)\in(0.2;0.8)^2 : x_2<1-x_1\}. \tag{6.36}$$

The method is applicable on multiply connected regions as well as on simply connected regions.

6.4. The method using the square root of a discrete Helmholtz operator on $\partial\Omega^h$

We briefly describe a third possible method, which we have not tested numerically.

Let $-\Delta^h$ be the stiffness matrix obtained by discretizing $-\left(\frac{d}{ds}\right)^2$ on $\partial\Omega^h$ using piecewise linear finite elements. $-\Delta^h$ is a positive semi-definite symmetric operator. Set

$$J := \sqrt{-\Delta^h + chI}\,, \text{ with } c>0. \tag{6.37}$$

The discrete Dirichlet problem can be written in the form

$$\begin{pmatrix} K_{11} & K_{13} \\ 0 & J \end{pmatrix} \begin{pmatrix} \underline{x}_1 \\ \underline{x}_3 \end{pmatrix} = \begin{pmatrix} \underline{b}_1 \\ J\underline{b}_3 \end{pmatrix}. \tag{6.38}$$

If one uses the matrix

$$\begin{pmatrix} G_{11} & G_{13} \\ G_{13}^T & G_{33} \end{pmatrix} \tag{6.39}$$

as an approximate inverse of

$$\begin{pmatrix} K_{11} & K_{13} \\ 0 & J \end{pmatrix}, \tag{6.40}$$

then one obtains the capacitance matrix

$$C = JG_{33}. \tag{6.41}$$

Algorithm 4.3 is applicable, but requires knowledge of J. The resulting method is known to be optimal in the sense that the matrix C is uniformly well conditioned in the S^{-1}–norm; compare Bjørstad and Widlund (1984) for the proof of a closely related theorem on domain decomposition methods.

We remark that for $\underline{b}_3$=0, algorithm 4.4 can be used applying J^2 only. However, we have the same theoretical difficulties with this least squares algorithm as with the method of section 6.3.

7. Dirichlet problems on general regions

In this section, we drop the assumption that all boundary nodes lie on the regular grid. When using a fast solver associated with a regular mesh, we now obtain nonzero residuals at certain interior points as well as on the boundary. As we will see, in this case, the method using exterior Neumann problems and the discrete dipole method cannot be applied directly. The non-optimal method of section 6.1 can be used straightforwardly in any case.

7.1. The method using exterior Neumann problems

The method described in section 6.2 is applicable in a certain sense, but inefficient in general. The reason is that (6.13) cannot be replaced by

$$\begin{pmatrix} \hat{G}_{22} & \hat{G}_{23} \\ \hat{G}_{23}^{T} & \hat{G}_{33} \end{pmatrix}^{-1}, \tag{7.1}$$

where $\hat{G}$ is defined in the same way as G, but using the regular triangulation $(\hat{\tau}_\nu)_{1\le\nu\le 2N^2}$ rather than $(\tau_\nu)_{1\le\nu\le 2N^2}$. A fast solver for the auxiliary problems on the square is normally not available.

7.2. The discrete dipole method

One might attempt to use the construction given by (6.32), with G replaced by $\hat{G}$ in (6.32). The resulting capacitance matrix is larger than the one in section 6, with rows and columns corresponding to irregular interior nodes as well as to boundary nodes. Our numerical tests have shown that it is ill-conditioned.

7.3. Outer and inner iterations

We outline a method suggested and analyzed by Dryja (1986) and then study variations of it in detail. We use the notation

$$L = K_{11}. \tag{7.2}$$

We partition this matrix in the following way,

$$L = \begin{pmatrix} L_{00} & L_{01} \\ L_{01}^T & L_{11} \end{pmatrix}. \tag{7.3}$$

Here the subscript 0 refers to interior regular nodes, while 1 refers to interior irregular nodes; see section 3.

Dryja shows that L is spectrally equivalent with

$$\tilde{L} = \begin{pmatrix} L_{00} & 0 \\ 0 & I \end{pmatrix}. \tag{7.4}$$

He then proposes to use $\tilde{L}$ to precondition L, solving problems involving L_{00} with the method discussed in section 6.2.

As a modification of this method, we consider the following method, which we call *DIRCG* (δ) ($\delta \in [0;1)$).

Algorithm 7.1 (*DIRCG* (δ)): Solve the problem $K_{11}\underline{x}_1 = \underline{b}_1$ using the conjugate gradient method in the form of algorithm 4.2b, with preconditioner $\hat{K}_{11}$. Whenever a problem of the form $\hat{K}_{11}\underline{z}_1 = \underline{r}_1$ is to be solved, perform an iteration which reduces the euclidean norm of the residual by the factor δ.

As before, the word residual refers to the quantity denoted by $\underline{q}$ in algorithm 4.2b.

For the outer iteration, we use algorithm 4.2b rather than algorithm 4.2a, for the following reason. If one replaces N^{-1} by any linear or nonlinear approximation, possibly dependent on j, and if $\underline{q}_j \to \underline{0}$ for $j \to \infty$, then $\underline{u}_j$ still converges to a solution of $M\underline{u} = \underline{b}$ in algorithm 4.2b. Even if the $\underline{q}_j$ converge to zero in algorithm 4.2a, it is not clear how to obtain a convergent approximation for $M^{-1}\underline{b}$.

We use the method of section 6.2 in the inner iteration, i.e. for the approximate solution of problems involving the matrix $\hat{K}_{11}$. We denote by *DIRCG* (1) the method obtained if $\hat{G}_{11}^{-1}$ rather than $\hat{K}_{11}$ is used as the preconditioner.

It follows from our results in section 3 that *DIRCG* (0) requires a number of (outer) iterations which is independent of the region as well as of the mesh size. This is confirmed by

Table X, which shows the number of outer iterations required to reduce the euclidean norm of the initial residual by 10^{-6} with the method *DIRCG* (10^{-10}). The right-hand side is $\sin(x_1+x_2)$, and the boundary conditions are homogeneous. *DIRCG* (10^{-10}) is, of course, quite inefficient, since the number of inner iterations is large.

There appears to be no known convergence theory for $\delta\in(0;1)$. We have tested *DIRCG* (δ) for $h=1/150$, using δ=0.05,0.10,0.15,$\cdots$,0.95,1.0, using test regions (5.8) and (5.10). Since (5.8), (5.10) are simply connected, we used the method of section 6.2.1 for the inner iteration. For both regions, a residual reduction by the factor 10^{-6} was accomplished fastest with δ=1.0, requiring 26 calls to the fast solver for region (5.8), and 27 calls for region (5.10). For region (5.8), the second best choice was δ=0.1, requiring 44 calls to the fast solver. For region (5.10), the second best choice was δ=0.05, with 56 calls. Some additional experiments also indicated that δ=1.0 is indeed the best $\delta\in[0;1]$ in the cases considered here.

Instead of the conjugate gradient method, one may use a different iterative method for the outer iteration. We have conducted some numerical experiments with the preconditioned two stage Richardson method; see Young (1971). For this case, a theory has been developed by Golub and Overton (1981). However, our experiments suggest that for our application, the conjugate gradient method is superior to the two stage Richardson method. We note that other outer iteration methods have been considered more recently by Golub and Overton (1986).

8. Summary and discussion of results

For Neumann problems on relatively simple domains, we have found that the finite element imbedding method is quite efficient. Counting the number of operations needed to achieve a prescribed accuracy, it is clear that the method in section 5.4 is the most efficient one among those considered here. The only methods we know of which could be more efficient are multigrid methods. A well-chosen multigrid algorithm, applied to the problem on the irregular region directly, would require substantially less work than the method in section 5.4, especially if the Full Multigrid method were used to solve to truncation error accuracy; compare, e.g., Chan and Saied (1985), Hackbusch (1985), p.94, and Stüben (1982). However, imbedding methods have certain advantages. Their implementation is very much simpler, in particular for higher order finite elements. A useful feature is the complete separation of issues concerning the geometry of the region from those of the solution of the boundary value problems. In section 4, we have described a quite general way of handling the geometry. An additional attractive feature is the delegation of almost all work to a fast Helmholtz solver on a rectangle, which makes it possible to use highly efficient, specialized software, such as the multigrid program MG00 by Foerster and Witsch (1982), or possibly even special hardware.

For Dirichlet problems, the methods have the advantages discussed above but are less efficient, unless the region is grid-aligned. The method using the exterior Neumann problem seems preferable to the one using discrete dipoles.

An alternative domain imbedding method has been described by Dendy (1982). In this method, artificial equations of the form $u^h(\underline{x})=0$ corresponding to mesh points $\underline{x}$ outside the region Ω are added to obtain a problem on the unit square. This problem is solved with a multigrid solver capable of treating equations with strongly discontinuous coefficients. The numerical results presented by Dendy, for a Dirichlet problem on a disk, are very encouraging. Work on a comparison between the two approaches, and possibly the use of our triangulation algorithm in combination with Dendy's method, has begun.

References

Astrakhantsev, G. P., On the Numerical Solution of Dirichlet's Problem in an Arbitrary Region, "Methods of Computational and Applied Mathematics," vol. 2, P.I. Marchuk, ed., Novosibirsk (1977).

Astrakhantsev, G. P., Methods of Fictitious Domains for a Second-Order Elliptic Equation with Natural Boundary Conditions, U.S.S.R. Computational Math. and Math. Phys. 18, No. 1, 114 (1978).

Astrakhantsev, G. P., Numerical Solution of Mixed Boundary Value Problems for Second-Order Elliptic Equations in an Arbitrary Domain, U.S.S.R. Computational Math. and Math. Phys. 25, No. 1, 129 (1985).

Bjørstad, P. E. and O. B. Widlund, unpublished (1981).

Bjørstad, P. E. and O. B. Widlund, Iterative Methods for the Solution of Elliptic Problems on Regions Partitioned into Substructures, Technical Report #136, Computer Science Department, New York University, New York (1984), to appear in SIAM J. Numer. Anal., 1986.

Chan, T. F. and F. Saied, A Comparison of Elliptic Solvers for General Two-Dimensional Regions, SIAM J. Sci. Stat. Comput. 6, No. 3 (1985).

Ciarlet, P. G., "The Finite Element Method for Elliptic Problems," North-Holland, Amsterdam (1978).

Cottle, R., Manifestations of the Schur Complement, Lin. Alg. Appl. 8, 189 (1974).

Dendy, J. E., Black Box Multigrid, J. Comp. Phys. 48, 366 (1982).

Dryja, M., A Finite Element-Capacitance Matrix Method for the Elliptic Problem, SIAM J. Numer. Anal. 20, 671 (1983).

Dryja, M., manuscript (1986)

Eisenstat, S. C., H. C. Elman and M. H. Schultz, Variational Iterative Methods for Nonsymmetric Systems of Linear Equations, SIAM J. Numer. Anal. 20, 345 (1983).

Elman, H. C., Y. Saad and P. E. Saylor, A Hybrid Chebyshev Krylov Subspace Algorithm for Solving Nonsymmetric Systems of Linear Equations, SIAM J. Sci. Stat. Comput. 7, 840 (1986).

Foerster, H. and K. Witsch, Multigrid Software for the Solution of Elliptic Problems on Rectangular Domains: MG00 (Release 1), *in* "Lecture Notes in Mathematics," vol. 960, Springer-Verlag, Berlin (1982).

Garabedian, P., "Partial Differential Equations," John Wiley, New York (1964).

Golub, G. and M. Overton, Convergence of a Two-Stage Richardson Iterative Procedure for Solving Systems of Linear Equations, *in* "Lecture Notes in Mathematics," vol. 912, Springer-Verlag (1982).

Golub, G. and M. Overton, The Convergence of Inexact Richardson and Chebyshev Iterative Methods for Solving Linear Systems, in preparation (1986).

Hackbusch, W., "Multigrid Methods and Applications," Springer-Verlag (1985).

Jennings, A., Influence of the Eigenvalue Spectrum on the Convergence Rate of the Conjugate Gradient Method, J. Inst. Math. Appl. 20, 61 (1977).

Korneev, V. G., Iterative Methods of Solving Systems of Equations for the Finite Element Method, USSR Computational Math. and Math. Phys. 17, no. 5, 109 (1977).

Kuznetsov, J. A. and A. M. Matsokin, A Matrix Analog of the Method of Fictitious Domains, preprint, Novosibirsk Computing Center (1974).

O'Leary, D. P. and O. B. Widlund, Capacitance Matrix Methods for the Helmholtz Equation on General Three-Dimensional Regions, Math. Comp. 33, 849 (1979).

Proskurowski, W., Numerical Solution of Helmholtz's Equation by Implicit Capacitance Matrix Methods, ACM Trans. Math. Software 5, 36 (1979).

Proskurowski, W. and O. B. Widlund, On the Numerical Solution of Helmholtz's Equation by the Capacitance Matrix Method, Math. Comp. 30, 433 (1976).

Proskurowski, W. and O. B. Widlund, A Finite Element-Capacitance Matrix Method for the Neumann Problem for Laplace's Equation, SIAM J. Sci. Stat. Comput. 1, 410 (1980).

Saad, Y. and M. H. Schultz, Conjugate Gradient-Like Algorithms for Solving Nonsymmetric Linear Systems, Math. Comp. 44, 417 (1985).

Shieh, A., On the Convergence of the Conjugate Gradient Method for Singular Capacitance Matrix Equations from the Neumann Problem of the Poisson Equation, Numer. Math. 29, 307 (1978).

Shieh, A., Fast Poisson Solvers on General Two-Dimensional Regions for the Dirichlet Problem, Numer. Math. 31, 405 (1979).

Stüben, K., MG01: A Multi-Grid Program to Solve $\Delta U - c(x,y)U = f(x,y)$ (on Ω), $U = g(x,y)$ (on $\partial\Omega$), on Nonrectangular Bounded Domains Ω, IMA-Report 82.02.02, GMD, St. Augustin (1982).

Stüben, K. and U. Trottenberg, Multigrid methods: fundamental algorithms, model problem analysis and applications, *in* "Lecture Notes in Mathematics," vol. 960, Springer-Verlag, Berlin (1982).

Widlund, O. B., Iterative Methods for Elliptic Problems on Regions Partitioned into Substructures and the Biharmonic Dirichlet Problem, *in* Proceedings of the Sixth International Conference on Computing Methods in Science and Engineering, Versailles, France, December 1983.

Young, D. M., "Iterative Solution of Large Linear Systems," Academic Press, New York (1971).

Table I. Neumann problems. Linear elements. (i) Periodicity conditions at $x_1=0$, $x_1=1$ and homogeneous Dirichlet conditions at $x_2=0, x_2=1$, (ii) homogeneous Dirichlet conditions, (iii) homogeneous Neumann conditions. Number of calls to fast solver required to reduce the euclidean norm of the residual by a factor of 10^{-6}.

region	h	# of calls		
		(i)	(ii)	(iii)
(5.8)	1/50	16	14	13
	1/100	18	16	15
	1/150	17	15	15
	1/200	16	15	14
	1/250	17	16	15
	1/300	18	16	*)
(5.9)	1/50	16	15	13
	1/100	18	17	16
	1/150	17	15	15
	1/200	17	15	15
	1/250	18	16	15
	1/300	18	16	*)
(5.10)	1/50	19	18	16
	1/100	21	20	17
	1/150	20	19	16
	1/200	20	19	16
	1/250	21	20	16
	1/300	22	20	*)

*) not computed

region	h	# of calls		
		(i)	(ii)	(iii)
(5.11)	1/50	13	12	10
	1/100	13	12	11
	1/150	13	12	11
	1/200	13	12	11
	1/250	14	12	11
	1/300	15	13	*)
(5.12)	1/50	23	24	21
	1/100	27	28	25
	1/150	21	21	19
	1/200	18	18	17
	1/250	23	24	21
	1/300	29	29	*)

*) not computed

Table II. Neumann problems. Quadratic elements. (i) (5.6), (5.7) with periodicity conditions at $x_1=0$, $x_1=1$ and homogeneous Dirichlet conditions at $x_2=0$, $x_2=1$, (ii) (5.5) with homogeneous Dirichlet conditions everywhere on $\partial(0;1)^2$, and (iii) (5.5) with homogeneous Neumann conditions everywhere on $\partial(0;1)^2$. Number of calls to fast solver required to reduce the euclidean norm of the residual by a factor of 10^{-6}.

region	h	# of calls		
		(i)	(ii)	(iii)
(5.8)	1/32	19	22	18
	1/64	22	24	21
	1/128	20	23	20
(5.9)	1/32	23	26	22
	1/64	22	23	21
	1/128	21	25	20
(5.10)	1/32	23	25	20
	1/64	25	27	23
	1/128	23	27	22

region	h	# of calls		
		(i)	(ii)	(iii)
(5.11)	1/32	22	26	22
	1/64	16	20	17
	1/128	18	21	18
(5.12)	1/32	26	32	29
	1/64	22	27	25
	1/128	24	28	26

Table III. Neumann problems. Linear elements. One multigrid cycle per auxiliary problem; (i) homogeneous Dirichlet conditions, (ii) homogeneous Neumann conditions on $\partial(0;1)^2$. Number of cycles needed to reduce the euclidean norm of the residual by a factor of 10^{-6}.

region	h	# of calls	
		(i)	(ii)
(5.8)	1/32	15	14
	1/64	17	15
	1/128	17	15
	1/256	17	16
(5.9)	1/32	19	15
	1/64	18	15
	1/128	17	15
	1/256	17	15
(5.10)	1/32	19	16
	1/64	21	18
	1/128	21	17
	1/256	21	17
(5.11)	1/32	21	18
	1/64	17	15
	1/128	18	16
	1/256	19	18
(5.12)	1/32	27	23
	1/64	22	19
	1/128	25	21
	1/256	28	25

Table IV. Neumann problems, c =10.0,1.0,0.1. Linear elements. Homogeneous Dirichlet conditions on the entire boundary $\partial(0;1)^2$. Number of calls to fast solver required to reduce the euclidean norm of the residual by a factor of 10^{-6}. For c =0.1, the prediction given by Jennings' formula is shown in parantheses.

region	h	# of calls
	1/50	15,16,19(22)
	1/100	17,19,21(24)
(5.8)	1/150	16,18,20(23)
	1/200	16,18,20(23)
	1/250	16,18,20(24)
	1/300	17,19,21(24)
	1/50	16,17,20(23)
	1/100	18,20,22(26)
(5.9)	1/150	17,19,21(23)
	1/200	16,18,20(23)
	1/250	17,19,21(24)
	1/300	18,20,22(25)
	1/50	19,21,24(28)
	1/100	20,24,27(31)
(5.10)	1/150	19,23,25(29)
	1/200	18,23,25(29)
	1/250	19,24,26(30)
	1/300	20,24,26(30)

region	h	# of calls
(5.11)	1/50	13,15,15(19)
	1/100	13,15,16(19)
	1/150	13,15,16(19)
	1/200	13,15,16(19)
	1/250	13,15,16(19)
	1/300	14,15,16(20)
(5.12)	1/50	23,26,29(37)
	1/100	27,31,34(43)
	1/150	21,24,26(32)
	1/200	19,21,23(29)
	1/250	24,27,29(37)
	1/300	29,33,36(44)

Table V. Dirichlet problems. Linear elements. Non-optimal method, homogeneous Dirichlet conditions on the entire boundary $\partial(0;1)^2$. Number of calls to fast solver required to reduce the euclidean norm of the residual by a factor of 10^{-6}.

region	h	# of calls
(5.8)	1/50	14
	1/100	22
	1/150	26
	1/200	29
	1/250	32
	1/300	36
(5.9)	1/50	15
	1/100	23
	1/150	28
	1/200	31
	1/250	35
	1/300	39
(6.9)	1/50	12
	1/100	17
	1/150	20
	1/200	24
	1/250	25
	1/300	28

Table VI. Dirichlet problems. Linear elements. Method based on exterior Neumann problems for the Poisson equation, homogeneous Dirichlet conditions on the entire boundary $\partial(0;1)^2$. Number of calls to fast solver required to reduce the euclidean norm of the residual by a factor of 10^{-6}.

region	h	# of calls
(5.8)	1/50	10
	1/100	10
	1/150	10
	1/200	10
	1/250	10
	1/300	10
(5.10)	1/50	12
	1/100	13
	1/150	13
	1/200	13
	1/250	13
	1/300	13
(6.9)	1/50	12
	1/100	13
	1/150	13
	1/200	13
	1/250	13
	1/300	13

Table VII. Dirichlet problems. Linear elements. Method based on exterior Neumann problems for the Helmholtz equation with Helmholtz constant 10, homogeneous Dirichlet conditions on the entire boundary $\partial(0;1)^2$. Number of calls to fast solver required to reduce the euclidean norm of the residual by a factor of 10^{-6}.

region	h	# of calls
(5.9)	1/50	14
	1/100	14
	1/150	14
	1/200	14
	1/250	14
	1/300	14

Table VIII. Dirichlet problems. Linear elements. Dipole method, homogeneous Dirichlet conditions on the entire boundary $\partial(0;1)^2$. Number of calls to fast solver required to reduce the euclidean norm of the residual by a factor of 10^{-6}.

region	h	# of calls
(5.8)	1/50	23
	1/100	25
	1/150	25
	1/200	25
	1/250	25
	1/300	25
(6.9)	1/50	31
	1/100	33
	1/150	35
	1/200	37
	1/250	37
	1/300	37
(6.36)	1/50	39
	1/100	31
	1/150	45
	1/200	47
	1/250	47
	1/300	49

Table IX. Dirichlet problems. Linear elements. Method *DIRCG* (10^{-10}), homogeneous Dirichlet conditions on the entire boundary $\partial(0;1)^2$. Number of outer iterations required to reduce the euclidean norm of the residual by a factor of 10^{-6}.

region	h	# of calls
(5.8)	1/50	5
	1/100	7
	1/150	6
(5.10)	1/50	5
	1/100	6
	1/150	6

Figure 1: Region (5.8).

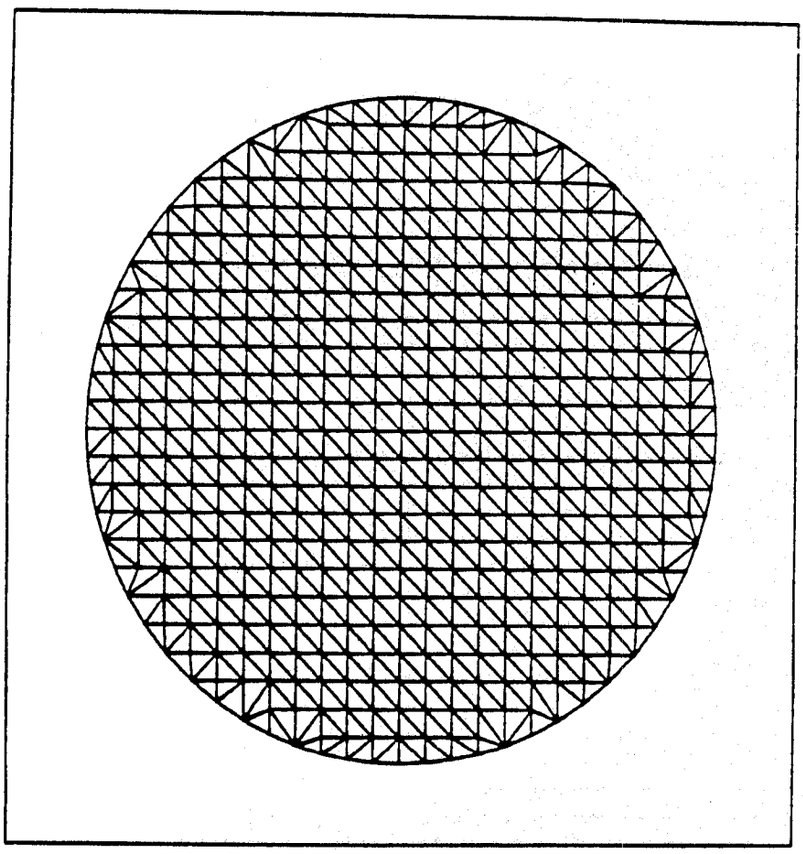

Figure 2: Triangulation of $(0;1)^2$ corresponding to Fig. 1.

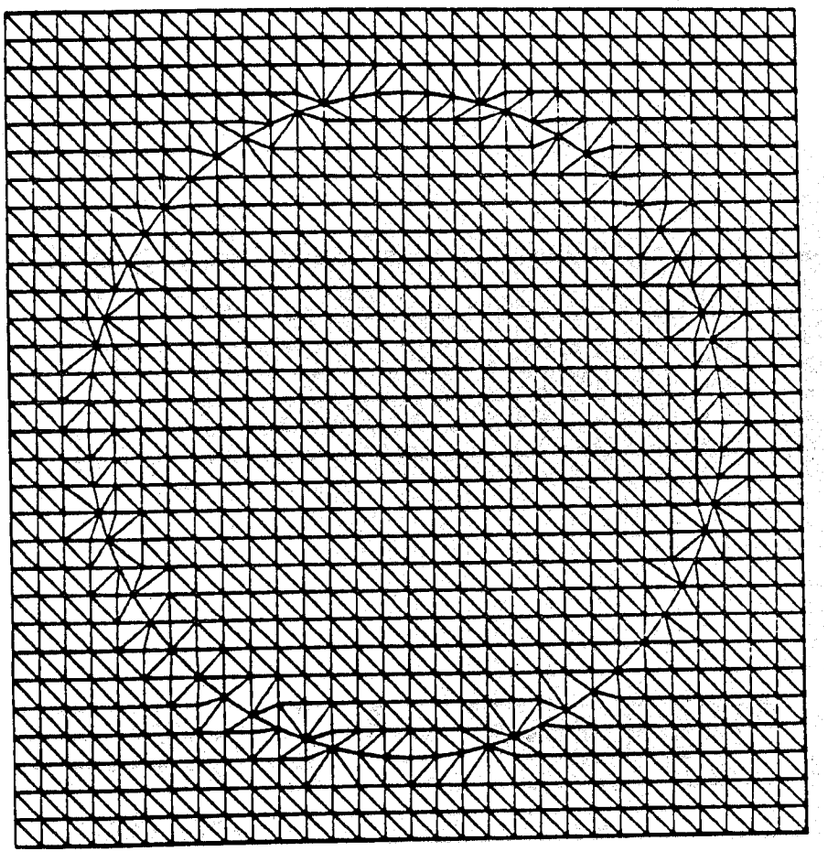

Figure 3: Region (5.9).

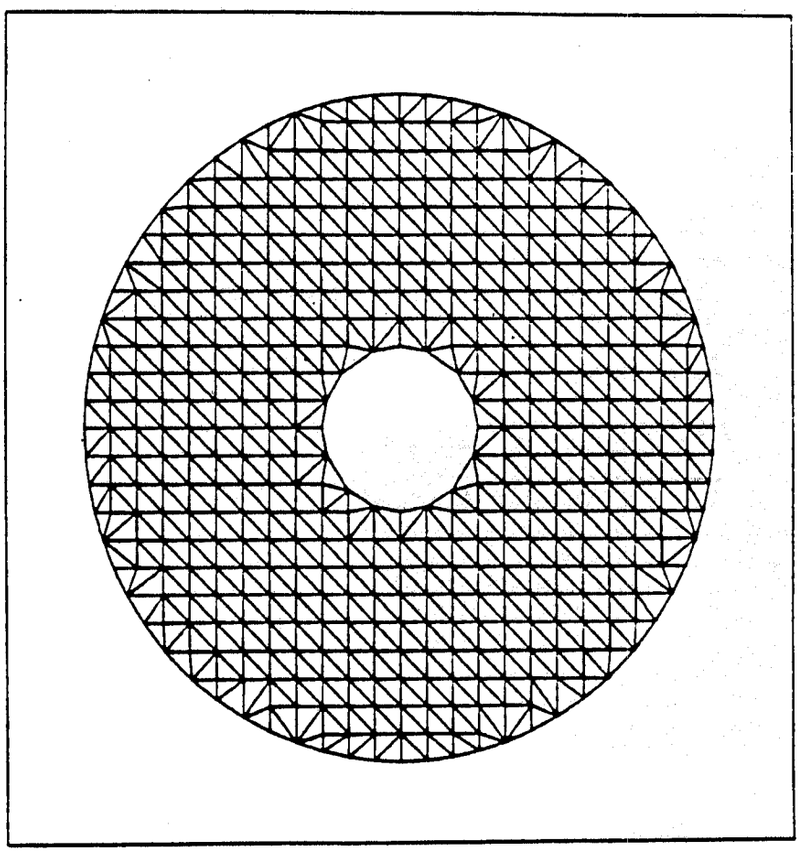

Figure 4: Region (5.10).

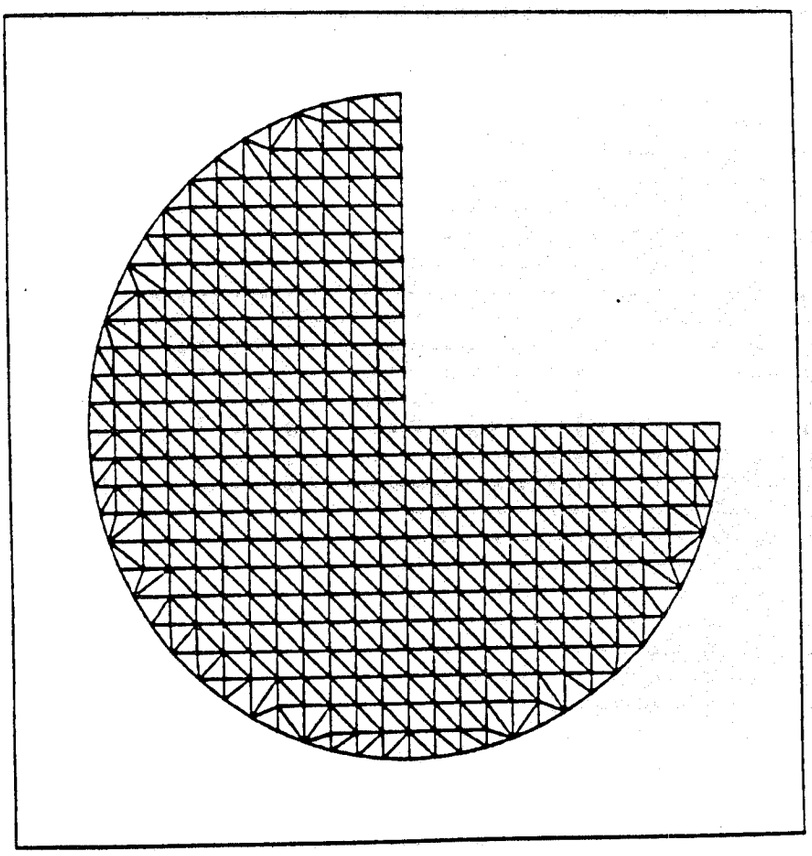

Figure 5: Region (5.11).

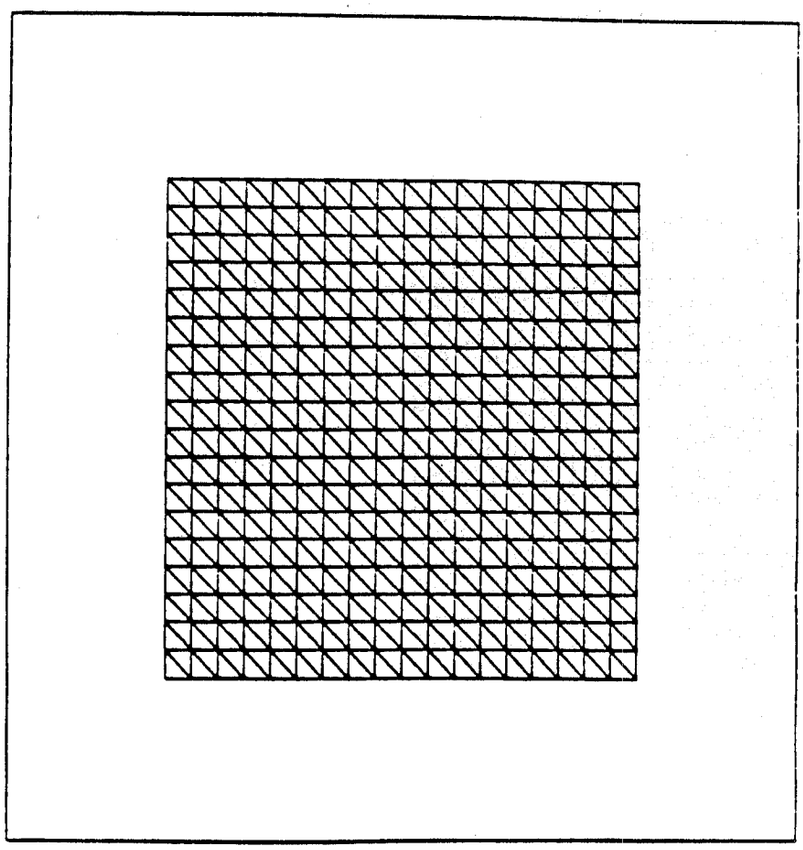

Figure 6: Region (5.12).

