Numerical Algorithms **36:** 95–112, 2004.

A domain embedding method using the optimal distributed control and a fast algorithm

Lori Badea [a] and Prabir Daripa [b,*]

[a] *Institute of Mathematics, Romanian Academy of Sciences, Bucharest, RO-70700, Rumania*
[b] *Department of Mathematics, Texas A&M University, College Station, TX-77843, USA*
E-mail: prabir.daripa@math.tamu.edu

Received 22 April 2003; accepted 24 February 2004
Communicated by C. Brezinski

We propose a domain embedding method to solve second order elliptic problems in arbitrary two-dimensional domains. The method is based on formulating the problem as an optimal distributed control problem inside a disc in which the arbitrary domain is embedded. The optimal distributed control problem inside the disc is solved rapidly using a fast algorithm developed by Daripa et al. [3,7,10–12]. The arbitrary domains can be simply or multiply connected and the proposed method can be applied, in principle, to a large number of elliptic problems. Numerical results obtained for Dirichlet problems associated with the Poisson equation in simply and multiply connected domains are presented. The computed solutions are found to be in good agreement with the exact solutions with moderate number of grid points in the domain.

Keywords: domain embedding methods, optimal distributed control

AMS subject classification: 93B40, 65N99, 65T50

1. Introduction

The embedding or fictitious domain methods which were developed specially in the seventies [1,9,18,28,30,31] have been a very active area of research in recent years because of their appeal and potential for applications in solving problems in complicated domains very efficiently. In these methods, complicated domains Ω where solutions of problems may be sought, are embedded into larger domains D with simple enough boundaries so that the solutions in these embedded domains can be constructed more efficiently. The use of these embedding methods are a commonplace these days for solving complicated problems arising in science and engineering. To this end, it is worth mentioning the domain embedding methods for Stokes equations [6], for fluid dynamics and electromagnetics [15], for the transonic flow calculation [33], and for the equilibrium

[*] Author for correspondence.

of the plasma calculation in a Tokamak [5]. Also, recently there has been an enormous progress in shape optimization using the fictitious domain approaches. We can cite here, for instance, the works of Daňková, Haslinger, Klarbring, Makinen, Neittaanmäki and Tiba (see [13,24–26,29]) among many others.

In [2], an embedding method is associated with a distributed optimal control problem (see [27]). There the problem is solved in an auxiliary domain D using a finite element method on a fairly structured mesh which allows the use of fast solvers. The auxiliary domain D contains the domain Ω and the solution in D is found as a solution of a distributed optimal control problem such that it satisfies the prescribed boundary conditions of the problem in the domain Ω. The same idea is also used in [14] where a least squares method is used. In [16,17], an embedding method is proposed in which a combination of Fourier approximations and boundary integral equations is used. Essentially, a Fourier approximation for a solution of the nonhomogeneous equation in D is found, and then, the solution in Ω for the homogeneous equation is sought using the boundary integral methods.

In recent years, progress in this field has been substantial, especially in the use of the Lagrange multiplier techniques. In this connection, the works of Girault, Glowinski, Hesla, Joseph, Kuznetsov, Lopez, Pan, Périaux [19–23] should be cited.

In a series of papers, Daripa et al. (see [7,10–12]) have developed fast algorithms to evaluate singular integral transforms within a disc and applied to solve a variety of problems. In [3], the fast algorithms were generalized, using an approach somewhat different from the one in [7], to almost all second order inhomogeneous elliptic problems in domains such as a disc, an annulus, and the exterior of a closed disc. In addition, this fast algorithm was applied to solve Poisson and Helmholtz equations with Dirichlet/Neumann conditions prescribed on the boundary of these three types of domains. In this paper, we propose a method to solve Dirichlet problems in arbitrary bounded domains using an embedding domain approach. This method, however, can easily be modified to solve Neumann problems as well. The embedding domain used is a disk as choosing a disc rather than a rectangle for the embedding domain as soon as a strip around the complex domain is used (see below) decreases the size of the optimal control problem which is solved in this proposed method. This method also makes use of a recently developed fast algorithm [3] of the authors to rapidly compute solutions, on the boundary of the irregular domain, of some inhomogeneous problems defined within the embedding disk. The method used in this paper differs from the one given in [4] where the embedding domain is a rectangle in which periodic solutions are constructed. The choice of a disk as an embedding domain brings in some subtleties in the method which are explained in this paper. Moreover, it will be interesting to compare the results of this method with the one reported in [4] which will suggest the course of further research with these methods. To this end, it is worth mentioning the works of Briscolini and Santangelo [8] and Elghaoui and Pasquetti [17].

The paper is organized as follows. In section 2 we provide a fast algorithm for a disc [3], and then couple it with the optimal distributed control problem. The resulting embedding method is then described in detail for the Dirichlet problem associated with the Poisson equation. The problems involving other elliptic equations or different boundary conditions (Neumann or mixed) may be treated in a similar fashion. It should be remarked that the proposed algorithm can be applied to rapidly solving nonhomogeneous elliptic problems in either simply or multiply connected domains. In section 3, we present some numerical examples for both types of domains. Our numerical results for the Dirichlet problem associated with the Poisson equation show the validity and high accuracy of the method. Finally, we provide some concluding remarks in section 4.

2. Description of the method

We consider, for the sake of simplicity, the following Dirichlet problem associated with the Poisson equation

$$\begin{aligned} \Delta u &= f \quad \text{in } \Omega, \\ u &= g \quad \text{on } \partial\Omega, \end{aligned} \tag{2.1}$$

where Ω is a bounded domain, not necessarily simply connected, in $\mathbb{R}^2$. Embedding Ω within a disc D and using an optimal distributed control, we look for an extension $\tilde{f}$ of f from Ω to D, such that $\tilde{f} = f$ in Ω and the trace on $\partial\Omega$ of the solution $\tilde{u}$ of the Dirichlet problem in D,

$$\begin{aligned} \Delta \tilde{u} &= \tilde{f} \quad \text{in } D, \\ \tilde{u} &= 0 \quad \text{on } \partial D, \end{aligned} \tag{2.2}$$

optimally approximates the given function g on $\partial\Omega$. In this way, the restriction of $\tilde{u}$ to Ω optimally approximates the solution u of problem (2.1). The zero boundary condition (2.2) is a natural choice for this Dirichlet problem. Below, we first present the fast algorithm for solving the problem (2.2) in D, and then describe an embedding method that uses this fast algorithm and the techniques of optimal distributed control.

2.1. Fast algorithm for a disc

We assume that the disc D is centered at the origin of the Cartesian coordinate system and has the radius R. Now, using the polar coordinates, we assume that for $0 < r < R$ the function $\tilde{f}$ can be written as a Fourier series

$$\tilde{f}\bigl(r\mathrm{e}^{\mathrm{i}\theta}\bigr) = \sum_{n=-\infty}^{\infty} f_n(r)\mathrm{e}^{\mathrm{i}n\theta}. \tag{2.3}$$

Then the Fourier coefficients $u_n(re^{i\theta})$, $-\infty < n < \infty$, of the solution $\tilde{u}(re^{i\theta})$ of equation (2.2) can be written as (see also [10,12])

$$\begin{aligned}
u_0(r) &= \int_0^r \rho\log(r) f_0(\rho)\,d\rho + \int_r^R \rho\log(\rho) f_0(\rho)\,d\rho - \int_0^R \rho\log(R) f_0(\rho)\,d\rho,\\
&\quad \text{for } n = 0,\\
u_n(r) &= -\int_0^r \frac{\rho}{2|n|}\left(\frac{\rho}{r}\right)^{|n|} f_n(\rho)\,d\rho - \int_r^R \frac{\rho}{2|n|}\left(\frac{r}{\rho}\right)^{|n|} f_n(\rho)\,d\rho\\
&\quad + \left(\frac{r}{R}\right)^{|n|}\int_0^R \frac{\rho}{2|n|}\left(\frac{\rho}{R}\right)^{|n|} f_n(\rho)\,d\rho, \quad \text{for } n \neq 0.
\end{aligned} \tag{2.4}$$

Therefore, the Fourier coefficients of the solution $\tilde{u}$ of problem (2.2) can be directly calculated by evaluating the integrals in (2.4) if one knows the Fourier coefficients of the extended function $\tilde{f}$. In fact, it follows from the equations in (2.4) that we have to compute expressions of the form

$$I_n(r) = I_n^1(r) + I_n^2(r) = \int_0^r \rho u_{1n}(r)u_{2n}(\rho)f_n(\rho)\,d\rho + \int_r^R \rho u_{2n}(r)u_{1n}(\rho)f_n(\rho)\,d\rho, \tag{2.5}$$

where

$$\begin{aligned}
u_{10}(r) &= \log(r), & u_{20}(r) &= 1, & &\text{for } n = 0,\\
u_{1n}(r) &= r^{-|n|}, & u_{2n}(r) &= r^{|n|}, & &\text{for } n \neq 0.
\end{aligned} \tag{2.6}$$

Using these notations, we can write

$$\begin{aligned}
u_0(r) &= I_0(r) - I_0(R), & &\text{for } n = 0,\\
u_n(r) &= -\frac{1}{2|n|}I_n(r) + \frac{1}{2|n|}\left(\frac{r}{R}\right)^{|n|} I_n(R), & &\text{for } n \neq 0.
\end{aligned} \tag{2.7}$$

Choosing a discretization $0 = r_1 < r_2 < \cdots < r_M = R$, not necessarily regular, in the interval $[0, R]$, we can calculate the integrals at the points of this discretization according to the following algorithm.

Algorithm 1.1 (Sequential algorithm for the integrals I_n).

1. Compute $I_n^1(r_m)$, $m = 2, \ldots, M$, as

$$\begin{aligned}
I_n^1(r_2) &= \int_{r_1}^{r_2} \rho u_{1n}(r_2)u_{2n}(\rho)f_n(\rho)\,d\rho,\\
I_n^1(r_m) &= \frac{u_{1n}(r_m)}{u_{1n}(r_{m-1})}I_n^1(r_{m-1}) + \int_{r_{m-1}}^{r_m} \rho u_{1n}(r_m)u_{2n}(\rho)f_n(\rho)\,d\rho,\\
&\quad m = 3, \ldots, M.
\end{aligned} \tag{2.8}$$

2. Compute $I_n^2(r_m)$, $m = M-1, \dots, 1$, as

$$I_n^2(r_{M-1}) = \int_{r_{M-1}}^{r_M} \rho u_{2n}(r_{M-1}) u_{1n}(\rho) f_n(\rho)\, \mathrm{d}\rho,$$
$$I_n^2(r_m) = \frac{u_{2n}(r_m)}{u_{2n}(r_{m+1})} I_n^2(r_{m+1}) + \int_{r_m}^{r_{m+1}} \rho u_{2n}(r_m) u_{1n}(\rho) f_n(\rho)\, \mathrm{d}\rho, \tag{2.9}$$
$$m = M-2, \dots, 1.$$

3. Compute $I_n(r_m)$, $m = 1, \dots, M$, as

$$\begin{aligned} &I_n(r_1) = I_n^2(r_1), \qquad I_n(r_M) = I_n^1(r_M), \\ &I_n(r_m) = I_n^1(r_m) + I_n^2(r_m), \quad m = 2, \dots, M-1. \end{aligned} \tag{2.10}$$

We can still reduce the amount of calculations by observing that $\tilde{f}$ is a real function, i.e. $f_{-n}(r) = \overline{f_n(r)}$. Therefore we have

$$\begin{aligned} I_{-n}^1(r_m) &= \overline{I_n^1(r_m)}, \quad m = 2, \dots, M, \\ I_{-n}^2(r_m) &= \overline{I_n^2(r_m)}, \quad m = 1, \dots, M-1, \\ I_{-n}(r_m) &= \overline{I_n(r_m)}, \quad m = 1, \dots, M. \end{aligned} \tag{2.11}$$

Consequently, we need to apply algorithm 1.1 only for nonnegative n. In order to calculate the solution $\tilde{u}$ of problem (2.2), knowing the extended function $\tilde{f}$, we apply the following procedure:

1. We find the Fourier coefficients of $\tilde{f}$ using the inverse fast Fourier transform.
2. We calculate the Fourier coefficients of $\tilde{u}$ using algorithm 1.1.
3. Using the fast Fourier transform we calculate $\tilde{u}$.

As we will see in the next section, the values of $\tilde{f}$ at the mesh nodes outside Ω are calculated using an optimal distributed control method. In the above steps 1 and 3 we use the fast Fourier transform. We know that if we use $N = 2^L$ terms in the Fourier series then the fast Fourier transform has a net operation count of order $\mathrm{O}(N \log N)$. Therefore, except for the operation count arising from the optimal distributed control part of the overall method outlined at the end of section 2.2, for M nodes in radial direction and N terms in the Fourier series, the net operation count in steps 1–3 is of the order $\mathrm{O}(MN \log N)$ for MN nodes in the domain, or equivalently of the order $\mathrm{O}(\log N)$ per point. This estimate includes the operation count involved in step 2 above which is of lower order. However, this complexity estimate does not include the calculations of the coefficients $f_n(r)$ by the optimal control method of the next section.

2.2. *Optimal distributed control*

The numerical computation of the integrals in (2.4) using algorithm 1.1 assumes the approximation of the Fourier coefficients $f_n(r)$ by some polynomial functions between two consecutive nodes of the discretization $0 = r_1 < r_2 < \cdots < r_M = R$. In the description of the embedding method, we assume $f_n(r)$ to be some continuous functions which are linear between two consecutive nodes. Consequently, denoting such a function by $\varphi_m(r)$, $m = 1, \ldots, M$, which takes the value 1 at r_m and 0 at the other points of the discretization according to the following formula

$$\varphi_m(r) = \begin{cases} \dfrac{r - r_{m-1}}{r_m - r_{m-1}}, & r_{m-1} \leqslant r \leqslant r_m, \\ \dfrac{r_{m+1} - r}{r_{m+1} - r_m}, & r_m \leqslant r \leqslant r_{m+1}, \end{cases} \tag{2.12}$$

we can write

$$f_n(r) = \sum_{m=1}^{M} f_n(r_m)\varphi_m(r). \tag{2.13}$$

With these approximations of the Fourier coefficients and taking N terms in the Fourier series, the function $\tilde{f}$ is approximated by

$$\tilde{f}\big(re^{i\theta}\big) = \sum_{n=-N/2}^{N/2-1} \sum_{m=1}^{M} f_n(r_m)\varphi_m(r)e^{in\theta}. \tag{2.14}$$

We see that $\tilde{f}(re^{i\theta})$ is also linearly approximated between two consecutive radial nodes, and in the following we implicitly assume that $f_n(r)$ and $\tilde{f}(re^{i\theta})$ are of the form (2.13) and (2.14), respectively. For a given r_i, $i = 1, \ldots, M$, from (2.14) we have

$$\tilde{f}\big(r_i e^{i\theta}\big) = \sum_{n=-N/2}^{N/2-1} f_n(r_i)e^{in\theta}, \tag{2.15}$$

and therefore,

$$f_n(r_i) = \frac{1}{N} \sum_{j=0}^{N-1} \tilde{f}\big(r_i e^{i\theta_j}\big)e^{-in\theta_j}, \tag{2.16}$$

where $\theta_j = 2\pi j/N$. Substituting (2.16) into (2.14) we get

$$\tilde{f}\big(re^{i\theta}\big) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \sum_{i=1}^{M} \sum_{j=0}^{N-1} \tilde{f}\big(r_i e^{i\theta_j}\big)\varphi_i(r)e^{in(\theta-\theta_j)}, \tag{2.17}$$

and writing

$$\tilde{f}\big(r_i e^{i\theta_j}\big) = \begin{cases} f\big(r_i e^{i\theta_j}\big) & \text{if } r_i e^{i\theta_j} \in \overline{\Omega}, \\ h\big(r_i e^{i\theta_j}\big) & \text{if } r_i e^{i\theta_j} \in \overline{D}\backslash\overline{\Omega}, \end{cases} \tag{2.18}$$

we have from (2.17)

$$\tilde{f}\big(re^{i\theta}\big) = f\big(re^{i\theta}\big) + h\big(re^{i\theta}\big), \tag{2.19}$$

where

$$f\bigl(r\mathrm{e}^{\mathrm{i}\theta}\bigr) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\sum_{r_i\mathrm{e}^{\mathrm{i}\theta_j}\in\overline{\Omega}} f\bigl(r_i\mathrm{e}^{\mathrm{i}\theta_j}\bigr)\varphi_i(r)\mathrm{e}^{\mathrm{i}n(\theta-\theta_j)} \tag{2.20}$$

and

$$h\bigl(r\mathrm{e}^{\mathrm{i}\theta}\bigr) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\sum_{r_i\mathrm{e}^{\mathrm{i}\theta_j}\in\overline{D}\backslash\overline{\Omega}} h\bigl(r_i\mathrm{e}^{\mathrm{i}\theta_j}\bigr)\varphi_i(r)\mathrm{e}^{\mathrm{i}n(\theta-\theta_j)}. \tag{2.21}$$

In the embedding method, using the optimal distributed control and the above approximation for $\tilde{f}$, we look for the extension $h(r\mathrm{e}^{\mathrm{i}\theta})$ of $\tilde{f}$, in particular for the values $h(r_i\mathrm{e}^{\mathrm{i}\theta_j})$, $r_i\mathrm{e}^{\mathrm{i}\theta_j} \in \overline{D}\backslash\overline{\Omega}$, such that

$$J(h) = \min_{\chi} J(\chi), \qquad J(\chi) = \frac{1}{2}\int_{\partial\Omega}\bigl[\tilde{u}(f+\chi)-g\bigr]^2, \tag{2.22}$$

where the function χ is of the form

$$\chi\bigl(r\mathrm{e}^{\mathrm{i}\theta}\bigr) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\sum_{r_i\mathrm{e}^{\mathrm{i}\theta_j}\in\overline{D}\backslash\overline{\Omega}} \chi_{ij}\varphi_i(r)\mathrm{e}^{\mathrm{i}n(\theta-\theta_j)}, \tag{2.23}$$

χ_{ij} being some real values, and $\tilde{u}(f+\chi)$ is the solution of the problem (2.2) corresponding to $\tilde{f} = f + \chi$. Since $J(\chi)$ is a differentiable convex function, its minimum is the solution of the following equation

$$J'(h)(\chi) \equiv \int_{\partial\Omega}\bigl[\tilde{u}(f+h)-g\bigr]\tilde{u}(\chi) = 0, \quad \text{for any } \chi, \tag{2.24}$$

where $J'(h)(\chi)$ is the Gâteaux derivative of J at h in χ direction. Since the solution $\tilde{u}$ of equation (2.2) depends linearly on the nonhomogeneous term $\tilde{f}$, and noticing that the functions belong to the finite linear space generated by the functions

$$\phi_{ij}\bigl(r\mathrm{e}^{\mathrm{i}\theta}\bigr) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\varphi_i(r)\mathrm{e}^{\mathrm{i}n(\theta-\theta_j)}, \quad 1 \leqslant i \leqslant M,\ 0 \leqslant j \leqslant N-1, \tag{2.25}$$

the equation (2.24) can be written as

$$\begin{aligned}&\sum_{r_i\mathrm{e}^{\mathrm{i}\theta_j}\in\overline{D}\backslash\overline{\Omega}} h_{ij}\int_{\partial\Omega}\tilde{u}\bigl(\phi_{ij}\bigl(r\mathrm{e}^{\mathrm{i}\theta}\bigr)\bigr)\tilde{u}\bigl(\phi_{kl}\bigl(r\mathrm{e}^{\mathrm{i}\theta}\bigr)\bigr)\\ &\quad= \int_{\partial\Omega} g\tilde{u}\bigl(\phi_{kl}\bigl(r\mathrm{e}^{\mathrm{i}\theta}\bigr)\bigr) - \sum_{r_i\mathrm{e}^{\mathrm{i}\theta_j}\in\overline{\Omega}} f_{ij}\int_{\partial\Omega}\tilde{u}\bigl(\phi_{ij}\bigl(r\mathrm{e}^{\mathrm{i}\theta}\bigr)\bigr)\tilde{u}\bigl(\phi_{kl}\bigl(r\mathrm{e}^{\mathrm{i}\theta}\bigr)\bigr),\\ &\text{for any } r_k\mathrm{e}^{\mathrm{i}\theta_l} \in \overline{D}\backslash\overline{\Omega},\end{aligned} \tag{2.26}$$

where we have used

$$h_{ij} = h\big(r_i \mathrm{e}^{\mathrm{i}\theta_j}\big) \quad \text{and} \quad f_{ij} = f\big(r_i \mathrm{e}^{\mathrm{i}\theta_j}\big).$$

Taking into account the particular form of $\phi_{ij}(r\mathrm{e}^{\mathrm{i}\theta})$ as a nonhomogeneous term of equation (2.2), we can apply directly (2.4) to find $\tilde{u}(\phi_{ij})$. First, we see that in equations (2.4) the Fourier coefficients $u_n(r)$ of the solution $\tilde{u}(r\mathrm{e}^{\mathrm{i}\theta})$ are some linear functions of the Fourier coefficients $f_n(r)$ of the nonhomogeneous term $\tilde{f}(r\mathrm{e}^{\mathrm{i}\theta})$. In order to specify this linear dependence, we write $(u_n(f_n))(r)$ for $u_n(r)$ which is useful for our purposes below. The Fourier coefficients of the functions $\phi_{ij}(r\mathrm{e}^{\mathrm{i}\theta})$ can be written as

$$\phi_{ij,n}(r) = \frac{1}{N}\varphi_i(r)\mathrm{e}^{-\mathrm{i}n\theta_j},$$

and therefore, the Fourier coefficients of $(\tilde{u}(\phi_{ij}))(r\mathrm{e}^{\mathrm{i}\theta})$ are given by

$$\big(u_n(\phi_{ij})\big)\big(r\mathrm{e}^{\mathrm{i}\theta}\big) = \frac{1}{N}\big(u_n(\varphi_i)\big)(r)\mathrm{e}^{-\mathrm{i}n\theta_j}. \tag{2.27}$$

Therefore, we can write

$$\big(\tilde{u}(\phi_{ij})\big)\big(r\mathrm{e}^{\mathrm{i}\theta}\big) = \frac{1}{N}\sum_{n=-N/2}^{N/2-1}\big(u_n(\varphi_i)\big)(r)\mathrm{e}^{\mathrm{i}n(\theta-\theta_j)}. \tag{2.28}$$

The expressions for $(u_n(\varphi_i))(r)$ can be found using (2.7) and the integrals $(I_n(\varphi_i))(r)$; here we have also indicated the explicit dependence of integral $I_n(r)$ on $\varphi_i(r)$. Taking into account the form (2.12) of the functions $\varphi_i(r)$ we get in this case

$$\big(I_n(\varphi_i)\big)(r) = \begin{cases} \displaystyle\int_{r_{i-1}}^{r_{i+1}} \rho u_{1n}(r)u_{2n}(\rho)\varphi_i(\rho)\,\mathrm{d}\rho & \text{if } r_{i+1} \leqslant r, \\ \displaystyle\int_{r_{i-1}}^{r} \rho u_{1n}(r)u_{2n}(\rho)\varphi_i(\rho)\,\mathrm{d}\rho & \\ \quad \displaystyle + \int_{r}^{r_{i+1}} \rho u_{2n}(r)u_{1n}(\rho)\varphi_i(\rho)\,\mathrm{d}\rho & \text{if } r_{i-1} \leqslant r \leqslant r_{i+1}, \\ \displaystyle\int_{r_{i-1}}^{r_{i+1}} \rho u_{2n}(r)u_{1n}(\rho)\varphi_i(\rho)\,\mathrm{d}\rho & \text{if } r \leqslant r_{i-1}. \end{cases} \tag{2.29}$$

To conclude this section, we summarize the steps involved in the above described embedding method using the optimal distributed control as follows.

1. In order to numerically evaluate the curvilinear integrals in (2.26), we calculate the values of the function $(\tilde{u}(\phi_{ij}))(r\mathrm{e}^{\mathrm{i}\theta})$ (given by (2.28)) at some points $r\mathrm{e}^{\mathrm{i}\theta} \in \partial\Omega$ for each mesh point $r_i\mathrm{e}^{\mathrm{i}\theta_j}$.

2. Using the above calculated values, we evaluate the matrix and the right-hand side of the algebraic linear system (2.26) by numerical integration. By solving this linear system, we find the values $h_{ij} = h(r_i\mathrm{e}^{\mathrm{i}\theta_j}) = \tilde{f}(r_i\mathrm{e}^{\mathrm{i}\theta_j})$ at the points $r_i\mathrm{e}^{\mathrm{i}\theta_j} \in \overline{D}\backslash\overline{\Omega}$.

3. Using the inverse fast Fourier transform we calculate the Fourier coefficients $f_n(r_i)$ of $\tilde{f}(r_i e^{i\theta})$, for all r_i, $i = 1, \dots, M$.

4. Using algorithm 1.1, we find the Fourier coefficients $u_n(r_i)$ of $\tilde{u}(r_i e^{i\theta})$, for all r_i, $i = 1, \dots, M$.

5. Finally, we determine the values $\tilde{u}(r_i e^{i\theta_j})$ of the solution $\tilde{u}$ at the mesh points $r_i e^{i\theta_j}$, using the fast Fourier transform to calculate the values of $\tilde{u}(r_i e^{i\theta})$, $i = 1, \dots, M$, at $\theta_1, \dots, \theta_N$.

3. Numerical examples

Our numerical experiments refer to the solution of the Dirichlet problem (2.1). We have tested the embedding method given in the previous section for the problem which has the exact solution

$$u(x, y) = \frac{1}{4}\bigl[1 + \tanh(\alpha x)\bigr]\bigl[1 + \tanh(\alpha y)\bigr] \tag{3.1}$$

with $\alpha = 7.22$. In the numerical examples in [4,16], this same function was the solution of a problem associated with the equation $\Delta u - \sigma u = f$, and is a very good example of a function with large variation in its gradient. The Dirichlet problem (2.1) with this as a solution has the inhomogeneous term f given by

$$\begin{aligned} f(x, y) = &-\left[\frac{2\alpha}{\exp(\alpha x) + \exp(-\alpha x)}\right]^2 \tanh(\alpha x)\bigl[1 + \tanh(\alpha y)\bigr] \\ &- \left[\frac{2\alpha}{\exp(\alpha y) + \exp(-\alpha y)}\right]^2 \tanh(\alpha y)\bigl[1 + \tanh(\alpha x)\bigr]. \end{aligned} \tag{3.2}$$

The disc D, the domain of the problem (2.2), has its center at the origin of the coordinate system. Its radius has been chosen as 0.98, and it is the radius of the circumscribed circle to the square, with its sides equal to 1.386, which has been used as an embedding domain in [4,16]. We have considered two types of domains for problem (2.1): a hexagonal domain as a simply connected domain, and a hexagonal domain with a circular hole as a doubly connected domain (see figure 1). In both cases, the hexagon has its center at the origin and the radius of its circumscribed circle is 0.577. The circular hole, for the second type of domain, has its radius equal to 0.075 and its center at $(0.0, -0.25)$.

The functions $f(x, y)$ and $u(x, y)$, plotted over the entire disc D, are shown in figure 2. In [16], the authors have suggested that, in order to reduce the computing time, the extension of the nonhomogeneous term f should be sought not in the whole $D \backslash \overline{\Omega}$, but only in a strip in $D \backslash \overline{\Omega}$ surrounding Ω. The values of $\tilde{f}$ outside this strip are kept fixed. In our numerical experiments, we have taken two extensions of f outside the strip: with zero values and with values calculated by formula (3.2). In our numerical tests, the errors in the computed solutions were very similar for the two cases of extension of f. Also, our numerical tests have shown that the error in the computed solution decreases

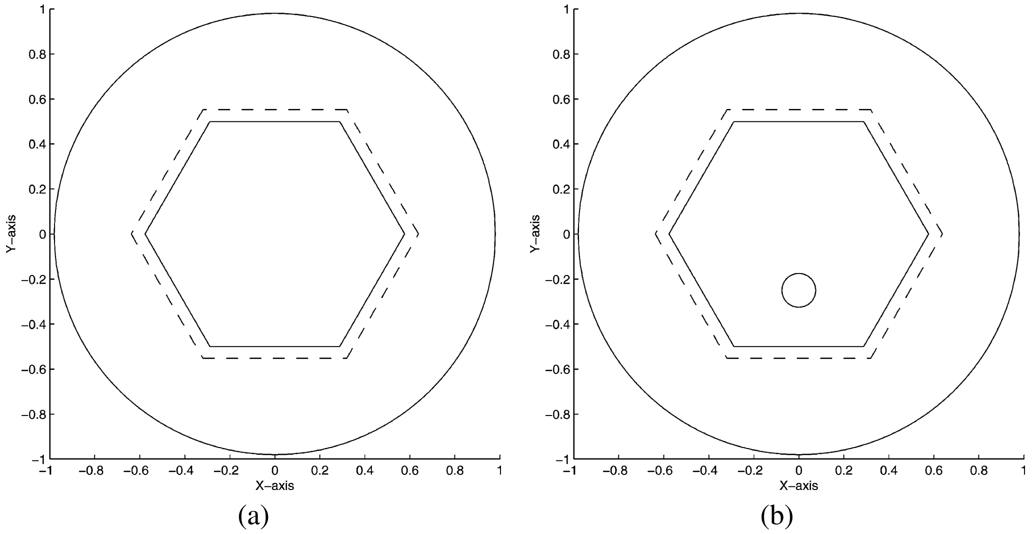


Figure 1. Domains: (a) hexagonal, (b) hexagonal with a circular hole.

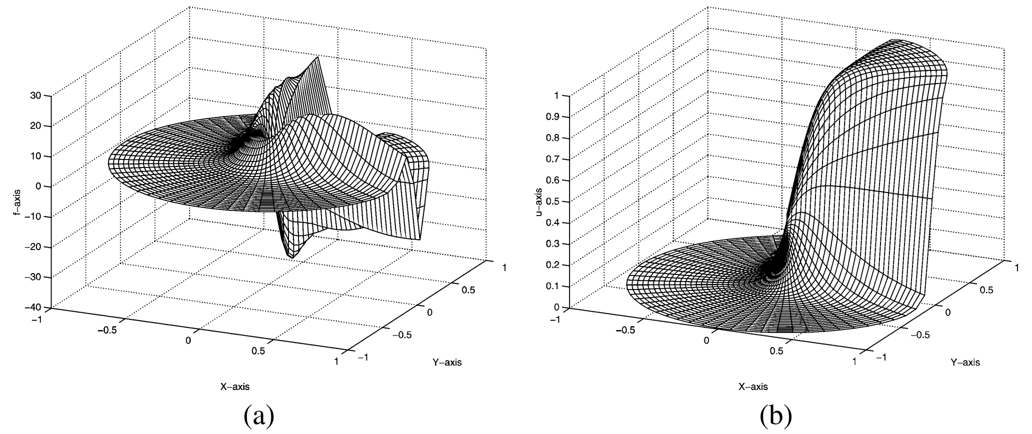


Figure 2. Functions: (a) $f(x, y)$, (b) $u(x, y)$.

with increase in the size of the strip initially, but after a certain large enough strip size this error does not decrease any more and in some instances starts increasing with further increase in the strip size. This comes from the fact that the number of basis functions ϕ_{ij} given in (2.25) must be less than the number of points on $\partial\Omega$ at which we evaluate the functions $\tilde{u}(\phi_{ij})$ in (2.27); otherwise the algebraic system (2.26) will be singular, the functions $\tilde{u}(\phi_{ij})$ being approximated on $\partial\Omega$ by piecewise linear functions. On the other hand, a small number of functions ϕ_{ij} in comparison with the mesh nodes in Ω will give a poor approximation of the solution $\tilde{u}$. Consequently, the error in the computed solution is sensitive to the width of the strip and a proper choice of this is important in obtaining good results.

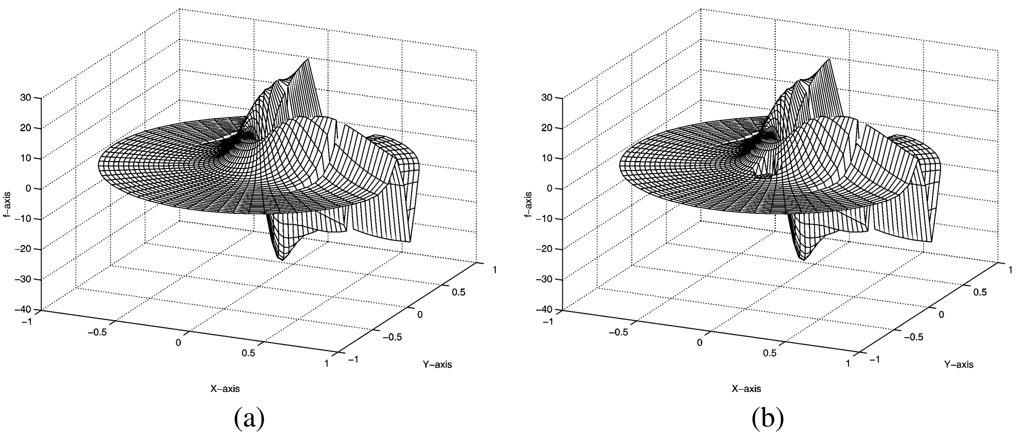


Figure 3. Initial function $f(x, y)$: (a) hexagonal domain, (b) hexagonal with a circular hole domain.

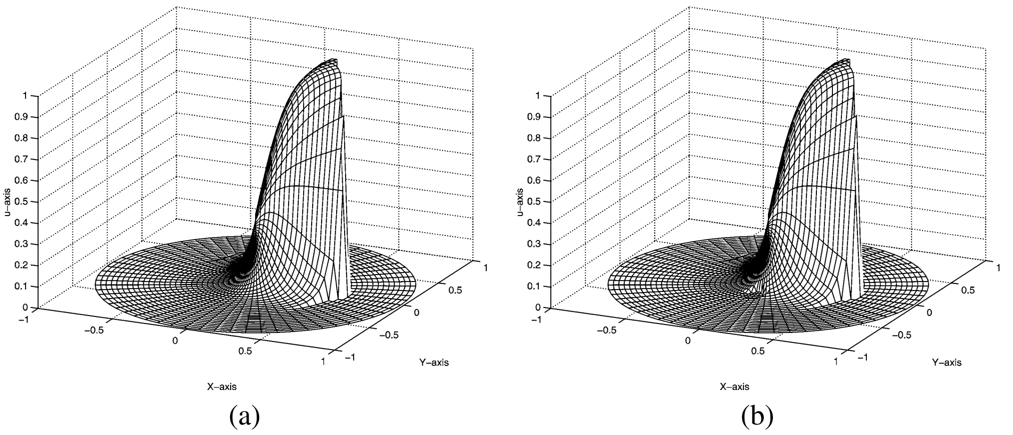
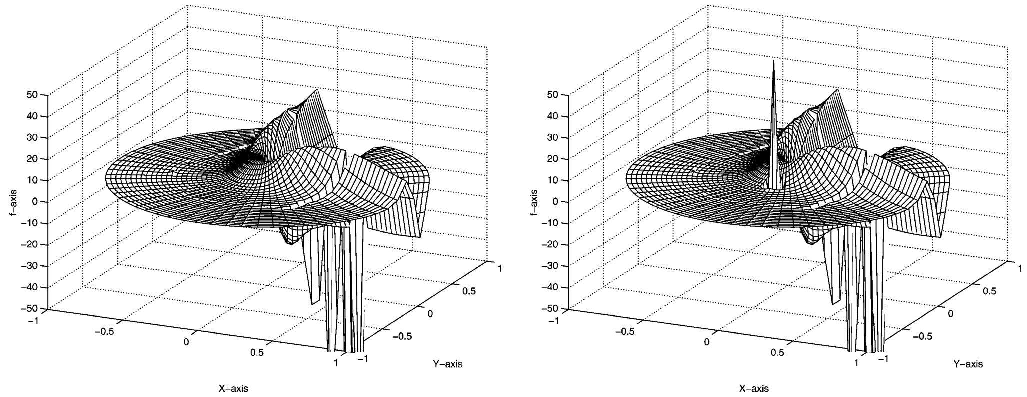


Figure 4. Exact solution $u(x, y)$: (a) hexagonal domain, (b) hexagonal with a circular hole domain.

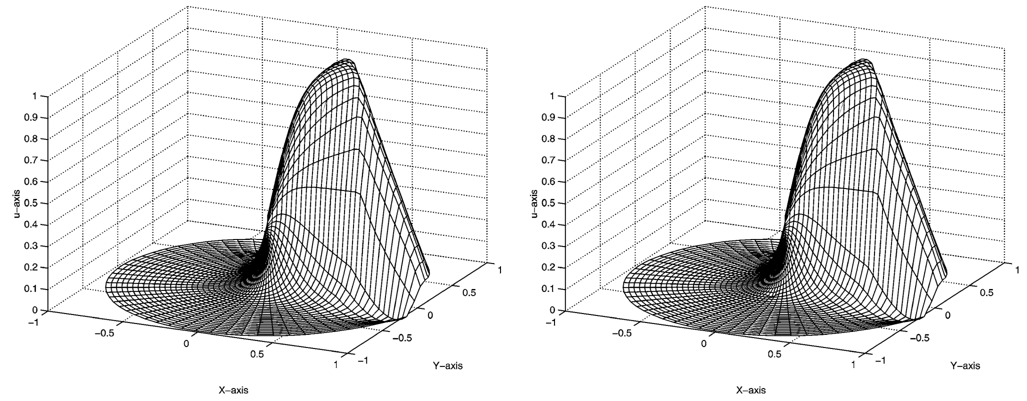
The strip we have taken in our tests is bounded by the boundaries of the hexagonal domain Ω and another hexagon with its center also at the origin of the Cartesian coordinate system (plotted with dashed line in figure 1). As we will see in the following, the width of this strip will be chosen depending on the total number of mesh nodes in D, i.e. the finer is the mesh on D the more nodes the strip contains. The initial function $\tilde{f}$ obtained by the extension of f with formula (3.2) outside the strip (i.e. in the domain bounded by the dashed boundary of the strip and the circle, in figure 1) is shown for each of the two domains in figure 3. This function has zero values in the strip in the case of the simply connected domain (figure 1(a)), and it has zero values in the strip and in the hole in the case of the doubly connected domain (figure 1(b)). Also, the solution $u(x, y)$ given in equation (3.1) is shown for each of the two domains Ω in figure 4.

In figure 5 we have plotted the optimal nonhomogeneous term $\tilde{f}$ which we have obtained for problem (2.2) starting from the above initial extension of f outside the



(a) (b)

Figure 5. Optimal $\tilde{f}(x, y)$: (a) hexagonal domain, (b) hexagonal with a circular hole domain.



(a) (b)

Figure 6. Computed $\tilde{u}(x, y)$: (a) hexagonal domain, (b) hexagonal with a circular hole domain.

strip. The values of the extended function $\tilde{f}$ in the disc with a circular hole, in the case of the doubly connected domain, have been computed in such a way that they satisfy the optimality condition (2.22), like those in the strip. In figure 6 we have plotted the solution $\tilde{u}$ of problem (2.2) corresponding to the optimal distributed control $\tilde{f}$. We see in figure 5 that the nonhomogeneous term $\tilde{f}$ has large oscillations within the strip. Naturally, these oscillations affect the solution $\tilde{u}$ outside the domain Ω, but this influence is very small, as seen in figure 6, in these computations where we have used 32 segments in radial direction, 64 terms in the Fourier expansions, and a strip size of one δ_r where δ_r is the discretization step size on the radius. As we shall see in the tables below the following: the solution $\tilde{u}$ of the problem (2.2) approximates very well the boundary condition g and the solution u of problem (2.1), even if these oscillations are not small. As we have already mentioned, the errors between the exact solution and the computed

Table 1
Errors inside the hexagonal domain.

$\delta_r \backslash N$	8	16	32	64
0.98/8	0.224E+00	0.911E−01	0.122E−01	0.105E−01
0.98/16	0.126E+00	0.383E−01	0.532E−02	0.316E−02
0.98/32	0.103E+00	0.246E−01	0.439E−02	0.814E−03
0.98/64	0.105E+00	0.283E−01	0.129E−02	0.205E−03

Table 2
Errors on the boundary of the hexagonal domain.

$\delta_r \backslash N$	8	16	32	64
0.98/8	0.125E+00	0.284E−01	0.134E−02	0.390E−04
0.98/16	0.125E+00	0.283E−01	0.130E−02	0.124E−05
0.98/32	0.112E+00	0.288E−01	0.644E−02	0.144E−05
0.98/64	0.109E+00	0.261E−01	0.108E−02	0.115E−05

Table 3
Errors inside the hexagonal domain with a circular hole.

$\delta_r \backslash N$	8	16	32	64
0.98/8	0.224E+00	0.910E−01	0.122E−01	0.105E−01
0.98/16	0.126E+00	0.383E−01	0.533E−02	0.316E−02
0.98/32	0.103E+00	0.246E−01	0.439E−02	0.821E−03
0.98/64	0.105E+00	0.283E−01	0.295E−02	0.213E−03

one are very similar for the two extensions of f outside the strip. For this reason, we give here only the tables corresponding to the extension of f with zero.

Tables 1 and 2 show absolute errors between the exact solution given in (3.1) and that obtained as an optimal solution of problem (2.22) in the hexagonal domain for several choices of the mesh sizes. The maximum value of this error calculated over all the mesh nodes of the domain Ω is tabulated in table 1 for each set of mesh sizes. The number of terms in the Fourier series expansion we have used to calculate the solution is denoted by N. In radial direction we have considered a regular mesh and we have denoted the distance between two consecutive mesh points by δ_r. The curvilinear integrals giving the matrix and the right hand side of the algebraic linear system (2.26) have been calculated by the trapezoidal rule using a regular discretization of the boundary $\partial\Omega$. The tests were made with 540 points on $\partial\Omega$. The strip has width $3\delta_r$ for the tests carried out with either $N = 64$ or $\delta_r = 0.98/64$, and $1\delta_r$ for the other ones. The maximum over all the points on the hexagonal boundary of the absolute errors for several values of N and δ_r is tabulated in table 2.

Tables 3 and 4 have been obtained for the case of the hexagonal domain with a circular hole. The tests were made with 540 and 72 nodes on the hexagonal and circular boundary, respectively.

Table 4
Errors on the boundary of the hexagonal domain with a circular hole.

$\delta_r \backslash N$	8	16	32	64
0.98/8	0.125E+00	0.284E−01	0.226E−02	0.223E−02
0.98/16	0.125E+00	0.283E−01	0.130E−02	0.497E−04
0.98/32	0.112E+00	0.288E−01	0.644E−02	0.144E−05
0.98/64	0.109E+00	0.261E−01	0.108E−02	0.127E−05

The tabulated data shows that the errors decrease monotonically as the number of nodes, N, in the circumferential direction increases for each radial mesh size δ_r fixed. It also shows that with N fixed, the errors also decrease monotonically as the mesh size δ_r decreases for most values of N, but for some values of fixed N, the errors decrease but not monotonically as the mesh size δ_r decreases. There are few instances of increase in the errors as the mesh size δ_r decreases with N fixed, but such increase is local in the sense that with further increase, as noticed in some of the tables, the errors start eventually decreasing. Thus, the method converges with respect to both N and δ_r, even if the error is not strictly decreasing as δ_r goes to zero for fixed N. A reasonable explanation for this local aberration in convergence is that the error is very sensitive to the width of the strip whose size which depends on δ_r also decreases with decreasing values of δ_r.

An appropriate data to study for the purpose of comparing convergence of this method with that in [4] is the main diagonal entries of each of the tables which show that the errors decrease as both mesh sizes goes to zero with their ratio fixed. The similar convergence study was done in [4] where the embedding domain was rectangular and the numerical tests were made for mesh 10×10, $20 \times 20, \ldots, 100 \times 100$ in order to retain the computational efficiency of the method. Thus, we see that both the methods converge: one in this paper and the one in [4].

Furthermore, the tabulated data shows that the error inside the domain, in most instances, is relatively more sensitive to the radial grid size δ_r than the error on the boundary. The bottom few entries in the last column of tables 2 and 4 show that the boundary data can be approximately satisfied for relatively modest number of grids (see, for example, the error corresponding to 64×64 mesh size, i.e. the last entry of the last column in these tables), and the corresponding numerical solution is a reasonably accurate one up to few decimal places. In fact, the tabulated data shows that the solution in the domain is accurate up to three decimal digits when the boundary condition is satisfied accurately up to five decimal digits.

The tabulated data also shows the extent of sensitivity of the errors to the simply and doubly connected domains. In fact, comparison of the errors obtained in the two types of domains shows that these are approximately of the same order for the same grid size. This observation perhaps can be taken as numerical evidence of the fact that the convergence of the method perhaps does not depend on whether the domain is simply or doubly connected.

Finally, these tables show that the error is small for relatively few nodes of the mesh. Comparison of these errors with the errors obtained for the similar numerical example given in [4], where the method uses a square as an embedding domain, shows that the numerical solutions obtained here in the interior of the domains are more accurate by at least one significant digit. For example, tabulated data in tables 1 and 3 shows that the errors in the interior of the domains obtained on a 64 × 64 grid are about one tenth of the one in [4, table 3] obtained on a 80 × 80 grid. The errors on the boundary in tables 2 and 4 obtained on a 64 × 64 grid are better than two-tenths of the one in [4, table 1] obtained on a 80 × 80 grid. Thus the method is more accurate on this specific test case. As another test case consider, for example, the tabulated data in tables 1 and 3 which shows that the errors in the interior of the domains obtained on a 32 × 32 grid are about half of the one in [4, table 3] obtained on a 50 × 50 grid. However, the errors on the boundary in tables 2 and 4 obtained on a 32 × 32 grid are about six times larger than the one in [4, table 1] on a 50 × 50 grid. This is more of an exception than the rule. The errors obtained on 8 × 8 and 16 × 16 grids are of the same order as those in [4, table 1]. We can conclude from our test cases that this method is more accurate for solutions in the interior of domains and also for the solutions on the boundary except for grids with a relatively small number of points. A possible explanation would be that we have taken 720 interpolation points on the boundary of the hexagonal domain for the example in [4], and only 540 for the example in this paper.

It is also worth noting that the error in the numerical solution for the interior of the domains decreases with increase in grid size at a faster rate with this method than the one proposed in [4]. This is pretty easy to recognize by inspecting the data along the diagonal entries of tables 1 and 3, and similar data (second column under $\mathrm{err}_{\max}$) in [4, table 3]. Therefore, in general, we can conclude that the method presented in this paper is more accurate. An explanation, among others perhaps, of this observation is as follows. In the method presented in [4], the values of the basis functions $\tilde{u}(e_{ij})$ at the nodes on $\partial\Omega$ are calculated by interpolation using their numerically computed values at the nodes of two-dimensional mesh. In the method present here, these values are exactly calculated as discussed in the text which perhaps contributes to the improved accuracy of the present method.

A theoretical convergence rate of these methods would be of interest but this study falls outside the scope of this paper.

4. Conclusions

In this paper we have proposed an embedding method based on an optimal distributed control and a fast algorithm for solving elliptic problems in a disc. The method proposed here differs from the one proposed in [4] in the choice of the embedding domains: this method here uses discs as embedding domains where as the method in [4] uses rectangles as embedding domains. In both methods, however, the solution of the inhomogeneous problem defined in the embedding domain is required on the boundary of the interior arbitrary domain so that the boundary integrals involving these solutions (the integrals in (2.26), for the method described in this paper) can be evaluated. This calls for use of some fast method that can evaluate the solution on the boundary of the arbitrary domain efficiently and accurately. These values of the solution on the boundary nodes of the interior domain are obtained by interpolating the mesh nodal values of the solution on the rectangular domain in [4], but by using exact formula (2.28) in the method proposed here. This can be one of the reasons why the errors obtained in the present method using an embedding disc are better than those obtained in [4] with rectangular embedding domains. Moreover, the fast algorithm described in [3] (see also [7]) is ideally suited for the calculation of the Fourier coefficients in (2.4) (see section 2.1) and hence has been used here.

Using this method one can solve elliptic problems having complicated shaped domains which may be either simply or multiply connected. Since our fast algorithm (see [3,7]) can be applied for almost all types of second order elliptic nonhomogeneous equations in a disc [3], this embedding method may be used to solve a wider class of problems in arbitrary domains. We have found the optimal extension of the nonhomogeneous term f by solving a linear algebraic system, as may be seen in section 2. The construction of this linear system needs the evaluation of some algebraic expressions at the points of a one-dimensional mesh along the boundary of the complicated shaped domain. These algebraic expressions, being completely independent, can be simultaneously solved on parallel machines. Once the optimal extension of f is obtained, we apply over the entire disc the fast algorithm described in section 2 to calculate the solution u of the problem (2.1). As we have seen in section 3, our numerical results for both, simply and doubly connected, domains show a high accuracy of the proposed method.

Acknowledgement

We would like to thank two referees for their helpful comments.

References

[1] G.P. Astrakmantsev, Methods of fictitious domains for a second order elliptic equation with natural boundary conditions, U.S.S.R. Comput. Math. Math. Phys. 18 (1978) 114–121.

[2] C. Atamian, Q.V. Dinh, R. Glowinski, J. He and J. Périaux, Control approach to fictitious-domain methods. Application to fluid dynamics and electro-magnetics, in: *Fourth Internat. Symposium on Domain Decomposition Methods for Partial Differential Equations*, eds. R. Glowinski, Y. Kuznetsov, G. Meurant, J. Périaux and O.B. Widlund (SIAM, Philadelphia, PA, 1991) pp. 275–309.

[3] L. Badea and P. Daripa, A fast algorithm for two-dimensional elliptical problems, Numer. Algorithms 30 (2002) 199–239.

[4] L. Badea and P. Daripa, On a Fourier method of embedding domains using an optimal distributed control, Numer. Algorithms 32 (2003) 261–273.

[5] J. Blum, Identification et contrôle de l'équilibre du plasma dans un Tokamak, in: *Modelos Matematicos en Fisica de Plasmas*, eds. J.I. Diaz and A. Galindo, Memorias de la Real Academia de Ciencias, Serie de Ciencias Exactas, t. XXX (1995) pp. 23–48.

[6] C. Borgers, Domain embedding methods for the Stokes equations, Numer. Math. 57(5) (1990) 435–452.

[7] L. Borges and P. Daripa, A fast parallel algorithms for the Poisson equation on a disk, J. Comput. Phys. 169 (2001) 151–192.

[8] M. Briscolini and P. Santangelo, Development of the mask method for incompressible unsteady flows, J. Comput. Phys. 84 (1989) 57–75.

[9] B.L. Buzbee, F.W. Dorr, J.A. George and G.H. Golub, The direct solution of the discrete Poisson equation on irregular regions, SIAM J. Numer. Anal. 8 (1971) 722–736.

[10] P. Daripa, A fast algorithm to solve nonhomogeneous Cauchy–Riemann equations in the complex plane, SIAM J. Sci. Statist. Comput. 13(6) (1992) 1418–1432.

[11] P. Daripa, A fast algorithm to solve the Beltrami equation with applications to quasiconformal mappings, J. Comput. Phys. 106 (1993) 355–365.

[12] P. Daripa and D. Mashat, Singular integral transforms and fast numerical algorithms I, Numer. Algorithms 18 (1998) 133–157.

[13] J. Daňková and J. Haslinger, Numerical realization of a fictitious domain approach used in shape optimization I. Distributed controls, Appl. Math. 41(2) (1996) 123–147.

[14] E.J. Dean, Q.V. Dinh, R. Glowinski, J. He, T.W. Pan and J. Périaux, Least squares/domain imbedding methods for Neumann problems: Applications to fluid dynamics, in: *Fifth Internat. Symposium on Domain Decomposition Methods for Partial Differential Equations*, eds. D.E. Keyes, T.F. Chan, G. Meurant, J.S. Scroggs and R.G. Voigteds (SIAM, Philadelphia, PA, 1991) pp. 451–475.

[15] Q.V. Dinh, R. Glowinski, J. He, T.W. Pan and J. Périaux, Lagrange multiplier approach to fictitious domain methods: Applications to fluid dynamics and electro-magnetics, in: *Fifth Internat. Symposium on Domain Decomposition Methods for Partial Differential Equations*, eds. D.E. Keyes, T.F. Chan, G. Meurant, J.S. Scroggs and R.G. Voigteds (SIAM, Philadelphia, PA, 1991) pp. 151–194.

[16] M. Elghaoui and R. Pasquetti, A spectral embedding method applied to the advection–diffusion equation, J. Comput. Phys. 125 (1996) 464–476.

[17] M. Elghaoui and R. Pasquetti, Mixed spectral boundary element embedding algorithms for the Navier–Stokes equations in the vorticity-stream function formulation, J. Comput. Phys. 153 (1999) 82–100.

[18] S.A. Finogenov and Y.A. Kuznetsov, Two-stage fictitious component methods for solving the Dirichlet boundary value problem, Soviet J. Numer. Anal. Math. Modelling 3 (1988) 301–323.

[19] V. Girault, R. Glowinski and H. Lopez, Error analysis of a finite element realization of a fictitious domain/domain decomposition method for elliptic problems, East–West J. Numer. Math. 5(1) (1997) 35–56.

[20] R. Glowinski and Y. Kuznetsov, On the solution of the Dirichlet problem for linear elliptic operators by a distributed Lagrange multiplier method, C. R. Acad. Sci. Paris Sér. I Math. 327(7) (1998) 693–698.

[21] R. Glowinski, T.-W. Pan, T.I. Hesla, D.D. Joseph and J. Périaux, A fictitious domain method with distributed Lagrange multipliers for the numerical simulation of a particulate flow, in: *Domain Decomposition Methods*, Boulder, CO, 1997, Contemporary Mathematics, Vol. 218 (Amer. Math. Soc., Providence, RI, 1998) pp. 121–137.

[22] R. Glowinski, T.-W. Pan and J. Périaux, Lagrange multiplier/fictitious domain method for the Dirichlet problem generalization to some flow problems, Japan J. Indust. Appl. Math. 12(1) (1995) 87–108.

[23] R. Glowinski, T.-W. Pan and J. Périaux, Fictitious domain/Lagrange multiplier methods for partial differential equations, in: *Domain-Based Parallelism and Problem Decomposition Methods in Computational Science and Engineering* (SIAM, Philadelphia, PA, 1995) pp. 177–192.

[24] J. Haslinger, Fictitious domain approaches in shape optimization, in: *Computational Methods for Optimal Design and Control*, Arlington, VA, 1997, Progr. Systems Control Theory, Vol. 24 (Birkhäuser, Boston, MA, 1998) pp. 237–248.

[25] J. Haslinger and A. Klarbring, Fictitious domain/mixed finite element approach for a class of optimal shape design problems, RAIRO Modél. Math. Anal. Numér. 29(4) (1995) 435–450.

[26] J. Haslinger and R.A.E. Makinen, Shape optimization of materially nonlinear bodies in contact, Appl. Math. 42(3) (1997) 171–193.

[27] J.L. Lions, *Optimal Control of Systems Governed by Partial Differential Equations* (Springer, Berlin, 1971).

[28] G.I. Marchuk, Y.A. Kuznetsov and A.M. Matsokin, Fictitious domain and domain decomposition methods, Soviet J. Numer. Anal. Math. Modelling 1 (1986) 3–35.

[29] P. Neittaanmäki and D. Tiba, An embedding of domain approach in free boundary problems and optimal design, SIAM J. Control Optim. 33(5) (1995) 1587–1602.

[30] D.P. O'Leary and O. Widlund, Capacitance matrix methods for the Helmholtz equation on general three-dimensional regions, Math. Comp. 3 (1979) 849–879.

[31] W. Proskurowsky and O.B. Widlund, On the numerical solution of Helmholtz equation by the capacitance matrix method, Math. Comp. 30 (1979) 433–468.

[32] J. Shen, Efficient spectral-Galerkin methods III: Polar and cylindrical geometries, SIAM J. Sci. Comput. 18 (1997) 1583–1604.

[33] D.P. Young, R.G. Melvin, M.B. Bieterman, F.T. Johnson, S.S. Samanth and J.E. Bussolety, A locally refined finite rectangular grid finite element method. Application to computational physics, J. Comput. Phys. 92 (1991) 1–66.