Comput Optim Appl
DOI 10.1007/s10589-007-9076-2

A new fictitious domain method in shape optimization

Karsten Eppler · Helmut Harbrecht · Mario S. Mommer

Received: 30 March 2006 / Revised: 2 May 2007

Abstract The present paper is concerned with investigating the capability of the smoothness preserving fictitious domain method from Mommer (IMA J. Numer. Anal. 26:503–524, 2006) to shape optimization problems. We consider the problem of maximizing the Dirichlet energy functional in the class of all simply connected domains with fixed volume, where the state equation involves an elliptic second order differential operator with non-constant coefficients. Numerical experiments in two dimensions validate that we arrive at a fast and robust algorithm for the solution of the considered class of problems. The proposed method can be applied to three dimensional shape optimization problems.

Keywords Shape optimization · Shape calculus · Fictitious domain method · Multiscale method

1 Introduction

In several papers (see [7, 10] for example), two of the authors developed efficient algorithms for the solution of several elliptic shape optimization problems. A boundary

K. Eppler
Institut für Numerische Mathematik, Technische Universität Dresden, Zellescher Weg 12–14, 01062 Dresden, Germany
e-mail: karsten.eppler@tu-dresden.de

H. Harbrecht (✉)
Institut für Numerische Mathematik, Universität Bonn, Wegelerstr. 6, 53115 Bonn, Germany
e-mail: harbrecht@ins.uni-bonn.de

M.S. Mommer
Interdisciplinary Center for Scientific Computing, University of Heidelberg, Im Neuenheimer Feld 368, 69120 Heidelberg, Germany
e-mail: mario.mommer@iwr.uni-heidelberg.de

variational approach was proposed in combination with boundary integral representations of the shape gradient and the shape Hessian. The considered class of model problems allowed the use of boundary integral equations to compute all ingredients of the functional, the gradient, and the Hessian, that arise from the state equation. In combination with a fast wavelet Galerkin method to solve the boundary integral equations, we obtained very efficient first and second order algorithms for shape problems in two and three spatial dimensions. In particular, the use of boundary element methods requires only a discretization of the free boundary. In our opinion this is very advantageous since, on the one hand, modern boundary integral methods reduce the complexity, and on the other hand, large deformations of the domains are realizable without remeshing. Moreover, exterior boundary value problems are treatable, as are the computation of free surfaces of liquid bubbles, or drops levitating in an electromagnetic field, cf. [8, 9, 25, 27]. Note that boundary element methods for shape optimization have been used also by other authors, see e.g. [25, 27, 29].

However, to be able to realize the optimal efficiency inherent in these advantages, it is of great help if the constraints and shape derivatives can be formulated in terms of *boundary integrals*. Consequently, an obvious constraint for the objective function is that such a formulation should be possible, see [7] for details.

In case of compactly supported cost functionals one can overcome this restriction by coupling finite and boundary elements (see [11]). Thus, the advantages of both methods are retained, namely fast access to values on the compact subset by finite elements on a fixed triangulation, and the simple treatment of the free boundary by boundary elements. Nevertheless, the restriction to state equations involving differential operators with *constant* coefficients remains.

The above mentioned techniques are not applicable to state equations involving elliptic differential operators with *non-constant* coefficients. Fictitious domain methods offer obviously a convenient tool to deal with such shape optimization problems while the complicated remeshing, required for finite element methods, is still avoided, see for example Haslinger et al. [14, 15], Kunisch and Peichl [20], Neitaanmäki and Tiba [24], and Slawig [31, 32].

Up to now, the success of fictitious domain methods was limited since traditional methods suffer from low orders of convergence. For instance, the fictitious domain-Lagrange multiplier approach converges only as $\mathcal{O}(h^{1/2})$ in the energy norm when approximating from uniform grids with mesh size h (see [17]). Even the rate of convergence of standard (i.e. based on isotropic refinements) adaptive methods is limited by $\mathcal{O}(N^{-1/2})$ and $\mathcal{O}(N^{-1/4})$ in two and three dimensions, respectively, when spending N degrees of freedom, independently of the order of the approximation spaces (see [22] for a detailed discussion).

These difficulties arise from non-smooth extensions of the solutions outside the intrinsic domain. In particular, the approximation near the boundary is rather poor. Unfortunately, accuracy near the boundary is important for the accurate computation of the shape derivatives. In [21, 22], one of the authors proposed a rather novel and promising *smoothness preserving* fictitious domain method. It is called smoothness preserving since the solution of the reformulated problem on the fictitious domain is as smooth as the solution of the problem on the intrinsic domain. Thus the method produces high orders of convergence. The present paper is devoted to demonstrate the

capability of this method when used in the context of shape optimization problems. Numerical results confirm that the shape functional and gradient are approximated with high accuracy.

We consider the problem of maximizing the Dirichlet energy functional in the class of simply connected domains of class C^2, where the state solves a standard elliptic boundary value problem of second order. To ensure non-degeneration the sought domain is supposed to have a given volume. For the sake of clearness in representation, we restrict ourselves to the two dimensional setting. However, it is important to remark that the extension of the present algorithms to three spatial dimensions is straightforward.

The paper is organized as follows. Section 2 is dedicated to the shape optimization problem. We introduce our model problem of maximizing the Dirichlet energy functional under a volume constraint. After deriving the shape derivatives, we consider a standard augmented Lagrangian algorithm to treat the volume constraint. The minimization problems in the inner loop are solved by a nonlinear Ritz-Galerkin method for the necessary condition. A vector valued boundary perturbation ansatz is employed in order to describe the boundary and its update. On the one hand, any domain of genus zero can be represented, on the other hand, the boundary representation is non-unique. Since therefore the surface mesh might degenerate, we add a regularization term to the objective. In Sect. 3 we present the numerical scheme to compute the state function. We introduce the smoothness preserving fictitious domain method and discuss the evaluation of domain integrals by numerical quadrature. In the last section (Sect. 4) we present numerical results to demonstrate the capability of our approach.

2 Shape optimization

2.1 The model problem

Let $\Omega \subset \mathbb{R}^2$ be a domain with boundary $\Gamma := \partial\Omega$. We consider the Dirichlet energy functional

$$J(\Omega) = \int_\Omega \langle \mathbf{A}\nabla u, \nabla u\rangle d\mathbf{x} = \int_\Omega f u \, d\mathbf{x}, \tag{2.1}$$

where the state function u solves the boundary value problem

$$\begin{aligned} -\mathrm{div}(\mathbf{A}\nabla u) &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \Gamma = \partial\Omega. \end{aligned} \tag{2.2}$$

Herein, we assume that the inhomogeneity $f : \mathbb{D} \to \mathbb{R}$ and the symmetric and strictly definite positive matrix $\mathbf{A}(\mathbf{x}) = [a_{ij}(\mathbf{x})]_{i,j=1}^2$ are sufficiently regular and defined in a sufficiently large *hold all* $\mathbb{D} \subset \mathbb{R}^2$.

The goal of the present paper is to maximize the Dirichlet energy (2.1) over the class Υ of admissible domains. We assign Υ to be the set of all simply connected

domains of the class C^2. To avoid degeneration we shall impose an equality constraint on the volume of the domain

$$V(\Omega) := \int_\Omega d\mathbf{x} \stackrel{!}{=} V_0, \tag{2.3}$$

cf. [2]. Consequently, we arrive at the following problem:

$$-J(\Omega) \to \min_{\Omega \in \Upsilon} \quad \text{subject to} \quad V(\Omega) = V_0. \tag{2.4}$$

In this paper we do not treat the question of existence and regularity of solutions. Instead, we will tacitly assume the existence of optimal domains, being *sufficiently regular* to apply a first order shape calculus.

2.2 Shape calculus

We briefly recall well known facts about the first order shape calculus, useful for the discussion of the necessary condition and the numerical algorithms. For a general overview on shape calculus, mainly based on the perturbation of identity (Murat and Simon) or the speed method (Sokolowski and Zolesio), we refer the reader for example to [5, 23, 28, 30, 33], and the references therein.

Let $\mathbf{n}$ denote the outer unit normal to the boundary Γ and consider a C^2-smooth boundary perturbation field $\mathbf{U} : \Gamma \to \mathbb{R}^2$. Then, the shape gradient to the functional (2.1) reads as

$$\nabla J(\Omega)[\mathbf{U}] = \int_\Gamma \langle \mathbf{U}, \mathbf{n} \rangle \langle \mathbf{A} \nabla u, \nabla u \rangle d\sigma, \tag{2.5}$$

since the local shape derivative $du = du[\mathbf{U}]$ satisfies

$$\begin{aligned} &\operatorname{div}(\mathbf{A} \nabla du) = 0 \quad \text{in } \Omega, \\ &du = -\langle \mathbf{U}, \mathbf{n} \rangle \frac{\partial u}{\partial \mathbf{n}} \quad \text{on } \Gamma. \end{aligned}$$

The gradient of the volume reads as

$$\nabla V(\Omega)[\mathbf{U}] = \int_\Gamma \langle \mathbf{U}, \mathbf{n} \rangle d\sigma. \tag{2.6}$$

2.3 Relaxation of the constraints

The minimization problem (2.4) implies to find the solution $(\Omega^\star, \lambda^\star) \in \Upsilon \times \mathbb{R}$ of the saddle point problem

$$(\Omega^\star, \lambda^\star) = \arg \inf_{\Omega \in \Upsilon} \sup_{\lambda \in \mathbb{R}} L_\alpha(\Omega, \lambda),$$

where $L_\alpha(\Omega, \lambda)$ denotes the augmented Lagrangian functional

$$L_\alpha(\Omega, \lambda) = -J(\Omega) + \lambda \big(V(\Omega) - V_0\big) + \frac{\alpha}{2} \big(V(\Omega) - V_0\big)^2. \tag{2.7}$$

Of course, the choice $\alpha = 0$ yields the pure Lagrangian while $\lambda = 0$ and $\alpha \to \infty$ is known as standard quadratic penalty method. However, both choices have some drawbacks from the numerical point of view, cf. [6, 18], for example.

In order to avoid these difficulties, we choose $\alpha > 0$ and consider the following standard augmented Lagrangian algorithm:

- initialization: choose initial guesses $\lambda^{(0)}$ for $\lambda^\star$ and $\Omega^{(0)}$ for $\Omega^\star$,
- inner iteration: solve

$$\Omega^{(n+1)} := \operatorname{argmin} L_\alpha(\Omega, \lambda^{(n)}) \tag{2.8}$$

  with initial guess $\Omega^{(n)}$,
- outer iteration: update

$$\lambda^{(n+1)} := \lambda^{(n)} - \alpha\big(V(\Omega^{(n+1)}) - V_0\big).$$

It is well known that this algorithm converges to $(\Omega^\star, \lambda^\star)$ provided that the initial guesses and α are appropriately chosen [6, 18]. Notice that the nonlinear problem (2.8) has to be solved iteratively (see the next section), requiring in each iteration step the numerical solution of the state equation.

Remark 2.1 The necessary condition to (2.7) is equivalent to the identity

$$\langle \mathbf{A}\nabla u, \nabla u\rangle \equiv \lambda^\star \quad \text{on } \Gamma^\star.$$

2.4 Ritz-Galerkin approximation of the shape problem

The boundary of a domain $\Omega \in \Upsilon$ can be parameterized by a bijective positive oriented curve

$$\boldsymbol{\gamma} : [0,1] \to \Gamma, \quad \boldsymbol{\gamma}(\phi) = \begin{bmatrix} \gamma_x(\phi) \\ \gamma_y(\phi) \end{bmatrix}, \tag{2.9}$$

such that

$$\gamma_x, \gamma_y \in \mathrm{C}^2_{\mathrm{per}}([0,1]) := \big\{ f \in C^2([0,1]) : f^{(i)}(0) = f^{(i)}(1),\ i = 0,1,2 \big\}.$$

Setting

$$\begin{aligned} &\varphi^\Gamma_{-N} := \sin(2\pi N\phi),\ \varphi^\Gamma_{1-N} := \sin\big(2\pi(N-1)\phi\big), \ldots, \varphi^\Gamma_{-1} := \sin(2\pi\phi), \\ &\varphi^\Gamma_0 := 1,\ \varphi^\Gamma_1 := \cos(2\pi\phi), \ldots, \varphi^\Gamma_N := \cos(2\pi N\phi), \end{aligned} \tag{2.10}$$

we define the space

$$V^\Gamma_N = \operatorname{span}\{\varphi^\Gamma_{-N}, \varphi^\Gamma_{1-N}, \ldots, \varphi^\Gamma_N\} \subset \mathrm{C}^2_{\mathrm{per}}([0,1]) \tag{2.11}$$

of all trigonometric polynomials of degree $\le 2N$. To discretize the shape optimization problem we make the ansatz

$$\boldsymbol{\gamma}_N = \sum_{k=-N}^{N} \begin{bmatrix} a_k \\ b_k \end{bmatrix} \varphi^\Gamma_k \in V^\Gamma_N \times V^\Gamma_N \tag{2.12}$$

with coefficient vectors $[a_k, b_k]^T \in \mathbb{R}^2$. Identifying the approximate domain Ω_N with this boundary curve, problem (2.8) becomes finite dimensional

$$\Omega_N^\star := \operatorname{argmin} L_\alpha(\Omega_N, \lambda^{(n)}).$$

This discrete problem leads to a nonlinear Ritz-Galerkin scheme for the necessary condition:

$$\text{seek} \quad \boldsymbol{\gamma}_N^\star \in V_N^\Gamma \times V_N^\Gamma \quad \text{such that} \quad \nabla L_\alpha(\Omega_N^\star, \lambda^\star)[\mathbf{v}_N] = 0 \quad \text{for all } \mathbf{v}_N \in V_N^\Gamma \times V_N^\Gamma.$$

For the numerical solution of this nonlinear variational equation we apply the quasi-Newton method updated by the inverse BFGS-rule without damping. A second order approximation is proposed for performing the line search update if a descent fails. For all the details we refer to [6, 12, 13, 18] and the references therein.

Remark 2.2 In the three dimensional case one considers the unit sphere $\mathbb{S}$ as parameter space and the ansatz spaces V_N^Γ consisting of spherical harmonics of order $\leq N$. Then, $\boldsymbol{\gamma}_N : \mathbb{S} \to \Gamma$ is defined according to

$$\boldsymbol{\gamma}_N = \sum_k \mathbf{a}_k \varphi_k^\Gamma \in V_N^\Gamma \times V_N^\Gamma \times V_N^\Gamma$$

with coefficients $\mathbf{a}_k \in \mathbb{R}^3$. This ansatz has been used in e.g. [19].

2.5 Regularization

On the one hand the ansatz (2.12) does not impose any restriction to the topology of the domain except for its genus. On the other hand, the parametric representation (2.9) of the domain Ω is not unique. In fact, if $\Xi : [0, 1] \to [0, 1]$ denotes any smooth 1-periodic bijective mapping, then the boundary curve $\boldsymbol{\gamma} \circ \Xi$ describes another parametrization of Ω.

For our purpose some parameterizations are preferable to others. Therefore, we introduce an penalty term for finding an appropriate parametrization of the free boundary. In order to discretize functions on the free boundary we will use (2.9) to map a subdivision of the parameter space $[0, 1]$ to the boundary. From this point of view it is quite obvious that, for numerical computations, a "nice" parametrization distributes equidistant grid points of $[0, 1]$ equidistantly on Γ. This is equivalent to the claim that the parametrization should be close to that with respect to the arc length, that is $\|\boldsymbol{\gamma}'\| \approx |\Gamma|$ on the parameter space $[0, 1]$. To realize this claim we introduce the *mesh functional*

$$M(\Omega) = \int_0^1 \left(\langle \boldsymbol{\gamma}', \boldsymbol{\gamma}' \rangle - |\Gamma|^2\right)^2 d\phi, \tag{2.13}$$

and solve for small $\beta > 0$ the regularized shape problem

$$-J(\Omega) + \beta M(\Omega) \to \min_{\Omega \in \Upsilon} \quad \text{subject to} \quad V(\Omega) = V_0 \tag{2.14}$$

instead of the original problem (2.4). We mention that the best numerical results are achieved when $\beta \to 0$ during the optimization procedure.

Remark 2.3 In three dimensions a "nice" parametrization maps a regular and uniform mesh of the reference manifold $\mathbb{S}$ onto a regular and uniform mesh on Γ. This is the case if orthonormal tangents $\{\mathbf{t}_i\}_{i=1,2}$ of the reference manifold $\mathbb{S}$ are mapped to orthogonal tangents of the boundary Γ with length $\approx \sqrt{|\Gamma|/|\mathbb{S}|}$. This means that the first fundamental tensor of differential geometry, given by

$$\mathcal{T}(\widehat{\mathbf{x}}) = \left[\left\langle \frac{\partial \boldsymbol{\gamma}(\widehat{\mathbf{x}})}{\partial \mathbf{t}_i}, \frac{\partial \boldsymbol{\gamma}(\widehat{\mathbf{x}})}{\partial \mathbf{t}_j} \right\rangle\right]_{i,j=1}^{2},$$

satisfies $\mathcal{T} \approx |\Gamma|^2/|\mathbb{S}|^2 \cdot \mathbf{I}$. Therefore, we consider the mesh functional

$$M(\Omega) = \int_{\mathbb{S}} \left\| \mathcal{T}(\widehat{\mathbf{x}}) - \frac{|\Gamma|}{|\mathbb{S}|}\mathbf{I} \right\|_F^2 d\sigma_{\widehat{\mathbf{x}}},$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

3 Numerical method to compute the state

3.1 The SPFD method

To compute the state given by (2.2) we use the smoothness preserving fictitious domain (SPFD) method, introduced in [21]. The SPFD method is a fairly new domain embedding technique that has performed well in experimental settings before, and as will be seen in the numerical results, it can fulfill its promise in more applied contexts.

To solve a boundary value problem with any fictitious domain method, one embeds the *intrinsic domain* Ω into a larger *fictitious domain*, for example, a periodic square $\mathbb{T} = (\mathbb{R} \setminus \mathbb{Z})^2$. Note that $\mathbb{T}$ can also be seen as a torus. The next step is to construct from the original problem some auxiliary problem on the fictitious domain such that the solutions of this auxiliary and the original problem coincide on the intrinsic domain. The SPFD method earns its name from the fact that the solution of this auxiliary problem is smooth. It *preserves the smoothness* of the original problem.

We assume that the right hand side f is in $L^2(\mathbb{T})$, which is the space of periodic, square integrable functions on the unit square. Similarly, we will also write $H^2(\mathbb{T})$ for periodic, two times weakly differentiable function on the same square. For the sake of simplicity, we shall assume from now on that the *hold all* satisfies $\mathbb{D} = \mathbb{T}$. Then, since the boundary is C^2, the solution of the state equation will be in $H^2(\Omega)$. Consider for a moment the more general, non-homogeneous boundary condition $u = g$ on Γ, with $g \in H^{3/2}(\Gamma)$, and consider the least-squares functional on $H^2(\mathbb{T})$.

$$\Phi(u^+) = \|C(Au^+ - f)\|^2_{L^2(\mathbb{T})} + \|Bu^+ - g\|^2_{H^{3/2}(\Gamma)}, \tag{3.1}$$

where $A : H^2(\mathbb{T}) \to L^2(\mathbb{T})$ is the differential operator, $B : H^2(\mathbb{T}) \to H^{3/2}(\Gamma)$ is the trace operator, and $C : L^2(\mathbb{T}) \to L^2(\mathbb{T})$ is such that Cv is the extension by zero of the restriction to Ω of $v \in L^2(\mathbb{T})$.

It is reasonably easy to check that Φ has a minimum, which is not unique but can be chosen to depend continuously on the data $b := [f, g]^T \in \mathcal{H} := L^2(\mathbb{T}) \times H^{3/2}(\Gamma)$. Thus, the operator $F : H^2(\mathbb{T}) \to \mathcal{H}$ associated with Φ, given by the operator matrix

$$F = \begin{bmatrix} CA \\ B \end{bmatrix},$$

is bounded, and, while it has a large kernel, it still has a bounded pseudoinverse. Furthermore, every minimizer of Φ is an extension of the solution to the original problem (see [21]). Thus, to compute the state, we shall solve the least-squares problem

$$\text{find} \quad u^+ \in H^2(\mathbb{T}) \quad \text{such that} \quad \|Fu^+ - b\|_{\mathcal{H}} \to \min, \tag{3.2}$$

and take $u = u^+_{|\Omega}$.

There are a few approaches to tackle such a problem. One detail that needs to be addressed is that its solution is not unique, i.e., F has a large kernel. The standard approach is to apply a regularization method. Especially, this has been used for the penalty FD method, see [16]. However, a regularization approach would introduce a family of problems that become progressively ill-conditioned as the regularization parameter tends to zero. This seems wasteful as the operator F has a bounded pseudoinverse and any two solutions of (3.2) agree on Ω. Therefore, we will ignore the lack of unicity and discretize directly (3.2) by an appropriate Ritz-Galerkin scheme. Then, we will use an iterative least-squares solver which is able to solve underdetermined linear systems of equations.

3.2 Discretization and solution of the discrete problems

To approximate solutions of (3.2), we will use dyadic grids of mesh size $h_j := 2^{-j}$, with $j \geq 0$ an integer. We write

$$\mathbb{T} = \bigcup_{\mathbf{k}=(k_x,k_y)\in\mathbb{Z}_j} Q_{j,\mathbf{k}},$$

where $\mathbb{Z}_j := (\mathbb{Z}/2^j\mathbb{Z})^2$, and $Q_{j,\mathbf{k}} := 2^{-j}[k_x, k_x + 1) \times [k_y, k_y + 1)$.

When trying to discretize the operator F on the given mesh, one quickly realizes that the operator C can yield a potentially fatal problem for numerical stability. To overcome this problem, we approximate C by the operator C_j, defined as follows. Given $v \in L^2(\mathbb{T})$, $C_j v$ is defined as the extension by zero of the restriction of v to Ω_j, where

$$\Omega_j := \bigcup_{\mathbf{k}\in\mathbb{Z}_j} \{Q_{j,\mathbf{k}} : Q_{j,\mathbf{k}} \cap \Omega \neq \emptyset\}.$$

Notice that this approximation is not as crude as it looks. It has been shown in [21] that if $C(Au^+ - f) = 0$, and $Au^+ - f \in H^s(\mathbb{T})$ for $s > 0$ such that $s - 1/2$ is not an integer, then

$$\|C_j(Au^+ - f)\|_{L^2(\mathbb{T})} \lesssim h_j^s \|Au^+ - f\|_{H^s(\mathbb{T})}.$$

Since one can always find such an extension u^+ whenever $u \in H^{s+2}(\Omega)$, this proves that the minimum of the modified least-squares functional

$$\Phi_j(u^+) = \|C_j(Au^+ - f)\|^2_{L^2(\mathbb{T})} + \|Bu^+ - g\|^2_{H^{3/2}(\Gamma)} \tag{3.3}$$

converges rapidly towards the minimum of Φ.

Next, let us choose suitable approximation spaces. In $H^2(\mathbb{T})$ we will approximate from the spaces

$$V_j^{\mathbb{T}} = \text{span}\{\varphi^{\mathbb{T}}_{j,\mathbf{k}} : \mathbf{k} \in \mathbb{Z}_j\}$$

of periodic cardinal B-splines $\varphi^{\mathbb{T}}_{j,\mathbf{k}}$ of order $m > 2$ on the given grid. These are C^{m-2}-functions that are multi-polynomials of degree $m-1$ on each cube $Q_{j,\mathbf{k}}$. In $L^2(\mathbb{T})$ we will approximate using the spaces

$$V_j^0 = \text{span}\{\varphi^0_{j,\mathbf{k},\mathbf{l}} : \mathbf{k} \in \mathbb{Z}_j, \mathbf{l} \in \mathcal{I}\},$$

where $\mathcal{I} := \{\mathbf{l} = (l_x, l_y) : 0 \le l_x, l_y < n\}$, consisting of discontinuous piecewise multi-polynomials of order n. The orthonormal basis functions $\varphi^0_{j,\mathbf{k},\mathbf{l}}$ are supported on $Q_{j,\mathbf{k}}$, defined as tensor products of Legendre polynomials up to degree $n-1$. Note that $C_j V_j^0 \subset V_j^0$ greatly simplifies the calculation of entries in the system matrix. We refer to Figs. 1 and 2 for the visualization of quadratic cardinal B-splines ($m = 3$) and the Legendre polynomials of order < 2 ($n = 2$). That will be the setting used in our numerical computations. Finally, to approximate in $H^{3/2}(\Gamma)$, we use (after

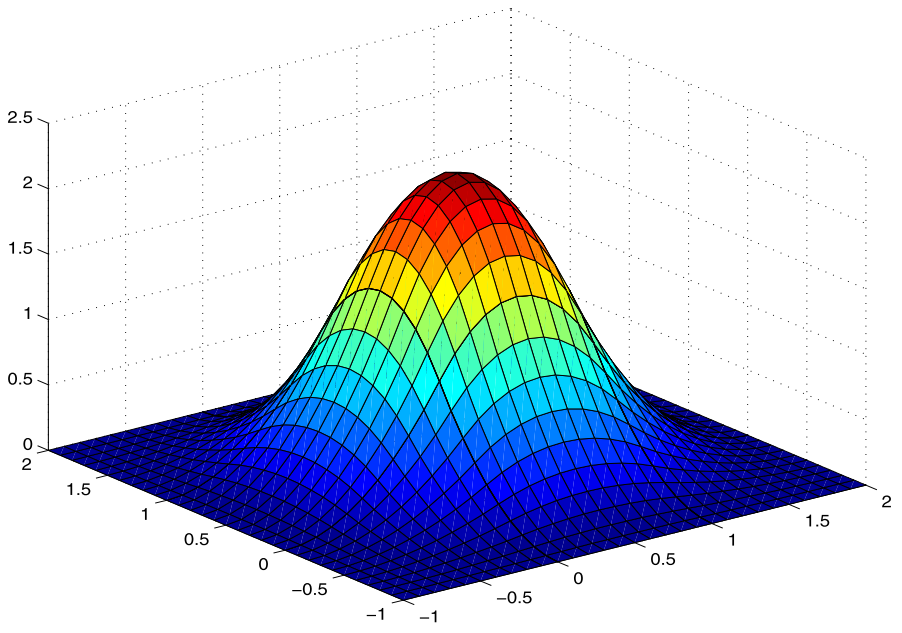


Fig. 1 Quadratic cardinal B-splines as ansatz functions ($m = 3$)

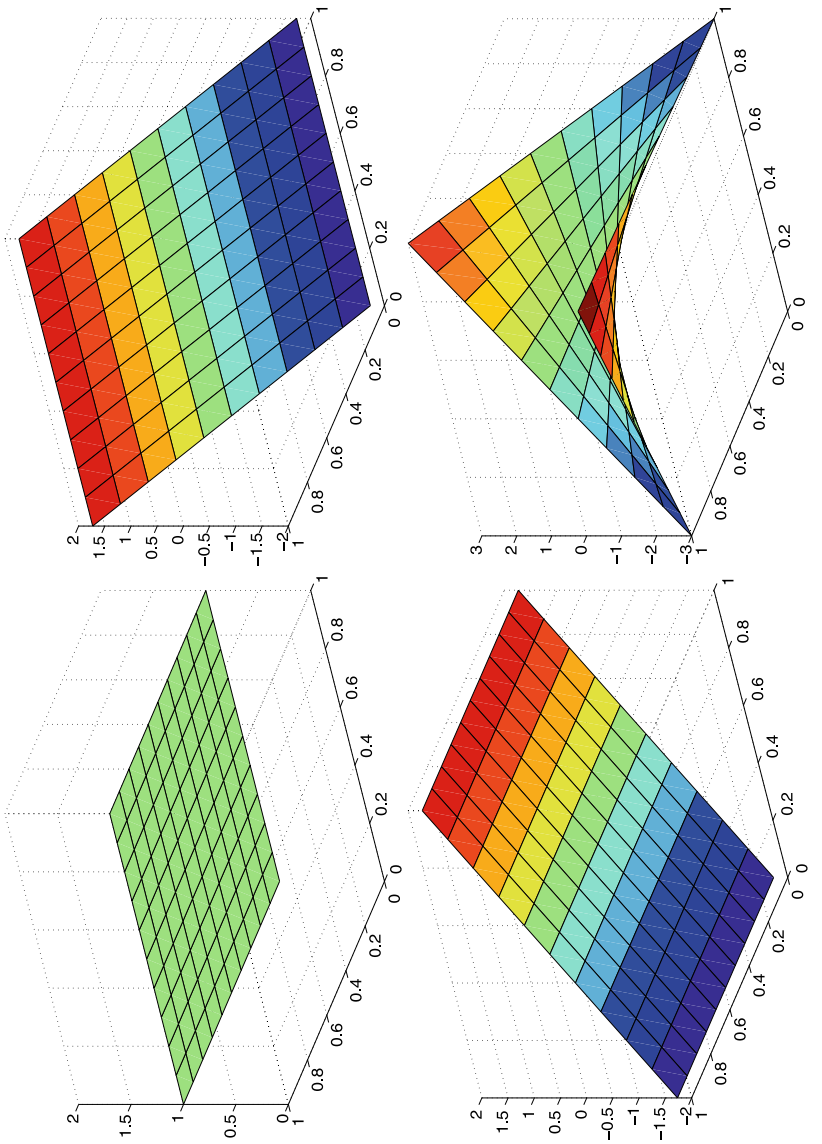


Fig. 2 Discontinuous Legendre polynomials of degree <2 as test functions

identifying Γ with $[0,1]$ by means of the parametrization (2.12)) the space $V_j^\Gamma := V_N^\Gamma$, where V_N^Γ is as defined in (2.11) with $N = 2^j$.

Next, we should introduce the discrete system matrices and load vectors. We have to compute

$$[\mathbf{A}_j]_{(\mathbf{k},\mathbf{l}),\mathbf{k}'} = -\int_{\Omega_j} \operatorname{div}(A\nabla\varphi^{\mathbb{T}}_{j,\mathbf{k}'})\varphi^0_{j,\mathbf{k},\mathbf{l}}d\mathbf{x}, \qquad [\mathbf{f}_j]_{(\mathbf{k},\mathbf{l})} = \int_{\Omega_j} f\varphi^0_{j,\mathbf{k},\mathbf{l}}d\mathbf{x},$$

$$[\mathbf{B}_j]_{k,\mathbf{k}'} = \int_0^1 (\varphi^{\mathbb{T}}_{j,\mathbf{k}'} \circ \boldsymbol{\gamma})\varphi^\Gamma_k d\phi, \qquad [\mathbf{g}_j]_k = \int_0^1 (g \circ \boldsymbol{\gamma})\varphi^\Gamma_k d\phi,$$

where $\boldsymbol{\gamma}$ denotes a suitable parametrization to Γ according to (2.9).

In order to deal with the different norms we need some suitable preconditioners. To compute the $H^{3/2}(\Gamma)$-norm of a function $g_j \in V_j^\Gamma$ we simply have to scale the coefficients of $\sin(k\phi)$ and $\cos(k\phi)$ by $k^{3/2}$. Thus, we shall introduce the diagonal matrix

$$[\mathbf{D}_j]_{k,l} = |k|^{3/2}\delta_{k,l}.$$

For preconditioning of the operator F we could use (as is done in [21]) a suitable wavelet transform, see e.g. [4]. Instead, we use the Bramble-Pasciak-Xu (BPX) multilevel preconditioner [3] associated with the discretization of $I - \operatorname{div}(\mathbf{A}\nabla)$. We indicate its application by the matrix $\mathbf{T}_j$.

We are now in the position to present the discrete least-squares problem: solve

$$\left\| \begin{bmatrix} \mathbf{A}_j \\ \mathbf{D}_j\mathbf{B}_j \end{bmatrix} \mathbf{T}_j\mathbf{v}_j - \begin{bmatrix} \mathbf{f}_j \\ \mathbf{D}_j\mathbf{g}_j \end{bmatrix} \right\| \to \min \tag{3.4}$$

and take $\mathbf{u}_j^+ = \mathbf{T}_j\mathbf{v}_j$.

We use the least-squares solver LSQR to solve the discrete least-squares problem (3.4) iteratively within a nested iteration. LSQR is a numerically stable iterative method for underdetermined linear systems of equations, based on the normal equations to (3.4). We refer the reader to [26] for the details concerning this solver.

Finally, we like to mention that it is not necessary to assemble the matrix $\mathbf{B}_j$ since matrix-vector products $\mathbf{B}_j\mathbf{x}$ and $\mathbf{B}_j^T\mathbf{x}$ can be efficiently evaluated by using the (inverse) fast Fourier transform.

3.3 Error estimates

The energy space of the least-squares formulation (3.1) is the Sobolev space $H^2(\mathbb{T})$. Therefore, since we use ansatz functions that are exact of order m, the best possible convergence rate is limited by h_j^{2m-4}, achieved in the $H^{4-m}(\mathbb{T})$-norm if $u^+ \in H^m(\mathbb{T})$.

Theorem 3.1 *Let $u_j^+ = \sum_{\mathbf{k}}[\mathbf{u}_j^+]\varphi^{\mathbb{T}}_{j,\mathbf{k}} \in V_j^{\mathbb{T}}$ denote a discrete solution of (3.4) and assume that there exists an $n \in [0, m-2]$ such that*

$$\|u - u_j^+\|_{H^{2-n}(\Omega)} \lesssim h_j^{2n}\|u\|_{H^{2+n}(\Omega)} \tag{3.5}$$

provided that $u \in H^{2+n}(\Omega)$. Then, if Γ is sufficiently smooth, the approximate shape functional and gradient

$$\widetilde{J}(\Omega) = \int_\Omega f u_j^+ d\mathbf{x}, \qquad \widetilde{\nabla J}(\Omega)[\mathbf{U}] = \int_\Gamma \langle \mathbf{U}, \mathbf{n}\rangle \langle \mathbf{A}\nabla u_j^+, \nabla u_j^+\rangle d\sigma,$$

satisfy the error estimates

$$|J(\Omega) - \widetilde{J}(\Omega)| = \mathcal{O}(h_j^{2n}), \qquad |\nabla J(\Omega)[\mathbf{U}] - \widetilde{\nabla J}(\Omega)[\mathbf{U}]| = \mathcal{O}(h_j^{\min\{2n,n+1\}}).$$

Proof The approximation error of the shape functional is estimated according to

$$\begin{aligned} |J(\Omega) - \widetilde{J}(\Omega)| &= \left| \int_\Omega f u d\mathbf{x} - \int_\Omega f u_j^+ d\mathbf{x} \right| \\ &\lesssim \|f\|_{H^{n-2}(\Omega)} \|u - u_j^+\|_{H^{2-n}(\Omega)} \\ &\lesssim h_j^{2n} \|f\|_{H^{n-2}(\Omega)} \|u\|_{H^{n-2}(\Omega)}. \end{aligned}$$

In case of the shape gradient we derive the assertion by

$$\begin{aligned} &|\nabla J(\Omega)[\mathbf{U}] - \widetilde{\nabla J}(\Omega)[\mathbf{U}]| \\ &\quad = \left| \int_\Gamma \langle \mathbf{U}, \mathbf{n}\rangle \big\{ \langle \mathbf{A}\nabla u, \nabla u\rangle - \langle \mathbf{A}\nabla u_j^+, \nabla u_j^+\rangle \big\} d\sigma \right| \\ &\quad \le \left| \int_\Gamma \langle \mathbf{U}, \mathbf{n}\rangle \langle \mathbf{A}\nabla(u - u_j^+), \nabla(u - u_j^+)\rangle d\sigma \right| \\ &\qquad + 2 \left| \int_\Gamma \langle \mathbf{A}\nabla u \langle \mathbf{U}, \mathbf{n}\rangle, \nabla(u - u_j^+)\rangle d\sigma \right| \\ &\quad \lesssim \|\langle \mathbf{U}, \mathbf{n}\rangle\|_{L^\infty(\Gamma)} \|u - u_j^+\|_{H^1(\Gamma)}^2 \\ &\qquad + 2 \|\nabla u \langle \mathbf{U}, \mathbf{n}\rangle\|_{H^{1/2}(\Gamma)} \|u - u_j^+\|_{H^{1/2}(\Gamma)}. \end{aligned}$$

Using (3.5) together with the trace theorem yields for the first term

$$\begin{aligned} \|\langle \mathbf{U}, \mathbf{n}\rangle\|_{L^\infty(\Gamma)} \|u - u_j^+\|_{H^1(\Gamma)}^2 &\lesssim \|\langle \mathbf{U}, \mathbf{n}\rangle\|_{L^\infty(\Gamma)} \|u - u_j^+\|_{H^{3/2}(\Omega)}^2 \\ &\lesssim h_j^{2n} \|\langle \mathbf{U}, \mathbf{n}\rangle\|_{L^\infty(\Gamma)} \|u\|_{H^{2+n}(\Omega)}^2. \end{aligned}$$

Likewise, the second term can be estimated by

$$\begin{aligned} \|\nabla u \langle \mathbf{U}, \mathbf{n}\rangle\|_{H^{1/2}(\Gamma)} \|u - u_j^+\|_{H^{1/2}(\Gamma)} &\lesssim \|\nabla u \langle \mathbf{U}, \mathbf{n}\rangle\|_{H^{1/2}(\Gamma)} \|u - u_j^+\|_{H^1(\Omega)} \\ &\lesssim h_j^{\min\{2n,n+1\}} \|\langle \mathbf{U}, \mathbf{n}\rangle\|_{C^1(\Gamma)} \|u\|_{H^{2+n}(\Omega)}^2. \quad \square \end{aligned}$$

Remark 3.2 For ansatz functions of order m the best possible rate of convergence is

$$|J(\Omega) - \widetilde{J}(\Omega)| = \mathcal{O}(h_j^{2m-4}), \qquad |\nabla J(\Omega)[\mathbf{U}] - \widetilde{\nabla J}(\Omega)[\mathbf{U}]| = \mathcal{O}(h_j^{\min\{2m-4,m-1\}}).$$

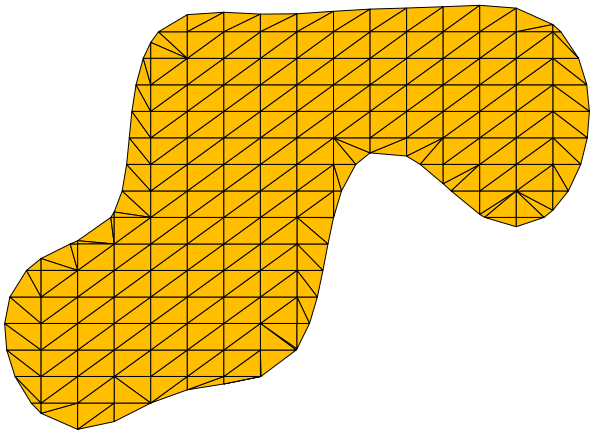


Fig. 3 Triangulation of the domain

We used piecewise quadratic ansatz functions ($m = 3$) in our numerical realization. In this case we can achieve a rate of $\mathcal{O}(h_j^2)$ for both, the shape functional and the gradient.

3.4 Computing domain integrals

In order to evaluate the Dirichlet energy (2.1) we have to approximate domain integrals

$$I(\Omega) := \int_\Omega f(\mathbf{x})\, d\mathbf{x} \tag{3.6}$$

for $f \in C(\overline{\Omega})$. This will be done as follows.

We compute the points of intersection of the boundary curve Γ and the underlying grid $\bigcup_{\mathbf{k}\in\mathbb{Z}} \partial Q_{j,\mathbf{k}}$. Then, we replace the boundary curve Γ by the piecewise linear curve $\widetilde{\Gamma}$ which connects these points by straight lines. The enclosed polygonal domain will be denoted by $\widetilde{\Omega}$.

We will next construct a suitable triangulation of $\widetilde{\Omega}$. We subdivide all elements $Q_{j,\mathbf{k}}$ that intersect the boundary $\widetilde{\Gamma}$ into suitable triangles to triangulate $Q_{j,\mathbf{k}} \cap \widetilde{\Omega}$. In the remaining part of $\widetilde{\Omega}$ we subdivide the elements $Q_{j,\mathbf{k}}$ into two triangles. Finally, we apply appropriate quadrature formulae for triangles. Figure 3 exemplifies a triangulation produced by our algorithm.

Theorem 3.3 *Assume that $\Omega \in C^2$ and $f \in C^2(\mathbb{D})$. Then, the above quadrature algorithm computes the integral $I(\Omega)$ from (3.6) with accuracy $\mathcal{O}(h_j^2)$ provided that the element quadrature formulae are exact for linear polynomials.*

Proof The triangulation consists of $\mathcal{O}(h_j^{-2})$ elements of volume $\mathcal{O}(h_j^2)$. Consequently, since the element quadrature formulae are exact of order two, we get an error of quadrature $\mathcal{O}(h_j^4)$ per element. Thus, denoting the result of the composite quadrature formula by $Q(\widetilde{\Omega})$, we conclude

$$|I(\widetilde{\Omega}) - Q(\widetilde{\Omega})| = \mathcal{O}(h_j^2). \tag{3.7}$$

We shall next estimate the error induced by the domain approximation. Since $\widetilde{\Gamma}$ is a piecewise linear approximation of step width $\sim h_j$ to the boundary curve Γ, the area $V(Q_{j,\mathbf{k}} \cap \Omega)$ of each square $Q_{j,\mathbf{k}}$ for which $Q_{j,\mathbf{k}} \cap \widetilde{\Gamma} \neq \emptyset$ is approximated of order

$$|V(Q_{j,\mathbf{k}} \cap \widetilde{\Omega}) - V(Q_{j,\mathbf{k}} \cap \Omega)| = \mathcal{O}(h_j^3).$$

Taking into account that there are at most $\mathcal{O}(h_j^{-1})$ squares that intersect the boundary curve, we conclude

$$|I(\Omega) - I(\widetilde{\Omega})| = \mathcal{O}(h_j^2). \tag{3.8}$$

Combining both estimates yields the assertion due to

$$|I(\Omega) - Q(\Omega)| \leq |I(\Omega) - I(\widetilde{\Omega})| + |I(\widetilde{\Omega}) - Q(\widetilde{\Omega})|. \qquad \square$$

Remark 3.4 In three dimensions one introduces a triangulation of the free surface and henceforth a tetrahedral mesh of the domain. As one readily verifies the same error estimate holds while the complexity of the algorithm is $\mathcal{O}(h_j^{-3})$ instead $\mathcal{O}(h_j^{-2})$.

4 Numerical experiments

4.1 Solving the state equation

We first investigate the asymptotic behaviour of our fictitious domain solver. We use lowest order ansatz functions, that is, quadratic B-splines ($m = 3$), and discontinuous piecewise bilinear test functions ($n = 2$).

To measure the rates of convergence of the SPFD method we will focus on a boundary value problem where the solution is known analytically. To this end, we consider the following boundary value problem

$$\begin{aligned} -\operatorname{div}(\mathbf{A}\nabla u) &= \cos(x)\big(4 + \sin^2(y)\big) - 6y\big(2 + \sin(x)\big) \quad \text{in } \Omega, \\ u &= \cos(x) + y^3 \quad \text{on } \Gamma, \end{aligned}$$

where

$$\mathbf{A}(x, y) = \begin{bmatrix} 4 - \sin^2(y) & -1 \\ -1 & 2 + \sin(x) \end{bmatrix}.$$

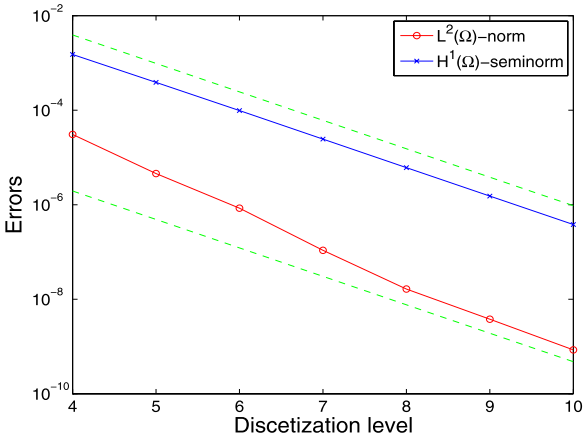
We choose the same domain Ω as underlying in Fig. 3. One readily verifies that the solution is given by the function $u = \cos(x) + y^3$.

We compute the numerical solution u_j^+ for different discretization levels j by the smoothness preserving fictitious domain method proposed in the previous section. Since $m = 3$ we expect in $H^1(\Omega)$ an at most quadratic rate of convergence. In Table 1 we tabulate the absolute errors with respect to the L^2-norm and H^1-seminorm on Ω, respectively. The bracketed values indicate the ratio of the previous error and the present error. It is about 4 which implies quadratic orders of convergence. We illustrated the different error curves also in Fig. 4. As indicated by the dashed lines one observes in fact quadratic rates of convergence for both norms.

Table 1 Errors of approximation and over-all computing times

j	$\|u^+ - u_j^+\|_{L^2(\Omega)}$	$\|\nabla(u^+ - u_j^+)\|_{L^2(\Omega)}$	cpu-time
4	3.1e–5	1.5e–3	0.3 s
5	4.6e–6 (6.7)	3.9e–4 (3.9)	1 s
6	8.5e–7 (5.4)	9.8e–5 (4.0)	6 s
7	1.1e–7 (7.8)	2.4e–5 (4.0)	30 s
8	1.6e–8 (6.6)	6.1e–6 (4.0)	128 s
9	3.8e–9 (4.4)	1.5e–6 (4.0)	10 min
10	8.5e–10 (4.5)	3.8e–7 (4.0)	44 min

Fig. 4 Rates of convergence



We emphasize that, according to Theorem 3.1, we will approximate both, the shape functional and the shape gradient, with *quadratic* orders of convergence.

The last column of Table 1 refers to the over-all computing times to produce the approximate solution u_j^+. The present implementation is still on experimental level, being a mixture of MATLAB and C-Codes. Nevertheless, the method is feasible and highly accurate.

In Fig. 5 we present the approximate solution u_j^+ at level $j = 6$, i.e., for 4096 ansatz functions. Indeed, we do not observe the characteristic singularity along the boundary Γ, common to most fictitious domain methods. The solution is smooth on the whole square.

4.2 Application to shape optimization problems

We shall finally solve a shape optimization problem. We choose the diffusion matrix

$$\mathbf{A}(x, y) = \begin{bmatrix} 4 + 2.75\sin(10x) & -1 \\ -1 & 2 + \sin(3x) \end{bmatrix}$$

and the inhomogeneity

$$f(x, y) = 2(1 - 3x^2)(1 - 3y^2)$$

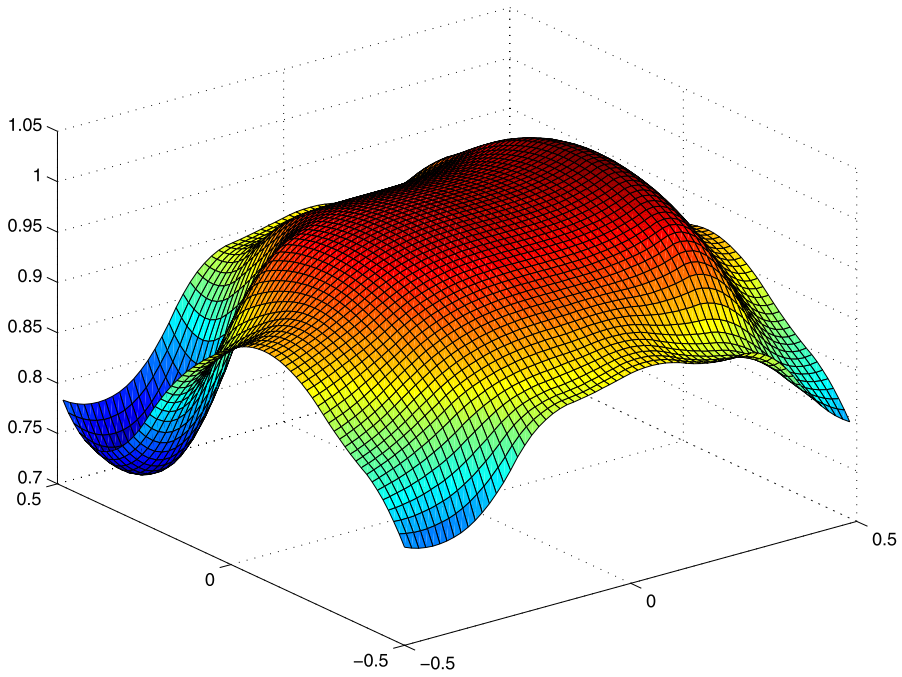


Fig. 5 The approximate solution u_j^+ at level $j = 6$

as the data of the state equation (2.2). Moreover, we consider the volume constraint $V(\Omega) \stackrel{!}{=} V_0 := 0.2$.

The numerical setting is as follows. To approximate the boundary curve we choose $N = 16$ which yields 66 shape design parameters (cf. Sect. 2.4). Moreover, we perform 5 inner and 20 outer iterations of the augmented Lagrangian algorithm (cf. Sect. 2.3), where $\alpha := 100$ does a good job (see (2.7)). The regularization parameter is chosen as $\beta^{(n)} = 2^{-n}/100$ where n denotes the number of the outer iteration. The discretization level of the fictitious domain method is set to $j := 7$.

The domain computed by our algorithm is shown in Fig. 6. The algorithm consumes about 1 hour cpu-time to derive this solution.

To be on safe ground we validated the result by comparing it with the solution of a shape optimization algorithm based on starlike domains and finite elements (on starlike domains one can define the triangulation via parametrization). The maximizing domains produced by both algorithms coincide.

We use finite Fourier series to approximate the parametrization, which means that we apply the p-method for domain approximation. Therefore, we would observe spurious oscillations and possibly divergence if the state solver is not accurate enough to resolve the boundary approximation. We emphasize that the present method is highly accurate and thus we do not observe such behaviour.

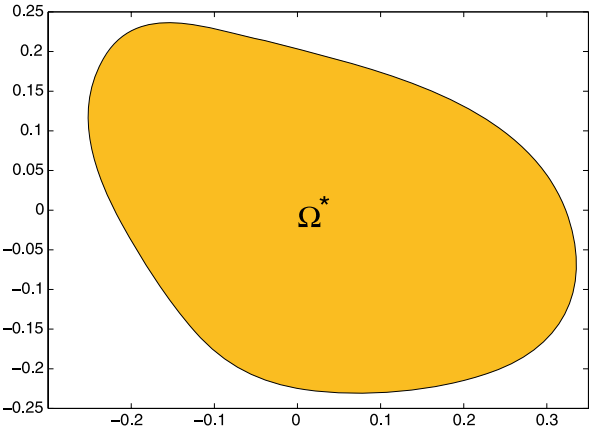


Fig. 6 The maximizing domain

5 Concluding remarks

In the present paper we applied the novel smoothness preserving fictitious domain method from [21, 22] to a shape optimization problem. We derived discretization techniques which are applicable to two and three dimensional problems. Numerical results demonstrated that the new fictitious domain method is a quite promising meshless pde-solver as required for the development of fast and robust algorithms in shape optimization. Meshless means here that the mesh plays a tangential role, cf. [1]. In particular, it has not to be adapted to the actual domain.

References

1. Babuška, I., Banerjee, U., Osborn, J.E.: Survey of meshless and generalized finite element methods: a unified approach. Acta Numer. **12**, 1–125 (2003)
2. Belov, S., Fujii, N.: Symmetry and sufficient condition of optimality in a domain optimization problem. Control Cybern. **26**, 45–56 (1997)
3. Bramble, J., Pasciak, J.E., Xu, J.: Parallel multilevel preconditioners. Math. Comput. **55**, 1–22 (1990)
4. Dahmen, W.: Wavelet and multiscale methods for operator equations. Acta Numer. **6**, 55–228 (1997)
5. Delfour, M., Zolesio, J.-P.: Shapes and Geometries. SIAM, Philadelphia (2001)
6. Dennis, J.E., Schnabel, R.B.: Numerical Methods for Nonlinear Equations and Unconstrained Optimization Techniques. Prentice-Hall, Englewood Cliffs (1983)
7. Eppler, K., Harbrecht, H.: 2nd order shape optimization using wavelet BEM. Optim. Method. Softw. **21**, 135–153 (2006)
8. Eppler, K., Harbrecht, H.: Exterior electromagnetic shaping using wavelet BEM. Math. Method. Appl. Sci. **28**, 387–405 (2005)
9. Eppler, K., Harbrecht, H.: Fast wavelet BEM for 3d electromagnetic shaping. Appl. Numer. Math. **54**, 537–554 (2005)
10. Eppler, K., Harbrecht, H.: Efficient treatment of stationary free boundary problems. Appl. Numer. Math. **56**, 1326–1339 (2006)
11. Eppler, K., Harbrecht, H.: Coupling of FEM and BEM in shape optimization. Numer. Math. **104**, 47–68 (2006)
12. Fiacco, A.V., McCormick, G.P.: Nonlinear Programming: Sequential Unconstrained Minimization Techniques. Wiley, New York (1968)
13. Fletcher, R.: Practical Methods for Optimization, vols. 1, 2. Wiley, New York (1980)
14. Haslinger, J., Neitaanmäki, P.: Finite Element Approximation for Optimal Shape, Material and Topology Design, 2nd edn. Wiley, Chichester (1996)

15. Haslinger, J., Mäkinen, R.A.E.: Introduction to Shape Optimization. Theory, Approximation and Computation. Advances in Design and Control, vol. 7. SIAM, Philadelphia (2003)
16. Glowinski, R., Pan, T.W., Wells, R.O. Jr., Zhou, X.: Wavelet and finite element solutions for the Neumann problem using fictitious domains. J. Comput. Phys. **126**, 40–51 (1996)
17. Glowinski, R., Pan, T.W., Periaux, J.: A fictitious domain method for Dirichlet problem and applications. Comput. Method. Appl. Mech. Eng. **111**(3–4), 283–303 (1994)
18. Grossmann, Ch., Terno, J.: Numerik der Optimierung. Teubner, Stuttgart (1993)
19. Harbrecht, H., Hohage, T.: Fast methods for three-dimensional inverse obstacle scattering. J. Integral Equ. Appl. **19**(3), 237–260 (2007)
20. Kunisch, K., Peichl, G.: Shape optimization for mixed boundary value problems on an embedding domain method. Dyn. Contin. Discret. Impuls. Syst. **4**, 439–478 (1998)
21. Mommer, M.S.: A smoothness preserving fictitious domain method for elliptic boundary value problems. IMA J. Numer. Anal. **26**, 503–524 (2006)
22. Mommer, M.S.: Towards a fictitious domain method with optimally smooth solutions. Ph.D. thesis, RWTH-Aachen, published online by the RWTH-Aachen (2005)
23. Murat, F., Simon, J.: Étude de problémes d'optimal design. In: Céa, J. (ed.) Optimization Techniques, Modeling and Optimization in the Service of Man. Lecture Notes on Computer Science, vol. 41, pp. 54–62. Springer, Berlin (1976)
24. Neitaanmäki, P., Tiba, D.: An embedding of domain approach in free boundary problems and optimal design. SIAM J. Control Optim. **33**, 1587–1602 (1995)
25. Novruzi, A., Roche, J.R.: Newton's method in shape optimisation: A three-dimensional case. BIT **40**, 102–120 (2000)
26. Paige, C.C., Saunders, M.A.: LSQR: An algorithm for sparse linear equations and sparse least squares. ACM Trans. Math. Softw. **8**, 43–71 (1982)
27. Pierre, M., Roche, J.-R.: Numerical simulation of tridimensional electromagnetic shaping of liquid metals. Numer. Math. **65**, 203–217 (1993)
28. Pironneau, O.: Optimal Shape Design for Elliptic Systems. Springer, New York (1983)
29. Roche, J.-R., Sokolowski, J.: Numerical methods for shape identification problems. Control Cybern. **25**, 867–894 (1996)
30. Simon, J.: Differentiation with respect to the domain in boundary value problems. Numer. Funct. Anal. Optim. **2**, 649–687 (1980)
31. Slawig, T.: Domain optimization for the stationary Stokes and Navier–Stokes equations by an embedding domain technique, Ph.D. thesis, TU Berlin (1998)
32. Slawig, T.: A formula for the derivative with respect to domain variations in Navier–Stokes flow based on an embedding domain method. SIAM J. Control Optim. **42**, 495–512 (2003)
33. Sokolowski, J., Zolesio, J.-P.: Introduction to Shape Optimization. Springer, Berlin (1992)