Journal of Computational Physics 228 (2009) 6291–6315

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Level-set, penalization and cartesian meshes: A paradigm for inverse problems and optimal design

Frédéric Chantalat [a], Charles-Henri Bruneau [a], Cédric Galusinski [b], Angelo Iollo [a,*]

[a] Université de Bordeaux, IMB and INRIA Futurs, MC2, CNRS UMR 5251, 351 Cours de la Libération, Talence F-33405, France
[b] Université du Sud Toulon Var, Imath, Avenue de l'Université, BP20132 La Garde Cedex F-83957, France

ARTICLE INFO

Article history:
Received 19 June 2008
Received in revised form 12 March 2009
Accepted 11 May 2009
Available online 18 May 2009

Keywords:
Inverse problem
Optimal design
Domain penalization
Level-set

ABSTRACT

The aim of this work is to combine penalization and level-set methods to solve inverse or shape optimization problems on uniform cartesian meshes. Penalization is a method to impose boundary conditions avoiding the use of body-fitted grids, whereas level-sets allow a natural non-parametric description of the geometries to be optimized. In this way, the optimization problem is set in a larger design space compared to classical parametric representation of the geometries, and, moreover, there is no need of remeshing at each optimization step. Special care is devoted to the solution of the governing equations in the vicinity of the penalized regions and a method is introduced to increase the accuracy of the discretization. Another essential feature of the optimization technique proposed is the shape gradient preconditioning. This aspect turns out to be crucial since the problem is infinite dimensional in the limit of grid resolution. Examples pertaining to model inverse problems and to shape design for Stokes flows are discussed, demonstrating the effectiveness of this approach.

1. Introduction

Inverse geometry problems arise in many different areas of research, from tomographic imaging in medical or environmental fields [1] to non-destructive control of materials [2], to name just a few. Also, as both European and international regulations on greenhouse gases emissions are growing stricter and as fuel prices are rising, the automotive and aeronautical industries, besides improvements on motors and masses of their engines, are required to look for aerodynamic ways to reduce oil consumption, particularly through an accurate shape design leading to drag reduction.

The main objective of this paper is to couple penalization methods [3] and level-set methods [4,6–8] to release some difficulties encountered when dealing with shape optimization problems. In shape optimization, a boundary is modified in order to minimize a cost functional, given certain performance requirements and geometric constraints. The techniques employed to solve inverse or shape optimization problems usually rely either on gradient methods or on evolutionary algorithms. In both cases, many solutions of the direct problem on different geometries are required. This implies routinely remeshing the modified shapes, which is a difficult and time-consuming task for realistic three-dimensional cases. In order to allow the feasibility of shape design applied to flows past complex geometries, we use the level-set method to describe the obstacles to be optimized. This way, the geometry is defined only by one scalar function and its geometric characteristics are readily accessible. Nevertheless, this aspect is not new; what is new is the application of an improved penalization method taking advantage of the distance level-set function to increase the discretization accuracy. Indeed, using a standard finite

* Corresponding author.
E-mail addresses: frederic.chantalat@math.u-bordeaux1.fr (F. Chantalat), bruneau@math.u-bordeaux1.fr (C.-H. Bruneau), galusins@univ-tln.fr (C. Galusinski), iollo@math.u-bordeaux1.fr (A. Iollo).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.05.017

difference scheme on a fixed cartesian grid,the boundary conditions are taken into account by adding a penalization term in the right-hand side. In the limit, this ensures that the relevant boundary condition is taken into account with an error, close to the boundary, that is proportional to the size of the grid. As it will be shown, higher accuracy can be recovered by a judicious choice of the penalization term. The key element for this choice is the distance level-set function: in order to compute an appropriate interpolation formula at the boundary, we need the normal to the boundary as well as the distance from it. These quantities are easily computed exploiting the level-set definition. In other words, one novelty of the present approach is to take advantage of the level-set in the formulation of the discretization method to solve the governing partial differential equations (PDE). An additional element of innovation steaming from coupling penalization and level-sets, is the possibility of setting inverse or optimization problems without the constraints imposed by an explicit parametrization of the shapes. For example, topologies different from those characterizing the first guess will likely be achieved in a natural way. In [9] an ersatz material approach based on a level-set method was used to satisfy an approximate continuity condition on the stress tensor in the context of structural optimization. That method can be interpreted as a penalization on the solution normal derivative leading to a flux continuity boundary condition across the interface. However, due to the displacement caused by the non-conforming mesh, this condition is not exactly imposed at the domain boundary. Here, we are mainly concerned with fluid-flow applications and therefore we focus on how to impose a Dirichlet type condition which results in a consistent gradient discretization at the boundary.

The initial part of this work deals with the implementation of this strategy on a two-dimensional inverse model problem satisfying the Poisson equation. We address there the question of the accuracy of the solution gradient at the boundary of a body when using cartesian meshes. The same computational method is then used to solve an adjoint problem that helps defining the perturbation of an error functional with respect to the geometry. Some additional procedures aiming at improving the convergence of a steepest descent method are introduced. First, a multiscale approach based on an appropriate choice of the cost function to be minimized is described. Then, the focus will be on avoiding local minima: a scanning method will show that changing the boundary conditions of the inverse problem can improve the convergence of the minimization. Finally, in the spirit of what was done by Allaire et al. [9], the coupling of a topological gradient [10–12] with the previously mentioned tools will be discussed. Indeed, the shape to be optimized can get stuck at optimal configurations with less holes than the actual geometries to be retrieved. It turns out that the relevance of adding such holes at given places is provided by the knowledge of the sign of the adjoint solution, initially introduced to compute the shape gradient, as detailed hereafter.

The second part of this article is devoted to a fluid-flow application: constrained optimal shape design. Considering an initial obstacle embedded in a two-dimensional Stokes flow, the goal is to find a geometry that has the same area as the initial one and that minimizes a given objective function. The drag force is taken as the functional to be minimized, or equivalently in absence of lift, the norm of the velocity gradient. The main results show that the idea of combining level-sets and penalization methods on cartesian meshes leads to the solution of more general classes of optimal shape problems. It is shown that an efficient regularization, usually attained by projection of the gradient in a finite dimensional space, is obtained in this context by an appropriate diffusion operator.

2. A model inverse problem

Let $\overline{\Omega}_{obs}$ be an obstacle embedded inside an open domain $\Omega \subset \mathbb{R}^2$. The borders are denoted $\partial\Omega_{obs}$ and $\partial\Omega$, respectively. The model problem we consider is the minimization of the least-square cost function:

$$J_m(\overline{\Omega}_{obs}, v(\overline{\Omega}_{obs})) = \frac{1}{2}\int_{\Omega\setminus\overline{\Omega}_{obs}} |v-u|^2\, d\Omega. \tag{1}$$

The relation between the shape Ω_{obs} and v is given by the following elliptic partial differential equation and boundary conditions (see Fig. 1):

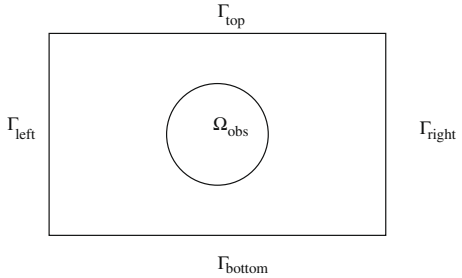

Fig. 1. Setting of the Poisson problem. $\partial\Omega = \Gamma_{left} \cup \Gamma_{right} \cup \Gamma_{top} \cup \Gamma_{bottom}$.

$$\begin{cases} -\Delta v = 1 \quad \text{in } \Omega \setminus \overline{\Omega}_{obs}, \\ \frac{\partial v}{\partial n} = 0 \quad \text{on } \Gamma_{left} \cup \Gamma_{right}, \\ v = 1 \quad \text{on } \Gamma_{top}, \ v = -1 \text{ on } \Gamma_{bottom}, \ v = 0 \text{ on } \partial\Omega_{obs}. \end{cases} \tag{2}$$

The function u is the solution of this Poisson equation with appropriate Dirichlet and Neumann boundary conditions corresponding to the optimal obstacle Ω^*_{obs} to be reconstructed and $\underline{n}$ is the outward pointing unit normal to a given edge.

2.1. Level-set representation and tracking of the obstacle

The main feature of the level-set method is to be able to represent a curve or a surface without an explicit parametrization. Instead, an interface such as the boundary of the body $\overline{\Omega}_{obs}$ is considered as the zero level-set of an higher-dimensional function ϕ defined as follows:

$$\begin{aligned} &\phi : \mathbb{R}^2 \times \mathbb{R}^+ \to \mathbb{R}, \\ &\phi(\underline{x}, t) = \begin{cases} \phi(\underline{x}, t) = +d(>0) & \text{if } \underline{x} \in \overline{\Omega}_{obs}(t), \\ \phi(\underline{x}, t) = -d(<0) & \text{if } \underline{x} \in \Omega \setminus \overline{\Omega}_{obs}(t), \\ \phi(\underline{x}, t) = 0 & \text{if } \underline{x} \in \partial\Omega_{obs}(t), \end{cases} \end{aligned} \tag{3}$$

where d is the distance from $\underline{x}$ to the border. At each node of the mesh, the value of this distance function ϕ is supposed to be known. Assuming that the displacement of the shape is determined by a continuous velocity vector field $\underline{V} : \mathbb{R}^2 \to \mathbb{R}^2$ such that $\left\{ \frac{d\underline{x}}{dt} = \underline{V}(\underline{x}(t)); \ \underline{x} \in \partial\Omega_{obs}(t) \right\}$, the evolution of the level-set function ϕ satisfies the first-order transport equation defined on the whole space $\mathbb{R}^2$:

$$\frac{\partial \phi}{\partial t} + \underline{V} \cdot \nabla \phi = 0. \tag{4}$$

By introducing the normal velocity $V_n = \underline{V} \cdot \frac{\nabla \phi}{|\nabla \phi|}$, this equation can be formulated as an initial value Hamilton–Jacobi equation,

$$\begin{cases} \frac{\partial \phi}{\partial t} + V_n |\nabla \phi| = 0, \\ \phi(\underline{x}, t = 0) = \phi_0. \end{cases} \tag{5}$$

If the velocity is defined only on the interface $\partial\Omega_{obs}(t)$, its definition on the border can be usually extended to the whole domain.

This approach for handling both representation and motion of an obstacle offers a number of advantages. First of all, it can be naturally extended in higher dimensions. Then, as it will be shown in several examples in this article, some types of topological changes such as the merging or breaking of the evolving edges given by the zero sets of $\phi(\underline{x}, t)$ are handled naturally. Furthermore, this method enables straightforward computation of geometric terms such as the exterior unit normal vector $\underline{n} = \frac{\nabla \phi}{|\nabla \phi|}$ or the curvature $\chi = \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right)$, whose expressions rely on the use of differentiation operators applied to the level-set function. Finally, on the fixed cartesian grid used to track the motion of the interface, the gradients of all quantities can be approximated using a finite difference discretization. For instance, the norm of the level-set gradient we use in the following is:

$$|\nabla \phi|^2 = \left(\frac{(\phi_{ij} - \min(\phi_{i+1j}, \phi_{i-1j}))}{\Delta x} \right)^2 + \left(\frac{(\phi_{ij} - \min(\phi_{ij+1}, \phi_{ij-1}))}{\Delta y} \right)^2, \tag{6}$$

where Δx and Δy stand for the space discretization steps, respectively, in the horizontal and vertical directions. In the numerical examples that we present in this paper pathological configurations at the limit of numerical resolution, such as small material islands, are systematically eliminated. In practice we change the sign of ϕ at a meshpoint whenever $\phi < 0$ and the neighbouring meshpoints are such that $\phi > 0$.

A higher order discretization of (4) or (5) is fulfilled by a WENO5 scheme [13], for a level-set function ϕ close to a distance function. A redistanciation procedure of the function ϕ has then to be periodically solved on a neighborhood of the interface. This procedure is performed by the fast-marching method [8], which is an efficient technique for redistanciation.

2.2. Penalization and gradient improvement

In the following sections, we discuss how to improve the accuracy of the solution gradients close to the boundary of an obstacle modeled by a penalization term in the governing equations. This is a key issue in shape design, since the shape derivative is often expressed in terms of the gradients of both the direct and adjoint problems. Therefore, if the solution is only first-order accurate in the vicinity of the body as it is for usual penalization methods, the solution gradients at the body are unreliable.

2.2.1. Handling Dirichlet boundary conditions using level-set

Building a mesh with cell edges directly coinciding with a frontier is the most usual way of representing an obstacle. The main advantage of this method is to enable straight enforcement of the Dirichlet boundary condition on discretization nodes of $\partial\Omega_{obs}$. However, meshing very complicated geometries can be a very costly task. This process might even become prohibitive when included in shape optimization loops or for other problems where the obstacle is modified. Coupling the level-set method with penalization allows to get rid of those issues, but the Dirichlet boundary condition must be imposed differently, as the grid no longer conforms to the body. Basically, the aim is now to find an approximation of the solution v^ϵ that is forced to be close to zero on each node inside the obstacle, thanks to the addition of a penalization term to Eq. (2):

$$\begin{cases} -\Delta v^\epsilon + \frac{1}{\epsilon} H(\phi) v^\epsilon = 1 \quad \text{in } \Omega, \\ \frac{\partial v^\epsilon}{\partial n} = 0 \quad \text{on } \Gamma_{left} \cup \Gamma_{right}, \\ v^\epsilon = 1 \quad \text{on } \Gamma_{top}, \\ v^\epsilon = -1 \quad \text{on } \Gamma_{bottom}, \end{cases} \tag{7}$$

where H is the Heaviside function

$$H(\phi(\underline{x},t)) = \begin{cases} 1 & \text{if } \phi(\underline{x},t) > 0, \\ 0 & \text{if } \phi(\underline{x},t) < 0. \end{cases}$$

Inspired by the Darcy equations in fluid dynamics, this approach is equivalent to considering the solid shape as a porous medium whose permeability ϵ is a very small parameter. On the discrete level, from a second-order approximation of the Laplace operator, we obtain an ill-conditioned non-symmetric linear system $A_h v_h^\epsilon = f_h$, with A_h including the boundary conditions on $\partial\Omega$ and a generic term

$$(A_h v_h)_{i,j} = \frac{-v^\epsilon_{i-1,j} + 2v^\epsilon_{i,j} - v^\epsilon_{i+1,j}}{\Delta x^2} + \frac{-v^\epsilon_{i,j-1} + 2v^\epsilon_{i,j} - v^\epsilon_{i,j+1}}{\Delta y^2} + \frac{1}{\epsilon} H(\phi_{i,j}) v^\epsilon_{i,j}, \tag{8}$$

that is solved by GMRES [14]. The penalized equation (7) allows to efficiently find an approximate solution to (2) in L^2 norm [3]. Nevertheless, the gradient of the solution on the boundary has no reason to be consistent in L^2 norm because of the mesh-depending obstacle. The main difficulty is then to compute an accurate Dirichlet to Neumann problem using a precise definition of the shape provided by the function ϕ.

2.2.2. Correction of the solution and its gradient near the boundary

The description of the geometry by means of the term $H(\phi)$ in (7) makes the obstacle mesh-dependent. The normal to such a numerical obstacle as well as $\frac{\partial v^\epsilon}{\partial n}$ are not accurate. In this subsection, we propose an iterative algorithm for the correction of the solution near the zero level-set of the distance function ϕ, so that the normal derivative is continuous across the interface.

This process relies on the idea of finding an adequate non-zero penalized term v^ϵ_{penal} in the obstacle such that the classical penalization appearing in (7) is replaced by

$$-\Delta v^\epsilon + \frac{1}{\epsilon} H(\phi)\left(v^\epsilon - v^\epsilon_{penal}\right) = 1 \quad \text{in } \Omega.$$

This inhomogeneous penalization is such that $v^\epsilon_{penal} = 0$ where $\phi = 0$ and such that the term $\nabla v^\epsilon_{penal} \cdot \nabla\phi$ approaches the expected normal derivative on the interface. This construction of v_{penal} is reached by an iterative process leading to

$$v^\epsilon_{penal} = \phi\alpha,$$

where α fits the normal derivative of v^ϵ on the interface. The scalar function α is constant along the field $\nabla\phi$ and is then a natural extension of the normal derivative of the solution. Note that with such an inhomogeneous penalization term, boundary layers due to classical penalization are reduced since the solution v^ϵ is then formally uniformly bounded (with respect to ϵ) in C^1 norm. Then, a consistent normal derivative of v^ϵ around the interface can then be computed with usual centered schemes.

Let $v^\epsilon_{(0)}$ be the solution of (7), which, as proved in [3], is of order ϵ inside the obstacle. The iterative method to compute v^ϵ_{penal} is the following:

- Step 1: We start by computing the normal gradient of the penalized solution

$$\alpha_k = \nabla v^\epsilon_{(k)} \cdot \frac{\nabla\phi}{|\nabla\phi|}, \quad \text{where } \partial_x v^\epsilon_{(k)} = \begin{cases} \frac{v^\epsilon_{(k)i+1,j} - v^\epsilon_{(k)i,j}}{\Delta x} & \text{if } \phi_{i+1,j} < \phi_{i-1,j} \text{ at point } (i,j), \\ \frac{v^\epsilon_{(k)i,j} - v^\epsilon_{(k)i-1,j}}{\Delta x} & \text{otherwise.} \end{cases}$$

- Step 2: The values of $\alpha_{(k)}$ are prolonged from the outside of Ω_{obs} to the inside by advancing for a few time steps the advection equation:

$$\partial_t \alpha_k + H(\phi + \sigma)\frac{\nabla\phi}{|\nabla\phi|}\nabla\alpha_k = 0, \tag{9}$$

where $\sigma = \frac{\Delta x}{2}$.

- Step 3: From the new value on the normal gradient in Ω_{obs}, we set $v^{\epsilon}_{penal,k} = \phi\alpha_k$ and determine a new approximation of the solution of (2) that satisfies, with unchanged boundary conditions on $\partial\Omega$, the modified penalized equation:

$$-\Delta v^{\epsilon}_{(k+1)} + \frac{1}{\epsilon}H(\phi)\left(v^{\epsilon}_{(k+1)} - v^{\epsilon}_{penal,k}\right) = 1 \quad \text{in } \Omega. \tag{10}$$

- Steps 1–3 are repeated until convergence.

The purpose of the following numerical test is to show the convergence of the iterative procedure and to compare the accuracy of the solution normal derivative at the border of the penalized region to its analytical value. The set-up of this test should not be confused with that of the optimization problem. A disc of radius $r_0 = 2$ centered at $(x_0, y_0) = (5,5)$ is placed in a computational domain $[0,10] \times [0,10]$. The distance function ϕ is defined to be negative inside this circle and we solve the equation $-\Delta v + \frac{1}{\epsilon}H(\phi)v = f$. The right-hand side f is chosen such that $v_{|\phi(\mathbf{x},t)<0} = (x - x_0)^2 + (y - y_0)^2 - r_0^2$. In Table 1 a comparison between the analytical normal gradient of the solution and the numerical one, over a layer of thickness $5\Delta x$ surrounding the interface, shows indeed that one iteration is enough to achieve significant improvement of the initial penalization and that our suggested technique is stable. Moreover, the error on the normal derivative is of order Δx.

In Fig. 2, we plot the trace of the solution to the above test problem at $y = 5$ as a function of the gradient improvement step. The mesh is 201×201. Small deviations at the boundary induce appreciable differences in the domain between the classical penalization method (iteration 0) and the final solution (iteration 4). In the penalized region ($y \leqslant 3$ or $y \geqslant 7$), while $v^{\epsilon}_{(0)}$ is of order ϵ when $v^{\epsilon}_{penal} = 0$ (iteration 0), this is no longer true at subsequent iterations, when $v^{\epsilon}_{(k+1)} = v^{\epsilon}_{penal,k} = \phi\alpha_k \neq 0$.

Table 1
Evolution of the relative L^{∞} error on the normal derivative of the solution of the test problem near the boundary $e_k = \frac{\|\partial_n(v^{\epsilon}_{(k)} - v_a)\|_{L^{\infty}}}{\|\partial_n v_a\|_{L^{\infty}}}$ as a function of the iterations to update $v^{\epsilon}_{penal,k}$. The iteration $k = 0$ corresponds to the homogeneous penalization, i.e., $v^{\epsilon}_{penal,k} = 0$. The analytical solution is denoted by v_a.

Resolution	e_0	e_1	e_2	e_3
101×101	0.2456	0.1259	0.1254	0.1254
201×201	0.2109	0.0679	0.0656	0.0655
401×401	0.2118	0.0369	0.0341	0.0339
801×801	0.2004	0.0207	0.0183	0.0181

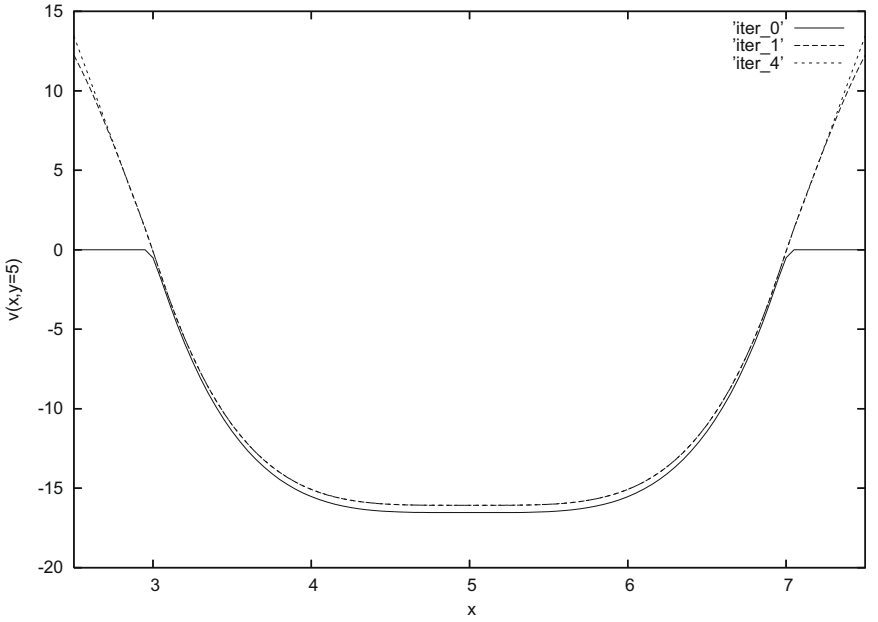

Fig. 2. Solution at $y = 5$ as a function of the gradient improvement iteration.

2.3. Adjoint formulation and shape derivative

Going back to the optimization problem, we discuss in this subsection a preliminary approach to solve our model inverse problem (1) and (2). The resolution relies first on the definition of an adjoint problem to the direct one (2), and then on the computation of the shape derivative in order to find an appropriate velocity for the displacement of the boundary of the obstacle.

2.3.1. Definition of the adjoint problem

Let $\lambda_m : \Omega \setminus \overline{\Omega}_{obs} \to \mathbb{R}$ and $\mu_m : \partial\Omega_{obs} \to \mathbb{R}$ be two smooth enough real-valued functions referred as Lagrange multipliers. We have $\Omega \setminus \overline{\Omega}_{obs} = \{\underline{x}/\phi(\underline{x}) < 0\}$, $\overline{\Omega}_{obs} = \{\underline{x}/\phi(\underline{x}) \geqslant 0\}$ and $\partial\Omega_{obs} = \{\underline{x}/\phi(\underline{x}) = 0\}$. In the following we switch between these definitions in order to make clear functional dependences.

As introduced in [5], the minimization of the objective function $J_m(\phi, v(\phi)) = \frac{1}{2}\int_{\{\underline{x}/\phi(\underline{x})<0\}} |v-u|^2\, d\Omega$, subject to the direct problem (2), is equivalent to find the stationary points of the Lagrangian functional

$$L_m(v, \lambda_m, \mu_m, \phi) = J_m + \int_{\{\underline{x}/\phi(\underline{x})<0\}} \lambda_m(-\Delta v - 1)\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_m v\, ds \tag{11}$$

This implies that the Fréchet derivative of L_m with respect to any small regular perturbation $\tilde{v} : \Omega \setminus \overline{\Omega}_{obs} \to \mathbb{R}$, satisfying $\tilde{v} = 0$ on $\Gamma_{top} \cup \Gamma_{bottom}$ and $\frac{\partial \tilde{v}}{\partial n} = 0$ on $\Gamma_{left} \cup \Gamma_{right}$ in order to match the boundary conditions of v, should be equal to zero at the minimum. Let the perturbation of L_m with respect to v

$$L_m(v + \tilde{v}, \lambda_m, \mu_m, \phi) - L_m(v, \lambda_m, \mu_m, \phi) = \delta L_m(v, \lambda_m, \mu_m, \phi) \cdot \tilde{v} + \text{h.o.t.},$$

we obtain

$$\begin{aligned}\delta L_{m,\tilde{v}} = \delta L_m(v, \lambda_m, \mu_m, \phi) \cdot \tilde{v} &= \int_{\{\underline{x}/\phi(\underline{x})<0\}} \tilde{v}(v-u)\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})<0\}} \lambda_m(-\Delta\tilde{v})\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_m \tilde{v}\, ds \\ &= \int_{\{\underline{x}/\phi(\underline{x})<0\}} \tilde{v}(v-u)\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})<0\}} (-\Delta\lambda_m)\tilde{v}\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}\cup\partial\Omega} \frac{\partial\lambda_m}{\partial n}\tilde{v}\, ds - \int_{\{\underline{x}/\phi(\underline{x})=0\}\cup\partial\Omega} \frac{\partial\tilde{v}}{\partial n}\lambda_m\, ds + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_m\tilde{v}\, ds,\end{aligned}$$

where $\underline{n}$ on $\partial\Omega_{obs}$ is oriented towards the interior of the obstacle. The cancellation of all integrals, given the boundary conditions for $\tilde{v}$ on $\partial\Omega$ written above, leads to the following equations for λ_m defining the adjoint problem of (2). This also provides a relation between λ_m and μ_m on $\partial\Omega_{obs}$, as $-\lambda_m\frac{\partial\tilde{v}}{\partial n} + \tilde{v}\frac{\partial\lambda_m}{\partial n} + \mu_m\tilde{v} = 0$ there. We have

$$\begin{cases} -\Delta\lambda_m = u - v \quad \text{in } \Omega \setminus \overline{\Omega}_{obs}, \\ \frac{\partial\lambda_m}{\partial n} = 0 \quad \text{on } \Gamma_{left} \cup \Gamma_{right}, \\ \lambda_m = 0 \quad \text{on } \Gamma_{top} \cup \Gamma_{bottom}, \\ \lambda_m = 0, \quad \mu_m = -\frac{\partial\lambda_m}{\partial n} \quad \text{on } \partial\Omega_{obs}. \end{cases} \tag{12}$$

It is worth noticing that the same penalization method, GMRES, and gradient improvement are applied to solve both the direct and adjoint problem.

2.3.2. Shape derivative

In order to apply a gradient method for the minimization of our model problem, we introduce the notion of shape gradient. Typically, this is about computing the Fréchet derivative of the functional $L_m = L_m(v, \lambda_m, \mu_m, \phi)$ with respect to a small displacement of the shape according to a vector field $\underline{V}$. This vector field will be then chosen so that it coincides with a descent direction. The formal context is the following: from a velocity field $\underline{V}$, and a given obstacle $\overline{\Omega}_{obs}$, a class of perturbed domains $\Omega_t = \Omega \setminus \overline{\Omega}_{obs}(t)$ can be defined, such that:

$$\Omega_t(\underline{V}) = \{T_t(\underline{x}, \underline{V}),\ \underline{x} \in \Omega \setminus \Omega_{obs}\}, \tag{13}$$

where $T_t(\underline{x},\underline{V})$ is the solution of $\dot{\underline{x}} = \underline{V}(\underline{x})$ at time t with initial value given by $\underline{x}$. We note ϕ_t the distance function related to the transformed domain $\Omega_t(\underline{V})$. The shape gradient of L_m in the direction of $\underline{V}$ is then computed thanks to the expression

$$\delta L_m(v, \lambda_m, \mu_m, \phi)[\underline{V}] = \left[\frac{d}{dt}(L_m(v, \lambda_m, \mu_m, \phi_t))\right]_{t=0}. \tag{14}$$

The Hadamard's theorem proves that the shape derivative depends in fact only on the normal velocity on the edge of the obstacle $\partial\Omega_{obs}$. Thanks to the distance level-set representation of the boundary, the perturbed geometry can be simply expressed by $\phi + \tilde{\phi}$, where $\tilde{\phi}$ is the normal displacement of the border. Because of the signed distance function definition, $\tilde{\phi} > 0$ amounts to a displacement in the direction of the normal coming out of Ω_{obs}. For our problem, taking into account the boundary conditions for the direct and adjoint solutions on the interface, this leads to:

$$\delta L_{m,\tilde{\phi}} = -\int_{\{\underline{x}/\phi(\underline{x})=0\}} \frac{(v-u)^2}{2}\tilde{\phi}\,ds - \int_{\{\underline{x}/\phi(\underline{x})=0\}} \lambda_m(-\Delta v - 1)\tilde{\phi}\,ds - \int_{\{\underline{x}/\phi(\underline{x})=0\}} \frac{\partial(\mu_m v)}{\partial n}\tilde{\phi}\,ds - \int_{\{\underline{x}/\phi(\underline{x})=0\}} \chi\mu_m v\tilde{\phi}\,ds$$
$$= -\int_{\{\underline{x}/\phi(\underline{x})=0\}} \left(\frac{(v-u)^2}{2} - \frac{\partial\lambda_m}{\partial n}\frac{\partial v}{\partial n}\right)\tilde{\phi}\,ds,$$

where χ is the curvature. Since the solution of the direct problem vanishes on $\partial\Omega_{obs}$ and the field equation is satisfied at each optimization step, we obtain the last equality in the equation above. We are interested in finding an expression of the shape perturbation such that the Lagrangian cost function decreases. Therefore, the optimal choice for the scalar $\tilde{\phi}$ is:

$$\tilde{\phi} = \frac{(v-u)^2}{2} - \frac{\partial v}{\partial n}\frac{\partial\lambda_m}{\partial n} \quad \text{on } \partial\Omega_{obs}. \tag{15}$$

In practice, close to the minimum, the first term of the sum is much smaller than the second one. Moreover, this second-order contribution has a fixed negative sign and hence it always aims at increasing the size of the obstacle. In some configurations this can spoil the convergence rate and therefore in the following numerical demonstrations we neglect it. In fact, this term drives the solution towards the trivial minimizer $\Omega\backslash\Omega_{obs} = \emptyset$.

2.3.3. Algorithm and preliminary results

The gradient descent method for shape reconstruction, based on level-set and penalization, is detailed in Algorithm 1. In order to perturb the geometry, we do not advect the level-set function representing the geometry by the Hamilton–Jacobi equation (5) with $-\frac{\partial v}{\partial n}\frac{\partial\lambda_m}{\partial n}$ as the normal velocity V_n. Instead, at every optimization iteration we ensure that ϕ is the signed distance function from the interface, thus having the property $|\nabla\phi| = 1$. Therefore, moving its zero level-set in the direction of the normal amounts to adding the displacement $\tilde{\phi} = -\frac{\partial v}{\partial n}\frac{\partial\lambda_m}{\partial n}$ to ϕ. Even if ϕ is a distance function, nothing ensures that the perturbed level-set function $\phi + \tilde{\phi}$ is a distance function as well. However, a fast-marching method performs a redistanciation at the beginning of each optimization step so that $|\nabla(\phi + \tilde{\phi})|$ is reinitialized to 1.

Algorithm 1. Gradient method for the model inverse problem

Initialization of a level-set function ϕ for an arbitrary starting shape
for $n = 1$ to *Niter_max* **do**
  Renormalization and redistanciation by fast-marching method of the level-set function ϕ, i.e. $|\nabla\phi| = 1$
  Solve the Direct Poisson problem for v (Penalization and GMRES) and improve $\frac{\partial v}{\partial n}$ on $\partial\Omega_{obs}$ by performing three steps of the gradient prolongation
  Solve the Adjoint Poisson problem for λ_m Problem (Penalization and GMRES) and improve $\frac{\partial\lambda_m}{\partial n}$ on $\partial\Omega_{obs}$ by performing three steps of the gradient prolongation
  Compute the perturbation $\tilde{\phi} = \frac{-\partial v}{\partial n}\frac{\partial\lambda_m}{\partial n}$ from the shape derivative
  Update the level-set function $\phi \leftarrow \phi + \tilde{\phi}$
end for

A first application of this algorithm is presented below. In a rectangular domain $\Omega = (0,10) \times (0,6)$, let u be the solution of (2) associated to an optimal obstacle Ω^*_{obs} designed as the union of two circles, both of radius 1, respectively centered at (3.5,5) and (6.5,5). The values of the field u in Ω are assumed to be known at every mesh point of a cartesian grid of 101 points in each direction. We attempt to restore those two centered circles on the same grid, starting from different initial configurations. In Fig. 3, snapshots illustrate the evolution of two different initial geometries. On the left, a rectangle embedding the optimal shape (case A); on the right, two rectangles that do not enclose the circles (case B). Initial, intermediate and final snapshots are presented. In Fig. 4, the evolution of the functional in terms of the optimization steps is presented. The maximum allowed displacement of the interface between two consecutive minimization iterations is $\frac{1}{2}\min(\Delta x, \Delta y)$.

Contrasted conclusions can be drawn from this example. In both cases, our gradient method has been able to reduce the objective function significantly (here about three orders of magnitude), with corresponding results close to the optimal shape. Also, topological changes such as boundary separation can be naturally performed thanks to the level-set method. Finally, large scale displacements of the initial obstacle can be achieved as well, as shown in the right column of Fig. 3. Nevertheless, due to the fact that we have picked the L^2 norm to measure the discrepancy between current and objective solutions, nothing guarantees the smoothness of the interface. Indeed, rather important oscillations tend to be created on the edge of the obstacle while getting closer and closer to the optimal field u. While this type of oscillations often occurs when using fixed step gradient methods for optimization, reducing the amplitude of the shape gradient near the minimum when the functional increases proves ineffective in our case. One reason for this is that we discretize the continuous adjoint equation, and hence the gradient computed is not consistent with the discrete problem on a finite mesh. Besides, close to the optimal solution, the error made by computing the discrete cost function becomes of the order of the

variations induced by shape perturbations. The objective function cannot then be used to effectively modify the optimization step.

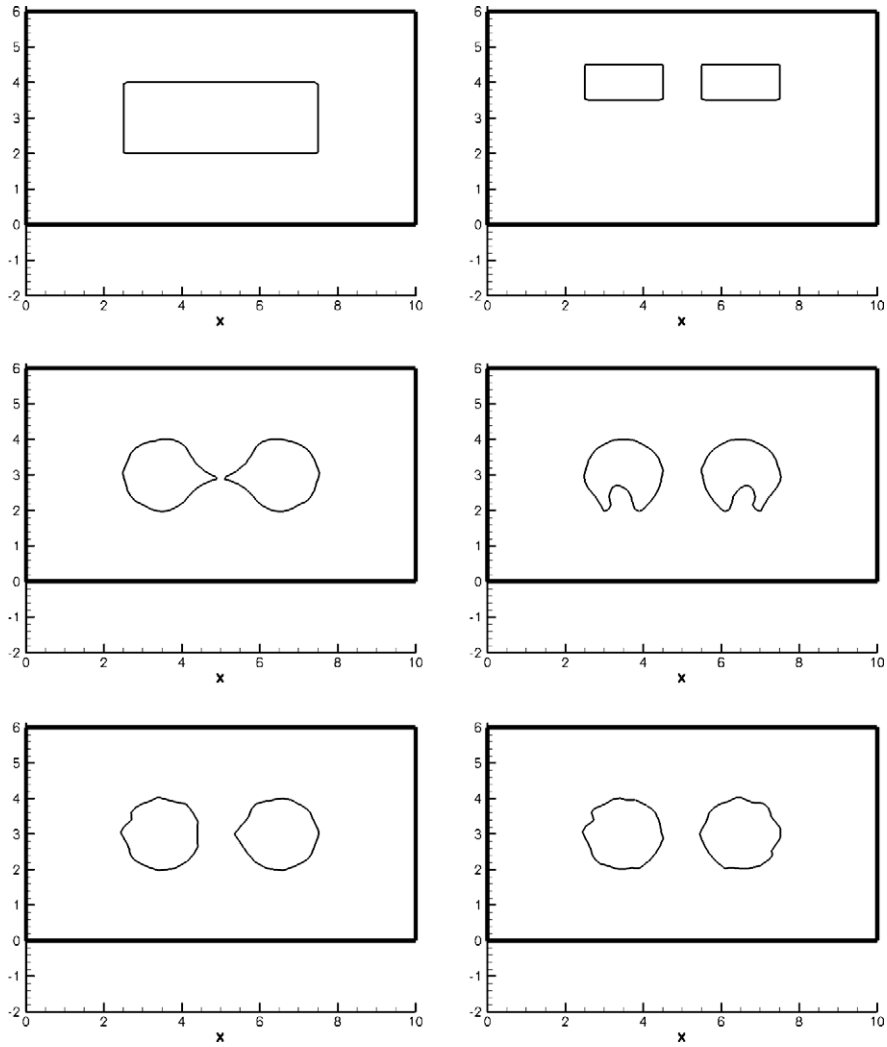

Fig. 3. Retrieving two circles from topologically different initial conditions. Evolution of the zero isoline of the level-set function for cases A (left) and B (right). From top to bottom optimization steps 0, 216, 8964.

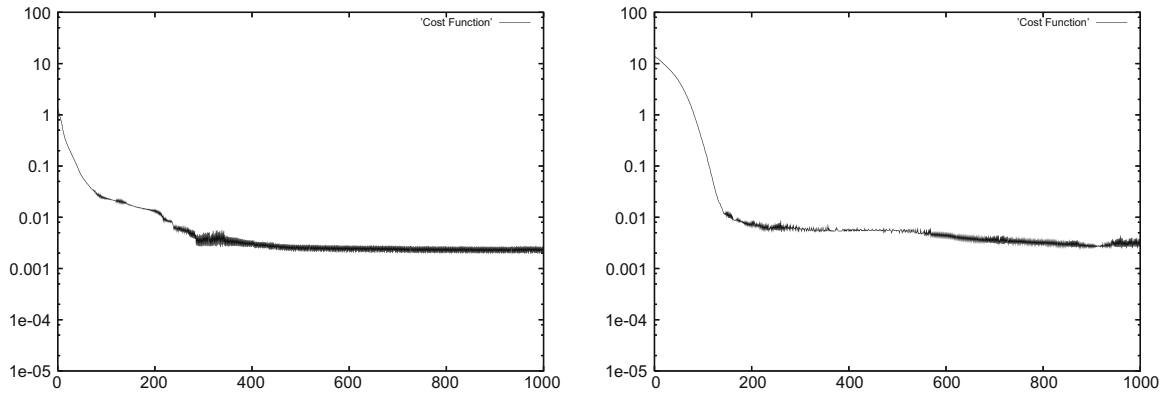

Fig. 4. Retrieving two circles from topologically different initial conditions; $\log(J_m)$ as a function of the number of iterations. Left: case A; right: case B.

2.4. Gradient preconditioning

2.4.1. Choice of the objective function

The main objective of the present section is to investigate the effects of a different definition of the cost function in order to precondition the shape gradient used in the descent algorithm. Similar ideas were presented in [15]. Basically, we intend to avoid the boundary oscillations at the minimum while allowing large overall displacements when the obstacle is located far from its optimal position. Under appropriate assumptions about the regularity of v and u, let us define the functionals:

$$\begin{aligned} J_s(\phi, v(\phi)) &= \frac{1}{2}\int_{\{\underline{x}/\phi(\underline{x})<0\}} \|\nabla v - \nabla u\|^2 \, d\Omega, \\ J_l(v(\phi)) &= \frac{1}{2}\int_{\{\underline{x}/\phi(\underline{x})<0\}} \|\nabla \psi - \nabla \psi^*\|^2 \, d\Omega \end{aligned} \tag{16}$$

where v is the solution of (2) and ψ solves the associated Poisson problem:

$$\begin{cases} -\Delta\psi = v & \text{in } \Omega \setminus \overline{\Omega}_{obs}, \\ \frac{\partial \psi}{\partial n} = 0 & \text{on } \Gamma_{left} \cup \Gamma_{right}, \\ \psi = 0 & \text{on } \Gamma_{top},\ \Gamma_{bottom}, \text{ and on } \partial\Omega_{obs}. \end{cases} \tag{17}$$

The same equation is satisfied by ψ^*, with u as the right-hand side. Both v and ψ are taken to be zero inside $\overline{\Omega}_{obs}$, so that $\Delta\psi = v = 0$ in the obstacle.

As previously done, we introduce the Lagrange multipliers λ_s, μ_s and λ_l, μ_l, as well as $\lambda_{l,2}$, $\mu_{l,2}$ for the constraints relative to the problem in ψ. We define the following Lagrangians

$$\begin{aligned} L_s(v, \lambda_s, \mu_s, \phi) &= J_s + \int_{\{\underline{x}/\phi(\underline{x})<0\}} \lambda_s(-\Delta v - 1)\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_s v \, ds, \\ L_l(v, \psi, \lambda_l, \mu_l, \phi) &= J_l + \int_{\{\underline{x}/\phi(\underline{x})<0\}} \lambda_l(-\Delta v - 1)\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_l v \, ds + \int_{\{\underline{x}/\phi(\underline{x})<0\}} \lambda_{l,2}(-\Delta\psi - v)\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_{l,2}\psi \, ds \end{aligned} \tag{18}$$

A derivation similar to that detailed in the previous section leads for the first Lagrangian to

$$\begin{aligned} \delta L_{s,\tilde{v}} &= \int_{\{\underline{x}/\phi(\underline{x})<0\}} \nabla\tilde{v}\cdot(\nabla v - \nabla u)\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})<0\}} \lambda_s(-\Delta\tilde{v})\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_s \tilde{v}\, ds \\ &= \int_{\{\underline{x}/\phi(\underline{x})<0\}} -\tilde{v}\Delta(v-u)\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})<0\}} (-\Delta\lambda_s)\tilde{v}\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}\cup\partial\Omega} \tilde{v}\frac{\partial(v-u)}{\partial n}\, ds + \int_{\{\underline{x}/\phi(\underline{x})=0\}\cup\partial\Omega} \frac{\partial\lambda_s}{\partial n}\tilde{v}\, ds \\ &\quad + \int_{\{\underline{x}/\phi(\underline{x})=0\}\cup\partial\Omega} -\frac{\partial\tilde{v}}{\partial n}\lambda_s\, ds + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_s\tilde{v}\, ds. \end{aligned}$$

For the second Lagrangian, we calculate separately the Fréchet derivatives of L_l with respect to ψ and v:

$$\begin{aligned} \delta L_{l,\tilde{\psi}} &= \int_{\{\underline{x}/\phi(\underline{x})<0\}} \nabla\tilde{\psi}\cdot(\nabla\psi - \nabla\psi^*)\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})<0\}} \lambda_{l,2}(-\Delta\tilde{\psi})\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_{l,2}\tilde{\psi}\, ds \\ &= -\int_{\{\underline{x}/\phi(\underline{x})<0\}} \tilde{\psi}\Delta(\psi-\psi^*) + \int_{\{\underline{x}/\phi(\underline{x})<0\}} \tilde{\psi}(-\Delta\lambda_{l,2})\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_{l,2}\tilde{\psi}\, ds + \int_{\{\underline{x}/\phi(\underline{x})=0\}\cup\partial\Omega} \tilde{\psi}\frac{\partial(\psi-\psi^*)}{\partial n}\, ds \\ &\quad + \int_{\{\underline{x}/\phi(\underline{x})=0\}\cup\partial\Omega} \tilde{\psi}\frac{\partial\lambda_{l,2}}{\partial n}\, ds - \int_{\{\underline{x}/\phi(\underline{x})=0\}\cup\partial\Omega} \lambda_{l,2}\frac{\partial\tilde{\psi}}{\partial n}\, ds, \end{aligned}$$

$$\begin{aligned} \delta L_{l,\tilde{v}} &= \int_{\{\underline{x}/\phi(\underline{x})<0\}} \lambda_l(-\Delta\tilde{v})\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_l\tilde{v}\, ds - \int_{\{\underline{x}/\phi(\underline{x})=0\}} \lambda_{l,2}\tilde{v}\, ds \\ &= \int_{\{\underline{x}/\phi(\underline{x})<0\}} \tilde{v}(-\Delta\lambda_l)\, d\Omega - \int_{\{\underline{x}/\phi(\underline{x})<0\}} \lambda_{l,2}\tilde{v}\, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}\cup\partial\Omega} \frac{\partial\lambda_l}{\partial n}\tilde{v}\, ds - \int_{\{\underline{x}/\phi(\underline{x})=0\}\cup\partial\Omega} \frac{\partial\tilde{v}}{\partial n}\lambda_l\, ds + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \mu_l\tilde{v}\, ds. \end{aligned}$$

The cancellation of all terms of $\delta L_{l,\tilde{\psi}}$ leads to:

$$\begin{cases} -\Delta\lambda_{l,2} = \Delta(\psi-\psi^*) & \text{in } \Omega\setminus\overline{\Omega}_{obs}, \\ \frac{\partial\lambda_{l,2}}{\partial n} = 0 & \text{on } \Gamma_{left}\cup\Gamma_{right}, \\ \lambda_{l,2} = 0 & \text{on } \Gamma_{top}\cup\Gamma_{bottom}, \\ \lambda_{l,2} = 0, \quad \mu_{l,2} = -\frac{\partial\lambda_{l,2}}{\partial n} - \frac{\partial(\psi-\psi^*)}{\partial n} & \text{on } \partial\Omega_{obs}. \end{cases}$$

Instead of solving the above equation for $\lambda_{l,2}$ we take $\lambda_{l,2} = \psi^* - \psi$ on $\Omega \backslash \Omega_{obs}$, thus implying that $\mu_{l,2} = 0$ on $\partial\Omega_{obs}$. This is an approximation that cancels as we reach the optimal shape, but it allows us to avoid the introduction of the Lagrange multiplier for the constitutive equation of ψ in (18). Finally, taking $\delta L_{s,\tilde{v}} = 0$ and $\delta L_{l,\tilde{v}} = 0$, the following adjoint problems are obtained:

$$\begin{cases} -\Delta\lambda_l = \lambda_{l,2} = \psi^* - \psi & \text{in } \Omega \setminus \overline{\Omega}_{obs}, \\ \frac{\partial\lambda_l}{\partial n} = 0 & \text{on } \Gamma_{left} \cup \Gamma_{right}, \\ \lambda_l = 0 & \text{on } \Gamma_{top} \cup \Gamma_{bottom}, \\ -\lambda_l = 0, \quad \mu_l = -\frac{\partial\lambda_l}{\partial n} & \text{on } \partial\Omega_{obs}, \end{cases} \qquad \begin{cases} -\Delta\lambda_s = -\Delta(u - v) & \text{in } \Omega \setminus \overline{\Omega}_{obs}, \\ \frac{\partial\lambda_s}{\partial n} = 0 & \text{on } \Gamma_{left} \cup \Gamma_{right}, \\ \lambda_s = 0 & \text{on } \Gamma_{top} \cup \Gamma_{bottom}, \\ \lambda_s = 0, \quad \mu_s = -\frac{\partial\lambda_s}{\partial n} - \frac{\partial(v-u)}{\partial n} & \text{on } \partial\Omega_{obs}. \end{cases} \tag{19}$$

The same numerical method used before still applies for the resolution of the equations above. Moreover, there is no other additional computing cost than the computation of ψ from v in the case of L_l minimization, since ψ^* is calculated only once outside of the steepest descent loop.

The normal perturbations of the level-set function ϕ of respective amplitudes $\tilde{\phi}_s$ and $\tilde{\phi}_l$, are chosen so that the Fréchet derivative of the appropriate functional along this direction is negative. Following the same steps as before, this results in taking

$$\tilde{\phi}_s = -\frac{\partial v}{\partial n}\left(\frac{\partial\lambda_s}{\partial n} + \frac{\partial(v-u)}{\partial n}\right) + \frac{1}{2}\|\nabla v - \nabla u\|^2, \tag{20}$$

$$\tilde{\phi}_l = -\frac{\partial v}{\partial n}\frac{\partial\lambda_l}{\partial n} + \frac{1}{2}\|\nabla\psi - \nabla\psi^*\|^2, \tag{21}$$

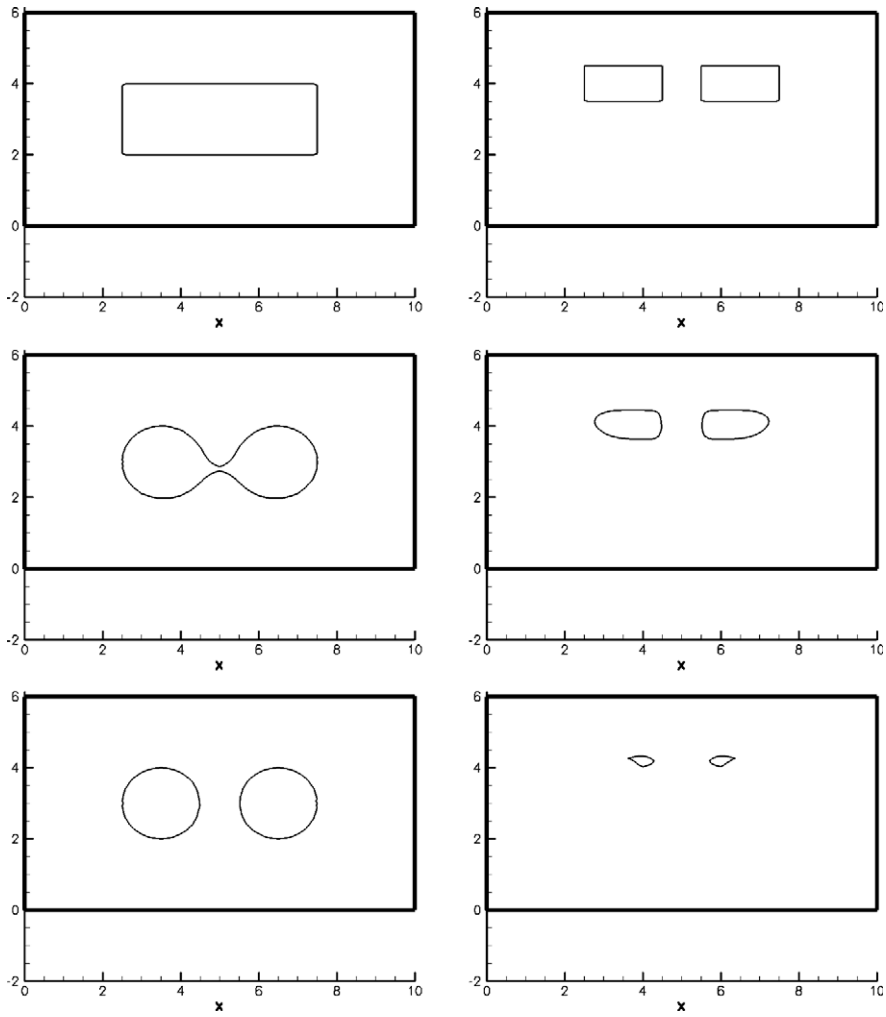

Fig. 5. Minimizing J_s instead of J_m. Evolution of the zero isoline of the level-set function for cases A (left) and B (right) as a function of the number of iterations. Obstacles from top to bottom at iterations: 0, 108, 972 (left) and 0, 36, 108 (right).

The arguments invoked for the shape gradient of J_m remain valid here to justify why the fixed-sign contributions (right terms) that appear in both $\tilde{\phi}_s$ and $\tilde{\phi}_l$ are neglected in the computations. We consider the same initial conditions and optimal solutions as for the test cases of Fig. 3. This time, instead of J_m, the cost function J_s is minimized. In Fig. 5, left column, it can be observed that a much smoother interface is obtained next to the minimum. At the same time, as also shown on the left column, a better convergence rate is achieved since the topological change now takes place after a smaller number of iterations and lower residuals are attained for given number of optimization steps, as seen in Fig. 6. However, despite improved control of boundary oscillations when choosing the L^2 norm of the gradient, one has to be cautious when using such functionals, as shown on the right column of the same figure. Actually, while a modification of the objective function allows us to find new paths of descent, the minimization of J_s does not always imply the minimization of J_m. As a consequence, what happens is that selecting J_s proves to be valuable only when treating small scale displacements (hence the subscript "s"). In other words, the minimizations of J_s and J_m coincide if the optimization process starts from an initial shape close to the optimal one. On the other hand, if one aims at handling macroscopic motion, reducing J_s appears to be counter-productive. In fact, trying to recenter the two rectangles with J_s, leads to a divergence of J_m and to a fading obstacle (Fig. 5, right column). We also performed the same test cases using J_l as the functional to minimize. The results, only partially shown here, presented a faster recentering of the two initial rectangles than the one in Fig. 3, right column, while completing topological separation for the simple rectangle more slowly. As a matter of fact, the interest of using J_l as the cost function is that it induces more uniform large scale displacements far from the solution. Indeed, as seen on Figs. 7 and 8, the boundary oscillations that are caused when close to retrieving the optimal body are more important than the ones observed when using J_s, and similar to the ones obtained with J_m.

The descents in the directions of $\tilde{\phi}_l$, $\tilde{\phi}_m$ and $\tilde{\phi}_s$ display a positive contribution on the optimization, either in terms of smoothness of the interface or in their capability of handling a specific displacement scale. One example of how to improve convergence thanks to the three functional definitions is based on ideas similar to multilevel optimization [21]. In the next examples, for a given multilevel cycle, the first two optimization iterations are dedicated to the minimization of J_l, and the next 15 are alternately performed three by three on the middle and small scales. Figs. 9 and 10 present the results relative to this multilevel algorithm. From a comparison with Fig. 5, we note that the retrieval of two smooth circles embedded inside an initial centered rectangle, that was obtained only with the small scales, is kept using all of them. Moreover, in this specific case, a similar mean accuracy is achieved as to what concerns J_m, since the oscillations shown in the residual plot are mainly due to a coarse numerical evaluation of the objective function. On the other hand, the results starting from the reunion of the off-centered rectangles are drastically improved.

The divergence of the medium scale residual, which was noticed when exclusively minimizing J_s, was avoided. So have the boundary fluctuations that occurred close to the minimum relative to J_m. While smoothness is maintained, the different paths of descent allow macroscopic displacements, which translates into a successful recentering of the two rectangles. However, the stagnation of J_m (see Fig. 10, right) does not allow to recover the optimal solution.

2.4.2. Scanning and topological derivative

In this section, we provide two additional examples of how penalization and level-sets can be combined to solve inverse problems. In the first example, we aim at finding alternate descent paths when close to local minima. In the same spirit of escaping local minima, the subsequent section provides an example of coupling between the shape and topological derivatives, as done in structural optimization in [9].

2.4.2.1. Scanning method. In the test case of Fig. 9, right column, we can observe that the minimum of the cost function J_m is not attained, i.e., the solution gets stuck at a non-optimal shape. We propose an association of our multilevel gradient algorithm with the "scanning" method presented hereafter. The idea is to change the orientation of the obstacle with respect to

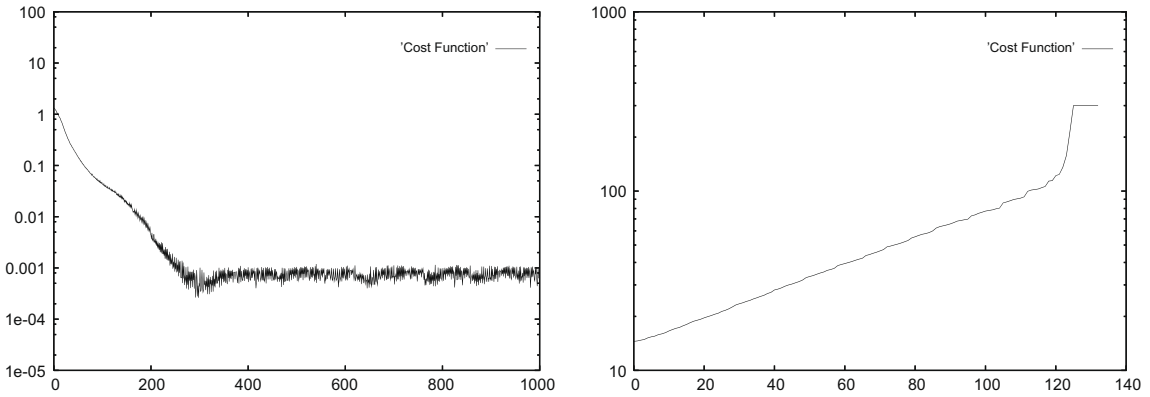

Fig. 6. Minimizing J_s instead of J_m. Evolution of the error in terms of $\log(J_m)$ as a function of the number of iterations for cases A (left) and B (right).

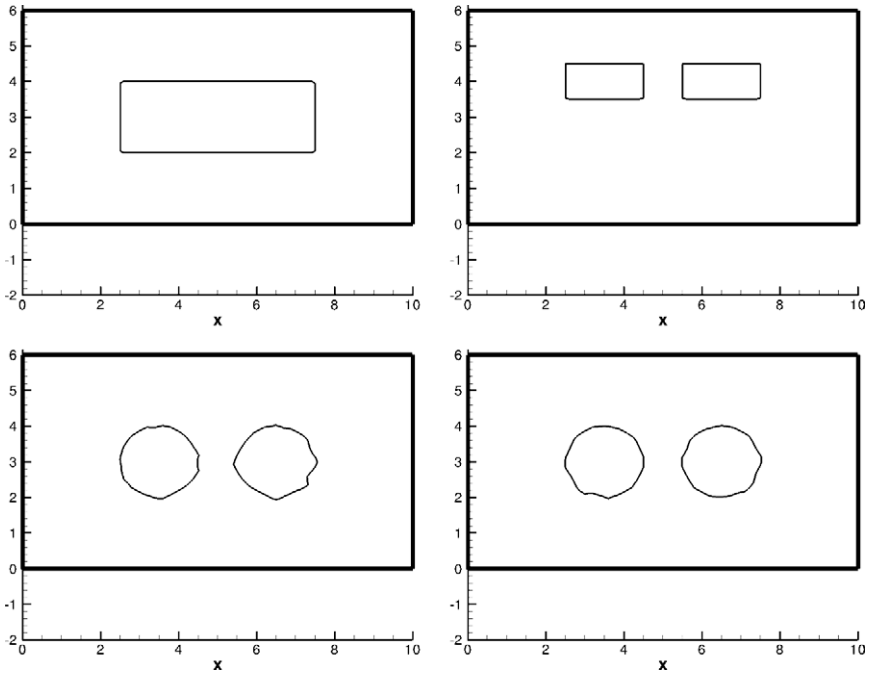

Fig. 7. Minimizing J_l instead of J_m. Initial (iteration 0) and final (iteration 9000) zero isolines of the level-set function for cases A (left) and B (right).

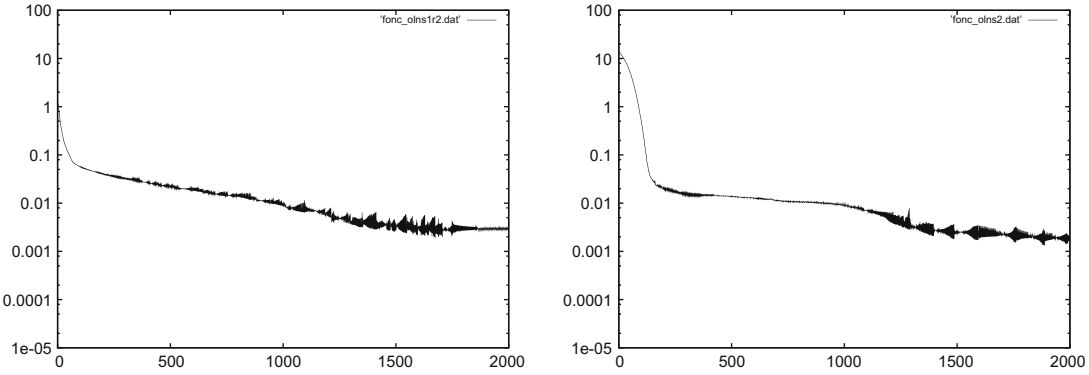

Fig. 8. Minimizing J_l instead of J_m. Evolution of the error in terms of $\log(J_m)$ as a function of the number of iterations for cases A (left) and B (right).

the boundaries during the minimization process. After completing a number of successive multilevel cycles at a given angular position of the body ζ_0, corresponding to N minimization steps, the distance function ϕ itself is rotated around (x_c, y_c) of an angle $\delta\zeta$. This motion is carried out by defining a velocity vector field $\underline{V}_{rot}$:

$$\underline{V}_{rot}(x,y) = \begin{bmatrix} (y_0 - y)\delta\zeta \\ (x - x_0)\delta\zeta \end{bmatrix}, \tag{22}$$

where $\dot{\theta} = \delta\zeta$ is the angular speed. Then, the WENO5 scheme is used to solve up to $s = 1$:

$$\begin{cases} \frac{\partial\phi}{\partial s} + \underline{V}_{rot} \cdot \nabla\phi = 0, \\ \phi(\underline{x}, s = 0) = \phi_{\zeta_0}. \end{cases} \tag{23}$$

The objective solution u of the Poisson problem (2) will be considered known at every intermediate obstacle position ζ_i of interest. This of course implies that additional data is included in the resolution of the inverse problem. However, information also happens to be retrieved under several angles in many real life applications, from medical imaging to non-destructive control of materials to name a few. In such cases, the modification of the direction along which the obstacle is seen is performed by changing the localization of the boundary conditions. From a computational standpoint, revolving the rectangular domain Ω around its center to account for this change of boundary conditions is not very convenient. This is why the

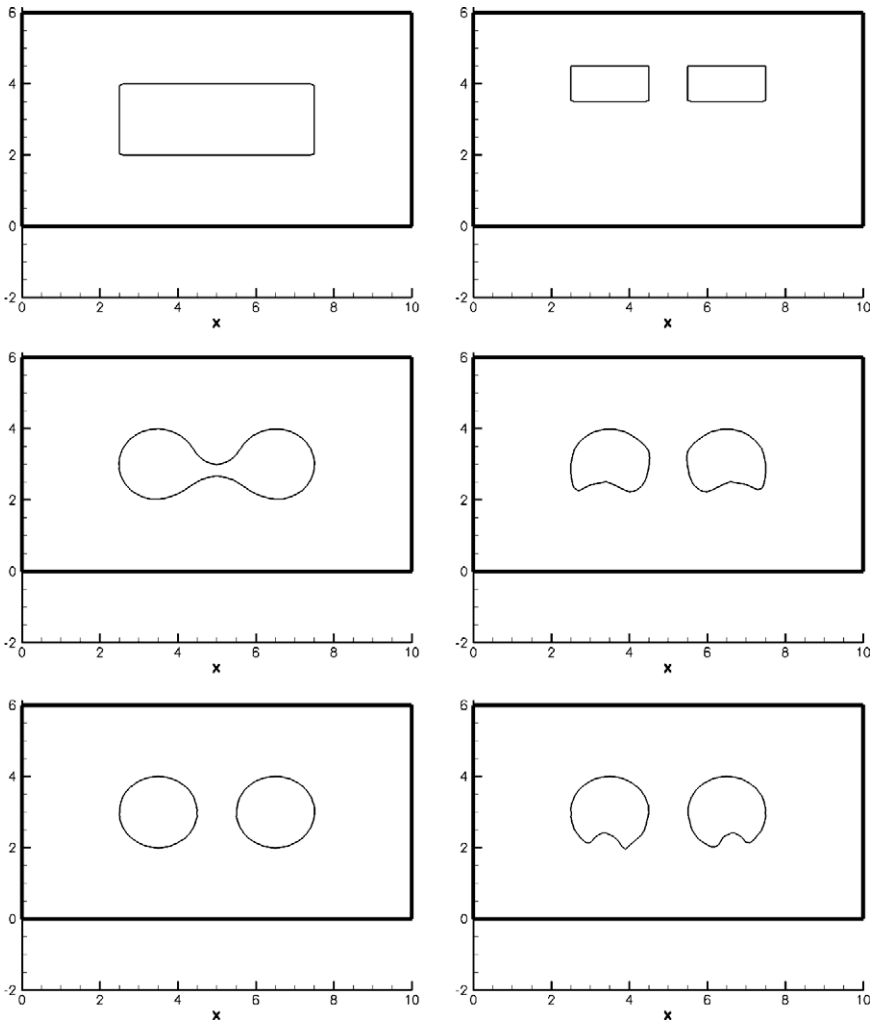

Fig. 9. Multiscale test cases. Evolution of the zero isoline of the level-set function for cases A (left) and B (right) as a function of the number of iterations. Obstacles from top to bottom after cycles: 0, 6, 60 and 0, 18, 60.

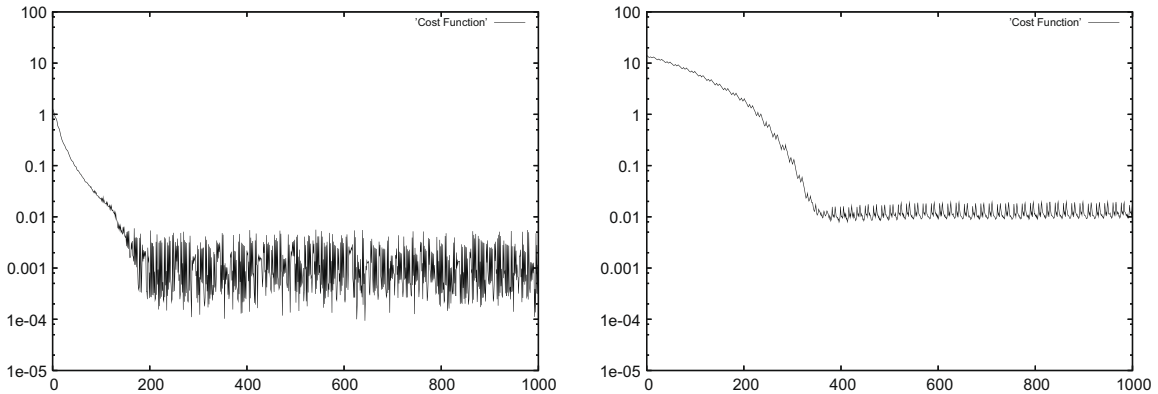

Fig. 10. Multiscale test cases, 1 cycle = 17 optimization steps. $\log(J_m)$ as a function of the number of iterations for cases A (left) and B (right).

most is made out of the versatility of the level-set method, as we instead choose to rotate the distance function itself in order to achieve a similar goal. Fig. 11 provides the final results for the two test cases used until now. After six successive multi-level cycles, i.e. 102 iterations, $\overline{\Omega}_{obs}$ is turned of $\frac{\pi}{4}$ anti-clockwise. The jumps observed on the plot of the residual J_m, Fig. 12, are

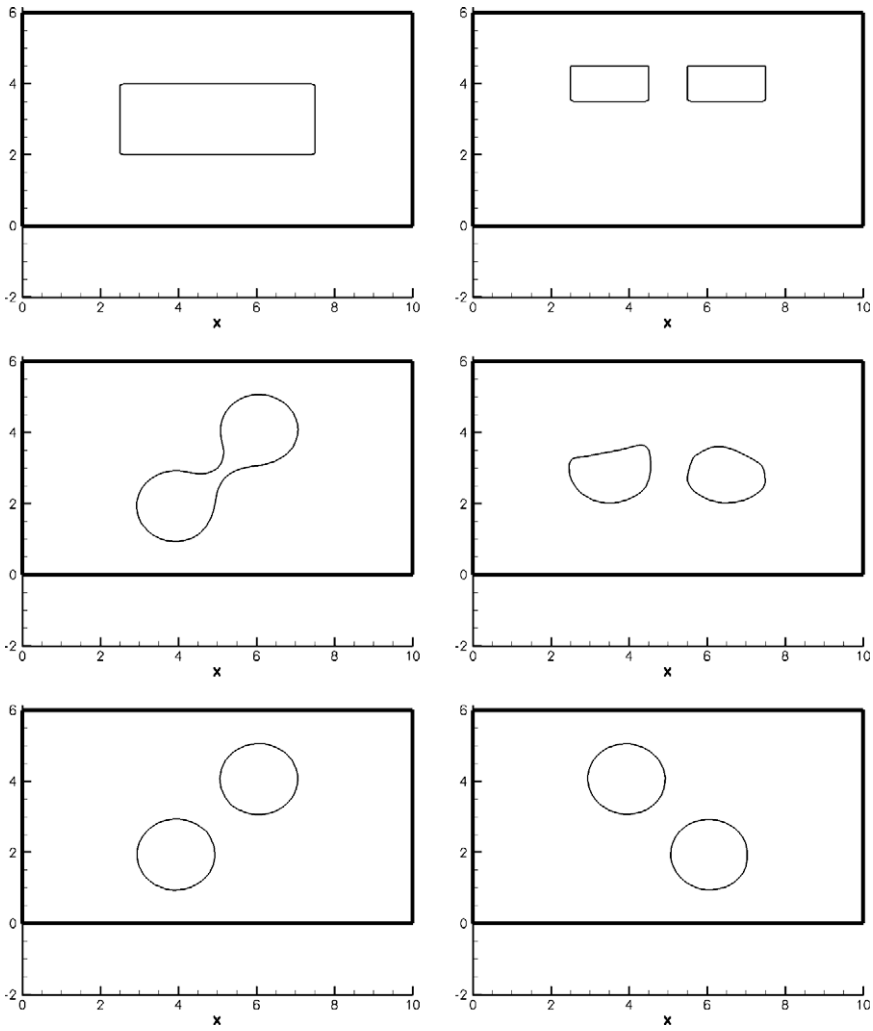

Fig. 11. Reconstruction of two unit circles with multiscale and scanning. Evolution of the zero isoline of the level-set function for cases A (left) and B (right) as a function of the number of iterations. Obstacles from top to bottom at iterations: 0, 102, 170 for case A and 0, 408, 714 for case B.

due to the rotation that induces sudden modifications on both the current and objective fields at these specific iterations, thus increasing the discrepancy between v and u.

Scanning actually helps retrieving an optimal body by escaping local minima or very flat functional regions as for case B above. In order to show the effectiveness of our method, a much more complicated case in terms of the objective shape is presented in Fig. 13. The unknown geometry to be retrieved is the acronym "MC2"; this obstacle is located at the center of a domain $[0,10] \times [0,10]$ which is meshed with 101 points in each direction. For this case it was impossible to get even close to the solution by the usual gradient descent.

2.4.2.2. Topological and shape gradients' coupling. This section aims at studying the coupling of the shape gradient with the topological gradient [9] in order to enlarge the search space of the descent method. The main idea of this last method relies in testing the optimality of the domain Ω with respect to perturbations induced by the introduction of infinitesimal balls that locally modify the sign of ϕ. The functional minimum is therefore relative to a larger class of perturbations and topological changes without external boundary deformation may occur. In order to take the entire computational domain into account for topological evolution, we start by writing $J_{m,total}(\phi, v(\phi))$ as the sum of the former functional for the medium scales $J_{m,1}(\phi, v(\phi)) = J_m(\phi, v(\phi))$ and a complementary one $J_{m,2}(\phi)$ just taking into account the obstacle, where we assume that $v = 0$:

$$\begin{cases} J_{m,total}(\phi, v(\phi)) = J_{m,1}(\phi, v(\phi)) + J_{m,2}(\phi), \\ J_{m,1}(\phi, v(\phi)) = \frac{1}{2} \int_{\{\underline{x}/\phi(\underline{x})<0\}} |v-u|^2 \, d\Omega, \\ J_{m,2}(\phi) = \frac{1}{2} \int_{\{\underline{x}/\phi(\underline{x})\geqslant 0\}} |v-u|^2 \, d\Omega. \end{cases} \tag{24}$$

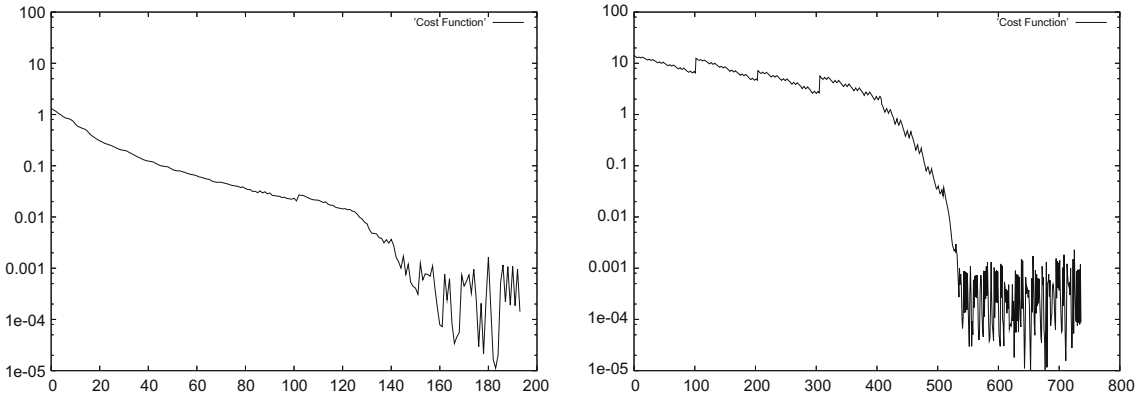

Fig. 12. Reconstruction of two unit circles with multiscale and scanning. $\log(J_m)$ as a function the number of iterations. 1 cycle = 17 optimization steps. Rotation of $\frac{\pi}{4}$ anti-clockwise every six cycles. Left: case A; right: case B.

If $B_{\rho,\underline{x}} = \{\underline{y} \in \mathbb{R}^2,\ |\underline{x} - \underline{y}| < \rho\}$ denotes the open ball of radius ρ centered at $\underline{x}$ and $|\cdot|$ is the Lebesgue measure, the topological derivatives of each term of the sum are then respectively defined by:

$$d_T J_{m,1}(\underline{x}) = \lim_{\rho \to 0} \frac{J_{m,1}(\phi_{1,\rho,\underline{x}}, v(\phi_{1,\rho,\underline{x}})) - J_{m,1}(\phi, v(\phi))}{|B_{\rho,\underline{x}}|} \quad \text{if } \underline{x} \in \Omega \setminus \overline{\Omega}_{obs}, \tag{25}$$

$$d_T J_{m,2}(\underline{x}) = \lim_{\rho \to 0} \frac{J_{m,2}(\phi_{2,\rho,\underline{x}}) - J_{m,2}(\phi)}{|B_{\rho,\underline{x}}|} \quad \text{if } \underline{x} \in \overline{\Omega}_{obs}, \tag{26}$$

where $\phi_{1,\rho,\underline{x}} < 0$ in $\Omega_{1,\rho,\underline{x}} = (\Omega \setminus \overline{\Omega}_{obs}) \setminus \overline{B_{\rho,\underline{x}}}$ if $\underline{x} \in \Omega \setminus \overline{\Omega}_{obs}$ and $\phi_{2,\rho,\underline{x}} \geqslant 0$ in $\Omega_{2,\rho,\underline{x}} = \overline{\Omega}_{obs} \setminus \overline{B_{\rho,\underline{x}}}$ if $\underline{x} \in \overline{\Omega}_{obs}$.

Following [16], the derivations of both topological gradients (25) and (26) lead to

$$d_T J_{m,total}(\underline{x}) = \begin{cases} -\lambda(\underline{x}) & \text{if } \underline{x} \in \Omega_{obs}, \\ \lambda(\underline{x}) & \text{if } \underline{x} \in \Omega \setminus \overline{\Omega}_{obs}, \end{cases} \tag{27}$$

where λ is the solution of the adjoint problem (12) on $\Omega \setminus \overline{\Omega}_{obs}$. In $\overline{\Omega}_{obs}$, λ satisfies

$$\begin{cases} -\Delta\lambda = u - v & \text{in } \Omega_{obs}, \\ \lambda = 0 & \text{on } \partial\Omega_{obs}. \end{cases} \tag{28}$$

The major conclusion to draw about (27) is that introducing the topological derivative in our algorithm does not require any additional computational cost. As a matter of fact, it is given by the value of the adjoint solution which was already needed in order to compute the shape gradient.

The whole algorithm, taking into account the multiscale, scanning, and topological derivative approaches, is summarized in Algorithm 2.

Algorithm 2. Full iterative algorithm for the inverse problem

Initialization of a level-set function ϕ for an arbitrary starting shape
for $n = 1$ to *Niter_max* **do**
  Renormalization and fast-marching method of the level-set function ϕ.
  If N consecutive optimization iterations have already been computed at a given observation angle ζ, rotate ϕ of an angle $\delta\zeta$.
  According to the scale related to n, solve the adequate Direct problem for v and improve its normal gradient. Solve the corresponding Adjoint problem for λ and improve its normal gradient.
  Compute the shape derivative in a zone close to the interface.
  If n is the first iteration corresponding to the medium scales at a new angle, nucleate a hole where λ has the greatest negative value further from the border.
  Update the level-set function.
end for

Fig. 14 introduces a test reconstruction obtained by coupling the topological and shape derivatives according to Algorithm 2. Starting the optimization from a rectangle centered in a domain $[0,10] \times [0,6]$ on a 101×101 mesh, the obstacle to be retrieved is a ring built by subtracting a disc of radius $R_{int} = 1$ from another one of radius $R_{ext} = 2$, both centered at (5,3). The method is able to retrieve the donut shape sought, whereas using the shape gradient, it was only possible to recover the outer interface.

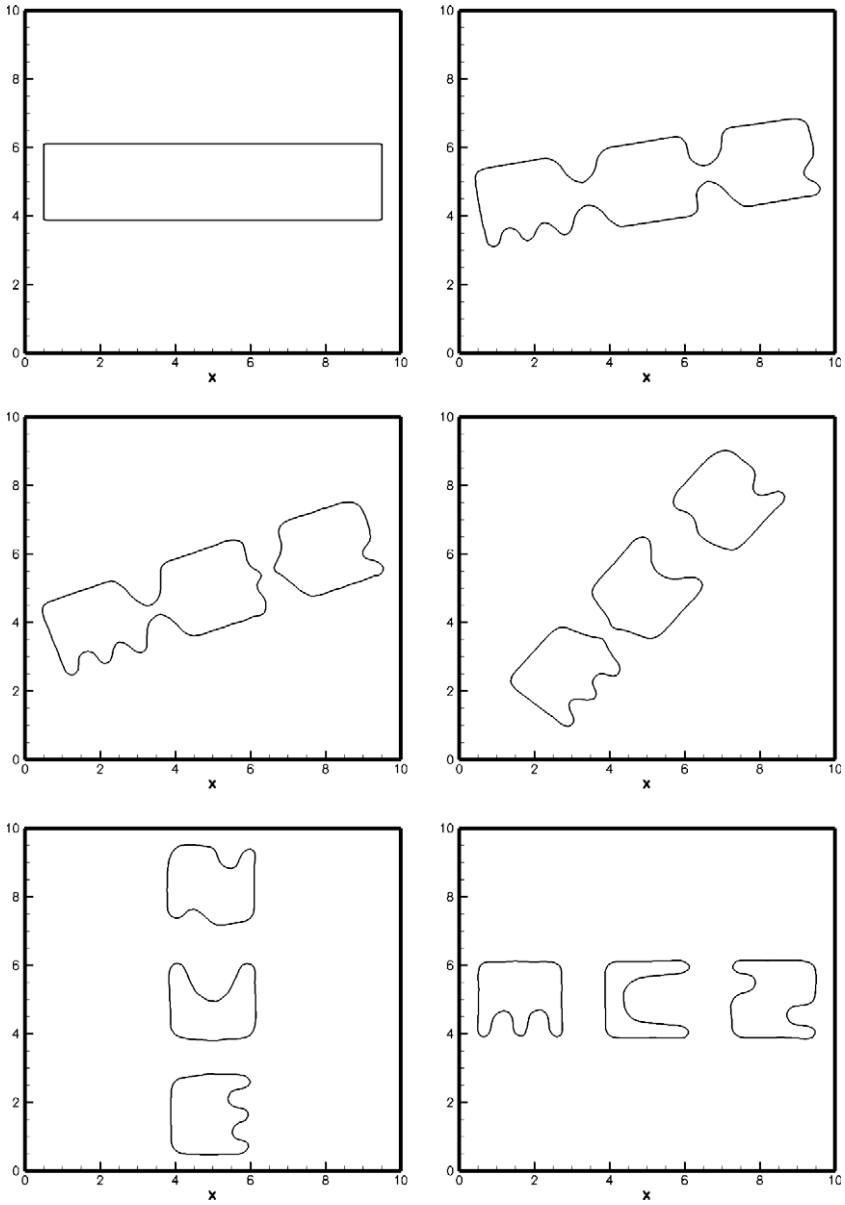

Fig. 13. Reconstruction of the acronym MC2 with small scales and scanning. Evolution of zero isoline of the level-set function as a function of the number of iterations. Every 60 iterations, rotation of $\frac{\pi}{18}$ anti-clockwise from 0 to $\frac{\pi}{2}$, and then back clockwise to 0. Obstacles from top left to bottom right at iterations: 0, 60, 120, 300, 540, 1080.

As a partial conclusion, our iterative approach mixing level-set and penalization, refined by a combination of additional numerical tools, has proven to be efficient for solving model inverse problems. By combining the improvement of the solution gradient next to the penalized obstacle with multilevel optimization and scanning, good accuracy is achieved on the retrieved obstacle interfaces. Several practical applications can ensue from this academical study. An immediate one, in view of the PDE governing the direct problem, is of course electric tomography.

3. Optimal shape design for Stokes flows

3.1. Setting of the problem

Inside a two-dimensional rectangular domain Ω similar to that of Fig. 1, we consider a viscous incompressible flow satisfying the Stokes equations past an obstacle $\overline{\Omega}_{obs}$ on which the no-slip condition is enforced. The goal is to find a configuration of this body that minimizes the drag, while preserving the initial geometry area during the optimization process.

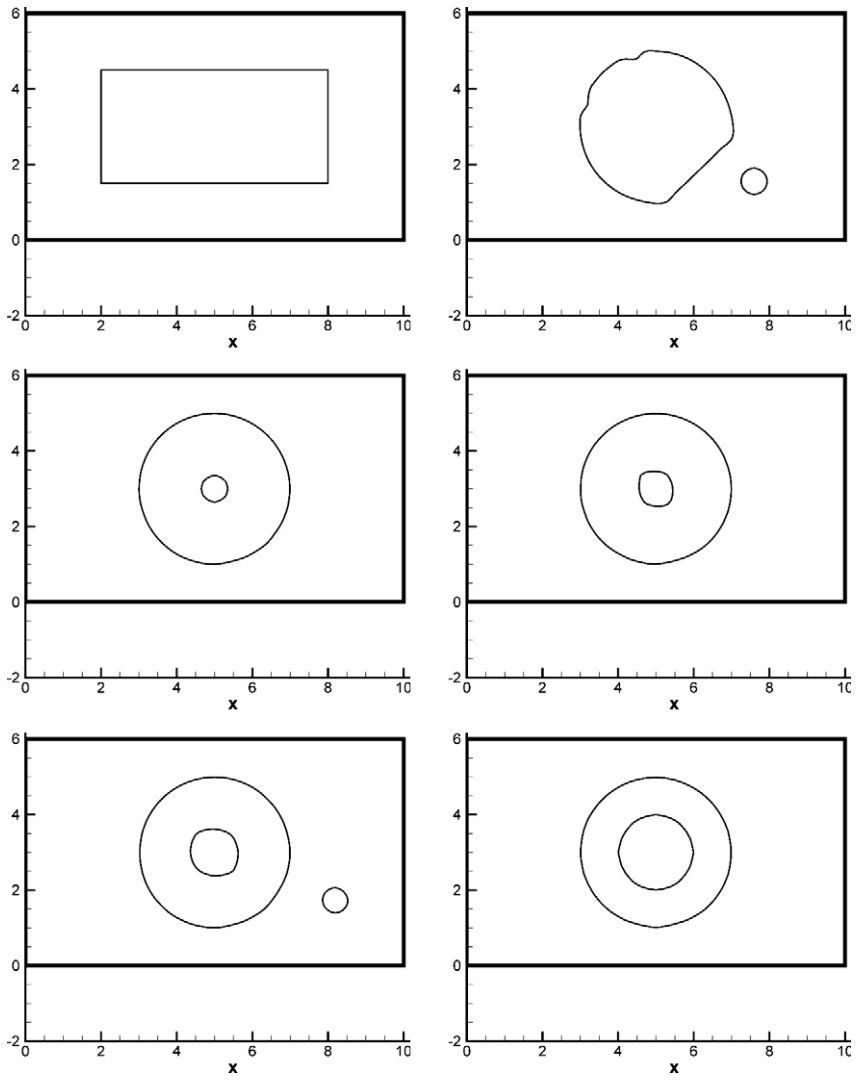

Fig. 14. Coupling of shape and topological derivatives. Evolution of zero isoline of the level-set function. Obstacles from left to right and from top to bottom at iteration: 0, 310, 620, 1000, 1230, 2000. One rotation of $\frac{\pi}{4}$ every 18 iterations.

Considering a symmetrical obstacle whose symmetry axis is aligned with the velocity at infinity, the lift is zero. In this case it can be shown that minimizing the drag is equivalent to minimizing the norm of the velocity gradient. Based on these assumptions, with the notations $\underline{x} = (x,y)$ for the position vector and $\underline{u} = (u,v)$ for the flow velocity, the drag reduction problem reads

$$\begin{aligned}
&\min_{\overline{\Omega}_{obs}\in\Theta} \quad J(\overline{\Omega}_{obs},(\underline{u}(\overline{\Omega}_{obs}),p(\overline{\Omega}_{obs}))) = \nu\int_{\Omega\setminus\overline{\Omega}_{obs}} |\nabla\underline{u}|^2\,d\Omega,\\
&\text{subject to} \quad \begin{cases} -\nu\Delta\underline{u}+\nabla p = 0 \quad \text{in } \Omega\setminus\overline{\Omega}_{obs},\\ \nabla\cdot\underline{u} = 0 \quad \text{in } \Omega\setminus\overline{\Omega}_{obs},\\ \underline{u} = 0 \quad \text{on } \partial\Omega_{obs}, \end{cases}\\
&\Theta = \{\Omega\setminus\overline{\Omega}_{obs},\ |\overline{\Omega}_{obs}| = |\overline{\Omega}_{obs,0}|\},
\end{aligned} \tag{29}$$

where $\underline{u}$ and p stand respectively for the velocity and pressure of the fluid, ν denotes the kinematic viscosity, and $|\overline{\Omega}_{obs,0}|$ is the initial obstacle area.

We will focus on two distinct optimal design problems to investigate the effects of different sets of boundary conditions on $\partial\Omega = \Gamma_{top}\cup\Gamma_{bottom}\cup\Gamma_{left}\cup\Gamma_{right}$. The first set of boundary conditions aims at approximating open frontiers, therefore we consider a large enough domain Ω compared to the size of the obstacle. Typically, a uniform flow velocity $\underline{u} = (1,0)$ is imposed at the inlet Γ_{left}, and similar Dirichlet conditions are assumed to be applied on the three remaining sides Γ_{right}, Γ_{top}

and Γ_{bottom}. In other words, we assume that the size of the fluid region is such that the horizontal stream is re-established on all edges.

The second set of boundary conditions on $\partial\Omega$ is the one relative to a 2D channel. At the inlet, a Poiseuille profile such that the flow rate is equal to 1 is imposed: $u|_{\Gamma_{left}} = \left(\frac{6y(H-y)}{H^3}\right)$. The height of the computational domain is denoted by H, while y is the ordinate of a given mesh point $\underline{x}$ in Ω. The top and bottom edges are now solid walls and therefore $\underline{u} = 0$ there. Finally, Γ_{right} is treated otherwise than in the previous case. It is indeed modeled thanks to a transparent Neumann-type boundary condition, derived so that the traction force at the exit is equal to the one of a reference flow [17]. With $\bar{\bar{\sigma}}$ standing for the stress tensor at a given point of Γ_{right}, $\underline{n}$ for the outward normal unit vector to the fluid domain, and Re for the Reynolds number, we have:

$$\bar{\bar{\sigma}}(\underline{u},p)\cdot\underline{n} = \bar{\bar{\sigma}}(\underline{u}_{ref},p_{ref})\cdot\underline{n} \quad \text{where } \bar{\bar{\sigma}}(\underline{u},p) = \begin{pmatrix} \frac{2}{Re}\frac{\partial u}{\partial x} - p & \frac{1}{Re}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \\ \frac{1}{Re}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) & \frac{2}{Re}\frac{\partial v}{\partial y} - p \end{pmatrix}. \tag{30}$$

In this case, a natural choice for the reference velocity and pressure fields is the Poiseuille flow and thus $\underline{u}_{ref}|_{\Gamma_{right}} = \underline{u}|_{\Gamma_{left}}$. The reference pressure is defined so that $p_{ref}|_{\Gamma_{right}} = 0$ at the exit.

3.2. Adjoint problem and shape gradient

In this section, the gradient of the cost function J with respect to a small perturbation of the body interface $\partial\Omega_{obs}$ in the direction of its normal is computed. For completeness, we present two derivations. The first, due to Pironneau [18], is based on sensitivity analysis. The second one, leading to the same results, makes use of the Lagrange multipliers technique.

Let us start by introducing a domain $(\Omega \setminus \overline{\Omega}'_{obs})$ very close to $(\Omega \setminus \overline{\Omega}_{obs})$ defined by its boundary $\Gamma' = \{\underline{x} + \alpha(\underline{x})\underline{n}(\underline{x}),\ \forall \underline{x} \in \partial(\Omega \setminus \overline{\Omega}_{obs})\}$. The amplitude of the displacement α is assumed to be small and sufficiently regular. It is also supposed to vanish on $\partial\Omega$, so that the only changes should occur in the vicinity the obstacle border. Since extending the velocity $\underline{u}$ to zero inside the shape fits the requirement of the Dirichlet no-slip boundary condition on $\partial\Omega_{obs}$, $\delta\underline{u} = \underline{u}(\Omega \setminus \overline{\Omega}'_{obs}) - \underline{u}(\Omega \setminus \Omega_{obs}) = \underline{u}' - \underline{u}$ is well-defined in Ω. The perturbation of the functional J can thus be written as the sum of two terms:

$$\delta J = \nu\delta\left(\int_{\Omega\setminus\overline{\Omega}_{obs}} |\nabla\underline{u}|^2\, d\Omega\right) = \underbrace{\nu\int_{\delta(\Omega\setminus\overline{\Omega}_{obs})} |\nabla\underline{u}|^2\, d\Omega}_{(a^*)} + \underbrace{2\,\nu\int_{\Omega\setminus\overline{\Omega}_{obs}} \nabla\delta\underline{u} : \nabla\underline{u}\, d\Omega}_{(b^*)} + o(\|\alpha\|), \tag{31}$$

where the notation $A{:}B$ stands for $\sum_{1\leqslant i,j\leqslant 2} A_{ij}B_{ij}$. Under the hypothesis of not perturbing $\partial\Omega$, the term (a^*) is approximated by:

$$\nu\int_{\delta(\Omega\setminus\overline{\Omega}_{obs})} |\nabla\underline{u}|^2\, d\Omega = \nu\int_{\partial(\Omega\setminus\overline{\Omega}_{obs})} \alpha|\nabla\underline{u}|^2\, ds + o(\|\alpha\|) = \nu\int_{\partial(\Omega\setminus\overline{\Omega}_{obs})} \alpha|\partial_n\underline{u}|^2\, ds + o(\|\alpha\|). \tag{32}$$

The velocities $\underline{u}'$ and $\underline{u}$ satisfy the same boundary conditions on the edges of the computational domain, and therefore we obtain

$$\begin{cases} -\nu\Delta\delta\underline{u} + \nabla\delta p = 0 & \text{in } \Omega\setminus\overline{\Omega}_{obs}, \\ \nabla\cdot\delta\underline{u} = 0 & \text{in } \Omega\setminus\overline{\Omega}_{obs}, \\ \delta\underline{u} = 0 & \text{on } \partial\Omega. \end{cases}$$

Finding the value of $\delta\underline{u}$ on $\partial\Omega_{obs}$ is less obvious. It requires the use of a Taylor expansion as well as the Dirichlet conditions for both the perturbed and the unperturbed velocities at the obstacle interface, i.e. $\underline{u}'|_{\partial\Omega'_{obs}} = 0$ and $\underline{u}|_{\partial\Omega_{obs}} = 0$. For $\underline{x} \in \partial\Omega_{obs}$

$$\underline{u}'|_{\partial\Omega'_{obs}} = \underline{u}'(\underline{x} + \alpha\underline{n}) = 0 \Rightarrow \underline{u}'(\underline{x}) + \alpha\partial_n\underline{u}'|_{\partial\Omega_{obs}} + o(|\alpha|) = u(\underline{x}),$$

which leads to the final boundary condition for $\delta\underline{u}$ in order to close the system

$$\delta\underline{u}| = -\alpha\partial_n\underline{u} \quad \text{on } \partial\Omega_{obs}.$$

We are now in the position to simplify the term (b^*) in (31) in the following way:

$$\begin{aligned} \nu\int_{\Omega\setminus\overline{\Omega}_{obs}} \nabla\delta\underline{u} : \nabla\underline{u}\, d\Omega &= \int_{\Omega\setminus\overline{\Omega}_{obs}} p\nabla\cdot\delta\underline{u}\, d\Omega - \int_{\partial\Omega\cup\partial\Omega_{obs}} p\delta\underline{u}\cdot\underline{n}\, ds + \nu\int_{\partial\Omega\cup\partial\Omega_{obs}} \partial_n\underline{u}\cdot\delta\underline{u}\, ds = \int_{\partial\Omega_{obs}} (\nu\partial_n\underline{u} - p\underline{n})\cdot(-\alpha\partial_n\underline{u})\, ds \\ &= -\int_{\partial\Omega_{obs}} \nu\alpha|\partial_n\underline{u}|^2\, ds, \end{aligned}$$

where we have exploited the fact that, since $\underline{u}$ is constant on the obstacle, $\underline{n}\cdot\partial_n\underline{u} = \underline{t}\cdot\partial_t\underline{u} = 0$ on $\partial\Omega_{obs}$, $\underline{t}$ being the tangent vector to the boundary. Finally, we have

$$\delta J = -v \int_{\partial\Omega_{obs}} \alpha |\partial_n \underline{u}|^2 \, ds + o(\|\alpha\|). \tag{33}$$

The same result can be retrieved by the adjoint method. Let $\underline{\lambda} : \mathbb{R}^2 \to \mathbb{R}^2$, $\mu_1 : \mathbb{R} \to \mathbb{R}$ and $\underline{\mu_2} : \mathbb{R}^2 \to \mathbb{R}^2$ be three smooth enough Lagrange multipliers. Solving problem (29) is equivalent to looking for the stationary points of

$$L((\underline{u}, p), \phi) = v \int_{\{\underline{x}/\phi(\underline{x})<0\}} |\nabla \underline{u}|^2 \, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})<0\}} \underline{\lambda} \cdot (v\Delta \underline{u} - \nabla p) \, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})<0\}} \mu_1 \nabla \cdot \underline{u} \, d\Omega + \int_{\{\underline{x}/\phi(\underline{x})=0\}} \underline{\mu_2} \cdot \underline{u} \, ds, \tag{34}$$

where the notation for representing the shape with the zero level-set of a distance function ϕ is used again.

Let us suppose that $(\tilde{\underline{u}}, \tilde{p})$ are infinitesimal perturbations of the solution of the Stokes problem. Assuming the velocity perturbations are cancelled on $\partial\Omega \cup \partial\Omega_{obs}$ to keep the boundary conditions unchanged for the perturbed flow, i.e. $\tilde{\underline{u}} = 0$ on $\partial\Omega$, the calculation of $\delta L_{(\tilde{\underline{u}}, \tilde{p})} = L((u + \tilde{\underline{u}}, p + \tilde{p}), \phi) - L((u, p), \phi) = 0$ leads us to the adjoint system satisfied by $(\underline{\lambda}, \mu_1)$

$$\begin{cases} v\Delta \underline{\lambda} - \nabla \mu_1 = 2\nabla p \quad \text{in } \Omega \setminus \overline{\Omega}_{obs}, \\ \nabla \cdot \underline{\lambda} = 0 \quad \text{in } \Omega \setminus \overline{\Omega}_{obs}, \\ \underline{\lambda} = 0 \quad \text{on } \partial\Omega \text{ and } \partial\Omega_{obs}. \end{cases} \tag{35}$$

Imposing that $\underline{\lambda} = 0$ and $\mu_1 = -2p$ in $\Omega \setminus \overline{\Omega}_{obs}$, Eq. (35) are satisfied, and $\underline{\mu_2}$ can be calculated on the geometry interface as

$$\delta L_{(\tilde{\underline{u}}, \tilde{p})} = 0 \Rightarrow \underline{\mu_2} = v\nabla \underline{\lambda} \cdot \underline{n} + 2p\underline{n} - 2v\nabla \underline{u} \cdot \underline{n} \quad \text{on } \partial\Omega_{obs}.$$

Taking this result into account as well as the Stokes equations, we compute the perturbation of the Lagrangian L according to a small modification of the level-set function ϕ with an amplitude $\tilde{\phi}$

$$\delta L_{\tilde{\phi}} = \delta J_{\tilde{\phi}} - \int_{\{\underline{x}/\phi(\underline{x})=0\}} \partial_n(\underline{\mu_2} \cdot \underline{u}) \tilde{\phi} \, ds = - \int_{\{\underline{x}/\phi(\underline{x})<0\}} v(|\partial_n \underline{u}|^2 + (p\underline{n} - 2v\partial_n \underline{u})\partial_n \underline{u}) \tilde{\phi} \, ds$$

and since $\underline{n} \cdot \partial_n \underline{u} = 0$ on $\partial\Omega_{obs}$, we have

$$\delta L_{\tilde{\phi}} = v \int_{\{\underline{x}/\phi(\underline{x})=0\}} \tilde{\phi} |\partial_n \underline{u}|^2 \, ds. \tag{36}$$

It now appears that α and $-\tilde{\phi}$ play the same roles. Hence, a sufficient condition to minimize the drag is

$$\tilde{\phi} = -|\partial_n \underline{u}|^2 \quad \text{on } \partial\Omega_{obs}, \tag{37}$$

provided that we limit the perturbation $\tilde{\phi}$ such that $\delta J < 0$.

If the area constraint were not imposed, successive perturbations of the boundary according to the above formula would converge to a domain where the geometry is removed. This matches the requirement for drag minimization in this specific hypothesis.

3.3. Penalization and improvement of the gradient at the interface

3.3.1. Solving the Stokes problem

The obstacle is considered as a porous medium whose porosity κ is an infinitesimal parameter. Typically, $\kappa = 10^{-8}$ in our computations. According to the Darcy equations, a penalization term is added on the velocity, and the properties of the distance function are used to make the distinction between fluid and solid. We recall that $\phi > 0$ inside the obstacle and negative elsewhere. The penalization term is such that the incompressibility hypothesis is verified inside the geometry: $\nabla \cdot \underline{u} = 0$ in $\overline{\Omega}_{obs}$. Furthermore, the Stokes problem is treated as an evolution problem. The governing equations to be actually solved in the whole domain Ω are written hereafter

$$\begin{cases} \partial_t \underline{u} - v\Delta \underline{u} + \nabla p + \frac{1}{\kappa} H(\phi) \underline{u} = 0 \quad \text{in } [0, T] \times \Omega, \\ \nabla \cdot \underline{u} = 0 \quad \text{in } \Omega, \\ +\text{boundary conditions} \quad \text{on } \partial\Omega. \end{cases} \tag{38}$$

From a discrete standpoint, the numerical simulation relies on the coupling of a second-order Gear-scheme in time with a second-order finite-difference approximation of the linear terms in space. The time integration problem is solved implicitly, whereas the spatial discretization is obtained on a uniform cartesian staggered grid G_h, as in the Marker and Cell (MAC) method [19]. For a given square cell (i,j), the discrete value of the pressure $p_{i,j}$ is located at the center. The collocation points of the horizontal component of the velocity field $u_{i-\frac{1}{2}j}$, $u_{i+\frac{1}{2}j}$ are respectively at the middle of the left and right edges. The collocation points of vertical component $v_{i-\frac{1}{2}j}$, $v_{i+\frac{1}{2}j}$ are found at the middle of the top and bottom edges. This discretization leads to a linear system $A_h U_h^n = B_h^{n-1}$ where $U_h^n = (U_h^n, p_h^n)$ is the Stokes solution to be computed for a given shape in the optimization algorithm. This is done through a multigrid solver from the finest grid chosen for the approximation G_p to a coarsest grid G_1 of 4 * 4 cells, see [20] for details.

3.3.2. Iterative improvement of the solution and its gradient

Previous paragraphs have established that the normal displacements of the zero level-set of the distance function ϕ can be directly found from the computation of the gradient of the Stokes solution along $\underline{n}$. We wish to adapt the iterative correction at the interface presented in Section 2.2, to the solution of (29). Our goal is to find a relevant non-null extension of the velocity $\underline{u}$ inside the penalized area where $\phi > 0$. This prolongation imposes that the isocontours $\{\underline{x}/\underline{u}(\underline{x}) = 0\}$ and $\{\underline{x}/\phi(\underline{x}) = 0\}$ will coincide in the limit of the numerical approximation. To this end, the method used in previous sections has to be slightly modified. The divergence-free constraint is satisfied outside the obstacle, whereas we impose that the divergence of the penalized solution must be equal to the divergence of its extension inside the obstacle. Hence $\underline{u}$ and p solve

$$\begin{cases} \partial_t \underline{u} - \nu\Delta\underline{u} + \nabla p + \frac{1}{\kappa}H(\phi)(\underline{u} - \underline{u}_{penal}) = 0 \quad \text{in } [0,T]\times\Omega, \\ \nabla\cdot\underline{u} = \begin{cases} 0 & \text{if } \phi < 0, \\ \nabla\cdot\underline{u}_{penal} & \text{if } \phi > 0, \end{cases} \\ +\text{boundary conditions} \quad \text{on } \partial\Omega, \end{cases} \tag{39}$$

where $\underline{u}_{penal} = \phi\beta$. The vector β is the constant prolongation of $\partial_n\underline{u}$ from the obstacle boundary to the interior of the obstacle along $\nabla\phi$. Note that the use of MAC grid leads to slight modifications in the space discretization compared to Section 2.2, because ϕ is defined on a grid twice as fine as the ones for $\underline{u}$ and p.

Just like for the Poisson problem, the solution improvement in $\Omega\setminus\overline{\Omega}_{obs}$ provided by the iterative extension technique is shown on an analytical example using the Stokes operator. A circle of radius $r = 1$ is placed at the center (x_0,y_0) of a square domain $[x_{left},x_{right}]\times[y_{bottom},y_{top}] = [0,10]\times[0,10]$. The right-hand side of the Stokes equation is such that the velocity components are both polynomial. They are designed to satisfy the no-slip boundary conditions not only at the top and bottom edges of Ω, but also at the obstacle interface. In addition, the velocity at the inlet (left) has a Poiseuille-like flow profile. Hence, we have

$$\begin{cases} u(x,y) = (y_{top}-y)(y-y_{bottom})((x-x_0)^2 + (y-y_0)^2 - r^2), \\ v(x,y) = (x-x_{left})(y_{top}-y)(y-y_{bottom})((x-x_0)^2 + (y-y_0)^2 - r^2). \end{cases}$$

Fig. 15 shows that the relative error in $\Omega\setminus\overline{\Omega}_{obs}$ between numerical and analytical velocity components decreases significantly and stabilises after only two iterations, like in the case of the Poisson problem.

3.3.3. Shape gradient smoothing and area preservation

Some of the methods for preconditioning the shape gradient that we presented above for the inverse Poisson problem, no longer make sense in this new framework. For example, this is the case for the scanning technique. This paragraph focuses on the preconditioning of the shape gradient $\tilde{\phi}$ through a multiscale approach similar to that of Section 2.4.1.

The perturbation of the interface is the product of the velocity derivatives in the direction normal to $\overline{\Omega}_{obs}$. In infinite dimensional space, this quantity might suffer from a loss of regularity [21]. Our objective is therefore to obtain a smoother shape perturbation by computing $\tilde{\phi}(t_{end})$ as the solution of an appropriate diffusion equation whose initial condition is given by the shape perturbation (37). We have

$$\begin{cases} \partial_t\tilde{\phi} = \xi\Delta\tilde{\phi} \quad \text{in } \Omega, \\ \tilde{\phi}(t=0,\underline{x}) = \tilde{\phi}_0(\underline{x}) \quad \text{in } \Omega, \\ \tilde{\phi} = \tilde{\phi}_0 \quad \text{on } \partial\Omega, \end{cases} \tag{40}$$

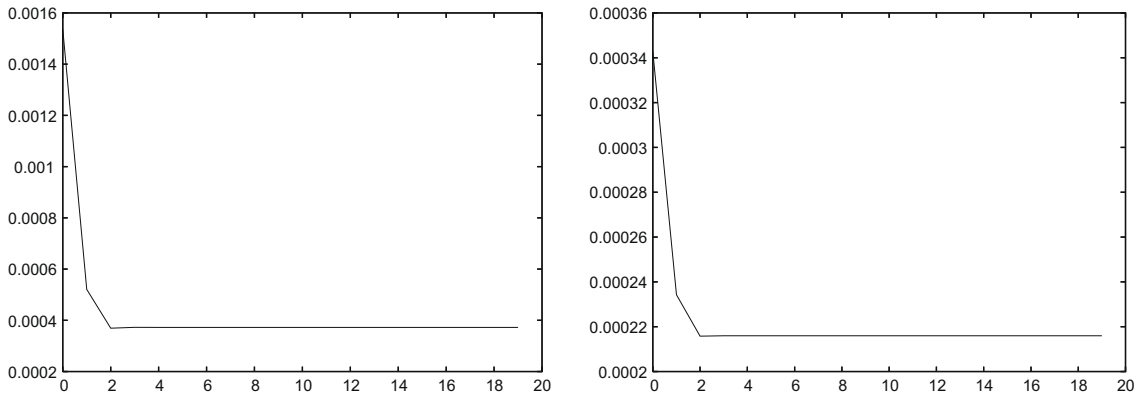

Fig. 15. Evolution of the relative error in $\Omega\setminus\Omega_{obs}$ as a function of the number of $\underline{u}_{penal}$ prolongations. Left: $E = \frac{\|u-u_{ana}\|_{l^2}}{\|u_{ana}\|_{l^2}}$. Right: $E = \frac{\|v-v_{ana}\|_{l^2}}{\|v_{ana}\|_{l^2}}$. Finest grid 128 * 128. Coarsest 4 * 4.

where $\tilde{\phi}_0 = -|\partial_n \underline{u}|^2$ defines both the initial and boundary conditions on $\partial\Omega$. This is equivalent to performing an high-frequency filtering of the initial shape perturbation.

The higher t_{end}, the more regular the shape perturbation at the final time. During the optimization, the smoothing time decreases by plateaus as a function of the total number of iterations. As a matter of fact, we start with $t_{end} = 1$ and fix this value for N consecutive iterations. The artificial viscosity ξ is initially set at iteration 1 in order that the relative error defined on a strip surrounding the interface

$$E_s = \frac{|\max(|\tilde{\phi}(t_{end})|) - \max(|\tilde{\phi}_0|)|}{|\max(|\tilde{\phi}_0|)|},$$

which is a monotonically increasing function of ξ, should not exceed a certain threshold E_{tol}. Should the error exceed this threshold again at some given optimization iteration on this plateau, ξ would be reduced accordingly. For the subsequent N iterations, the fixed value of the final time is then decreased of 0.1. The only parameter to vary at this new given level is again the artificial viscosity, whose value is initially kept the same as it was in the previous plateau. This process is repeated until $t_{end} = 0$ is reached. Every time the shape derivative is smoothed, we find a coefficient β so that $\beta \max(|\tilde{\phi}(t_{end})|) = \max(|\tilde{\phi}_0|)$. Finally, the displacement of the front is performed by $\beta\tilde{\phi}(t_{end})$.

The last aspect of the design problem is the geometric constraint of area conservation. At every iteration, for the perturbed shape to have the same volume as the initial geometry, a projection must be carried out on the gradient. Two approaches are considered here. One is based on the application of the Leray decomposition to $\tilde{\phi}$ in order to find a divergence-free projection of the normal displacement vector field $\underline{\tilde{\phi}} = \tilde{\phi}\underline{n}$. This vector field is written as the sum of the curl of a vector field $\underline{\gamma}$ and the gradient of a scalar field ψ

$$\underline{\tilde{\phi}} = \nabla \wedge \underline{\gamma} + \nabla\psi. \tag{41}$$

Let us consider the divergence of Eq. (41). Since the divergence of the curl is null, $\nabla \wedge \underline{\gamma} = \underline{\tilde{\phi}} - \nabla\psi$ is the projection we are looking for. The scalar field ψ is found by solving the Poisson equation

$$\begin{cases} \Delta\psi = \nabla \cdot \underline{\tilde{\phi}} & \text{in } \Omega, \\ \psi = 0 & \text{on } \partial\Omega. \end{cases} \tag{42}$$

The boundary condition for ψ is chosen so that this vector field decomposition is direct. Finally, the divergence-free perturbation is $\tilde{\phi}_2 = (\nabla \wedge \underline{\gamma}) \cdot \underline{n}$. However due to numerical error, this perturbation is not exactly divergence free and during the optimization this small error in preserving the area may build up. In order to preserve exactly the volume constraints without affecting the direction of descent, a small homothetical correction is applied to the level-set function. At the beginning of each optimization iteration, if ϕ_0 denotes the distance function corresponding to the initial obstacle configuration and ϕ is the current one, we find $\tau \in \mathbb{R}$ such that

$$A_\tau(\phi) = \int_\Omega H_r(\phi + \tau)\, d\Omega = A(\phi_0) = \int_\Omega H_r(\phi_0)\, d\Omega, \tag{43}$$

where H_r is the following regularization of the Heaviside function

$$H_r(\phi) = \begin{cases} 1 & \text{if } \phi > \frac{\Delta x}{2}, \\ 0 & \text{if } \phi < -\frac{\Delta x}{2}, \\ \frac{1}{2}\left(1.0 + \frac{\phi}{\Delta x/2} + \frac{1}{\pi}\sin\left(\frac{\pi\phi}{\Delta x/2}\right)\right). \end{cases}$$

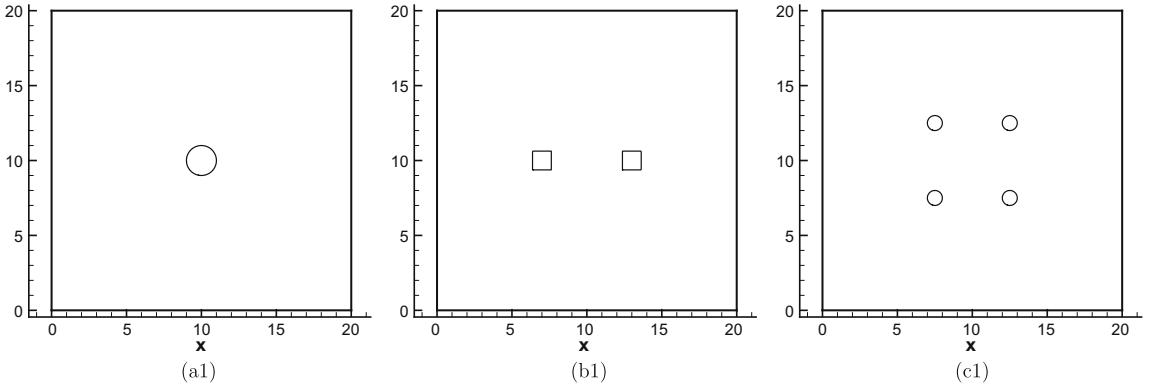

Fig. 16. Optimal design of obstacles of a fixed area π starting from a circle (a1), two rectangles (b1) and four circles (c1).

The Leray decomposition has a non-local nature due to the solution of a Poisson problem. Hence, the redistribution of the displacement field induced by the decomposition to satisfy the incompressibility constraint may be affected by non-local features of the gradient field. In general this may lead in turn to a shape perturbation that is not necessarily a descent direc-

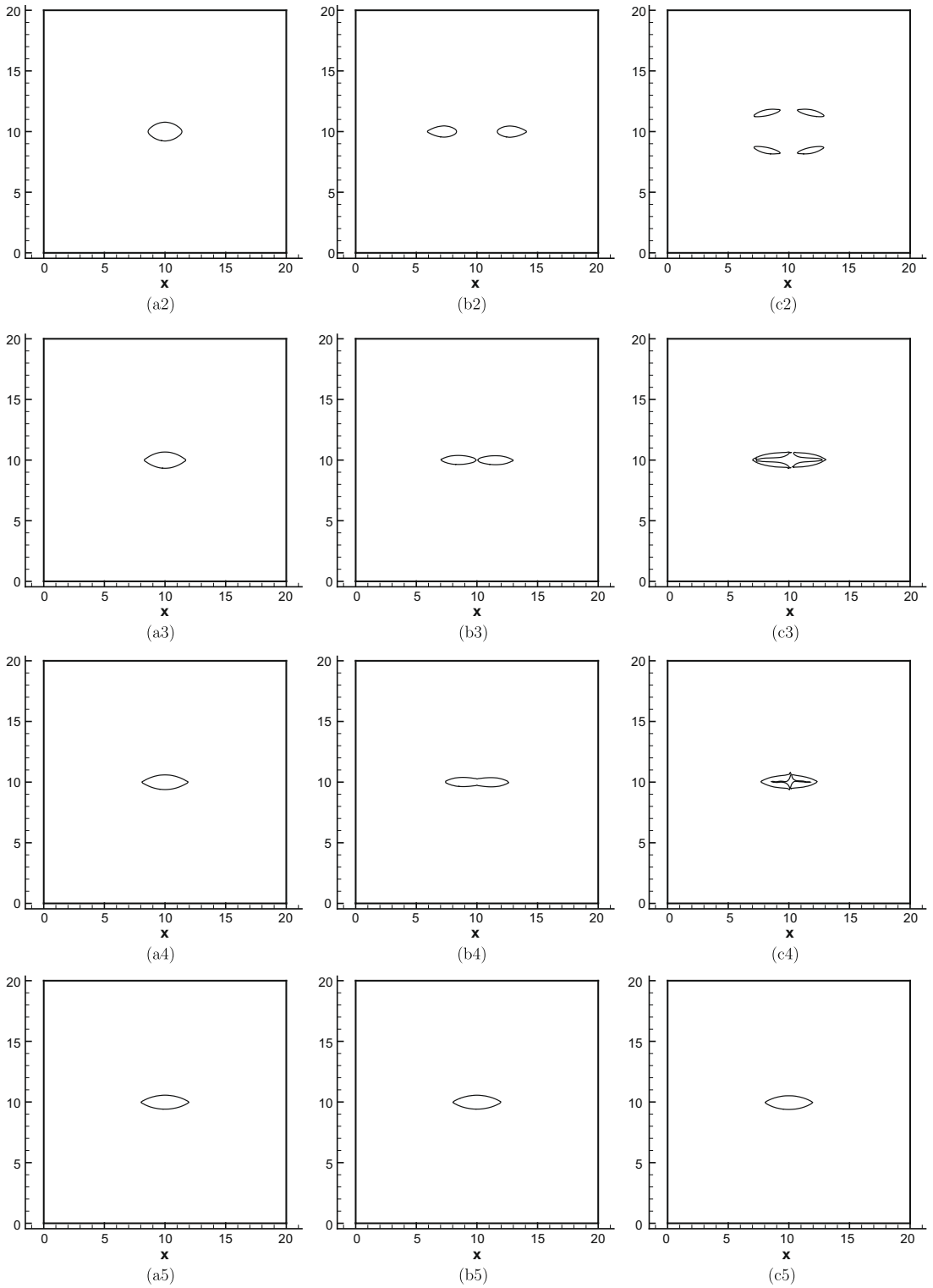

Fig. 17. Evolutions of the zero isolines of the level-set function as a function of the number of iterations. Obstacles at iterations 0, 20, 40, 60, 6000 (a2–a5); 0, 40, 200, 240, 6000 (b2–b5); 0, 200, 1200, 2000, 6000 (c2–c5). With Leray projection.

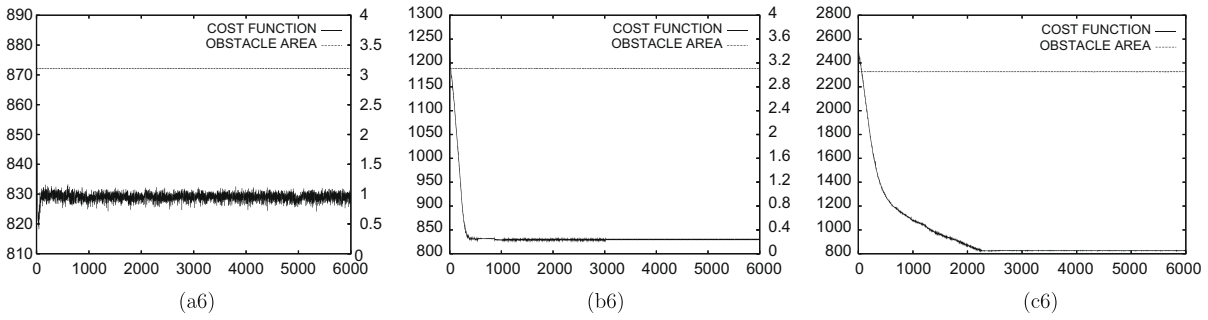

Fig. 18. Functional convergence starting from a circle (a6), two rectangles (b6) and four circles (c6).

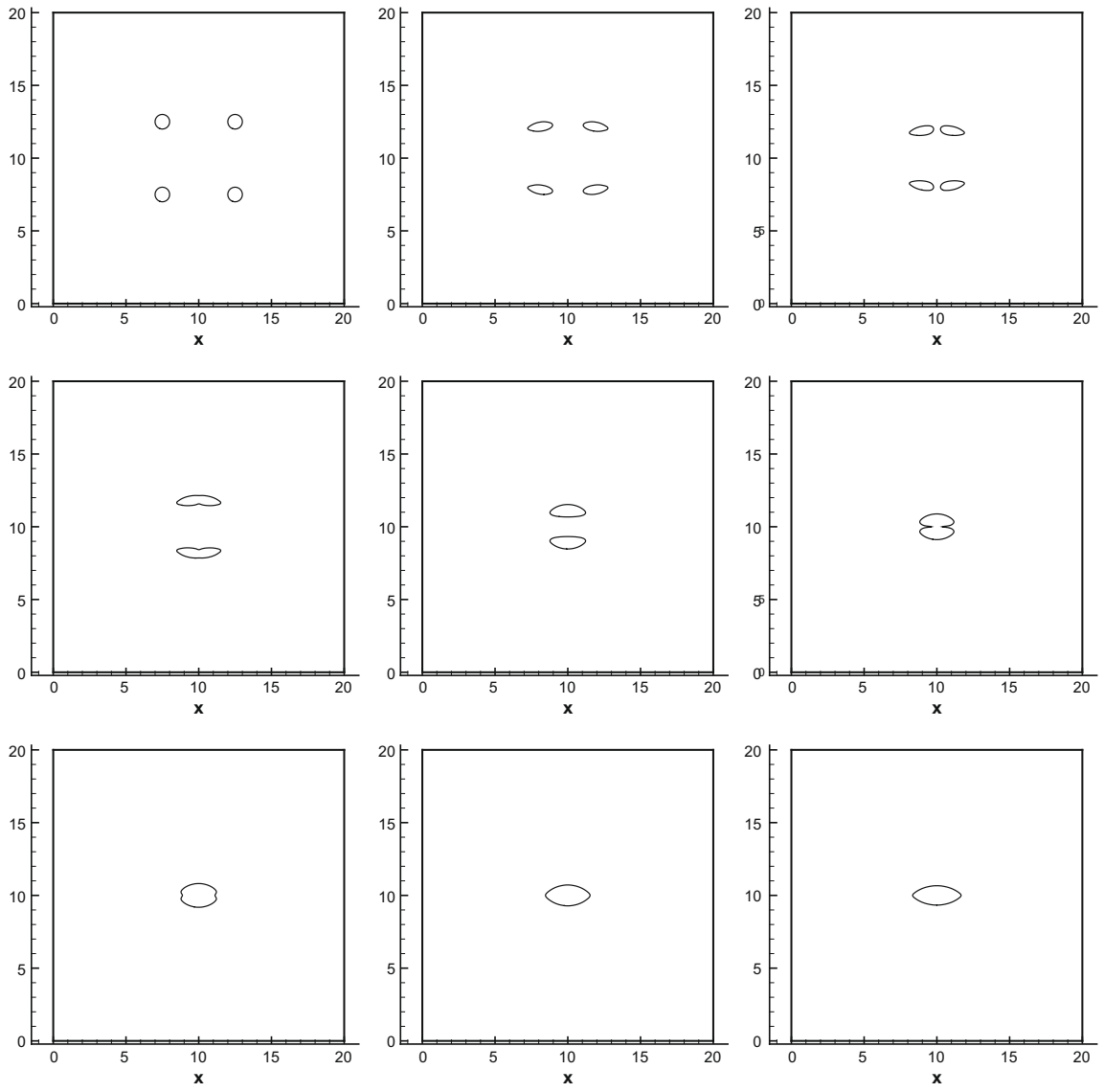

Fig. 19. Evolutions of the zero isolines of the level-set function as a function of the number of iterations. Obstacle at iterations 0, 100, 200, 240, 400, 520, 540, 600, 2000. From up left to right bottom. Without Leray projection.

tion. However, the fact that the Leray decomposition is non-local can be advantageous in the sense that it naturally provides a more uniform displacement field over the whole domain, and in particular on the obstacle boundary. On the other hand,

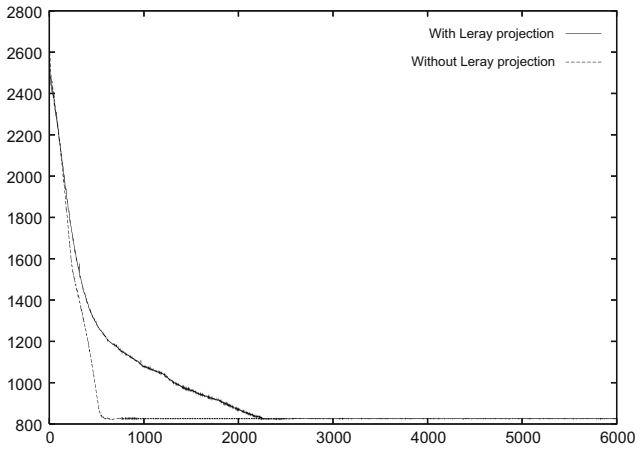

Fig. 20. Functional convergence in the four cylinder case. Comparison of the convergence history with and without Leray projection.

the classical approach to satisfy the volume constraint consists in a simple translation of $\tilde{\phi}$ of a quantity equal to the average displacement on $\partial\Omega_{obs}$. In practice this is done by simply using the same homothetical transformation described above. However, while the latter method is always a descent direction, the same cannot be proven for Leray decomposition as long as the gradient is defined only on $\partial\Omega_{obs}$.

3.3.4. Results

In this paragraph, optimal design results are presented for an obstacle whose area is constrained to remain π at all optimization steps. In Fig. 16 the case of an open domain flow $[0,20] \times [0,20]$ with uniform velocity Dirichlet condition on all edges of $\partial\Omega$ is presented. Indeed, keeping the same size of the computational domain, the optimal bodies retrieved for channel flows are very similar. With a large enough domain, a small influence of the boundary conditions on the exterior domain is observed. Different topologies (a1, b1 and c1) are studied for the initial shape. The final time for the gradient smoothing by the diffusion operator decreases every $N = 100$ iterations in all tests. The finest grids for computing the velocity components and the pressure are $256 * 256$.

In Fig. 17, we present the evolution of the obstacles as a function of the optimization step. Configurations (b) and (c) show large scale displacements to get to the minimum. Of course, if the initial bodies were close enough, they would have an optimization history like that of (a). The final representations of shapes by the zero level-set do not suffer from boundary oscillations as the ones presented in the inverse problem chapter. This is one effect of applying a smooth normal displacement. Besides the improvement on the accuracy of the final bodies, we have observed that stronger smoothing in the early iterations provides more homogeneous values of the shape gradient around the interface. This speeds up the minimization because larger displacements steps are allowed. In cases (b) and (c) it was impossible to obtain convergence to the optimal body without gradient preconditioning.

The convergence history in terms of functional values is presented in Fig. 18. Cases (a) and (b) reach the optimal solution in a much smaller number of iterations than case (c).

Cases (a)–(c) were investigated also by using the classical approach to satisfy the volume constraint. After perturbing the shape by $\tilde{\phi}$, the volume constraint was imposed by an homothetical transform of the geometry. The results in terms of convergence rates for cases (a) and (b) are similar for the two approaches, although slightly faster convergence is achieved by using the classic approach. For case (c), the classic approach undergoes a different descent path, leading to a drastically faster convergence, as shown in Figs. 19 and 20. However, the geometry perturbations obtained with Leray projection proved to be systematically smoother as compared to the ones obtained through the sole homothetical transform. Indeed, without the Leray projection, shape instabilities develop as the optimization proceeds, requiring an additional smoothing by averaging ϕ at each grid point over the four neighbours.

4. Conclusions

We proposed an iterative method to solve either inverse or shape optimization problems based on penalization, level-sets and cartesian grids. The level-set representation of the geometries allowed us to avoid generations of body-fitted meshes at each minimization step. We showed cases where the optimal geometry was far from the initial one and had a different topology. It was possible to retrieve multiply connected bodies from simply connected initial condition and vice versa. For such cases automatic remeshing as usually done in shape optimization, would be arduous. A penalization technique was incorporated in this framework in order to accurately solve the governing equations. It was extended in order to addressed the issue of improving the solution gradients near the solid interfaces. This was achieved by an inexpensive iterative method

taking advantage of the level-set formulation and converging in a few iterations. The cost of such iterations is limited since the initial condition is different from the solution only very close to the boundary. Since the level-set approach for optimization is intrinsically infinite dimensional (in the limit of grid resolution) an important issue in this work was gradient preconditioning. This was granted whether by an appropriate choice of the norm in the cost function or by a diffusion operator. Both methods rely on the fact that there exist a natural prolongation of the shape gradient away from the border to be optimized, thanks to the level-set representation. These preconditioning techniques were essential to get the convergence in most cases. In conclusion, we presented an optimization method that allows to consider large classes of geometries at a reduced computational cost. Future work will extend these results to the Navier–Stokes equations.

References

[1] O. Dorn, E.L. Miller, C.M. Rappaport, A shape reconstruction method for electromagnetic tomograpy using adjoint fields and level sets, Inverse Probl. 16 (2000) 1119–1156.
[2] E. Francini, T. Höft, F. Santosa, An inverse problem in nondestructive evaluation of spot-welds, Inverse Probl. 22 (2006) 645–661.
[3] Ph. Angot, Ch.-H. Bruneau, P. Fabrie, A penalization method to take into account obstacles in incompressible viscous flows, Numer. Math. 81 (1999).
[4] S. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[5] J. Céa, Conception optimale ou identification de formes calcul rapide de la dérivé directionnelle de la fonction cot, Math. Model. Numer. Anal. 20 (3) (1986) 371–402.
[6] S. J Osher, F. Santosa, Level set methods for optimization problems involving geometry and constraints. I. Frequencies of a two-density inhomogeneous drum, J. Comput. Phys. 171 (2001) 272–288.
[7] J.A. Sethian, Evolution and application of level-set and fastmarching methods for advancing fronts, J. Comput. Phys. 169 (2001) 503–555.
[8] J.A. Sethian, A fast marching level set method for monotonically advancing fronts, Proc. Natl. Acad. Sci. USA 93 (1996) 1591–1593.
[9] G. Allaire, F. De Gournay, F. Jouve, A.-M. Toader, Structural optimization using topological and shape sensitivity via a level set method, Contr. Cybernet. 34 (2005) 59–80.
[10] J. Céa, S. Garreau, P. Guillaume, M. Masmoudi, The shape and topological optimizations connection, Comput. Method Appl. Mech. Eng. 188 (4) (2000) 713–726.
[11] H. Eschenauer, H. Kobelev, A. Schumacher, A bubble method for topology and shape optimization of structures, Struct. Optim. 8 (1994) 142–151.
[12] J. Sokolowski, A. Zochowski, On the topological derivative in shape optimization, SIAM J. Contr. Optim. 37 (1999) 1251–1272.
[13] G.-S. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228.
[14] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, J. Sci. Stat. Comput. 7 (1986) 856–869.
[15] B. Protas, T.R. Bewley, G. Hagen, A computational framework for the regularization of adjoint analysis in multiscale PDE systems, J. Comput. Phys. 195 (2004) 49–89.
[16] M. Burger, B. Hackl, W. Ring, Incorporating topological derivatives into level set methods, J. Comput. Phys. 194 (2004) 344–362.
[17] C.-H. Bruneau, P. Fabrie, Effective downstream boundary conditions for incompressible Navier–Stokes equations, Int. J. Numer. Method Fluid 19 (1994) 693–705.
[18] O. Pironneau, On optimum profiles in Stokes flow, J. Fluid Mech. 59 (1973) 117–128.
[19] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, Phys. Fluid 8 (1965) 2182–2189.
[20] C-H. Bruneau, M. Saad, The 2D-lid driven cavity problem revisited, Comput. Fluid 35 (2006) 326–348.
[21] B. Koobus, N. Marco, A. Dervieux, An additive multilevel preconditioning method, J. Sci. Comput. 12 (3) (1997) 233–251.