Mimetic Finite Difference method for Shape Optimization problems

Paola F. Antonietti[1], Nadia Bigoni[1], and Marco Verani[1]

MOX-Dipartimento di Matematica, Politecnico di Milano, P.zza Leonardo da Vinci 32, 20133 Milano, Italy.
e-mails: paola.antonietti@polimi.it, nadia.bigoni@polimi.it, marco.verani@polimi.it.

Abstract. We test the performance of the Mimetic Finite Difference method applied to a wide class of shape optimization problems. Adaptive strategies based on heuristic error indicators are also considered to validate the effectiveness of the numerical scheme.

1 Introduction

In this paper we are interested in solving shape optimization problems by using the Mimetic Finite Difference (MFD) method. For an introduction of the MFD method applied to elliptic problems we refer to, e.g., [3, 4]. Thanks to a great flexibility allowed in the choice of the grid, the MFD method turns out to be a very promising technology in the context of the approximation of shape optimization problems. Standard numerical methods such as finite elements, finite volumes and spectral elements, usually require a massive use of re-meshing techniques to preserve the geometrical regularity of the computational domain and as a consequence, the computational cost could become prohibitive (see, e.g., [7]). Since the MFD method can deal with grids made of very general polygons/polyhedra, we investigate the possibility of obtain reliable numerical simulations without resorting to any re-meshing strategy.

The paper is organized as follows: in Section 2 we study the application of the MFD method to three different shape optimization problems. The first two examples are classical shape optimization problems governed by an elliptic equation and a Stokes equation, respectively, whereas the last example is related to the solution of an elliptic free-boundary problem. Finally, in Section 3 we briefly explore the possibility of incorporating an adaptive procedure into the optimization process by showing some numerical results.

2 MFD method applied to Shape Optimization problems

Let Ω be an open, bounded set of $\mathbb{R}^2$, with a polygonal boundary $\Gamma := \partial\Omega$, and let $\mathcal{J}(\Omega, y(\Omega))$ be a cost functional, which depends on Ω itself and on the solution $y(\Omega)$ of the following boundary value problem

$$Ly(\Omega) = 0 \quad \text{in } \Omega, \tag{1}$$

where L is a differential operator. We are interested in solving the following minimization problem:

$$\text{find } \Omega^* \in \mathcal{A} : \mathcal{J}(\Omega^*, y(\Omega^*)) = \inf_{\Omega \in \mathcal{A}} \mathcal{J}(\Omega, y(\Omega)), \tag{2}$$

where $\mathcal{A}$ is a set of admissible domains in $\mathbb{R}^2$. Assuming that a local minimizer Ω^* of (2) exists, the shape optimization problem (2) can be solved by building a sequence of domains $\{\Omega^{(k)}\}$ for $k > 0$, in such a way that $\Omega^{(k)}$ converges to Ω^* as k approaches infinity (c.f. [5]). Roughly speaking, if $\Omega^{(k)}$ is the domain at iteration k, then $\Omega^{(k+1)}$ is updated by

$$\Omega^{(k+1)} = \Omega^{(k)} + \mu^{(k)} V^{(k)},$$

where $\mu^{(k)}$ is a parameter which regulates the "length" of the movement and $V^{(k)}$ is an *admissible* descent direction. In other words, $V^{(k)}$ is such that the *shape derivative* $d\mathcal{J}(\Omega^{(k)}; V^{(k)})$ of the considered functional is negative. For a precise definition of the *shape derivative* and other technical details we refer to [5]. Problem (2) can be solved by means of the steepest-descent like algorithm (c.f. [6]), described in Algorithm 1.

Algorithm 2.1: Shape optimization algorithm

1. Given the initial domain $\Omega^{(0)}$, set $k = 1$;
2. `SOLVE` problem (1) and find $y^{(k)} = y(\Omega^{(k)})$;
3. `COMPUTE` the shape derivative $d\mathcal{J}(\Omega; V)$ of the given functional;
4. `COMPUTE` a descent direction $V^{(k)}$;
5. `FIND` an admissible stepsize $\mu^{(k)}$;
6. `UPDATE` $\Omega^{(k+1)} = \Omega^{(k)} + \mu^{(k)} V^{(k)}$, set $k = k + 1$ and `GOTO` step 2.

2.1 Elliptic problem

In the first example we solve the benchmark problem introduced in [6]. Let us consider the domain $\Omega \subset \mathbb{R}^2$ with $\partial\Omega = \Gamma_f \cup \Sigma_1 \cup \Sigma_2$, where Γ_f, Σ_1 and Σ_2 are disjoint (possibly empty) subset of $\partial\Omega$. Moreover, let D be an open bounded subset of Ω (see Figure 1). Then, let $y(\Omega)$ be the solution of the following elliptic problem:

$$-\Delta y = 0 \quad \text{in } \Omega, \quad y = 0 \quad \text{on } \Sigma_1, \quad \partial_n y = 0 \quad \text{on } \Sigma_2, \quad \partial_n y = 1 \quad \text{on } \Gamma_f. \tag{3}$$

The cost functional we aim at minimizing is set as

$$\mathcal{J}(\Omega, y(\Omega)) := \frac{1}{2} \int_D (y(\Omega) - z_g)^2 \ \mathrm{d}x + \frac{\gamma}{2} \left(\int_\Gamma \mathrm{dS} - P \right)^2, \tag{4}$$

where γ and P are positive constants. The set $\mathcal{A}$ of admissible domains is represented by all domains obtained through a deformation of Ω by keeping Σ_1 and Σ_2 fixed and by moving only Γ_f in such a way that $\Gamma_f \cap D = \emptyset$. The mimetic discretization of problem (3) and a detailed discussion of the MFD method for elliptic

problems can be found for example in [4].
Let $\mathbf{x} = (x_1, x_2)$, and let $\|\cdot\|$ denote the Euclidean norm. In the numerical test, we choose the region D equal to the half ring $\{2 \leq \|\mathbf{x}\| \leq 2.5\} \cap \{x_2 > 0\}$ and z_g in (4) as the exact solution of (3) on $\Omega = \{1 < \|\mathbf{x}\| < 3\} \cap \{x_2 > 0\}$. A global minimizer exists and it is exactly given by $\Omega^* = \{1 < \|\mathbf{x}\| < 3\} \cap \{x_2 > 0\}$ (c.f. Figure 1). We

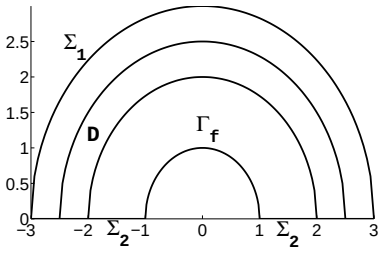

Fig. 1. Sketch of the optimal domain Ω^*.

point out that in this set of experiments no adaptive nor remeshing technique are used. In Figure 2 we report the initial computational domain (left) and a zoom of the obtained configuration (right) after four iterations of Algorithm 1.

The computed value of the functional (4) starts from 3.2990e-01 and after four iterations reduces to 9.3176e-04. A finer grid in the region D is employed in order to accurately approximate the cost functional. We can state that the method seems to be sufficiently robust despite most of the elements around the moving boundary become very stretched.

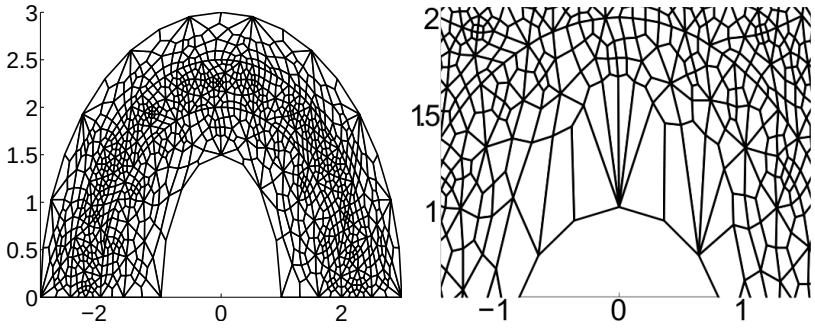

Fig. 2. Shape optimization problem (non-adaptive strategy). Initial computational domain Ω_0 (left) and zoom of the final obtained domain (right).

2.2 Drag minimization

In the second example, we are interested in modeling the flow of a fluid around an obstacle. In this numerical test, the initial computational domain is set as follows:

$$\Omega : \{[-1,-1]\times[0,1]\} \cap \{x^2+y^2 \ \geq 0.16\},$$

(see Figure 3 (left)). In this case, the obstacle is represented by the half circle lying on the lower part of the domain and it is denoted by Γ_f. The remaining parts of the boundary are labeled as follows: $\Gamma_{\text{in}} = \{(x,y) : \ x=-1\}$ and $\Gamma_{\text{out}} = \{(x,y) : \ x=1\}$ are the inflow and the outflow layers, respectively while $\Gamma_s = \{(x,y) : \ y=0\}$ and $\Gamma_w = \{(x,y) : \ y=1\}$ are the lower and upper part of the channel, respectively. The set $\mathcal{A}$ of admissible domains contains all domains obtained through a deformation of Ω by moving only Γ_f and keeping fixed the remaining parts of the boundary. The fluid flow is modeled by the following linear Stokes problem:

$$\begin{cases} -\operatorname{div}(\mathbf{T}(\mathbf{u},p)) = 0 & \text{in } \Omega, \\ \operatorname{div}\mathbf{u} = 0 & \text{in } \Omega, \\ \mathbf{u} = \mathbf{u}_d & \text{on } \Gamma_w \cup \Gamma_f \cup \Gamma_{\text{in}}, \\ \mathbf{T}(\mathbf{u},p)\cdot\mathbf{n} = 0 & \text{on } \Gamma_{\text{out}}, \\ \mathbf{u}\cdot\mathbf{n} = 0 \ \ (\mathbf{T}(\mathbf{u},p)\cdot\mathbf{n})\cdot\mathbf{t} = 0 & \text{on } \Gamma_s, \end{cases} \tag{5}$$

where $\mathbf{T}(\mathbf{u},p) := 2\epsilon(\mathbf{u}) - p\mathbf{I}$ denotes the Cauchy stress tensor. We set

$$\mathbf{u}_d = \begin{cases} [1-y^2 \ 0]^T & \text{on } \Gamma_{\text{in}} \\ 0 & \text{on } \Gamma_f \cup \Gamma_w \ . \end{cases}$$

Note that on Γ_s we impose an axial-symmetry boundary condition while on Γ_w we set a non-slip boundary condition. Again, we do not add any details about the mimetic discretization of problem (5) and refer to [2].
In this example, we choose to minimize the following cost functional:

$$\mathcal{J}(\Omega,\mathbf{u},p) := -\int_{\Gamma_f} (\mathbf{T}(\mathbf{u},p)\mathbf{n})\cdot\widehat{\mathbf{v}}_\infty \,\mathrm{dS} + \frac{\lambda}{2}\left(|\Omega_0| - \int_\Omega \mathrm{dx}\right)^2, \tag{6}$$

where $(\mathbf{u},p)$ solves (5), $\widehat{\mathbf{v}}_\infty = [1,\ 0]$ is the direction of the fluid and $|\Omega_0|$ is a given target volume value. The first term of (6) represents the drag of the fluid, while the second one penalizes the volume constraint. In Figure 3 we plot a zoom of the initial and final computational domains. The computed value of the drag starts from 4.41887e-01 and it reduces progressively to 3.69713e-02 after five iterations. We note that the obtained final configuration is in agreement with the so-called "rugby-ball" optimal shape known in the literature [8].

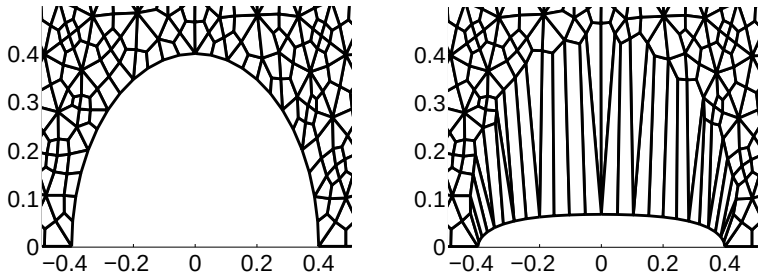

Fig. 3. Drag minimization. Zooms of the initial (left) and final (right) computed domains.

2.3 Free-boundary problem

In the last example, we are interested in solving the free-boundary elliptic problem taken from [9]. We consider an annular domain, where the fixed boundary is $\Gamma = \{\|\mathbf{x}\| = \mathbf{1}\}$ and we want to find the *moving* free-boundary $\Gamma_f := \partial\Omega \setminus \overline{\Gamma}$, so that

$$-\Delta u = 0 \quad \text{in } \Omega, \quad u = 1 \quad \text{on } \Gamma, \quad u = 0 \quad \text{on } \Gamma_f, \quad \frac{\partial u}{\partial n} = -1 \quad \text{on } \Gamma_f \ . \tag{7}$$

We formulate problem (7) as a shape optimization problem as follows. Let u solve the following *auxiliary* boundary value problem:

$$-\Delta u = 0 \qquad \text{in } \Omega, \quad u = 1 \quad \text{on } \Gamma, \quad \alpha u + \frac{\partial u}{\partial n} = -1 \quad \text{on } \Gamma_f \ . \tag{8}$$

Then, we choose a proper cost functional in order to incorporate the Dirichlet boundary condition set on Γ_f in the original free-boundary problem (7), i.e.,

$$\mathcal{J}(\Omega, u) = \int_{\Gamma_f} u^2 \, \mathrm{dS}. \tag{9}$$

Since the exact solution of the free-boundary problem (7) is zero on Γ_f, the dumping parameter $\alpha > 0$ appearing in (8) can be chosen freely. However, following [9], it turns out that $\alpha = H$, with H being the mean curvature of Γ_f, is a good choice leading in practice to a robust numerical procedure. We iteratively solve the problem (8) on the half-annulus by imposing proper axial-symmetry boundary conditions on the x-axis (c.f. Figure 4). The final obtained computational domain is reported in Figure 5 (left). The value of the computed functional reduces progressively, as shown in Figure 5 (right), where we plot the value of (9) versus the number of total iterations.

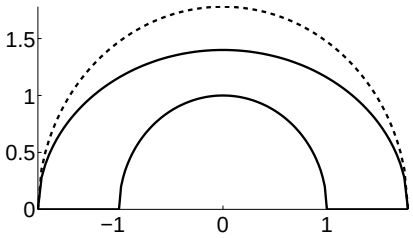

Fig. 4. Free-boundary problem. Sketch of the initial (solid line) and final (dotted line) domains.

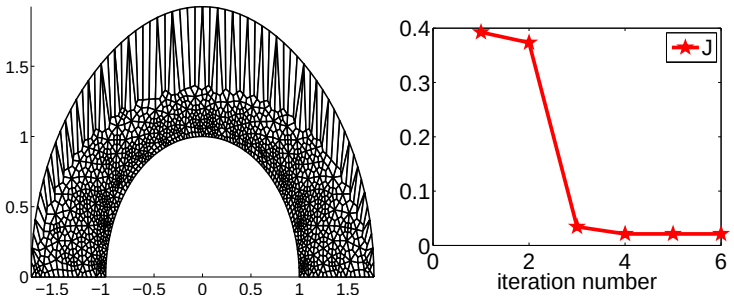

Fig. 5. Free-boundary problem. Final computational domain (left) and computed functional (9) versus the number of iteration (right).

3 Adaptive strategy

In this section, we briefly explore the possibility of incorporating mesh adaptivity into the optimization process. An example of a similar approach in the FEM context can be found in [7].
We run the same numerical experiment presented in Section 2.1 and we decide a priori to perform an adaptive refinement step every two iterations of the minimization process. To perform the adaptive procedure we employ heuristic indicators, given by the sum of the following two local error indicators:

- (η_1) for every polygon $E \subset \Omega_h$ the indicator $\eta_1(E)$ is the local discrete $H^1(E)$ seminorm of the MFD approximate solution to (3);
- (η_2) for every polygon $E \subset D$ the indicator $\eta_2(E)$ is the MFD approximation of $\frac{1}{2}\int_E (y(\Omega) - z_g)^2 \, dV$ and is set to zero outside D (c.f. Figure 2 (left)).

The local error indicators $(\eta_1 + \eta_2)(E)$ are then used to mark the elements that has to be refined, while the marking procedure relies on the Dörfler strategy with marking parameter $\theta = 0.5$. For a more detailed description of the refinement modulus we refer to [1, Section 4.1].
The optimal discrete configuration is obtained by the algorithm after 6 iterations (cfr. Figure 1 for the exact optimal domain). In Figure 6 we plot two zooms of the final configuration. Comparing Figure 2 (right) with Figure 6 (right) we can observe that here, thanks to the adaptive procedure, the element close to the moving boundary result less stretched. Moreover, due to the error indicator η_2, the adaptive algorithm correctly refines the elements in the region D (see Figure 6 (left)).

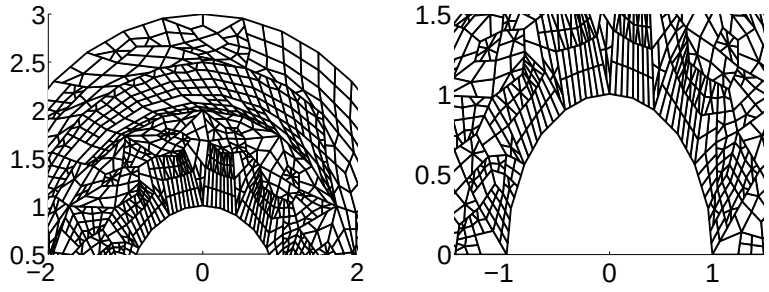

Fig. 6. Shape optimization problem (adaptive strategy). Zooms of the final obtained configuration.

We analyze the performance of the adaptive and non-adaptive strategies by comparing the obtained value of the cost functional. In Table 3 we report the computed values of $J_1 = \frac{1}{2}\int_D (y(\Omega) - z_g)^2 \, dV$ together with the corresponding number of degrees of freedom. From a closer inspection, it is evident that at comparable number of degrees of freedom the adaptive strategy obtains lower values of $\mathcal{J}$.

Table 1. Adaptive and non-adaptive strategies: cost functional versus dofs.

	non-adaptive		adaptive	
iteration	ndofs	J_1	ndofs	J_1
0	1207	9,990754E-03	157	8,780796E-03
1	1207	6,501192E-03	157	5,103216E-03
2	1207	1,493240E-03	283	5,735032E-04
3	1207	9,618140E-04	283	4,441652E-04
4	1207	8,911968E-04	564	1,368972E-04
5	-	-	564	5,677177E-05
6	-	-	1194	7,538973E-05

References

1. P. F. Antonietti, L. Beirão da Veiga, C. Lovadina, and M. Verani. Hierarchical a posteriori error estimators for the mimetic discretization of elliptic problems. *SIAM J. Numer. Anal.*, 51(1):654–675, 2013.
2. L. Beirão da Veiga, V. Gyrya, K. Lipnikov, and G. Manzini. Mimetic finite difference method for the Stokes problem on polygonal meshes. *J. Comput. Phys.*, 228(19):7215–7232, 2009.
3. F. Brezzi, A. Buffa, and K. Lipnikov. Mimetic finite differences for elliptic problems. *M2AN Math. Model. Numer. Anal.*, 43(2):277–295, 2009.
4. F. Brezzi, K. Lipnikov, and M. Shashkov. Convergence of the mimetic finite difference method for diffusion problems on polyhedral meshes. *SIAM J. Numer. Anal.*, 43(5):1872–1896 (electronic), 2005.
5. M. C. Delfour and J.-P. Zolésio. *Shapes and geometries*, volume 22 of *Advances in Design and Control*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2011.
6. G. Doğan, P. Morin, R. H. Nochetto, and M. Verani. Discrete gradient flows for shape optimization and applications. *Comput. Methods Appl. Mech. Engrg.*, 196(37-40):3898–3914, 2007.
7. P. Morin, R. Nochetto, M. Pauletti, and M. Verani. Adaptive finite element method for shape optimization. *ESAIM - Control, Optimisation and Calculus of Variations*, 18(4):1122–1149, 2012.
8. O. Pironneau. On optimum profiles in Stokes flow. *J. Fluid Mech.*, 59:117–128, 1973.
9. T. Tiihonen. Shape optimization and trial methods for free boundary problems. *RAIRO Modél. Math. Anal. Numér.*, 31(7):805–825, 1997.