A parameterized level set topology optimization strategy using FEniCS and radial basis functions with compact support

Giovanna C. Andrade[1], Sandra A. Santos[1]

[1]*Departamento de Matemática Aplicada, Universidade Estadual de Campinas*
Rua Sérgio Buarque de Holanda, 651 13083-859, Campinas, SP, Brasil
g168633@dac.unicamp.br, sasantos@unicamp.br

Abstract. This work addresses level-set based methods for structural topology optimization. Compact support C2-Wendland radial-based functions are employed to parameterize the level sets. Structures are assumed to be in static equilibrium, under plane stress state and built with isotropic linear elastic materials, and the classical compliance minimization problem is considered. Aiming to save the computational effort of the potentially large number of linear systems to be solved, the computational linear algebra of the resulting sparse interpolation matrix is explored in contrast with the use of classic multiquadric radial basis functions, with dense matrices in the related systems. Implementation is made in `Python` using the `FEniCS` project for the resolution of the equilibrium system and the sensitivity analysis, based on a recent work of Laurain, that uses direct numerical resolution of the Hamilton-Jacobi equation for the level set update. Both strategies are, therefore, compared. Additionally, our approach is put into perspective with the radial basis function parameterization `Matlab` code of Wei et al. Numerical results of experiments with classical benchmark problems are presented.

Keywords: Topology Optimization, level sets, radial basis functions.

1 Introduction

Introduced by Osher and Sethian [1], level-set based methods rest upon numerical algorithms for computing approximated solutions for the Hamilton-Jacobi partial differential equation (PDE). To cope with the intrinsic limitations associated with mesh refinement, several alternatives have been proposed for structural topology optimization (see e.g. [2] and references therein).

As highlighted in the survey of van Dijk et al., three general features distinguish the level-set topology optimization strategies: the parameterization of the level-set function; the mechanical model, and the optimization strategy. Based on radial basis functions, smooth parameterizations have been developed (cf. [3–5]). Among the most popular choices are the multiquadrics functions. Nevertheless, the compact support Wendland functions may show computational advantage as far as memory and time savings, and this is the choice of this work. Concerning the mechanical model, we have based the geometry mapping upon the ersatz material distribution within the finite element analysis, and the structural model involves linear elasticity. To accomplish the optimization strategy, the sensitivities have been updated by means of the Hilbertian extension (cf. [6, 7]) and an augmented Lagrangian scheme has been adopted, with a consistent updating of the related elements (cf. [8]).

2 The problem formulation

The structures considered in this work are assumed to be under *plane stress state* and therefore can be treated as two-dimensional [9]. Since we are interested in determining the optimum topology and geometry of a structure, the optimization variable of the problem is a set of $\mathbb{R}^2$. Let us consider a reference domain $\mathcal{D} = [0, l_x]\times[0, l_y] \subset \mathbb{R}^2$ and $\mathcal{P}(\mathcal{D})$ the set of its subsets. Then, each *design domain* considered is a set $\Omega \in \mathcal{P}(\mathcal{D})$, assumed to be in *static equilibrium*. Boundary conditions such as the external traction and support region are defined for the reference and design domain, as illustrated and described in Figure 1. *Static equilibrium* state can then be formulated as the

following boundary value problem

$$\begin{cases} \nabla \cdot \boldsymbol{\sigma}(\boldsymbol{x}) = -\boldsymbol{f}, & \boldsymbol{x} \in \Omega, \\ \boldsymbol{u}(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Gamma_D, \\ \sigma(\boldsymbol{x}) \cdot \hat{\boldsymbol{n}} = \boldsymbol{g}, & \boldsymbol{x} \in \Gamma_N, \end{cases} \tag{1}$$

where $\boldsymbol{u}$ denotes the *displacement vector*, $\boldsymbol{\sigma}(\boldsymbol{x}) = Ae(\boldsymbol{u})$ denotes the *stress*, being A the *material stiffness tensor*, and $e(\boldsymbol{u}) = (Ju + Ju^T)/2$ the *linearized strain*, with Jacobian J. In specific, we consider structures built with isotropic materials, so that the fourth order tensor A can be expressed in terms of the *Lamé constants* and $\boldsymbol{\sigma}(\boldsymbol{x}) = \mu e(\boldsymbol{u}) + \lambda \nabla \cdot \boldsymbol{u} I$. This problem can be solved using *finite element analysis* and therefore it is convenient to rewrite this problem in variational form. Let $\mathbb{V} = \{\boldsymbol{v} \in (\mathbb{H}^1(\Omega))^2 : \boldsymbol{v} = 0 \text{ em } \Gamma_D\}$ be the set of admissible displacements. Following [10], the problem of eq. (1) can be expressed as finding $\boldsymbol{u} \in \mathbb{V}$ that satisfies

$$a(\boldsymbol{u}, \boldsymbol{v}) = l(\boldsymbol{v}), \ \ \forall \ \boldsymbol{v} \in \mathbb{V}, \tag{2}$$

in which we define $a : \mathbb{V} \times \mathbb{V} \to \mathbb{R}$ and $l : \mathbb{V} \to \mathbb{R}$ by means of

$$a(\boldsymbol{u}, \boldsymbol{v}) := \int_\Omega A\, e(\boldsymbol{u}) : e(\boldsymbol{v})\, \boldsymbol{dx}, \quad l(\boldsymbol{v}) := \int_{\Gamma_N} \boldsymbol{v} \cdot \boldsymbol{g}\, \boldsymbol{ds} + \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v}\, \boldsymbol{dx}\,. \tag{3}$$

The solution $\boldsymbol{u}$ of this boundary value problem is the displacement that the structure suffers once submitted to the given conditions. Once $\boldsymbol{u}$ has been obtained, the *compliance* of the structure is defined as $J(\Omega) = a(\boldsymbol{u}, \boldsymbol{u}) = l(\boldsymbol{u})$. The optimization problem we consider can then be formulated as

$$\begin{aligned} &\text{minimize}_\Omega \quad J(\Omega) \\ &\text{subject to} \quad \mathcal{C}(\Omega) \leq 0, \quad \Omega \in \mathcal{P}(\mathcal{D}), \end{aligned} \tag{4}$$

where, given a prescribed volume fraction $\overline{V}$, we define the volume constraint as $\mathcal{C}(\Omega) = V(\Omega) - \overline{V}$, with $V(\Omega) = \frac{1}{l_x l_y} \int_\Omega \boldsymbol{dx}$.

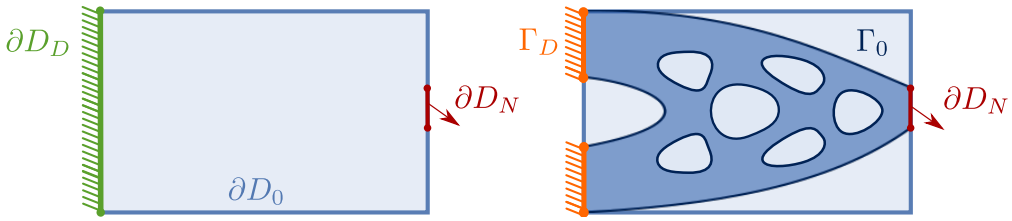

Figure 1. Reference domain $\mathcal{D}$ (left), example of design domain $\Omega \in \mathcal{P}(\mathcal{D})$ (right) and associated boundary conditions. Boundary $\partial\mathcal{D}$ is divided in three disjoints parts: $\partial\mathcal{D} = \partial\mathcal{D}_D \cup \partial\mathcal{D}_N \cup \partial D_0$, in which $\partial\mathcal{D}_D$ refers to the homogeneous Dirichlet condition; $\partial\mathcal{D}_N$ denotes the non-homogeneous Neumann condition, and $\partial\mathcal{D}_0$ denotes the homogeneous Neumann condition. As for the boundary $\partial\Omega$, we have $\partial\Omega = \Gamma_D \cup \Gamma_N$, with $\Gamma_D \subset \partial D_D$ and $\Gamma_N = \Gamma_0 \cup \partial\mathcal{D}_N$.

For the implementation, we consider the unconstrained problem of minimizing the following augmented Lagrangian cost functional [11]

$$\mathcal{L}_\rho(\Omega, \Lambda) = J(\Omega) + \frac{\rho}{2} \left(\max\{0, \mathcal{C}(\Omega) + \Lambda/\rho\}\right)^2, \tag{5}$$

in which $\Lambda \in \mathbb{R}_+$ denotes the Lagrange multiplier, and $\rho > 0$ is the penalty parameter. Also, we consider an *ersatz material approach*, which means that we assume that $D \backslash \Omega$ is treated as a weak phase, filled with material with elasticity tensor ϵA, where ϵ is a fixed small parameter. Then, the problems of eq. (1) and eq. (2) can be extended to the reference domain $\mathcal{D}$ [7, 12]. It is important to observe that the static equilibrium of the structure is often written as an equality constrain, since it must be always satisfied. In practice, however, the equilibrium is primarily imposed for each generated structure, and only the volume restriction is treated as a constraint.

3 Parameterized level set approach

In contrast with methods that rely on an explicit parameterization of the design domain, in level set methods, the geometry and topology are represented implicitly as the zero level set of a higher dimensional function, as

illustrated in Figure 2. According with [13], such methods are characterized when dynamics is added to the level set function by allowing it to vary along a fictitious time. Defining $\phi : D \times \mathbb{R}^+ \mapsto \mathbb{R}$ as the *level set function (LSF)*, we will assume the following convention

$$\begin{cases} \phi(\boldsymbol{x},t) > 0 \iff \boldsymbol{x} \in \Omega \cap D, \\ \phi(\boldsymbol{x},t) = 0 \iff \boldsymbol{x} \in \partial\Omega \cap D, \\ \phi(\boldsymbol{x},t) < 0 \iff \boldsymbol{x} \in D \backslash (\Omega \cup \partial\Omega), \end{cases} \tag{6}$$

Therefore, for each fixed t, the domain Ω_t and its boundary $\partial\Omega_t$ can be determined as $\Omega_t = \{\boldsymbol{x} \in D : \phi(\boldsymbol{x},t) > 0\}$ and $\partial\Omega_t = \{\boldsymbol{x} \in D : \phi(\boldsymbol{x},t) = 0\}$, respectively. Assuming that $\partial\Omega_t$ can be parameterized as a curve $\boldsymbol{x}(t)$, then this curve satisfies $\phi(\boldsymbol{x}(t),t) = 0$. The boundary movement along time can thus be obtained by taking the time derivative, which by the chain rule leads to the *Hamilton-Jacobi (HJ) Equation*

$$\partial_t \phi(\boldsymbol{x}(t),t) + \boldsymbol{v}(\boldsymbol{x}(t),t) \cdot \nabla\phi(\boldsymbol{x}(t),t) = 0, \quad \text{for } (\boldsymbol{x}(t),t) \text{ in } \partial\Omega_t \times \mathbb{R}^+. \tag{7}$$

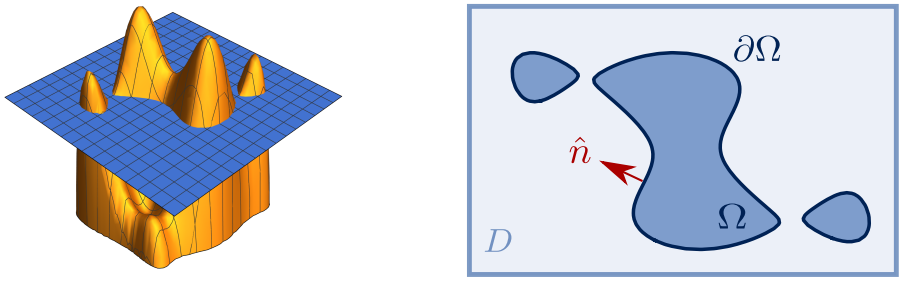

Figure 2. Implicit representation of a domain and normal vector orientation.

The term $\boldsymbol{v}(\boldsymbol{x}(t),t) := \boldsymbol{x}'(t)$ denotes the boundary velocity field. Furthermore, if v_n and v_t denote the normal and tangent components of this field, respectively, then, since the unit outward normal vector to $\partial\Omega_t$ can be written as $\hat{\boldsymbol{n}} = -\frac{\nabla\phi}{\|\nabla\phi\|}$ and $\hat{\boldsymbol{t}}$ and $\nabla\phi$ are orthogonal, we obtain an equivalent expression that involves only the normal component of the field

$$\partial_t \phi(\boldsymbol{x}(t),t) - v_n(\boldsymbol{x}(t),t)\|\nabla\phi(\boldsymbol{x}(t),t)\| = 0, \quad \text{for } (\boldsymbol{x}(t),t) \text{ in } \partial\Omega_t \times \mathbb{R}^+. \tag{8}$$

If the velocity field or its normal component are extended to the reference domain, then equations (8) and (9) may be written for all $\boldsymbol{x}$ in $\mathcal{D}$ and can be used to update the level set equation and, therefore, the design domain. This characterizes *conventional level set methods* and finite difference schemes such as upwind differencing can be used for solving the PDE and updating the nodal values of the LSF [7, 13, 14]. On the other hand, on the *parameterized level set method* adopted here, the LSF is considered to be a linear combination of spatially fixed basis functions. In specific, we will consider *radial basis functions* (RBF's). Two basic procedures need then to be defined: the **initialization** of the parameters and their **evolution**. Both will be made following the strategy proposed by [3, 5], as summarized next.

Let $\Sigma = \{\boldsymbol{\xi}^{(i)} = (x_1^{(i)}, x_2^{(i)}), i = 1,\ldots,N\} \subset \mathcal{D}$ be a discretization of the reference domain. Then, $\mathcal{R}(\|\boldsymbol{x} - \boldsymbol{\xi}^{(i)}\|)$ denotes an RBF centered in $\xi^{(i)}$. Assuming that the basis functions are fixed during the evolution process, we define the *parameterized level set function* (PLSF) as $s : D \times \mathbb{R}^+ \mapsto \mathbb{R}$ that satisfies

$$s(\boldsymbol{x},t) = \boldsymbol{r}(\boldsymbol{x})^\intercal \boldsymbol{\alpha}(t) = \sum_{i=1}^{N} \alpha_i(t) \mathcal{R}(\|\boldsymbol{x} - \boldsymbol{\xi}^{(i)}\|). \tag{9}$$

For the parameters **initialization**, we assume that nodal values ϕ_i of the level set are known at each point $\xi^{(i)}$. Then, $\boldsymbol{\alpha}^{(0)}$ is the solution of the interpolation problem $s(\xi^{(i)}) = \phi_i$ for $i = 1,\ldots,N$. In matrix form, this can be expressed as $G\boldsymbol{\alpha} = \Phi$, in which $g_{ij} = \mathcal{R}(\|\boldsymbol{\xi}^{(i)} - \boldsymbol{\xi}^{(j)}\|)$, $i,j = 1,\ldots,N$ and $\Phi = (\phi_1,\ldots,\phi_N)$. Non singularity of G depends on the type of RBF chosen. In [5], authors work mainly with *multiquadrics* (MQ), for which a first-degree polynomial term is added to eq. (9) to ensure positive definiteness [15]. In this work we consider *C2-Wendland's compactly supported* (CS) RBF, defined as

$$\mathcal{R}^{\mathsf{CS}}(\|\boldsymbol{x} - \boldsymbol{\xi}\|) = (\max(0, 1 - r(\boldsymbol{x})))^4 (4r(\boldsymbol{x}) + 1), \quad r(\boldsymbol{x}) = \frac{1}{d_{sp}} \sqrt{\|\boldsymbol{x} - \boldsymbol{\xi}\|^2 + c^2}, \tag{10}$$

where c is a small shape parameter and d_{sp} is the radius of support. In contrast with multiquadric RBF, this choice leads to banded interpolation matrices with bandwidth depending directly on d_{sp}, as shown in Figure 3.

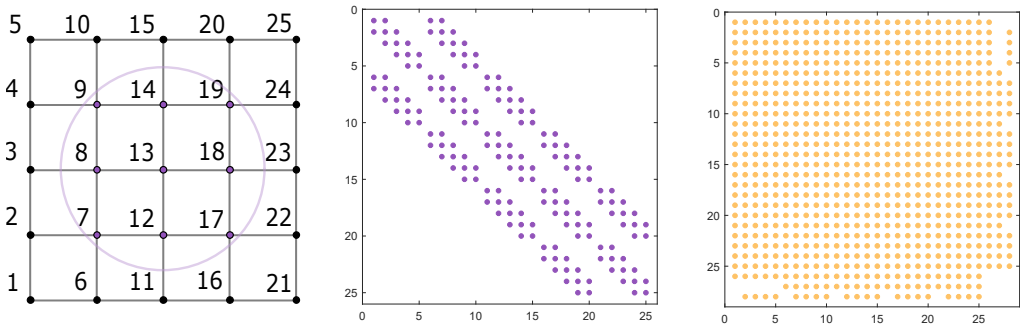

Figure 3. Example with a 5×5 square mesh (left) and the structure of the interpolation matrices associated, respectively, with a CS-RBF with $d_{sp} = 1.5\Delta x$ (where Δx is the element size) – (center) and MQ-RBF's (right).

To express the **evolution** of the PLSF, we may evaluate expression (9) by either eq. (7) or eq. (8). In this work, we consider the latter, which results in

$$\boldsymbol{r}(\boldsymbol{x})^{\intercal}\boldsymbol{\alpha}'(t) + v_n(\boldsymbol{x}(t), t)\|\mathrm{J}_r(\boldsymbol{x})^{\intercal}\boldsymbol{\alpha}(t)\| = 0. \tag{11}$$

By evaluating this expression in each knot $\xi^{(i)}$, which can be considered a collocation formulation of the general method of lines [3], we obtain a set of N ordinary differential equations (ODE) on t which can be written in matrix form as $G\boldsymbol{\alpha}'(t) = \boldsymbol{B}(\boldsymbol{\alpha}, t)$ by defining the j-th component of the right-hand side vector as $B_j(\boldsymbol{\alpha}, t) = v_n(\boldsymbol{\xi}^{(j)}, t)\|\mathrm{J}_r(\boldsymbol{\xi}^{(j)})^{\intercal}\boldsymbol{\alpha}(t)\|$, $j = 1, \ldots, N$. This ODE system can be solved using a first-order forward Euler's method, which gives

$$\boldsymbol{\alpha}^{(k+1)} = \boldsymbol{\alpha}^{(k)} + \Delta t G^{-1}\boldsymbol{B}(\boldsymbol{\alpha}^{(k)}, t). \tag{12}$$

This means that for each evolution step, a linear system involving G must be solved and therefore, the sparsity inherent of the CS-RBF's is valuable for saving computational effort. Moreover, since knots are fixed, this matrix is constant and computational linear algebra can be explored. Additionally, implementation of the gradient term on eq. (12) can be done easily due to the parameterization and in the particular case of CS-RBF's, using sparse matrices. We also consider the approximate re-initialization scheme and the delta function as used by Wei et al. [5] to prevent the LSF to becoming too flat or too steep and to avoid unbounded growth of the LSF.

Choosing the velocity field and extending it appropriately is a key point for the optimization scheme, since we need the motion of the PLSF to be towards a minimizer of the cost functional. In finite dimension, it is known that one classical way of obtaining a descent direction is to take (minus) the gradient of the cost function - the *steepest descent direction* [16]. An analogous approach can be considered by taking the shape derivative of the cost functional (5), which can be defined in terms of the concept of Frechét differentiability of an application. Following [12], let us denote as $d\mathcal{L}_\rho(\Omega, \Lambda; \boldsymbol{\theta})$ the shape derivative of $\mathcal{L}_\rho$ in direction $\boldsymbol{\theta} \in W^{1,\infty}(\mathbb{R}^2, \mathbb{R}^2)$. Since this differential operator is a linear and the chain rule applies [17, §7.3], the shape derivative of eq. (5) can then be computed based on $dJ(\Omega, \boldsymbol{\theta})$ [12, Theorem 7] and $dV(\Omega, \boldsymbol{\theta})$ [12, Remark 6] and results in

$$d\mathcal{L}_\rho(\Omega, \Lambda; \boldsymbol{\theta}) = \int_{\Gamma_0} \mathcal{G}\boldsymbol{\theta} \cdot \hat{\boldsymbol{n}} ds, \quad \text{with } \mathcal{G} = -Ae(\boldsymbol{u}) : e(\boldsymbol{u}) + \frac{\rho}{l_x l_y}\max\{0, \mathcal{C}(\Omega) + \Lambda/\rho\}. \tag{13}$$

Since the evolution scheme proposed requires only the normal component of the field $\boldsymbol{v}$ (cf. eq. (11)), we consider the *Hilbertian extension* approach proposed by Gournay [6], which consists in solving the scalar variational problem of finding $\theta \in H^1(\mathcal{D})$ that satisfies

$$\langle \theta, \beta \rangle_{H^1(\mathcal{D})} = -d\mathcal{L}_\rho(\Omega, \Lambda; \beta\hat{\boldsymbol{n}}),\ \forall \beta \in H^1(\mathcal{D}), \quad \text{where } \langle \theta, \beta \rangle_{H^1(\mathcal{D})} = \int_{\mathcal{D}} (a\nabla\theta \cdot \nabla\beta + \theta\,\beta)\ d\boldsymbol{x}, \tag{14}$$

being $\langle \cdot, \cdot \rangle_{H^1(\mathcal{D})}$ an inner product on $H^1(\mathcal{D})$ and $a \in \mathbb{R}_+$ the extension-regularization parameter. Then, since $d\mathcal{L}_\rho(\Omega, \Lambda, \theta\hat{\boldsymbol{n}}) = -\langle \theta, \theta \rangle_{H^1(\mathcal{D})} < 0$, $v_n = \theta/\|\theta\|_{H^1(\mathcal{D})}$ is a descent direction. As shown in [6], this choice results in a more regular velocity field than the one defined by the *natural extension method*, considered by Wei et al. [5]. Our choice is also facilitated by FEniCS and it is similar to Laurain's [7], that considered the distributed expression of the shape derivative and works with eq. (7) instead of eq. (8). Discussion on different extension procedures can be followed on the review article of Dijk et al. [2].

The optimization scheme adopted is summarized in Algorithm 1. In contrast with the approach of Wei et al. [5], the main differentials of our work are the cost functional definition, the augmented Lagrangian scheme adopted and the choice of the velocity field. On the other hand, from Laurain [7] we inherit details on the solution of PDE's and constrast our parameterized approach with the direct resolution of the HJ equation adopted by the author.

Algorithm 1: Parameterized level set (PLS) topology optimization algorithm

Input: mesh size, volume fraction, initial design domain Ω^0, initial parameters Λ^0, ρ^0.
Solve interpolation problem for initial coefficients $\boldsymbol{\alpha}$.
for *k = 0, . . . , until convergence* **do**
  Step 1: Solve linear elasticity system (eq. (1)) for Ω^k and evaluate cost functional $\mathcal{L}_{\rho^k}(\Omega^k, \Lambda^k)$.
  Step 2: Solve variational problem (eq. (14)) to obtain velocity field θ_n.
  Step 3: Evolve $\boldsymbol{\alpha}$ according to equation eq. (12).
  Step 4: Update Lagrange multiplier Λ^{k+1} and penalty parameter ρ^{k+1}.
end

4 Numerical experiments

We have prepared a `Python` implementation of Algorithm 1. This section contains some preliminary results on two benchmark examples, the symmetric and asymmetric cantilever beam. For the solution of PDE's using finite elements (Steps 1 and 2 of Algorithm 1), `FEniCS` project (`https://FEniCSproject.org/`) was the choice. All the experiments were run in a DELL Inspiron 5584 notebook, Intel Core i7-8565U processor clocked at 1.80GHz, 8GB RAM (64-bit), using Python Version 3.8.5 and `FEniCS` Version 2019.2.0.dev0. For sparse matrix operations, library `Scipy` was used along with `UMFPACK` routines for linear systems. Additionally, `Matlab` Version R2019b was used to run the code provided by Wei et al. [5], for comparative purposes.

We consider $l_x = 2$ and $l_y = 1$ for the working domain $\mathcal{D}$ and a unit force traction vertically loaded at $(l_x, l_y/2)$ for the symmetric case and at $(l_x, 0)$ for the asymmetric case. The material's Young modulus and Poisson's ratio were set to $E = 1$ and $\nu = 0.3$ and $\epsilon = 10^{-9}$ was considered for the ersatz material approach. Also, we consider a prescribed volume fraction of 50% and the level set is initialized as suggested by Wei et al. [5, §3.1], except for a scale factor to adjust to our normalized design domain (cf. Fig. 5 - *Experiment I-a*). As for the augmented Lagrangian scheme, we consider $\Lambda^0 = 30, \rho^0 = 2.0, \rho^{max} = 5 \times 10^3$ and refer to [8, Algorithm 4.1] for details on the updating. Following the book's notation, we set $\gamma = 1.1$ and $\tau = 0.9$. For the inner product of eq. (14), we have considered $a = 0.0005\sqrt{5}$, where $\sqrt{5}$ corresponds to the diameter of the working domain. Experiments with different mesh sizes (N_x, N_y) were performed with the only restriction that $N_x/l_x = N_y/l_y$, so each element is a square. For the finite element analysis, the mesh is refined by partitioning each square by its diagonal into four equal triangular elements and we refer to Laurain [7, §6.2–§6.5] for details.

Figure 4. Cantilever beams: symmetric (left) and asymmetric (right), with associated boundary conditions.

We have first considered the symmetric cantilever beam and compared our results with the ones obtained with the codes of Wei et al. [5] and Laurain [7]. For consistency, we use the same initialization adopted in each work, with a 150×75 mesh for the former (*Experiment I-a*) and 120×60 for the latter (*Experiment I-b*). Laurain's code attained fraction of volume of 0.5423 and compliance of 51.88, with 79 iterations (see [7, Fig. 3(a)]). The 88-line `Matlab` code, with their provided modification to run the CS-RBF took 74 iterations to obtain fraction of volume of 0.5003 and compliance of 6.00E+5; whereas, with the discretization 60×30, it achieved fraction of volume of 0.5004 and compliance of 5.98E+5 with 88 iterations, and a structure qualitatively more similar to ours [5, Fig. 14(a)]). The CS-RBF parameters are fixed as $c = 10^{-4}$ and $d_{sp} = 2\Delta x$. Our convergence check is the same of [7], with relative tolerance of 0.5%, and variation of the eight previous iterations. The results are illustrated in Fig. 5, and detailed in Table 1. The initialization used for *Experiment I-b* is the default considered for our code and for this specific case, the evolution of the optimization parameters are depicted in Fig. 6. For the *Experiment II*, we consider the asymmetric cantilever beam and contrast the performance obtained with different

values of d_{sp} for the CS-RBF. Results are illustrated in Fig. 7 and described on Table 1. In both cases, our runs demanded fewer iterations than the counterparts, what is consistent with Gournay [6], but not as stringent as his results. Nevertheless, better calibration of the stopping condition might avoid the nonregular boundaries of Fig. 7.

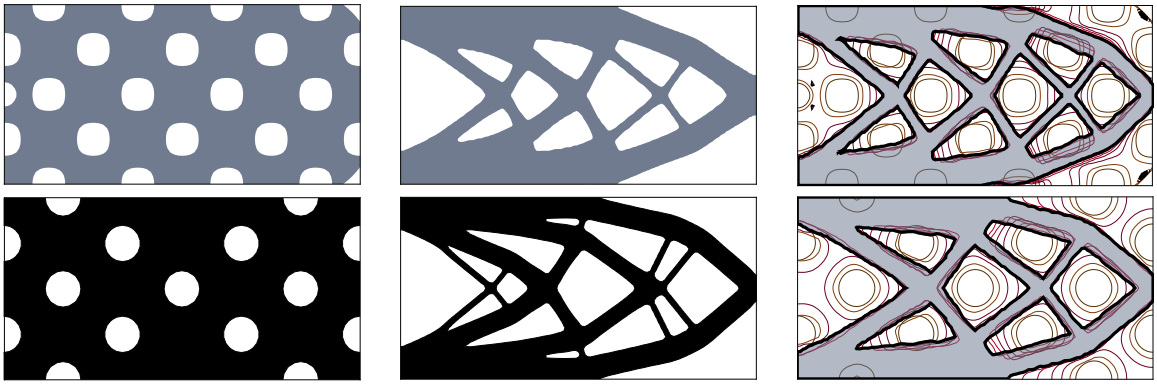

Figure 5. Results obtained on *Experiment I-a* (first row) and *I-b* (second row). Images refer to LSF initialization (first column); final design obtained by reference works (second column – [7] (top) and [5] (bottom)), and design evolution obtained by our code with corresponding initialization and same mesh (third column).

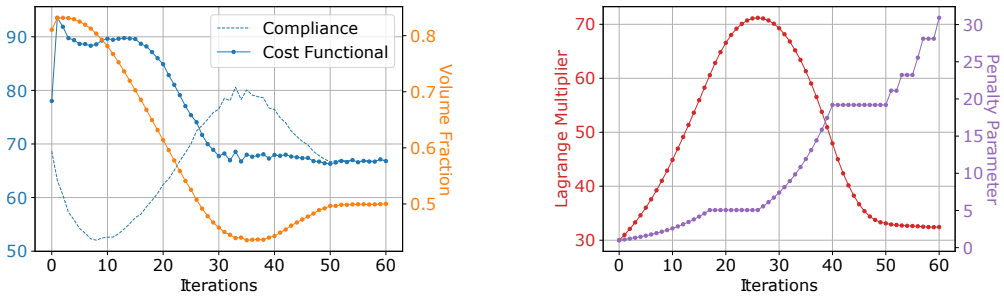

Figure 6. Problem elements (left) and augmented Lagrangian parameters (right) associated with *Experiment I-b*.

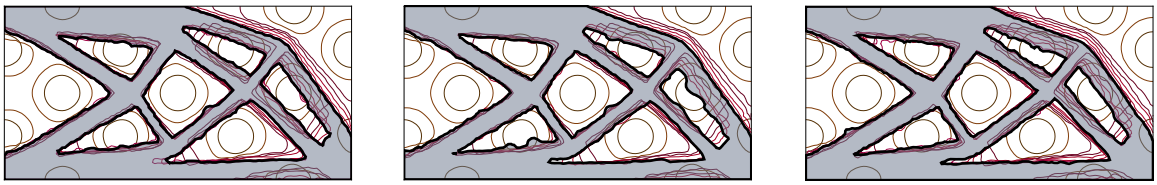

Figure 7. Design domain evolution with $d_{sp} = 2\Delta x$ (left), $d_{sp} = 4\Delta x$ (center) and $d_{sp} = 6\Delta x$ (right)

Table 1. Quantitative results of Experiments I and II.

	Experiment I		*Experiment II*		
	I-a	*I-b*	$d_{sp} = 2\Delta x$	$d_{sp} = 4\Delta x$	$d_{sp} = 6\Delta x$
Iterations	58	60	61	59	64
Fraction of volume	0.4997	0.5002	0.4986	0.5012	0.4990
Compliance	66.37	66.79	76.94	76.50	76.47
Functional cost	66.34	66.82	76.79	76.62	76.36

5 Final remarks

Motivated by the works of Wei et al. [5] and Laurain [7], the contributions of this study are threefold: (i) enhancement of the compatibility of an augmented Lagrangian scheme within the cost functional of the PLS algorithm for topology optimization; (ii) extension of the velocity field by the Hilbertian approach, alternatively to the natural strategy; (iii) implementation in `Python`, employing resources of the `FEniCS` project, of the PLS-based on C2-Wendland RBF's algorithm.

Despite preliminary, our numerical results reveal significant features of the adopted combination of elements, aiming at saving computational effort. For future experimentation, the study of Luo et al. [18] may be valuable concerning the choice of the radius of support of the RBF's. With respect to the Hilbertian extension of the velocity, we have observed a strong influence of the extension-regularization parameter. Its trade-off capability certainly offers room for improvement. The dynamics of the augmented Lagrangian scheme has great impact as well. Further investigations also include varying the initial configuration, refining the mesh and analysing other benchmark problems, such as the half-wheel and the MBB beam.

Acknowledgements. This work was partially supported by FAPESP (2018/24293-0 and 2020/00521-4) and CNPq (305010/2020-4).

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, vol. 79, n. 1, pp. 12–49, 1988.
[2] N. P. Van Dijk, K. Maute, M. Langelaar, and F. Van Keulen. Level-set methods for structural topology optimization: A review. *Structural and Multidisciplinary Optimization*, vol. 48, n. 3, pp. 437–472, 2013.
[3] S. Wang and M. Y. Wang. Radial basis functions and level set method for structural topology optimization. *International Journal for Numerical Methods in Engineering*, vol. 65, n. 12, pp. 2060–2090, 2006.
[4] Z. Luo, M. Y. Wang, S. Wang, and P. Wei. A level set-based parameterization method for structural shape and topology optimization. *International Journal for Numerical Methods in Engineering*, vol. 76, pp. 1–26, 2008.
[5] P. Wei, Z. Li, X. Li, and M. Y. Wang. An 88-line MATLAB code for the parameterized level set method based topology optimization using radial basis functions. *Structural and Multidisciplinary Optimization*, vol. 58, n. 2, pp. 831–849, 2018.
[6] de F. Gournay. Velocity extensin for the level-set method and multiple eigenvalues in shape optimization. *SIAM Journal on Control and Optimization*, vol. 45, n. 1, pp. 343–367, 2006.
[7] A. Laurain. A level set-based structural optimization code using FEniCS. *Structural and Multidisciplinary Optimization*, vol. 58, n. 3, pp. 1311–1334, 2018.
[8] E. G. Birgin and J. M. Martínez. *Practical Augmented Lagrangian Methods for Constrained Optimization*. SIAM, Philadelphia, USA, 2014.
[9] J. N. Reddy. *An Introduction to Continuum Mechanics*. Cambridge University Press, 2nd edition, 2013.
[10] Z. Chen. *Finite Element Methods and Their Applications*. Springer, Berlin Heidelberg, 2005.
[11] R. T. Rockafellar. The multiplier method of hestenes and powell applied to convex programming. *Journal of Optimization Theory and Applications volume*, vol. 12, pp. 555–562, 1973.
[12] G. Allaire, F. Jouve, and A.-M. Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, vol. 194, pp. 363–393, 2004.
[13] S. Osher and R. P. Fedkiw. Level Set Methods: An Overview and Some Recent Results. *Journal of Computational Physics*, vol. 169, n. 2, pp. 463–502, 2001.
[14] V. J. Challis. A discrete level-set topology optimization code written in Matlab. *Structural and Multidisciplinary Optimization*, vol. 41, n. 3, pp. 453–464, 2010.
[15] M. D. Buhmann. *Radial Basis Functions: Theory and Implementations*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, Cambridge, 2003.
[16] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, second edition, 2006.
[17] D. Luenberger. *Optimization by Vector Space Methods*. Professional Series. Wiley, 1997.
[18] Z. Luo, L. Tong, M. Y. Wang, and S. Wang. Shape and topology optimization of compliant mechanisms using a parameterization level set method. *Journal of Computational Physics*, vol. 227, n. 1, pp. 680–705, 2007.