

A LAGRANGIAN SHAPE AND TOPOLOGY OPTIMIZATION FRAMEWORK BASED ON SEMI-DISCRETE OPTIMAL TRANSPORT

C. DAPOGNY¹, B. LEVY² AND E. OUDET³

¹ *Sorbonne Université, Université Paris Cité, CNRS, Inria, Laboratoire Jacques-Louis Lions, LJLL, F-75005 Paris, France.*

² *Centre Inria de Saclay, Université Paris Saclay, Laboratoire de Mathématiques d'Orsay.*

³ *Univ. Grenoble Alpes, CNRS, Grenoble INP¹, LJK, 38000 Grenoble, France,*

ABSTRACT. This article revolves around shape and topology optimization, in the applicative context where the objective and constraint functionals depend on the solution to a physical boundary value problem posed on the optimized domain. We introduce a novel numerical framework based on modern concepts from computational geometry, optimal transport and numerical analysis. Its pivotal feature is a representation of the optimized shape by the cells of an adapted version of a Laguerre diagram. Although such objects are originally described by a collection of seed points and weights, recent results from optimal transport theory suggest a more intuitive parametrization in terms of the seed points and measures of the associated cells. The discrete polygonal mesh of the shape induced by this diagram is a convenient support for the deployment of the Virtual Element Method for the numerical solution of the boundary value problem characterizing the physics at play and for the evaluation of the functionals of the domain involved in the optimization problem. The sensitivities of the latter are derived next; at first, we calculate their derivatives with respect to the positions of the vertices of the Laguerre diagram by shape calculus techniques; a suitable adjoint methodology is then developed to express them in terms of the seed points and cell measures of the diagram. The evolution of the shape through the optimization process is realized by first updating the design variables according to these sensitivities and then reconstructing the diagram thanks to efficient algorithms from computational geometry. Our shape and topology optimization strategy is versatile: it can be applied to a whole gammut of physical situations (such as thermal and structural mechanics) and optimal design settings (including single- and multi-phase problems). It is Lagrangian and body-fitted by essence, and it thereby benefits from all the assets of an explicit, meshed representation of the shape at each stage of the process. Yet, it naturally handles dramatic motions of the optimized shape, including topological changes, in a very robust fashion. These features, among others, are illustrated by a series of 2d numerical examples.

CONTENTS

1. Introduction and related works	2
2. Presentation of the shape optimization problem	6
2.1. Notations	6
2.2. Mathematical models for conductive media and elastic structures	7
2.3. Shape optimization in a nutshell	8
3. Discretization and evolution of shapes using modified Laguerre diagrams	10
3.1. Representation of shapes via Laguerre diagrams	10
3.2. Parametrization of the diagram representation via the seed points and cell measures	13
3.3. General sketch of the Laguerre diagram-based shape and topology optimization method	16
4. Geometric computations on Ω	17
4.1. Calculation of the diagram $\mathbf{V}(\mathbf{s}, \nu)$ associated to seed points \mathbf{s} and cell measures ν	18
4.2. Smoothing of a diagram by a variant of Lloyd's algorithm	24
4.3. Additional operations on diagrams	25
5. Mechanical computations on Ω via the Virtual Element Method	25
5.1. Notations	26

¹Institute of Engineering Univ. Grenoble Alpes

5.2.	The Virtual Element Method for the 2d conductivity equation	26
5.3.	Resolution of the linear elasticity system with the Virtual Element Method	30
6.	Sensitivity of a shape functional with respect to seed points and volume fractions	32
6.1.	Discretization and differentiation of $J(\Omega)$ in terms of the vertices of a diagram for Ω	33
6.2.	Calculation of the derivative of a functional with respect to the positions of the seeds and the cell measures	35
6.3.	The Hilbertian method	36
6.4.	Exploitation of these derivatives with a constrained optimization algorithm	39
7.	Numerical examples	41
7.1.	Optimization of the perimeter of shapes	41
7.2.	Optimization of the eigenvalues of the Dirichlet Laplace operator	42
7.3.	Shape optimization in two-phase conductivity	44
7.4.	Optimization of the shape of an elastic cantilever beam	47
7.5.	Optimization of the shape of a bridge by combination of shape and topological derivatives	50
7.6.	Minimization of the stress within a bridge structure	52
8.	Conclusions and perspectives	53
Appendix A.	Calculation and differentiation of the coordinates of the vertices of a diagram with respect to its seed points and weights	55
Appendix B.	Second order derivatives of the Kantorovich functional with respect to seeds and weights	58
Appendix C.	Proof of Theorem 3.1	62
Appendix D.	Implementation details of the Virtual Element Method	65
D.1.	The case of the conductivity equation	65
D.2.	The case of the linear elasticity system	67
D.3.	Addition of a term of order 0	70
	References	72

1. INTRODUCTION AND RELATED WORKS

Since the early implementations of optimal design, the identification of a means to represent the shape $\Omega \subset \mathbb{R}^d$ ($d = 2$ or 3 in practice) which is amenable to all the operations of the numerical resolution has been a recurring issue – especially when physical applications are addressed, where the behavior of the optimized shape is modeled by boundary value problems, such as the conductivity or the linear elasticity equations. It is indeed notoriously difficult to reconcile accurate simulations of these boundary value problems with a robust description of the evolution of Ω through the optimization process. The historical “Lagrangian” approaches for shape optimization hinge on a (simplicial) mesh of Ω which is updated between the iterations of the workflow; while this representation naturally offers a convenient support for mechanical computations via the Finite Element Method, it makes it difficult to track the deformations of Ω in a robust way, especially when they include topological changes [90, 105]. More recent solutions to carry out this thorny evolution operation resort to implicit, Eulerian representations of the shape Ω . The most popular practice in this category is certainly to use the level set method [8, 101, 116, 130]. The latter brings into play a fixed mesh \mathcal{T} of a large “hold-all” domain D : the optimized design $\Omega \subset D$ is encoded into a so-called “level set function” $\phi : D \rightarrow \mathbb{R}$, discretized on \mathcal{T} in practice, which is negative (resp. positive) inside Ω (resp. inside $D \setminus \bar{\Omega}$). Arbitrary updates of Ω are accounted for via “simple” updates of ϕ – for instance, small deformations of the boundary $\partial\Omega$ translate into an advection or a Hamilton-Jacobi equation for ϕ . Unfortunately, this flexibility has a price: since no mesh of the shape Ω is available, physical boundary value problems posed on Ω have to be replaced by approximate counterparts, posed on the total domain D , which are solved on the fixed mesh \mathcal{T} . This practice often leverages a fictitious domain method, such as the ersatz material approximation in structural mechanics, see e.g. [3]; in any event, it results in an inaccurate numerical solution. The main competing Eulerian strategy for optimal design, the so-called phase-field method, shares fairly similar spirit, assets and drawbacks [22, 26, 126]. On the extreme side of this line of thinking, density-based topology

optimization methods [17], such as the popular Solid Isotropic Material with Penalization (SIMP) method in structural mechanics (see also [25] about artificial porosity methods in the context of fluid mechanics), downright relax the optimal design problem in terms of a “blurred” version ρ of the characteristic function of the optimized design, defined on (the mesh \mathcal{T} of) D and taking values in the interval $[0, 1]$. Intuitively, ρ encodes the “local density” of material: regions where $\rho(\mathbf{x}) \equiv 1$ (resp. $\rho(\mathbf{x}) \equiv 0$) are full (resp. devoid) of material and the intermediate values $\rho(\mathbf{x}) \in (0, 1)$ represent regions filled by a microscopic arrangement of material and void in proportions $\rho(\mathbf{x})$ and $(1 - \rho(\mathbf{x}))$. The physical problem under scrutiny is approximated and reformulated to accommodate such a description of the design, notably by endowing regions where ρ takes intermediate values with fictitious material properties. Last but not least, and without entering into details, let us eventually mention that several of the aforementioned approaches have been combined, see e.g. [4, 5, 6] for a concurrent use of the level set method with simplicial remeshing algorithms, or [36] for a quite sophisticated Lagrangian method using remeshing.

The present article introduces a novel framework for shape and topology optimization which is based on concepts and techniques from computational geometry and optimal transport. This construction elaborates on the previous work [83] of one of the authors, taking place in the context of computational fluid dynamics, and on its recent applications [77, 96, 97] in the field of cosmology, see also [45] about related ideas. The cornerstone of our strategy is a representation of the optimized shape Ω by the cells of a (modified version of a) Laguerre diagram, a generalization of the well-known notion of Voronoi diagram in computational geometry [14, 24]. Although the natural definition of such objects involves a collection $\mathbf{s} = \{\mathbf{s}_1, \dots, \mathbf{s}_N\} \in \mathbb{R}^{dN}$ of seed points and associated weights $\boldsymbol{\psi} = \{\psi_1, \dots, \psi_N\} \in \mathbb{R}^N$, we leverage recent results from optimal transport theory to parametrize them by the seed points \mathbf{s} and the measures $\boldsymbol{\nu} = \{\nu_1, \dots, \nu_N\} \in \mathbb{R}^N$ of the associated cells. In the present applications, the shape Ω is iteratively optimized with respect to an objective and constraint functions $J(\Omega)$ and $\mathbf{G}(\Omega)$. Their evaluation and that of their derivatives, which in turn determine the velocity field governing the evolution of Ω through the process, raise the need to solve one or several boundary value problems posed on Ω ; this is a major additional difficulty with respect to the aforementioned works [83, 77, 96, 97], where heuristic simplifications made it possible to express the velocity field for Ω in terms of geometric quantities of the diagram only. In order to solve these boundary value problems, we take advantage of the tessellation of Ω into convex polytopes induced by its diagram representation; this offers a convenient support for the use of the Virtual Element Method – a recent variant of the classical Finite Element Method adapted to polygonal meshes, see e.g. [12]. The calculation of the sensitivities of the optimization criteria $J(\Omega)$ and $\mathbf{G}(\Omega)$ with respect to the seed points \mathbf{s} and cell measures $\boldsymbol{\nu}$ of the diagram is realized in two steps: we first apply shape calculus techniques to compute their sensitivities with respect to the vertices of the diagram and we next employ a suitable adjoint method to reformulate them in terms of \mathbf{s} and $\boldsymbol{\nu}$. A descent direction for the optimization problem is then inferred from this information, and the parameters \mathbf{s} and $\boldsymbol{\nu}$ of the Laguerre diagram for Ω are updated; the new diagram associated to the resulting seed points and weights is generated, revealing the next iterate of the shape Ω . This strategy is Lagrangian by nature since the evolution of Ω is realized by tracking the motion of the defining seed points; it consistently features an exact, body-fitted description of the shape Ω . In the meantime, it leaves the room for a robust account of dramatic deformations between iterations, including topological changes: the deformations of (the mesh of) Ω are never tracked explicitly; rather, the shape is rediscovered after each iteration, once the defining Laguerre diagram has been calculated anew.

As reflected by the previous discussion, one originality of our approach is that it leverages recent concepts and techniques pertaining to quite different fields: shapes are represented via Laguerre diagrams, a notion from computational geometry and optimal transport; the Virtual Element Method from numerical analysis is used for the solution of the physical boundary value problems at play on polygonal meshes; eventually, fine techniques from shape and topology optimization are used to orchestrate the optimal design process.

To the best of our knowledge, the present work is the first one using all these ingredients together even though, admittedly some of them have already been combined in the literature. Voronoi-like diagrams, polygonal meshes, and numerical methods dedicated to this type of structures have been considered in the perspective of optimal design or more generally interface motion tracking, as we now briefly overview, without exhaustivity. In [61], the cells of a Voronoi diagram are used to represent the many individual entities of a granular medium. The collection is subjected to the viscous, elastic and contact forces between cells, which are modeled by a finite-dimensional system of ordinary differential equations. The contributions

[123, 124] arise in the field of fluid dynamics; the 2d polygonal mesh induced by a Voronoi diagram is used as computational support for the finite volume calculations involved in the simulation of the flow, notably for the reconstruction of the fluxes at the interface between cells. Although this discretization is not fitted with the surface of the fluid, the defining seed points evolve according to the flow velocity. A high-order version of this strategy was proposed in [59], see also [51] for another contribution in this setting. Still in the context of fluid mechanics, the article [45] associates a Laguerre diagram to a collection of fluid particles; its weights are adjusted to take into account the compressibility or incompressibility of the fluid. A simple, approximate numerical solver based on discrete differential operators allows to compute the velocity of the fluid. This framework has been recently revisited in [108], where entropy-regularized transport plans are used to ease the calculation of the Laguerre cell attached to each particle. It can be adapted to the case of free surfaces, owing to a modified version of this entropy-regularized optimal transport formulation and of the Sinkhorn algorithm used to calculate the transport plan. Voronoi diagrams appear more systematically as the pivotal ingredient of the so-called Voronoi Implicit Interface Method introduced in [113, 114], see also [134] for a mathematical analysis in the context of multiphase motion driven by curvature. Briefly, a partition of a fixed domain into multiple phases is represented via the Voronoi diagram generated by a set of seed objects (and not mere seed points). The evolution of these phases is achieved by first calculating the domains induced by the ε -offset of the collection of edges (in 2d) or faces (in 3d) of the diagram, then by realizing their evolution via the “classical” level set method. The updated collection of phases is retrieved as the Voronoi diagram of the resulting objects. Beyond the field of computational physics, Voronoi diagrams have inspired popular formats for shapes in applicative disciplines such as computer graphics, see e.g. the article [2] where the motion of a shape is tracked by introducing a clipped Voronoi tessellation of a template shape, whose cells are rigidly deformed to match a target.

Closer to the setting of the present article, the specific features of Voronoi diagrams have inspired various original representations of the shape in optimal design. [To our knowledge, the first numerical model connecting the notion of optimal transport to geometric optimization is the article \[31\] about reflector optimization, see \[43\] for an elegant numerical implementation of this formulation.](#) The recent contribution [53] is dedicated to trusses, i.e. networks of bars. The considered structures are represented by the edges of a Voronoi diagram contained in a fixed “hold-all” domain $D \subset \mathbb{R}^d$; when it comes to calculating their elastic behavior, a density function $\rho : D \rightarrow [0, 1]$ is generated, featuring a continuous interpolation between the value 1 inside the bars and that 0 in void. This function is parametrized by the seed points of the diagram, and the optimal design problem is solved in terms of these variables. A quite similar strategy is used in the article [80], dealing with the optimization of foams.

The use of a polygonal computational mesh of the shape Ω or a larger hold-all domain D is not extraneous to the field of shape and topology optimization, either. For instance, the contributions [127, 11, 34] rely on a density topology optimization method, where the hold-all domain D is equipped with a fixed polygonal mesh serving as the support for all the mechanical computations of the workflow, which are performed by means of the Virtual Element Method. The added value of the use of the Virtual Element Method on a general polygonal mesh with respect to the more classical practice of the Finite Element Method on a simplicial mesh hinges on its reported superior stability properties. To the best of our knowledge, only a few recent contributions leverage the greater flexibility of general polygonal meshes over simplicial meshes to propose shape and topology optimization methods featuring a body-fitted discretization of the optimized shape. In [94, 95], the authors rely on the level set method; a level set function $\phi : D \rightarrow \mathbb{R}$ for the shape Ω is defined on a fixed hexahedral mesh of the computational domain D and at each iteration, the 0 isosurface of ϕ is explicitly discretized in this mesh, thus revealing a polygonal mesh of the shape. Mechanical computations are performed on this computational support for Ω thanks to a moving least-square approximation, which allows to calculate the shape derivative of the minimized function of the domain. A similar strategy is used in [57], where a polygonal mesh of the computational domain D is used and the level set function $\phi : D \rightarrow \mathbb{R}$ for the shape Ω is represented as a discontinuous Finite Element function. Again, at each iteration of the process, the 0 level set of ϕ for the shape Ω is explicitly discretized in the mesh of D , which results in a new, non conforming polygonal mesh of the latter, on which Finite Element analyses can be conducted by a discontinuous Galerkin solver. The topological derivative of the minimized function of the domain is calculated and the level set function ϕ is updated from this information. Let us eventually mention the related, very recent work [56] where a very fine simplicial mesh of D is used to update the level set function,

and a coarse polygonal mesh is generated when it comes to mechanical computations, whose elements are agglomerations of the simplices of the fine mesh.

To conclude this brief overview of related works, let us point out that, from the conceptual viewpoint, our shape and topology optimization framework shares similarities with that developed in [74, 20]. In there, a decomposition of the computational domain D into K phases is described by a so-called minimization diagram. The latter is attached to a collection $\{\phi_k\}_{k=1,\dots,K}$ of scalar functions $\phi_k : D \rightarrow \mathbb{R}$, in such a way that each phase $k = 1, \dots, K$ is made of those points $\mathbf{x} \in D$ where $\phi_k(\mathbf{x})$ is minimum among $\phi_1(\mathbf{x}), \dots, \phi_K(\mathbf{x})$. The motion of the boundary between each phase is accounted for by a quite similar mechanism as in the context of the level set method. We also mention the recent contribution [21] about the reconstruction of a potential in an inverse problem, which is piecewise constant with respect to the cells of a Voronoi diagram.

The shape and topology optimization paradigm introduced in this article is recent, and it is still at the stage of inception. It does not have the level of maturity of well-established frameworks, such as the aforementioned density-based and level set methods. In particular, its implementation and use are perhaps more delicate, as its main numerical ingredients – the solution of semi-discrete optimal transport problems, the construction of Laguerre diagrams, the solution of boundary-value problems by the Virtual Element Method – are not yet available as efficient, open-source libraries. Yet, we believe that it is an original, elegant and promising framework for shape and topology optimization, that offers unique advantages, notably a fully differentiable representation of shapes with respect to design parameters, the ability to track arbitrarily large deformations in a body-fitted Lagrangian setting... The present article focuses on the exposition of our shape and topology optimization framework in two space dimensions, where its main ingredients can be conveniently described with a minimum level of technicality and its efficiency can be appraised on benchmark examples from the literature. There is no conceptual obstruction to the three-dimensional extension of this work, but only an increase in the implementation burden. Indeed, the chief tools involved – notably, the calculation of Laguerre diagrams and the Virtual Element Method – are already available in this situation. To emphasize this fact, our presentation takes place in the general, d dimensional case whenever it is possible without jeopardizing with clarity. On a different note, as we have already noted, our strategy leverages concepts and algorithms pertaining to disciplines which seldom appear together in the literature: computational geometry, via the handling of Laguerre diagrams, numerical analysis for the simulation of physical phenomena (and notably, the Virtual Element Method), and optimal design for the notions of shape and topological derivatives, the adjoint method, etc. In order to ensure a relatively self-contained presentation, we provide a reasonable amount of background material about these ingredients, including proofs when they help intuition or when they are not readily available in the precise context of interest. Our implementation essentially relies on open-source libraries, and we aim to make it available in a short future, together with associated tutorials. We believe that the methods introduced in this work will find useful applications beyond shape optimization, e.g. in computational physics.

The remainder of this article is organized as follows. The next Section 2 sets the scene of the shape optimization problems analyzed in this article. In particular, we introduce the main notation used throughout and recall a few basic notions about shape optimization. Section 3 is devoted to the numerical representation of our shapes in terms of the seed points and cell measures of a Laguerre diagram; the general shape optimization workflow adopted in this context is then presented. The subsequent Sections 4 to 6 are more in-depth presentations of the main operations involved therein. Section 4 deals with the geometric computations attached to Laguerre diagrams involved in our framework, and notably their construction, cell regularization, resampling, etc. Section 5 sketches the implementation of the Virtual Element Method for the solution of the conductivity equation and the linear elasticity system. Eventually, Section 6 details the calculation of the objective and constraint functionals featured in our shape optimization problem and their sensitivities with respect to the parameters of our shape representation, namely the seed points and cell measures of the representing diagram. A series of numerical examples is presented and discussed in Section 7 to showcase the main features of our approaches. In Section 8, we draw a few conclusions about this strategy and we outline natural perspectives for future work. The article ends with a series of appendices, devoted to the proofs of some technical results which are not completely new, but are adaptations of existing results, or known facts which are not easily found under the needed form in the literature. It also contains several useful implementation details.

2. PRESENTATION OF THE SHAPE OPTIMIZATION PROBLEM

Our main purpose in this article is to optimize shapes, that is, bounded, Lipschitz domains Ω of \mathbb{R}^d , where $d = 2, 3$ in practice. We consider optimization problems of the general form:

$$(P) \quad \min_{\Omega} J(\Omega) \quad \text{s.t.} \quad \mathbf{G}(\Omega) = 0,$$

where $J(\Omega)$ is an objective criterion, and $\mathbf{G}(\Omega) := (G_1(\Omega), \dots, G_p(\Omega))$ is a collection of p scalar-valued equality constraint functionals.

Elementary examples of shape functionals are the volume $\text{Vol}(\Omega)$ and the perimeter $\text{Per}(\Omega)$, defined by:

$$(2.1) \quad \text{Vol}(\Omega) = \int_{\Omega} dx, \quad \text{and} \quad \text{Per}(\Omega) = \int_{\partial\Omega} ds.$$

More involved instances about $J(\Omega)$ and $\mathbf{G}(\Omega)$ depend on the physical behavior of Ω via a so-called state function u_{Ω} , which is the solution to a boundary value problem posed on Ω .

After setting a few notations about calculus in [Section 2.1](#), we describe in [Section 2.2](#) the applicative situations considered in the present article, namely those of conductive media and of elastic structures. Eventually, in [Section 2.3](#), we recall a few facts about differentiation with respect to the domain.

2.1. Notations

Throughout this article, vectors \mathbf{x} in a Euclidean space \mathbb{R}^N appear in bold face, and their components are denoted by $\mathbf{x} = (x_1, \dots, x_N)$. Moreover, we rely on the following conventions:

- For $N \geq 1$, the Euclidean space \mathbb{R}^N is equipped with its canonical basis $\{\mathbf{e}_i\}_{i=1, \dots, N}$. The attached Euclidean inner product is denoted by $\langle \cdot, \cdot \rangle$, and sometimes only with a \cdot , and the norm is denoted by $|\cdot|$. We also introduce the supremum norm:

$$|\mathbf{x}|_{\infty} := \sup_{i=1, \dots, N} |x_i|, \quad \mathbf{x} \in \mathbb{R}^N.$$

- The open ball in \mathbb{R}^N with center $\mathbf{x} \in \mathbb{R}^N$ and radius $r > 0$ is denoted by $B(\mathbf{x}, r)$.
- Let $f : \mathbb{R}^N \rightarrow \mathbb{R}$ be a (smooth enough) scalar function and let $\mathbf{x} \in \mathbb{R}^N$; we denote by $\frac{\partial f}{\partial \mathbf{x}}(\mathbf{x})$ the derivative of f at \mathbf{x} , i.e. $\frac{\partial f}{\partial \mathbf{x}}(\mathbf{x}) : \mathbb{R}^N \rightarrow \mathbb{R}$ is the linear mapping giving rise to the expansion:

$$f(\mathbf{x} + \widehat{\mathbf{x}}) = f(\mathbf{x}) + \frac{\partial f}{\partial \mathbf{x}}(\mathbf{x})(\widehat{\mathbf{x}}) + o(\widehat{\mathbf{x}}), \quad \text{where} \quad \frac{o(\widehat{\mathbf{x}})}{|\widehat{\mathbf{x}}|} \rightarrow 0 \quad \text{as} \quad \widehat{\mathbf{x}} \rightarrow 0.$$

The gradient $\nabla f(\mathbf{x}) \in \mathbb{R}^N$ of f at \mathbf{x} is the Riesz representative of this linear mapping, that is:

$$\forall \widehat{\mathbf{x}} \in \mathbb{R}^N, \quad \langle \nabla f(\mathbf{x}), \widehat{\mathbf{x}} \rangle = \frac{\partial f}{\partial \mathbf{x}}(\mathbf{x})(\widehat{\mathbf{x}}).$$

When needed, we shall explicitly indicate in the operator $\nabla_{\mathbf{x}}$ the variables \mathbf{x} involved in the derivative.

- Still considering a (smooth enough) scalar function $f : \mathbb{R}^N \rightarrow \mathbb{R}$, the Hessian matrix $[\nabla^2 f(\mathbf{x})]$ of f at \mathbf{x} is the $N \times N$ matrix with entries:

$$[\nabla^2 f(\mathbf{x})]_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}(\mathbf{x}), \quad i, j = 1, \dots, N.$$

- For $M, N \geq 1$, the derivative of a smooth enough vector-valued function $\boldsymbol{\psi} = (\psi_1, \dots, \psi_M) : \mathbb{R}^N \rightarrow \mathbb{R}^M$ is the linear mapping $\frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}}(\mathbf{x}) : \mathbb{R}^N \rightarrow \mathbb{R}^M$ featured in the expansion

$$\boldsymbol{\psi}(\mathbf{x} + \widehat{\mathbf{x}}) = \boldsymbol{\psi}(\mathbf{x}) + \frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}}(\mathbf{x})(\widehat{\mathbf{x}}) + o(\widehat{\mathbf{x}}), \quad \text{where} \quad \frac{o(\widehat{\mathbf{x}})}{|\widehat{\mathbf{x}}|} \rightarrow 0 \quad \text{as} \quad \widehat{\mathbf{x}} \rightarrow 0.$$

The $M \times N$ matrix $\nabla \boldsymbol{\psi}(\mathbf{x})$ associated to this derivative – sometimes also denoted by $[\nabla \boldsymbol{\psi}(\mathbf{x})]$ for clarity – is that with entries:

$$[\nabla \boldsymbol{\psi}(\mathbf{x})]_{ij} = \frac{\partial \psi_i}{\partial x_j}(\mathbf{x}), \quad i = 1, \dots, M, \quad j = 1, \dots, N.$$

As a result, the following relation holds true:

$$\forall \widehat{\mathbf{x}} \in \mathbb{R}^N, \quad \frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}}(\mathbf{x})(\widehat{\mathbf{x}}) = [\nabla \boldsymbol{\psi}(\mathbf{x})] \widehat{\mathbf{x}},$$

where the right-hand side features the matrix-vector product between the $M \times N$ matrix $[\nabla\psi(\mathbf{x})]$ and the vector $\widehat{\mathbf{x}} \in \mathbb{R}^N$. Note the (classical and hopefully harmless) ambiguity of these notations in the case $M = 1$ where $\nabla\psi(\mathbf{x})$ is a vector with size N and $[\nabla\psi(\mathbf{x})]$ is the transpose $1 \times N$ matrix.

More specific notations will be introduced when needed, later in the article.

2.2. Mathematical models for conductive media and elastic structures

In this section, we introduce the two physical applications of our shape optimization problems. We refer to classical treatises about continuum mechanics such as [128] for further details, see also [66] about linear elasticity.

2.2.1. Optimal design of conductive media

Let the shape Ω stand for an electric conductor in \mathbb{R}^d ($d = 2$ or 3). It is filled by a material with smooth conductivity $\gamma \in \mathcal{C}^\infty(\mathbb{R}^d)$, satisfying the following ellipticity condition:

$$\exists 0 < \underline{\gamma} \leq \bar{\gamma} < \infty \quad \text{s.t.} \quad \forall \mathbf{x} \in \mathbb{R}^d, \quad \underline{\gamma} \leq \gamma(\mathbf{x}) \leq \bar{\gamma}.$$

The boundary $\partial\Omega$ is maintained at fixed potential 0. Introducing a source $f \in \mathcal{C}^\infty(\mathbb{R}^d)$, the voltage potential u_Ω inside Ω is the unique solution in $H_0^1(\Omega)$ to the following boundary value problem:

$$(2.1) \quad \begin{cases} -\operatorname{div}(\gamma \nabla u_\Omega) = f & \text{in } \Omega, \\ u_\Omega = 0 & \text{on } \partial\Omega. \end{cases}$$

The associated variational formulation reads:

$$(2.2) \quad \text{Search for } u_\Omega \in H_0^1(\Omega) \quad \text{s.t.} \quad \forall v \in H_0^1(\Omega), \quad \int_\Omega \gamma \nabla u_\Omega \cdot \nabla v \, d\mathbf{x} = \int_\Omega f v \, d\mathbf{x}.$$

In this context, a typical objective function measuring the performance of Ω is the compliance

$$(2.3) \quad C(\Omega) = \int_\Omega \gamma |\nabla u_\Omega|^2 \, d\mathbf{x} = \int_\Omega f u_\Omega \, d\mathbf{x},$$

which appraises the energy stored inside the shape Ω , or equivalently, the work done by the source. One may also be interested in minimizing the first eigenvalue λ_Ω^1 (resp. the k^{th} eigenvalue λ_Ω^k) of the operator $-\operatorname{div}(\gamma \nabla \cdot)$ on Ω equipped with Dirichlet boundary conditions: λ_Ω^1 is the smallest positive number λ (resp. the k^{th} smallest positive number) for which there exists a non trivial function $u \in H_0^1(\Omega)$ satisfying

$$(2.4) \quad \begin{cases} -\operatorname{div}(\gamma \nabla u) = \lambda u & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega. \end{cases}$$

From the physical viewpoint, λ_Ω^1 accounts for the minimum energy of a potential u with unit squared amplitude, as reflected by the classical Rayleigh quotient formula:

$$(2.5) \quad \lambda_\Omega^1 = \min_{\substack{u \in H_0^1(\Omega), \\ u \neq 0}} \frac{\int_\Omega \gamma |\nabla u|^2 \, d\mathbf{x}}{\int_\Omega u^2 \, d\mathbf{x}}.$$

Remark 2.1. We shall also consider a two-phase variant of this setting, where the shape Ω represents one phase made of a material with conductivity $\gamma_1 \in \mathcal{C}^\infty(\mathbb{R}^d)$ within a fixed domain D made of a material with different conductivity $\gamma_0 \in \mathcal{C}^\infty(\mathbb{R}^d)$. One then aims to optimize the repartition $D = \Omega \cup (D \setminus \bar{\Omega})$ of both materials within D with respect to, e.g. the compliance of the total device D ; see [Section 7.3](#) for a numerical example in this setting.

2.2.2. Optimization of the shape of elastic structures

In this section, $\Omega \subset \mathbb{R}^d$ stands for a mechanical structure, whose boundary is divided into three disjoint pieces:

$$\partial\Omega = \overline{\Gamma_D} \cup \overline{\Gamma_N} \cup \overline{\Gamma}.$$

In this decomposition,

- The shape Ω is clamped on Γ_D ;

- Surface loads $\mathbf{g} : \Gamma_N \rightarrow \mathbb{R}^d$ are applied on Γ_N ;
- The region Γ is free of applied efforts.

In practice, Γ_D and Γ_N are often imposed by the context, so that only the free boundary Γ is subject to optimization. Introducing (smooth) body forces $\mathbf{f} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ representing e.g. gravity, the displacement \mathbf{u}_Ω of Ω belongs to the space

$$H_{\Gamma_D}^1(\Omega)^d := \{\mathbf{u} \in H^1(\Omega)^d, \mathbf{u} = \mathbf{0} \text{ on } \Gamma_D\}.$$

It is the unique solution in the latter to the linearized elasticity system:

$$(Elas) \quad \begin{cases} -\operatorname{div}(Ae(\mathbf{u}_\Omega)) = \mathbf{f} & \text{in } \Omega, \\ \mathbf{u}_\Omega = \mathbf{0} & \text{on } \Gamma_D, \\ Ae(\mathbf{u}_\Omega)\mathbf{n} = \mathbf{g} & \text{on } \Gamma_N, \\ Ae(\mathbf{u}_\Omega)\mathbf{n} = \mathbf{0} & \text{on } \Gamma. \end{cases}$$

Here, $e(\mathbf{u}) := \frac{1}{2}(\nabla\mathbf{u} + \nabla\mathbf{u}^T)$ is the linearized strain tensor induced by a vector field $\mathbf{u} : \Omega \rightarrow \mathbb{R}^d$. The material properties of the elastic medium are encoded in the Hooke's tensor A , which reads:

$$\text{For all symmetric matrix } \xi \in \mathbb{R}^{d \times d}, \quad A\xi = 2\mu\xi + \lambda\operatorname{tr}(\xi)\mathbf{I},$$

where $\lambda > 0$ and $\mu > 0$ are the Lamé coefficients of the material. The boundary value problem (Elas) rewrites, under variational form:

$$(2.6) \quad \text{Search for } \mathbf{u}_\Omega \in H_{\Gamma_D}^1(\Omega)^d \text{ s.t. } \forall \mathbf{v} \in H_{\Gamma_D}^1(\Omega)^d, \quad \int_{\Omega} Ae(\mathbf{u}_\Omega) : e(\mathbf{v}) \, dx = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, dx + \int_{\Gamma_N} \mathbf{g} \cdot \mathbf{v} \, ds.$$

The shape optimization problem (P) under scrutiny in the present context of mechanical engineering may for instance consist in minimizing the compliance $J(\Omega) = C(\Omega)$ of the structure as a means to maximize its rigidity,

$$(2.7) \quad C(\Omega) := \int_{\Omega} Ae(\mathbf{u}_\Omega) : e(\mathbf{u}_\Omega) \, dx = \int_{\Omega} \mathbf{f} \cdot \mathbf{u}_\Omega \, dx + \int_{\Gamma_N} \mathbf{g} \cdot \mathbf{u}_\Omega \, ds,$$

under a volume constraint $G(\Omega) = \operatorname{Vol}(\Omega) - V_T$, where V_T is a volume target.

Remark 2.2. *For simplicity of the presentation, we have assumed that all the data of the models of the previous and present sections (the domain Ω , the forces \mathbf{f} , \mathbf{g} , etc.) are smooth. This framework is by no means the minimal one guaranteeing the well-posedness of these models, and the validity of our developments.*

2.3. Shape optimization in a nutshell

The mathematical analysis and numerical resolution of a shape optimization problem of the form (P) hinges on the sensitivities of the objective and constraint functions $J(\Omega)$ and $\mathbf{G}(\Omega)$ with respect to the domain Ω – a notion which can be appraised in various ways. In this work, we mainly rely on the boundary variation method of Hadamard, see for instance [9, 69, 91, 119]. The latter is based on variations of a given shape $\Omega \subset \mathbb{R}^d$ of the form:

$$\Omega_\theta := (\operatorname{Id} + \boldsymbol{\theta})(\Omega), \quad \boldsymbol{\theta} \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d), \quad \|\boldsymbol{\theta}\|_{W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)} < 1.$$

Loosely speaking, Ω_θ is obtained by moving the points of Ω according to the “small” vector field $\boldsymbol{\theta}$, see Fig. 1.

One function $F(\Omega)$ of the domain is called shape differentiable at a particular shape Ω when the underlying mapping $\boldsymbol{\theta} \mapsto F(\Omega_\theta)$, defined from a neighborhood of $\mathbf{0}$ in $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ into \mathbb{R} , is Fréchet differentiable at $\mathbf{0}$. Its derivative $\boldsymbol{\theta} \mapsto F'(\Omega)(\boldsymbol{\theta})$ is called the shape derivative of F at Ω and it satisfies the following expansion:

$$F(\Omega_\theta) = F(\Omega) + F'(\Omega)(\boldsymbol{\theta}) + o(\boldsymbol{\theta}), \quad \text{where } \frac{o(\boldsymbol{\theta})}{\|\boldsymbol{\theta}\|_{W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)}} \xrightarrow{\boldsymbol{\theta} \rightarrow \mathbf{0}} 0.$$

Intuitively, a “small” perturbation of Ω following a “descent direction” $\boldsymbol{\theta}$, i.e. a deformation such that $F'(\Omega)(\boldsymbol{\theta}) < 0$, for a short pseudo-time step $\tau > 0$ produces a shape $\Omega_{\tau\boldsymbol{\theta}}$ with a “better” performance (i.e. a lower value), as measured in terms of F :

$$F(\Omega_{\tau\boldsymbol{\theta}}) \approx F(\Omega) + \tau F'(\Omega)(\boldsymbol{\theta}) < F(\Omega).$$

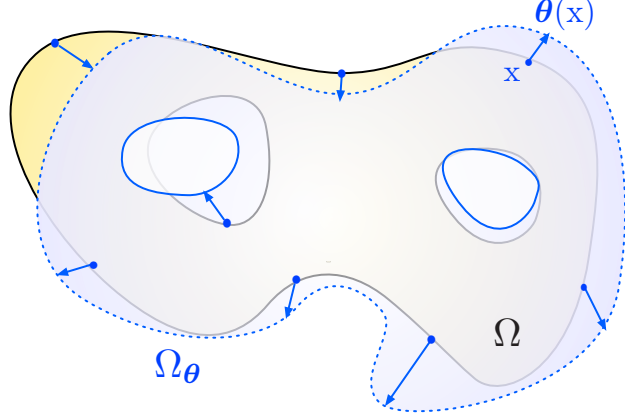


FIGURE 1. Variation Ω_θ of a shape Ω in the sense of Hadamard's method.

Simple examples of shape derivatives are those of the volume and perimeter functionals defined in (2.1). Under mild assumptions that we omit for brevity, these read:

$$(2.8) \quad \text{Vol}'(\Omega)(\theta) = \int_{\Omega} \text{div}(\theta) \, dx = \int_{\partial\Omega} \theta \cdot \mathbf{n} \, ds, \quad \text{and} \quad \text{Per}'(\Omega)(\theta) = \int_{\partial\Omega} \kappa \theta \cdot \mathbf{n} \, ds,$$

where $\kappa : \partial\Omega \rightarrow \mathbb{R}$ is the mean curvature of $\partial\Omega$.

The analysis of the “physical” functionals of interest in this article is more intricate, as they involve the solution u_Ω to a boundary value problem posed on Ω , see e.g. (2.3) and (2.7). Nevertheless, the classical adjoint method allows to calculate their shape derivatives in closed form, in terms of u_Ω and possibly of an adjoint state p_Ω , solution to a boundary value problem similar to that for u_Ω . Although it is non trivial, this technical issue is well-known in the literature, and we do not enter into details, referring to e.g. [7, 81], and also [106] for a comprehensive introduction.

Remark 2.3. *The adjoint method will be deployed (and detailed) in Section 6.2 below to achieve another purpose, that of converting a derivative with respect to the vertices of a Laguerre diagram into derivatives with respect to its seed points and cell measures.*

For further reference, let us note that the shape derivative of a generic shape functional $F(\Omega)$ has often two different, albeit equivalent structures:

- A volume form

$$(2.9) \quad F'(\Omega)(\theta) = \int_{\Omega} (\mathbf{t}_\Omega \cdot \theta + S_\Omega : \nabla \theta) \, dx,$$

featuring an integral over the domain Ω of the deformation θ and its derivative, and some vector- and matrix-valued functions $\mathbf{t}_\Omega : \rightarrow \mathbb{R}^d$ and $S_\Omega : \Omega \rightarrow \mathbb{R}^{d \times d}$ depending on the function $F(\Omega)$.

- A surface form

$$(2.10) \quad F'(\Omega)(\theta) = \int_{\partial\Omega} v_\Omega \theta \cdot \mathbf{n} \, ds,$$

involving a scalar field $v_\Omega : \partial\Omega \rightarrow \mathbb{R}$ which depends on $F(\Omega)$.

The surface expression (2.10) is often deemed more appealing: it indeed reflects the natural fact that the shape derivative of a “regular enough” objective function depends only on the normal component of the deformation θ on $\partial\Omega$, see for instance [49, 69] about the so-called Structure theorem for shape derivatives. Besides, it easily lends itself to the identification of a descent direction for $F(\Omega)$, as letting $\theta = -v_\Omega \mathbf{n}$ on $\partial\Omega$ straightforwardly ensures that $F'(\Omega)(\theta) < 0$. However, the volume form (2.9) is usually more suitable for mathematical analysis, see e.g. [71, 64].

Remark 2.4. *The deformations θ considered in the practice of the method of Hadamard are often restricted to a subset of $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$, which is for instance made of vector fields with higher regularity, or vector fields vanishing on a fixed, non optimizable region of space.*

Remark 2.5. *The notion of shape derivative is not the only means to understand differentiation with respect to the domain. The alternative concept of topological derivative appraises the sensitivity of a function $F(\Omega)$ with respect to the nucleation of a “small” hole inside Ω . It relies on variations of Ω of the form:*

$$\Omega_{\mathbf{x},r} := \Omega \setminus \overline{B(\mathbf{x},r)}, \text{ where } \mathbf{x} \in \Omega \text{ and } r \ll 1.$$

The function $F(\Omega)$ is then said to have a topological derivative $d_T F(\Omega)(\mathbf{x})$ at $\mathbf{x} \in \Omega$ if the following expansion holds:

$$F(\Omega_{\mathbf{x},r}) = F(\Omega) + r^d d_T F(\Omega)(\mathbf{x}) + o(r^d), \text{ where } \frac{o(r^d)}{r^d} \xrightarrow{r \rightarrow 0} 0.$$

We refer to [63, 118] for the seminal contributions about topological derivatives and to [10, 99] for recent overviews.

3. DISCRETIZATION AND EVOLUTION OF SHAPES USING MODIFIED LAGUERRE DIAGRAMS

This section details the representation of shapes adopted in our framework. The first Section 3.1 introduces basic notions about Laguerre diagrams, in their original version and in the modified version introduced in [83]. In the next Section 3.2, we explain how such objects can be conveniently parametrized by the seed points and the measures of their cells. Finally, Section 3.3 sketches our optimization strategy, whose most critical operations are more extensively described in the next parts of the article.

3.1. Representation of shapes via Laguerre diagrams

Let $D \subset \mathbb{R}^d$ be a fixed bounded and polygonal domain, with unit normal vector $\mathbf{n} : \partial D \rightarrow \mathbb{R}^d$, containing all the shapes of interest. We start with the classical definitions of Voronoi and Laguerre diagrams, the latter being more general, weighted versions of the former; we refer to [14, 24] about these classical notions from computational geometry.

Definition 3.1.

- Let $\mathbf{s} = \{\mathbf{s}_i\}_{i=1,\dots,N} \in \mathbb{R}^{dN}$ be a collection of seed points. The Voronoi diagram $\mathbf{Vor}(\mathbf{s})$ induced by \mathbf{s} is the following decomposition of \overline{D} :

$$(3.1) \quad \overline{D} = \bigcup_{i=1}^N \text{Vor}_i(\mathbf{s}),$$

where for $i = 1, \dots, N$, the Voronoi cell $\text{Vor}_i(\mathbf{s})$ is defined by:

$$(3.2) \quad \text{Vor}_i(\mathbf{s}) := \{\mathbf{x} \in \overline{D}, |\mathbf{x} - \mathbf{s}_i|^2 \leq |\mathbf{x} - \mathbf{s}_j|^2, \forall j \neq i\}.$$

- Let $\mathbf{s} = \{\mathbf{s}_i\}_{i=1,\dots,N} \in \mathbb{R}^{dN}$ be a collection of seed points and $\boldsymbol{\psi} = \{\psi_i\}_{i=1,\dots,N} \in \mathbb{R}^N$ be a set of associated weights. The (classical) Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ attached to \mathbf{s} and $\boldsymbol{\psi}$ is the following decomposition of \overline{D} :

$$(3.3) \quad \overline{D} = \bigcup_{i=1}^N \text{Lag}_i(\mathbf{s}, \boldsymbol{\psi}),$$

where for $i = 1, \dots, N$, the Laguerre cell $\text{Lag}_i(\mathbf{s}, \boldsymbol{\psi})$ reads:

$$(3.4) \quad \text{Lag}_i(\mathbf{s}, \boldsymbol{\psi}) := \{\mathbf{x} \in \overline{D}, |\mathbf{x} - \mathbf{s}_i|^2 - \psi_i \leq |\mathbf{x} - \mathbf{s}_j|^2 - \psi_j, \forall j \neq i\}.$$

The concept of Laguerre diagram paves the way to a way of representing a shape $\Omega \subset D$, as a subset of the cells of a diagram of the form (3.3) and (3.4), associated to suitably chosen seed points \mathbf{s} and weights $\boldsymbol{\psi}$:

$$(3.5) \quad \overline{\Omega} = \bigcup_{i=1}^K \text{Lag}_i(\mathbf{s}, \boldsymbol{\psi}), \text{ where } K < N,$$

and the cells $\text{Lag}_i(\mathbf{s}, \boldsymbol{\psi})$, $i = K + 1, \dots, N$ form a decomposition of the void phase $D \setminus \overline{\Omega}$. In addition to this option, we shall also rely on another point of view, introduced in [83], which leverages a slightly modified version of the notion of Laguerre diagram.

Definition 3.2. Let $\Omega \subset D$ be a shape; two sets $\mathbf{s} = \{\mathbf{s}_i\}_{i=1,\dots,N} \in \mathbb{R}^{dN}$ and $\boldsymbol{\psi} = \{\psi_i\}_{i=1,\dots,N} \in \mathbb{R}^N$ of seed points and weights generate a (modified) Laguerre diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ for Ω if the following decomposition holds:

$$(3.6) \quad \overline{\Omega} = \bigcup_{i=1}^N V_i(\mathbf{s}, \boldsymbol{\psi}),$$

where each closed cell $V_i(\mathbf{s}, \boldsymbol{\psi})$ is obtained by intersection of the corresponding Laguerre cell $\text{Lag}_i(\mathbf{s}, \boldsymbol{\psi})$ in (3.4) with the ball centered at \mathbf{s}_i with radius $\psi_i^{1/2}$:

$$(3.7) \quad V_i(\mathbf{s}, \boldsymbol{\psi}) = \text{Lag}_i(\mathbf{s}, \boldsymbol{\psi}) \cap \overline{B(\mathbf{s}_i, \psi_i^{1/2})}.$$

Both types of decomposition (3.5) and (3.6) will be used in the present work to represent a shape $\Omega \subset D$. The analyses of their mathematical properties are very similar, and since “classical” diagrams of the form (3.3) and (3.4) are more familiar in the literature (although not in the present shape and topology optimization context), we essentially focus our presentation on the modified diagrams of Definition 3.2, pointing out the differences with their classical counterparts when relevant.

The next statement draws the main geometric features of the representation (3.6) and sets related notations used throughout the sequel, see Fig. 2. For simplicity, it is only provided in the case of two space dimensions. Its elementary proof is completely similar to that of the corresponding statement for “classical” Laguerre diagrams (3.3), about which we refer to e.g. [14], and it is omitted for brevity.

Definition 3.3. Let $d = 2$, and let $\Omega \subset D$ be a shape defined via the diagram (3.6) associated to seed points $\mathbf{s} = \{\mathbf{s}_i\}_{i=1,\dots,N}$ and weights $\boldsymbol{\psi} = \{\psi_i\}_{i=1,\dots,N}$. The following facts hold true:

- (i) The cells $V_i(\mathbf{s}, \boldsymbol{\psi})$ in (3.7) are closed, bounded and convex subsets of \mathbb{R}^2 .
- (ii) The intersection $V_i(\mathbf{s}, \boldsymbol{\psi}) \cap V_j(\mathbf{s}, \boldsymbol{\psi})$ between two different cells $i \neq j$ is either empty, or a point or a (closed) straight segment \mathbf{e}_{ij} orthogonal to $\mathbf{s}_i \mathbf{s}_j$.
- (iii) When the intersection $V_i(\mathbf{s}, \boldsymbol{\psi}) \cap V_j(\mathbf{s}, \boldsymbol{\psi})$ is a segment with positive length, $V_j(\mathbf{s}, \boldsymbol{\psi})$ is called a neighbor of $V_i(\mathbf{s}, \boldsymbol{\psi})$. The index set of the neighbors of the i^{th} cell is denoted by $\mathcal{N}_i \subset \{1, \dots, i-1, i+1, \dots, N\}$.
- (iv) We denote by $\mathcal{E} \subset \{1, \dots, N\}^2$ the set of neighboring pairs of cells in the diagram (3.6), that is, $(i, j) \in \mathcal{E}$ if $j \in \mathcal{N}_i$ (and so $i \in \mathcal{N}_j$).
- (v) For each $i = 1, \dots, N$, the boundary $\partial V_i(\mathbf{s}, \boldsymbol{\psi})$ can be decomposed as:

$$\partial V_i(\mathbf{s}, \boldsymbol{\psi}) = \left(\bigcup_{j \in \mathcal{N}_i} \mathbf{e}_{ij} \right) \cup \mathcal{C}_i \cup \mathcal{L}_i,$$

where

- For $j \in \mathcal{N}_i$, $\mathbf{e}_{ij} := \partial V_i(\mathbf{s}, \boldsymbol{\psi}) \cap \partial V_j(\mathbf{s}, \boldsymbol{\psi})$ is a line segment;
 - $\mathcal{C}_i = \partial \Omega \cap \partial V_i(\mathbf{s}, \boldsymbol{\psi})$ is a (possibly empty) collection of circular arcs;
 - $\mathcal{L}_i = \partial D \cap \partial V_i(\mathbf{s}, \boldsymbol{\psi})$ is a (possibly empty) collection of line segments.
- (vi) The endpoints of the (straight or curvilinear) pieces of this representation are called the vertices of the diagram (3.6). We denote by $\mathbf{q} := \{\mathbf{q}_j\}_{j=1,\dots,M} \in \mathbb{R}^{2M}$ the collection of these M vertices.

Remark 3.1. Although the seed points of a Voronoi diagram obviously belong to the interior of their respective cells, the defining seed points of a Laguerre diagram may lie far outside; actually, they may even lie outside the computational domain D although, for implementation stability purposes, we shall impose that they remain inside the latter, see Section 6.3.

To illustrate this point, let us consider a collection of seed points $\mathbf{s} = \{\mathbf{s}_i\}_{i=1,\dots,N} \in \mathbb{R}^{dN}$. Let \mathbf{h} be an arbitrary vector in \mathbb{R}^d ; we introduce the collection of seed points $\tilde{\mathbf{s}} = \{\mathbf{s}_i + \mathbf{h}\}_{i=1,\dots,N}$ resulting from a common translation of $\mathbf{s}_1, \dots, \mathbf{s}_N$ by \mathbf{h} and the weight vector $\tilde{\boldsymbol{\psi}}$ given by $\tilde{\psi}_i = 2\mathbf{s}_i \cdot \mathbf{h}$, $i = 1, \dots, N$. Then,

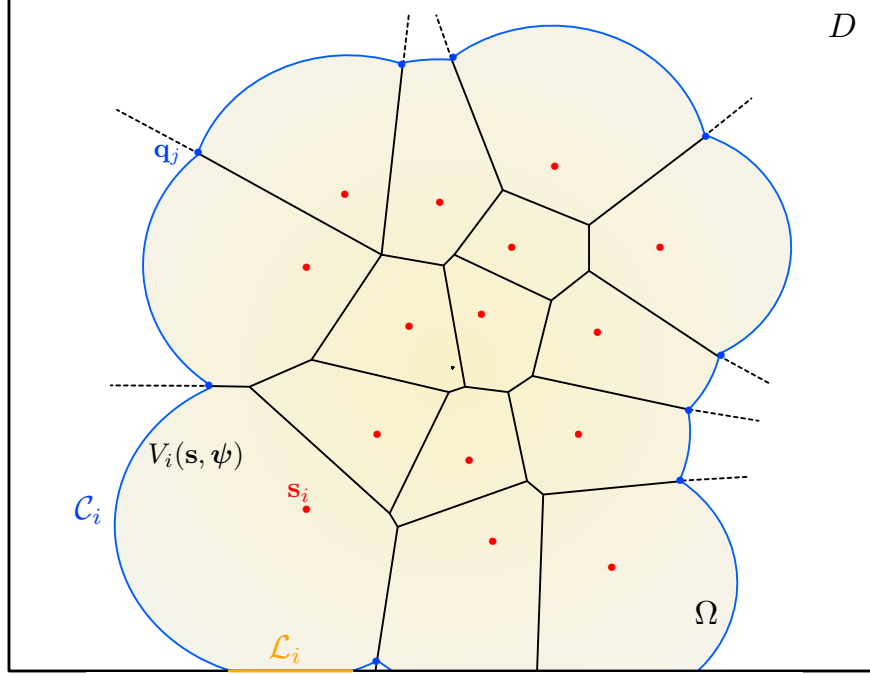


FIGURE 2. Representation (3.6) of the shape $\Omega \subset D$ as the modified Laguerre diagram (3.6) associated to the seed points $\mathbf{s} = \{\mathbf{s}_1, \dots, \mathbf{s}_N\}$ and weights $\boldsymbol{\psi} = \{\psi_1, \dots, \psi_N\}$.

for any index $i = 1, \dots, N$ and any point $\mathbf{x} \in D$, a simple calculation reveals that:

$$\begin{aligned}
 \mathbf{x} \in \text{Lag}_i(\tilde{\mathbf{s}}, \tilde{\boldsymbol{\psi}}) &\Leftrightarrow |\mathbf{x} - \mathbf{h} - \mathbf{s}_i|^2 - 2\mathbf{s}_i \cdot \mathbf{h} \leq |\mathbf{x} - \mathbf{h} - \mathbf{s}_j|^2 - 2\mathbf{s}_j \cdot \mathbf{h} \quad \forall j \neq i \\
 &\Leftrightarrow |\mathbf{x} - \mathbf{s}_i|^2 - 2\mathbf{h} \cdot (\mathbf{x} - \mathbf{s}_i) + |\mathbf{h}|^2 - 2\mathbf{s}_i \cdot \mathbf{h} \leq |\mathbf{x} - \mathbf{s}_j|^2 - 2\mathbf{h} \cdot (\mathbf{x} - \mathbf{s}_j) + |\mathbf{h}|^2 - 2\mathbf{s}_j \cdot \mathbf{h} \quad \forall j \neq i \\
 &\Leftrightarrow |\mathbf{x} - \mathbf{s}_i|^2 \leq |\mathbf{x} - \mathbf{s}_j|^2 \quad \forall j \neq i \\
 &\Leftrightarrow \mathbf{x} \in \text{Vor}_i(\mathbf{s}).
 \end{aligned}$$

In other terms, the Voronoi tessellation $\text{Vor}(\mathbf{s})$ of D , whose seed points $\mathbf{s}_1, \dots, \mathbf{s}_N$ belong to the interior of their respective cells, has the same cells as the Laguerre diagram $\text{Lag}(\tilde{\mathbf{s}}, \tilde{\boldsymbol{\psi}})$ whose seed points lie arbitrarily far away.

Remark 3.2. Although, strictly speaking, a decomposition of the form (3.6) is not a classical Laguerre diagram, it can be understood as the diagram associated to the collection of “seed objects” $\mathbf{s} \cup \{\mathbf{s}_0\}$ and weights $\boldsymbol{\psi} \cup \{\psi_0\}$, where the additional object \mathbf{s}_0 is the whole domain D and $\psi_0 = 0$, see [83]. A straightforward adaptation of the Definition 3.1 of a Laguerre diagram to allow seed objects (and not just seed points) shows indeed that the cells of this new diagram are exactly those $V_i(\mathbf{s}, \boldsymbol{\psi})$ in (3.7) for $i = 1, \dots, N$, with the following additional “void cell”, associated to \mathbf{s}_0 :

$$(3.8) \quad V_0(\mathbf{s}, \boldsymbol{\psi}) := \{\mathbf{x} \in \overline{D}, \forall i = 1, \dots, N, \quad 0 \leq |\mathbf{x} - \mathbf{s}_i|^2 - \psi_i\}.$$

Hence, up to a small abuse of terminology, we shall refer to (3.6) as a (modified) Laguerre diagram for Ω .

An insightful geometric viewpoint about classical and modified Laguerre diagrams (3.3) and (3.4) and (3.6) and (3.7) is to consider them as minimization diagrams of a family of functions, as we now briefly explain in the case $d = 2$ for simplicity, see Fig. 3 for an illustration. The Voronoi diagram $\text{Vor}(\mathbf{s})$ of a set of seed points $\mathbf{s} \in \mathbb{R}^{2N}$ is generated by the 3d paraboloids given by the graphs of the functions $\mathbf{x} \mapsto p_i(\mathbf{x}) = |\mathbf{x} - \mathbf{s}_i|^2$, in the sense that:

$$\text{For each } i = 1, \dots, N, \quad \text{Vor}_i(\mathbf{s}) = \left\{ \mathbf{x} \in \overline{D} \text{ s.t. } i \in \arg \min_{j=1, \dots, N} p_j(\mathbf{x}) \right\},$$

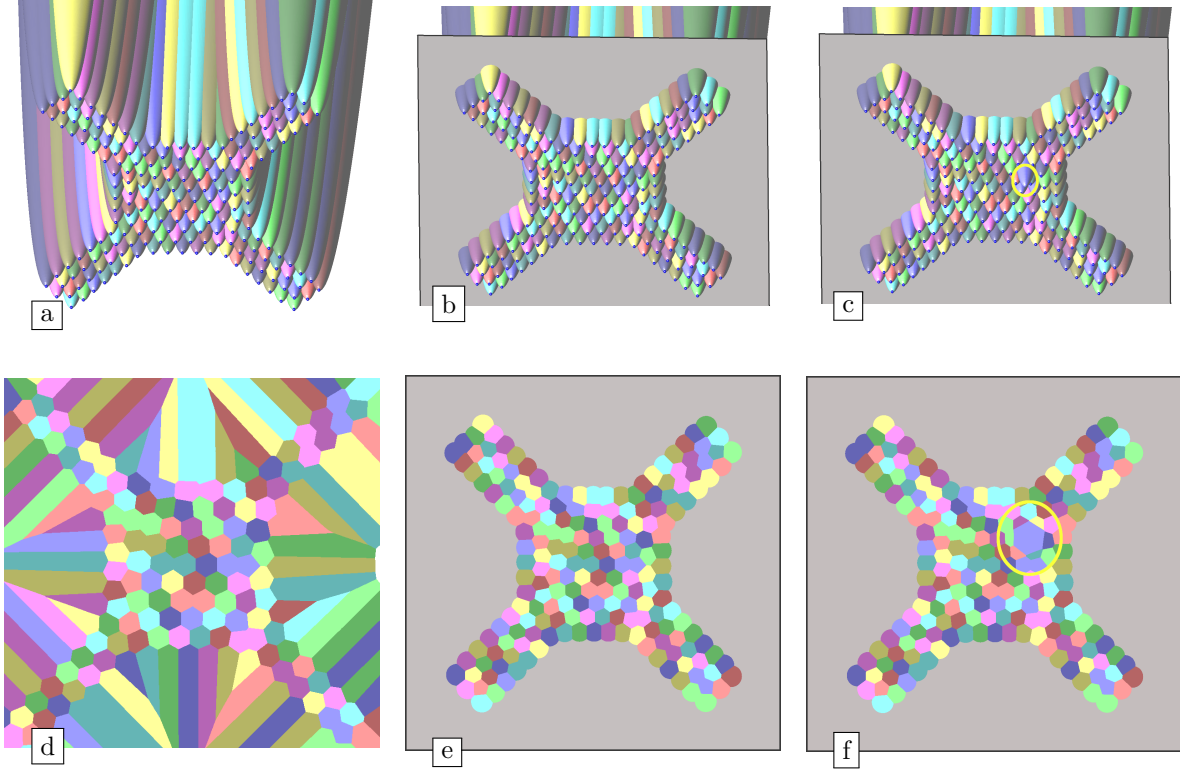


FIGURE 3. (a) Two-dimensional Voronoi diagrams (and “classical” Laguerre diagrams) can be described by collections of three-dimensional paraboloids with axes aligned with the third coordinate vector; (d) The 2d diagram is obtained by “seeing this collection from below”; (b) Modified Laguerre diagrams are obtained by inserting the 3d lower half-space in the diagram, and (d) The intersections between the paraboloids and this half-space create circular arcs at the boundaries of the modified cells; (c) The weights ψ_i of a (classical or modified) Laguerre diagram correspond to shifts of the parabolas along the vertical axis; (f) This results in a change of the area of the cells.

i.e. $\text{Vor}_i(\mathbf{s})$ is the set of points $\mathbf{x} \in \overline{D}$ where $p_i(\mathbf{x})$ is minimal among $p_1(\mathbf{x}), \dots, p_N(\mathbf{x})$. The “classical” Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ involving weights $\boldsymbol{\psi} \in \mathbb{R}^N$ is obtained by “shifting” each of the aforementioned 3d paraboloids by the corresponding weight ψ_i in the negative \mathbf{e}_3 direction. Eventually, a corresponding representation of the modified diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ is obtained by adding the function $p_0(\mathbf{x}) = 0$ to the minimization diagram, i.e. by intersecting each of the 3d paraboloids with the lower half-space $\{\mathbf{x} = (x_1, x_2, x_3) \in \mathbb{R}^3 \text{ s.t. } x_3 \leq 0\}$.

3.2. Parametrization of the diagram representation via the seed points and cell measures

The effect of the weights $\boldsymbol{\psi}$ on the shapes of the cells $V_i(\mathbf{s}, \boldsymbol{\psi})$ as encoded in (3.7) is implicit and difficult to handle. It turns out that the decomposition (3.6) of Ω can be equivalently characterized by the seed points $\mathbf{s} \in \mathbb{R}^{dN}$ and the vector

$$\boldsymbol{\nu} := \{\nu_i\}_{i=1, \dots, N} \in \mathbb{R}^N, \quad \nu_i = |V_i(\mathbf{s}, \boldsymbol{\psi})|$$

gathering the measures of the cells. This is handful for a variety of purposes:

- In structure mechanics, shape and topology optimization problems often feature a constraint on the volume $\text{Vol}(\Omega)$, which is particularly easy to impose when a clear control of the measures of the individual cells of $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ is available, see Remark 6.5.
- A parametrization of $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ in terms of the measures of its cells paves the way to a simple adaptation strategy of the local size of the diagram, see e.g. [135] about this topic.

- As we shall see in [Section 4](#), our construction algorithm for $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ may fail when it contains empty cells. The identification and prevention of this feature are delicate from the datum of the weight vector $\boldsymbol{\psi}$, but natural when $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ is parametrized in terms of the cell measures ν . In this spirit, this control also allows to prevent low volume, low quality elements.
- Such a control is handful in applications beyond optimal design; for instance, in fluid mechanics, retaining cells with constant measures is a natural means to account for the incompressibility of the fluid, see [\[83\]](#).

This section discusses this convenient alternative viewpoint which will be used throughout the following. It relies on optimal transport; for the convenience of the reader, we recall a few useful facts about this theory, referring to [\[75, 79, 104, 112, 129\]](#) for more exhaustive treatments.

Let X stand for the compact set \bar{D} . For $\mathbf{x}, \mathbf{y} \in X$, we denote by $c(\mathbf{x}, \mathbf{y}) = |\mathbf{x} - \mathbf{y}|^2$ the quadratic ground cost on X , which intuitively represents the cost of moving one unit of mass from the position \mathbf{x} to \mathbf{y} . When μ and ν are positive measures on X with equal mass $\mu(X) = \nu(X)$, the Monge formulation of the optimal transport problem of μ onto ν consists in the search for a mapping $T : X \rightarrow X$ realizing the transfer of the mass of μ onto ν at minimal cost, that is:

$$(OT-M) \quad \min \left\{ \int_X c(\mathbf{x}, T(\mathbf{x})) d\mu(\mathbf{x}), T : X \rightarrow X \text{ is } (\mu, \nu)\text{-measurable with } T_{\#}\mu = \nu \right\},$$

where $T_{\#}\mu$ is the push-forward of the measure μ by T , i.e.

$$\text{For all } \nu\text{-measurable subset } B \subset X, \quad T_{\#}\mu(B) = \mu(T^{-1}(B)).$$

Unfortunately, the problem (OT-M) is often ill-posed; for instance, an admissible mapping T in (OT-M) may easily fail to exist. One therefore usually considers instead the so-called [Kantorovich](#) relaxation of (OT-M):

$$(OT-K) \quad \min \left\{ \int_{X \times X} c(\mathbf{x}, \mathbf{y}) d\pi(\mathbf{x}, \mathbf{y}), \pi \text{ is a positive measure on } X \times X \text{ s.t. } \pi_1 = \mu, \pi_2 = \nu \right\}.$$

Here, π_1 and π_2 denote the first and second marginals of the measure π on $X \times X$:

$$\text{For all } \pi_1, \pi_2\text{-measurable subsets } A \subset X, B \subset X, \quad \pi_1(A) = \pi(A \times X) \text{ and } \pi_2(B) = \pi(X \times B).$$

In (OT-K), the minimization is realized over positive measures π on $X \times X$ (also called transport plans or couplings) with marginals μ and ν . Intuitively, when A and B are μ - and ν -measurable subsets of X , $\pi(A \times B)$ is the ‘‘quantity of mass’’ transferred from A to B . The problem (OT-K) is often better behaved than (OT-M) from the mathematical viewpoint; for instance, it is immediate to see that it is convex.

Let us specialize this abstract setting to our purpose. Let $\mathbf{s} = \{\mathbf{s}_i\}_{i=1, \dots, N} \in \mathbb{R}^{dN}$ be a collection of seed points, and let $\boldsymbol{\psi} = \{\psi_i\}_{i=1, \dots, N}$ be a weight vector. According to [Remark 3.2](#) we let $\psi_0 := 0$ be the weight for the seed object $\mathbf{s}_0 = D$. Likewise, when $\boldsymbol{\nu} = \{\nu_i\}_{i=1, \dots, N}$ is a collection of volume targets such that

$$\forall i = 1, \dots, N, \quad \nu_i \geq 0, \quad \text{and} \quad \sum_{i=0}^N \nu_i \leq |D|,$$

we denote by $\nu_0 := |D| - \sum_{i=1}^N \nu_i$ the volume of the corresponding ‘‘void phase’’ $V_0(\mathbf{s}, \boldsymbol{\psi})$ in [\(3.8\)](#). We then aim to apply the aforementioned optimal transport theory to the situation where $\mu := \mathbb{1}_D dx$ is the restriction to X of the d -dimensional Lebesgue measure and

$$(3.9) \quad \nu := \sum_{i=1}^N \nu_i \delta_{\mathbf{s}_i} + \nu_0 \frac{1}{|D|} \mathbb{1}_D dx$$

is the discrete measure induced by the objects $\{\mathbf{s}_i\}_{i=1, \dots, N} \cup \{\mathbf{s}_0\}$. In this perspective, we shall often invoke the following assumptions about \mathbf{s} and $\boldsymbol{\psi}$ to rule out very degenerate situations as far as the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ is concerned:

(G1)

All the cells $V_i(\mathbf{s}, \boldsymbol{\psi})$ have positive Lebesgue measure, $i = 0, 1, \dots, N$ (in particular, they are non empty).

(G2)

No three distinct seed points $\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k$ are aligned.

(G3) For each vertex \mathbf{q} at the intersection between 2 neighboring cells $V_i(\mathbf{s}, \boldsymbol{\psi}), V_j(\mathbf{s}, \boldsymbol{\psi}), i, j = 1, \dots, N$, and the void phase $V_0(\mathbf{s}, \boldsymbol{\psi}), \mathbf{q}, \mathbf{s}_i$ and \mathbf{s}_j are not aligned.

(G4) Each vertex \mathbf{q} is at the intersection of at most 3 distinct cells

$$V_i(\mathbf{s}, \boldsymbol{\psi}), V_j(\mathbf{s}, \boldsymbol{\psi}), V_k(\mathbf{s}, \boldsymbol{\psi}), i, j, k = 0, 1, \dots, N.$$

(G5) The boundary ∂D of the computational domain is smooth at each vertex \mathbf{q} lying on ∂D .

Such a point is moreover at the intersection of at most 2 distinct cells $V_i(\mathbf{s}, \boldsymbol{\psi}), V_j(\mathbf{s}, \boldsymbol{\psi}), i, j = 0, \dots, N$.

(G6) For each vertex \mathbf{q} at the intersection of ∂D and 2 distinct cells $V_i(\mathbf{s}, \boldsymbol{\psi}), V_j(\mathbf{s}, \boldsymbol{\psi}), i, j = 1, \dots, N$, the line $\partial V_i(\mathbf{s}, \boldsymbol{\psi}) \cap \partial V_j(\mathbf{s}, \boldsymbol{\psi})$ is not orthogonal to the unit normal vector $\mathbf{n}(\mathbf{q})$ to ∂D .

(G7) For each vertex \mathbf{q} at the intersection between ∂D , one cell $V_i(\mathbf{s}, \boldsymbol{\psi})$ and the void $V_0(\mathbf{s}, \boldsymbol{\psi})$, the line $\mathbf{q}\mathbf{s}_i$ is not orthogonal to $\mathbf{n}(\mathbf{q})$.

The desired description of shapes Ω in terms of the seed points \mathbf{s} and cell measures $\boldsymbol{\nu}$ of an associated diagram (3.6) is enabled by the following result. Its proof, which is an adaptation of those of Theorems 37 and 40 in [88], is postponed to Appendix C.

Theorem 3.1. *Let $D \subset \mathbb{R}^d$ be a bounded, Lipschitz domain and $X := \overline{D}$. Let $\mathbf{s} = \{\mathbf{s}_i\}_{i=1, \dots, N} \in \mathbb{R}^{dN}$ be a set of seed points in D , and let $\boldsymbol{\nu} = \{\nu_i\}_{i=1, \dots, N} \in \mathbb{R}^N$ be a vector of associated cell measures, satisfying:*

$$(3.10) \quad \forall i = 1, \dots, N, \quad \nu_i > 0, \quad \text{and} \quad \sum_{i=1}^N \nu_i < |D|.$$

The following statements hold true:

(i) There exists a unique weight vector $\boldsymbol{\psi}^* \equiv \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}) \in \mathbb{R}^N$ such that:

$$(3.11) \quad \forall i = 1, \dots, N, \quad |V_i(\mathbf{s}, \boldsymbol{\psi}^*)| = \nu_i, \quad \text{and so} \quad |V_0(\mathbf{s}, \boldsymbol{\psi}^*)| = \nu_0.$$

(ii) The vector $\boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})$ is one solution to the maximization problem

$$(3.12) \quad \max_{\boldsymbol{\psi} \in \mathbb{R}^N} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}),$$

where the so-called Kantorovich functional $K : \mathbb{R}_s^{dN} \times \mathbb{R}_\nu^N \times \mathbb{R}_\psi^N \rightarrow \mathbb{R}$ is defined by:

$$(3.13) \quad K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) := \sum_{i=1}^N \int_{V_i(\mathbf{s}, \boldsymbol{\psi})} (|\mathbf{x} - \mathbf{s}_i|^2 - \psi_i) \, d\mathbf{x} + \sum_{i=1}^N \nu_i \psi_i.$$

(iii) Assuming that the genericity conditions (G1) to (G7) hold true, this solution is unique. Moreover, K is smooth on an open neighborhood of $(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))$ in $\mathbb{R}_s^{dN} \times \mathbb{R}_\nu^N \times \mathbb{R}_\psi^N$, and $\boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})$ is the unique solution to the equation:

$$(3.14) \quad \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) = \mathbf{0}, \quad \text{where} \quad \mathbf{F} : \mathbb{R}_s^{dN} \times \mathbb{R}_\nu^N \times \mathbb{R}_\psi^N \rightarrow \mathbb{R}^N \quad \text{is defined by} \quad \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) := \nabla_\psi K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}).$$

(iv) There exists a unique optimal transport mapping T_ψ between the measures μ and ν in the Monge formulation (OT-M), which is given by:

$$(3.15) \quad \text{For a.e. } \mathbf{x} \in X, \quad T_\psi(\mathbf{x}) = \begin{cases} \mathbf{s}_i & \text{if } \mathbf{x} \in V_i(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})), \quad i = 1, \dots, N, \\ \mathbf{x} & \text{otherwise.} \end{cases}$$

Remark 3.3. *The uniqueness statement (iii) in the above theorem actually holds true under weaker assumptions than the collection (G1) to (G7). It only requires that the mapping $\boldsymbol{\psi} \mapsto K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$ be differentiable, i.e. loosely speaking that the area $|V_i(\mathbf{s}, \boldsymbol{\psi})|$ of each cell be differentiable with respect to $\boldsymbol{\psi}$, while (G1) to (G7) imply that all the individual vertices of the cells of $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ are differentiable with respect to $\boldsymbol{\psi}$, see Remark B.1 below about this point.*

Remark 3.4. The situation where $\sum_{i=1}^N \nu_i = |D|$ corresponds to the “classical” setting where the cell $V_0(\mathbf{s}, \boldsymbol{\psi})$ is empty and the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ coincides with the Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$. Then, there still exists a collection of weights $\boldsymbol{\psi}$ fulfilling the conclusions of [Theorem 3.1](#), but the latter is only unique up to the addition of a common factor to all the weights ψ_i , $i = 1, \dots, N$ [[88](#)].

Remark 3.5. Our representation [\(3.6\)](#) of shapes and their deformations echoes to the framework of “linearized optimal transport”, recently proposed in [[48](#), [87](#), [131](#)]. Briefly, the latter advocates to parametrize measures ν on X by the unique optimal transport mapping between a fixed reference measure μ on X and ν , i.e. the solution to (OT-M). By comparison, our parametrization [\(3.6\)](#) of shapes $\Omega \subset D$ boils down to considering the particular class [\(3.15\)](#) of optimal transport mapping between the uniform measure $\mu = \frac{1}{|D|} \mathbb{1}_D dx$ on X and measures ν of the form [\(3.9\)](#).

3.3. General sketch of the Laguerre diagram-based shape and topology optimization method

This section outlines our algorithmic strategy for the resolution of a generic shape and topology optimization problem of the form [\(P\)](#), in which the objective and constraint functionals $J(\Omega)$ and $\mathbf{G}(\Omega)$ depend on the optimized shape Ω via the solution u_Ω to a boundary value problem posed on Ω , as in the examples of [Section 2.2](#). The method is summarized in [Algorithm 1](#); its main steps are illustrated on [Fig. 4](#) and they are more thoroughly described in the next sections of the article.

Algorithm 1 Laguerre diagram based shape and topology optimization algorithm for [\(P\)](#).

Input: Initial shape Ω^0 characterized via [\(3.6\)](#) and [\(3.7\)](#) by the datum of:

- The collection of seed points $\mathbf{s}^0 = \{\mathbf{s}_1^0, \dots, \mathbf{s}_N^0\}$;
- The vector $\boldsymbol{\nu}^0 := \{\nu_1^0, \dots, \nu_N^0\}$ gathering the measures of the cells.

for $n = 0, \dots$, until convergence **do**

- (1) Calculate the unique weight vector $\boldsymbol{\psi}^n \in \mathbb{R}^N$ such that:

$$\forall i = 1, \dots, N, \quad |V_i(\mathbf{s}^n, \boldsymbol{\psi}^n)| = \nu_i^n.$$

- (2) (Optionally) Modify this diagram by resampling and Lloyd regularization.
(3) Construct a polygonal mesh \mathcal{T}^n for Ω^n from the diagram $\mathbf{V}(\mathbf{s}^n, \boldsymbol{\psi}^n)$.
(4) Calculate the solutions u_{Ω^n} , p_{Ω^n} to the state and adjoint equations on the mesh \mathcal{T}^n .
(5) Calculate a descent direction $(\mathbf{h}^n, \hat{\boldsymbol{\nu}}^n)$ for the problem [\(P\)](#), that is:
- A collection $\mathbf{h}^n = \{\mathbf{h}_i^n\}_{i=1, \dots, N} \in \mathbb{R}^{dN}$ of vectors attached to the seed points \mathbf{s}^n ;
 - An update vector $\hat{\boldsymbol{\nu}}^n \in \mathbb{R}^N$ for the measures.
- (6) Update the seed points and measure vectors \mathbf{s}^n and $\boldsymbol{\nu}^n$ as:

$$\mathbf{s}_i^{n+1} = \mathbf{s}_i^n + \tau^n \mathbf{h}_i^n, \quad \text{and} \quad \boldsymbol{\nu}^{n+1} = \boldsymbol{\nu}^n + \rho^n \hat{\boldsymbol{\nu}}^n,$$

where $\tau^n > 0$ and $\rho^n > 0$ are suitable descent steps.

end for

return Seed points \mathbf{s}^n and measures $\boldsymbol{\nu}^n$ representing the optimized shape Ω^n via [\(3.6\)](#) and [\(3.7\)](#).

At each iteration $n = 0, \dots$, we label with an n superscript the actual instances of the various objects at stake. The shape $\Omega^n \subset D$ is consistently described by the seed points $\mathbf{s}^n \in \mathbb{R}^{dN}$ and the vector $\boldsymbol{\nu}^n \in \mathbb{R}^N$ of the cell measures of a diagram $\mathbf{V}(\mathbf{s}^n, \boldsymbol{\psi}^n)$ of the form [\(3.6\)](#) and [\(3.7\)](#). The iteration starts with the calculation of the unique weight vector $\boldsymbol{\psi}^n \in \mathbb{R}^N$ such that the measure of each cell $V_i(\mathbf{s}^n, \boldsymbol{\psi}^n)$ equals ν_i , see [Section 4.1](#). The diagram associated to these data is computed, as described in [Section 4.1](#). A number of post-processing operations are conducted on the latter, such as regularization, resampling, etc., see [Sections 4.2](#) and [4.3](#). Note that these operations may alter the number N of seed points – a detail which is omitted in the present sketch for simplicity. A polygonal mesh \mathcal{T}^n of Ω^n is then constructed from this diagram, see again [Section 4](#). Then, the solution u_{Ω^n} to the mechanical system of interest (and that p_{Ω^n} to the adjoint system) is calculated on the mesh \mathcal{T}^n ; this task relies on the Virtual Element Method, and its treatment is described in [Section 5](#). The derivatives of the objective and constraint functions $J(\Omega)$ and $\mathbf{G}(\Omega)$ are evaluated, at first with respect to the vertices \mathbf{q}^n of the polygonal mesh \mathcal{T}^n – see [Section 6.1](#) – then with respect to the seed points and

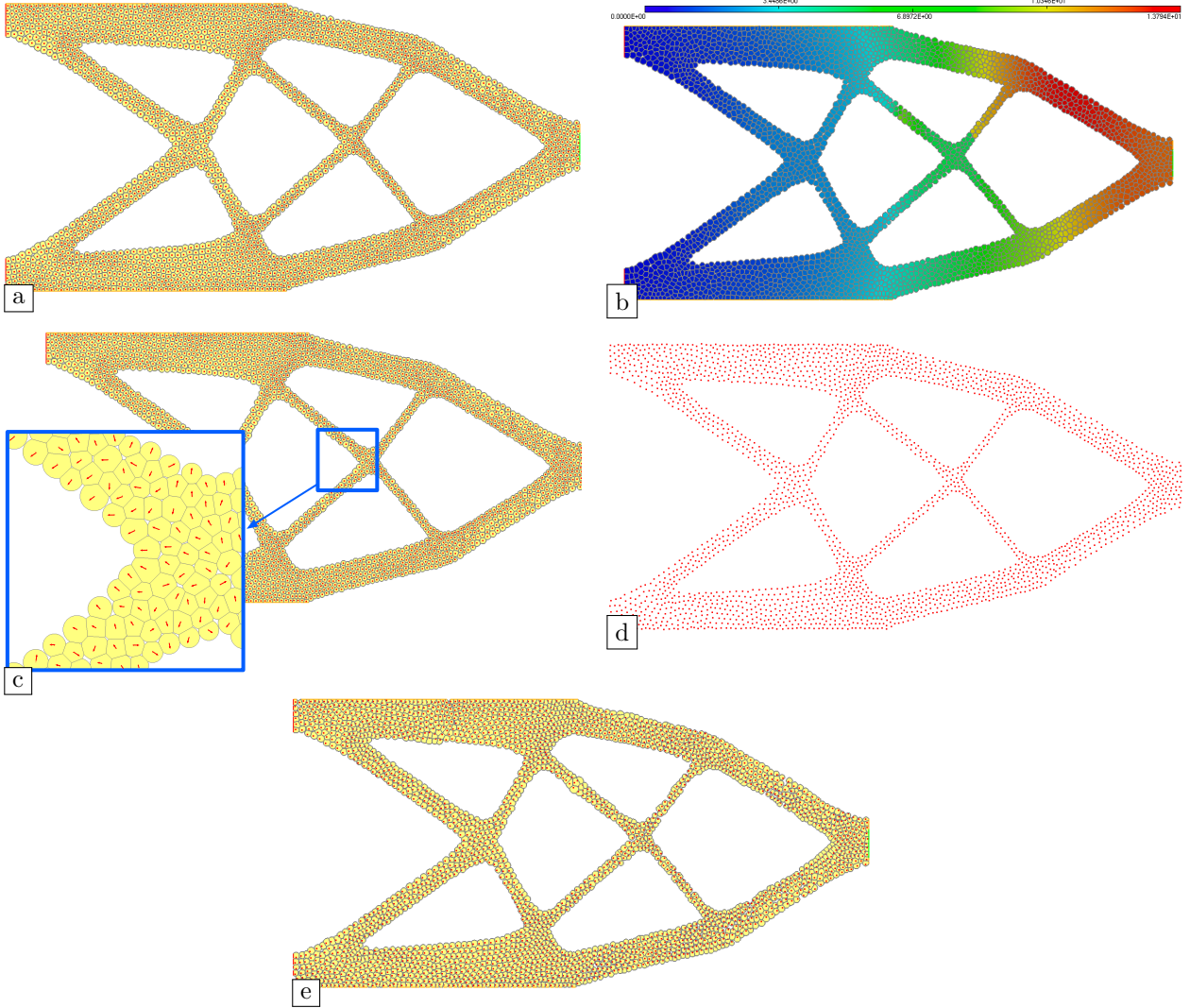


FIGURE 4. Illustration of the main steps of [Algorithm 1](#); (a) At the beginning of each iteration, the weight vector ψ^n is computed, and the diagram $\mathbf{V}(\mathbf{s}^n, \psi^n)$ for Ω^n is constructed (the seed points are depicted in red); (b) The state equation for u_{Ω^n} is solved by the Virtual Element Method on the polygonal mesh \mathcal{T}^n of Ω^n induced by $\mathbf{V}(\mathbf{s}^n, \psi^n)$; (c) A descent direction is calculated as a vector field \mathbf{h}^n for the seed points and an update ν^n for the cell measures; (d) The seed points \mathbf{s}^{n+1} of the new diagram for Ω^{n+1} are obtained by moving those in \mathbf{s}^n along \mathbf{h}^n ; the cell measures ν^{n+1} are obtained likewise; (e) The new weight vector ψ^{n+1} and the diagram $\mathbf{V}(\mathbf{s}^{n+1}, \psi^{n+1})$ for Ω^{n+1} are computed.

cell measures \mathbf{s}^n and ν^n defining the representation (3.6) of Ω^n , see [Section 6.2](#). A descent direction for the considered optimization problem is inferred thanks to a constrained optimization algorithm under the form of update vectors for seeds and cell measures, see [Sections 6.3](#) and [6.4](#). This procedure is iterated until convergence.

4. GEOMETRIC COMPUTATIONS ON Ω

This section provides a few details about the non trivial operations from algorithmic geometry involved in [Algorithm 1](#). In [Section 4.1](#), we discuss the calculation of the unique weight vector $\psi^*(\mathbf{s}, \nu)$ supplied by [Theorem 3.1](#), guaranteeing that the diagram $\mathbf{V}(\mathbf{s}, \psi^*(\mathbf{s}, \nu))$ complies with the measure constraints (3.11);

we notably describe the computation of this diagram. In [Section 4.2](#), we present a variant of the well-known Lloyd’s algorithm aimed at improving the aspect of the cells of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$, in the perspective of realizing accurate mechanical computations. Eventually, in [Section 4.3](#), we sketch simple numerical recipes to adjust the local density of cells and remove “small” components disconnected from the main structure in the course of the iterative process.

4.1. Calculation of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\nu})$ associated to seed points \mathbf{s} and cell measures $\boldsymbol{\nu}$

Throughout this section, $\mathbf{s} \in \mathbb{R}^{dN}$ and $\boldsymbol{\nu} \in \mathbb{R}^N$ are collections of seed points and weights satisfying the assumptions of [Theorem 3.1](#); notably, they satisfy (G1) to (G7). We describe the calculation of the unique weight vector $\boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})$ guaranteeing that each cell $i = 1, \dots, N$ of the modified diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))$ has measure ν_i , together with the practical construction of this diagram. Our strategy is based on [Theorem 3.1](#), whereby $\boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}) \in \mathbb{R}^N$ is characterized as the unique solution to the non linear equation (3.14), that we rewrite below for convenience:

$$(4.1) \quad \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) = 0, \text{ where } \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) := \nabla_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}).$$

This feature paves the way to a Newton-Raphson algorithm [73], based on the differentiability of $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \cdot)$, and thus on the second order differentiability of the Kantorovich functional $K(\mathbf{s}, \boldsymbol{\nu}, \cdot)$ with respect to weights $\boldsymbol{\psi}$, which hold true since \mathbf{s} and $\boldsymbol{\nu}$ are generic in the sense of (G1) to (G7).

This numerical strategy is sketched in [Algorithm 2](#) and it is illustrated in [Fig. 5](#). Briefly, the algorithm starts with a suitable initial guess $\boldsymbol{\psi}^0$. At each iteration $n = 0, \dots$, whose corresponding objects are labelled with an n superscript, we first construct the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi}^n)$. The entries of the gradient $\nabla_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n) \in \mathbb{R}^N$ and Hessian matrix $[\nabla_{\boldsymbol{\psi}}^2 K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)] \in \mathbb{R}^{N \times N}$ of the Kantorovich functional are then calculated from this geometric support. The update step $\mathbf{p}^n \in \mathbb{R}^N$ for the weight vector is then obtained by the solution of the following Newton linear system with size N :

$$(4.2) \quad [\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)] \mathbf{p}^n = -\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n).$$

Eventually, a descent parameter $\alpha^n > 0$ is chosen to update the weight vector $\boldsymbol{\psi}^n$ into the next iterate $\boldsymbol{\psi}^{n+1}$. We provide a little more details about the main steps of this process in the next subsections.

Algorithm 2 Computation of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ associated to given seed points \mathbf{s} and cell measures $\boldsymbol{\nu}$

Inputs:

- Fixed collection $\mathbf{s} = \{\mathbf{s}_1, \dots, \mathbf{s}_N\}$ of seed points.
- Fixed collection $\boldsymbol{\nu} = \{\nu_1, \dots, \nu_N\}$ of cell measures.
- Initial weight vector $\boldsymbol{\psi}^0 = \{\psi_1^0, \dots, \psi_N^0\}$ such that for all $i = 1, \dots, N$, $V_i(\mathbf{s}, \boldsymbol{\psi}^0) \neq \emptyset$.
- Tolerance parameter $\varepsilon_{\text{Newt}}$ about the fulfillment of (4.1).

for $n = 0, \dots$, until convergence **do**

- (1) Compute the Laguerre diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi}^n)$
- (2) Calculate the gradient $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n) = \nabla_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)$ of the functional $K(\mathbf{s}, \boldsymbol{\nu}, \cdot)$ at $\boldsymbol{\psi}^n$.
- (3) **if** $|\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)|_{\infty} < \varepsilon_{\text{Newt}}$ **then exit loop**
- (4) Calculate the $N \times N$ Hessian matrix $[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)] = [\nabla_{\boldsymbol{\psi}}^2 K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)]$ of $K(\mathbf{s}, \boldsymbol{\nu}, \cdot)$ at $\boldsymbol{\psi}^n$.
- (5) Calculate the solution $\mathbf{p}^n \in \mathbb{R}^N$ to the linear system (4.2).
- (6) Select a suitable descent parameter $\alpha^n > 0$.
- (7) Update the weight vector as: $\boldsymbol{\psi}^{n+1} = \boldsymbol{\psi}^n + \alpha^n \mathbf{p}^n$.

end for

return Diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ satisfying $|V_i(\mathbf{s}, \boldsymbol{\psi}^n)| = \nu_i$, for all $i = 1, \dots, N$, up to precision $\varepsilon_{\text{Newt}}$.

4.1.1. Initialization

According to [Theorem 3.1](#), the initial weight vector $\boldsymbol{\psi}^0$ must be such that all the cells $V_i(\mathbf{s}, \boldsymbol{\psi}^0)$, $i = 0, 1, \dots, N$ (i.e. including the “void” phase $V_0(\mathbf{s}, \boldsymbol{\psi}^0)$ given by (3.8)) are non empty. In particular, this property is meant to ensure that the Kantorovich mapping $K(\mathbf{s}, \boldsymbol{\nu}, \cdot)$ is differentiable at $\boldsymbol{\psi}^0$, see [Remark 3.3](#).

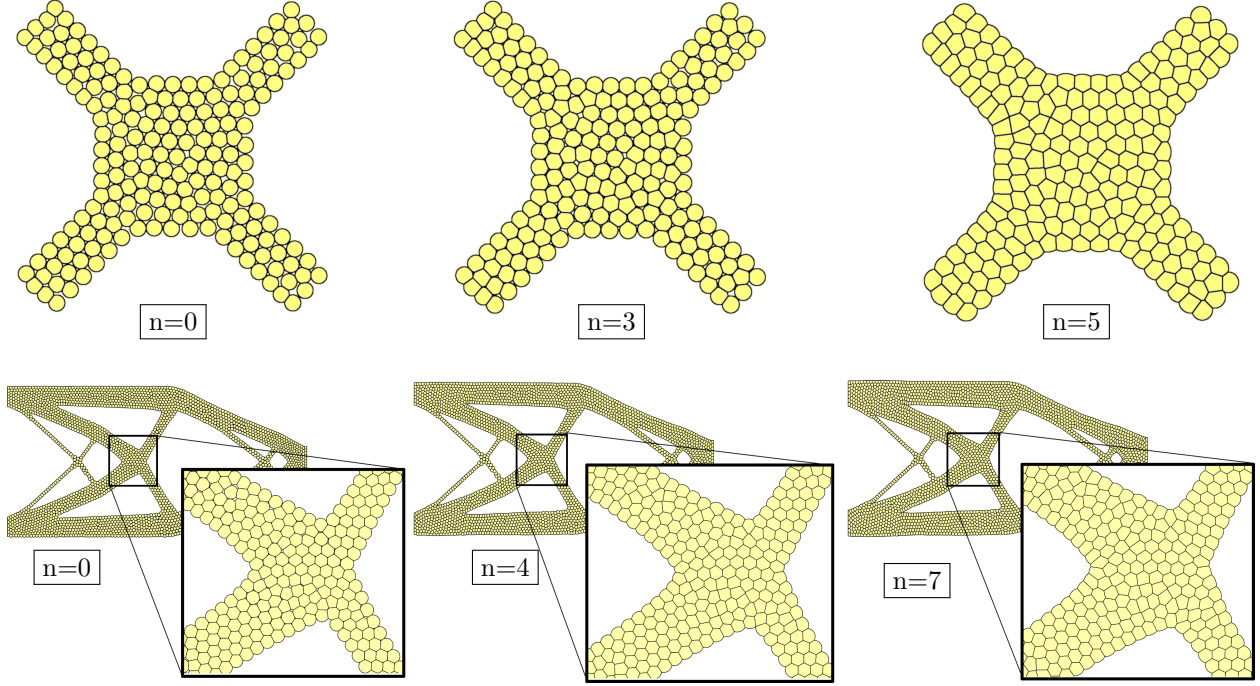


FIGURE 5. Three different Newton iterations in the calculation of the modified diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ in Section 4.1. The weight vector $\boldsymbol{\psi}$ is initialized with a value $\boldsymbol{\psi}^0$ ensuring that none of the cells $V_i(\mathbf{s}, \boldsymbol{\psi}^0)$ is empty, and it is iteratively updated by Newton iterations. This procedure appraises contacts between neighboring cells, and at convergence, $\boldsymbol{\psi}^n$ is such that the measure of each cell $|V_i(\mathbf{s}, \boldsymbol{\psi}^n)|$ corresponds to the prescribed value ν_i , up to the precision parameter $\varepsilon_{\text{Newt}}$.

One sufficient condition for this to hold is to select a common positive value $\bar{\psi} > 0$ for all the ψ_i^0 , $i = 1, \dots, N$. Indeed, the classical Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi}^0)$ then coincides with the Voronoi diagram $\mathbf{Vor}(\mathbf{s})$ associated to the seed points \mathbf{s} , where each cell $\text{Vor}_i(\mathbf{s})$ contains at least its associated seed point \mathbf{s}_i . In turn, each modified cell $V_i(\mathbf{s}, \boldsymbol{\psi}^0) = \text{Lag}_i(\mathbf{s}, \boldsymbol{\psi}^0) \cap B(\mathbf{s}_i, \psi_i^{1/2})$ is non empty, $i = 1, \dots, N$, and since the common value $\bar{\psi}$ is arbitrary, it is easy to guarantee that $V_0(\mathbf{s}, \boldsymbol{\psi}^0)$ is also non empty. In practice, we choose $\boldsymbol{\psi}^0 = \{1, 1, \dots, 1\}$ as initial weight vector.

If all the target measures ν_i are equal to a common value $\bar{\nu}$, a simple heuristic suggests an initialization procedure with “better” practical behavior. Intuitively, if a seed point \mathbf{s}_i lies “sufficiently far away” from all the other seeds \mathbf{s}_j , $j \neq i$, then the classical Laguerre cell $\text{Lag}_i(\mathbf{s}, \boldsymbol{\psi})$ contains the ball $B(\mathbf{s}_i, \psi_i^{1/2})$; thus the modified cell $V_i(\mathbf{s}, \boldsymbol{\psi})$ coincides with this ball. Going further, it is easy to show that if the seed points \mathbf{s}_i are “well separated” from one another in the sense that:

$$(4.3) \quad |\mathbf{s}_j - \mathbf{s}_i| > \sqrt{\frac{\nu_i}{\pi}} + \sqrt{\frac{\nu_j}{\pi}} \quad \text{in 2d, and} \quad |\mathbf{s}_j - \mathbf{s}_i| > \left(\frac{3\nu_i}{4\pi}\right)^{\frac{1}{3}} + \left(\frac{3\nu_j}{4\pi}\right)^{\frac{1}{3}} \quad \text{in 3d,}$$

then the optimal transport problem (3.14) is trivial, its solution being given by:

$$(4.4) \quad \psi_i = \begin{cases} \frac{\bar{\nu}}{\pi} & \text{if } d = 2, \\ \left(\frac{3\bar{\nu}}{4\pi}\right)^{\frac{2}{3}} & \text{if } d = 3, \end{cases} \quad i = 1, \dots, N,$$

and the balls $B(\mathbf{s}_i, \psi_i^{1/2})$ are disjoint from one another.

In general, the data \mathbf{s} and $\boldsymbol{\nu}$ do not satisfy the separation property (4.3), and the cells of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ “interact” through their common boundaries, see Fig. 5. Then, several iterations of Algorithm 2

are needed to achieve convergence (typically 5 to 10), but in practice, the heuristic initialization (4.4) saves a couple of Newton iterations.

Last but not least, when the target measures ν_i are different, the choice (4.4) still proves quite efficient when used with the average value $\bar{\nu} := \frac{1}{N} \sum_{i=1}^N \nu_i$.

4.1.2. Computation of Laguerre diagrams

This section briefly describes the construction of the modified diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ associated to given seed points $\mathbf{s} \in \mathbb{R}^{dN}$ and weights $\boldsymbol{\psi} \in \mathbb{R}^N$. We consider the 2d case $d = 2$ for simplicity, and refer to [83] and the bibliography therein for further details, including the treatment of the 3d case. The task under scrutiny is decomposed into two steps.

Step 1: We compute the classical Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ associated to \mathbf{s} and $\boldsymbol{\psi}$.

Several algorithms are available to achieve this task, whose outcome is illustrated on Fig. 6 (a). In the present article, we rely on the iterative insertion algorithm of Bowyer and Watson proposed in [28, 132], see also [58] – a strategy originally intended to construct Voronoi diagrams (or their dual, Delaunay triangulations), which can handle more general Laguerre diagrams up to simple adaptations, see [14].

The Bowyer-Watson algorithm proceeds by iterative insertion of the seed points $\mathbf{s}_1, \dots, \mathbf{s}_N$. Each stage $n = 1, \dots, N$ of the process starts with the datum of the Laguerre diagram $\mathbf{Lag}(\widehat{\mathbf{s}}^{n-1}, \widehat{\boldsymbol{\psi}}^{n-1})$ associated to the subcollections $\widehat{\mathbf{s}}^{n-1} := \{\mathbf{s}_1, \dots, \mathbf{s}_{n-1}\}$ and $\widehat{\boldsymbol{\psi}}^{n-1} := \{\psi_1, \dots, \psi_{n-1}\}$ of seed points and weights, with an obvious adaptation of this setting in the case $n = 1$. This diagram is stored under the dual form of a so-called regular triangulation \mathcal{T}^{n-1} of D , see Fig. 6 (b): the vertices of \mathcal{T}^{n-1} are exactly $\mathbf{s}_1, \dots, \mathbf{s}_{n-1}$ and its triangles are those connecting the 3-uples of seed points $\{\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k\} \subset \widehat{\mathbf{s}}^{n-1}$ (with disjoint indices i, j, k) whose associated cells share a common vertex \mathbf{q} . Working with this dual structure is convenient since all dual cells (i.e. the triangles of \mathcal{T}^{n-1}) have the same number of vertices, whereas storing directly the Laguerre cells of $\mathbf{Lag}(\widehat{\mathbf{s}}^{n-1}, \widehat{\boldsymbol{\psi}}^{n-1})$ would require a more complicated data structure, since the latter have different numbers of vertices.

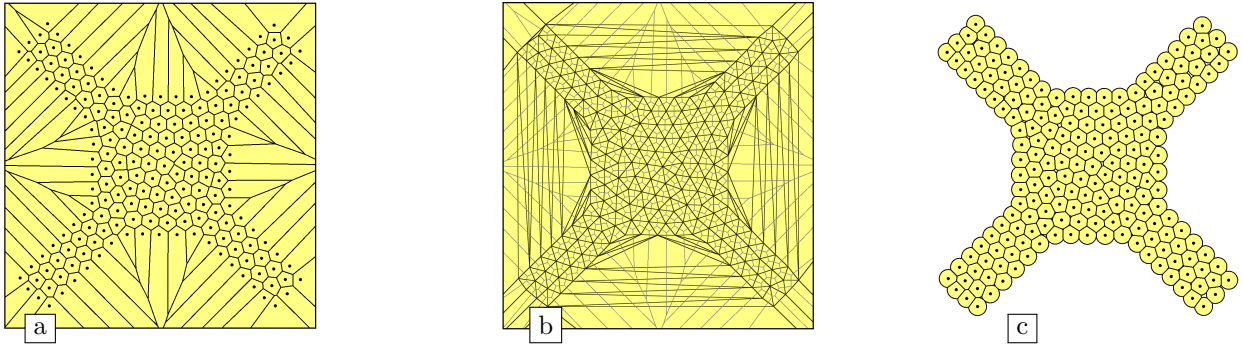


FIGURE 6. *Computation of the modified Laguerre diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ in Section 4.1.2: (a) “Classical” Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$; (b) Associated dual regular triangulation used to store $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ internally; (c) Modified diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ obtained from $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ after clipping each cell $\text{Lag}_i(\mathbf{s}, \boldsymbol{\psi})$ by the ball with center \mathbf{s}_i and radius $\sqrt{\psi_i}$.*

The algorithm then inserts the seed \mathbf{s}_n into $\mathbf{Lag}(\widehat{\mathbf{s}}^{n-1}, \widehat{\boldsymbol{\psi}}^{n-1})$, see Fig. 7. To achieve this, one first identifies the vertices \mathbf{q} of $\mathbf{Lag}(\widehat{\mathbf{s}}^{n-1}, \widehat{\boldsymbol{\psi}}^{n-1})$ that are in conflict with the presence of \mathbf{s}_n , in the sense that they cannot exist in the updated diagram $\mathbf{Lag}(\widehat{\mathbf{s}}^n, \widehat{\boldsymbol{\psi}}^n)$ because they lie inside the Laguerre cell $\text{Lag}_n(\widehat{\mathbf{s}}^n, \widehat{\boldsymbol{\psi}}^n)$ of the new seed point \mathbf{s}_n in the latter. Such a conflicting vertex \mathbf{q} , defined by the intersection of the three Laguerre cells attached to the seeds $\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k$, is characterized by the following relation:

$$(4.5) \quad |\mathbf{q} - \mathbf{s}_n|^2 - \psi_n < R_{ijk}, \text{ where } R_{ijk} := |\mathbf{q} - \mathbf{s}_i|^2 - \psi_i = |\mathbf{q} - \mathbf{s}_j|^2 - \psi_j = |\mathbf{q} - \mathbf{s}_k|^2 - \psi_k.$$

The equalities featured in (4.5) yield a linear system characterizing the coordinates of \mathbf{q} , see also Appendix A. By solving the latter with the classical Cramer’s formulas, substituting the resulting values into the above

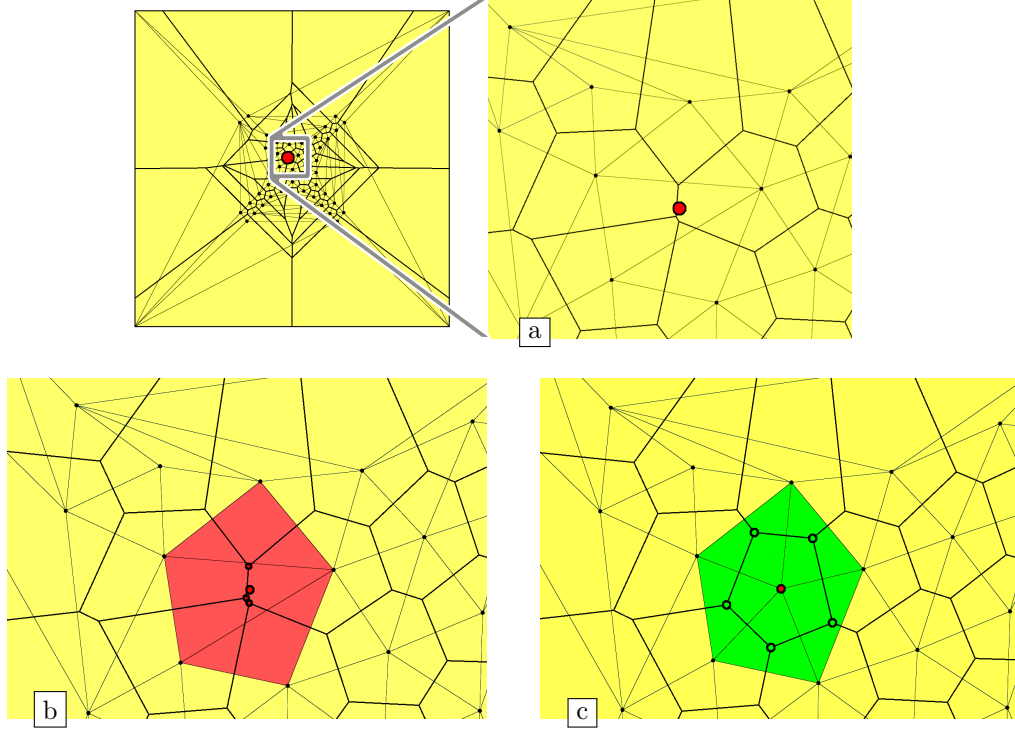


FIGURE 7. Iterative construction of the classical Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ with the Bowyer-Watson algorithm in [Section 4.1.2](#); (a) Insertion of a new seed \mathbf{s}_n (in red) into the diagram $\mathbf{Lag}(\widehat{\mathbf{s}}^{n-1}, \widehat{\boldsymbol{\psi}}^{n-1})$ under construction; (b) Some of the vertices of the current diagram (in red) cannot exist in the new diagram because they lie inside the Laguerre cell of the new seed \mathbf{s}_n ; (c) The cavity formed by the associated dual triangles (in red) is removed and the vertices of the new Laguerre cell are dual to triangles connecting the new seed to those at the border of the cavity (in green).

condition and symmetrizing the determinant by row manipulations, [\(4.5\)](#) rewrites:

$$(4.6) \quad C(\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k, \mathbf{s}_n, \psi_i, \psi_j, \psi_k, \psi_n) := \begin{vmatrix} s_{i,1} & s_{i,2} & s_{i,1}^2 + s_{i,2}^2 - \psi_i & 1 \\ s_{j,1} & s_{j,2} & s_{j,1}^2 + s_{j,2}^2 - \psi_j & 1 \\ s_{k,1} & s_{k,2} & s_{k,1}^2 + s_{k,2}^2 - \psi_k & 1 \\ s_{n,1} & s_{n,2} & s_{n,1}^2 + s_{n,2}^2 - \psi_n & 1 \end{vmatrix} > 0.$$

One then constructs the cavity \mathcal{C}^{n-1} , defined as the collection of the dual triangles in \mathcal{T}^{n-1} associated to the vertices in conflict with \mathbf{s}_n , see the red triangles in [Fig. 7](#) (b). With a small abuse of notations, we use the same notation for this collection of triangles and the closed subset of \overline{D} formed by their reunion. It can be shown that \mathcal{C}^{n-1} is a connected set containing \mathbf{s}_n . Hence the identification of \mathcal{C}^{n-1} starts by finding one triangle $T \in \mathcal{T}^{n-1}$ containing \mathbf{s}_n , and then proceeds by propagation through the neighbors of T while they are in conflict with \mathbf{s}_n , in the sense that [\(4.6\)](#) holds, see [Fig. 7](#) (b). Finally, the triangles in the cavity \mathcal{C}^{n-1} are removed from the dual triangulation \mathcal{T}^{n-1} , and the new triangles dual to the vertices of the new Laguerre cell $\mathbf{Lag}_n(\widehat{\mathbf{s}}^n, \widehat{\boldsymbol{\psi}}^n)$ are constructed. Each of them connects \mathbf{s}_n with an edge of the boundary of \mathcal{C}^{n-1} , see [Fig. 7](#) (c). This procedure is iterated until all the seed points in \mathbf{s} have been inserted.

The Bowyer-Watson algorithm crucially hinges on geometric predicates, that is, procedures taking combinatorial decisions based on the fulfillment of inequalities between the coordinates of some geometric objects. Notably, the conflict test [\(4.6\)](#) is based on a predicate; moreover, the aforementioned search for a dual triangle in \mathcal{C}^{n-1} containing the inserted seed point \mathbf{s}_n requires another predicate, measuring the relative

orientation of three points. The implementation of both predicates requires a particular attention, in order to overcome the lack of precision of standard computer arithmetics. They amount to evaluate the sign of a polynomial function evaluated at the coordinates of the seed points \mathbf{s}_i and weights ψ_i . Such a quantity can be exactly computed at a reasonable computational cost thanks to a combination of different techniques, see [117, 76] and references therein.

The above Bowyer-Watson procedure also rests on the assumption that the seed points \mathbf{s} and the weights ψ are in a generic configuration (see again [Theorem 3.1](#) and [Remark 3.3](#)). In practice, this assumption guarantees that the determinant in the characterization (4.6) of the dual triangles in the cavity \mathcal{C}^{n-1} does not vanish – a situation where \mathbf{s}_n would lie on the boundary of this cavity, making the procedure invalid. However, “pathologic” configurations where this determinant vanishes happen in practice, and they require a specific treatment. For instance, if $\psi = \mathbf{0}$ (and then $\mathbf{Lag}(\mathbf{s}, \psi)$ is the Voronoi diagram $\mathbf{Vor}(\mathbf{s})$), the predicate (4.6) evaluates whether the added seed point \mathbf{s}_n lies inside the disk circumscribed to the triangle with vertices $\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k$; the determinant in (4.6) then equals zero if the four seed points $\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k, \mathbf{s}_n$ are cocyclic. In this case, two triangulations of the cavity \mathcal{C}^{n-1} based on \mathbf{s}_n are possible, and one would need to consistently choose among them.

One possibility to treat situations where the determinant $C(\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k, \mathbf{s}_n, \psi_i, \psi_j, \psi_k, \psi_n)$ vanishes is to evade from singular configurations thanks to the symbolic perturbation approach [52]: we conceptually perturb each weight ψ_i as $\psi_i + \varepsilon^i$, where $\varepsilon \ll 1$ is a very small parameter, and we consider the expansion of the associated determinant in terms of the powers of ε :

$$(4.7) \quad C(\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k, \mathbf{s}_l, \psi_i + \varepsilon^i, \psi_j + \varepsilon^j, \psi_k + \varepsilon^k, \psi_n + \varepsilon^n) = \begin{vmatrix} s_{i,1} & s_{i,2} & s_{i,1}^2 + s_{i,2}^2 - \psi_i - \varepsilon^i & 1 \\ s_{j,1} & s_{j,2} & s_{j,1}^2 + s_{j,2}^2 - \psi_j - \varepsilon^j & 1 \\ s_{k,1} & s_{k,2} & s_{k,1}^2 + s_{k,2}^2 - \psi_k - \varepsilon^k & 1 \\ s_{n,1} & s_{n,2} & s_{n,1}^2 + s_{n,2}^2 - \psi_n - \varepsilon^n & 1 \end{vmatrix} \\ = C(\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k, \mathbf{s}_l, \psi_i, \psi_j, \psi_k, \psi_n) - \varepsilon^i \begin{vmatrix} s_{j,1} & s_{j,2} & 1 \\ s_{k,1} & s_{k,2} & 1 \\ s_{n,1} & s_{n,2} & 1 \end{vmatrix} + \varepsilon^j \begin{vmatrix} s_{i,1} & s_{i,2} & 1 \\ s_{k,1} & s_{k,2} & 1 \\ s_{n,1} & s_{n,2} & 1 \end{vmatrix} \\ - \varepsilon^k \begin{vmatrix} s_{i,1} & s_{i,2} & 1 \\ s_{j,1} & s_{j,2} & 1 \\ s_{n,1} & s_{n,2} & 1 \end{vmatrix} + \varepsilon^n \begin{vmatrix} s_{i,1} & s_{i,2} & 1 \\ s_{j,1} & s_{j,2} & 1 \\ s_{k,1} & s_{k,2} & 1 \end{vmatrix},$$

where the second equality follows from the multilinearity of the determinant. In the last expression of the above right-hand side, the factor of ε^n does not vanish since according to (G2) the seed points $\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k$ are not aligned; hence, using the sign of the leading term in the above expansion in powers of ε results in a modified predicate which never returns 0 and takes consistent combinatorial decisions in the presence of degeneracies.

Step 2: Computation of $\mathbf{V}(\mathbf{s}, \psi)$ by truncation of $\mathbf{Lag}(\mathbf{s}, \psi)$.

For each $i = 1, \dots, N$, the i^{th} cell $V_i(\mathbf{s}, \psi)$ of the modified diagram $\mathbf{V}(\mathbf{s}, \psi)$ is obtained by computing the intersection between its classical counterpart $\text{Lag}_i(\mathbf{s}, \psi)$ with the ball centered at \mathbf{s}_i with radius $\sqrt{\psi_i}$, as depicted in [Fig. 6](#) (c). According to [Definition 3.3](#), the boundary of $V_i(\mathbf{s}, \psi)$ is composed of line segments – some of them pertaining to the boundary of the computational domain D , possibly bearing particular references inherited from the latter – and, occasionally, circular arcs, see [Figs. 2](#) and [5](#).

As we shall see more precisely in the next [Section 5](#), the mechanical computations involved in the evaluation of the shape functionals of our optimization problem (P) rely on a discretization of Ω made of convex polytopes. The latter is constructed via the procedure described in [78]: a number n_{arc} is chosen, and each circular arc is discretized into a polygonal line made of n_{arc} line segments and $(n_{\text{arc}} + 1)$ vertices. The calculation of the positions of these vertices from the datum of the seed points \mathbf{s} is detailed in [Appendix A](#).

4.1.3. Computation of the gradient and of the Hessian matrix of the Kantorovich functional

In Steps 2 and 4 of [Algorithm 2](#), the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi}^n)$ is used to evaluate the entries of the gradient $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n) = \nabla_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)$ and of the Hessian matrix $\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n) = \nabla_{\boldsymbol{\psi}}^2 K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)$ of the Kantorovich functional $K(\mathbf{s}, \boldsymbol{\nu}, \cdot)$ at $\boldsymbol{\psi}^n$.

These components depend on the measures of the cells $|V_i(\mathbf{s}, \boldsymbol{\psi}^n)|$ of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi}^n)$ and on the lengths of their edges \mathbf{e}_{ij} . Their expressions have been calculated in e.g. [\[73, 79\]](#); for the sake of completeness, they are recalled in [Appendix B](#), where an intuitive proof is provided under simplifying assumptions.

4.1.4. Solution mechanism for the linear systems

The calculation of the Newton step \mathbf{p}^n in Step 5 of [Algorithm 2](#) is based on the solution of the following linear system with size $N \times N$:

$$(4.8) \quad [\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)] \mathbf{p}^n = -\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n).$$

When two-dimensional applications are concerned, featuring a relatively small number N of seed points (say, $N < 10^5$), direct methods such as Gaussian elimination or LDL^T factorization are available.

Larger configurations raise the need to use an iterative solver, that takes advantage of the sparse nature of the matrix involved in [\(4.8\)](#), see again its expression in [Appendix B](#). The Conjugate Gradient algorithm [\[70\]](#) combined with the simple Jacobi preconditionner (which involves division by the diagonal coefficients) is an efficient candidate in this perspective.

When very large problems are considered (where the number N of seed points exceeds 10^6), yet another strategy is handy. The latter is based on the fact that the linear system [\(4.8\)](#) corresponds to the discretization of a Poisson equation with \mathbb{P}_1 finite elements. This calls for a multigrid method, that can solve such a Poisson system in linear time. In our case, the mesh characterizing this discretization is irregular and we recommend the use of an algebraic multigrid method [\[50\]](#). Roughly speaking, such a multi-grid method creates a hierarchy of operators by merging nodes in the graph that corresponds to the non-zero entries of the sparse matrix. This technique made it possible to solve 3d problems featuring up to 10^8 points in our previous works [\[77, 84\]](#).

4.1.5. Stopping criterion

The components of the gradient $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n) = \nabla_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)$ of the Kantorovich functional express the difference between the target cell measures ν_i and the measures $|V_i(\mathbf{s}, \boldsymbol{\psi}^n)|$ of the cells of the actual version $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi}^n)$ of the diagram. Hence, the supremum norm of this gradient appraises the largest error on the fulfillment of the cell measure constraint. This geometric interpretation of $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)$ suggests a natural convergence criterion for [Algorithm 2](#): the Newton procedure is terminated as soon as the largest measure error is smaller than 1 % of the smallest measure prescription, i.e. when the following inequality is satisfied:

$$|\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)|_{\infty} < \varepsilon_{\text{Newt}}, \quad \text{where } \varepsilon_{\text{Newt}} = 0.01 \left(\min_{i=1, \dots, N} \nu_i \right).$$

4.1.6. Determination of the descent parameter

The selection of a suitable descent parameter α^n for the update of the weight vector $\boldsymbol{\psi}$ in Step 6 of [Algorithm 2](#) deserves a particular attention. Indeed, the invertibility of the Hessian matrix $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)$ of the Kantorovich functional involved in the linear system [\(4.8\)](#) for the Newton step \mathbf{p}^n crucially requires that none of the cells $V_i(\mathbf{s}, \boldsymbol{\psi}^n)$ is empty, $n = 0, \dots, N$, see again the expressions in [Appendix B](#). Hence, the choice of α^n must guarantee that no cell $V_i(\mathbf{s}, \boldsymbol{\psi}^{n+1})$ in the updated diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi}^{n+1})$ is empty.

To ensure this property, we rely on the Kitagawa-Méridot-Thibert strategy, described in [Algorithm 3](#) and analyzed in [\[73\]](#): starting from the Newton step $\alpha^{n,0} = 1$, the descent parameter is halved until the magnitude of the gradient $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)$ has sufficiently decreased while the size of the smallest cell in $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi}^{n+1})$ is larger than a certain threshold ν_{\min} ; in practice, we take:

$$(4.9) \quad \nu_{\min} = \frac{1}{2} \min \left(\min_{i=0,1, \dots, N} |V_i(\mathbf{s}, \mathbf{0})|, \min_{i=0,1, \dots, N} \nu_i \right).$$

Using the so-computed descent parameter, the convergence of the Newton algorithm [Algorithm 2](#) is proven, see [\[73\]](#) for the details.

Algorithm 3 Determination of a suitable descent parameter for the update of the weight vector

Input: The current Newton iterate n in [Algorithm 2](#), characterized by the datum of:

- The collection of seed points $\mathbf{s} = \{s_1, \dots, s_N\}$;
- The current weight vector $\boldsymbol{\psi}^n$;
- The current Newton step vector \mathbf{p}^n ;
- The initial guess about the descent parameter $\alpha^{n,0} = 1$.

for $k = 0, \dots$, until convergence **do**

(1) **if** $\min_{i=0, \dots, N} |V_i(\boldsymbol{\psi}^n + \alpha^{n,k} \mathbf{p}^n)| > \nu_{\min}$ and $|\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n + \alpha^{n,k} \mathbf{p}^n)| \leq (1 - \frac{\alpha^{n,k}}{2}) |\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)|$ **exit**
loop

(2) Update the descent parameter as: $\alpha^{n,k+1} = \alpha^{n,k} / 2$.

(3) Compute the new diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi}^n + \alpha^{n,k+1} \mathbf{p}^n)$

end for

return Descent parameter $\alpha^n := \alpha^{n,k}$.

4.2. Smoothing of a diagram by a variant of Lloyd’s algorithm

As we shall describe more extensively in [Section 5](#), the polygonal mesh induced by the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ representing the shape Ω in our optimal design [Algorithm 1](#) is used, in particular, as the numerical support for the solution of boundary value problems posed on Ω , by means of the Virtual Element Method. The accuracy of the latter, like that of any Galerkin-based solution strategy, is tightly related to the “quality” of this polygonal mesh; intuitively, this notion appraises how close its polygons are from being regular [[122](#), [121](#)]. Unfortunately, [Algorithm 1](#) does not offer any guarantee about the quality of the cells of the diagrams $\mathbf{V}(\mathbf{s}^n, \boldsymbol{\psi}^n)$ at play, and it is necessary to periodically stop the process, say every 3, 4 iterations, to improve their quality.

One possibility to achieve this goal borrows from the famous Lloyd’s algorithm for Voronoi diagrams [[82](#)]. Roughly speaking, the latter produces a “well-shaped” centroidal Voronoi tessellation (i.e. where the seed point of each cell coincides with its centroid) from an arbitrary initial diagram $\mathbf{Vor}(\mathbf{s})$; it proceeds within a series of iterations, by replacing the seed point \mathbf{s}_i of each cell $i = 1, \dots, N$ by the centroid \mathbf{c}_i of the latter, and then computing the new diagram $\mathbf{Vor}(\mathbf{c})$. Often, a relaxed version of this procedure is used, whereby each seed point \mathbf{s}_i is moved towards \mathbf{c}_i for a short pseudo-time step $\alpha \in (0, 1)$.

In our strategy, we rely on a natural extension of this procedure to the present context of Laguerre diagrams in which the measure of each cell is imposed. This was recently proposed in [[44](#), [45](#), [133](#)], see also [[27](#)]; it is sketched in [Algorithm 4](#).

Algorithm 4 Smoothing of a diagram with constrained cell measures by a variant of Lloyd’s algorithm

Inputs:

- Diagram $\mathbf{V}(\mathbf{s}^0, \boldsymbol{\psi}^0)$ induced by:
 - An initial collection of seed points $\mathbf{s}^0 \in \mathbb{R}^N$;
 - A given vector $\boldsymbol{\nu} \in \mathbb{R}^N$ of cell measures;
 - The unique weight vector $\boldsymbol{\psi}^0 \in \mathbb{R}^N$ such that each cell $V_i(\mathbf{s}^0, \boldsymbol{\psi}^0)$ has measure ν_i .
- Relaxation parameter $\alpha \in (0, 1)$.

for $n = 0, \dots$, until convergence **do**

(1) Calculate the collection $\mathbf{c}^n := \{\mathbf{c}_i^n\}_{i=1, \dots, N}$ of the centroids $\mathbf{c}_i^n \in \mathbb{R}^d$ of the cells $V_i(\mathbf{s}^n, \boldsymbol{\psi}^n)$.

(2) Update the seed points as: $\mathbf{s}^{n+1} = (1 - \alpha) \mathbf{s}^n + \alpha \mathbf{c}^n$.

(3) Calculate the weight vector $\boldsymbol{\psi}^{n+1}$ guaranteeing that $|V_i(\mathbf{s}^{n+1}, \boldsymbol{\psi}^{n+1})| = \nu_i$ for $i = 1, \dots, N$.

(4) Compute the new diagram $\mathbf{V}(\mathbf{s}^{n+1}, \boldsymbol{\psi}^{n+1})$.

end for

return Diagram $\mathbf{V}(\mathbf{s}^n, \boldsymbol{\psi}^n)$ where each cell $V_i(\mathbf{s}^n, \boldsymbol{\psi}^n)$ is “well-shaped” and has measure ν_i

4.3. Additional operations on diagrams

In this section, we outline two geometric operations on diagrams of the form (3.6) and (3.7) that significantly improve the efficiency of the shape optimization Algorithm 2.

4.3.1. Resampling of a diagram

As will be exemplified in the numerical examples of Section 7, the optimization of the shape Ω^n generally entails significant changes in its volume through the iterations $n = 0, \dots$ of the process; it is often relevant to dynamically adjust the number N of cells of the defining diagram $\mathbf{V}(\mathbf{s}^n, \boldsymbol{\psi}^n)$ according to these changes.

To achieve this, we periodically carry out a simple resampling procedure, at the end of every one iteration over, say, 3 or 4 of Algorithm 1: a desired average value $\bar{\nu}$ for the measures of the cells of the diagram of the shape Ω^n is given, and we infer the suitable number N^n of cells in the latter via the following relation:

$$N^n = \frac{\text{Vol}(\Omega^n)}{\bar{\nu}}.$$

We then add or delete seed points from the collection \mathbf{s}^n to attain this number, according to the following rules:

- Seed points \mathbf{s}_i are added inside the regions of Ω^n lying far from the boundary of Ω^n (so that this entity is unchanged in the process), where the local cell measures are largest. The measures ν_i of the corresponding added cells are set to $\bar{\nu}$, and the (larger) measures of the neighboring cells are decreased so that the volume of Ω^n is unaltered by this operation.
- The deleted seed points \mathbf{s}_i lie far from $\partial\Omega^n$, and they correspond to cells $V_i(\mathbf{s}^n, \boldsymbol{\psi}^n)$ with “small” measures. Their mass is redistributed to the neighboring cells, so that $\text{Vol}(\Omega^n)$ is unchanged in the process.

4.3.2. Elimination of material “islands” disconnected from the main structure

As described in Section 4.1.2, some of the edges of the cell in the diagrams $\mathbf{V}(\mathbf{s}^n, \boldsymbol{\psi}^n)$ produced by Algorithm 1 may bear particular labels, inherited from the boundary ∂D of the computational domain. These notably serve to identify the regions of $\partial\Omega^n$ bearing particular boundary conditions in the physical problem at play. For instance, in the settings of Sections 2.2.1 and 2.2.2, they characterize the regions Γ_D and Γ_N supporting homogeneous Dirichlet and inhomogeneous Neumann boundary conditions. In practice, it may be desirable to remove the components of Ω^n that are not connected to any cell bearing such a label. For instance, in the situation of Section 2.2.2, the regions of Ω^n that are not connected to Γ_D cause the problem (Elas) to be ill-posed, since the elastic displacement \mathbf{u}_{Ω^n} is only characterized by this problem up to a rigid-body motion inside each such material “islands”.

The elimination of these regions is realized thanks to a simple algorithmic procedure, based on the connectivity of $\mathbf{V}(\mathbf{s}^n, \boldsymbol{\psi}^n)$. We start by storing into a pile all the indices $i \in \{1, \dots, N\}$ belonging to cells having one edge bearing the desired label. We then travel the cells of the diagram by propagating through the neighbors of the elements in the pile, and we thus obtain all the indices of the connected component of Ω^n attached to the label at stake. We eventually discard all the cells (i.e. the associated seed points) in $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ that have not been visited in this process.

5. MECHANICAL COMPUTATIONS ON Ω VIA THE VIRTUAL ELEMENT METHOD

In this section, we describe – in 2d for simplicity – the resolution of physical boundary value problems of the form (Cond) and (Elas) on a polygonal mesh \mathcal{T} of the shape Ω . Different numerical methods are available to achieve this goal. Although they are slightly unusual in the physical contexts at stake in this article, let us notably mention finite volume methods – see [33] for the treatment of heat conduction problems, or [72] for linear elasticity –, discontinuous Galerkin methods [32], or the recent Network Element Method [38, 39].

In the present work, we rely on the Virtual Element Method, which is an elegant adaptation of the well-known Finite Element Method to the solution of boundary value problems on arbitrary polygonal meshes. Although fairly recent, this method has aroused much enthusiasm within the academic and industrial communities; it lends itself to rigorous analysis in a spirit relatively similar to that of the Finite Element Method. This admittedly comes at the expense of a relatively higher level of implementation complexity, especially in three space dimensions, partly due to the necessity of coping with general polygonal meshes.

We refer to e.g. [12, 15, 40, 60, 125] for extensive presentations of the Virtual Element Method, to [18, 86] for comparisons with the Finite Element Method on model problems, and to [42] for a recent open-source implementation.

After setting notations in Section 5.1, we present this method in Section 5.2 in the context of the 2d conductivity equation (Cond), where its salient features can be exposed in a relatively non technical manner. This presentation is not intended to be minimal; rather, it prepares the ground for the treatment of the linear elasticity system which is discussed next in Section 5.3. Practical implementation details are deferred to Appendix D.

5.1. Notations

Throughout this section, Ω is a two-dimensional shape equipped with a polygonal mesh \mathcal{T} . The latter is composed of N (closed) elements E_i , $i = 1, \dots, N$, and M vertices, denoted by \mathbf{q}_j , $j = 1, \dots, M$. Each polygon $E \in \mathcal{T}$ is characterized by its n^E vertices, which form a subset $\{\mathbf{q}_j^E\}_{j=1, \dots, n^E}$ of $\{\mathbf{q}_j\}_{j=1, \dots, M}$; these are numbered in a counterclockwise fashion and we set $\mathbf{q}_0^E := \mathbf{q}_{n^E}^E$ and $\mathbf{q}_{n^E+1}^E := \mathbf{q}_1^E$, see Fig. 8 (a). We also introduce the following notation:

- The diameter of E is denoted by $h^E := \sup_{x, y \in E} |x - y|$.
- For $i = 1, \dots, n^E - 1$, \mathbf{e}_i is the edge with endpoints \mathbf{q}_i^E and \mathbf{q}_{i+1}^E ; $\mathbf{e}_0 = \mathbf{e}_{n^E}$ stands for the edge between $\mathbf{q}_{n^E}^E$ and \mathbf{q}_1^E .
- For $i = 0, \dots, n^E$, we denote by $\mathbf{n}_{\mathbf{e}_i}$ the unit normal vector to \mathbf{e}_i , pointing outward E .
- For $i = 1, \dots, n^E$, we denote by $\hat{\mathbf{e}}_i$ the edge between \mathbf{q}_{i-1}^E and \mathbf{q}_{i+1}^E , and by $\mathbf{n}_{\hat{\mathbf{e}}_i}$ the unit normal vector to $\hat{\mathbf{e}}_i$, oriented so that $\langle \mathbf{n}_{\hat{\mathbf{e}}_i}, \mathbf{n}_{\mathbf{e}_j} \rangle \geq 0$ for $j \in \{i-1, i\}$. A simple calculation shows that

$$(5.1) \quad |\mathbf{e}_{i-1}| \mathbf{n}_{\mathbf{e}_{i-1}} + |\mathbf{e}_i| \mathbf{n}_{\mathbf{e}_i} = |\hat{\mathbf{e}}_i| \mathbf{n}_{\hat{\mathbf{e}}_i}.$$

- For any continuous function $v : E \rightarrow \mathbb{R}$, we denote by

$$\bar{v} = \frac{1}{n^E} \sum_{j=1}^{n^E} v(\mathbf{q}_j^E)$$

the average of v over the vertices of E ; the same notation is used when v is replaced by a vector field or a tensor field on E .

- Accordingly, $\bar{\mathbf{q}}^E = (\bar{q}_1^E, \bar{q}_2^E)$ is the average position of the vertices of E , i.e.

$$(5.2) \quad \bar{\mathbf{q}}^E = \frac{1}{n^E} \sum_{j=1}^{n^E} \mathbf{q}_j^E;$$

note that $\bar{\mathbf{q}}^E$ differs from the centroid of E .

- For any integrable function $v : E \rightarrow \mathbb{R}$, we denote the average of v over E by

$$\langle v \rangle = \frac{1}{|E|} \int_E v(\mathbf{x}) \, d\mathbf{x};$$

the same notation is used when v is replaced by a vector field or a tensor field on E .

5.2. The Virtual Element Method for the 2d conductivity equation

This section deals with the implementation of the Virtual Element Method for the solution of the conductivity equation equipped with homogeneous Dirichlet boundary conditions:

$$(Cond) \quad \begin{cases} -\operatorname{div}(\gamma \nabla u) = f & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega, \end{cases}$$

where the sought function $u \in H_0^1(\Omega)$ is denoted without reference to the (fixed) domain Ω in this section, the right-hand side $f \in L^2(\Omega)$ stands for a source term, and the conductivity γ is assumed to be a positive constant. Our presentation relies on the pedagogic article [125], see also [15]. We provide a general overview of the method in Section 5.2.1 before focusing on the definition of the local space of discrete functions in Section 5.2.2.

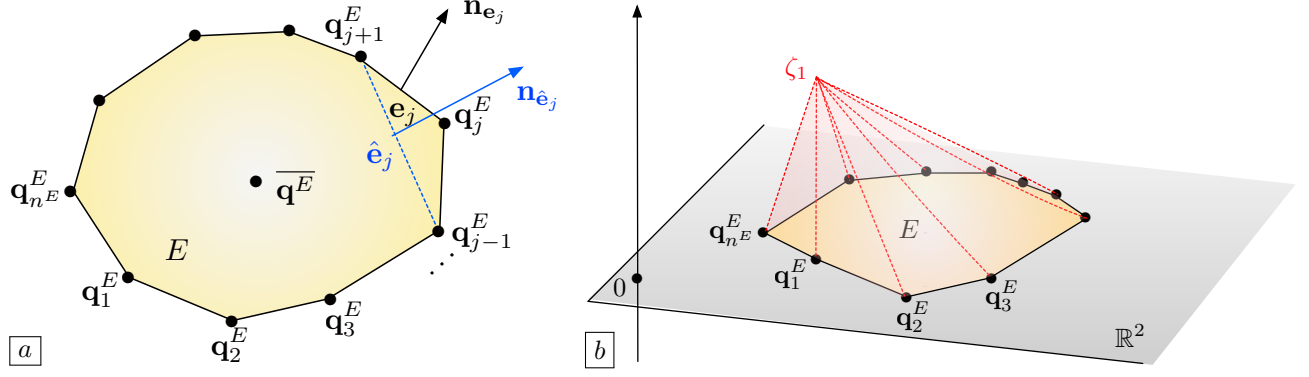


FIGURE 8. (a) One polygonal element $E \in \mathcal{T}$; (b) Graph of the basis function ζ_1 , attached to vertex \mathbf{q}_1^E (see Section 5.2.2).

5.2.1. General presentation of the method

Like the Finite Element Method (see e.g. [37]), the Virtual Element Method leverages the variational formulation (2.2) of (Cond), which reads:

$$(5.3) \quad \text{Search for } u \in V \text{ s.t. } \forall v \in V, \quad a(u, v) = \ell(v).$$

Here, the functional space V is $H_0^1(\Omega)$, and we have posed:

$$(5.4) \quad \forall u, v \in V, \quad a(u, v) = \gamma \int_{\Omega} \nabla u \cdot \nabla v \, dx \quad \text{and} \quad \ell(v) = \int_{\Omega} f v \, dx.$$

The discretization of (5.3) relies on a finite-dimensional subspace $\mathcal{W}_{\mathcal{T}}$ of V ; the numerical approximation $u_{\mathcal{T}} \in \mathcal{W}_{\mathcal{T}}$ of u is sought as the solution to the problem:

$$(5.5) \quad \text{Search for } u_{\mathcal{T}} \in \mathcal{W}_{\mathcal{T}} \text{ s.t. } \forall v \in \mathcal{W}_{\mathcal{T}}, \quad a(u_{\mathcal{T}}, v) = \ell(v).$$

The space $\mathcal{W}_{\mathcal{T}}$ is defined by:

$$(5.6) \quad \mathcal{W}_{\mathcal{T}} = \{u \in \mathcal{C}(\overline{\Omega}) \text{ and } u|_E \in \mathcal{W}(E) \text{ for all } E \in \mathcal{T}\},$$

where for each element $E \in \mathcal{T}$, $\mathcal{W}(E)$ is a local space of functions on E whose definition is the topic of the next Section 5.2.2. For the moment, let us solely mention that the dimension of $\mathcal{W}_{\mathcal{T}}$ coincides with the number M of vertices in the mesh \mathcal{T} , and that the values of functions $u \in \mathcal{W}_{\mathcal{T}}$ at the vertices \mathbf{q}_j , $j = 1, \dots, M$ are the degrees of freedom of the discretization. We introduce a basis $\{\varphi_k\}_{k=1, \dots, M}$ of $\mathcal{W}_{\mathcal{T}}$ as follows: for $k = 1, \dots, M$, the function $\varphi_k \in \mathcal{W}_{\mathcal{T}}$ is attached to the vertex \mathbf{q}_k in the sense that:

$$(5.7) \quad \forall l = 1, \dots, M, \quad \varphi_k(\mathbf{q}_l) = \begin{cases} 1 & \text{if } k = l, \\ 0 & \text{otherwise.} \end{cases}$$

By writing the decomposition $u_{\mathcal{T}} = \sum_{l=1}^M u_l \varphi_l$ of the sought function $u_{\mathcal{T}}$ on this basis, and using $v = \varphi_k$ as test function in (5.5) for $k = 1, \dots, M$, the problem (5.3) and (5.4) boils down to the following $M \times M$ linear system:

$$(5.8) \quad K_{\mathcal{T}} U_{\mathcal{T}} = F_{\mathcal{T}},$$

where $U_{\mathcal{T}} := (u_1, \dots, u_M) \in \mathbb{R}^M$, and the entries of the stiffness matrix $K_{\mathcal{T}} \in \mathbb{R}^{M \times M}$ and force vector $F_{\mathcal{T}} \in \mathbb{R}^M$ are given by:

$$(5.9) \quad K_{\mathcal{T},kl} = a(\varphi_l, \varphi_k), \quad \text{and} \quad F_{\mathcal{T},k} = \ell(\varphi_k), \quad k, l = 1, \dots, M.$$

The integrals defining these quantities are sums of contributions from each element $E \in \mathcal{T}$, i.e.

$$(5.10) \quad a(\varphi_l, \varphi_k) = \sum_{E \in \mathcal{T}} a^E(\varphi_l, \varphi_k), \quad \text{where } a^E(u, v) := \gamma \int_E \nabla u \cdot \nabla v \, dx, \quad \text{and}$$

$$F_{\mathcal{T}, k} = \sum_{E \in \mathcal{T}} \ell^E(\varphi_k), \quad \text{where } \ell^E(v) := \int_E f v \, dx.$$

Hence, the assembly of (5.8) is realized by calculating the local, element-wise contributions $a^E(\varphi_l, \varphi_k)$, $\ell^E(\varphi_k)$ to the entries $K_{\mathcal{T}, kl}$ of the global stiffness matrix and $F_{\mathcal{T}, k}$ of the global force vector, respectively. These computations are discussed in the next [Section 5.2.2](#) and [Appendix D](#).

5.2.2. The local space $\mathcal{W}(E)$ attached to an element $E \in \mathcal{T}$

Throughout this section, E is a given polygon in \mathcal{T} ; we define $\mathcal{W}(E)$ as the space of functions $u : E \rightarrow \mathbb{R}$ satisfying the following properties:

- The restriction $u|_{\mathbf{e}} : \mathbf{e} \rightarrow \mathbb{R}$ of u to any edge \mathbf{e} of E is affine;
- The differential operator of (Cond) cancels u , i.e. $-\operatorname{div}(\gamma \nabla u) = -\gamma \Delta u = 0$ in E .

Obviously, the functions $u \in \mathcal{W}(E)$ are uniquely determined by their values at the vertices \mathbf{q}_j^E of E , $j = 1, \dots, n^E$; in particular, $\mathcal{W}(E)$ has dimension n^E and it contains the space $\mathcal{P}(E)$ of affine functions on E :

$$\mathcal{P}(E) := \{u(\mathbf{x}) = a + \langle \mathbf{b}, \mathbf{x} \rangle, \quad a \in \mathbb{R}, \quad \mathbf{b} \in \mathbb{R}^2\}.$$

We then define a basis $\{\zeta_i\}_{i=1, \dots, n^E}$ of $\mathcal{W}(E)$ as follows: for $i = 1, \dots, n^E$, ζ_i is the unique function in $\mathcal{W}(E)$ such that:

$$(5.11) \quad \zeta_i(\mathbf{q}_j^E) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

Observe that the dependency of the functions ζ_i on the actual element E is omitted for notational simplicity.

Remark 5.1.

- Several definitions of $\mathcal{W}(E)$ are actually possible, all of them leading to discrete functions $u \in \mathcal{W}_{\mathcal{T}}$ whose degrees of freedom are their values at the vertices of \mathcal{T} . In fact, a careful selection of $\mathcal{W}(E)$ may significantly ease the numerical implementation, see for instance [Appendix D.3](#) for an alternative definition of the behavior of functions $u \in \mathcal{W}(E)$ inside E which lends itself to an easier evaluation of integrals of the form $\int_E uv \, dx$, $u, v \in \mathcal{W}(E)$, or the appendix in [60] for another definition which is better adapted to the evaluation of quadrature formulas in the 3d elasticity setting.
- Higher-order versions of $\mathcal{W}(E)$, featuring higher degree polynomials, could also be considered, leading to higher-order approximations $u_{\mathcal{T}}$ of u .

With these notations, the global basis functions φ_k defined in (5.7) can be expressed in terms of the local functions ζ_i ; more precisely:

$$\forall k = 1, \dots, M, \quad E \in \mathcal{T}, \quad \varphi_k|_E = \begin{cases} \zeta_i & \text{if there exists } i \in \{1, \dots, n^E\} \text{ s.t. } \mathbf{q}_i^E = \mathbf{q}_k, \\ 0 & \text{otherwise.} \end{cases}$$

As a result, the computation of the entries of the stiffness matrix $K_{\mathcal{T}}$ and force vector $F_{\mathcal{T}}$ in (5.10) boils down to that of the local building blocks

$$(5.12) \quad K_{ij}^E := a^E(\zeta_i, \zeta_j) \quad \text{and} \quad F_i^E := \ell^E(\zeta_i), \quad i, j = 1, \dots, n^E.$$

Unfortunately, no explicit expression of the ζ_i is available to realize these computations; the key idea of the Virtual Element Method is to take advantage of the definition of $\mathcal{W}(E)$ to approximate them with a computation which uses the sole property (5.11). To achieve this, let us introduce:

- The subspace $\mathcal{W}_R(E) \subset \mathcal{P}(E)$ of constant functions on E :

$$\mathcal{W}_R(E) := \operatorname{span}\{m_1\}, \quad \text{where } m_1(\mathbf{x}) = 1.$$

- The subspace $\mathcal{W}_C(E) \subset \mathcal{P}(E)$ of functions with zero mean value over the vertices of E :

$$\mathcal{W}_C(E) := \{u \in \mathcal{P}(E), \bar{u} = 0\}.$$

It is spanned by the functions

$$m_2(\mathbf{x}) = x_1 - \overline{q_1^E}, \text{ and } m_3(\mathbf{x}) = x_2 - \overline{q_2^E},$$

where we recall the notation $\overline{\mathbf{q}^E}$ for the average of the vertices of E , see (5.2).

It is easily seen that $\mathcal{P}(E)$ is the direct sum of $\mathcal{W}_R(E)$ and $\mathcal{W}_C(E)$:

$$\mathcal{P}(E) = \mathcal{W}_R(E) \oplus \mathcal{W}_C(E).$$

We next introduce projection operators π_R and π_C onto $\mathcal{W}_R(E)$ and $\mathcal{W}_C(E)$, respectively: for any (smooth enough) function $v : E \rightarrow \mathbb{R}$, $\pi_R v \in \mathcal{W}_R(E)$ and $\pi_C v \in \mathcal{W}_C(E)$ are respectively defined by:

$$\pi_R v(\mathbf{x}) = \bar{v}, \text{ and } \pi_C v(\mathbf{x}) = \langle \nabla v \rangle \cdot (\mathbf{x} - \overline{\mathbf{q}^E}).$$

With these notations, the following relations are easily verified:

$$\forall c \in \mathcal{W}_C(E), \pi_R c = 0, \pi_C c = c \text{ and } \forall r \in \mathcal{W}_R(E), \pi_C r = 0, \pi_R r = r.$$

We finally define the projection $\pi_{\mathcal{P}} : \mathcal{W}(E) \rightarrow \mathcal{P}(E)$ onto affine functions by:

$$(5.13) \quad \pi_{\mathcal{P}} v = \pi_C v + \pi_R v.$$

The following lemma draws elementary, albeit crucial properties of these objects.

Lemma 5.1. *The following facts hold true:*

- (i) *The operators π_R and π_C are the “orthogonal projections” from $\mathcal{W}(E)$ onto $\mathcal{W}_R(E)$ and $\mathcal{W}_C(E)$, respectively, in the sense of the bilinear form $a^E(\cdot, \cdot)$:*

$$\forall r \in \mathcal{W}_R(E), c \in \mathcal{W}_C(E), v \in \mathcal{W}(E), \quad a^E(r, v) = 0 \text{ and } a^E(c, v - \pi_C v) = 0.$$

- (ii) *The following decomposition of $a^E(\cdot, \cdot)$ holds true:*

$$(5.14) \quad \forall u, v \in \mathcal{W}(E), \quad a^E(u, v) = a^E(\pi_C u, \pi_C v) + a^E(u - \pi_{\mathcal{P}} u, v - \pi_{\mathcal{P}} v).$$

Proof. (i) The first relation is obvious since functions in $\mathcal{W}_R(E)$ are constant on E . The second one arises from the fact that the gradient ∇c of any function $c \in \mathcal{W}_C(E)$ is constant over E :

$$a^E(c, v - \pi_C v) = \gamma \nabla c \cdot \left(\int_E (\nabla v - \nabla(\pi_C v)) \, d\mathbf{x} \right) = \gamma \nabla c \cdot \left(|E| \langle \nabla v \rangle - |E| \langle \nabla v \rangle \right) = 0.$$

- (ii) A straightforward calculation based on the identities in (i) yields:

$$\begin{aligned} a^E(u, v) &= a^E(\pi_R u + \pi_C u + (u - \pi_{\mathcal{P}} v), \pi_R v + \pi_C v + (v - \pi_{\mathcal{P}} v)) \\ &= a^E(\pi_C u + (u - \pi_{\mathcal{P}} v), \pi_C v + (v - \pi_{\mathcal{P}} v)) \\ &= a^E(\pi_C u, \pi_C v) + a^E(u - \pi_{\mathcal{P}} v, v - \pi_{\mathcal{P}} v). \end{aligned}$$

□

Let us now note that (5.14) gives rise to a decomposition of the local stiffness matrix $K^E \in \mathbb{R}^{n^E \times n^E}$ in (5.12) into two terms:

$$K^E = P^E + S^E.$$

Here, the matrices P^E and $S^E \in \mathbb{R}^{n^E \times n^E}$ are defined by:

$$\forall i, j = 1, \dots, n^E, \quad P_{ij}^E = a^E(\pi_C \zeta_i, \pi_C \zeta_j) \text{ and } S_{ij}^E = a^E(\zeta_i - \pi_{\mathcal{P}} \zeta_i, \zeta_j - \pi_{\mathcal{P}} \zeta_j).$$

The entries of the matrix P^E can be calculated exactly; indeed, the projection $\pi_C u$ of a function $u \in \mathcal{W}(E)$ has coefficients over the basis $\{m_\alpha\}_{\alpha=1,2,3}$ that can be calculated solely from the values of u at the vertices of E , see Appendix D.1 about this fact.

On the other hand, the second term $a^E(u - \pi_{\mathcal{P}} u, v - \pi_{\mathcal{P}} v)$ in the right-hand side of (5.14) (associated to S^E) appraises the behavior of $a^E(\cdot, \cdot)$ on those functions in $\mathcal{W}(E)$ that are not affine. It is not easily

computed, and it is therefore replaced by another bilinear form \tilde{s}^E (and matrix \tilde{S}^E) which can be calculated just from the values of u and v at the vertices of E (see [Appendix D.1](#) about this calculation):

$$\tilde{s}^E(u - \pi_{\mathcal{P}}u, v - \pi_{\mathcal{P}}v), \text{ where } \tilde{s}^E(w, z) := \alpha^E \sum_{i=1}^{n^E} w(\mathbf{q}_i^E) z(\mathbf{q}_i^E),$$

and α^E is a constant which is chosen so that \tilde{s}^E scale as a^E upon refinement of the mesh \mathcal{T} ; in this simple case, we take $\alpha^E = 1$.

This replacement gives rise to the modified bilinear form $\tilde{a}^E(\cdot, \cdot)$ given by:

$$\forall u, v \in \mathcal{W}(E), \quad \tilde{a}^E(u, v) = a^E(\pi_{\mathcal{C}}u, \pi_{\mathcal{C}}v) + \tilde{s}^E(u - \pi_{\mathcal{P}}u, v - \pi_{\mathcal{P}}v).$$

It guarantees the so-called polynomial consistency of \tilde{a}^E with a^E (or ‘‘patch test’’ in engineering): $\tilde{a}^E(u, v)$ and $a^E(u, v)$ coincide when at least one of the functions u or v belongs to $\mathcal{P}(E)$. Indeed,

$$\begin{aligned} \forall p \in \mathcal{P}(E), \quad v \in \mathcal{W}(E), \quad \tilde{a}^E(p, v) &= a^E(\pi_{\mathcal{C}}p, \pi_{\mathcal{C}}v) \\ &= a^E(\pi_{\mathcal{C}}p, v), \\ &= a^E(p, v). \end{aligned}$$

This method can be proved to be convergent, see [\[40\]](#). Intuitively, it behaves at least as well as if only linear polynomials were the only basis functions on each element.

Remark 5.2. *The strategy illustrated in this section can be adapted to the solution of eigenvalue problems of the form [\(2.4\)](#), see [\[62\]](#). The main additional operation required to achieve this purpose is the assembly of the mass matrix, which is described in [Appendix D.3](#).*

5.3. Resolution of the linear elasticity system with the Virtual Element Method

In this section, whose presentation is based on [\[18, 60\]](#), we briefly describe how the strategy presented in the previous [Section 5.2](#) can be adapted to deal with the solution of the 2d linear elasticity system. To emphasize the similarities between the treatments of both situations, we use the same notations for corresponding quantities when no confusion is possible.

Let us recall the setting of [Section 2.2.2](#): the boundary of the shape Ω is decomposed into three disjoint regions $\Gamma_D, \Gamma_N, \Gamma$, where Γ_D is a clamping area, Γ_N is the support of external loads $\mathbf{g} : \Gamma_N \rightarrow \mathbb{R}^2$ and Γ is effort-free. Assuming the presence of body forces $\mathbf{f} : \Omega \rightarrow \mathbb{R}^2$, the elastic displacement $\mathbf{u} : \Omega \rightarrow \mathbb{R}^2$ is the unique solution in $H_{\Gamma_D}^1(\Omega)^2$ to the boundary value problem:

$$(Elas) \quad \begin{cases} -\operatorname{div}(Ae(\mathbf{u})) = \mathbf{f} & \text{in } \Omega, \\ \mathbf{u} = \mathbf{0} & \text{on } \Gamma_D, \\ Ae(\mathbf{u})\mathbf{n} = \mathbf{g} & \text{on } \Gamma_N, \\ Ae(\mathbf{u})\mathbf{n} = \mathbf{0} & \text{on } \Gamma. \end{cases}$$

The associated variational formulation reads, see [\(2.6\)](#):

Search for $\mathbf{u} \in V$ s.t. $\forall \mathbf{v} \in V, \quad a(\mathbf{u}, \mathbf{v}) = \ell(\mathbf{v})$, where $V = H_{\Gamma_D}^1(\Omega)^2$,

$$a(\mathbf{u}, \mathbf{v}) = \int_{\Omega} Ae(\mathbf{u}) : e(\mathbf{v}) \, d\mathbf{x}, \text{ and } \ell(\mathbf{v}) = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, d\mathbf{x} + \int_{\Gamma_N} \mathbf{g} \cdot \mathbf{v} \, ds.$$

As described in [Section 5.2.1](#), the cornerstone of the development of the Virtual Element Method lies in the definition of the local space of discrete functions $\mathcal{W}(E)$ and the assembly of the local stiffness matrix $K^E \in \mathbb{R}^{2n^E \times 2n^E}$ attached to a given polygon E in \mathcal{T} . The local space $\mathcal{W}(E)$ reads:

$$\mathcal{W}(E) = \left\{ \mathbf{u} : E \rightarrow \mathbb{R}^2, \quad \mathbf{u} \text{ is affine on each edge of } \partial E \text{ and } -\operatorname{div}(Ae(\mathbf{u})) = 0 \right\}.$$

Again, an element $\mathbf{u} \in \mathcal{W}(E)$ is uniquely determined by its values at the vertices $\mathbf{q}_j^E, j = 1, \dots, n^E$ of E and $\mathcal{W}(E)$ has dimension $2n^E$. Besides, $\mathcal{W}(E)$ contains the space $\mathcal{P}(E)$ of affine functions, given by:

$$\mathcal{P}(E) := \left\{ \mathbf{u}(\mathbf{x}) = \mathbf{a} + B\mathbf{x}, \quad \mathbf{a} \in \mathbb{R}^2, \quad B \in \mathbb{R}^{2 \times 2} \right\}.$$

Let us introduce the natural basis $\{\zeta_j\}_{j=1,\dots,2n^E}$ of $\mathcal{W}(E)$, defined by, for $i = 1, \dots, n^E$:

$$(5.15) \quad \zeta_{2i-1}(\mathbf{q}_i^E) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \text{ and } \zeta_{2i-1}(\mathbf{q}_j^E) = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \text{ for } j \neq i;$$

$$(5.16) \quad \zeta_{2i}(\mathbf{q}_i^E) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \text{ and } \zeta_{2i}(\mathbf{q}_j^E) = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \text{ for } j \neq i.$$

In a similar state of mind as in [Section 5.2](#), among the elements in $\mathcal{P}(E)$, we shall distinguish the rigid-body motions $\mathbf{u} \in \mathcal{W}_R(E)$, whose strain $e(\mathbf{u})$ equals 0, and the fields $\mathbf{u} \in \mathcal{W}_C(E)$ with constant strain $e(\mathbf{u})$; precisely:

- The subspace $\mathcal{W}_R(E) \subset \mathcal{P}(E)$ of rigid-body motions is defined by

$$\mathcal{W}_R(E) := \left\{ \mathbf{a} + M(\mathbf{x} - \overline{\mathbf{q}}^E), \mathbf{a} \in \mathbb{R}^2, M \in \mathbb{R}^{2 \times 2}, M^T = -M \right\};$$

it is spanned by the three vector fields:

$$(5.17) \quad \mathbf{r}_1(\mathbf{x}) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \mathbf{r}_2(\mathbf{x}) = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \text{and} \quad \mathbf{r}_3(\mathbf{x}) = \begin{pmatrix} -(x_2 - \overline{q}_2^E) \\ x_1 - \overline{q}_1^E \end{pmatrix}.$$

- The subspace $\mathcal{W}_C(E) \subset \mathcal{P}(E)$ of vector fields with constant strain is defined by

$$\mathcal{W}_C(E) := \left\{ S(\mathbf{x} - \overline{\mathbf{q}}^E), S \in \mathbb{R}^{2 \times 2}, S^T = S \right\};$$

it is spanned by the three vector fields:

$$(5.18) \quad \mathbf{c}_1(\mathbf{x}) = \begin{pmatrix} x_1 - \overline{q}_1^E \\ 0 \end{pmatrix}, \quad \mathbf{c}_2(\mathbf{x}) = \begin{pmatrix} 0 \\ x_2 - \overline{q}_2^E \end{pmatrix} \quad \text{and} \quad \mathbf{c}_3(\mathbf{x}) = \begin{pmatrix} x_2 - \overline{q}_2^E \\ x_1 - \overline{q}_1^E \end{pmatrix}.$$

Again, it is easily verified that $\mathcal{P}(E)$ is the direct sum of $\mathcal{W}_R(E)$ and $\mathcal{W}_C(E)$:

$$\mathcal{P}(E) = \mathcal{W}_R(E) \oplus \mathcal{W}_C(E).$$

We next define the projections π_R and π_C from $\mathcal{W}(E)$ onto $\mathcal{W}_R(E)$ and $\mathcal{W}_C(E)$, respectively.

- For any (smooth enough) vector field $\mathbf{u} : E \rightarrow \mathbb{R}^2$, $\pi_R \mathbf{u} \in \mathcal{W}_R(E)$ is defined by:

$$\pi_R \mathbf{u}(\mathbf{x}) = \bar{\mathbf{u}} + \langle \omega(\mathbf{u}) \rangle \begin{pmatrix} -(x_2 - \overline{q}_2^E) \\ x_1 - \overline{q}_1^E \end{pmatrix}, \quad \mathbf{x} \in E;$$

where $\omega(\mathbf{u}) := \frac{1}{2} \left(\frac{\partial u_2}{\partial x_1} - \frac{\partial u_1}{\partial x_2} \right)$ is (half) the scalar curl of \mathbf{u} .

- For any (smooth enough) vector field $\mathbf{u} : E \rightarrow \mathbb{R}^2$, $\pi_C \mathbf{u} \in \mathcal{W}_C(E)$ is defined by:

$$\pi_C \mathbf{u}(\mathbf{x}) = \langle e(\mathbf{u}) \rangle (\mathbf{x} - \overline{\mathbf{q}}^E).$$

The following relations are then easily verified:

$$\forall \mathbf{r} \in \mathcal{W}_R(E), \pi_C \mathbf{r} = \mathbf{0}, \pi_R \mathbf{r} = \mathbf{r}, \text{ and } \forall \mathbf{c} \in \mathcal{W}_C(E), \pi_R \mathbf{c} = \mathbf{0}, \pi_C \mathbf{c} = \mathbf{c}.$$

From these data, we define the projection operator $\pi_{\mathcal{P}}$ onto affine functions in $\mathcal{P}(E)$ by:

$$(5.19) \quad \pi_{\mathcal{P}} \mathbf{u} = \pi_C \mathbf{u} + \pi_R \mathbf{u}.$$

The next lemma is the counterpart of [Lemma 5.1](#) in the present linear elasticity setting; its very similar proof is omitted for brevity.

Lemma 5.2. *The following facts hold true:*

- The operators π_R and π_C are the ‘‘orthogonal projections’’ from $\mathcal{W}(E)$ onto $\mathcal{W}_R(E)$ and $\mathcal{W}_C(E)$, respectively, in the sense of the bilinear form $a^E(\cdot, \cdot)$:

$$\forall \mathbf{r} \in \mathcal{W}_R(E), \mathbf{c} \in \mathcal{W}_C(E), \mathbf{v} \in \mathcal{W}(E), \quad a^E(\mathbf{r}, \mathbf{v}) = 0 \text{ and } a^E(\mathbf{c}, \mathbf{v} - \pi_C \mathbf{v}) = 0.$$

- The following decomposition of $a^E(\cdot, \cdot)$ holds:

$$(5.20) \quad \forall \mathbf{u}, \mathbf{v} \in \mathcal{W}(E), \quad a^E(\mathbf{u}, \mathbf{v}) = a^E(\pi_C \mathbf{u}, \pi_C \mathbf{v}) + a^E(\mathbf{u} - \pi_{\mathcal{P}} \mathbf{u}, \mathbf{v} - \pi_{\mathcal{P}} \mathbf{v}).$$

Let us now discuss the calculation of the local stiffness matrix

$$K^E \in \mathbb{R}^{2n^E \times 2n^E}, \quad K_{ij}^E = a(\zeta_i, \zeta_j), \quad i, j = 1, \dots, 2n^E.$$

Like in [Section 5.2.2](#), [\(5.20\)](#) induces a convenient decomposition K^E into two blocks:

$$K^E = P^E + S^E, \quad P^E, S^E \in \mathbb{R}^{2n^E \times 2n^E}.$$

Here,

- The first term in the right-hand side of [\(5.20\)](#) gives rise to the matrix $P^E \in \mathbb{R}^{2n^E \times 2n^E}$ defined by:

$$\forall i, j = 1, \dots, 2n^E, \quad P_{ij}^E = a^E(\pi_C \zeta_i, \pi_C \zeta_j),$$

which roughly speaking corresponds to the block of K^E over $\mathcal{P}(E)$. The entries of P^E can be calculated exactly, since the coefficients of the projection $\pi_C \mathbf{u}$ of a vector field $\mathbf{u} \in \mathcal{W}(E)$ in the basis $\{\mathbf{c}_\alpha\}_{\alpha=1,2,3}$ can be calculated from the values of \mathbf{u} at the vertices of E , see [Appendix D.2](#).

- The second term in the right-hand side of [\(5.20\)](#) appraises the behavior of the bilinear form $a^E(\cdot, \cdot)$ over those functions in $\mathcal{W}(E)$ that are not affine. Again, this second block is replaced by a suitable quantity which can be calculated just from the values of \mathbf{u} and \mathbf{v} at the vertices of E .

As the result of these considerations, we replaced the local bilinear form $a^E(\cdot, \cdot)$ with

$$\tilde{a}^E(\mathbf{u}, \mathbf{v}) = a^E(\pi_C \mathbf{u}, \pi_C \mathbf{v}) + \tilde{s}^E(\mathbf{u} - \pi_P \mathbf{u}, \mathbf{v} - \pi_P \mathbf{v}), \quad \text{where } \tilde{s}^E(\mathbf{u}, \mathbf{v}) = \alpha^E \sum_{i=1}^n \mathbf{u}(\mathbf{q}_i) \cdot \mathbf{v}(\mathbf{q}_i),$$

and the coefficient α^E depends on the element E so that \tilde{s}^E scale as a^E upon refinement of the mesh \mathcal{T} . The implementation details of this methodology are presented in [Appendix D.2](#).

6. SENSITIVITY OF A SHAPE FUNCTIONAL WITH RESPECT TO SEED POINTS AND VOLUME FRACTIONS

In this section, we return to our main purpose of solving the shape and topology optimization problem

$$(P) \quad \min_{\Omega} J(\Omega) \quad \text{s.t.} \quad \mathbf{G}(\Omega) = 0,$$

where $J(\Omega)$ and $\mathbf{G}(\Omega)$ depend on Ω via the solution u_Ω to a boundary value problem such as [\(Cond\)](#) or [\(Elas\)](#). We describe how the shape derivative of a general objective function $J(\Omega)$ (or similarly, that of a general constraint function $\mathbf{G}(\Omega)$) can be expressed in terms of the defining variables of a diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ representing Ω in the sense that [\(3.6\)](#) holds. In this perspective, in [Section 6.1](#), we first deal with the discretization of $J(\Omega)$ into a function $J(\mathbf{q})$ of the vertices $\mathbf{q} \in \mathbb{R}^{dM}$ of the diagram, and we expose how the theoretical expression of the shape derivative $J'(\Omega)(\boldsymbol{\theta})$ allows to calculate the gradient $\nabla_{\mathbf{q}} J(\mathbf{q})$ of this discrete function. As we have mentioned, $J(\mathbf{q})$ (resp. $\mathbf{G}(\mathbf{q})$) in turn induces a function $\tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ (resp. $\tilde{\mathbf{G}}(\mathbf{s}, \boldsymbol{\nu})$) of the seed points $\mathbf{s} \in \mathbb{R}^{dN}$ and cell measures $\boldsymbol{\nu} \in \mathbb{R}^N$ of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$. This gives rise to our discrete shape and topology optimization problem:

$$(P\text{-disc}) \quad \min_{(\mathbf{s}, \boldsymbol{\nu})} \tilde{J}(\mathbf{s}, \boldsymbol{\nu}) \quad \text{s.t.} \quad \tilde{\mathbf{G}}(\mathbf{s}, \boldsymbol{\nu}) = 0.$$

We explain in [Section 6.2](#) how the sensitivities $\nabla_{\mathbf{s}} \tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ and $\nabla_{\boldsymbol{\nu}} \tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ of $J(\Omega)$ with respect to the parameters $(\mathbf{s}, \boldsymbol{\nu})$ of the diagram for Ω can be inferred from the gradient $\nabla_{\mathbf{q}} J(\mathbf{q})$. In [Section 6.3](#), we recall the Hilbertian trick, which allows among other things to smooth these sensitivities in a consistent manner. This procedure is also a key algorithmic component of the constrained optimization algorithm used to tackle [\(P\)](#), that we finally describe in [Section 6.4](#).

Notation. In this section, the shape $\Omega \subset \mathbb{R}^d$ is described by a modified diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ of the form [\(3.6\)](#) and [\(3.7\)](#). We consider a generic function $J(\Omega)$ of Ω , standing for either the objective or a constraint in [\(P\)](#). To simplify the presentation and without loss of generality, we shall sometimes assume that the space dimension d equals 2, and that $J(\Omega)$ is of the form

$$(6.1) \quad J(\Omega) = \int_{\Omega} j(u_\Omega) \, dx,$$

where $j : \mathbb{R} \rightarrow \mathbb{R}$ is a smooth function, satisfying adequate growth conditions, and u_Ω is the solution to the conductivity equation [\(Cond\)](#). As we have hinted at, $J(\Omega)$ induces several discrete versions, one depending

on the vertices \mathbf{q} of the diagram, that we shall denote by $J(\mathbf{q})$ with a small abuse of notation, and in turn, another function $\tilde{J}(\mathbf{s}, \boldsymbol{\nu})$, depending on the seed points \mathbf{s} and the cell measures $\boldsymbol{\nu}$.

6.1. Discretization and differentiation of $J(\Omega)$ in terms of the vertices of a diagram for Ω

In this section, we discuss the discretization of the shape functional $J(\Omega)$ as a function $J(\mathbf{q})$ of the vertices $\mathbf{q} \in \mathbb{R}^{dM}$ of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ representing Ω as (3.6) and the calculation of the gradient $\nabla_{\mathbf{q}} J(\mathbf{q})$.

As we have mentioned, in our applications, $J(\Omega)$ depends on Ω via the solution u_Ω to a boundary value problem posed on Ω , see e.g. (6.1). The latter is solved along the lines of Section 5 by applying the Virtual Element Method on the polygonal mesh \mathcal{T} resulting from the discretization of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$, see Section 4. The discrete solution is a function of the vertices \mathbf{q} of the mesh \mathcal{T} , and it gives rise in turn to a discrete objective function $J(\mathbf{q})$ once the formula (6.1) is discretized by suitable quadrature formulas. In the present work, we rely on very basic expressions, which are exemplified in the forthcoming discussion. More elaborate rules are available in the context of general polygonal elements [35, 120]; for brevity, we do not emphasize on this classical, albeit technical issue.

The calculation of the gradient $\nabla_{\mathbf{q}} J(\mathbf{q})$ is a slightly more delicate issue. It falls under the “discretize-then-optimize” setting, in which the derivatives of the discrete quantities of interest are used [105]; several methods can be used to achieve this goal. One convenient practice relies on automatic differentiation, see e.g. [90] for an introduction in the shape optimization context. In a nutshell, the gradient of the mapping $\mathbf{q} \mapsto J(\mathbf{q})$ can be calculated concurrently with its evaluation by leveraging adequate data structures and by implementing all the constituent operations of the evaluation pipeline (including the assembly and inversion of the stiffness matrix involved in calculations by the Virtual Element Method) together with their derivatives. This elegant framework is unfortunately very stringent, and it is not always compatible with realistic applications where the mechanical solver is used in a black-box fashion.

When automatic differentiation of $J(\mathbf{q})$ is not available, the computation of $\nabla_{\mathbf{q}} J(\mathbf{q})$ can be realized “by hand”. In fact, the derivatives of the aforementioned building blocks of the evaluation of $\mathbf{q} \mapsto J(\mathbf{q})$ can be calculated explicitly, in principle. Even though this strategy could be adopted in our context, it is still very intrusive as it requires a complete knowledge of the mechanical solver, notably.

For this reason, we consider yet another option, borrowing from the “optimize-then-discretize” paradigm in optimal design, which mainly relies on the continuous formula for the shape derivative $J'(\Omega)(\boldsymbol{\theta})$. More precisely, we leverage an elegant remark from [19]: the sensitivity $\frac{\partial J}{\partial \mathbf{q}_i}(\mathbf{q})(\mathbf{h}_i)$ of $J(\mathbf{q})$ to a perturbation of the i^{th} vertex \mathbf{q}_i in the direction $\mathbf{h}_i \in \mathbb{R}^d$ can be approximated by evaluating (a discretization of) the volume form of the continuous shape derivative $\boldsymbol{\theta} \mapsto J'(\Omega)(\boldsymbol{\theta})$ with the particular deformation field $\boldsymbol{\theta} = \varphi_i \mathbf{h}_i$, where φ_i is the unique function in the discrete space $\mathcal{W}_{\mathcal{T}}$ given by (5.6) taking the value 1 at vertex i and 0 elsewhere, see (5.7):

$$\frac{\partial J}{\partial \mathbf{q}_i}(\mathbf{q})(\mathbf{h}_i) = J'(\Omega)(\varphi_i \mathbf{h}_i), \text{ where } \varphi_i \in \mathcal{W}_{\mathcal{T}} \text{ is defined by: } \varphi_i(\mathbf{q}_j) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

In a nutshell, this practice is legitimated by the fact that the theoretical formula for the volume form of $J'(\Omega)(\boldsymbol{\theta})$ can be obtained by the exact same adjoint-based trail as that for the discrete derivative $\frac{\partial J}{\partial \mathbf{q}_i}(\mathbf{q})(\mathbf{h}_i)$. We refer to e.g. [65] and §2.9 in [67] for related discussions about the (non) commutation of discretization and differentiation in optimal design.

In order to describe this possibility more precisely, let us assume that the shape derivative $J'(\Omega)(\boldsymbol{\theta})$ is available under the following volume form (2.9):

$$J'(\Omega)(\boldsymbol{\theta}) = \int_{\Omega} (\mathbf{t}_{\Omega} \cdot \boldsymbol{\theta} + S_{\Omega} : \nabla \boldsymbol{\theta}) \, \mathrm{d}\mathbf{x}, \text{ for some } \mathbf{t}_{\Omega} : \Omega \rightarrow \mathbb{R}^d \text{ and } S_{\Omega} : \Omega \rightarrow \mathbb{R}^{d \times d},$$

and let us suppose for a moment that the fields \mathbf{t}_{Ω} and S_{Ω} are given approximate, discrete counterparts $\mathbf{t}_{\mathcal{T}}$ and $S_{\mathcal{T}}$ that are constant in restriction to each element $E \in \mathcal{T}$:

$$(6.2) \quad J'(\Omega)(\boldsymbol{\theta}) \approx J'_{\mathcal{T}}(\boldsymbol{\theta}) := \int_{\Omega} (\mathbf{t}_{\mathcal{T}} \cdot \boldsymbol{\theta} + S_{\mathcal{T}} : \nabla \boldsymbol{\theta}) \, \mathrm{d}\mathbf{x}.$$

The gradient of $J(\mathbf{q})$ with respect to the coordinates of the i^{th} vertex of the mesh is then approximated as:

$$\nabla_{\mathbf{q}_i} J(\mathbf{q}) \approx \begin{pmatrix} J'_{\mathcal{T}}(\boldsymbol{\theta}_i^1) \\ \vdots \\ J'_{\mathcal{T}}(\boldsymbol{\theta}_i^d) \end{pmatrix},$$

where for $k = 1, \dots, d$, $\boldsymbol{\theta}_i^k \in \mathcal{W}_{\mathcal{T}}^d$ is the unique vector-valued discrete function such that:

$$(6.3) \quad \boldsymbol{\theta}_i^k(\mathbf{q}_j) = \mathbf{e}_k \text{ if } i = j \text{ and } \mathbf{0} \text{ otherwise.}$$

The aforementioned approximations of \mathbf{t}_{Ω} , S_{Ω} by $\mathbf{t}_{\mathcal{T}}$ and $S_{\mathcal{T}}$ can be realized in various ways. For instance one could use the simple formulas:

$$\mathbf{t}_{\mathcal{T}}|_E = \frac{1}{n^E} \sum_{j=1}^{n^E} \mathbf{t}_{\Omega}(\mathbf{q}_j^E), \text{ and } S_{\mathcal{T}}|_E = \frac{1}{n^E} \sum_{j=1}^{n^E} S_{\Omega}(\mathbf{q}_j^E),$$

where we have used the notations of [Section 5.1](#).

When S_{Ω} involves the gradient of the state function u_{Ω} , say for instance $S_{\Omega} = \gamma \nabla u_{\Omega} \otimes \nabla u_{\Omega}$, one could also make the approximation:

$$S_{\mathcal{T}}|_E = \gamma_E \nabla(\pi_{\mathcal{P}} u) \otimes \nabla(\pi_{\mathcal{P}} u), \text{ where } \gamma_E := \frac{1}{n^E} \sum_{j=1}^{n^E} \gamma(\mathbf{q}_j^E),$$

and $\pi_{\mathcal{P}}$ is the projector over first-order polynomial functions in $\mathcal{P}(E)$, defined in [\(5.19\)](#).

In any event, the integrals featured in the quantities $J'_{\mathcal{T}}(\boldsymbol{\theta}_i^k)$ can be naturally decomposed as sums of integrals running over the elements E of the mesh \mathcal{T} :

$$J'_{\mathcal{T}}(\boldsymbol{\theta}_i^k) = \sum_{E \in \mathcal{T}} \left(\mathbf{t}_{\mathcal{T}}|_E \cdot \mathbf{t}_E(\boldsymbol{\theta}_i^k) + S_{\mathcal{T}}|_E : S_E(\boldsymbol{\theta}_i^k) \right),$$

where the vectors $\mathbf{t}_E(\boldsymbol{\theta}_i^k)$ and matrices $S_E(\boldsymbol{\theta}_i^k)$ are defined by:

$$\mathbf{t}_E(\boldsymbol{\theta}_i^k) = \int_E \boldsymbol{\theta}_i^k \, d\mathbf{x} \text{ and of the matrices } S_E(\boldsymbol{\theta}_i^k) = \int_E \nabla \boldsymbol{\theta}_i^k \, d\mathbf{x}.$$

These basic ingredients can be calculated or approximated in explicit terms of the physical characteristics of the element E . For instance, in the case where the space dimension d equals 2, we approximate $\mathbf{t}_E(\boldsymbol{\theta}_i^1)$ and $\mathbf{t}_E(\boldsymbol{\theta}_i^2)$ as:

$$\mathbf{t}_E(\boldsymbol{\theta}_i^1) \approx \frac{1}{n^E} |E| \begin{pmatrix} 1 \\ 0 \end{pmatrix} \text{ and } \mathbf{t}_E(\boldsymbol{\theta}_i^2) \approx \frac{1}{n^E} |E| \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

As far as the matrices $S_E(\boldsymbol{\theta}_i^1)$, $S_E(\boldsymbol{\theta}_i^2)$ are concerned, similar integrations by parts to those used in [Appendix D.1](#) yield:

$$S_E(\boldsymbol{\theta}_i^1) = \frac{1}{2} \begin{pmatrix} |\widehat{\mathbf{e}}_j|(\mathbf{n}_{\widehat{\mathbf{e}}_j})_1 & |\widehat{\mathbf{e}}_j|(\mathbf{n}_{\widehat{\mathbf{e}}_j})_2 \\ 0 & 0 \end{pmatrix}, \text{ and } S_E(\boldsymbol{\theta}_i^2) = \frac{1}{2} \begin{pmatrix} 0 & 0 \\ |\widehat{\mathbf{e}}_j|(\mathbf{n}_{\widehat{\mathbf{e}}_j})_1 & |\widehat{\mathbf{e}}_j|(\mathbf{n}_{\widehat{\mathbf{e}}_j})_2 \end{pmatrix}.$$

Remark 6.1. *In principle, the continuous shape derivative $J'(\Omega)(\boldsymbol{\theta})$ depends only on the values of the deformation $\boldsymbol{\theta}$ on the boundary $\partial\Omega$, see [Section 2.3](#) about this general feature. On the contrary, it is not exactly true that the approximation $J'_{\mathcal{T}}(\boldsymbol{\theta}_i^k)$ in [\(6.2\)](#) vanishes when the deformation $\boldsymbol{\theta}_i^k$ is associated to an internal vertex \mathbf{q}_i . This is a manifestation of the lack of commutation between the operations of discretization and differentiation in optimal design, see again [\[65\]](#) and §2.9 in [\[67\]](#). However, we expect (and indeed verify) that if the approximation [\(6.2\)](#) is “good enough”, the entries of $\nabla_{\mathbf{q}} J(\mathbf{q})$ attached to internal vertices should be “very small” when compared to those related to boundary vertices. For this reason, it is enough to compute the quantities $J'_{\mathcal{T}}(\boldsymbol{\theta}_j^k)$ for the indices $j = 1, \dots, M$ of the boundary vertices of Ω . Such issues, related to the processing and use of the volume form of the shape derivative $J'(\Omega)(\boldsymbol{\theta})$ are by now fairly well documented in the literature, see e.g. [\[54, 64, 71\]](#).*

6.2. Calculation of the derivative of a functional with respect to the positions of the seeds and the cell measures

We have seen in [Section 6.1](#) how a general function of the domain $J(\Omega)$ can be discretized into a function $J(\mathbf{q})$ depending on the positions $\mathbf{q} \in \mathbb{R}^{dM}$ of the vertices of the mesh \mathcal{T} , and how to calculate the gradient $\nabla_{\mathbf{q}}J(\mathbf{q})$. In turn, \mathbf{q} depends on the seed points \mathbf{s} and the weights $\boldsymbol{\psi}$ of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ of Ω , and, again, on the seed points \mathbf{s} and the cell measures $\boldsymbol{\nu}$, which are the design variables of our discrete version of the shape and topology optimization problem (P), as explained in [Section 3.3](#). This raises the need to express the derivative of a general function $J(\mathbf{q})$ of the vertices \mathbf{q} of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ in terms of \mathbf{s} and $\boldsymbol{\nu}$.

To formulate this issue rigorously, let us write the relation between the seeds points $\mathbf{s} \in \mathbb{R}^{dN}$, the weights $\boldsymbol{\psi} \in \mathbb{R}^N$ and the vertices \mathbf{q} of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ under the abstract form:

$$(6.4) \quad \mathbf{q} = \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})),$$

where $\mathbf{Q} : \mathbb{R}_s^{dN} \times \mathbb{R}_\psi^N \rightarrow \mathbb{R}^{dM}$ produces the collection of vertices of the mesh \mathcal{T} induced by $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$, and, as in [Section 3.2](#), $\boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})$ is the unique weight vector such that each cell $V_i(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))$ has measure ν_i , $i = 1, \dots, N$. We then wish to calculate the derivatives of the functional $\tilde{J} : \mathbb{R}_s^{dN} \times \mathbb{R}_\nu^N \rightarrow \mathbb{R}$ defined by:

$$(6.5) \quad \tilde{J}(\mathbf{s}, \boldsymbol{\nu}) := J(\mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))).$$

This is the purpose of the next result.

Proposition 6.1. *Let $J : \mathbb{R}^{dM} \rightarrow \mathbb{R}$ be a differentiable function. Then the composite function $\tilde{J} : \mathbb{R}_s^{dN} \times \mathbb{R}_\nu^N \rightarrow \mathbb{R}$ in (6.5) is differentiable almost everywhere, and its derivatives read:*

$$(6.6) \quad \nabla_s \tilde{J}(\mathbf{s}, \boldsymbol{\nu}) = \left[\nabla_s \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \nabla_{\mathbf{q}} J(\mathbf{q}) + \left[\nabla_s \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \mathbf{p}(\mathbf{s}, \boldsymbol{\nu}),$$

and

$$(6.7) \quad \nabla_{\boldsymbol{\nu}} \tilde{J}(\mathbf{s}, \boldsymbol{\nu}) = -\mathbf{p}(\mathbf{s}, \boldsymbol{\nu}).$$

Here, the function $\mathbf{F} : \mathbb{R}_s^{dN} \times \mathbb{R}_\nu^N \times \mathbb{R}_\psi^N \rightarrow \mathbb{R}^N$ is the derivative $\nabla_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$, see (3.14); the adjoint vector $\mathbf{p}(\mathbf{s}, \boldsymbol{\nu}) \in \mathbb{R}^N$ is defined as the solution to the $N \times N$ system:

$$(6.8) \quad \left[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \mathbf{p}(\mathbf{s}, \boldsymbol{\nu}) = - \left[\nabla_{\boldsymbol{\psi}} \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \nabla_{\mathbf{q}} J(\mathbf{q}).$$

In these formulas,

- The gradient $\nabla_{\mathbf{q}} J(\mathbf{q})$ is a vector in \mathbb{R}^{dM} , whose expression is assumed to be known.
- The matrices $[\nabla_s \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))]$ and $[\nabla_{\boldsymbol{\psi}} \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))]$ have respective sizes $dM \times dN$ and $dM \times N$; their explicit expressions (in 2d) are the subject of [Appendix A](#).
- The matrix $[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))]^T$ has size $N \times N$, and its calculation (in 2d) is provided in [Appendix B](#).

Proof. At first, [Proposition A.1](#) below states that the mapping \mathbf{Q} is differentiable at any point $(\mathbf{s}, \boldsymbol{\psi}) \in \mathbb{R}^{dN} \times \mathbb{R}^N$ satisfying the assumptions of [Theorem 3.1](#). These points form an open subset of $\mathbb{R}^{dN} \times \mathbb{R}^N$ whose complement has null Lebesgue measure. Moreover, for any such point $(\mathbf{s}, \boldsymbol{\nu}) \in \mathbb{R}^{dN} \times \mathbb{R}^N$, we have seen that $\boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})$ is the unique solution to the equation (3.14). By the implicit function theorem and the regularity of \mathbf{F} expressed in [Theorem 3.1](#), it follows that $\boldsymbol{\psi}^*$ is differentiable at such a point $(\mathbf{s}, \boldsymbol{\nu})$, and by composition, \tilde{J} is differentiable at a.e. point $(\mathbf{s}, \boldsymbol{\nu})$ in $\mathbb{R}^{dN} \times \mathbb{R}^N$.

Let us now prove (6.6). At any point $(\mathbf{s}, \boldsymbol{\nu})$ where \tilde{J} is differentiable, the chain rule shows that:

$$\forall \mathbf{h} \in \mathbb{R}^{dN}, \quad \frac{\partial \tilde{J}}{\partial \mathbf{s}}(\mathbf{s}, \boldsymbol{\nu})(\mathbf{h}) = \frac{\partial J}{\partial \mathbf{q}}(\mathbf{q}) \left(\frac{\partial \mathbf{Q}}{\partial \mathbf{s}}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))(\mathbf{h}) + \frac{\partial \mathbf{Q}}{\partial \boldsymbol{\psi}}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \left(\frac{\partial \boldsymbol{\psi}^*}{\partial \mathbf{s}}(\mathbf{s}, \boldsymbol{\nu})(\mathbf{h}) \right) \right),$$

where we have set $\mathbf{q} := \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))$. This rewrites, in terms of gradients and matrices:

$$(6.9) \quad \nabla_s \tilde{J}(\mathbf{s}, \boldsymbol{\nu}) = \left(\left[\nabla_s \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T + \left[\nabla_s \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}) \right]^T \left[\nabla_{\boldsymbol{\psi}} \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \right) \nabla_{\mathbf{q}} J(\mathbf{q}).$$

All the terms are explicit in this formula, except for the matrix $[\nabla_s \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})]$. The calculation of the latter is difficult, since the function $\mathbf{s} \mapsto \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})$ is only known implicitly, through the relation (3.14). To overcome

this difficulty, we use the classical adjoint technique for the differentiation of a quantity depending on the solution to a parametrized implicit equation, see again [81, 106]. Differentiation with respect to \mathbf{s} in (3.14) yields:

$$(6.10) \quad \underbrace{\left[\nabla_{\mathbf{s}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]}_{N \times dN \text{ matrix}} + \underbrace{\left[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]}_{N \times N \text{ matrix}} \underbrace{\left[\nabla_{\mathbf{s}} \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}) \right]}_{N \times dN \text{ matrix}} = 0.$$

We then introduce the adjoint state $\mathbf{p}(\mathbf{s}, \boldsymbol{\nu}) \in \mathbb{R}^N$, defined as the solution to the $N \times N$ linear system:

$$\left[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \mathbf{p}(\mathbf{s}, \boldsymbol{\nu}) = - \left[\nabla_{\boldsymbol{\psi}} \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \nabla_{\mathbf{q}} J(\mathbf{q}).$$

Injecting this identity into (6.9), we obtain:

$$\nabla_{\mathbf{s}} \tilde{J}(\mathbf{s}, \boldsymbol{\nu}) = \left[\nabla_{\mathbf{s}} \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \nabla_{\mathbf{q}} J(\mathbf{q}) - \left[\nabla_{\mathbf{s}} \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}) \right]^T \left[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \mathbf{p}(\mathbf{s}, \boldsymbol{\nu}).$$

Eventually, using (6.10) to transform the last expression in the above right-hand side, we obtain:

$$\nabla_{\mathbf{s}} \tilde{J}(\mathbf{s}, \boldsymbol{\nu}) = \left[\nabla_{\mathbf{s}} \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \nabla_{\mathbf{q}} J(\mathbf{q}) + \left[\nabla_{\mathbf{s}} \mathbf{F}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \mathbf{p}(\mathbf{s}, \boldsymbol{\nu}),$$

which is the desired formula.

Let us then consider the second formula (6.7). Again, a simple use of the chain rule yields:

$$(6.11) \quad \begin{aligned} \nabla_{\boldsymbol{\nu}} \tilde{J}(\mathbf{s}, \boldsymbol{\nu}) &= \nabla_{\boldsymbol{\nu}} \left(J(\mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))) \right) \\ &= \left[\nabla_{\boldsymbol{\nu}} \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}) \right]^T \left[\nabla_{\boldsymbol{\psi}} \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \nabla_{\mathbf{q}} J(\mathbf{q}), \end{aligned}$$

where $\mathbf{q} := \mathbf{Q}(\mathbf{s}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}))$. As in the proof of (6.6), we rely on the adjoint method to eliminate the difficult term $\left[\nabla_{\boldsymbol{\nu}} \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}) \right]^T$ from this expression. Recalling the defining system (6.8) for $\mathbf{p}(\mathbf{s}, \boldsymbol{\nu}) \in \mathbb{R}^N$, we obtain immediately:

$$\begin{aligned} \nabla_{\boldsymbol{\nu}} \tilde{J}(\mathbf{s}, \boldsymbol{\nu}) &= - \left[\nabla_{\boldsymbol{\nu}} \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu}) \right]^T \left[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})) \right]^T \mathbf{p}(\mathbf{s}, \boldsymbol{\nu}) \\ &= -\mathbf{p}(\mathbf{s}, \boldsymbol{\nu}), \end{aligned}$$

which is the desired expression. \square

Remark 6.2. When the optimized shape Ω is only one phase within a large computational domain D and is represented as a subcollection of the cells of a “true” Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ of D , of the form (3.3) and (3.4), similar considerations show that formula (6.7) reads instead:

$$\nabla_{\boldsymbol{\nu}} \tilde{J}(\mathbf{s}, \boldsymbol{\nu}) = - \left(\mathbf{p}(\mathbf{s}, \boldsymbol{\nu}) - \frac{1}{N} \sum_{i=1}^N p_i(\mathbf{s}, \boldsymbol{\nu}) \right).$$

Remark 6.3. The geometric calculations needed in the present section for the expressions of the matrices $\left[\nabla_{\mathbf{s}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) \right]$, $\left[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) \right]$, $\left[\nabla_{\mathbf{s}} \mathbf{Q}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) \right]$ and $\left[\nabla_{\boldsymbol{\psi}} \mathbf{Q}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) \right]$ are provided in *Appendices A* and *B*. They need only be implemented once and for all. In particular, this implementation is independent of the particular objective function.

6.3. The Hilbertian method

We have described in the previous *Sections 6.1* and *6.2* how to calculate the sensitivities $\nabla_{\mathbf{s}} \tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ and $\nabla_{\boldsymbol{\nu}} \tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ of a function J , depending on the domain Ω via the seeds $\mathbf{s} \in \mathbb{R}^{dN}$ and the cell measures $\boldsymbol{\nu} \in \mathbb{R}^N$ of a diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ of the form (3.6). Unfortunately, a direct use of these derivatives in the solution of our discrete shape and topology optimization problem (P-disc) often proves awkward in practice: on the one hand, they may show very irregular variations in space (i.e. they may have very different components along the indices $i = 1, \dots, N$ of neighboring cells), thus causing numerical instabilities. On the other hand, in multiple applications, it is desirable to handle sensitivities that take into account certain requirements, for instance that some of their components associated to cells located in non optimizable regions of space should vanish. In this section, we explain how the so-called Hilbertian method allows to achieve both purposes with

rigorous guarantees that the resulting modified sensitivities retain descent properties. The application in shape optimization of this classical procedure in the context of gradient flows is exposed in e.g. [7, 30, 46, 90].

6.3.1. The general strategy

Let us describe the use of the Hilbertian method in the illustrative perspective of regularizing the gradient $\nabla_{\mathbf{s}}\tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ of a discrete objective function $\tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ at a given point $(\mathbf{s}, \boldsymbol{\nu}) \in \mathbb{R}^{dN} \times \mathbb{R}^N$.

To this end, let us recall that $\nabla_{\mathbf{s}}\tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ is obtained by identifying the derivative $\mathbf{h} \mapsto \frac{\partial \tilde{J}}{\partial \mathbf{s}}(\mathbf{s}, \boldsymbol{\nu})(\mathbf{h})$ with an element in the space \mathbb{R}^{dN} via the following relation:

$$\forall \mathbf{h} \in \mathbb{R}^{dN}, \quad \frac{\partial \tilde{J}}{\partial \mathbf{s}}(\mathbf{s}, \boldsymbol{\nu})(\mathbf{h}) = \left\langle \nabla_{\mathbf{s}}\tilde{J}(\mathbf{s}, \boldsymbol{\nu}), \mathbf{h} \right\rangle.$$

The main idea of the Hilbertian method is to introduce a subspace V of the space \mathbb{R}^{dN} of seed points, equipped with an inner product $a_{\mathbf{s}}(\cdot, \cdot)$ different from the canonical one $\langle \cdot, \cdot \rangle$, and to search for the gradient $\mathbf{h}_J \in V$ of $\tilde{J}(\cdot, \boldsymbol{\nu})$ with respect to this inner product. In other terms, we solve the following variational problem:

$$(6.12) \quad \text{Search for } \mathbf{h}_J \in V \text{ s.t. for all } \mathbf{h} \in V, \quad a_{\mathbf{s}}(\mathbf{h}_J, \mathbf{h}) = \frac{\partial \tilde{J}}{\partial \mathbf{s}}(\mathbf{s}, \boldsymbol{\nu})(\mathbf{h}).$$

Depending on the nature of the space V and its inner product $a_{\mathbf{s}}(\cdot, \cdot)$, the gradient \mathbf{h}_J may for instance be more regular than its counterpart $\nabla_{\mathbf{s}}\tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ associated to the canonical inner product, some of its components may vanish, etc.

Our choices of V and $a_{\mathbf{s}}(\cdot, \cdot)$ in the present context rely on the smoothing effect of the discrete Laplace operator acting on mass–spring networks, see for instance [111] or [92] for reviews about the role of such discrete Laplace operators in the field of computer graphics. We take $V = \mathbb{R}^{dN}$ and search for the regularized gradient of $\tilde{J}(\cdot, \boldsymbol{\nu})$ at \mathbf{s} as the solution $\mathbf{h}_J := \{\mathbf{h}_{J,i}\}_{i=1,\dots,N} \in \mathbb{R}^{dN}$ to the following minimization problem:

$$(6.13) \quad \min_{\mathbf{h} \in \mathbb{R}^{dN}} E(\mathbf{h}), \text{ where } E(\mathbf{h}) := \frac{\alpha^2}{2} \sum_{\substack{(i,j) \in \mathcal{E} \\ j > i}} |\mathbf{h}_i - \mathbf{h}_j|^2 + \frac{1}{2} \sum_{i=1}^N |\mathbf{h}_i|^2 - \sum_{i=1}^N \left\langle \nabla_{\mathbf{s}_i}\tilde{J}(\mathbf{s}, \boldsymbol{\nu}), \mathbf{h}_i \right\rangle,$$

where α is a user-defined parameter encoding a regularization length scale. Intuitively, the solution \mathbf{h}_J to (6.13) is a version of $\nabla_{\mathbf{s}}\tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ whose spatial variations are averaged over a region with radius α . The strictly convex optimization problem (6.13) has a unique solution, which is characterized by the following first-order optimality condition:

$$\text{For } i = 1, \dots, N, \quad \alpha^2 \sum_{j \in \mathcal{N}_i} (\mathbf{h}_i - \mathbf{h}_j) + \mathbf{h}_i = \nabla_{\mathbf{s}_i}\tilde{J}(\mathbf{s}, \boldsymbol{\nu}).$$

This rewrites as a $2N \times 2N$ matrix system:

$$(\alpha^2 A_{\mathbf{s}} + \mathbf{I})\mathbf{h} = \nabla_{\mathbf{s}}\tilde{J}(\mathbf{s}, \boldsymbol{\nu}),$$

where the matrix $A_{\mathbf{s}} \in \mathbb{R}^{2N \times 2N}$ is given by the entries:

$$(6.14) \quad A_{\mathbf{s}, 2(i-1)+1, 2(j-1)+1} = A_{\mathbf{s}, 2(i-1)+2, 2(j-1)+2} = \begin{cases} \#\mathcal{N}_i & \text{if } j = i, \\ -1 & \text{if } j \in \mathcal{N}_i, \\ 0 & \text{otherwise,} \end{cases} \quad \text{and} \\ A_{\mathbf{s}, 2(i-1)+1, 2(j-1)+2} = A_{\mathbf{s}, 2(i-1)+2, 2(j-1)+1} = 0, \quad i, j = 1, \dots, N.$$

The inner product $a_{\mathbf{s}}(\cdot, \cdot)$ on $V = \mathbb{R}^{dN}$ associated to this strategy is:

$$(6.15) \quad \forall \mathbf{h}^1, \mathbf{h}^2 \in V, \quad a_{\mathbf{s}}(\mathbf{h}^1, \mathbf{h}^2) = \left\langle (\alpha^2 A_{\mathbf{s}} + \mathbf{I})\mathbf{h}^1, \mathbf{h}^2 \right\rangle.$$

Remark 6.4. A similar strategy can be applied to impose smooth spatial variations of the gradient $\nabla_{\boldsymbol{\nu}}\tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ of \tilde{J} with respect to the measures of cells. One then relies on the choice $V = \mathbb{R}^N$, equipped with the inner

product:

$$(6.16) \quad a_{\nu}(\boldsymbol{\nu}^1, \boldsymbol{\nu}^2) = \left\langle A_{\nu} \boldsymbol{\nu}^1, \boldsymbol{\nu}^2 \right\rangle, \text{ where } A_{\nu, ij} = \begin{cases} \#\mathcal{N}_i & \text{if } j = i, \\ -1 & \text{if } j \in \mathcal{N}_i, \\ 0 & \text{otherwise,} \end{cases}$$

6.3.2. Imposing that seeds stay inside the computational domain

The Hilbertian method naturally allows to enforce certain constraints on the regularized sensitivity \mathbf{h}_J of J with respect to the seed points \mathbf{s} in (6.12). For instance \mathbf{h}_J can be imposed to have null components over the indices i inside a given subset $I \subset \{1, \dots, N\}$ of the seed points by simply taking the following Hilbert space V in (6.12):

$$V := \{\mathbf{h} = (\mathbf{h}_1, \dots, \mathbf{h}_N) \in \mathbb{R}^{dN} \text{ s.t. } \mathbf{h}_i = \mathbf{0} \text{ for } i \in I\}.$$

This is useful for a variety of purposes, for instance to impose that certain cells should be altered as little as possible in the course of the optimization process, see e.g. the examples of Sections 7.3 to 7.6 where this operation is applied to the seed points and measures of cells bearing the Dirichlet or inhomogeneous Neumann boundary conditions of the elasticity problem at stake.

In this section, we exemplify how the seed points \mathbf{s} of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ featured in Algorithm 1 can be constrained to stay inside the computational domain D during the optimization process thanks to this practice, see also Remark 3.1 about this point.

Let $(\mathbf{s}, \boldsymbol{\nu}) \in \mathbb{R}^{dN} \times \mathbb{R}^N$ be a given stage of the process of Algorithm 1, and let $\boldsymbol{\psi} = \boldsymbol{\psi}^*(\mathbf{s}, \boldsymbol{\nu})$. To simplify the presentation, we assume that the seed points of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ are numbered in such a way that the first $B \leq N$ ones are those adjacent to the external boundary ∂D . For notational simplicity and without loss of generality, we also assume that for each $i = 1, \dots, B$ at most one edge of each cell $V_i(\mathbf{s}, \boldsymbol{\psi})$ belongs to the external boundary ∂D , and we denote by \mathbf{n}_i the unit normal vector to this edge, pointing outward D . We then search for a gradient \mathbf{h} of \tilde{J} at $(\mathbf{s}, \boldsymbol{\nu})$ with the property that, for each index $i = 1, \dots, B$, the normal component $\mathbf{h}_i \cdot \mathbf{n}_i$ of the displacement \mathbf{h}_i should vanish:

$$(6.17) \quad \forall i = 1, \dots, B, \quad \mathbf{h}_i \cdot \mathbf{n}_i = 0.$$

One possibility to enforce this requirement is to proceed by exact penalization, i.e. to take $V = \mathbb{R}^{dN}$ and replace the inner product (6.15) with:

$$a_{\mathbf{s}}(\mathbf{h}^1, \mathbf{h}^2) = \left\langle (\alpha^2 A_{\mathbf{s}} + \mathbf{I}) \mathbf{h}^1, \mathbf{h}^2 \right\rangle + \frac{1}{\varepsilon} \sum_{i=1}^B (\mathbf{h}_i^1 \cdot \mathbf{n}_i) (\mathbf{h}_i^2 \cdot \mathbf{n}_i),$$

where $A_{\mathbf{s}} \in \mathbb{R}^{2N \times 2N}$ is the matrix defined in (6.14) and ε is a ‘‘sufficiently small’’ parameter.

Another strategy consists in setting:

$$V = \{\mathbf{h} \in \mathbb{R}^{dN}, \mathbf{h}_i \cdot \mathbf{n}_i = 0 \text{ for } i = 1, \dots, B\}, \text{ with the inner product (6.15).}$$

The identification problem (6.12) for the gradient \mathbf{h}_J amounts to solving the optimization problem (6.13) under the additional constraint (6.17). This is again a convex optimization problem, whose first-order optimality conditions read:

$$\exists \boldsymbol{\lambda} \in \mathbb{R}^B \text{ s.t. } \begin{cases} \alpha^2 \sum_{j \in \mathcal{N}_i} (\mathbf{h}_i - \mathbf{h}_j) + \mathbf{h}_i + \lambda_i \mathbf{n}_i & = \nabla_{\mathbf{s}_i} \tilde{J}(\mathbf{s}) & \text{for } i = 1, \dots, B, \\ \alpha^2 \sum_{j \in \mathcal{N}_i} (\mathbf{h}_i - \mathbf{h}_j) + \mathbf{h}_i & = \nabla_{\mathbf{s}_i} \tilde{J}(\mathbf{s}) & \text{for } i = B + 1, \dots, N, \\ \mathbf{h}_i \cdot \mathbf{n}_i & = 0 & \text{for } i = 1, \dots, B. \end{cases}$$

The search for the pair $(\mathbf{h}, \boldsymbol{\lambda}) \in \mathbb{R}^{2N} \times \mathbb{R}^B$ rewrites as the following $(2N + B) \times (2N + B)$ matrix system:

$$\left(\begin{array}{c|c} \alpha^2 A_{\mathbf{s}} + \mathbf{I} & L \\ \hline L^T & \mathbf{0} \end{array} \right) \begin{pmatrix} \mathbf{h} \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \nabla_{\mathbf{s}} \tilde{J}(\mathbf{s}) \\ \mathbf{0} \end{pmatrix},$$

where the matrix $L \in \mathbb{R}^{2N \times B}$ is defined by:

$$L_{2(i-1)+1,j} = \begin{cases} n_{i,1} & \text{if } j = i, \\ 0 & \text{otherwise,} \end{cases} \quad \text{and} \\ L_{2(i-1)+2,i} = \begin{cases} n_{i,2} & \text{if } j = i, \\ 0 & \text{otherwise,} \end{cases} \quad n_{i,2} \text{ for } i = 1, \dots, N, \quad j = 1, \dots, B.$$

6.4. Exploitation of these derivatives with a constrained optimization algorithm

In this section, we describe how the derivatives of the objective and constraint functions $J(\Omega)$, $\mathbf{G}(\Omega)$ with respect to the seeds \mathbf{s} and measures ν calculated in the previous [Sections 6.1 to 6.3](#) are exploited to solve our discrete constrained optimization problem:

$$(P\text{-disc}) \quad \min_{(\mathbf{s}, \nu)} \tilde{J}(\mathbf{s}, \nu) \quad \text{s.t.} \quad \tilde{\mathbf{G}}(\mathbf{s}, \nu) = 0.$$

This task relies on a constrained optimization algorithm developed in one of our previous works [\[55\]](#) which is sketched briefly in [Section 6.4.1](#). We then describe its application to our particular context in [Section 6.4.2](#).

6.4.1. The null-space optimization algorithm in a nutshell

Let V be a Hilbert space, equipped with the inner product $a(\cdot, \cdot)$. We consider an optimization problem of the form:

$$(6.18) \quad \min_{h \in V} J(h) \quad \text{s.t.} \quad \mathbf{G}(h) = 0,$$

where $J : V \rightarrow \mathbb{R}$ is a smooth objective function and $\mathbf{G} : V \rightarrow \mathbb{R}^p$ accounts for a collection of p smooth equality constraints. We denote by $\theta_J, \theta_{G_i} \in V$, $i = 1, \dots, p$ the gradients of J and G_i at a particular given value h of the optimized variable, that is:

$$\forall \zeta \in V, \quad J'(h)(\zeta) = a(\theta_J, \zeta), \quad \text{and} \quad G'_i(h)(\zeta) = a(\theta_{G_i}, \zeta).$$

Let $h \in V$ stand for an arbitrary stage of the solution process for [\(6.18\)](#). We search for an update of h of the form $h + \Delta t \xi$, where $\Delta t > 0$ is a suitable descent step, and the direction $\xi \in V$ is of the form:

$$(6.19) \quad \xi = -(\alpha_J \xi_J + \alpha_G \xi_G).$$

The two contributions ξ_J and $\xi_G \in V$ are orthogonal, and they are characterized by the following requirements:

- The null-space step ξ_J is the best descent direction for $J(h)$ which leaves the values of the constraint functional $\mathbf{G}(h)$ unaltered at first order. Formally, ξ_J is the projection of the gradient θ_J of $J(h)$ onto the null space $\text{Ker}(\mathbf{G}'(h)) = \text{span} \{ \theta_{G_1}, \dots, \theta_{G_p} \}^\perp$ of the constraints. It is of the form

$$(6.20) \quad \xi_J = \theta_J + \sum_{j=1}^p \lambda_j \theta_{G_j} \quad \text{for some } \boldsymbol{\lambda} \in \mathbb{R}^p.$$

The requirement that $a(\xi_J, \theta_{G_j}) = 0$ for $j = 1, \dots, p$ shows that the vector $\boldsymbol{\lambda} \in \mathbb{R}^p$ is the solution to the following $p \times p$ system:

$$(6.21) \quad S \boldsymbol{\lambda} = \mathbf{b}, \quad \text{where } S \in \mathbb{R}^{p \times p} \text{ and } \mathbf{b} \in \mathbb{R}^p \text{ are given by the entries:}$$

$$S_{ij} = a(\theta_{G_i}, \theta_{G_j}), \quad \text{and } b_i = -a(\theta_J, \theta_{G_i}).$$

- The range space step ξ_G aims to decrease the violation of constraints. It belongs to $\text{Ker}(\mathbf{G}'(h))^\perp = \text{span} \{ \theta_{G_i} \}_{i=1, \dots, p}$, and so it can be expressed as

$$(6.22) \quad \xi_G = \sum_{j=1}^p \beta_j \theta_{G_j}, \quad \text{for some coefficient vector } \boldsymbol{\beta} \in \mathbb{R}^p.$$

The components β_i , $i = 1, \dots, p$ are determined by imposing the following decrease in the value of G in the course of one iteration:

$$\mathbf{G}(h + \Delta t \xi) = (1 - \Delta t \alpha_G) \mathbf{G}(h),$$

which we approximate by

$$\mathbf{G}'(h)(\xi_G) \approx \mathbf{G}(h).$$

This requirement is discretized as:

$$(6.23) \quad S\boldsymbol{\beta} = \mathbf{c}, \text{ where } S \in \mathbb{R}^{p \times p} \text{ is the matrix (6.21) and } \mathbf{c} \in \mathbb{R}^p \text{ has entries } c_i = G_i(h), \quad i = 1, \dots, p.$$

As far as the choice of the descent step Δt is concerned, we rely on a merit function: the update $h + \Delta t \xi$ of h should result in a decrease in the value of the function

$$M(h) := \alpha_J \left(J(h) - \boldsymbol{\lambda} \cdot \mathbf{G}(h) \right) + \frac{\alpha_G}{2} S^{-1} \mathbf{G}(h) \cdot \mathbf{G}(h),$$

whose negative gradient is ξ by construction.

This null-space optimization algorithm proceeds by iterations of the form (6.19), which decrease the value of the objective function $J(h)$ while gradually enforcing the constraints $G(h) = 0$, until a local minimum of the problem (6.18) is attained. We refer to the article [55] for more quantitative convergence guarantees regarding this strategy.

6.4.2. Application to the context of interest

Let us now apply the general optimization strategy of Section 6.4.1 to our discrete shape and topology optimization problem (P-disc). Since the optimized variables $\mathbf{s} \in \mathbb{R}^{dN}$ and $\boldsymbol{\nu} \in \mathbb{R}^N$ are of very independent natures, we treat optimization with respect to each of them separately, via an alternating descent strategy:

- When it comes to updating the seed points \mathbf{s} , we apply the Hilbertian method of Section 6.4.1 with the space $V = \mathbb{R}^{dN}$ and the inner product $a_{\mathbf{s}}(\cdot, \cdot)$ defined in (6.15). The gradients $\boldsymbol{\theta}_{J,\mathbf{s}}, \boldsymbol{\theta}_{G_i,\mathbf{s}}, i = 1, \dots, p$, are calculated by solving the identification problem (6.12) from the derivatives of $\mathbf{s} \mapsto \tilde{J}(\mathbf{s}, \boldsymbol{\nu})$ and $\mathbf{s} \mapsto \tilde{\mathbf{G}}(\mathbf{s}, \boldsymbol{\nu})$ supplied by Proposition 6.1. The coefficients $\boldsymbol{\lambda}_{\mathbf{s}} \in \mathbb{R}^p$ and $\boldsymbol{\beta}_{\mathbf{s}} \in \mathbb{R}^p$ involved in the expressions (6.20) and (6.22) of the steps $\boldsymbol{\xi}_{J,\mathbf{s}}$ and $\boldsymbol{\xi}_{G,\mathbf{s}}$ are then easily calculated from these data, see (6.21) and (6.23), and the resulting descent direction $\boldsymbol{\xi}_{\mathbf{s}} \in \mathbb{R}^{dN}$ for the seed points \mathbf{s} reads:

$$\boldsymbol{\xi}_{\mathbf{s}} = -\alpha_{J,\mathbf{s}} \boldsymbol{\xi}_{J,\mathbf{s}} - \alpha_{G,\mathbf{s}} \boldsymbol{\xi}_{G,\mathbf{s}}, \text{ where } \boldsymbol{\xi}_{J,\mathbf{s}} := \boldsymbol{\theta}_{J,\mathbf{s}} - \sum_{i=1}^p \lambda_{\mathbf{s},i} \boldsymbol{\theta}_{G_i,\mathbf{s}} \text{ and } \boldsymbol{\xi}_{G,\mathbf{s}} = \sum_{i=1}^p \beta_{\mathbf{s},i} \boldsymbol{\theta}_{G_i,\mathbf{s}},$$

with the weights $\alpha_{J,\mathbf{s}}$ and $\alpha_{G,\mathbf{s}}$ for the relative contributions of the null space and range space steps to the descent direction.

- When it comes to updating the vector $\boldsymbol{\nu}$, we use the space $V = \mathbb{R}^N$ and the inner product $a_{\boldsymbol{\nu}}(\cdot, \cdot)$ in (6.16). Similar considerations yield the following descent direction $\boldsymbol{\xi}_{\boldsymbol{\nu}} \in \mathbb{R}^N$ for the measures $\boldsymbol{\nu}$:

$$\boldsymbol{\xi}_{\boldsymbol{\nu}} = -\alpha_{J,\boldsymbol{\nu}} \boldsymbol{\xi}_{J,\boldsymbol{\nu}} - \alpha_{G,\boldsymbol{\nu}} \boldsymbol{\xi}_{G,\boldsymbol{\nu}}, \text{ where } \boldsymbol{\xi}_{J,\boldsymbol{\nu}} := \boldsymbol{\theta}_{J,\boldsymbol{\nu}} - \sum_{i=1}^p \lambda_{\boldsymbol{\nu},i} \boldsymbol{\theta}_{G_i,\boldsymbol{\nu}} \text{ and } \boldsymbol{\xi}_{G,\boldsymbol{\nu}} = \sum_{i=1}^p \beta_{\boldsymbol{\nu},i} \boldsymbol{\theta}_{G_i,\boldsymbol{\nu}},$$

where again $\boldsymbol{\lambda}_{\boldsymbol{\nu}}, \boldsymbol{\beta}_{\boldsymbol{\nu}} \in \mathbb{R}^p$ are calculated via (6.21) and (6.23), and $\alpha_{J,\boldsymbol{\nu}}$ and $\alpha_{G,\boldsymbol{\nu}}$ are suitable weights.

Moreover, as suggested in [55], instead of considering fixed weights $\alpha_{J,\mathbf{s}}, \alpha_{G,\mathbf{s}}$ for the contributions of the null space and range space steps, we consistently select these coefficients as:

$$\alpha_{J,\mathbf{s}} = \frac{A_{J,\mathbf{s}} \mathbf{h}_{\text{av}}}{|\boldsymbol{\xi}_{J,\mathbf{s}}|_{\infty}} \text{ and } \alpha_{G,\mathbf{s}} = \frac{A_{G,\mathbf{s}} \mathbf{h}_{\text{av}}}{|\boldsymbol{\xi}_{G,\mathbf{s}}|_{\infty}}$$

where $A_{J,\mathbf{s}}$, and $A_{G,\mathbf{s}} \in [0, 1]$ are fixed parameters controlling the decrease rates of the objective function and the violation of the constraint, and \mathbf{h}_{av} is the typical distance between two neighboring seed points. This allows to bound the maximum displacement of a seed induced by the update $\boldsymbol{\xi}_{\mathbf{s}}$ in this step by $(A_{J,\mathbf{s}} + A_{G,\mathbf{s}}) \mathbf{h}_{\text{av}}$. Likewise, we take

$$\alpha_{J,\boldsymbol{\nu}} = \frac{A_{J,\boldsymbol{\nu}} V_{\text{av}}}{|\boldsymbol{\xi}_{J,\boldsymbol{\nu}}|_{\infty}}, \text{ and } \alpha_{G,\boldsymbol{\nu}} = \frac{A_{G,\boldsymbol{\nu}} V_{\text{av}}}{|\boldsymbol{\xi}_{G,\boldsymbol{\nu}}|_{\infty}},$$

where $A_{J,\boldsymbol{\nu}}, A_{G,\boldsymbol{\nu}} \in [0, 1]$ and V_{av} is the average volume of cells. Doing so ensures that the maximum change in a local volume fraction induced in the update of $\boldsymbol{\nu}$ by $\boldsymbol{\xi}_{\boldsymbol{\nu}}$ is $(A_{J,\boldsymbol{\nu}} + A_{G,\boldsymbol{\nu}}) V_{\text{av}}$.

Remark 6.5. In the optimization problems featured in the examples of the next [Section 7](#), only $p = 1$ equality constraint is imposed, which is related to the volume of the shape Ω as: $G(\Omega) = \text{Vol}(\Omega) - V_T$. In our framework, this constraint can be accounted for by imposing that the measure of each cell be consistently equal to a given fraction of the target volume V_T throughout the iterations of the process. By doing so (P-disc) is reduced to an unconstrained optimization problem over the sole seed points \mathbf{s} , see the examples in [Sections 7.1](#) and [7.2](#). On the contrary, the treatment of the volume constraint with the help of the above strategy, which solely concerns the sum of all individual cell measures, leaves more freedom in the evolution of the optimized design, see [Sections 7.3](#) to [7.6](#). In such a situation, the foregoing formulas conveniently simplify into:

$$\boldsymbol{\theta}_{G,\mathbf{s}} = 0, \text{ and } \boldsymbol{\theta}_{G,\boldsymbol{\nu}} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \in \mathbb{R}^N,$$

so that the optimization with respect to the seed points \mathbf{s} is unconstrained, and the coefficients featured in the above scheme [\(6.20\)](#), [\(6.22\)](#) and [\(6.23\)](#) read

$$S = a_{\boldsymbol{\nu}}(\boldsymbol{\theta}_{G,\boldsymbol{\nu}}, \boldsymbol{\theta}_{G,\boldsymbol{\nu}}) = N, \quad \lambda = \frac{1}{N} a_{\boldsymbol{\nu}}(\boldsymbol{\theta}_{J,\boldsymbol{\nu}}, \boldsymbol{\theta}_{G,\boldsymbol{\nu}}) \text{ and } \beta = \frac{1}{N} (\widetilde{\text{Vol}}(\mathbf{s}, \boldsymbol{\nu}) - V_T).$$

Note that if the volume constraint is satisfied at the current stage $(\mathbf{s}, \boldsymbol{\nu})$ of the optimization process, then $\xi_G = 0$. In particular, if the vector $\boldsymbol{\nu}$ of cell measures satisfies $\sum_{i=1}^N \nu_i = V_T$, the update [\(6.19\)](#) of the null-space optimization algorithm for [\(6.18\)](#) coincides (up to a positive coefficient) with that of the unconstrained gradient algorithm for the minimization of $\tilde{J}(\cdot, \boldsymbol{\nu})$.

7. NUMERICAL EXAMPLES

In this section, we present various 2d numerical examples to illustrate the main features of our shape optimization [Algorithm 1](#). After starting in [Section 7.1](#) with a “simple” geometric optimization problem, we revisit in [Section 7.2](#) the classical subject of optimization of the eigenvalues of the Laplace operator with our new framework. We then turn to more concrete physical applications in the contexts of thermal devices in [Section 7.3](#) and of mechanical structures in [Sections 7.4](#) to [7.6](#).

7.1. Optimization of the perimeter of shapes

Our first example deals with a well-known shape optimization problem, featuring only geometric functionals of the domain, i.e. no physical boundary value problem is involved. We minimize the perimeter of the shape $\Omega \subset \mathbb{R}^2$ under an equality constraint on its volume:

$$(7.1) \quad \min_{\Omega} \text{Per}(\Omega) \text{ s.t. } \text{Vol}(\Omega) = V_T,$$

where $\text{Per}(\Omega)$ and $\text{Vol}(\Omega)$ are defined by [\(2.1\)](#) and V_T is a volume target.

Among the numerous motivations for [\(7.1\)](#), let us mention that the flow induced by its solution from an initial shape Ω^0 is the well-known mean curvature flow, which proceeds by steadily blurring its sharp features (i.e. its bumps and creases with high mean curvature) while preserving its total volume, see e.g. [\[85\]](#) about the latter.

From the mathematical viewpoint, such problems have been extensively considered in the literature, see e.g. [\[102, 107\]](#). In particular, according to the famous isoperimetric inequality, the unique global minimizers to [\(7.1\)](#) are 2d disks with radius $\sqrt{V_T/\pi}$.

Remark 7.1. In this example, the discrete versions $\text{Vol}(\mathbf{q})$ and $\text{Per}(\mathbf{q})$ of the volume and perimeter $\text{Vol}(\Omega)$ and $\text{Per}(\Omega)$ of the shape Ω as well as their derivatives can be explicitly calculated from the polygonal mesh \mathcal{T} induced by the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ representing Ω . Indeed, simple calculations show that:

$$\text{Vol}(\mathbf{q}) = \sum_{E \in \mathcal{T}} |E|,$$

where the measure of an element E with vertices $\mathbf{q}_1^E, \dots, \mathbf{q}_n^E$ can be calculated by the following formula, where we use the notations of [Section 5.1](#):

$$|E| = \frac{1}{2} \sum_{j=1}^n |\hat{\mathbf{e}}_j| \mathbf{n}_{\hat{\mathbf{e}}_j} \cdot \mathbf{q}_j^E.$$

Taking derivatives, then summing over the elements $E \in \mathcal{T}$, an elementary calculation shows that the gradient of the volume functional equals:

$$\nabla_{\mathbf{q}_j} \text{Vol}(\mathbf{q}) = \begin{cases} \frac{1}{2} |\hat{\mathbf{e}}_j| \mathbf{n}_{\hat{\mathbf{e}}_j} & \text{if } \mathbf{q}_j \text{ is a vertex on the boundary } \partial\Omega \\ \mathbf{0} & \text{otherwise.} \end{cases}$$

Likewise, assuming for notational simplicity that the vertices \mathbf{q}_j are enumerated in such a way that the border $\partial\Omega$ is discretized by the $n_b < M$ first vertices $\mathbf{q}_1, \dots, \mathbf{q}_{n_b} = \mathbf{q}_0$ of the collection \mathbf{q} and the associated line segments $\mathbf{q}_{j-1}\mathbf{q}_j$, $j = 1, \dots, n_b$, one has

$$\text{Per}(\mathbf{q}) = \sum_{j=1}^{n_b} |\mathbf{q}_j \mathbf{q}_{j+1}|,$$

and so

$$\nabla_{\mathbf{q}_j} \text{Per}(\mathbf{q}) = \begin{cases} \frac{\mathbf{q}_j - \mathbf{q}_{j-1}}{|\mathbf{q}_j - \mathbf{q}_{j-1}|} + \frac{\mathbf{q}_j - \mathbf{q}_{j+1}}{|\mathbf{q}_j - \mathbf{q}_{j+1}|} & \text{if } j = 1, \dots, n_b, \\ \mathbf{0} & \text{otherwise.} \end{cases}$$

We apply the “free boundary” version of our shape and topology optimization [Algorithm 1](#) to the solution of this problem. The initial shape Ω^0 is represented by a diagram $\mathbf{V}(\mathbf{s}^0, \boldsymbol{\psi}^0)$ made of a random collection of $N = 500$ cells with equal measures $\bar{\nu} := V_T/N$. [In this example, for simplicity](#), we impose that throughout the optimization process, the cells $V_i(\mathbf{s}^n, \boldsymbol{\psi}^n)$, $i = 1, \dots, N$, satisfy this measure constraint, so that our optimization procedure becomes constraint free, as noted in [Remark 6.5](#). A few snapshots of the solution process are depicted on [Fig. 9](#). As expected, the optimized shape is close to a disk. Remarkably, the shape undergoes drastic topological changes during the process: our algorithm is able to detect that the only means to decrease the total perimeter of the shape Ω by moving the defining seed points \mathbf{s} without altering the measures $\boldsymbol{\nu}$ of the cells is to aggregate the cells. [The complexity of the evolution process, induced by the choice of an initial shape so different from the expected disk, explains the a priori surprisingly large number of iterations \(200\) needed to achieve convergence.](#) Note that this operation – and the underlying shape sensitivity of the objective functional – is completely different from that of a more classical boundary variation algorithm as in e.g. [\[8, 6, 105\]](#), whereby each individual cell would shrink, regardless of the others, without capturing this collective behavior. [This trend of our algorithm to favor transport of cells over independent adjustments of their measures is a general consequence of the parametrization of shapes in terms of the seed points and cell measures of a defining diagram; this feature is reflected in the formulas of Remark 6.5, where the gradient \$\boldsymbol{\theta}_{\text{Vol}, \boldsymbol{\nu}}\$ of the volume functional \$\text{Vol}\(\Omega\)\$ has identical components for all cells.](#) We shall retrieve this trend in the mechanical example of [Section 7.4](#).

7.2. Optimization of the eigenvalues of the Dirichlet Laplace operator

Our first example involving boundary value problems concerns the minimization of some of the eigenvalues of the Laplace operator equipped with Dirichlet boundary conditions. We consider the problem:

$$(7.2) \quad \min_{\Omega} \lambda_{\Omega}^{(k)} \text{ s.t. } \text{Vol}(\Omega) = V_T,$$

where we recall from [Section 2.2.1](#) that $\lambda_{\Omega}^{(k)}$ is the k^{th} lowest real number for which there exists a non trivial function $u \in H_0^1(\Omega)$ such that

$$-\Delta u = \lambda_{\Omega}^{(k)} u, \text{ with the normalization } \int_{\Omega} u_{\Omega}^2 dx = 1.$$

These eigenvalues arise in multiple applications, such as acoustics (where they are related to the properties of sound propagation), thermal conduction (where they encode the dissipation rate of a peak of temperature over time), and structure mechanics (where they represent the vibration modes of a thin membrane), to name a few. Spectral shape optimization problems, of the form [\(7.2\)](#), have been extensively studied in the literature, from both theoretical [\[16, 93, 68\]](#) and numerical [\[13, 103\]](#) viewpoints.

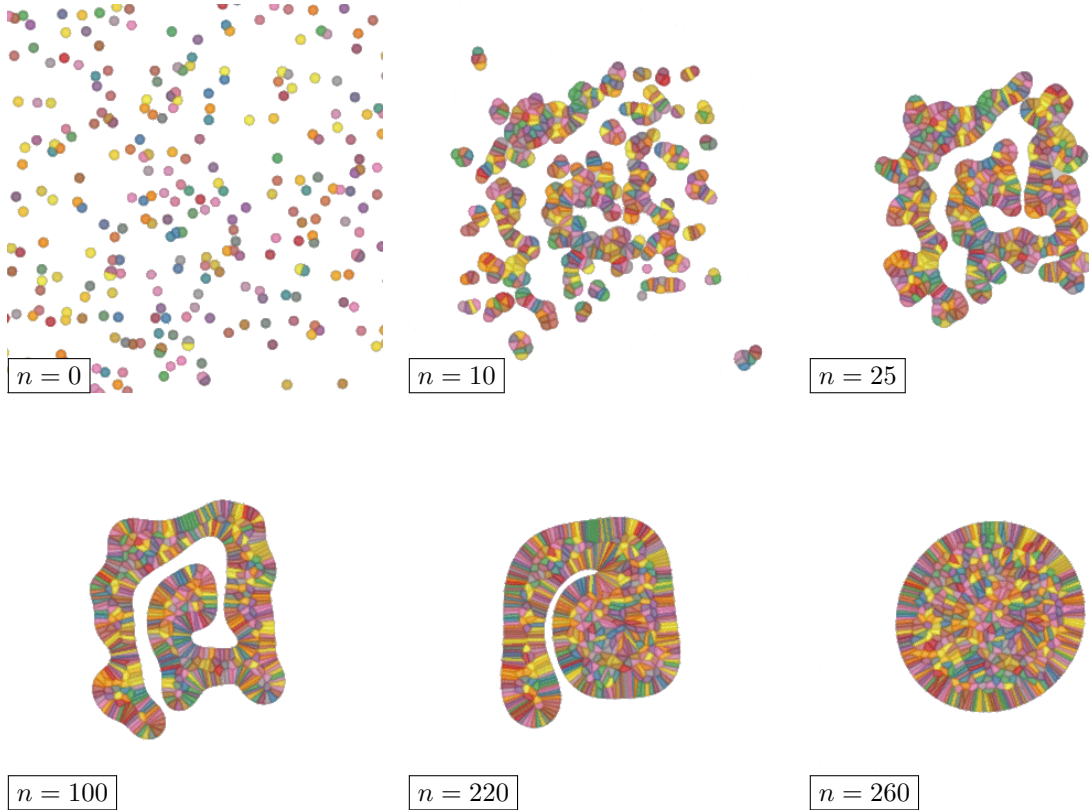


FIGURE 9. A few intermediate shapes Ω^n produced in the course of the solution of the perimeter minimization problem (7.1) in Section 7.1. The cells $V_i(\mathbf{s}^n, \psi^n)$ of the diagram, $i = 1, \dots, N$, are represented with different colors to help visualization.

We address the numerical solution of the problem (7.2) thanks to the “free boundary” strategy described in Section 3.3. The shape Ω is described by a diagram $\mathbf{V}(\mathbf{s}, \psi)$ attached to collections \mathbf{s} and ψ of N seed points and weights, respectively. As in the previous Section 7.1, the measures of the cells are consistently set to the common value $\bar{v} := V_T/N$, so that the solution of (7.2) boils down to the unconstrained minimization of $\lambda_\Omega^{(k)}$ with respect to the positions \mathbf{s} of the seed points, see again Remark 6.5. On a different note, the sensitivity of the discrete version of $\lambda_\Omega^{(k)}$ with respect to the vertices \mathbf{q} of $\mathbf{V}(\mathbf{s}, \psi)$ is calculated by automatic differentiation before being interpreted in terms of the positions \mathbf{s} of the seed points, see Sections 6.1 and 6.2.

In this context, we conduct two numerical experiments. At first, we address the minimization of the first Dirichlet eigenvalue, i.e. $k = 1$ in (7.2). It is well-known since the seminal conjecture of Lord Rayleigh [109] that disks with radius $\sqrt{V_T/\pi}$ are the unique minimizers of (7.2), see [29] for a rigorous proof of the associated Faber-Krahn inequality. The initial shape Ω^0 is depicted in Fig. 10. It is discretized with $N = 1000$ cells with equal measures $\bar{v} = V_T/N$. We apply 200 iterations of our optimization Algorithm 1 to solve this problem; a few intermediate shapes are depicted on Figs. 10 and 11, and the associated convergence history is reported in Fig. 13 (a); as expected, the optimized shape is a disk. Note that disconnected structures naturally appear during the optimization process. While they are harmless to the resolution, we explicitly enforce the connectedness of the shape at each iteration by removing and randomly resampling the cells that are disconnected from the main region.

We next turn to a more challenging problem of the form (7.2), featuring the 5th Dirichlet eigenvalue, i.e. $k = 5$ in (7.2). The fifth eigenmode is indeed suspected to be the lowest order eigenvalue that is minimized by a shape which is neither a disk, nor a reunion of disks. Moreover, it is expected that, at the optimum, this eigenvalue is multiple, leading to a non differentiable behavior of this function of the domain. Starting from an initial shape Ω^0 made of $N = 1000$ cells with equal measures, we apply 1000 iterations of our shape and

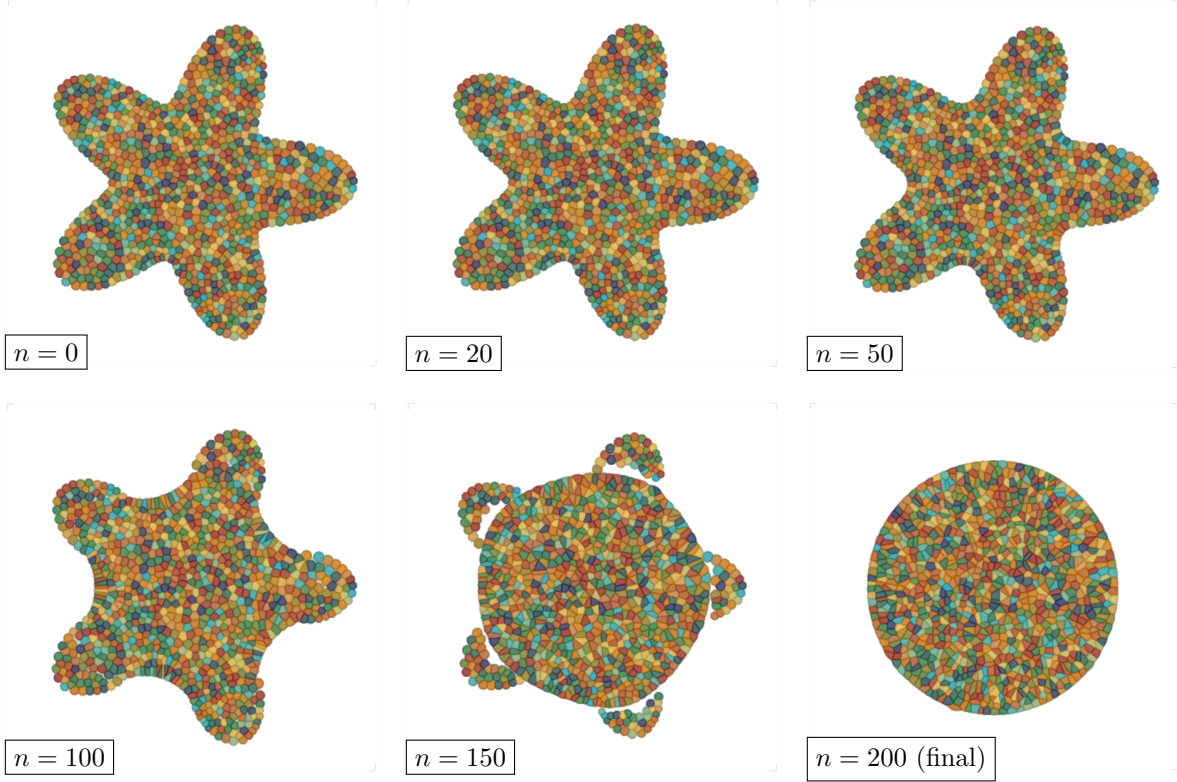


FIGURE 10. Intermediate shapes Ω^n produced in the course of the minimization of the first eigenvalue of the Dirichlet Laplace operator in [Section 7.2](#).

topology optimization [Algorithm 1](#) to the solution of this problem. A few intermediate shapes Ω^n arising in the course of the process are depicted on [Fig. 12](#), and the associated convergence history is reported in [Fig. 13](#) (b). The resulting optimized shape is qualitatively very similar to the candidate for minimizer evidenced in [\[103\]](#).

7.3. Shape optimization in two-phase conductivity

From this section on, we deal with shape optimization problems which are more directly related to mechanical applications. We first consider the setting of the conductivity equation presented in [Section 2.2.1](#), and notably its two-phase version, evoked in [Remark 2.1](#).

The situation under scrutiny is that depicted in [Fig. 14](#) (a): inside a fixed computational domain D with size 1×1 , the shape $\Omega \subset D$ accounts for one phase $\Omega_1 := \Omega$ occupied by a material with high conductivity $\gamma_1 = 10$, the complementary phase $\Omega_0 := D \setminus \bar{\Omega}$ being filled by a material with low conductivity $\gamma_0 = 1$. The temperature u_Ω within D is set to 0 on a region $\Gamma_D \subset \partial D$ located at the left-hand side of ∂D , while the remaining part $\Gamma_N := \partial D \setminus \bar{\Gamma}_D$ is insulated from the outside. Assuming the presence of a constant source $f \equiv 1$ within D , u_Ω is the solution to the following two-phase conductivity problem:

$$(7.3) \quad \begin{cases} -\operatorname{div}(\gamma_\Omega \nabla u_\Omega) = f & \text{in } D, \\ u_\Omega = 0 & \text{on } \Gamma_D, \\ \gamma_\Omega \frac{\partial u_\Omega}{\partial n} = 0 & \text{on } \Gamma_N, \end{cases} \quad \text{where } \gamma_\Omega(\mathbf{x}) = \begin{cases} \gamma_1 & \text{if } \mathbf{x} \in \Omega_1, \\ \gamma_0 & \text{otherwise.} \end{cases}$$

We aim to minimize the mean temperature $T(\Omega)$ within D , i.e. we solve:

$$(7.4) \quad \min_{\Omega \subset D} T(\Omega) \text{ s.t. } \operatorname{Vol}(\Omega) = V_T,$$

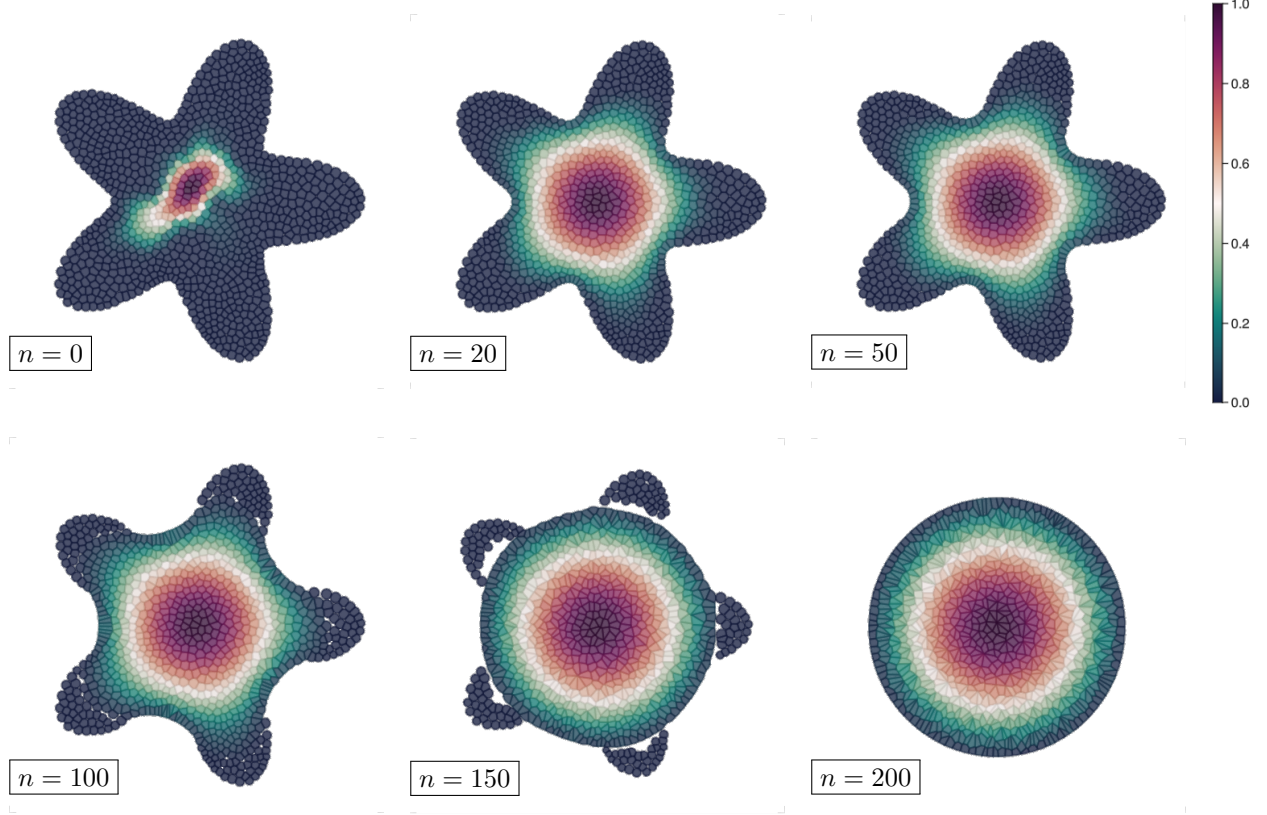


FIGURE 11. Graph of the first eigenfunction of the Dirichlet Laplace operator at various iterations of the minimization process of the first Dirichlet eigenvalue in [Section 7.2](#).

where V_T is a volume target, and we have defined:

$$T(\Omega) = \frac{1}{\text{Vol}(D)} \int_D u_\Omega \, dx.$$

In this context, the shape Ω is represented via a classical Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \psi)$ of D : complementary subcollections of cells of the latter account for Ω and $D \setminus \overline{\Omega}$, in the sense that [\(3.5\)](#) holds true. Moreover, contrary to our practice in the previous [Sections 7.1](#) and [7.2](#), we rely on the non intrusive approach of [Section 6.1](#) to calculate the derivative of $T(\Omega)$ with respect to the seed points and weights of the representing diagram. In this perspective, we recall in the following proposition the expression of the shape derivative of $T(\Omega)$; since the result is fairly classical in the literature, we omit the proof for brevity, see e.g. [\[7\]](#).

Proposition 7.1. *The functional $T(\Omega)$ is shape differentiable at any shape $\Omega \subset D$, and its derivatives reads for any deformation field $\boldsymbol{\theta}$ vanishing on $\overline{\Gamma_D}$:*

$$T'(\Omega)(\boldsymbol{\theta}) = \frac{1}{\text{Vol}(D)} \int_D \text{div}(\boldsymbol{\theta})u_\Omega \, dx + \int_D \gamma_\Omega(\text{div}\boldsymbol{\theta}\mathbf{I} - \nabla\boldsymbol{\theta} - \nabla\boldsymbol{\theta}^T)\nabla u_\Omega \cdot \nabla p_\Omega \, dx - \int_D \text{div}(f\boldsymbol{\theta})p_\Omega \, dx,$$

where the adjoint state p_Ω is the unique $H^1(D)$ solution to the boundary value problem:

$$\begin{cases} -\text{div}(\gamma_\Omega \nabla p_\Omega) = -\frac{1}{\text{Vol}(D)} & \text{in } D, \\ p_\Omega = 0 & \text{on } \Gamma_D, \\ \gamma_\Omega \frac{\partial p_\Omega}{\partial n} = 0 & \text{on } \Gamma_N. \end{cases}$$

Remark 7.2. *Since in our context the source f is identically equal to 1, we immediately see that the adjoint state p_Ω equals $p_\Omega = -\frac{1}{\text{Vol}(D)}u_\Omega$. This self-adjoint property of the problem [\(7.4\)](#) conveniently allows to avoid the solution of an extra boundary value problem at each stage of the optimization process.*

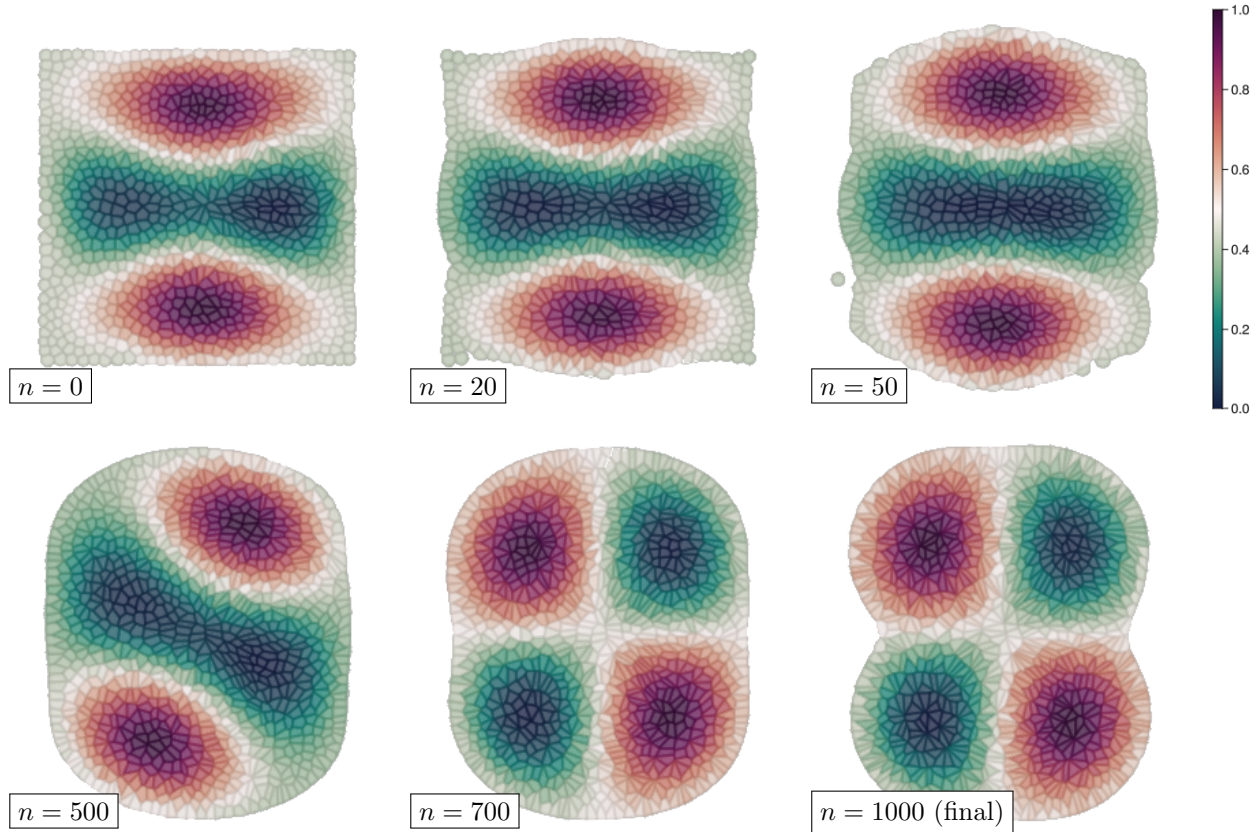


FIGURE 12. *Intermediate shapes arising in the course of the minimization of the 5th Dirichlet eigenvalue $\lambda_{\Omega}^{(5)}$ in Section 7.2; the colors refer to the values of an associated eigenfunction.*

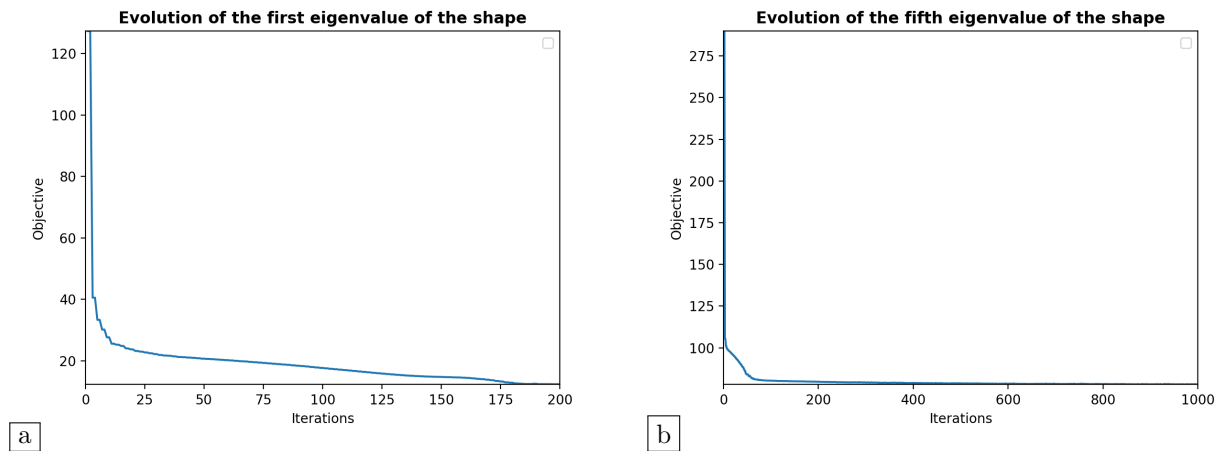


FIGURE 13. *(a) Evolution of the first eigenvalue $\lambda_{\Omega}^{(1)}$ of the shape in the course of the first experiment of Section 7.2; (b) Evolution of the fifth eigenvalue in the course of the second experiment.*

We perform 200 iterations of our shape and topology optimization [Algorithm 1](#) to solve this problem. On average, the considered diagrams contain about 3,500 cells, and the total computation takes about 45

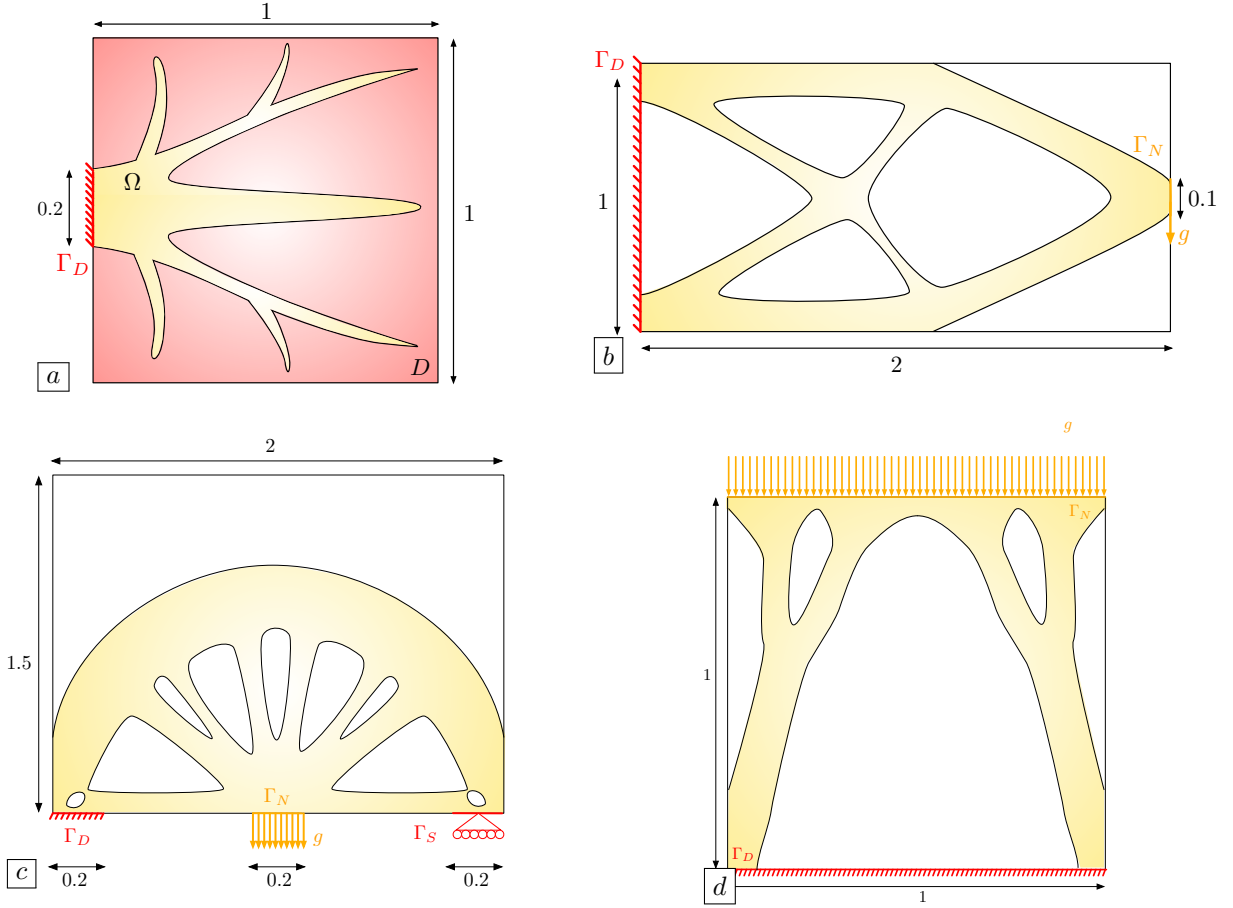


FIGURE 14. *Settings of the physical numerical examples of Section 7: (a) The heat diffuser test case of Section 7.3; (b) The elastic cantilever example of Section 7.4; (c) The optimal bridge example of Section 7.5; (d) The self-sustaining bridge of Section 7.6.*

min on a regular Macbook Pro laptop with 2 GHz Quad-Core Intel Core i5 and 16 Gb of memory. A few snapshots of the optimization path are presented in Fig. 15, and the associated convergence histories are reported in Fig. 16. Again, the shape undergoes dramatic deformations in the course of the process, while being consistently equipped with an explicit discretization, which is updated in a Lagrangian manner.

7.4. Optimization of the shape of an elastic cantilever beam

The examples in this section take place in the physical context of mechanical structures presented in Section 2.2.2. The considered shapes Ω are enclosed in a fixed computational box D with size 2×1 ; they are fixed on the left-hand side Γ_D of the boundary ∂D , and surface loads $\mathbf{g} : \Gamma_N \rightarrow \mathbb{R}^2$ are applied on a region Γ_N at the right-hand side of ∂D . The remaining region $\Gamma := \partial\Omega \setminus (\overline{\Gamma_D} \cup \overline{\Gamma_N})$ is the only one which is subject to optimization, see Fig. 14 (b). Omitting body forces for simplicity, the displacement $\mathbf{u}_\Omega : \Omega \rightarrow \mathbb{R}^d$ of the shape is the solution to the linear elasticity system:

$$(7.5) \quad \begin{cases} -\operatorname{div}(Ae(\mathbf{u}_\Omega)) = \mathbf{0} & \text{in } \Omega, \\ \mathbf{u}_\Omega = \mathbf{0} & \text{on } \Gamma_D, \\ Ae(\mathbf{u}_\Omega)\mathbf{n} = \mathbf{g} & \text{on } \Gamma_N, \\ Ae(\mathbf{u}_\Omega)\mathbf{n} = \mathbf{0} & \text{on } \Gamma. \end{cases}$$

In this situation, we consider the following shape optimization problem:

$$(7.6) \quad \min_{\Omega \subset D} C(\Omega) \text{ s.t. } \operatorname{Vol}(\Omega) = V_T,$$

where $C(\Omega)$ is the compliance (2.7) of Ω and the volume target V_T is set to 0.7.

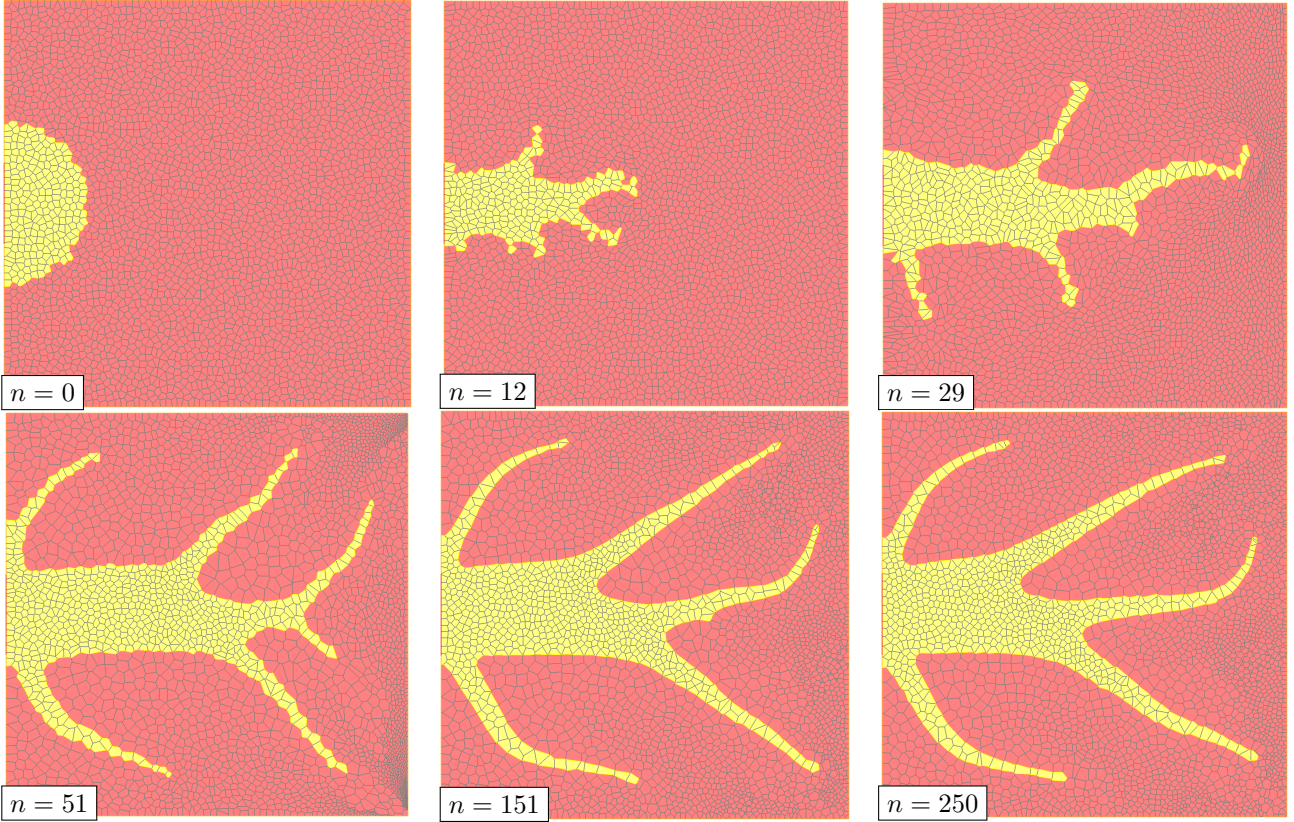


FIGURE 15. A few intermediate design Ω^n in the solution of the two-phase conductivity optimization problem of [Section 7.3](#).

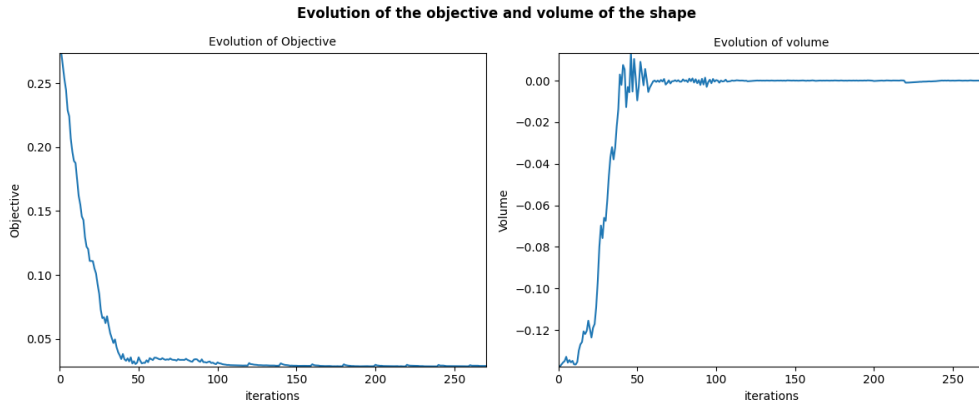


FIGURE 16. Evolution of the objective $T(\Omega)$ and constraint $G(\Omega) = \text{Vol}(\Omega) - V_T$ during the solution of the two-phase conductivity optimization problem of [Section 7.3](#).

To address this problem, we consider both shape optimization strategies presented in [Section 3.2](#). In a first experiment, we rely on the “free boundary” approach: the shape is discretized as a diagram $\mathbf{V}(\mathbf{s}, \psi)$, see [Definition 3.2](#). The $N = 3000$ cells of this diagram are consistently endowed with the same measure $\text{Vol}(\Omega)/N$. At each stage of the optimization process, the derivative of the objective function $C(\Omega)$ with respect to the vertices \mathbf{q} of this diagram is calculated via automatic differentiation, and the sensitivity of this function with respect to the seed points \mathbf{s} of the diagram is inferred along the lines of [Section 6.1](#), while

the cell measures ν are kept fixed. Concurrently, we periodically decrease the measures of all cells by a fixed amount until the desired volume constraint V_T is attained. The optimized design and a few intermediate shapes resulting from this procedure are depicted in Fig. 17. Remarkably, the topology of the shape changes dramatically in the course of the evolution, while no topological derivative is involved in the process. Holes spontaneously appear in mechanically relevant locations inside the structure, pretty much in the same manner as the drastic topological changes occurring in the example of Section 7.1 were nevertheless relevant from the optimization viewpoint. This observation confirms that the notion of sensitivity with respect to the domain associated to our representation of shapes via diagrams of the form $\mathbf{V}(\mathbf{s}, \psi)$ contains a much richer amount of information when compared to more classical boundary variation algorithms. One drawback of this approach is the difficulty to control the connectedness of the shape when optimizing the volume of the cells by means of the strategy of Section 6.2, which is why the heuristic volume update strategy outlined above is preferred to the use of the formulas for derivative with respect to volume fractions.

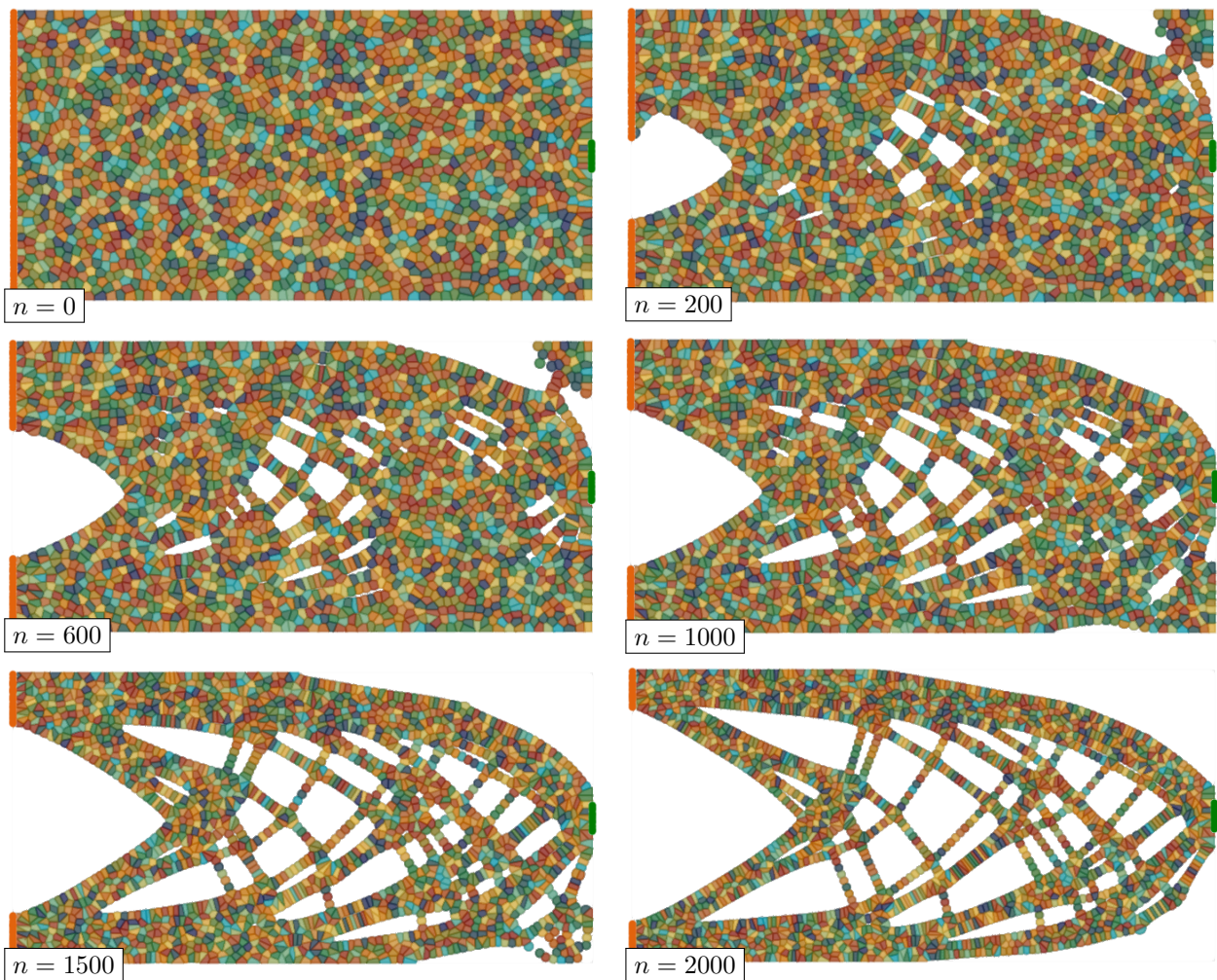


FIGURE 17. Various iterates in the solution of the cantilever optimization problem of Section 7.4 using a discretization of shapes by modified diagrams (3.6) and (3.7).

We conduct a second experiment in the same physical context, where we rely on the two-phase diagram approach for the representation of the shape Ω : the total computational domain D is equipped with a classical Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \psi)$, and the shape Ω is defined as a subcollection of the cells of this diagram, see Definition 3.1 and (3.5). In this setting, we follow the “optimize-then-discretize” approach of Section 6.1 to calculate the sensitivity of $C(\Omega)$ with respect to the vertices \mathbf{q} of the diagram (although

automatic differentiation could be used as in the previous examples), before expressing this information in terms of the seed points \mathbf{s} and cell measures $\boldsymbol{\nu}$ of the diagram, along the lines of [Section 6.2](#). This practice relies on the continuous formula for the shape derivative of $C(\Omega)$. This result is fairly classical in the literature (again, see e.g. [\[7\]](#)) and we limit ourselves with the statement of the result.

Proposition 7.2. *The shape derivative of the compliance $C(\Omega)$ reads, for any vector field $\boldsymbol{\theta}$ vanishing on $\overline{\Gamma_D} \cup \overline{\Gamma_N}$:*

$$C'(\Omega)(\boldsymbol{\theta}) = - \int_{\Omega} \operatorname{div}(\boldsymbol{\theta}) A e(\mathbf{u}_{\Omega}) : e(\mathbf{u}_{\Omega}) \, d\mathbf{x} + 4\mu \int_{\Omega} (\nabla \mathbf{u}_{\Omega} \nabla \boldsymbol{\theta}) : e(\mathbf{u}_{\Omega}) \, d\mathbf{x} + 2\lambda \int_{\Omega} \operatorname{tr}(\nabla \mathbf{u}_{\Omega} \nabla \boldsymbol{\theta}) \operatorname{div}(\mathbf{u}_{\Omega}) \, d\mathbf{x}.$$

Note that the compliance $C(\Omega)$ has the convenient property to make the adjoint state \mathbf{p}_{Ω} equal to the state function \mathbf{u}_{Ω} (up to a sign).

We apply our optimization [Algorithm 1](#) to the solution of this problem. Actually, acknowledging that the retained discretization of shapes causes the boundary to be a little rough, we add a very small penalization to the objective function $C(\Omega)$ of the problem by the perimeter functional $\operatorname{Per}(\Omega)$. The featured Laguerre diagrams contain on average 7000 cells, and the computation proceeds in 200 iterations, for a total time of about 80 min. A few iterates of the optimization process are depicted on [Fig. 18](#), and the associated convergence histories are represented on [Fig. 19](#). The optimization path is much smoother in this second experiment as in the former one, which can be explained by the fact that variations of the measures of the cells do not incur non differentiability of the objective function so easily as in the “free boundary” context. Indeed, holes cannot emerge naturally inside the bulk structure; this leaves much less room for one of the conditions [\(G1\)](#) to [\(G7\)](#) for the differentiability of the vertices of the diagram with respect to seed points and cell measures to become violated. However, one drawback of this approach is admittedly that the total computational domain D has to be equipped with a diagram, thus resulting in an increase in computational burden, especially in the perspective of the 3d extension of this work.

The above experiments suggest a few comments about the relative merits of preferring classical Laguerre diagrams $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ over their modified versions $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ to represent the shape Ω . On the one hand, the classical representation $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ is natural when Ω accounts for one phase within a larger domain D , whose complement is also involved in the physical problem at stake, as in the previous [Section 7.3](#). As we have just seen, the parametrization $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ of Ω also proves useful when it is the only phase involved, as it leads to a smoother optimization path. Moreover, even in such situations where Ω is the only phase at play, a discretization of its complement could be handful, e.g. to account for constraints about the distance between different members of the shape, as discussed in [\[89\]](#). On the other hand, the classical Laguerre diagram $\mathbf{Lag}(\mathbf{s}, \boldsymbol{\psi})$ representation of Ω comes at the price of fewer possibilities for topological changes as holes cannot emerge inside the bulk of the structure unless a specific operator is used to this end, see the next [Section 7.5](#) about the use of topological derivatives. Moreover, the computational burden is increased since the total domain D has to be discretized by Laguerre cells.

7.5. Optimization of the shape of a bridge by combination of shape and topological derivatives

This section deals with the optimization of another type of linearly elastic structures, namely a two-dimensional bridge. The considered shapes Ω are enclosed in a box D with size 2×1.5 ; they are clamped on a region Γ_D around their lower-left corner, and the vertical displacement is prevented on another region Γ_S around their lower-right corner. A unit vertical load $\mathbf{g} = (0, -1)$ is applied on a region Γ_N at the middle of their lower side, see [Fig. 14](#) (c).

In this example, we again minimize the compliance of the structure under a volume constraint, i.e. [\(7.6\)](#) is solved, with the volume target $V_T = 0.7$. We rely on a discretization of the shape Ω by two-phase Laguerre diagrams of D , see [Definition 3.1](#) and [\(3.5\)](#), and we calculate the sensitivity of the compliance $C(\Omega)$ with respect to the defining parameters \mathbf{s} and $\boldsymbol{\nu}$ of the diagram by means of the “optimize-then-discretize” approach presented in [Section 6.1](#). We also add another ingredient with respect to the workflow of the previous section: we periodically use the topological derivative to try and drill a tiny hole inside Ω in an optimal way, see [Remark 2.5](#). In this perspective, let us recall the following result about the topological derivative of the compliance functional in two space dimensions, see [\[63, 99\]](#).

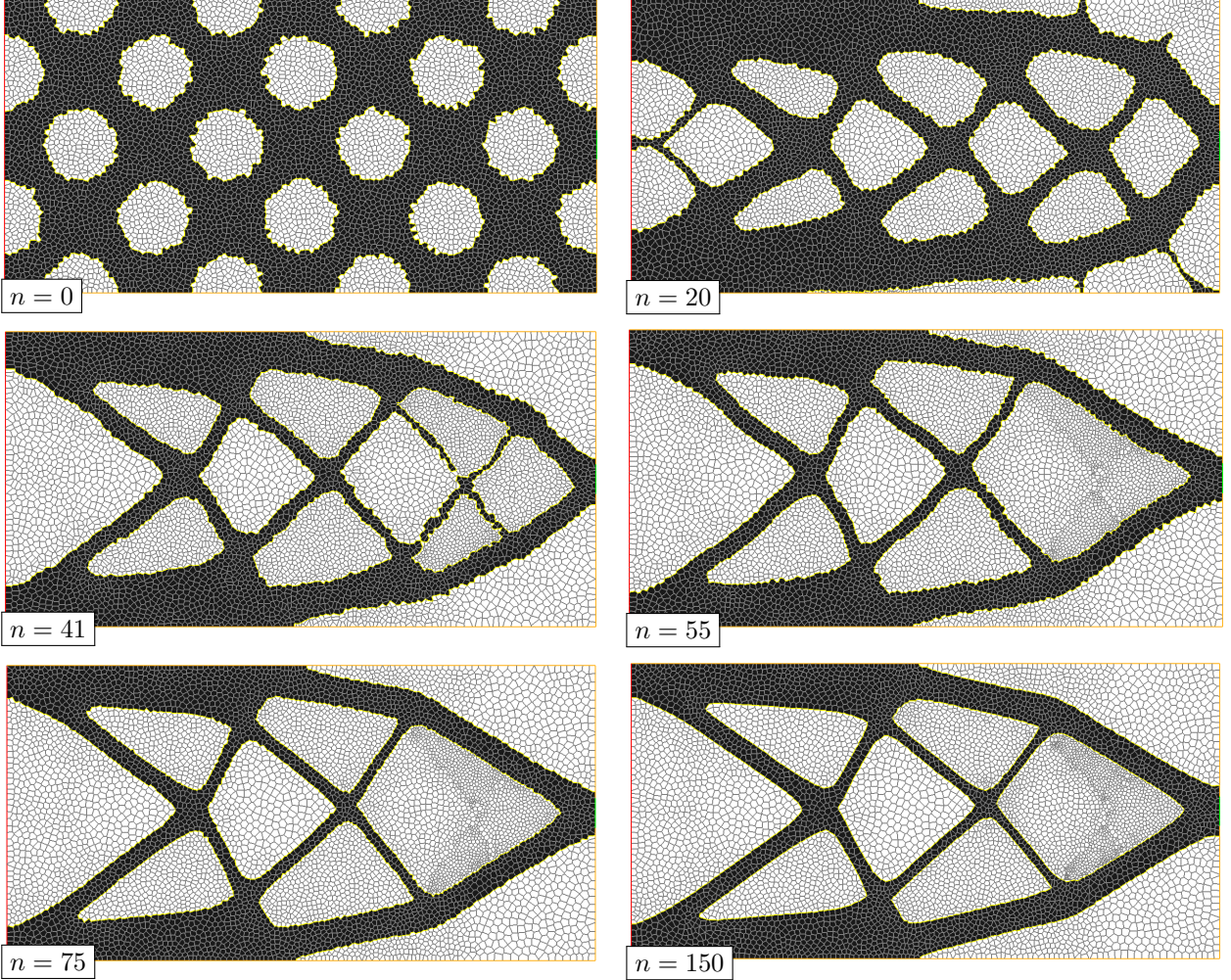


FIGURE 18. Various iterates in the solution of the cantilever optimization problem of [Section 7.4](#) via a discretization of shapes using two-phase Laguerre diagrams of the computational domain D .

Theorem 7.1. *Let $d = 2$, and let $\Omega \subset \mathbb{R}^2$ be a bounded, Lipschitz domain. The elastic compliance $C(\Omega)$ defined in [\(2.7\)](#) has a topological derivative at every point $\mathbf{x} \in \Omega$, which reads:*

$$d_T C(\Omega)(\mathbf{x}) = \frac{\pi(\lambda + 2\mu)}{2\mu(\lambda + \mu)} \left(4\mu A e(\mathbf{u}_\Omega) : e(\mathbf{u}_\Omega) + (\lambda - \mu) \text{tr}(A e(\mathbf{u}_\Omega)) \text{tr}(e(\mathbf{u}_\Omega)) \right) (\mathbf{x}).$$

Starting from an initial shape Ω^0 with trivial topology, see [Fig. 20](#), we apply 200 iterations of our shape and topology optimization [Algorithm 1](#), augmented with the aforementioned use of topological derivatives. More precisely, once every 10 iterations, the quantity $d_T C(\Omega)$ is calculated inside each cell defining the current shape, and a fixed fraction (2.5%) of the volume $\text{Vol}(\Omega)$ is removed by relabelling the cells where $d_T C(\Omega)$ has the most negative values as exterior cells. On average, the considered Laguerre diagrams contain about 7000 cells, and the total computation takes about 80 min. A few intermediate shapes arising during the resolution are depicted in [Fig. 20](#), and the convergence history is provided in [Fig. 21](#). Although the initial shape has a very poor topology, the shape develops very non trivial features in the course of the optimization process, thanks to the use of topological derivatives. The final, optimized design is very reminiscent of those obtained in the same context in e.g. [\[41\]](#).

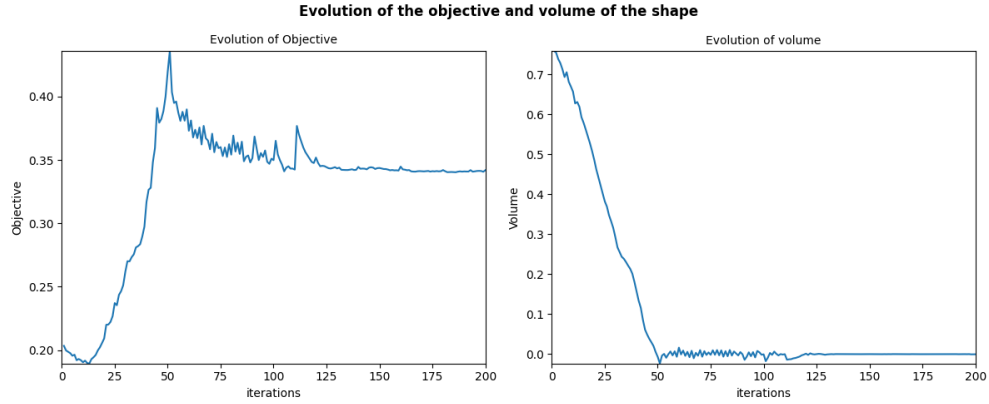


FIGURE 19. Evolution of the objective (mechanical compliance) and volume during the solution of the cantilever optimization problem of [Section 7.4](#).

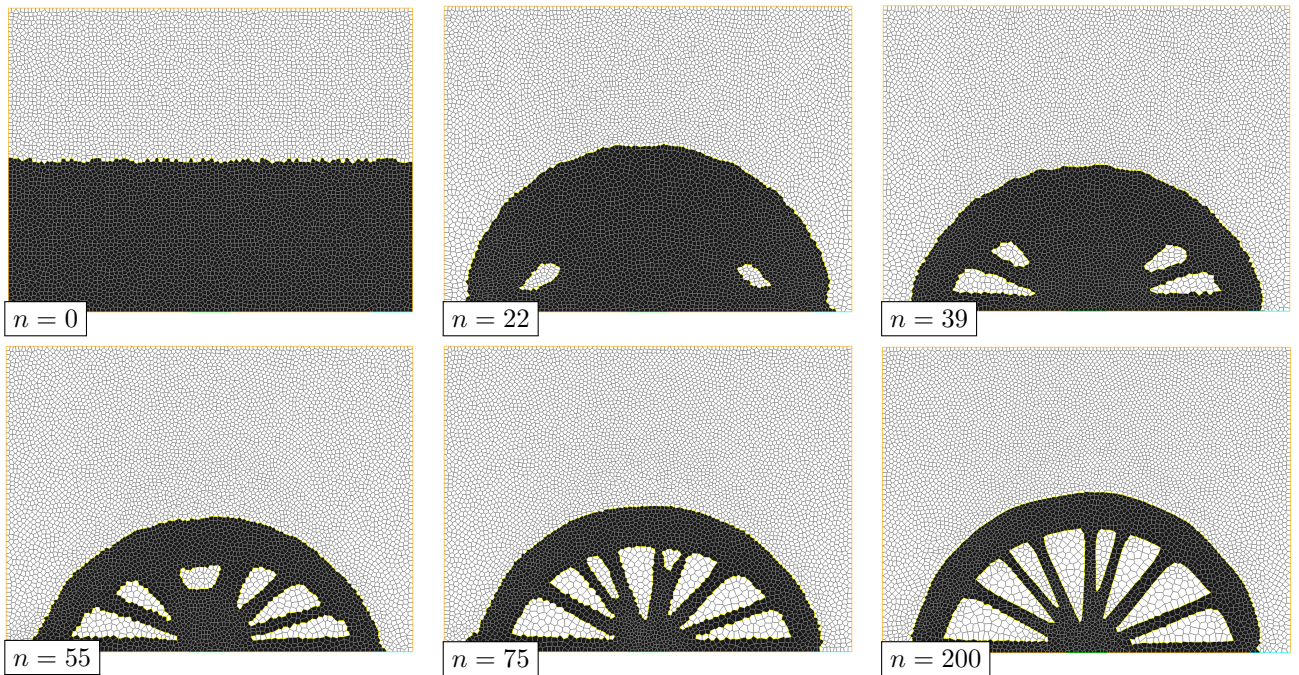


FIGURE 20. Various iterates in the solution of the bridge optimization problem of [Section 7.5](#).

7.6. Minimization of the stress within a bridge structure

Our last numerical example deals with the minimization of the stress within a two-dimensional self-sustaining bridge, which is depicted in [Fig. 14](#) (d): the shapes are contained in a square-shaped box D with size 1×1 ; they are clamped on their bottom side Γ_D , and a uniform vertical load $\mathbf{g} = (0, -1)$ is applied on their upper side Γ_N , which stands for the deck of the structure. Omitting body forces, the displacement $\mathbf{u}_\Omega : \Omega \rightarrow \mathbb{R}^2$ of the structure is characterized by the boundary value problem [\(7.5\)](#).

In this setting, we aim to minimize an integral quantity of the stress within the structure, i.e. we solve:

$$(7.7) \quad \min_{\Omega} S(\Omega) \text{ s.t. } \text{Vol}(\Omega) = V_T.$$

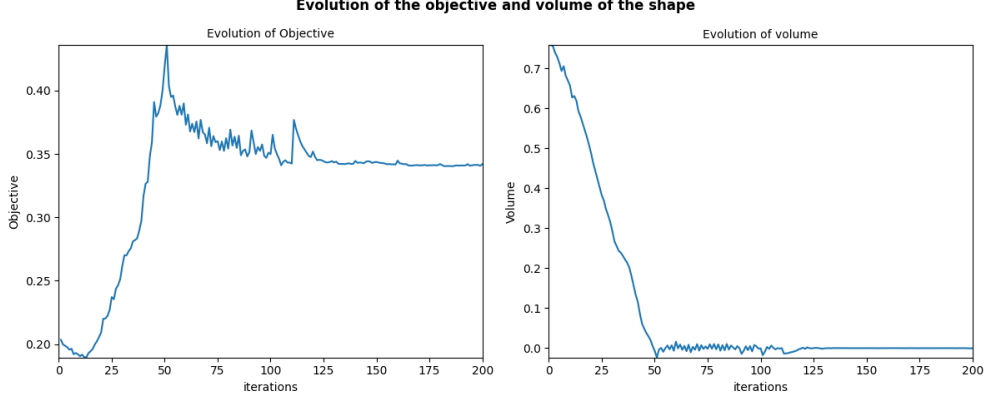


FIGURE 21. *Evolution of the objective (mechanical compliance) and volume during the solution of the bridge optimization problem of Section 7.5.*

In this formulation, the stress function $S(\Omega)$ is defined by:

$$S(\Omega) = \int_{\Omega} \|\sigma(\mathbf{u}_{\Omega})\|^2 \, d\mathbf{x}, \text{ where } \sigma(\mathbf{u}_{\Omega}) := A\epsilon(\mathbf{u}_{\Omega}),$$

and the volume target V_T is set to 0.8.

We address the solution of (7.7) with the two-phase Laguerre diagram discretization presented in Section 3.1 and used in the previous examples Sections 7.4 and 7.5. Here again, the sensitivities of the objective function with respect to the seed points \mathbf{s} and cell measures ν defining the diagrams at stake are calculated by the optimize-then-discretize approach presented in Section 6.1. From the implementation viewpoint, the main difference between this example and those addressed in the previous Sections 7.4 and 7.5 is that the optimization problem is no longer self-adjoint, i.e. the shape derivative of $S(\Omega)$ involves an adjoint state \mathbf{p}_{Ω} which is not easily related to \mathbf{u}_{Ω} , as revealed by the following result, about which we refer again to e.g. [7].

Proposition 7.3. *The function $S(\Omega)$ is shape differentiable at any smooth enough shape Ω , and its derivative reads, for any vector field $\boldsymbol{\theta}$ vanishing on $\Gamma_D \cup \Gamma_N$:*

$$(7.8) \quad S'(\Omega)(\boldsymbol{\theta}) = \int_{\Omega} \operatorname{div}(\boldsymbol{\theta}) A\epsilon(\mathbf{u}_{\Omega}) : e(\mathbf{p}_{\Omega}) \, d\mathbf{x} - 2\mu \int_{\Omega} \left((\nabla \mathbf{u}_{\Omega} \nabla \boldsymbol{\theta}) : e(\mathbf{p}_{\Omega}) + (\nabla \mathbf{p}_{\Omega} \nabla \boldsymbol{\theta}) : e(\mathbf{u}_{\Omega}) \right) \, d\mathbf{x} \\ - \lambda \int_{\Omega} \left(\operatorname{tr}(\nabla \mathbf{u}_{\Omega} \nabla \boldsymbol{\theta}) \operatorname{div}(\mathbf{p}_{\Omega}) + \operatorname{tr}(\nabla \mathbf{p}_{\Omega} \nabla \boldsymbol{\theta}) \operatorname{div}(\mathbf{u}_{\Omega}) \right) \, d\mathbf{x},$$

where the adjoint state \mathbf{p}_{Ω} is the solution to the following variational problem:

$$\text{Search for } \mathbf{p}_{\Omega} \in H_{\Gamma_D}^1(\Omega)^2 \text{ s.t. } \forall \mathbf{v} \in H_{\Gamma_D}^1(\Omega)^2, \quad \int_{\Omega} A\epsilon(\mathbf{p}_{\Omega}) : e(\mathbf{v}) \, d\mathbf{x} = -2 \int_{\Omega} A\epsilon(\mathbf{u}_{\Omega}) : A\epsilon(\mathbf{v}) \, d\mathbf{x}.$$

We apply 250 iterations of our shape and topology optimization Algorithm 1 to solve the problem (7.7). The average number N of cells of the considered Laguerre diagrams is 8000, and the total computation takes about 3 hours. We note that the calculation of the adjoint state \mathbf{p}_{Ω} requires the solution to the boundary value problem (7.8) for \mathbf{p}_{Ω} at each iteration of the process, in addition to that of (7.5), which explains the significant increase in computational burden observed in this case. A few snapshots of the optimization process are reported on Fig. 22. Again, the shape dramatically changes topology in the course of the solution process.

8. CONCLUSIONS AND PERSPECTIVES

In this article, we have presented a novel numerical framework for optimal design, inspired by modern concepts from fields so diverse as optimal transport, computational geometry, and shape and topology optimization. The key feature of our method is a consistent body-fitted representation of the shape, in terms of a (modified version of a) Laguerre diagram. The latter is parametrized in terms of the seed points and

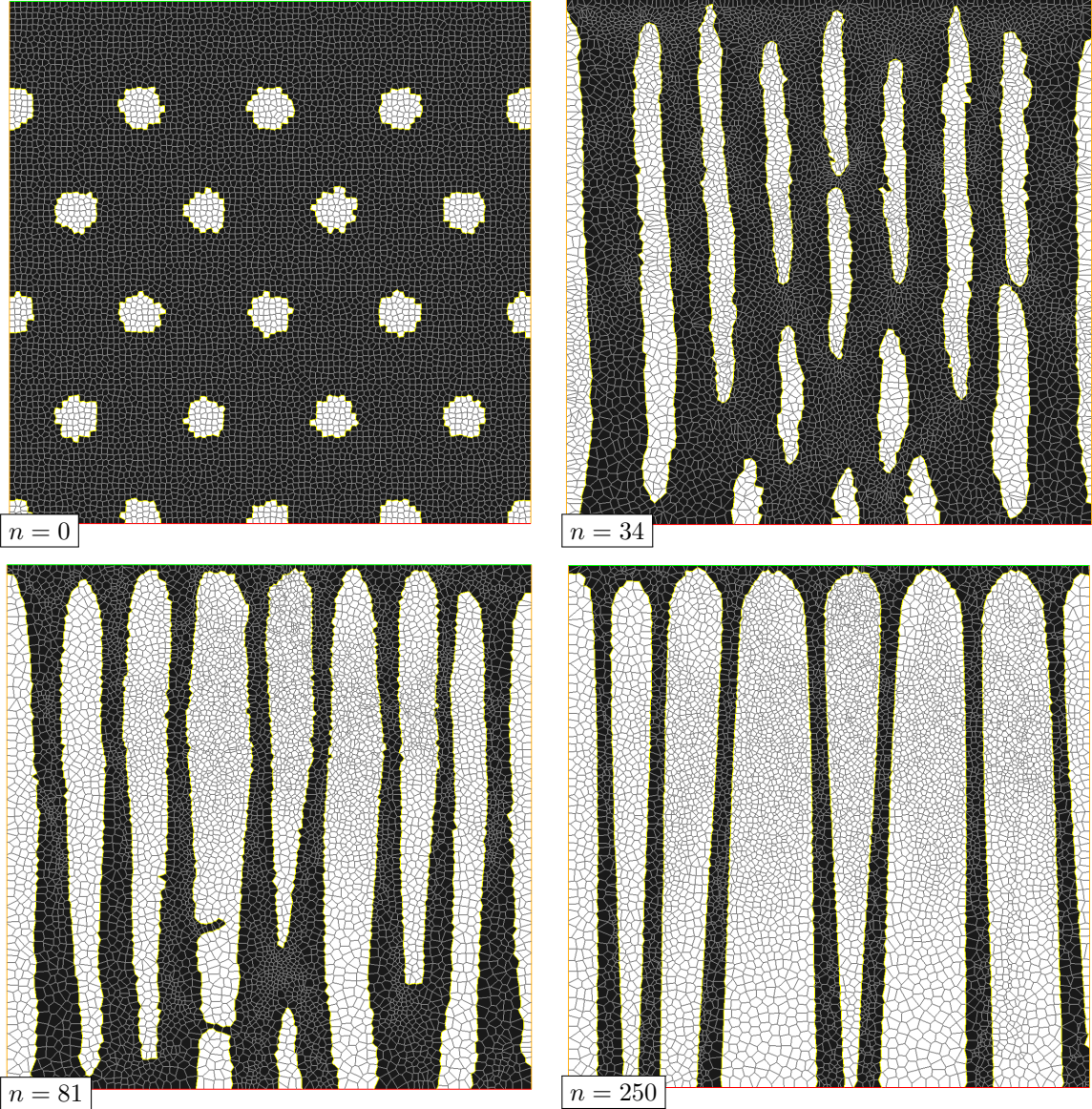


FIGURE 22. Various iterates in the solution of the stress minimization problem within a self-sustaining bridge of [Section 7.6](#).

the measures of its cells, as permitted by deep results from optimal transport theory. The evolution of the shape through the iterations of the process is driven by the shape gradients of the objective and constraint functionals, which are suitably expressed in terms of these defining variables of Laguerre diagrams. Our numerical strategy is Lagrangian in nature, since the evolution of the shape is tracked via the motion of the seed points and weights of the diagram. However, dramatic updates can be accounted for in a robust fashion, including topological changes, since the diagram is never deformed itself – it is rediscovered from the updated seed points and cell measures. This framework can generally handle complex physical situations, featuring multiple phases, and we hope that it can serve to other applications. Remarkably, it features a description of a wide variety of shapes by a relatively small number of parameters, which pleads in favor of the use of such representation in connection with reduced basis methods, or of the use of neural networks for a parametrization of shapes by such a diagram, and the realization of its optimization with machine learning techniques.

For the sake of simplicity, we have focused our presentation and numerical illustrations on the case of two space dimensions, although the underlying theory holds true in 3d, and the critical numerical ingredients of the method (notably, the computation of Laguerre diagrams) are already available in this context. In future work, we ambition to adapt this numerical methodology to this theoretically similar, albeit numerically much more involved (and much richer) 3d context.

On the longer term, and from a more theoretical viewpoint, it would be fascinating to appraise more rigorously the new type of shape evolution featured by this method. Indeed, the examples of [Sections 7.1](#) and [7.4](#) have demonstrated how different the induced notion of sensitivity with respect to the design is from more “classical” concepts such as shape and topological derivatives: we suspect that these issues are somehow related to the idea of linearized optimal transport, see [Remark 3.5](#).

Acknowledgements. The work of C.D. and E.O. is partially supported by the projects [ANR-18-CE40-0013 SHAPO](#), [ANR-22-CE46-0006 StableProxies](#) and [ANR-24-CE40-2216 STOIQUES](#), financed by the French Agence Nationale de la Recherche (ANR). E.O. was also supported by the “Institut Universitaire de France”. The work of B.L. is supported by the COSMOGRAM-launchpad Inria Action Exploratoire grant.

APPENDIX A. CALCULATION AND DIFFERENTIATION OF THE COORDINATES OF THE VERTICES OF A DIAGRAM WITH RESPECT TO ITS SEED POINTS AND WEIGHTS

In this appendix, we detail the calculation of the positions of the vertices $\mathbf{q} = \{\mathbf{q}_j\}_{j=1,\dots,M} \in \mathbb{R}^{dM}$ of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ in [\(3.6\)](#) from the datum of its seed points \mathbf{s} and weights $\boldsymbol{\psi}$. We also investigate the calculation of their derivatives with respect to \mathbf{s} and $\boldsymbol{\psi}$, which is a central ingredient in the procedure of [Section 6.2](#) for expressing differentiation of functions with respect to the positions \mathbf{q} of the vertices of the diagram in terms of the generating seed points \mathbf{s} and weights $\boldsymbol{\psi}$. These formulas are admittedly not new, but their rigorous exposition and establishment are not so easily found in the literature. Our presentation focuses on the case $d = 2$, but similar analyses could be conducted in three space dimensions.

Let us start with the following general result.

Proposition A.1. *Let $\mathbf{s} \in \mathbb{R}^{dN}$ and $\boldsymbol{\psi} \in \mathbb{R}^N$ be sets of seed points and weights satisfying the genericity assumptions [\(G1\)](#) to [\(G7\)](#) and let $\mathbf{q} = \{\mathbf{q}_j\}_{j=1,\dots,M} \in \mathbb{R}^{dM}$ denote the vertices of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$. Then for $\widehat{\mathbf{s}} \in \mathbb{R}^{dN}$ and $\widehat{\boldsymbol{\psi}} \in \mathbb{R}^N$ small enough, the perturbed diagram $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}})$ has the same neighbour relations as $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$.*

Moreover, there exists a mapping $\mathbf{m} : \mathbb{R}_s^{dN} \times \mathbb{R}_\psi^N \rightarrow W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ which is differentiable in a neighborhood of $(\mathbf{0}, \mathbf{0})$ such that:

$$\forall i = 1, \dots, N, \quad \mathbf{m}(\widehat{\mathbf{s}}, \widehat{\boldsymbol{\psi}}) \left(V_i(\mathbf{s}, \boldsymbol{\psi}) \right) = V_i(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}}),$$

and

The vertices of the diagram $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}})$ are exactly the points $\mathbf{m}(\widehat{\mathbf{s}}, \widehat{\boldsymbol{\psi}})(\mathbf{q}_j)$, $j = 1, \dots, M$.

Proof. At first, we prove that $i \neq j \in \{1, \dots, N\}$ are the indices of neighbor cells in the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ if and only if the cells i and j are neighbors in $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}})$ for small enough perturbations $\widehat{\mathbf{s}} \in \mathbb{R}^{dN}$ and $\widehat{\boldsymbol{\psi}} \in \mathbb{R}^N$. For simplicity, we only deal with the case where the edge \mathbf{e}_{ij} between the cells $V_i(\mathbf{s}, \boldsymbol{\psi})$ and $V_j(\mathbf{s}, \boldsymbol{\psi})$ in $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ lies entirely in the open set D , the treatment of the situation where it intersects ∂D being similar.

To achieve our purpose, let us first observe that $V_i(\mathbf{s}, \boldsymbol{\psi})$ and $V_j(\mathbf{s}, \boldsymbol{\psi})$ are neighbors in the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ if and only if the following property holds true:

$$\exists \mathbf{x} \in D \text{ s.t. } \begin{cases} |\mathbf{x} - \mathbf{s}_i|^2 - \psi_i = |\mathbf{x} - \mathbf{s}_j|^2 - \psi_j, \\ |\mathbf{x} - \mathbf{s}_i|^2 - \psi_i < |\mathbf{x} - \mathbf{s}_k|^2 - \psi_k, \quad \text{for each } k \notin \{i, j\}. \end{cases}$$

In turn, this is equivalent to the positivity of the optimal value

$$\max_{\mathbf{x}} \left\{ \min_{k \notin \{i, j\}} \left\{ |\mathbf{x} - \mathbf{s}_k|^2 - \psi_k - \left(|\mathbf{x} - \mathbf{s}_i|^2 - \psi_i \right) \right\}, \mathbf{x} \in D, |\mathbf{x} - \mathbf{s}_i|^2 - \psi_i = |\mathbf{x} - \mathbf{s}_j|^2 - \psi_j \right\}.$$

We now invoke the continuity of the optimal value of a constrained optimization program with respect to perturbations of its parameters, see e.g. Prop. 4.4 in [\[23\]](#), whose assumptions are satisfied in the present

case. Hence, if the cells indexed by $i \neq j$ are neighbors in the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$, then they are also neighbors in the diagram $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}})$ for small enough $\widehat{\mathbf{s}} \in \mathbb{R}^{dN}$, $\widehat{\boldsymbol{\psi}} \in \mathbb{R}^N$. Note that a similar argument shows that if $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ satisfies (G1) to (G7), then so does $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}})$.

Conversely, let us now prove that there exists $\varepsilon > 0$ such that, for any perturbations $\widehat{\mathbf{s}} \in \mathbb{R}^{dN}$, $\widehat{\boldsymbol{\psi}} \in \mathbb{R}^N$ with $|\widehat{\mathbf{s}}| + |\widehat{\boldsymbol{\psi}}| < \varepsilon$, the diagram $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}})$ cannot have neighbor cells that are not already neighbors in $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$. To achieve this, we proceed by contradiction, assuming that there exist sequences $\widehat{\mathbf{s}}^n \in \mathbb{R}^{dN}$, $\widehat{\boldsymbol{\psi}}^n \in \mathbb{R}^N$ converging to $\mathbf{0}$ and index sequences $i^n \neq j^n \in \{1, \dots, N\}$ such that the cells with indices i^n , j^n are neighbors in $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}^n, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}}^n)$ but not in $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$. Actually, since the number N of seed points is finite, we may extract a subsequence (still labeled by n) along which these indices i, j are independent of n . Let then \mathbf{a}^n and \mathbf{b}^n denote the vertices of the edge between the cells i and j in $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}^n, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}}^n)$, and let $k^n, l^n \notin \{i, j\}$ denote the (distinct) indices such that \mathbf{a}^n (resp. \mathbf{b}^n) is at the intersection between the cells i, j and k^n (resp. l^n) in $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}^n, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}}^n)$. Up to extraction of another subsequence, we may assume that these indices do not depend on n , and we denote them by $k \neq l$. We may also assume that \mathbf{a}^n and \mathbf{b}^n converge to limiting positions \mathbf{a}^* and \mathbf{b}^* , respectively. Both points belong to the boundaries $\partial V_i(\mathbf{s}, \boldsymbol{\psi})$ and $\partial V_j(\mathbf{s}, \boldsymbol{\psi})$. Since $V_i(\mathbf{s}, \boldsymbol{\psi})$ and $V_j(\mathbf{s}, \boldsymbol{\psi})$ are not neighbors in $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$, one must then have $\mathbf{a}^* = \mathbf{b}^*$ and this single vertex is at the intersection between the four cells $V_i(\mathbf{s}, \boldsymbol{\psi})$, $V_j(\mathbf{s}, \boldsymbol{\psi})$, $V_k(\mathbf{s}, \boldsymbol{\psi})$ and $V_l(\mathbf{s}, \boldsymbol{\psi})$. This contradicts Assumption (G2), and the statement that i and j are not neighbors in $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ is absurd.

At this point, we have proved that for $\widehat{\mathbf{s}} \in \mathbb{R}^{dN}$, $\widehat{\boldsymbol{\psi}} \in \mathbb{R}^N$ small enough, the perturbed diagram $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}})$ has the same neighbor relations as $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$. Now, each cell $V_i(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}})$ is a closed, convex subset of \overline{D} whose non degenerate edges are completely characterized by their endpoints. The calculations below show that all the vertices of this diagram behave as differentiable functions in a neighborhood of $(\mathbf{s}, \boldsymbol{\psi})$. We may then define $\mathbf{m}(\widehat{\mathbf{s}}, \widehat{\boldsymbol{\psi}})$ as an affine function in restriction to every edge of the cells $V_i(\mathbf{s}, \boldsymbol{\psi})$, and then extend the latter into D as a whole by barycentric coordinates extension. This induces a $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ mapping $\mathbf{m}(\widehat{\mathbf{s}}, \widehat{\boldsymbol{\psi}})$ which is a differentiable of the variables $(\mathbf{s}, \boldsymbol{\psi})$ in the neighborhood of $(\mathbf{0}, \mathbf{0}) \in \mathbb{R}^{dN} \times \mathbb{R}^N$. \square

We provide the missing ingredients of the above proof by showing that the vertices of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ are smooth functions of the seed points \mathbf{s} and weights $\boldsymbol{\psi}$ when they are slight perturbations of reference values satisfying (G1) to (G7). Meanwhile, we provide explicit characterizations of their derivatives, which are useful in particular in the numerical implementation of the methods presented in Section 6. We proceed in the case $d = 2$ for simplicity, but a similar analysis could be conducted when $d = 3$, up to an increased level of tediousness. We distinguish between several types of vertices, which are illustrated on Fig. 23.

Case 1: Vertices at the intersection of three cells. Let $\mathbf{q} \in \mathbb{R}^2$ be the vertex at the intersection between three given cells $V_i(\mathbf{s}, \boldsymbol{\psi})$, $V_j(\mathbf{s}, \boldsymbol{\psi})$ and $V_k(\mathbf{s}, \boldsymbol{\psi})$, for some distinct indices $i, j, k \in \{1, \dots, N\}$. This vertex is characterized by the following relations:

$$\begin{cases} |\mathbf{q} - \mathbf{s}_i|^2 - \psi_i &= |\mathbf{q} - \mathbf{s}_j|^2 - \psi_j \\ |\mathbf{q} - \mathbf{s}_i|^2 - \psi_i &= |\mathbf{q} - \mathbf{s}_k|^2 - \psi_k, \end{cases}$$

which rewrites:

$$\begin{cases} \langle \mathbf{q}, \mathbf{s}_j - \mathbf{s}_i \rangle &= \frac{1}{2}(|\mathbf{s}_j|^2 - |\mathbf{s}_i|^2 - (\psi_j - \psi_i)) \\ \langle \mathbf{q}, \mathbf{s}_k - \mathbf{s}_i \rangle &= \frac{1}{2}(|\mathbf{s}_k|^2 - |\mathbf{s}_i|^2 - (\psi_k - \psi_i)). \end{cases}$$

Invoking Assumptions (G2) and (G3), this system uniquely determines \mathbf{q} as a smooth function of $\mathbf{s}_i, \mathbf{s}_j, \mathbf{s}_k$ and ψ_i, ψ_j, ψ_k , a dependence which is not made explicit for notational simplicity. Taking derivatives in the previous system then leads to the following 2×2 linear systems for the sensitivities $[\nabla_{\mathbf{s}_i} \mathbf{q}] \in \mathbb{R}^{2 \times 2}$ and $\frac{\partial \mathbf{q}}{\partial \psi_i} \in \mathbb{R}^2$:

$$\begin{pmatrix} \mathbf{s}_j - \mathbf{s}_i \\ \mathbf{s}_k - \mathbf{s}_i \end{pmatrix} [\nabla_{\mathbf{s}_i} \mathbf{q}] = \begin{pmatrix} \mathbf{q} - \mathbf{s}_i \\ \mathbf{q} - \mathbf{s}_i \end{pmatrix}, \text{ and } \begin{pmatrix} \mathbf{s}_j - \mathbf{s}_i \\ \mathbf{s}_k - \mathbf{s}_i \end{pmatrix} \frac{\partial \mathbf{q}}{\partial \psi_i} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix},$$

which are invertible on account of Assumption (G2). Similar expressions hold true for the derivatives $[\nabla_{\mathbf{s}_j} \mathbf{q}]$, $[\nabla_{\mathbf{s}_k} \mathbf{q}]$ and $\frac{\partial \mathbf{q}}{\partial \psi_j}, \frac{\partial \mathbf{q}}{\partial \psi_k}$.

Case 2: Vertices at the intersection of two cells and void. Let now $\mathbf{q} \in \mathbb{R}^2$ be one vertex at the intersection between the cells $V_i(\mathbf{s}, \boldsymbol{\psi})$, $V_j(\mathbf{s}, \boldsymbol{\psi})$ ($i \neq j$) and the void phase $V_0(\mathbf{s}, \boldsymbol{\psi})$. Then, \mathbf{q} satisfies the following

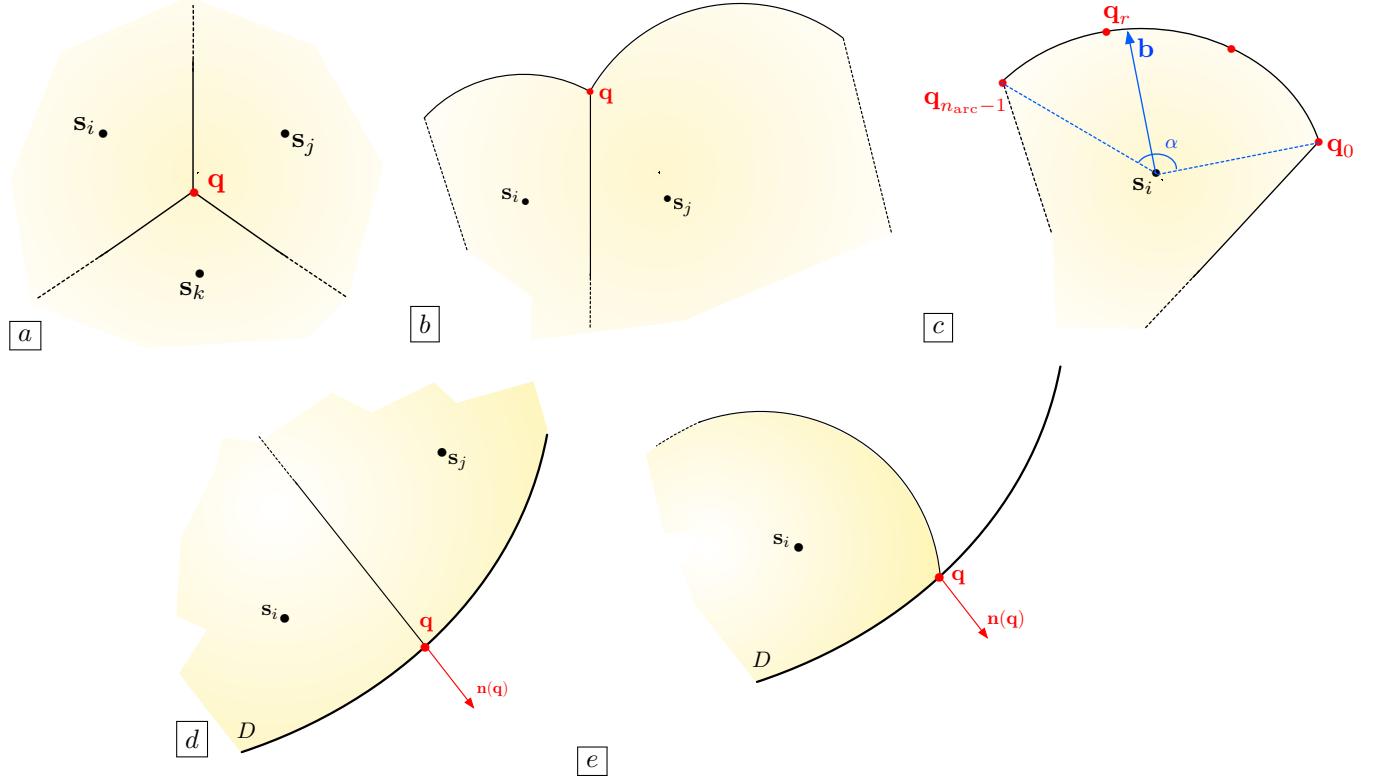


FIGURE 23. Different categories of vertices in a diagram $\mathbf{V}(\mathbf{s}, \psi)$: (a) Case of a vertex \mathbf{q} at the intersection between 3 cells; (b) Case of a vertex between 2 cells and the void phase; (c) Case of a vertex resulting from the discretization of an exterior circle; (d) Case of a vertex at the intersection between 2 cells and the boundary ∂D of the computational domain D ; (e) Case of a vertex at the intersection between one cell, the void phase, and ∂D .

system of equations:

$$\begin{cases} \langle \mathbf{q}, \mathbf{s}_j - \mathbf{s}_i \rangle &= \frac{1}{2}(|\mathbf{s}_j|^2 - |\mathbf{s}_i|^2 - (\psi_j - \psi_i)) \\ |\mathbf{q} - \mathbf{s}_i|^2 &= \psi_i. \end{cases}$$

Note that there may be several such vertices, and so the above system may accordingly have multiple solutions. Assumption (G3) and the implicit function theorem together imply that \mathbf{q} behaves as a smooth function of \mathbf{s}_i , \mathbf{s}_j , ψ_i and ψ_j – actually, any solution to this system induces a smooth branch of solutions for small perturbations of these parameters. Taking derivatives, it follows:

$$\begin{pmatrix} \mathbf{s}_j - \mathbf{s}_i \\ \mathbf{q} - \mathbf{s}_i \end{pmatrix} [\nabla_{\mathbf{s}_i} \mathbf{q}] = \begin{pmatrix} \mathbf{q} - \mathbf{s}_i \\ \mathbf{q} - \mathbf{s}_i \end{pmatrix}, \text{ and } \begin{pmatrix} \mathbf{s}_j - \mathbf{s}_i \\ \mathbf{q} - \mathbf{s}_i \end{pmatrix} \frac{\partial \mathbf{q}}{\partial \psi_i} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}.$$

These systems are invertible on account of Assumption (G3). Again, similar expressions hold for the derivatives $[\nabla_{\mathbf{s}_j} \mathbf{q}]$ and $\frac{\partial \mathbf{q}}{\partial \psi_j}$.

Case 3: Vertices corresponding to the discretization of a circular arc. Consider a circular arc pertaining to the boundary of the cell $V_i(\mathbf{s}, \psi)$, and let \mathbf{q}_r , $r = 0, \dots, n_{\text{arc}}$ be the vertices corresponding to its discretization, enumerated counterclockwise, see Section 4. Let $\alpha \in (0, 2\pi)$ be the angle from $\mathbf{s}_i \mathbf{q}_0$ to $\mathbf{s}_i \mathbf{q}_{n_{\text{arc}}}$, which is a smooth function of \mathbf{s}_i , \mathbf{q}_0 and $\mathbf{q}_{n_{\text{arc}}}$, and, in turn, as revealed by the previous item, a smooth function of the seed points \mathbf{s} and weights ψ . More precisely, it holds:

$$\cos \alpha = \left\langle \frac{\mathbf{s}_i \mathbf{q}_0}{|\mathbf{s}_i \mathbf{q}_0|}, \frac{\mathbf{s}_i \mathbf{q}_{n_{\text{arc}}}}{|\mathbf{s}_i \mathbf{q}_{n_{\text{arc}}}|} \right\rangle, \quad \sin \alpha = \det \begin{pmatrix} \frac{\mathbf{s}_i \mathbf{q}_0}{|\mathbf{s}_i \mathbf{q}_0|}, \frac{\mathbf{s}_i \mathbf{q}_{n_{\text{arc}}}}{|\mathbf{s}_i \mathbf{q}_{n_{\text{arc}}}|} \end{pmatrix},$$

whence we infer $\alpha \in (0, 2\pi)$ via the formula

$$\alpha = 2\text{acot}\left(\frac{1 + \cos \alpha}{\sin \alpha}\right) = \pi - 2\text{atan}\left(\frac{1 + \cos \alpha}{\sin \alpha}\right).$$

Then, let us introduce the vector $\mathbf{b} = \mathbf{s}_i \mathbf{q}_0^\perp := (-(s_{i,2} - q_{0,2}), s_{i,1} - q_{0,1})$, where $\mathbf{u}^\perp := (-u_2, u_1)$ denotes the 90° counterclockwise rotate of a vector $\mathbf{u} = (u_1, u_2) \in \mathbb{R}^2$. Each vertex \mathbf{q}_r ($r = 1, \dots, n_{\text{arc}} - 1$) is determined by the following relation:

$$\mathbf{s}_i \mathbf{q}_r = \cos(t_r \alpha) \mathbf{s}_i \mathbf{q}_0 + \sin(t_r \alpha) \mathbf{b}, \text{ where } t_r = \frac{r}{n_{\text{arc}}}.$$

The differentiability of \mathbf{q}_r and the closed form expressions of its derivatives with respect to \mathbf{s} and $\boldsymbol{\psi}$ are straightforward consequences of this formula.

Since the cells of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ are restricted to the bounded computational domain $D \subset \mathbb{R}^2$, two additional situations should be considered, which are depicted on [Fig. 23](#) (e) (f). To address them, it is convenient to introduce a level set function $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}$ for D , that is

$$\forall \mathbf{x} \in \mathbb{R}^2, \quad \begin{cases} \phi(\mathbf{x}) < 0 & \text{if } \mathbf{x} \in D, \\ \phi(\mathbf{x}) = 0 & \text{if } \mathbf{x} \in \partial D, \\ \phi(\mathbf{x}) > 0 & \text{otherwise.} \end{cases}$$

Then, the unit normal vector $\mathbf{n}(\mathbf{x})$ to ∂D pointing outward D is given by $\mathbf{n}(\mathbf{x}) = \frac{\nabla \phi(\mathbf{x})}{|\nabla \phi(\mathbf{x})|}$ for ds a.e. $\mathbf{x} \in \partial D$, see e.g. [\[100, 115\]](#) about general features of such a level set representation of domains.

Case 2b: Vertices corresponding to the intersection between two cells and ∂D . Let $\mathbf{q} \in \mathbb{R}^2$ be one vertex at the intersection between the cells $V_i(\mathbf{s}, \boldsymbol{\psi})$, $V_j(\mathbf{s}, \boldsymbol{\psi})$ and ∂D . Then the following system of equations holds:

$$\begin{cases} |\mathbf{q} - \mathbf{s}_i|^2 - \psi_i & = & |\mathbf{q} - \mathbf{s}_j|^2 - \psi_j \\ \phi(\mathbf{q}) & = & 0. \end{cases}$$

As in the previous items, the combination of Assumptions (G5) and (G6) with the implicit function theorem shows that this \mathbf{q} behaves as a smooth function of the parameters \mathbf{s} and $\boldsymbol{\psi}$ when the latter are slightly perturbed from reference values satisfying (G1) to (G7). In the meantime, taking derivatives in the above equations yields immediately the following systems:

$$\begin{pmatrix} \mathbf{s}_j - \mathbf{s}_i \\ \mathbf{n}(\mathbf{q}) \end{pmatrix} [\nabla_{\mathbf{s}_i} \mathbf{q}] = \begin{pmatrix} \mathbf{q} - \mathbf{s}_i \\ \mathbf{0} \end{pmatrix}, \text{ and } \begin{pmatrix} \mathbf{s}_j - \mathbf{s}_i \\ \mathbf{n}(\mathbf{q}) \end{pmatrix} \frac{\partial \mathbf{q}}{\partial \psi_i} = \begin{pmatrix} \frac{1}{2} \\ 0 \end{pmatrix},$$

which are invertible on account of (G5) and (G6). Similar relations allow to characterize the derivatives $[\nabla_{\mathbf{s}_j} \mathbf{q}]$ and $\frac{\partial \mathbf{q}}{\partial \psi_j}$.

Case 3b: Vertices at the intersection between one exterior circle and ∂D . Let $\mathbf{q} \in \mathbb{R}^2$ be one vertex lying at the intersection between the cell $V_i(\mathbf{s}, \boldsymbol{\psi})$ ($i = 1, \dots, N$), the void phase $V_0(\mathbf{s}, \boldsymbol{\psi})$ and ∂D . Then,

$$\begin{cases} |\mathbf{q} - \mathbf{s}_i|^2 - \psi_i & = & 0 \\ \phi(\mathbf{q}) & = & 0, \end{cases}$$

Using (G1) to (G7) and the implicit function theorem, we see that this system characterizes a smooth function of \mathbf{s} and $\boldsymbol{\psi}$. In the meantime, taking derivatives into these equations, we obtain the following invertible systems:

$$\begin{pmatrix} \mathbf{q} - \mathbf{s}_i \\ \mathbf{n}(\mathbf{q}) \end{pmatrix} [\nabla_{\mathbf{s}_i} \mathbf{q}] = \begin{pmatrix} \mathbf{q} - \mathbf{s}_i \\ \mathbf{0} \end{pmatrix}, \text{ and } \begin{pmatrix} \mathbf{q} - \mathbf{s}_i \\ \mathbf{n}(\mathbf{q}) \end{pmatrix} \frac{\partial \mathbf{q}}{\partial \psi_i} = \begin{pmatrix} \frac{1}{2} \\ 0 \end{pmatrix}.$$

APPENDIX B. SECOND ORDER DERIVATIVES OF THE [KANTOROVICH](#) FUNCTIONAL WITH RESPECT TO SEEDS AND WEIGHTS

Let $\mathbf{s} \in \mathbb{R}^{dN}$, $\boldsymbol{\psi} \in \mathbb{R}^N$ be generic seed points and weights, in the sense that (G1) to (G7) hold true. Let also $\boldsymbol{\nu} = \{\nu_1, \dots, \nu_N\}$ be a vector of prescribed measures. Restricting to the case of two space dimensions

$d = 2$ for simplicity, this section details the calculation of the partial derivatives of the derivative $\mathbf{F} : \mathbb{R}_s^{dN} \times \mathbb{R}_\nu^N \times \mathbb{R}_\psi^N \rightarrow \mathbb{R}^N$ of the Kantorovich functional introduced in [Theorem 3.1](#):

$$\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) = \left(\nu_i - |V_i(\mathbf{s}, \boldsymbol{\psi})| \right)_{i=1, \dots, N}.$$

Let us recall that this result is useful for a variety of purposes in our framework: on the one hand, it is the key ingredient of the Newton-Raphson algorithm implemented for the resolution of [\(3.11\)](#), see [Section 4.1](#). On the other hand, it is needed in the procedure of [Section 6](#) for expressing the derivative of a quantity with respect to the vertices \mathbf{q} of the diagram in terms of the defining seed points \mathbf{s} and weights $\boldsymbol{\psi}$. Again, the results of this appendix are not completely new, see e.g. [\[44, 88, 98\]](#) where fairly similar calculations are conducted in a formal fashion. The calculation of the derivatives of \mathbf{F} with respect to the weights $\boldsymbol{\psi}$ is also achieved in [\[88\]](#), and in [\[47\]](#) in a more general context. Here, we present an elementary mixture of arguments used in the aforementioned references.

Let us introduce a few additional notations, which are illustrated on [Fig. 24](#):

- For any index $i = 1, \dots, N$ and any neighbor $j \in \mathcal{N}_i$ of i , we recall that \mathbf{e}_{ij} is the line segment $V_i(\mathbf{s}, \boldsymbol{\psi}) \cap V_j(\mathbf{s}, \boldsymbol{\psi})$, and we denote by \mathbf{m}_{ij} the midpoint of \mathbf{e}_{ij} .
- For $i = 1, \dots, N$, we denote by $\mathcal{C}_{i,r}$, $r = 1, \dots, n_{c,i}$ the circular arcs composing the boundary of the cell $V_i(\mathbf{s}, \boldsymbol{\psi})$; the value $n_{c,i} = 0$ indicates that the cell is not on the boundary of the associated shape Ω . We also let $\mathcal{C}_i := \bigcup_{r=1}^{n_{c,i}} \mathcal{C}_{i,r}$.
- For $i = 1, \dots, N$ and $r = 1, \dots, n_{c,i}$, the endpoints of the circular arc $\mathcal{C}_{i,r}$ are denoted by $\mathbf{q}_{i,r}^0, \mathbf{q}_{i,r}^1$ when the latter is oriented counterclockwise, and $\theta_{i,r} \in (0, 2\pi)$ is its angular aperture.
- For $i = 1, \dots, N$, we denote by \mathbf{n}_i the unit normal vector field to $\partial V_i(\mathbf{s}, \boldsymbol{\psi})$, pointing outward $V_i(\mathbf{s}, \boldsymbol{\psi})$. This latter is given for ds-almost every point $\mathbf{x} \in \partial V_i(\mathbf{s}, \boldsymbol{\psi})$ by the following expressions:

$$\mathbf{n}(\mathbf{x}) = \begin{cases} \frac{\mathbf{s}_j - \mathbf{s}_i}{|\mathbf{s}_j - \mathbf{s}_i|} & \text{if } \mathbf{x} \in \mathbf{e}_{ij} \text{ for some } j \in \mathcal{N}_i, \\ \frac{\mathbf{x} - \mathbf{s}_i}{|\mathbf{x} - \mathbf{s}_i|} & \text{if } \mathbf{x} \in \mathcal{C}_i. \end{cases}$$

The result of interest in this appendix is the following.

Proposition B.1. *Let $\mathbf{s} = \{\mathbf{s}_i\}_{i=1, \dots, N} \in \mathbb{R}^{dN}$ and $\boldsymbol{\psi} = \{\psi_i\}_{i=1, \dots, N} \in \mathbb{R}^N$ be sets of seed points and weights satisfying the genericity Assumptions (G1) to (G7). Then, for all $i = 1, \dots, N$, the function \mathbf{F} is differentiable at $(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$. It holds, for $i = 1, \dots, N$:*

$$\nabla_{\mathbf{s}_i} F_i(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) = - \sum_{j \in \mathcal{N}_i} \frac{|\mathbf{e}_{ij}|}{|\mathbf{s}_j - \mathbf{s}_i|} (\mathbf{m}_{ij} - \mathbf{s}_i) - \sum_{r=1}^{n_{c,i}} (\mathbf{q}_{i,r}^0 - \mathbf{q}_{i,r}^1)^\perp, \text{ and for } j \neq i \quad \nabla_{\mathbf{s}_j} F_i(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) = \frac{|\mathbf{e}_{ij}|}{|\mathbf{s}_j - \mathbf{s}_i|} (\mathbf{m}_{ij} - \mathbf{s}_i),$$

where $\mathbf{u}^\perp := (-u_2, u_1)$ is the 90° counterclockwise rotate of a vector $\mathbf{u} = (u_1, u_2) \in \mathbb{R}^2$. Besides,

$$\frac{\partial F_i}{\partial \psi_i}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) = -\frac{1}{2} \left(\sum_{j \in \mathcal{N}_i} \frac{|\mathbf{e}_{ij}|}{|\mathbf{s}_j - \mathbf{s}_i|} \right) - \sum_{r=1}^{n_{c,i}} \theta_{i,r}, \text{ and for } j \neq i \quad \frac{\partial F_i}{\partial \psi_j}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) = \frac{1}{2} \frac{|\mathbf{e}_{ij}|}{|\mathbf{s}_j - \mathbf{s}_i|}.$$

Proof. Judging from the definition [\(3.14\)](#) of the function $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \cdot)$, the proof boils down to the calculation of the derivatives of the measures

$$A_i(\mathbf{s}, \boldsymbol{\psi}) := |V_i(\mathbf{s}, \boldsymbol{\psi})|, \quad i = 1, \dots, N$$

with respect to the seed points $\mathbf{s} \in \mathbb{R}^{dN}$ and weights $\boldsymbol{\psi} \in \mathbb{R}^N$ of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$. The main idea builds on the conclusion of [Proposition A.1](#), whereby there exists $\varepsilon > 0$ such that for any perturbation $(\widehat{\mathbf{s}}, \widehat{\boldsymbol{\psi}}) \in \mathbb{R}_s^{dN} \times \mathbb{R}_\psi^N$ with $|\widehat{\mathbf{s}}| + |\widehat{\boldsymbol{\psi}}| < \varepsilon$, the diagram $\mathbf{V}(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}})$ is obtained from the reference diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ by application of a “smooth” deformation

$$\text{Id} + \mathbf{m}(\widehat{\mathbf{s}}, \widehat{\boldsymbol{\psi}}), \text{ where } \mathbf{m}(\widehat{\mathbf{s}}, \widehat{\boldsymbol{\psi}}) \in W^{1, \infty}(\mathbb{R}^2, \mathbb{R}^2).$$

The derivative of $(\widehat{\mathbf{s}}, \widehat{\boldsymbol{\psi}}) \mapsto A_i(\mathbf{s} + \widehat{\mathbf{s}}, \boldsymbol{\psi} + \widehat{\boldsymbol{\psi}})$ thus results from the combination of the chain rule with the formula [\(2.8\)](#) for the derivative of the volume of a domain with respect to arbitrary perturbations of its boundary.

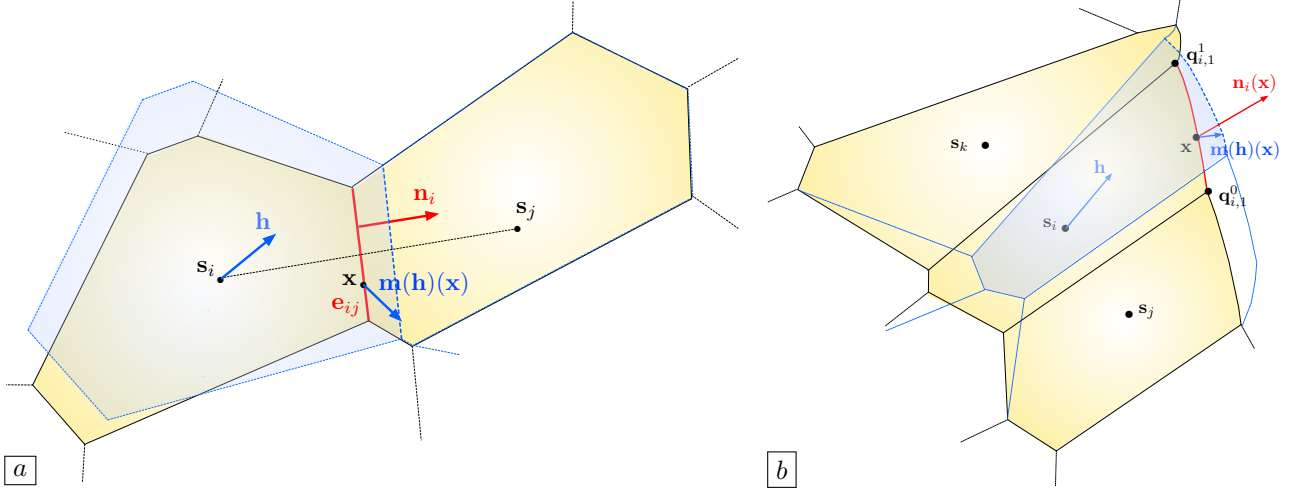


FIGURE 24. Deformation of (a) An interior cell; (b) A cell containing at least one circular arc in the diagram $\mathbf{V}(\mathbf{s}, \psi)$, induced by a small perturbation of the corresponding seed point, as considered in [Appendix B](#).

We first apply this strategy to calculate the derivative of $A_i(\mathbf{s}, \psi)$ with respect to \mathbf{s}_i , for a given index $i = 1, \dots, N$. For any vector $\mathbf{h} \in \mathbb{R}^2$, let $\mathbf{s}(\mathbf{h}) := (\mathbf{s}_1, \dots, \mathbf{s}_{i-1}, \mathbf{s}_i + \mathbf{h}, \mathbf{s}_{i+1}, \dots, \mathbf{s}_N)$ be the collection of seed points resulting from a translation \mathbf{h} of the i^{th} element of \mathbf{s} . As we have recalled, when $|\mathbf{h}| < \varepsilon$, the cell $V_i(\mathbf{s}(\mathbf{h}), \psi)$ is of the form

$$(B.1) \quad V_i(\mathbf{s}(\mathbf{h}), \psi) = (\text{Id} + \mathbf{m}(\mathbf{h})) \left(V_i(\mathbf{s}, \psi) \right),$$

for a certain differentiable mapping $B(\mathbf{0}, \varepsilon) \ni \mathbf{h} \mapsto \mathbf{m}(\mathbf{h}) \in W^{1, \infty}(\mathbb{R}^2, \mathbb{R}^2)$. The combination of the chain rule with formula (2.8) yields:

$$(B.2) \quad \begin{aligned} \frac{\partial A_i}{\partial \mathbf{s}_i}(\mathbf{s}, \psi)(\mathbf{h}) &= \int_{\partial V_i(\mathbf{s}, \psi)} \left\langle \frac{\partial \mathbf{m}}{\partial \mathbf{h}}(\mathbf{0})(\mathbf{h})(\mathbf{x}), \mathbf{n}_i(\mathbf{x}) \right\rangle ds(\mathbf{x}) \\ &= \sum_{j \in \mathcal{N}_i} \int_{\mathbf{e}_{ij}} \left\langle \frac{\partial \mathbf{m}}{\partial \mathbf{h}}(\mathbf{0})(\mathbf{h})(\mathbf{x}), \mathbf{n}_i(\mathbf{x}) \right\rangle ds(\mathbf{x}) + \sum_{r=1}^{n_{c,i}} \int_{\mathcal{C}_{i,r}} \left\langle \frac{\partial \mathbf{m}}{\partial \mathbf{h}}(\mathbf{0})(\mathbf{h})(\mathbf{x}), \mathbf{n}_i(\mathbf{x}) \right\rangle ds(\mathbf{x}), \end{aligned}$$

and we now proceed to glean information about the derivative $\mathbb{R}^2 \ni \mathbf{h} \mapsto \frac{\partial \mathbf{m}}{\partial \mathbf{h}}(\mathbf{0})(\mathbf{h}) \in W^{1, \infty}(\mathbb{R}^2, \mathbb{R}^2)$.

Let $j \in \mathcal{N}_i$, and \mathbf{x} be an arbitrary point on the edge \mathbf{e}_{ij} . For any perturbation $\mathbf{h} \in B(\mathbf{0}, \varepsilon)$, the point $\mathbf{m}(\mathbf{h})(\mathbf{x})$ belongs to the intersection of the cells $V_i(\mathbf{s}(\mathbf{h}), \psi)$ and $V_j(\mathbf{s}(\mathbf{h}), \psi)$, which implies that:

$$|\mathbf{x} + \mathbf{m}(\mathbf{h})(\mathbf{x}) - \mathbf{s}_i - \mathbf{h}|^2 - \psi_i = |\mathbf{x} + \mathbf{m}(\mathbf{h})(\mathbf{x}) - \mathbf{s}_j|^2 - \psi_j.$$

A simple calculation then yields:

$$2\langle \mathbf{m}(\mathbf{h})(\mathbf{x}), \mathbf{s}_j - \mathbf{s}_i \rangle = 2\langle \mathbf{x} - \mathbf{s}_i, \mathbf{h} \rangle + |\mathbf{x} - \mathbf{s}_j|^2 - |\mathbf{x} - \mathbf{s}_i|^2 - (\psi_j - \psi_i) + o(\mathbf{h}),$$

and so:

$$\left\langle \frac{\partial \mathbf{m}}{\partial \mathbf{h}}(\mathbf{0})(\mathbf{h})(\mathbf{x}), \mathbf{s}_j - \mathbf{s}_i \right\rangle = \langle \mathbf{x} - \mathbf{s}_i, \mathbf{h} \rangle.$$

As a result, we have proved the expression

$$(B.3) \quad \forall \mathbf{x} \in \mathbf{e}_{ij}, \quad \left\langle \frac{\partial \mathbf{m}}{\partial \mathbf{h}}(\mathbf{0})(\mathbf{h})(\mathbf{x}), \mathbf{n}_i \right\rangle = \left\langle \frac{\mathbf{x} - \mathbf{s}_i}{|\mathbf{s}_j - \mathbf{s}_i|}, \mathbf{h} \right\rangle.$$

Let now \mathbf{x} be a point on one of the circular arcs $\mathcal{C}_{i,r}$, $r = 1, \dots, n_{c,i}$. Again, the definition of $\mathcal{C}_{i,r}$ readily implies that, for any perturbation vector $|\mathbf{h}| < \varepsilon$, we have:

$$|\mathbf{x} + \mathbf{m}(\mathbf{h})(\mathbf{x}) - \mathbf{s}_i - \mathbf{h}|^2 = \psi_i.$$

Hence, we obtain

$$\langle \mathbf{m}(\mathbf{h})(\mathbf{x}), \mathbf{x} - \mathbf{s}_i \rangle = \langle \mathbf{x} - \mathbf{s}_i, \mathbf{h} \rangle - \frac{1}{2} |\mathbf{x} - \mathbf{s}_i|^2 + \frac{1}{2} \psi_i + o(\mathbf{h}),$$

and so:

$$(B.4) \quad \forall \mathbf{x} \in \mathcal{C}_{i,r}, \quad \left\langle \frac{\partial \mathbf{m}}{\partial \mathbf{h}}(\mathbf{0})(\mathbf{h})(\mathbf{x}), \mathbf{n}_i(\mathbf{x}) \right\rangle = \left\langle \frac{\mathbf{x} - \mathbf{s}_i}{|\mathbf{x} - \mathbf{s}_i|}, \mathbf{h} \right\rangle.$$

Eventually, combining (B.2) with (B.3) and (B.4), we obtain:

$$\begin{aligned} \frac{\partial A_i}{\partial \mathbf{s}_i}(\mathbf{s}, \boldsymbol{\psi})(\mathbf{h}) &= \left\langle \sum_{j \in \mathcal{N}_i} \int_{\mathbf{e}_{ij}} \frac{\mathbf{x} - \mathbf{s}_i}{|\mathbf{s}_j - \mathbf{s}_i|} ds(\mathbf{x}), \mathbf{h} \right\rangle + \left\langle \sum_{r=1}^{n_{c,i}} \int_{\mathcal{C}_{i,r}} \frac{\mathbf{x} - \mathbf{s}_i}{|\mathbf{x} - \mathbf{s}_i|} ds(\mathbf{x}), \mathbf{h} \right\rangle \\ &= \left\langle \sum_{j \in \mathcal{N}_i} \frac{|\mathbf{e}_{ij}|}{|\mathbf{s}_j - \mathbf{s}_i|} (\mathbf{m}_{ij} - \mathbf{s}_i), \mathbf{h} \right\rangle + \left\langle \sum_{r=1}^{n_{c,i}} (\mathbf{q}_{i,r}^0 - \mathbf{s}_i)^\perp - (\mathbf{q}_{i,r}^1 - \mathbf{s}_i)^\perp, \mathbf{h} \right\rangle, \end{aligned}$$

where the second line follows from an elementary calculation. This yields the desired formula.

Let us now apply the same strategy to calculate the derivative of $A_i(\mathbf{s}, \boldsymbol{\psi})$ with respect to the weight ψ_i . A perturbation of ψ_i of the form $\psi_i + h$ for $|h| < \varepsilon$ incurs a deformation of the i^{th} cell of the diagram of the form:

$$V_i(\mathbf{s}, \boldsymbol{\psi} + h\mathbf{e}_i) = (\text{Id} + \mathbf{m}(h))(V_i(\mathbf{s}, \boldsymbol{\psi})),$$

for some differentiable mapping $B(0, \varepsilon) \ni h \mapsto \mathbf{m}(h) \in W^{1,\infty}(\mathbb{R}^2, \mathbb{R}^2)$. Again, the combination of (2.8) with the chain rule yields:

$$(B.5) \quad \begin{aligned} \frac{\partial A_i}{\partial \psi_i}(\mathbf{s}, \boldsymbol{\psi})(h) &= \int_{\partial V_i(\mathbf{s}, \boldsymbol{\psi})} \left\langle \frac{\partial \mathbf{m}}{\partial h}(\mathbf{0})(h)(\mathbf{x}), \mathbf{n}_i(\mathbf{x}) \right\rangle ds(\mathbf{x}) \\ &= \sum_{j \in \mathcal{N}_i} \int_{\mathbf{e}_{ij}} \left\langle \frac{\partial \mathbf{m}}{\partial h}(\mathbf{0})(h)(\mathbf{x}), \mathbf{n}_i(\mathbf{x}) \right\rangle ds(\mathbf{x}) + \sum_{r=1}^{n_{c,i}} \int_{\mathcal{C}_{i,r}} \left\langle \frac{\partial \mathbf{m}}{\partial h}(\mathbf{0})(h)(\mathbf{x}), \mathbf{n}_i(\mathbf{x}) \right\rangle ds(\mathbf{x}), \end{aligned}$$

and we now aim to characterize the derivative $\mathbb{R} \ni h \mapsto \frac{\partial \mathbf{m}}{\partial h}(\mathbf{0})(h) \in W^{1,\infty}(\mathbb{R}^2, \mathbb{R}^2)$.

To this end, let us first consider an index $j \in \mathcal{N}_i$, and let $\mathbf{x} \in \mathbf{e}_{ij}$ be given. For any $h \in B(0, \varepsilon)$, it holds:

$$|\mathbf{x} + \mathbf{m}(h)(\mathbf{x}) - \mathbf{s}_i|^2 - \psi_i - h = |\mathbf{x} + \mathbf{m}(h)(\mathbf{x}) - \mathbf{s}_j|^2 - \psi_j,$$

and so

$$\langle \mathbf{m}(h)(\mathbf{x}), \mathbf{s}_j - \mathbf{s}_i \rangle = \frac{h}{2} + |\mathbf{x} - \mathbf{s}_j|^2 - |\mathbf{x} - \mathbf{s}_i|^2 - (\psi_j - \psi_i) + o(h).$$

We then obtain the following formula involving the derivative of the mapping m :

$$\forall \mathbf{x} \in \mathbf{e}_{ij}, \quad \left\langle \frac{\partial \mathbf{m}}{\partial h}(\mathbf{0})(h)(\mathbf{x}), \mathbf{s}_j - \mathbf{s}_i \right\rangle = \frac{h}{2},$$

which rewrites:

$$(B.6) \quad \forall \mathbf{x} \in \mathbf{e}_{ij}, \quad \left\langle \frac{\partial \mathbf{m}}{\partial h}(\mathbf{0})(h)(\mathbf{x}), \mathbf{n}_i(\mathbf{x}) \right\rangle = \frac{h}{2} \frac{1}{|\mathbf{s}_j - \mathbf{s}_i|}.$$

Let us now consider a point \mathbf{x} located on one of the circular arcs $\mathcal{C}_{i,r}$, $r = 1, \dots, n_{c,i}$. It holds, for all $h \in B(0, \varepsilon)$:

$$|\mathbf{x} + m(h)(\mathbf{x}) - \mathbf{s}_i|^2 = \psi_i + h,$$

and so:

$$(B.7) \quad \left\langle \frac{\partial \mathbf{m}}{\partial h}(\mathbf{0})(h), \mathbf{n}_i(\mathbf{x}) \right\rangle = \frac{1}{2|\mathbf{x} - \mathbf{s}_i|} = \frac{1}{2} \psi_i^{-1/2}.$$

Eventually, combining (B.5) with (B.6) and (B.7), we arrive at:

$$\begin{aligned} \frac{\partial A_i}{\partial \psi_i}(\mathbf{s}, \boldsymbol{\psi})(h) &= \left\langle \sum_{j \in \mathcal{N}_i} \int_{\mathbf{e}_{ij}} \frac{1}{2} \frac{1}{|\mathbf{s}_j - \mathbf{s}_i|} d\mathbf{s}(\mathbf{x}), h \right\rangle + \left\langle \sum_{r=1}^{n_{c,i}} \int_{\mathcal{C}_{i,r}} \frac{1}{2} \psi_i^{-1/2} d\mathbf{s}(\mathbf{x}), h \right\rangle \\ &= \left\langle \frac{1}{2} \sum_{j \in \mathcal{N}_i} \frac{|\mathbf{e}_{ij}|}{|\mathbf{s}_j - \mathbf{s}_i|}, h \right\rangle + \left\langle \frac{1}{2} \sum_{r=1}^{n_{c,i}} \theta_{i,r}, h \right\rangle, \end{aligned}$$

which yields the desired formula.

The expressions of the partial derivatives of the mapping $(\mathbf{s}, \boldsymbol{\psi}) \mapsto A_i(\mathbf{s}, \boldsymbol{\psi})$ with respect to the position \mathbf{s}_j and weight ψ_j of a neighboring index $j \in \mathcal{N}_i$ are obtained in a completely similar fashion, and we omit the details for brevity. The proof of Proposition B.1 is therefore complete. \square

Remark B.1. *The differentiability of the mapping $\boldsymbol{\psi} \mapsto K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$ – which essentially involves the areas $|V_i(\mathbf{s}, \boldsymbol{\psi})|$ of the cells of the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ – holds true under weaker assumptions than the collection (G1) to (G7), which implies the differentiability of all the individual vertices of the diagram, see for instance [47, 88]. To keep the presentation elementary and self-contained, we limit ourselves with the presented results, which are sufficient for our purpose.*

Corollary B.1. *Let $\mathbf{s} \in \mathbb{R}^{dN}$ and $\boldsymbol{\psi} \in \mathbb{R}^N$ be collections of seed points and weights satisfying (G1) to (G7). Then, $[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})]$ is a negative definite $N \times N$ matrix, which is therefore invertible.*

Proof. The above assumptions imply in particular that the diagram $\mathbf{V}(\mathbf{s}, \boldsymbol{\psi})$ contains no empty cell, so that the weights ψ_i are positive for $i = 1, \dots, N$ (see Step 2 in the next Appendix C where the same argument is used). Then, the matrix $[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})]$ is strictly diagonal dominant, and by a classical argument, it is invertible. Since its diagonal entries are negative, the claim follows. \square

Remark B.2. *In the case where the void phase $V_0(\mathbf{s}, \boldsymbol{\psi})$ is empty, $[\nabla_{\boldsymbol{\psi}} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})]$ has a one-dimensional kernel, which reflects the fact that the collection of weights $\boldsymbol{\psi}$ realizing a given measure constraint for each cell is unique up to addition of a common real number to all the ψ_i .*

APPENDIX C. PROOF OF THEOREM 3.1

The proof is decomposed into four steps.

Step 1: We prove that the function $\mathbb{R}^N \ni \boldsymbol{\psi} \mapsto K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) \in \mathbb{R}$ defined by (3.13) is concave, differentiable, and we calculate its derivative.

For $\mathbf{x} \in X$, we denote for convenience $\mathbf{s}_0 \equiv \mathbf{s}_0(\mathbf{x}) = \mathbf{x}$. For a given weight vector $\boldsymbol{\psi} \in \mathbb{R}^N$, let us define the measurable mappings $I_{\boldsymbol{\psi}} : X \rightarrow \{0, \dots, N\}$ and $T_{\boldsymbol{\psi}} : X \rightarrow X$ by:

$$I_{\boldsymbol{\psi}}(\mathbf{x}) = \arg \min_{i=0, \dots, N} (|\mathbf{x} - \mathbf{s}_i|^2 - \psi_i), \text{ and } T_{\boldsymbol{\psi}}(\mathbf{x}) = \mathbf{s}_{I_{\boldsymbol{\psi}}(\mathbf{x})},$$

where we recall the convention $\psi_0 = 0$. Note that the points $\mathbf{x} \in X$ where the above minimum is not uniquely attained form a set of Lebesgue measure 0, so that $I_{\boldsymbol{\psi}}$ and $T_{\boldsymbol{\psi}}$ are well-defined a.e. on X . This definition immediately implies that, for any measurable mapping $I : X \rightarrow \{0, \dots, N\}$, it holds:

$$|\mathbf{x} - \mathbf{s}_{I_{\boldsymbol{\psi}}(\mathbf{x})}|^2 - \psi_{I_{\boldsymbol{\psi}}(\mathbf{x})} \leq |\mathbf{x} - \mathbf{s}_{I(\mathbf{x})}|^2 - \psi_{I(\mathbf{x})} \text{ for a.e. } \mathbf{x} \in X.$$

It follows that, for any weight vectors $\boldsymbol{\psi}, \boldsymbol{\varphi} \in \mathbb{R}^N$:

$$\begin{aligned} (C.1) \quad K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\varphi}) &= \int_D (|\mathbf{x} - \mathbf{s}_{I_{\boldsymbol{\varphi}}(\mathbf{x})}(\mathbf{x})| - \varphi_{I_{\boldsymbol{\varphi}}(\mathbf{x})}) d\mathbf{x} + \sum_{i=1}^N \nu_i \varphi_i \\ &\leq \int_D (|\mathbf{x} - \mathbf{s}_{I_{\boldsymbol{\psi}}(\mathbf{x})}(\mathbf{x})| - \varphi_{I_{\boldsymbol{\psi}}(\mathbf{x})}) d\mathbf{x} + \sum_{i=1}^N \nu_i \varphi_i \\ &= \sum_{i=0}^N \int_{V_i(\mathbf{s}, \boldsymbol{\psi})} (|\mathbf{x} - \mathbf{s}_i|^2 - \varphi_i) d\mathbf{x} + \sum_{i=1}^N \nu_i \varphi_i \\ &= K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) + \langle \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}), \boldsymbol{\varphi} - \boldsymbol{\psi} \rangle, \end{aligned}$$

where we recall that the components of $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) \in \mathbb{R}^N$ are defined by:

$$(C.2) \quad F_i(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) := \nu_i - |V_i(\mathbf{s}, \boldsymbol{\psi})|, \quad i = 1, \dots, N.$$

Hence, the superdifferential $\partial_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$ of the mapping $\boldsymbol{\varphi} \mapsto K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\varphi})$ at an arbitrary weight vector $\boldsymbol{\psi} \in \mathbb{R}^N$ contains $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) \in \mathbb{R}^N$, and it is in particular non empty. On the other hand, the inequality (C.1) expresses $K(\mathbf{s}, \boldsymbol{\nu}, \cdot)$ as the minimum of a collection of affine functions:

$$K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\varphi}) = \min_{\boldsymbol{\psi} \in \mathbb{R}^N} \left(K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) + \langle \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}), \boldsymbol{\varphi} - \boldsymbol{\psi} \rangle \right), \quad \boldsymbol{\varphi} \in \mathbb{R}^N.$$

This discussion shows that the mapping $\boldsymbol{\psi} \mapsto K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$ is concave, and thus differentiable at a.e. $\boldsymbol{\psi} \in \mathbb{R}^N$; besides, its gradient at any point $\boldsymbol{\psi}$ where it is differentiable equals $\nabla_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) = \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$, see [110], Th. 25.4.

The C^1 character of $\boldsymbol{\psi} \mapsto K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$ follows from the same argument as in the proof of Th. 40 in [88]. At first, a simple application of the Lebesgue dominated convergence theorem shows that the mapping $\boldsymbol{\psi} \rightarrow \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$ is continuous on \mathbb{R}^N . On the other hand, according to Th. 25.6 in [110] the superdifferential $\partial_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$ at any vector $\boldsymbol{\psi} \in \mathbb{R}^N$ reads:

$$\partial_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) = \left\{ \lim_{n \rightarrow \infty} \mathbf{g}^n, \quad \mathbf{g}^n = \nabla_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n), \quad \boldsymbol{\psi}^n \xrightarrow{n \rightarrow \infty} \boldsymbol{\psi} \text{ and } K(\mathbf{s}, \boldsymbol{\nu}, \cdot) \text{ is differentiable at } \boldsymbol{\psi}^n \right\}.$$

Now, the continuity of $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \cdot)$ implies that, for any sequence $\boldsymbol{\psi}^n \rightarrow \boldsymbol{\psi}$ such that $K(\mathbf{s}, \boldsymbol{\nu}, \cdot)$ is differentiable at $\boldsymbol{\psi}^n$, the unique element $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n) = \nabla_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)$ in $\partial_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}^n)$ converges to $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$ as $n \rightarrow \infty$. The superdifferential $\partial_{\boldsymbol{\psi}} K(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi})$ is thus reduced to a single point, and so $K(\mathbf{s}, \boldsymbol{\nu}, \cdot)$ is differentiable on \mathbb{R}^N , with continuous derivative $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \cdot)$. This completes the first step in the proof of [Theorem 3.1](#); in passing, we have proved that the maximizers $\boldsymbol{\psi} \in \mathbb{R}^N$ in (3.12) are characterized by the equation

$$\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\psi}) = \mathbf{0}.$$

Step 2: We prove the existence of a weight vector $\boldsymbol{\psi}^$ satisfying (3.11).*

To this end, we adapt the argument of the proof of Corollary 39 in [88]. This starts with a few remarks:

- For any $\boldsymbol{\psi} \in \mathbb{R}^N$ and $i = 1, \dots, N$, the mapping $\mathbb{R} \ni t \mapsto |V_i(\mathbf{s}, \boldsymbol{\psi} + t\mathbf{e}_i)|$ is non decreasing.
- For any $\boldsymbol{\psi} \in \mathbb{R}^N$, and $i, j = 1, \dots, N$, $i \neq j$, the mapping $\mathbb{R} \ni t \mapsto |V_j(\mathbf{s}, \boldsymbol{\psi} + t\mathbf{e}_i)|$ is non increasing.
- If the cell $V_i(\mathbf{s}, \boldsymbol{\psi})$ is non empty, then the weight ψ_i must satisfy $\psi_i \geq 0$, since any point $\mathbf{x} \in V_i(\mathbf{s}, \boldsymbol{\psi})$ satisfies in particular $|\mathbf{x} - \mathbf{s}_i|^2 - \psi_i \leq 0$.
- In the same spirit, if $V_i(\mathbf{s}, \boldsymbol{\psi})$ has positive Lebesgue measure, one has $\psi_i > 0$.
- If the cell $V_0(\mathbf{s}, \boldsymbol{\psi})$ in (3.8) is non empty, it holds:

$$\forall i = 1, \dots, N, \quad \psi_i \leq \text{diam}(D)^2.$$

Now, let $\boldsymbol{\nu} \in \mathbb{R}^N$ be a vector satisfying (3.10), and let $\mathcal{K} \subset \mathbb{R}^N$ be the set defined by

$$\mathcal{K} = \left\{ \boldsymbol{\psi} \in \mathbb{R}^N \text{ s.t. } |V_i(\mathbf{s}, \boldsymbol{\psi})| \leq \nu_i \text{ for all } i = 1, \dots, N \right\}.$$

According to the previous observations, for all $\boldsymbol{\psi} \in \mathcal{K}$, it holds $|V_0(\mathbf{s}, \boldsymbol{\psi})| \geq \nu_0 > 0$, and so $\mathcal{K} \subset [0, \text{diam}(D)^2]^N$. Moreover, since the mapping $\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \cdot)$ is continuous, \mathcal{K} is closed, and it is therefore compact. Let us then consider one solution $\boldsymbol{\psi}^*$ to the following maximization problem:

$$(C.3) \quad \max_{\boldsymbol{\psi} \in \mathcal{K}} W(\boldsymbol{\psi}), \quad \text{where } W(\boldsymbol{\psi}) = \sum_{i=1}^N \psi_i.$$

Assume that there exists $i \in \{1, \dots, N\}$ such that $|V_i(\mathbf{s}, \boldsymbol{\psi}^*)| < \nu_i$. Then by the continuity of $F(\mathbf{s}, \boldsymbol{\nu}, \cdot)$, there exists $t > 0$ such that $|V_i(\mathbf{s}, \boldsymbol{\psi}^* + t\mathbf{e}_i)| < \nu_i$, and by the foregoing observations, one also has:

$$|V_j(\mathbf{s}, \boldsymbol{\psi}^* + t\mathbf{e}_i)| \leq |V_j(\mathbf{s}, \boldsymbol{\psi}^*)| \leq \nu_j \text{ for } j = 1, \dots, N, \quad j \neq i.$$

Hence, the vector $(\boldsymbol{\psi}^* + t\mathbf{e}_i)$ also belongs to \mathcal{K} and $W(\boldsymbol{\psi}^* + t\mathbf{e}_i) > W(\boldsymbol{\psi}^*)$, which contradicts the definition of $\boldsymbol{\psi}^*$. We have thus proved that any maximizer $\boldsymbol{\psi}^*$ in (C.3) satisfies the desired property:

$$|V_i(\mathbf{s}, \boldsymbol{\psi}^*)| = \nu_i, \quad \text{for } i = 1, \dots, N.$$

Step 3: We prove the uniqueness of ψ^* under the genericity assumptions (G1) to (G7).

We have seen in Corollary B.1 that under the assumptions (G1) to (G7), the mapping $\psi \mapsto \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \psi)$ is differentiable at ψ^* , and that the derivative $[\nabla_{\psi} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \psi^*)]$ is an invertible $N \times N$ matrix.

Let us assume that there exists another weight vector φ^* satisfying (3.11). Then ψ^* and φ^* are both maximizers of the concave Kantorovich functional $K(\mathbf{s}, \boldsymbol{\nu}, \cdot)$, and so, for any $t \in (0, 1)$, it holds:

$$\mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, (1-t)\psi^* + t\varphi^*) = \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \psi^*) = \mathbf{0}.$$

Rearranging the above equality, dividing both sides by t , then letting t tend to 0, we obtain:

$$[\nabla_{\psi} \mathbf{F}(\mathbf{s}, \boldsymbol{\nu}, \psi^*)](\varphi^* - \psi^*) = \mathbf{0},$$

which proves that $\psi^* = \varphi^*$, as expected.

Step 4: We use duality to conclude that (3.15) is the optimal transport mapping between μ and ν .

Let us recall the well-known expression of the dual maximization problem attached to the Kantorovich minimization problem (OT-K):

$$(OT-D) \quad \max_{\psi \in \mathcal{C}(X)} \left(\int_X \min_{\mathbf{y} \in X} (c(\mathbf{x}, \mathbf{y}) - \psi(\mathbf{y})) \, d\mu(\mathbf{y}) + \nu_0 \frac{1}{|D|} \int_X \psi(\mathbf{y}) \, d\mathbf{y} + \sum_{i=1}^N \nu_i \psi_i \right),$$

see e.g. [112]. The weak duality phenomenon reads, in the present situation:

$$(C.4) \quad \max (OT-D) \leq \min (OT-K).$$

Note that actually, strong duality holds, i.e. $\max (OT-D) = \min (OT-K)$, see Th. 1.39 in [112]. We shall not need this fact, which will actually turn out to be a by-product of our analysis.

Let $\psi \in \mathbb{R}^N$ be the weight vector supplied by the second step, guaranteeing the equality (3.11), and let T_{ψ} be the mapping defined in (3.15); we recall from Step 2 that $\psi_i > 0$ for all $i = 1, \dots, N$ and we retain the convention $\psi_0 = 0$. By construction, $(T_{\psi})_{\#}\mu = \nu$, so that $(\text{Id}, T_{\psi})_{\#}\mu$ is an admissible transport plan in the problem (OT-K); hence:

$$\min (OT-K) \leq \int_X c(\mathbf{x}, T_{\psi}(\mathbf{x})) \, d\mu(\mathbf{x}).$$

On the other hand, let us define the (discontinuous) function $\tilde{\psi} : X \rightarrow \mathbb{R}$ by

$$\forall \mathbf{y} \in X, \quad \tilde{\psi}(\mathbf{y}) = \begin{cases} \psi_i & \text{if } \mathbf{y} = \mathbf{s}_i, i = 1, \dots, N, \\ \psi_0 = 0 & \text{otherwise.} \end{cases}$$

It follows from the definition of T_{ψ} that:

$$\forall \mathbf{x} \in X, \quad c(\mathbf{x}, T_{\psi}(\mathbf{x})) - \tilde{\psi}(T_{\psi}(\mathbf{x})) = \min_{\mathbf{y} \in X} (c(\mathbf{x}, \mathbf{y}) - \tilde{\psi}(\mathbf{y})).$$

Integrating this identity, we obtain:

$$(C.5) \quad \int_X c(\mathbf{x}, T_{\psi}(\mathbf{x})) \, d\mu(\mathbf{x}) - \sum_{i=1}^N \nu_i \psi_i = \int_X \min_{\mathbf{y} \in X} (c(\mathbf{x}, \mathbf{y}) - \tilde{\psi}(\mathbf{y})) \, d\mu(\mathbf{x}).$$

We now approximate the discontinuous function $\tilde{\psi}$ in the last term in the above right-hand side. To this end, let us first introduce the step function χ_{ε} defined for $\varepsilon > 0$ small enough by

$$\forall \mathbf{y} \in X, \quad \chi_{\varepsilon}(\mathbf{y}) = \begin{cases} \psi_i & \text{if } |\mathbf{y} - \mathbf{s}_i| < \varepsilon \text{ for some } i = 1, \dots, N, \\ 0 & \text{otherwise.} \end{cases}$$

We also introduce one function $\zeta_{\varepsilon} \in \mathcal{C}(X)$ satisfying the properties:

$$\begin{cases} \zeta_{\varepsilon}(\mathbf{y}) = \psi_i & \text{if } |\mathbf{y} - \mathbf{s}_i| < \frac{\varepsilon}{2} \text{ for } i = 1, \dots, N, \\ 0 \leq \zeta_{\varepsilon}(\mathbf{y}) \leq \psi_i & \text{if } \frac{\varepsilon}{2} \leq |\mathbf{y} - \mathbf{s}_i| < \varepsilon \text{ for } i = 1, \dots, N, \\ 0 & \text{otherwise,} \end{cases}$$

i.e. ζ_ε is a smoothed version of $\tilde{\psi}$, whose support is contained in a collection of balls of radius ε around the \mathbf{s}_i , $i = 1, \dots, N$; the latter can easily be constructed thanks to mollifiers. Then, it holds:

$$\min_{\mathbf{y} \in X} \left(c(\mathbf{x}, \mathbf{y}) - \chi_\varepsilon(\mathbf{y}) \right) \leq \min_{\mathbf{y} \in X} \left(c(\mathbf{x}, \mathbf{y}) - \zeta_\varepsilon(\mathbf{y}) \right) \leq \min_{\mathbf{y} \in X} \left(c(\mathbf{x}, \mathbf{y}) - \tilde{\psi}(\mathbf{y}) \right).$$

On the other hand, the first term in the above left-hand side can be estimated as:

$$\begin{aligned} \min_{\mathbf{y} \in X} \left(c(\mathbf{x}, \mathbf{y}) - \chi_\varepsilon(\mathbf{y}) \right) &\geq \min \left(0, \min_{\substack{i=1, \dots, N \\ \mathbf{h} \in B(0, \varepsilon)}} \left(c(\mathbf{x}, \mathbf{s}_i + \mathbf{h}) - \psi_i \right) \right) \\ &= \min \left(0, \min_{\substack{i=1, \dots, N \\ \mathbf{h} \in B(0, \varepsilon)}} \left(c(\mathbf{x} - \mathbf{h}, \mathbf{s}_i) - \psi_i \right) \right) \\ &\geq \min \left(0, \min_{i=1, \dots, N} \left(c(\mathbf{x}, \mathbf{s}_i) - \psi_i \right) \right) - L\varepsilon \\ &= \min_{\mathbf{y} \in X} \left(c(\mathbf{x}, \mathbf{y}) - \tilde{\psi}(\mathbf{y}) \right) - L\varepsilon, \end{aligned}$$

where L is a Lipschitz constant for the cost $c(\mathbf{x}, \mathbf{y}) = |\mathbf{x} - \mathbf{y}|^2$ on $X \times X$, and we have used the particular difference structure of the latter when passing from the first to the second line. It follows that:

$$\min_{\mathbf{y} \in X} \left(c(\mathbf{x}, \mathbf{y}) - \tilde{\psi}(\mathbf{y}) \right) \leq \min_{\mathbf{y} \in X} \left(c(\mathbf{x}, \mathbf{y}) - \zeta_\varepsilon(\mathbf{y}) \right) + L\varepsilon,$$

and so, returning to (C.5):

$$\begin{aligned} \int_X c(\mathbf{x}, T_\psi(\mathbf{x})) \, d\mu(\mathbf{x}) &\leq \int_X \min_{\mathbf{y} \in X} \left(c(\mathbf{x}, \mathbf{y}) - \zeta_\varepsilon(\mathbf{y}) \right) \, d\mu(\mathbf{x}) + \sum_{i=1}^N \nu_i \psi_i + L\varepsilon \\ &\leq \int_X \min_{\mathbf{y} \in X} \left(c(\mathbf{x}, \mathbf{y}) - \zeta_\varepsilon(\mathbf{y}) \right) \, d\mu(\mathbf{x}) + \nu_0 \frac{1}{|D|} \int_X \zeta_\varepsilon(\mathbf{y}) \, d\mathbf{y} + \sum_{i=1}^N \nu_i \psi_i + L\varepsilon \\ &\leq \max(\text{OT-D}) + L\varepsilon. \end{aligned}$$

Since ε is arbitrary, we may let ε tend to 0 in the above inequality; recalling the weak duality inequality (C.4), we have thus proved that T_ψ is indeed an optimal transport mapping.

The uniqueness of such an optimal transport mapping is a classical fact in optimal transport theory, see e.g. Th. 1.17 in [112]. This concludes the proof of [Theorem 3.1](#).

APPENDIX D. IMPLEMENTATION DETAILS OF THE VIRTUAL ELEMENT METHOD

In this section, we describe the practical implementation of the Virtual Element Method for the solution of the conductivity equation (Cond) and of the linear elasticity system (Elas).

D.1. The case of the conductivity equation

Slipping into the notation of [Section 5.2](#) and closely following [125], we now provide a little details about the computation of the entries of the local stiffness matrix $K^E \in \mathbb{R}^{n^E \times n^E}$ defined by

$$\forall i, j = 1, \dots, n, \quad K_{ij}^E = \tilde{a}^E(\zeta_i, \zeta_j),$$

where the discrete bilinear form $\tilde{a}^E(u, v)$ reads

$$\forall u, v \in \mathcal{W}(E), \quad \tilde{a}^E(u, v) = a^E(\pi_C u, \pi_C v) + \tilde{s}^E(u - \pi_P u, v - \pi_P v),$$

and the basis $\{\zeta_i\}_{i=1, \dots, n^E}$ of $\mathcal{W}(E)$ is characterized by (5.11). As discussed in [Section 5.2](#), this formula induces a decomposition of K^E as the sum of two contributions:

$$K^E = P^E + \tilde{S}^E, \quad \text{where } P_{ij}^E := a^E(\pi_C \zeta_i, \pi_C \zeta_j) \text{ and } \tilde{S}_{ij}^E = \tilde{s}^E(\zeta_i - \pi_P \zeta_i, \zeta_j - \pi_P \zeta_j).$$

The cornerstone of the calculation of both matrices is therefore the calculation of the matrix $\Pi \in \mathbb{R}^{3 \times n^E}$ of the projection operator π_P onto affine functions, expressed in the bases $\{\zeta_i\}_{i=1, \dots, n^E}$ of $\mathcal{W}(E)$ and $\{m_\alpha\}_{\alpha=1, 2, 3}$ of $\mathcal{P}(E)$, i.e. the i^{th} column of Π gathers the entries of $\pi_P \zeta_i$ in the basis m_α :

$$\pi_P \zeta_i(\mathbf{x}) = \sum_{\alpha=1}^3 \Pi_{\alpha i} m_\alpha(\mathbf{x}), \quad i = 1, \dots, n^E, \quad \mathbf{x} \in E.$$

By using the following characterization of $\pi_{\mathcal{P}}$,

$$\overline{\pi_{\mathcal{P}}\zeta_i} = \overline{\zeta_i} = \frac{1}{n^E} \text{ and } a^E(\pi_{\mathcal{P}}\zeta_i, m_\alpha) = a^E(\zeta_i, m_\alpha), \quad \alpha = 2, 3,$$

we obtain the identity:

$$(D.1) \quad \tilde{G}\Pi = B,$$

where

- The matrix $\tilde{G} \in \mathbb{R}^{3 \times 3}$ is defined by:

$$\tilde{G}_{11} = 1 \text{ and } \tilde{G}_{12} = \tilde{G}_{13} = 0,$$

as well as:

$$\tilde{G}_{\alpha,\beta} = \int_E \nabla m_\alpha \cdot \nabla m_\beta \, d\mathbf{x} \text{ for } \alpha = 2, 3 \text{ and } \beta = 1, 2, 3.$$

- The matrix $B \in \mathbb{R}^{3 \times n^E}$ is given by:

$$B_{1i} = \frac{1}{n^E}, \text{ and } B_{\alpha i} = \int_E \nabla \zeta_i \cdot \nabla m_\alpha \, d\mathbf{x} \text{ for } i = 1, \dots, n^E \text{ and } \alpha = 2, 3.$$

The entries of B can be calculated in closed form by using the facts that ∇m_α is constant on E and that the basis functions ζ_i are affine on each edge of ∂E . Indeed, Green's formula implies that, for $\alpha = 2, 3$:

$$B_{\alpha,i} = \left(\int_E \nabla \zeta_i \, d\mathbf{x} \right) \cdot \nabla m_\alpha = \left(\int_{\partial E} \zeta_i \mathbf{n} \, dl \right) \cdot \nabla m_\alpha = \frac{1}{2} |\hat{\mathbf{e}}_i| \mathbf{n}_{\hat{\mathbf{e}}_i} \cdot \nabla m_\alpha.$$

Hence, the system (D.1) features the explicit matrix \tilde{G} and right-hand side B , and it can be solved for the matrix Π . The practical implementation is further simplified thanks to the introduction of the matrix $D \in \mathbb{R}^{n^E \times 3}$ corresponding to the expression of the polynomials m_α , $\alpha = 1, 2, 3$ in the basis $\{\zeta_i\}_{i=1, \dots, n^E}$ of $\mathcal{W}(E)$, that is:

$$m_\alpha(\mathbf{x}) = \sum_{i=1}^{n^E} D_{i\alpha} \zeta_i(\mathbf{x}), \quad \mathbf{x} \in E.$$

The coefficients of this matrix read, by definition of the ζ_i :

$$D_{i\alpha} = m_\alpha(\mathbf{q}_i^E),$$

and so

$$D_{i1} = 1, \quad D_{i2} = q_{i,1}^E - \overline{q_1^E}, \text{ and } D_{i3} = q_{i,2}^E - \overline{q_2^E} \text{ for } i = 1, \dots, n^E.$$

The definitions of the matrices \tilde{G} , B and D yield the following relation:

$$\tilde{G} = BD.$$

Indeed, elementary calculations lead to:

$$\tilde{G}_{11} = 1 = \sum_{i=1}^{n^E} B_{1i} D_{i1}, \quad \tilde{G}_{1\beta} = 0 = \sum_{i=1}^{n^E} B_{1i} D_{i\beta} \text{ for } \beta = 2, 3,$$

and likewise, for $\alpha = 2, 3$,

$$\tilde{G}_{\alpha\beta} = \int_E \nabla m_\alpha \cdot \nabla m_\beta \, d\mathbf{x} = \sum_{i=1}^{n^E} D_{i\beta} \int_E \nabla m_\alpha \cdot \nabla \zeta_i \, d\mathbf{x} = \sum_{i=1}^{n^E} D_{i\beta} B_{\alpha i}.$$

Once the matrix Π is calculated, P^E is inferred from the formula:

$$(D.2) \quad P^E = \gamma \Pi^T G \Pi, \text{ where } G \text{ is the matrix with entries } G_{\alpha\beta} = \int_E \nabla m_\alpha \cdot \nabla m_\beta \, d\mathbf{x}, \quad \alpha, \beta = 1, 2, 3.$$

Likewise, \tilde{S}^E is obtained as:

$$(D.3) \quad \tilde{S}^E = \alpha^E \left(\mathbf{I} - D\Pi \right)^T \left(\mathbf{I} - D\Pi \right), \text{ where } \alpha^E = 1,$$

and \mathbf{I} is the identity matrix with size $n^E \times n^E$.

Summarizing, the computations of the matrices P^E and S^E (and thus K^E) proceed as follows:

- (1) Assemble the matrices B and D ;
- (2) Calculate $\tilde{G} = BD \in \mathbb{R}^{3 \times 3}$;
- (3) Compute $\Pi \in \mathbb{R}^{3 \times n^E}$ as the solution to the matrix system

$$\tilde{G}\Pi = B.$$

- (4) The matrix P^E and \tilde{S}^E are inferred from the formulas (D.2) and (D.3).

Eventually, the entries F_i^E of the local force vector $F^E \in \mathbb{R}^{n^E}$ are simply approximated as:

$$F_i^E = \int_E f \zeta_i \, d\mathbf{x} \approx f(\overline{\mathbf{q}}^E) \int_E \zeta_i \, d\mathbf{x} \approx f(\overline{\mathbf{q}}^E) \frac{|E|}{n^E}, \quad i = 1, \dots, n^E.$$

Remark D.1. *An important consequence of the above analysis is that the projection $\pi_{\mathcal{P}}\zeta$ of a function $\zeta \in \mathcal{W}_{\mathcal{T}}$ onto affine functions depends only on the values $\zeta(\mathbf{q}_i^E)$ ($i = 1, \dots, n^E$) of ζ at the vertices of the considered element E*

D.2. The case of the linear elasticity system

The present section deals with the linear elasticity system (Elas). Relying on [18, 60], we briefly describe its numerical solution by the Virtual Element Method.

As emphasized in Section 5 and Appendix D.1, the most critical operation in this perspective is the calculation of the local stiffness matrix $K^E \in \mathbb{R}^{2n^E \times 2n^E}$ defined by

$$\forall i, j = 1, \dots, 2n^E, \quad K_{ij}^E = \tilde{a}^E(\zeta_i, \zeta_j),$$

where \tilde{a}^E is given by:

$$\forall u, v \in \mathcal{W}(E), \quad \tilde{a}^E(\mathbf{u}, \mathbf{v}) = a^E(\pi_C \mathbf{u}, \pi_C \mathbf{v}) + \tilde{s}^E(\mathbf{u} - \pi_{\mathcal{P}} \mathbf{u}, \mathbf{v} - \pi_{\mathcal{P}} \mathbf{v}),$$

and the basis functions ζ_i , $i = 1, \dots, 2n^E$ are characterized by (5.15) and (5.16). Again, we decompose K^E as:

$K^E = P^E + \tilde{S}^E$, where $P^E, \tilde{S}^E \in \mathbb{R}^{2n^E \times 2n^E}$ are the matrices

$$P_{ij}^E := a^E(\pi_C \zeta_i, \pi_C \zeta_j) \text{ and } \tilde{S}_{ij}^E := \tilde{s}^E(\zeta_i - \pi_{\mathcal{P}} \zeta_i, \zeta_j - \pi_{\mathcal{P}} \zeta_j).$$

Let $W_C \in \mathbb{R}^{2n^E \times 3}$ be the transpose of the matrix of π_C in the bases $\{\zeta_i\}_{i=1, \dots, 2n^E}$ and $\{\mathbf{c}_\alpha\}_{\alpha=1, 2, 3}$, i.e. the i^{th} line of W_C gathers the coordinates of the projection $\pi_C \zeta_i$ over the functions $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3$ in (5.18):

$$\forall i = 1, \dots, 2n^E, \quad \pi_C \zeta_i(\mathbf{x}) = \sum_{\beta=1}^3 (W_C)_{i\beta} \mathbf{c}_\beta(\mathbf{x}), \quad \mathbf{x} \in E.$$

Likewise, let $W_R \in \mathbb{R}^{2n^E \times 3}$ be the transpose of the matrix of π_R in the bases $\{\zeta_i\}_{i=1, \dots, 2n^E}$ and $\{\mathbf{r}_\alpha\}_{\alpha=1, 2, 3}$:

$$\forall i = 1, \dots, 2n^E, \quad \pi_R \zeta_i(\mathbf{x}) = \sum_{\beta=1}^3 (W_R)_{i\beta} \mathbf{r}_\beta(\mathbf{x}).$$

With these notations, the first block $P^E \in \mathbb{R}^{2n^E \times 2n^E}$ of K^E reads,

$$P_{ij}^E = a^E(\pi_C \zeta_i, \pi_C \zeta_j) = \sum_{\alpha, \beta=1}^3 (W_C)_{i\alpha} (W_C)_{j\beta} a^E(\mathbf{c}_\alpha, \mathbf{c}_\beta),$$

and so, in matrix form:

$$(D.4) \quad P^E = W_C D W_C^T, \text{ where } D \in \mathbb{R}^{3 \times 3} \text{ is defined by } D_{\alpha\beta} = a^E(\mathbf{c}_\alpha, \mathbf{c}_\beta) = \int_E A e(\mathbf{c}_\alpha) : e(\mathbf{c}_\beta) \, d\mathbf{x}.$$

An elementary calculation yields:

$$D = |E| \begin{pmatrix} 2\mu + \lambda & \lambda & 0 \\ \lambda & 2\mu + \lambda & 0 \\ 0 & 0 & 4\mu \end{pmatrix},$$

and we shall detail below how to calculate the matrices W_C and W_R in practice.

As far as the stabilizing block \tilde{S}^E of K^E is concerned, we need to calculate the matrices P_C and $P_R \in \mathbb{R}^{2n^E \times 2n^E}$ of the projections $\pi_C : \mathcal{W}(E) \rightarrow \mathcal{W}(E)$ and $\pi_R : \mathcal{W}(E) \rightarrow \mathcal{W}(E)$ in the basis $\{\zeta_i\}_{i=1, \dots, 2n^E}$. To this end, let us introduce the matrix $N_C \in \mathbb{R}^{2n^E \times 3}$, whose α^{th} column ($\alpha = 1, 2, 3$) contains the coordinates of \mathbf{c}_α over the basis $\{\zeta_i\}$, that is:

$$\forall \alpha = 1, 2, 3, \quad \mathbf{c}_\alpha(\mathbf{x}) = \sum_{i=1}^{2n^E} (N_C)_{i\alpha} \zeta_i(\mathbf{x}), \quad \mathbf{x} \in E.$$

With these notations, it holds, for $j = 1, \dots, 2n^E$:

$$\begin{aligned} \pi_C \zeta_j &= \sum_{\alpha=1}^3 (W_C)_{j\alpha} \mathbf{c}_\alpha \\ &= \sum_{i=1}^{2n^E} \left(\sum_{\alpha=1}^3 (N_C)_{i\alpha} (W_C)_{j\alpha} \right) \zeta_i \end{aligned}$$

and so, $P_C \in \mathbb{R}^{2n^E \times 2n^E}$ reads:

$$P_C = N_C W_C^T.$$

Likewise, let $N_R \in \mathbb{R}^{2n^E \times 3}$ be the matrix whose α^{th} column ($\alpha = 1, 2, 3$) contains the coordinates of \mathbf{r}_α in the basis ζ_i , that is:

$$\forall \alpha = 1, 2, 3, \quad \mathbf{r}_\alpha(\mathbf{x}) = \sum_{i=1}^{2n^E} (N_R)_{i\alpha} \zeta_i(\mathbf{x}), \quad \mathbf{x} \in E;$$

it holds:

$$(D.5) \quad P_R = N_R W_R^T.$$

Finally, the matrix $P_P \in \mathbb{R}^{2n^E \times 2n^E}$ of the projection π_P over the space $\mathcal{P}(E)$ of affine functions reads:

$$(D.6) \quad P_P = P_R + P_C.$$

With these notations, \tilde{S}^E can be computed via the following formula:

$$(D.7) \quad S^E = \alpha^E (\mathbf{I} - P_P)^T (\mathbf{I} - P_P),$$

where \mathbf{I} is the $2n^E \times 2n^E$ identity matrix.

It follows from the formulas (D.4) to (D.7) that the basic ingredients of the practical implementation of the Virtual Element Method dedicated to the solution of the linear elasticity system (Elas) are the assembly of the matrices N_C, N_R, W_C, W_R , and the calculation of the coefficient α^E ; we now detail these operations.

Assembly of the matrices W_C, W_R

For $i = 1, \dots, n^E$, let us introduce the vector $\mathbf{a}_i \in \mathbb{R}^2$ defined by

$$\mathbf{a}_i = \frac{|\hat{\mathbf{e}}_i|}{2|E|} \mathbf{n}_{\hat{\mathbf{e}}_i},$$

where we recall the notation $\mathbf{n}_{\hat{\mathbf{e}}_i}$ in (5.1).

Let us first consider the matrix $W_R \in \mathbb{R}^{2n^E \times 3}$; for $i = 1, \dots, n^E$ and $\mathbf{x} \in E$, we have

$$\pi_R \zeta_{2i-1}(\mathbf{x}) = \overline{\zeta_{2i-1}} + \langle \omega(\zeta_{2i-1}) \rangle \begin{pmatrix} -(x_2 - \overline{q_2^E}) \\ x_1 - \overline{q_1^E} \end{pmatrix},$$

where obviously, $\overline{\zeta_{2i-1}} = \frac{1}{n^E} \mathbf{r}_1$ and a simple calculation shows that:

$$\begin{aligned}
\langle \omega(\zeta_{2i-1}) \rangle &= -\frac{1}{2|E|} \int_E \frac{\partial}{\partial x_2} (\zeta_{2i-1})_1 \, d\mathbf{x} \\
&= -\frac{1}{2|E|} \int_{\partial E} (\zeta_{2i-1})_1 n_2 \, d\ell \\
&= -\frac{1}{4|E|} (|\mathbf{e}_{i-1}| n_{\mathbf{e}_{i-1}} + |\mathbf{e}_i| n_{\mathbf{e}_i})_2 \\
&= -\frac{1}{|E|} \frac{|\hat{\mathbf{e}}_i|}{4} (n_{\hat{\mathbf{e}}_i})_2 \\
&= -\frac{1}{2} (a_i)_2.
\end{aligned}$$

Hence, it follows:

$$\pi_R \zeta_{2i-1}(\mathbf{x}) = \frac{1}{n^E} \mathbf{r}_1(\mathbf{x}) - \frac{1}{2} (a_i)_2 \mathbf{r}_3(\mathbf{x}).$$

A similar calculation yields:

$$\pi_R \zeta_{2i}(\mathbf{x}) = \frac{1}{n^E} \mathbf{r}_2(\mathbf{x}) + \frac{1}{2} (a_i)_1 \mathbf{r}_3(\mathbf{x}).$$

Likewise, considering the entries of W_C , we obtain:

$$\pi_C \zeta_{2i-1}(\mathbf{x}) = (a_i)_1 \mathbf{c}_1(\mathbf{x}) + \frac{1}{2} (a_i)_2 \mathbf{c}_3(\mathbf{x}), \text{ and } \pi_C \zeta_{2i}(\mathbf{x}) = (a_i)_2 \mathbf{c}_2(\mathbf{x}) + \frac{1}{2} (a_i)_1 \mathbf{c}_3(\mathbf{x}).$$

Summarizing, the matrices W_R and W_C read, in the respective bases $\{\mathbf{r}_\alpha\}_{\alpha=1,2,3}$, $\{\zeta_i\}_{i=1,\dots,2n^E}$ and $\{\mathbf{c}_\alpha\}_{\alpha=1,2,3}$, $\{\zeta_i\}_{i=1,\dots,2n^E}$:

$$W_R = \begin{pmatrix} \vdots & \vdots & \vdots \\ \frac{1}{n^E} & 0 & -\frac{1}{2}(a_i)_2 \\ 0 & \frac{1}{n^E} & \frac{1}{2}(a_i)_1 \\ \vdots & \vdots & \vdots \end{pmatrix}, \text{ and } W_C = \begin{pmatrix} \vdots & \vdots & \vdots \\ (a_i)_1 & 0 & \frac{1}{2}(a_i)_2 \\ 0 & (a_i)_2 & \frac{1}{2}(a_i)_1 \\ \vdots & \vdots & \vdots \end{pmatrix}.$$

Assembly of the matrices N_C, N_R

It follows from the definition (5.15) and (5.16) of the functions ζ_i that:

$$\mathbf{c}_\alpha(\mathbf{x}) = \sum_{i=1}^{n^E} (c_\alpha(\mathbf{q}_i^E))_1 \zeta_{2i-1}(\mathbf{x}) + \sum_{i=1}^{n^E} (c_\alpha(\mathbf{q}_i^E))_2 \zeta_{2i}(\mathbf{x}), \quad \mathbf{x} \in E.$$

Hence, a simple calculation based on the explicit form (5.18) of the basis functions \mathbf{c}_j reveals that:

$$N_C = \begin{pmatrix} \vdots & \vdots & \vdots \\ (c_1(\mathbf{q}_i^E))_1 & (c_2(\mathbf{q}_i^E))_1 & (c_3(\mathbf{q}_i^E))_1 \\ (c_1(\mathbf{q}_i^E))_2 & (c_2(\mathbf{q}_i^E))_2 & (c_3(\mathbf{q}_i^E))_2 \\ \vdots & \vdots & \vdots \end{pmatrix} = \begin{pmatrix} \vdots & \vdots & \vdots \\ q_{i,1}^E - \overline{q_1^E} & 0 & q_{i,2}^E - \overline{q_2^E} \\ 0 & q_{i,2}^E - \overline{q_2^E} & q_{i,1}^E - \overline{q_1^E} \\ \vdots & \vdots & \vdots \end{pmatrix}.$$

Similarly, the following expression of N_R is derived:

$$N_R = \begin{pmatrix} \vdots & \vdots & \vdots \\ 1 & 0 & -q_{i,2}^E - \overline{q_2^E} \\ 0 & 1 & q_{i,1}^E - \overline{q_1^E} \\ \vdots & \vdots & \vdots \end{pmatrix}.$$

Calculation of α^E

As we have mentioned, the coefficient α^E is chosen so as to ensure the stability of the method. The idea consists in taking α^E such that the block $\{s^E(\mathbf{c}_k, \mathbf{c}_l)\}_{k,l=1,2,3}$, corresponding to the application of the stabilizing form \tilde{s}^E to constant strain fields, scale like $\{a^E(\mathbf{c}_k, \mathbf{c}_l)\}_{k,l=1,2,3}$, which is exactly the matrix D , when the mesh \mathcal{T} is refined. Hence, we require that $\tilde{s}^E(\mathbf{c}_k, \mathbf{c}_l) = \alpha^E (N_C^T N_C)_{kl}$ be comparable with D_{kl} , which motivates the choice

$$\alpha^E = \frac{\text{tr}(D)}{\text{tr}(N_C^T N_C)}.$$

Eventually, let us mention that the approximation of the local force vector $F^E \in \mathbb{R}^{2n^E}$ whose entries equal

$$F_i^E = \int_E \mathbf{f} \cdot \boldsymbol{\zeta}_i \, d\mathbf{x} + \int_{\partial E \cap \Gamma_N} \mathbf{g} \cdot \boldsymbol{\zeta}_i \, dl, \quad i = 1, \dots, 2n^E$$

is realized in a completely similar fashion as in the case of the conductivity equation, see [Appendix D.1](#).

D.3. Addition of a term of order 0

For the sake of simplicity and without loss of generality, let us return to the setting of the conductivity equation (Cond) of [Section 2.2.1](#). In multiple applications, starting from the solution of eigenvalue problems of the form (2.5) (see [Section 7.2](#) for a numerical example), the numerical implementation requires to calculate the mass matrix $M_{\mathcal{T}} \in \mathbb{R}^{M \times M}$ in addition to the stiffness matrix $K_{\mathcal{T}}$ in (5.9). This matrix $M_{\mathcal{T}}$ is defined by:

$$\forall k, l = 1, \dots, M, \quad M_{\mathcal{T},kl} = \int_{\Omega} \varphi_k \varphi_l \, d\mathbf{x},$$

where the basis $\{\varphi_k\}_{k=1,\dots,M}$ of the virtual element space $\mathcal{W}_{\mathcal{T}}$ is that characterized by (5.7). Arguing as in [Section 5.2.1](#), $M_{\mathcal{T}}$ is assembled from the local contributions $M^E \in \mathbb{R}^{n^E \times n^E}$ attached to the individual elements $E \in \mathcal{T}$, defined by:

$$(D.8) \quad M_{ij}^E = \int_E \zeta_i \zeta_j \, d\mathbf{x}.$$

To achieve this computation, we introduce the L^2 projector $\pi_{\mathcal{P}}^0 : \mathcal{W}(E) \rightarrow \mathcal{P}(E)$ over $\mathcal{P}(E)$, defined by

$$\forall m \in \mathcal{P}(E), \quad \int_E (\pi_{\mathcal{P}}^0 \zeta) m \, d\mathbf{x} = \int_E \zeta m \, d\mathbf{x}.$$

Unfortunately, this operator cannot be expressed in closed form, as the last term in the above right-hand side cannot be readily computed from the degrees of freedom of the function ζ .

To overcome this issue, we leverage an elegant observation from [1] whereby a slight modification of the local space $\mathcal{W}(E)$ in [Section 5.2.2](#), which is completely transparent in the numerical implementation, enables this computation at no additional cost. Let us first define the larger space $\widetilde{\mathcal{W}}(E)$ of local functions attached to E by:

$$\widetilde{\mathcal{W}}(E) = \left\{ \zeta : E \rightarrow \mathbb{R}, \zeta \text{ is affine on each edge of } \partial E, -\Delta \zeta \text{ is an affine function on } E \right\}.$$

We note that the projection operator $\pi_{\mathcal{P}} : \mathcal{W}(E) \rightarrow \mathcal{P}(E)$ introduced in (5.13) can actually be extended as $\pi_{\mathcal{P}} : \widetilde{\mathcal{W}}(E) \rightarrow \mathcal{P}(E)$; actually, for a given function $\zeta \in \widetilde{\mathcal{W}}(E)$, the same calculations as in the previous

Appendix D.1 reveal that $\pi_{\mathcal{P}}\zeta$ can still be calculated from the sole values of ζ at the vertices of E . We now introduce the subspace $\mathcal{Z}(E) \subset \widetilde{\mathcal{W}}(E)$ defined by:

$$(D.9) \quad \mathcal{Z}(E) = \left\{ \zeta : E \rightarrow \mathbb{R}, \zeta \text{ is affine on each edge of } \partial E, -\Delta\zeta \text{ is affine on } E, \text{ and} \right. \\ \left. \forall m \in \mathcal{P}(E), \int_E \zeta m \, d\mathbf{x} = \int_E (\pi_{\mathcal{P}}\zeta)m \, d\mathbf{x} \right\}.$$

The key remark about $\mathcal{Z}(E)$ is that its elements can be characterized by the exact same degrees of freedom as those of the space $\mathcal{W}(E)$. Before proceeding, let us provide a simple technical fact:

Lemma D.1. *The mapping $L := \pi_{\mathcal{P}}^0 \circ \Delta^{-1} : \mathcal{P}(E) \rightarrow \mathcal{P}(E)$ is an isomorphism, where Δ stands for the Laplace operator on E with homogeneous Dirichlet boundary conditions.*

Proof. Since $\mathcal{P}(E)$ is a finite-dimensional space, it is enough to prove that L is injective. Let then $m \in \mathcal{P}(E)$ be such that $Lm = 0$, and let $q = \Delta^{-1}m \in H_0^1(E)$ be the function defined by

$$\forall w \in H_0^1(E), \quad \int_E \nabla q \cdot \nabla w \, d\mathbf{x} = \int_E mw \, d\mathbf{x}.$$

It follows:

$$0 = \int_E Lm m \, d\mathbf{x} = \int_E q m \, d\mathbf{x} = \int_E |\nabla q|^2 \, d\mathbf{x}.$$

Hence, $q = 0$, and so $m = 0$, as expected. \square

The result of interest is now the following.

Lemma D.2. *For all $i = 1, \dots, n^E$, there exists a unique function $\zeta_i \in \mathcal{Z}(E)$ such that*

$$(D.10) \quad \zeta_i(\mathbf{q}_j^E) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

Proof. Let $i \in \{1, \dots, n^E\}$ be an arbitrary index; we first assume that there exists one function $\zeta \in \mathcal{Z}(E)$ satisfying (D.10), and we prove that it is necessarily unique. To this end, let $m := \Delta\zeta \in \mathcal{P}(E)$, and let us introduce the unique function $\tilde{\zeta}_i \in \mathcal{W}(E)$ such that $\tilde{\zeta}_i(\mathbf{q}_j^E) = 1$ if $i = j$ and 0 otherwise, see [Section 5.2.2](#). As noted in [Remark D.1](#), the functions $\pi_{\mathcal{P}}\zeta$ and $\pi_{\mathcal{P}}\tilde{\zeta}_i$ coincide. Then, introducing the basis $\{m_\alpha\}_{\alpha=1,2,3}$ of $\mathcal{P}(E)$ defined in [Section 5.2.2](#), the difference $r = \zeta - \tilde{\zeta}_i$ satisfies:

$$\begin{aligned} r = 0 \text{ on } \partial E, \quad \Delta r = m, \text{ and for } \alpha = 1, 2, 3, \quad \int_E r m_\alpha \, d\mathbf{x} &= \int_E \zeta m_\alpha \, d\mathbf{x} - \int_E \tilde{\zeta}_i m_\alpha \, d\mathbf{x}, \\ &= \int_E (\pi_{\mathcal{P}}\zeta - \pi_{\mathcal{P}}^0 \tilde{\zeta}_i) m_\alpha \, d\mathbf{x} \\ &= \int_E (\pi_{\mathcal{P}}\tilde{\zeta}_i - \pi_{\mathcal{P}}^0 \tilde{\zeta}_i) m_\alpha \, d\mathbf{x}, \end{aligned}$$

where the second line follows from the definition (D.9) of $\mathcal{Z}(E)$ and the final line is a consequence of the equality between $\pi_{\mathcal{P}}\zeta$ and $\pi_{\mathcal{P}}\tilde{\zeta}_i$. It follows from this discussion and the previous [Lemma D.1](#) that:

$$\Delta\zeta = L^{-1} \left(\pi_{\mathcal{P}}\tilde{\zeta}_i - \pi_{\mathcal{P}}^0 \tilde{\zeta}_i \right).$$

Hence, ζ is the solution to the boundary value problem:

$$\begin{cases} -\Delta\zeta = -L^{-1}(\pi_{\mathcal{P}}\tilde{\zeta}_i - \pi_{\mathcal{P}}^0 \tilde{\zeta}_i) & \text{in } E, \\ \zeta = \tilde{\zeta}_i & \text{on } \partial E, \end{cases}$$

so that the function ζ satisfying (D.10) is necessarily unique.

Conversely, defining ζ by the above formula immediately yields one function $\zeta \in \mathcal{Z}(E)$ satisfying (D.10), which terminates the proof. \square

Let us summarize the foregoing discussion:

- The functions pertaining to $\mathcal{W}(E)$ and in $\mathcal{Z}(E)$ can both be characterized by their values at the vertices of E : for any collection of values $\{v_i\}_{i=1,\dots,n^E} \in \mathbb{R}^{n^E}$, there exist unique functions $w \in \mathcal{W}(E)$ and $z \in \mathcal{Z}(E)$ such that:

$$w(\mathbf{q}_i^E) = v_i \text{ and } z(\mathbf{q}_i^E) = v_i.$$

Both functions w and z are different, but, as noted in [Remark D.1](#), their projections onto affine functions coincide:

$$\pi_{\mathcal{P}}z = \pi_{\mathcal{P}}w.$$

- By definition, the projectors $\pi_{\mathcal{P}}$ and $\pi_{\mathcal{P}}^0$ coincide on the space $\mathcal{Z}(E)$:

$$\forall \zeta \in \mathcal{Z}(E), \quad \pi_{\mathcal{P}}^0 \zeta = \pi_{\mathcal{P}} \zeta.$$

Introducing the basis $\{\zeta_i\}_{i=1,\dots,n^E}$ of the local space $\mathcal{Z}(E)$ defined in [Lemma D.2](#), the local mass matrix M^E expressed in terms of this basis reads as in [\(D.8\)](#). Like in the previous sections, we may approximate this matrix as:

$$\forall i, j = 1, \dots, n^E \quad M_{ij}^E = \int_E (\pi_{\mathcal{P}}^0 \zeta_i)(\pi_{\mathcal{P}}^0 \zeta_j) \, d\mathbf{x} + \int_E (\zeta_i - \pi_{\mathcal{P}}^0 \zeta_i)(\zeta_j - \pi_{\mathcal{P}}^0 \zeta_j) \, d\mathbf{x}.$$

Here, the first contribution can be calculated exactly on account of the previous observations; as for the second one, it can be replaced by a stabilization term, in the same spirit as in [Section 5.2.2](#).

REFERENCES

- [1] B. AHMAD, A. ALSAEDI, F. BREZZI, L. D. MARINI, AND A. RUSSO, *Equivalent projectors for virtual element methods*, *Computers & Mathematics with Applications*, 66 (2013), pp. 376–391.
- [2] B. ALLAIN, J.-S. FRANCO, AND E. BOYER, *An efficient volumetric framework for shape tracking*, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 268–276.
- [3] G. ALLAIRE, *Shape optimization by the homogenization method*, vol. 146, Springer Science & Business Media, 2002.
- [4] G. ALLAIRE, C. DAPOGNY, AND P. FREY, *Topology and geometry optimization of elastic structures by exact deformation of simplicial mesh*, *Comptes Rendus Mathématique*, 349 (2011), pp. 999–1003.
- [5] ———, *A mesh evolution algorithm based on the level set method for geometry and topology optimization*, *Structural and Multidisciplinary Optimization*, 48 (2013), pp. 711–715.
- [6] ———, *Shape optimization with a level set based mesh evolution method*, *Computer Methods in Applied Mechanics and Engineering*, 282 (2014), pp. 22–53.
- [7] G. ALLAIRE, C. DAPOGNY, AND F. JOUVE, *Shape and topology optimization*, in *Geometric partial differential equations, part II*, A. Bonito and R. Nochetto eds., *Handbook of Numerical Analysis*, vol. 22, (2021), pp. 1–132.
- [8] G. ALLAIRE, F. JOUVE, AND A.-M. TOADER, *Structural optimization using sensitivity analysis and a level-set method*, *Journal of computational physics*, 194 (2004), pp. 363–393.
- [9] G. ALLAIRE AND M. SCHOENAUER, *Conception optimale de structures*, vol. 58, Springer, 2007.
- [10] S. AMSTUTZ, *An introduction to the topological derivative*, *Engineering Computations*, (2021).
- [11] P. F. ANTONIETTI, M. BRUGGI, S. SCACCHI, AND M. VERANI, *On the virtual element method for topology optimization on polygonal meshes: A numerical study*, *Computers & Mathematics with Applications*, 74 (2017), pp. 1091–1109.
- [12] P. F. ANTONIETTI, L. B. DA VEIGA, AND G. MANZINI, *The Virtual Element Method and its Applications*, vol. 31, Springer Nature, 2022.
- [13] P. R. ANTUNES AND P. FREITAS, *Numerical optimization of low eigenvalues of the dirichlet and neumann laplacians*, *Journal of Optimization Theory and Applications*, 154 (2012), pp. 235–257.
- [14] F. AURENHAMMER, *Power diagrams: properties, algorithms and applications*, *SIAM Journal on Computing*, 16 (1987), pp. 78–96.
- [15] L. BEIRÃO DA VEIGA, F. BREZZI, L. D. MARINI, AND A. RUSSO, *The hitchhiker’s guide to the virtual element method*, *Mathematical models and methods in applied sciences*, 24 (2014), pp. 1541–1573.
- [16] Z. BELHACHMI, D. BUCUR, G. BUTTAZZO, AND J.-M. SAC-EPÉE, *Shape optimization problems for eigenvalues of elliptic operators*, *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik: Applied Mathematics and Mechanics*, 86 (2006), pp. 171–184.
- [17] M. P. BENDSOE AND O. SIGMUND, *Topology optimization: theory, methods, and applications*, Springer Science & Business Media, 2013.
- [18] K. BERBATOV, B. LAZAROV, AND A. JIVKOV, *A guide to the finite and virtual element methods for elasticity*, *Applied Numerical Mathematics*, 169 (2021), pp. 351–395.
- [19] M. BERGGREN, *A unified discrete–continuous sensitivity analysis method for shape optimization*, in *Applied and Numerical Partial Differential Equations: Scientific Computing in Simulation, Optimization and Control in a Multidisciplinary Context*, Springer, 2009, pp. 25–39.
- [20] E. BIRGIN AND T. MENEZES, *Sensitivity analysis and tailored design of minimization diagrams*, *Mathematics of Computation*, 92 (2023), pp. 2715–2768.
- [21] E. G. BIRGIN, A. LAURAIN, AND D. R. SOUZA, *Reconstruction of voronoi diagrams in inverse potential problems*, (2024).

- [22] L. BLANK, H. GARCKE, L. SARBU, T. SRISUPATTARAWANIT, V. STYLES, AND A. VOIGT, *Phase-field approaches to structural topology optimization*, in *Constrained optimization and optimal control for partial differential equations*, Springer, 2012, pp. 245–256.
- [23] J. F. BONNANS AND A. SHAPIRO, *Perturbation analysis of optimization problems*, Springer Science & Business Media, 2013.
- [24] B. BOOTS, K. SUGIHARA, S. N. CHIU, AND A. OKABE, *Spatial tessellations: concepts and applications of Voronoi diagrams*, John Wiley & Sons, 2009.
- [25] T. BORRVALL AND J. PETERSSON, *Topology optimization of fluids in stokes flow*, *International journal for numerical methods in fluids*, 41 (2003), pp. 77–107.
- [26] B. BOURDIN AND A. CHAMBOLLE, *Design-dependent loads in topology optimization*, *ESAIM: Control, Optimisation and Calculus of Variations*, 9 (2003), pp. 19–48.
- [27] D. P. BOURNE AND S. M. ROPER, *Centroidal power diagrams, lloyd’s algorithm, and applications to optimal location problems*, *SIAM Journal on Numerical Analysis*, 53 (2015), pp. 2545–2569.
- [28] A. BOWYER, *Computing dirichlet tessellations*, *Comput. J.*, 24 (1981), pp. 162–166.
- [29] D. BUCUR AND P. FREITAS, *A free boundary approach to the faber-krahn inequality*, *Geometric and Computational Spectral Theory*, 700 (2015), pp. 73–86.
- [30] M. BURGER, *A framework for the construction of level set methods for shape optimization and reconstruction*, *Interfaces and Free boundaries*, 5 (2003), pp. 301–329.
- [31] L. CAFFARELLI AND V. OLIKER, *Weak solutions of one inverse problem in geometric optics*, *Journal of Mathematical Sciences*, 154 (2008), pp. 39–49.
- [32] A. CANGIANI, E. H. GEORGIOULIS, AND P. HOUSTON, *hp-version discontinuous galerkin methods on polygonal and polyhedral meshes*, *Mathematical Models and Methods in Applied Sciences*, 24 (2014), pp. 2009–2041.
- [33] A. CEBULA AND D. TALER, *Finite Volume Method in Heat Conduction*, Springer Netherlands, 2014, pp. 1645–1658.
- [34] H. CHI, A. PEREIRA, I. F. MENEZES, AND G. H. PAULINO, *Virtual element method (vem)-based topology optimization: an integrated framework*, *Structural and Multidisciplinary Optimization*, 62 (2020), pp. 1089–1114.
- [35] E. B. CHIN AND N. SUKUMAR, *Scaled boundary cubature scheme for numerical integration over planar regions with affine and curved boundaries*, *Computer Methods in Applied Mechanics and Engineering*, 380 (2021), p. 113796.
- [36] A. N. CHRISTIANSEN, M. NOBEL-JORGENSEN, N. AAGE, O. SIGMUND, AND J. A. BÆRENTZEN, *Topology optimization using an explicit interface representation*, *Structural and Multidisciplinary Optimization*, 49 (2014), pp. 387–399.
- [37] P. G. CIARLET, *The finite element method for elliptic problems*, vol. 40, Siam, 2002.
- [38] J. COATLÉVEN, *Principles of a network element method*, *Journal of Computational Physics*, 433 (2021), p. 110197.
- [39] ———, *Network element methods for linear elasticity*, *Comptes Rendus. Mécanique*, 351 (2023), pp. 1–26.
- [40] L. B. DA VEIGA, F. BREZZI, AND L. D. MARINI, *Virtual elements for linear elasticity problems*, *SIAM Journal on Numerical Analysis*, 51 (2013), pp. 794–812.
- [41] C. DAPOGNY AND F. FEPPON, *Shape optimization using a level set based mesh evolution method: an overview and tutorial*, *Hal preprint: <https://hal.archives-ouvertes.fr/hal-03881641>*, (2022).
- [42] F. DASSI, *Vem++, a c++ library to handle and play with the virtual element method*, *arXiv preprint arXiv:2310.05748*, (2023).
- [43] P. M. M. DE CASTRO, Q. MÉRIGOT, AND B. THIBERT, *Far-field reflector problem and intersection of paraboloids*, *Numerische Mathematik*, 134 (2016), pp. 389–411.
- [44] F. DE GOES, K. BREEDEN, V. OSTROMOUKHOV, AND M. DESBRUN, *Blue noise through optimal transport*, *ACM Transactions on Graphics (TOG)*, 31 (2012), pp. 1–11.
- [45] F. DE GOES, C. WALLEZ, J. HUANG, D. PAVLOV, AND M. DESBRUN, *Power particles: an incompressible fluid solver based on power diagrams*, *ACM Transactions on Graphics*, 34 (2015), pp. Art–No.
- [46] F. DE GOURNAY, *Velocity extension for the level-set method and multiple eigenvalues in shape optimization*, *SIAM journal on control and optimization*, 45 (2006), pp. 343–367.
- [47] F. DE GOURNAY, J. KAHN, AND L. LEBRAT, *Differentiation and regularity of semi-discrete optimal transport with respect to the parameters of the discrete measure*, *Numerische Mathematik*, 141 (2019), pp. 429–453.
- [48] A. DELALANDE AND Q. MERIGOT, *Quantitative stability of optimal transport maps under variations of the target measure*, *arXiv preprint arXiv:2103.05934*, (2021).
- [49] M. C. DELFOUR AND J.-P. ZOLÉSIO, *Shapes and geometries: metrics, analysis, differential calculus, and optimization*, SIAM, 2011.
- [50] D. DEMIDOV, *Amgcl: An efficient, flexible, and extensible algebraic multigrid implementation*, *Lobachevskii J. of Math.*, 40 (2019), pp. 535–546.
- [51] B. DESPRÉS, *Lagrangian voronoi meshes and particle dynamics with shocks*, *Computer Methods in Applied Mechanics and Engineering*, 418 (2024), p. 116427.
- [52] H. EDELSBRUNNER AND E. P. MÜCKE, *Simulation of simplicity: a technique to cope with degenerate cases in geometric algorithms*, *ACM Trans. Graph.*, 9 (1990), pp. 66–104.
- [53] F. FENG, S. XIONG, Z. LIU, Z. XIAN, Y. ZHOU, H. KOBAYASHI, A. KAWAMOTO, T. NOMURA, AND B. ZHU, *Cellular topology optimization on differentiable voronoi diagrams*, *arXiv preprint arXiv:2204.10313*, (2022).
- [54] F. FEPPON, G. ALLAIRE, F. BORDEU, J. CORTIAL, AND C. DAPOGNY, *Shape optimization of a coupled thermal fluid-structure problem in a level set mesh evolution framework*, *SeMA Journal*, (2019), pp. 1–46.

- [55] F. FEPPON, G. ALLAIRE, AND C. DAPOGNY, *Null space gradient flows for constrained optimization with applications to shape optimization*, ESAIM: Control, Optimisation and Calculus of Variations, 26 (2020), p. 90.
- [56] R. FERNANDES, E. GEORGIOULIS, AND A. PAGANINI, *Level-set shape optimization via polytopic discontinuous galerkin methods*, (2024).
- [57] N. FERRO, S. MICHELETTI, N. PAROLINI, S. PEROTTO, M. VERANI, AND P. F. ANTONIETTI, *Level set-fitted polytopal meshes with application to structural topology optimization*, arXiv preprint arXiv:2309.11389, (2023).
- [58] P. J. FREY AND P.-L. GEORGE, *Mesh generation: application to finite elements*, ISTE, 2007.
- [59] E. GABURRO, W. BOSCHERI, S. CHIOCCHETTI, C. KLINGENBERG, V. SPRINGEL, AND M. DUMBSER, *High order direct arbitrary-lagrangian-eulerian schemes on moving voronoi meshes with topology changes*, Journal of Computational Physics, 407 (2020), p. 109167.
- [60] A. L. GAIN, C. TALISCHI, AND G. H. PAULINO, *On the virtual element method for three-dimensional linear elasticity problems on arbitrary polyhedral meshes*, Computer Methods in Applied Mechanics and Engineering, 282 (2014), pp. 132–160.
- [61] S. GALINDO-TORRES AND D. PEDROSO, *Molecular dynamics simulations of complex-shaped particles using voronoi-based spheropolyhedra*, Physical Review E, 81 (2010), p. 061303.
- [62] F. GARDINI AND G. VACCA, *Virtual element method for second-order elliptic eigenvalue problems*, IMA Journal of Numerical Analysis, 38 (2018), pp. 2026–2054.
- [63] S. GARREAU, P. GUILLAUME, AND M. MASMOUDI, *The topological asymptotic for pde systems: the elasticity case*, SIAM journal on control and optimization, 39 (2001), pp. 1756–1778.
- [64] M. GIACOMINI, O. PANTZ, AND K. TRABELSI, *Volumetric expressions of the shape gradient of the compliance in structural shape optimization*, arXiv preprint arXiv:1701.05762, (2017).
- [65] R. GLOWINSKI AND J. HE, *On shape optimization and related issues*, in Computational Methods for Optimal Design and Control: Proceedings of the AFOSR Workshop on Optimal Design and Control Arlington, Virginia 30 September–3 October, 1997, Springer, 1998, pp. 151–179.
- [66] P. L. GOULD AND Y. FENG, *Introduction to linear elasticity*, Springer, 1994.
- [67] M. D. GUNZBURGER, *Perspectives in flow control and optimization*, SIAM, 2002.
- [68] A. HENROT, *Extremum problems for eigenvalues of elliptic operators*, Springer Science & Business Media, 2006.
- [69] A. HENROT AND M. PIERRE, *Shape Variation and Optimization*, EMS Tracts in Mathematics Vol. 28, 2018.
- [70] M. R. HESTENES AND E. STIEFEL, *Methods of conjugate gradients for solving linear systems*, J Res NIST, 49 (1952), pp. 409–436.
- [71] R. HIPTMAIR, A. PAGANINI, AND S. SARGHEINI, *Comparison of approximate shape gradients*, BIT Numerical Mathematics, 55 (2015), pp. 459–485.
- [72] H. JASAK AND H. WELLER, *Application of the finite volume method and unstructured meshes to linear elasticity*, International journal for numerical methods in engineering, 48 (2000), pp. 267–287.
- [73] J. KITAGAWA, Q. MÉRIGOT, AND B. THIBERT, *A newton algorithm for semi-discrete optimal transport*, CoRR, abs/1603.05579 (2016).
- [74] A. LAURAIN, *Analysis and application of a lower envelope method for sharp-interface multiphase problems*, arXiv preprint arXiv:2112.02401, (2021).
- [75] B. LÉVY, *A numerical algorithm for $L_{\{2\}}$ semi-discrete optimal transport in 3d*, ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique, 49 (2015), pp. 1693–1715.
- [76] B. LÉVY, *Robustness and efficiency of geometric programs: The predicate construction kit (PCK)*, Computer-Aided Design, 72 (2016), pp. 3–12.
- [77] B. LÉVY, Y. BRENIER, AND R. MOHAYAEI, *Monge amp\ere gravity: from the large deviation principle to cosmological simulations through optimal transport*, arXiv preprint arXiv:2404.07697, (2024).
- [78] B. LÉVY AND Y. LIU, *L^p centroidal voronoi tessellation and its applications*, ACM Transactions on Graphics (TOG), 29 (2010), pp. 1–11.
- [79] B. LÉVY AND E. L. SCHWINDT, *Notions of optimal transport theory and how to implement them on a computer*, Computers & Graphics, 72 (2018), pp. 135–148.
- [80] M. LI, J. HU, W. CHEN, W. KONG, AND J. HUANG, *Explicit topology optimization of conforming voronoi foams*, arXiv preprint arXiv:2308.04001, (2023).
- [81] J. LIONS, *Optimal control of systems governed by partial differential equations*, Grundlehren der mathematischen Wissenschaften, Springer-Verlag, 1971.
- [82] S. LLOYD, *Least squares quantization in pcm*, IEEE transactions on information theory, 28 (1982), pp. 129–137.
- [83] B. LÉVY, *Partial optimal transport for a constant-volume lagrangian mesh with free boundaries*, Journal of Computational Physics, 451 (2022), p. 110838.
- [84] ———, *Large-scale semi-discrete optimal transport with distributed voronoi diagrams*, 2024.
- [85] C. MANTEGAZZA, *Lecture notes on mean curvature flow*, vol. 290, Springer Science & Business Media, 2011.
- [86] M. MENGOLINI, M. F. BENEDETTO, AND A. M. ARAGÓN, *An engineering perspective to the virtual element method and its interplay with the standard finite element method*, Computer Methods in Applied Mechanics and Engineering, 350 (2019), pp. 995–1023.
- [87] Q. MÉRIGOT, A. DELALANDE, AND F. CHAZAL, *Quantitative stability of optimal transport maps and linearization of the 2-wasserstein space*, in International Conference on Artificial Intelligence and Statistics, PMLR, 2020, pp. 3186–3196.

- [88] Q. MERIGOT AND B. THIBERT, *Optimal transport: discretization and algorithms*, in Handbook of Numerical Analysis, vol. 22, Elsevier, 2021, pp. 133–212.
- [89] G. MICHAÏLIDIS, *Manufacturing constraints and multi-phase shape and topology optimization via a level-set method*, PhD thesis, 2014.
- [90] B. MOHAMMADI AND O. PIRONNEAU, *Applied shape optimization for fluids*, Oxford university press, 2010.
- [91] F. MURAT AND J. SIMON, *Sur le contrôle par un domaine géométrique*, Pré-publication du Laboratoire d'Analyse Numérique,(76015), (1976).
- [92] A. NEALEN, M. MÜLLER, R. KEISER, E. BOXERMAN, AND M. CARLSON, *Physically based deformable models in computer graphics*, in Computer graphics forum, vol. 25, Wiley Online Library, 2006, pp. 809–836.
- [93] B.-T. NGUYEN, *Geometrical structure of laplacian eigenfunctions*, siam REVIEW, 55 (2013), pp. 601–667.
- [94] S. H. NGUYEN AND H.-G. KIM, *Level set based shape optimization using trimmed hexahedral meshes*, Computer Methods in Applied Mechanics and Engineering, 345 (2019), pp. 555–583.
- [95] S. H. NGUYEN, D. SOHN, AND H.-G. KIM, *A novel hr-adaptive mesh refinement scheme for stress-constrained shape and topology optimization using level-set-based trimmed meshes*, Structural and Multidisciplinary Optimization, 65 (2022), p. 71.
- [96] F. NIKAKHTAR, R. K. SHETH, B. LÉVY, AND R. MOHAYAEI, *Optimal transport reconstruction of baryon acoustic oscillations*, Physical Review Letters, 129 (2022), p. 251101.
- [97] F. NIKAKHTAR, R. K. SHETH, N. PADMANABHAN, B. LÉVY, AND R. MOHAYAEI, *Displacement field analysis via optimal transport: Multi-tracer approach to cosmological reconstruction*, arXiv preprint arXiv:2403.11951, (2024).
- [98] V. NIVOLIERI AND B. LÉVY, *Approximating functions on a mesh with restricted voronoi diagrams*, in Computer Graphics Forum, vol. 32, Wiley Online Library, 2013, pp. 83–92.
- [99] A. A. NOVOTNY AND J. SOKOŁOWSKI, *Topological derivatives in shape optimization*, Springer Science & Business Media, 2012.
- [100] S. OSHER AND R. FEDKIW, *Level set methods and dynamic implicit surfaces*, vol. 153, Springer Science & Business Media, 2006.
- [101] S. J. OSHER AND F. SANTOSA, *Level set methods for optimization problems involving geometry and constraints: I. frequencies of a two-density inhomogeneous drum*, Journal of Computational Physics, 171 (2001), pp. 272–288.
- [102] R. OSSERMAN, *The isoperimetric inequality*, Bulletin of the American Mathematical Society, 84 (1978), pp. 1182–1238.
- [103] É. OUDET, *Numerical minimization of eigenmodes of a membrane with respect to the domain*, ESAIM: Control, Optimization and Calculus of Variations, 10 (2004), pp. 315–330.
- [104] G. PEYRÉ, M. CUTURI, ET AL., *Computational optimal transport: With applications to data science*, Foundations and Trends in Machine Learning, 11 (2019), pp. 355–607.
- [105] O. PIRONNEAU, *Optimal shape design for elliptic systems*, Springer, 1982.
- [106] R.-E. PLESSIX, *A review of the adjoint-state method for computing the gradient of a functional with geophysical applications*, Geophysical Journal International, 167 (2006), pp. 495–503.
- [107] G. PÓLYA AND G. SZEGÖ, *Isoperimetric inequalities in mathematical physics*, no. 27, Princeton University Press, 1951.
- [108] Z. QU, M. LI, F. DE GOES, AND C. JIANG, *The power particle-in-cell method*, ACM Transactions on Graphics (TOG), 41 (2022), pp. 1–13.
- [109] J. W. S. B. RAYLEIGH, *The theory of sound*, vol. 2, Macmillan, 1896.
- [110] R. T. ROCKAFELLAR, *Convex analysis*, vol. 18, Princeton university press, 1970.
- [111] G. SAN-VICENTE-OTAMENDI, *Designing deformable models of soft tissue for virtual surgery planning and simulation using the Mass-Spring Model*, PhD thesis, Universidad de Navarra, 2011.
- [112] F. SANTAMBROGIO, *Optimal transport for applied mathematicians*, Birkäuser, 2015.
- [113] R. I. SAYE AND J. A. SETHIAN, *The voronoi implicit interface method for computing multiphase physics*, Proceedings of the National Academy of Sciences, 108 (2011), pp. 19498–19503.
- [114] ———, *Analysis and applications of the voronoi implicit interface method*, Journal of Computational Physics, 231 (2012), pp. 6051–6085.
- [115] J. A. SETHIAN, *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*, vol. 3, Cambridge university press, 1999.
- [116] J. A. SETHIAN AND A. WIEGMANN, *Structural boundary design via level set and immersed interface methods*, Journal of computational physics, 163 (2000), pp. 489–528.
- [117] J. R. SHEWCHUK, *Robust adaptive floating-point geometric predicates*, in Proceedings of the Twelfth Annual Symposium on Computational Geometry, Philadelphia, PA, USA, May 24–26, 1996, 1996, pp. 141–150.
- [118] J. SOKOŁOWSKI AND A. ZOCHOWSKI, *On the topological derivative in shape optimization*, SIAM Journal on Control and Optimization, 37 (1999), pp. 1251–1272.
- [119] J. SOKOŁOWSKI AND J.-P. ZOLÉSIO, *Introduction to shape optimization*, Springer, 1992.
- [120] A. SOMMARIVA AND M. VIANELLO, *Gauss–green cubature and moment computation over arbitrary geometries*, Journal of Computational and Applied Mathematics, 231 (2009), pp. 886–896.
- [121] T. SORGENTE, S. BERRONE, S. BIASOTTI, G. MANZINI, M. SPAGNUOLO, AND F. VICINI, *Mesh optimization for the virtual element method: How small can an agglomerated mesh become?*, arXiv preprint arXiv:2404.11484, (2024).
- [122] T. SORGENTE, S. BIASOTTI, G. MANZINI, AND M. SPAGNUOLO, *The role of mesh quality and mesh quality indicators in the virtual element method*, Advances in Computational Mathematics, 48 (2022), p. 3.

- [123] V. SPRINGEL, *E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh*, Monthly Notices of the Royal Astronomical Society, 401 (2010), pp. 791–851.
- [124] ———, *Hydrodynamic simulations on a moving voronoi mesh*, arXiv preprint arXiv:1109.2218, (2011).
- [125] O. J. SUTTON, *The virtual element method in 50 lines of matlab*, Numerical Algorithms, 75 (2017), pp. 1141–1159.
- [126] A. TAKEZAWA, S. NISHIWAKI, AND M. KITAMURA, *Shape and topology optimization based on the phase field method and sensitivity analysis*, Journal of Computational Physics, 229 (2010), pp. 2697–2718.
- [127] C. TALISCHI, G. H. PAULINO, A. PEREIRA, AND I. F. MENEZES, *Polygonal finite elements for topology optimization: A unifying paradigm*, International journal for numerical methods in engineering, 82 (2010), pp. 671–698.
- [128] R. TEMAM AND A. MIRANVILLE, *Mathematical modeling in continuum mechanics*, Cambridge University Press, 2005.
- [129] C. VILLANI, *Optimal transport: old and new*, vol. 338, Springer, 2009.
- [130] M. Y. WANG, X. WANG, AND D. GUO, *A level set method for structural topology optimization*, Computer methods in applied mechanics and engineering, 192 (2003), pp. 227–246.
- [131] W. WANG, D. SLEPCEV, S. BASU, J. A. OZOLEK, AND G. K. ROHDE, *A linear optimal transportation framework for quantifying and visualizing variations in sets of images*, International journal of computer vision, 101 (2013), pp. 254–269.
- [132] D. WATSON, *Computing the n-dimensional delaunay tessellation with application to voronoi polytopes*, Comput. J., 24 (1981), pp. 167–172.
- [133] S.-Q. XIN, B. LÉVY, Z. CHEN, L. CHU, Y. YU, C. TU, AND W. WANG, *Centroidal power diagrams with capacity constraints: Computation, applications, and extension*, ACM Transactions on Graphics (TOG), 35 (2016), pp. 1–12.
- [134] A. ZAITZEFF, S. ESEDOGLU, AND K. GARIKIPATI, *On the voronoi implicit interface method*, SIAM Journal on Scientific Computing, 41 (2019), pp. A2407–A2429.
- [135] X. ZHAI, F. HOU, H. QIN, AND A. HAO, *Fluid simulation with adaptive staggered power particles on gpus*, IEEE Transactions on Visualization and Computer Graphics, 26 (2020), pp. 2234–2246.