Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 282 (2014) 22–53
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

# Shape optimization with a level set based mesh evolution method

G. Allaire[a], C. Dapogny[b,c,*], P. Frey[b,d]

[a] *CMAP, UMR 7641, Ecole Polytechnique, 91128 Palaiseau, France*
[b] *Sorbonne Universités, UPMC Univ Paris 06, UMR 7598, Laboratoire J.-L. Lions, F-75005 Paris, France*
[c] *Renault DREAM-DELT'A Guyancourt, France*
[d] *CNRS, UMR 7598, Laboratoire Jacques-Louis Lions, F-75005, Paris, France*

Received 23 January 2014; received in revised form 18 July 2014; accepted 20 August 2014
Available online 10 September 2014

**Abstract**

In this article, we discuss an approach for geometry and topology optimization of structures which benefits from an accurate description of shapes at each stage of the iterative process – by means of a mesh amenable for mechanical analyses – while retaining the whole versatility of the level set method when it comes to accounting for their evolution. The key ingredients of this method are two operators for switching from a meshed representation of a domain to an implicit one, and conversely; this notably brings into play an algorithm for generating the signed distance function to an arbitrary discrete domain, and a mesh generation algorithm for implicitly-defined geometries.


*Keywords:* Geometry and topology optimization; Level set method; Meshing and remeshing; Signed distance function; Linear elasticity

## 1. Introduction

In the simulation of a free or moving boundary problem driven by a physical motion, one usually has to reconcile numerical accuracy with robustness: the more faithful the representation of the tracked boundary, the more accurate the computation of the motion (i.e. the velocity field driving the motion), and unfortunately, the more tedious the numerical implementation. This issue is especially critical in shape optimization which features problems where the changes in geometry and topology of shapes in the course of the evolution are often dramatic.

Roughly speaking, in the field of shape and topology optimization, three main classes of techniques can be distinguished, depending on the description of shapes they involve:

- *Density-based methods*, such as the SIMP method [1] or the homogenization method [2,3], transform the problem of finding the optimal shape $\Omega \subset \mathbb{R}^d$ with respect to a mechanical criterion $J(\Omega)$ into that of finding the optimal density function $\rho : D \to [0, 1]$ of a mixture of material and void inside a fixed working domain $D$. The shape optimization problem has to be translated into this rather different setting, but the main difficulties are the absence of

* Correspondence to: Laboratoire Jacques-Louis Lions, 4 place Jussieu, 75005 Paris, France. Tel.: +33 13028839748.
*E-mail address:* dapogny@ann.jussieu.fr (C. Dapogny).

http://dx.doi.org/10.1016/j.cma.2014.08.028

a clear definition of the shape boundary and the penalization process which, in the end, should deliver a 'classical' shape without intermediate densities.

- *Eulerian methods*, such as the phase field method [4], or the level set method [5–8] account for shapes in an implicit way; for instance, in the latter case, a large, fixed working domain $D$, meshed once and for all is introduced, and a shape $\Omega \subset D$ is described in terms of a scalar function $\phi : D \to \mathbb{R}$ whose negative subdomain matches with $\Omega$. Finite element analyses cannot be performed directly on $\Omega$ since it is not meshed exactly, and approximations have to be made to trade mechanical problems posed on $\Omega$ for problems posed on $D$. The most notorious of them is the so-called Ersatz material approach, which consists in filling the 'void' $D \setminus \overline{\Omega}$ with a very soft material to avoid degeneracy in the stiffness matrix (however, alternatives exist, which are based on e.g. the immersed boundary method [7], or the XFEM method [9,10]).
- *Lagrangian methods* are perhaps the most natural ones and date back to the early hours of computational structural optimization [11,12]; shapes are represented by means of a computational mesh (or a CAD model [13]), which enables accurate mechanical analyses. The general drawback of this class of methods lies in that this mesh (or whatever explicit representation of shapes is used) has to be updated in the course of the optimization process, which is a notoriously difficult operation, especially in 3d. Note that there is still ongoing research in this direction [14–16].

Of course, this rough classification ignores the numerous particular instances of each category of methods and combinations between them (see the recent review papers [17,18] for a stronger emphasis on level-set based structural optimization).

In the present paper, we describe in details our work, briefly announced in [19,20], and propose a shape optimization strategy which benefits from the flexibility of the level set method for tracking evolution of shapes, including topological changes, while enjoying an exact, meshed description of shapes. In a nutshell, our approach relies on a level set description of a shape under evolution as a guide for driving the usual mesh operators (edge split, edge collapse, etc...) involved in the update of its mesh. It can be regarded as a systematic and robust way of modifying and controlling the level of detail of this mesh. While they rely on similar mesh operations, Lagrangian methods such as the one presented in [15] generally prove more combinatorial (especially when it comes to the case of three space dimensions), since they use heuristics to avoid producing an invalid mesh or to account for topological changes in the shape.

Admittedly, the idea of combining an implicit domain evolution method with an explicit type of shape representation is not new: in the two-dimensional work [21], the evolution of shapes is tracked on a triangular mesh $\mathcal{T}$ of a working domain $D$ owing to the level set method, and at each iteration of the process, an exact mesh of the current shape $\Omega$ is obtained by relocating vertices of $\mathcal{T}$ onto $\partial\Omega$. In [22], a similar strategy is applied for dealing with the motion of shapes; a computational mesh for any shape $\Omega$ arising during the process is then constructed by first identifying the intersection points of the implicitly-defined boundary $\partial\Omega$ with the edges of the computational mesh $\mathcal{T}$ of $D$, then using them as an input for a Delaunay-based mesh generation algorithm. Last but not least, let us mention the work in [23] (Chap. 5), taken over in [24], in which the evolution of shapes is dealt with by using the level set method on a finite difference grid of the working domain $D$, and an original meshing algorithm for implicit geometries is used to get an exact representation of shapes. The precisely calculated shape gradient must then be projected back to the Cartesian grid of $D$ to close the loop.

Our method has something in common with this last work: a computational domain $D$ is defined, and is equipped with an unstructured mesh which is modified from one iteration of the algorithm to the next, in such a way that any shape $\Omega$ arising in the course of the process is *explicitly discretized* in this mesh—i.e. the vertices, edges, faces (and tetrahedra in 3d) of a mesh of $\Omega$ also exist as elements of the ambient mesh of $D$. In such a configuration, we shall also say that (a mesh of) $\Omega$ exists as a *submesh* of that of $D$. This kind of representation allows for accurate finite element analyses, held on a well-defined, high-quality mesh of $\Omega$ (which is possibly adapted to an error estimate for the mechanical problem at stake), and lends itself to the use of the level set method in an unstructured mesh framework, to account for large shape deformations (including possible topological changes). It relies crucially on efficient algorithms for moving back and forth, from a situation where a shape $\Omega$ is known as a submesh of the computational mesh of $D$ to a level set description of $\Omega$ on a (unstructured) mesh of $D$.

This strategy presents several attractive assets; first, no projection between different meshes is needed between the computation of a descent direction for the considered objective function of the domain (which occurs when the