# INTERNSHIP PROPOSAL: TOPOLOGICAL OPTIMIZATION OF GEOMETRIC FILTRATIONS FOR PARAMETRIZED MODELS

**Keywords:** Topological Data Analysis (TDA), Invertible Neural Networks, Neural Ordinary Differential Equation (ODE), Nonconvex Optimization.

**City and Country:** Champs-sur-Marne, France **or** Sophia-Antipolis, France

**Team:** Laboratoire d'Informatique Gaspard Monge, Université Gustave Eiffel **or** DataShape, Centre Inria d'Université Côte d'Azur.

**Internship Advisor:** Théo Lacombe, `theo.lacombe@univ-eiffel.fr` and Mathieu Carrière, `mathieu.carriere@inria.fr`.

**General presentation of the topic:** Topological Data Analysis (TDA) is a set of data science tools rooted in algebraic topology that enables to extract topological descriptors from structured objects, such as graphs, time series, or point clouds. In practice, these topological descriptors depend on the choice of a *filtration*, that is a map $f : \mathcal{X} \to \mathbb{R}$, where $\mathcal{X}$ is the space where the data lives (typically, $\mathcal{X} = \mathbb{R}^d$; theoretically, $\mathcal{X}$ can be any structured topological space such as, e.g, a manifold). The corresponding topological descriptor is a set of points in the Euclidean plane $\mathbb{R}^2$ supported on the upper half-plane $\{(b, d),\ d > b\} \subset \mathbb{R}^2$. It is called the *persistence diagram* $\mathrm{Dgm}(f)$ of the filtration $f$, and each point in this diagram encodes the presence of a topological feature (such as a loop, a cavity, etc) detected in the family $\{F_\alpha\}_\alpha$ of sublevel sets of $f$: $F_\alpha := \{x \in \mathcal{X} : f(x) \leq \alpha\}$ for a wide range of $\alpha$. See Figure 1.

If we let $P = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ denote a $d$-dimensional point cloud, a standard filtration choice is to take $f_P : x \mapsto \mathrm{dist}(x, P)$, that is referred to as the *Rips filtration* of $P$. It is well-known that the map $P \mapsto \mathrm{Dgm}(f_P)$ is Lipschitz [2], hence differentiable almost everywhere. This theoretically allows one to compute a gradient, and thus enables the minimization of objective functions that involve topological terms through gradient descent. For instance, one may want to remove small topological components from the point cloud—considered as noise—, or in contrast to enforce some topological structure to be created in the point cloud, which is doable by optimizing an appropriated loss. See Figure 2.

While this is theoretically appealing and numerically feasible, a practical limitation immediately appears: the gradient of the map $P \mapsto \mathrm{Dgm}(f_P)$ only depends on critical points of $f_P$, which in practice means that at each gradient step, only few points in $P$ are updated, hin-

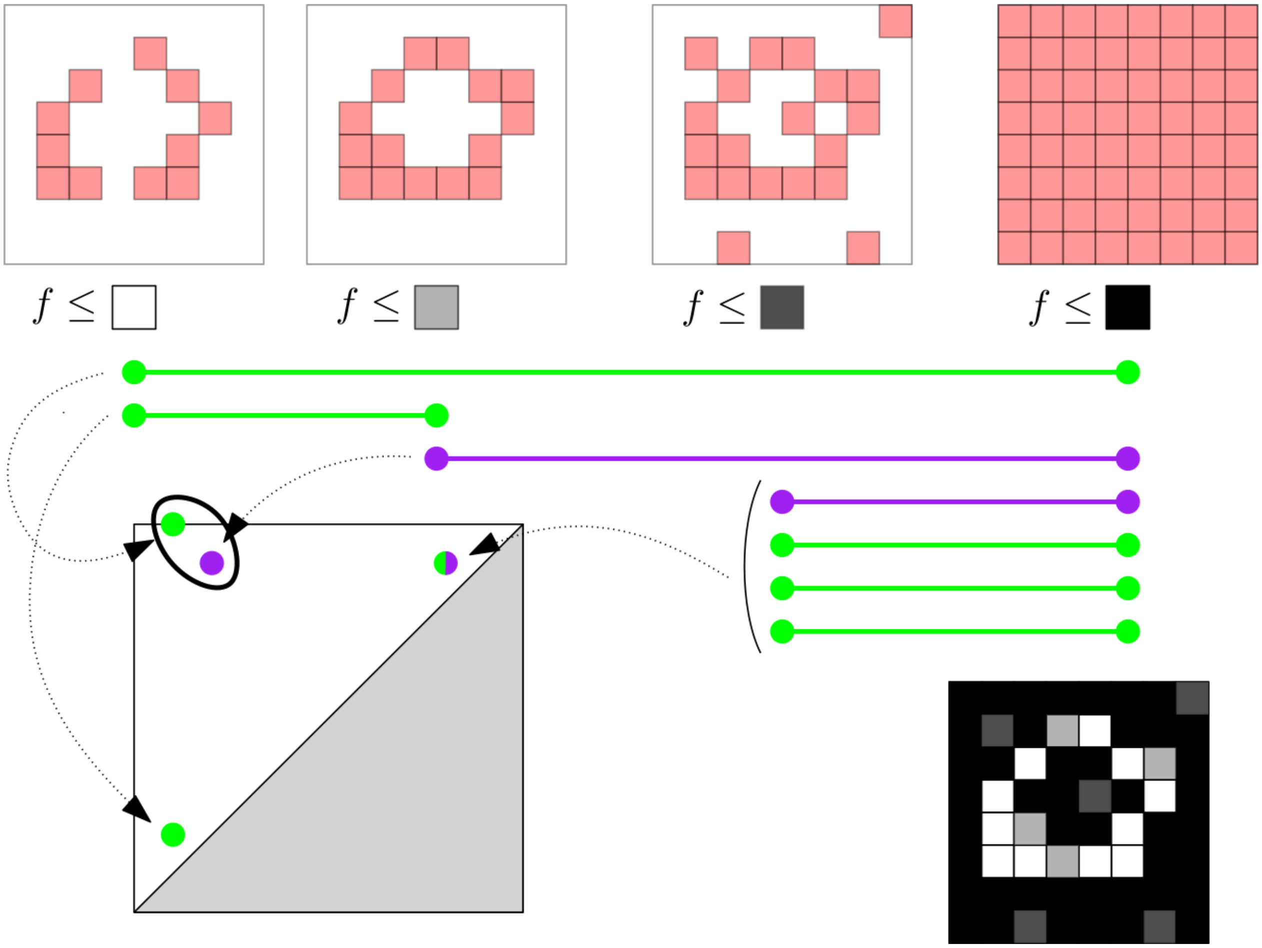


Figure 1: Construction of a persistence diagram from the sublevel sets of the grey scale (shown in pink on the top of the figure) defined on an input image (shown in lower right). Each time a connected component (resp. a loop) appears in the sublevel sets, one creates a new green (resp. purple) bar until it disappears by being merged (resp. filled in). The bars are then represented as above-diagonal points in the Euclidean plane. Points that are far away from the diagonal (circled in the figure) are usually considered relevant signal whereas the others are due to noise and/or numerical artifacts.

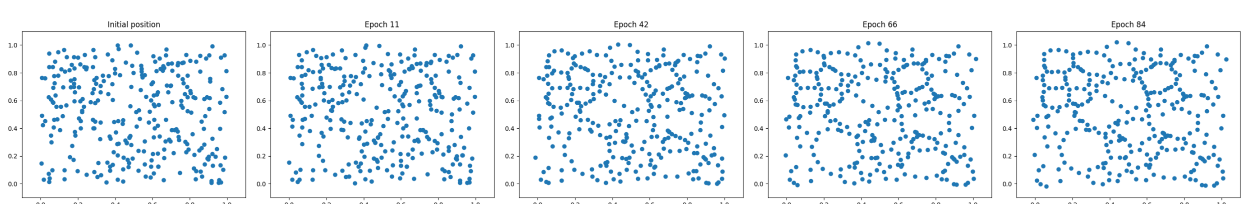


Figure 2: Gradual optimization of a point cloud $P$ so that the number of holes is maximized, which is achieved by maximizing the distances to the diagonal of points in $\mathrm{Dgm}(f_P)$.

dering the computational efficiency of this approach. Moreover, the combinatorial nature of $\mathrm{Dgm}(f_P)$ usually leads to non-smooth optimization with oscillations and lack of convergence.

**Objectives of the internship:** To improve on this issue, one appealing option would be to consider a parameterized vector field $v(\cdot, \theta) : \mathbb{R}^d \to \mathbb{R}^d$. Such a vector field induces a *flow* through the Ordinary Differential Equation (ODE)

$$\frac{\mathrm{d}x}{\mathrm{d}t}(t) = v(x(t), \theta), \tag{1}$$

so that we may define $\Phi(x, \theta) = \int_0^1 v(x(t), \theta)\mathrm{d}t$ where $x(t)$ is a solution of the ODE with $x(0) = x$ (that is, $x$ follows the flow described by the ODE until time $t = 1$). Such flows are known to typically induce diffeomorphisms over $\mathbb{R}^d$, hence providing us a smooth way to move all points in $P$ all together.

Therefore, instead of looking at the problem of optimizing $P \mapsto L(\mathrm{Dgm}(f_P))$ for some loss $L$, we may consider the following optimization problem:

$$\text{minimize} \quad \theta \mapsto L\left(\mathrm{Dgm}\left(f_{\Phi(P,\theta)}\right)\right). \tag{2}$$

The benefits of considering this problem is that the gradient with respect to $\theta$ will depend on all points in $P$ instead of only few ones. Using appropriate vector field models can also help smoothing the optimization process and defining and proving mathematical guarantees about the convergence and quality of the limit.

It is worth noting that different kind of parameterized vector fields may be considered; in this internship we may consider two standard approaches:

- The LDDMM approach, in the vein of [1],
- The recent *Neural ODE* model [3], where $v(\cdot, \theta)$ is actually encoded by a neural network.

Both will be studied during the internship, with inherent pro and cons being explored.

The project will include both theoretical (studying the smoothness of the map, the convergence (rate) of the optimization scheme) and numerical aspects (implementation and comparison with the state-of-the-art methods for topological optimization [4, 5, 6]). Depending on the quality of the results, the work may lead to a publication in a scientific venue, either in machine learning or computational topology.

**Expected abilities of the candidate:** The student must be familiar with standard statistical and machine learning notions (optimization, classification, estimation, overfitting, etc). A background in Topological Data Analysis is appreciated. A background in Deep Learning and/or neural ODE is *not* required (but will be appreciated as well). More importantly, the will to implement and experiment with such models is crucial.

# References

[1] Faisal Beg, Michael Miller, Alain Trouvé, and Laurent Younes. *Computing Large Deformation Metric Mappings via Geodesic Flows of Diffeomorphisms.* International Journal of Computer Vision, 61:139–157, 2005.

[2] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. *Stability of persistence diagrams.* Discrete & Computational Geometry, 37(1):103–120, 2007.

[3] Ricky Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. *Neural Ordinary Differential Equations.* In Advances in Neural Information Processing Systems 32 (NeurIPS 2018). Curran Associates, Inc., 2018.

[4] Mathieu Carrière, Frédéric Chazal, Marc Glisse, Yuichi Ike, Hariprasad Kannan, and Yuhei Umeda: *Optimizing persistent homology based functions.* In 38th International Conference on Machine Learning (ICML 2021), volume 139, pages 1294–1303. PMLR, 2021.

[5] Jacob Leygonie, Mathieu Carrière, Théo Lacombe, and Steve Oudot: *A gradient sampling algorithm for stratified maps with applications to topological data analysis.* In CoRR. arXiv:2109.00530, 2021.

[6] Jacob Leygonie, Steve Oudot, and Ulrike Tillmann: *A framework for differential calculus on persistence barcodes.* Foundations of Computational Mathematics, 2021.