Multi-Parameter Persistent Homology is Practical (Extended Abstract)

Michael Kerber
Institute of Geometry
Graz University of Technology
8010 Graz, Austria
kerber@tugraz.at

Abstract

Multi-parameter persistent homology is a branch of topological data analysis that is notorious for being more difficult than the standard (one-parameter) version, both in theory and for algorithmic problems. We report on three ongoing projects that demonstrates that multi-parameter method are applicable to large data sets. For instance, natural bi-filtrations generalizing Vietoris-Rips or alpha filtrations for hundred of thousands of points can be decomposed within seconds in their indecomposable parts.

1 Introduction

Persistent homology is a main branch of topological data analysis that analyzes data sets under the lens of topological changes with respect to a real scale parameter. This paradigm has been established around 20 years ago and since then witnesses increasing interest in applied fields, ranging from image analysis over neuroscience to material science and cosmology.

The standard theory of persistent homology is restricted to a single real scale parameter. However, in many applications it might be natural to consider several parameters – think of a color image containing with RGB-values or a metric space equipped with a function. Multi-parameter persistent homology is an extension to this type of data. The first result for that theory was a negative one [3], showing that the infamous persistent barcode has no analogue for two and more parameters. Despite this initial setback, researchers have identified several partial, yet informative invariants [3, 10, 13, 15].

These theoretical advances raise the question of practical efficiency. While some concepts are provably hard to compute [2], several papers present polynomial time (approximation) algorithms and report on implementations and a promising practical performance [1, 4, 8, 11, 14]. Based on this work, we identify the following three steps in the pipeline of multi-parameter persistence computation:

Complexification We transform the input data into a combinatorial chain complex parameterized over $\mathbb{R}^d$. Typically, this chain complex arises as the boundary matrix of a simplicial complex that is filtered with d parameters.

Compression The chain complex from the first step is typically large. Of interest is primarily its persistence module which captures the homology of the chain complex. We search for a smaller structure to represent this persistence module either precisely or approximately.

Comprehension Finally, we aim for an understanding of the underlying data sets from its homological properties. Algorithmic tools for this step are, for instance, to compare two persistence modules, or to decompose a module in its indecomposable parts.

2 Recent developments

We report on several ongoing projects which fit into the pipeline of multi-parameter persistence and significantly push the limits of input sizes that it can be applied for.

Multi-cover filtrations. A standard construction in TDA is to replace a point cloud with balls of a radius r, and to study the topological evolution when r varies. A natural extension is to restrict to the area of Euclidean space that is covered by at least k of the balls ($k = 1$ the yields the definition above). This yields a bi-filtration with parameters r and k capturing more information about the point cloud than the standard construction. Previous approaches either produced prohibitively large complexes [17] or lead to zigzag structures that are more diffult to process in subsequent steps [9].

We found a simple construction based on higher-order Delaunay triangulations which yields a bi-filtered simplicial complex and is of small size [5]. Hence, it can be interpreted as a multi-parameter analogue of the alpha filtration. Moreover, we prove an equivalence result of our construction with the rhomboid tiling from [9] which allows us to use their existing efficient implementation for computations. With that we can, for instance, compute the multi-filtration of 200,000 points in the plane for $k \leq 4$ in around one minute.

Minimal presentations. A *(graded) presentation matrix* is a way to express the homological information of a persistence module: The rows of the matrix represent the generators of homology groups, the columns encode the relations between them. Ideally, one aims for a *minimal presentation*, that is, one with as few generators and relations as possible. Lesnick and Wright [14] proposed an algorithm to compute a minimal presentation for 2-parameter persistence modules. Their algorithm is implemented in Rivet and outperforms the more general algorithms provided in computer algebra systems like Magma and Singular.

We present improvements over the Lesnick-Wright algorithm which lead to further dramatic performance gains, both in terms of runtime and memory consumption [12]. In simple terms, the Lesnick-Wright algorithm iterates over a large 2-dimensional grid and spends most of the time to look for those grid cells where computation is needed. We show how to avoid this grid iteration and instead locate the places that require computation directly.

Our approach improved the previous algorithm on every tested instance and computes minimal presentations for large inputs efficiently. For instance, for a function-Rips bifiltration consisting of one million simplices, the minimal presentation is computed in around 25 seconds. The resulting presentation matrix is only of dimension 11×7 underlining the importance of this compression step.

Decomposition of persistence modules. While there is no barcode of a multi-parameter persistence module into intervals [3], every such module admits a unique decomposition into indecomposable pieces by the the Krull-Remak-Schmidt theorem from commutative algebra. This decomposition reveals valuable information about the module, for instance whether it decomposes into interval or rectangle modules. Also, the decomposition allows for the computation of the multi-parameter bottleneck distance [7]. The problem is polynomial-time solvable for more general modules with the so-called Meataxe algorithm [16]. Recently, Dey and Xin proposed a more specialized algorithm based on matrix reduction [8] with an improved complexity.

We provide a first implementation of the Dey-Xin-algorithm to decompose persistence modules given as a presentation matrix [6]. We realized that its practical performance depends a lot on design choices for the low-level primitives (such as the representation of the involved matrices). By carefully such choices, our prototypical implementation decomposes matrices with ten-thousands of rows and columns within seconds. We currently investigate the impact of further improvement ideas.

3 Next steps

We are currently experiencing a leap in algorthmic capabilities for multi-parameter persistence. While the limits can probably be pushed further, we believe that the most important question is: how can multi-parameter persistence used in real-world applications? Armed with efficient algorithmic tools, we think that it is now the time to address this question.

References

[1] S. Biasotti, A. Cerri, P. Frosini, and D. Giorgi. A new algorithm for computing the 2-dimensional matching distance between size functions. *Pattern Recognition Letters*, 32(14):1735–1746, 2011.

[2] H. Bjerkevik, M. Botnan, and M. Kerber. Computing the interleaving distance is NP-hard. *Foundations of Computational Mathematics*, 20:1237–1271, 2020.

[3] G. Carlsson and A. Zomorodian. The theory of multidimensional persistence. *Discrete & Computational Geometry*, 42(1):71–93, 2009.

[4] R. Corbet, U. Fugacci, M. Kerber, C. Landi, and B. Wang. A kernel for multi-parameter persistent homology. *Accepted for Shape Modelling International (SMI)*, 2019.

[5] R. Corbet, M. Kerber, M. Lesnick, and G. Osang. Computing the multicover bifiltration. Submitted.

[6] T. Dey and M. Kerber. Fast decomposition of multi-parameter persistence modules. In preparation.

[7] T. Dey and C. Xin. Computing bottleneck distance for 2-D interval decomposable modules. In *34th International Symposium on Computational Geometry (SoCG 2018)*, pages 32:1–32:15, 2018.

[8] T. Dey and C. Xin. Generalized persistence algorithm for decomposing multi-parameter persistence modules. arXiv:1904.03766, 2019.

[9] H. Edelsbrunner and G. Osang. The Multi-cover Persistence of Euclidean Balls. In *34th International Symposium on Computational Geometry (SoCG 2018)*, pages 34:1–34:14, 2018.

[10] H. Harrington, N. Otter, H. Schenck, and U. Tillmann. Stratifying multiparameter persistent homology. *SIAM Journal on Applied Algebraic Geometry*, 3(3):439–471, 2019.

[11] M. Kerber and A. Nigmetov. Efficient approximation of the matching distance for 2-parameter persistence. In *36th International Symposium on Computational Geometry (SoCG 2020)*, 2020.

[12] M. Kerber and A. Rolle. Fast minimal presentations of bi-graded persistence modules. Accepted for SIAM Symposium on Algorithm Engineering and Experiments (ALENEX 2021).

[13] M. Lesnick and M. Wright. Interactive visualization of 2-D persistence modules. arXiv:1512.00180, 2015.

[14] M. Lesnick and M. Wright. Computing minimal presentations and bigraded betti numbers of 2-parameter persistent homology. arXiv:1902.05708, 2019.

[15] E. Miller. Data structures for real multiparameter persistence modules. *arXiv:1709.08155*, 2017.

[16] R. Parker. The computer calculation of modular characters (the meat-axe). *Computational group theory*, pages 267–274, 1984.

[17] D. Sheehy. A multicover nerve for geometric inference. In *Canadian Conference on Computational Geometry*, 2012.