SIAM J. MATH. DATA SCI.
Vol. 5, No. 2, pp. 346–372

© 2023 Society for Industrial and Applied Mathematics

Time-Inhomogeneous Diffusion Geometry and Topology*

Guillaume Huguet†, Alexander Tong‡, Bastian Rieck§, Jessie Huang¶, Manik Kuchroo‖, Matthew Hirn#, Guy Wolf††, and Smita Krishnaswamy‡‡

Abstract. Diffusion condensation is a dynamic process that yields a sequence of multiscale data representations that aim to encode meaningful abstractions. It has proven effective for manifold learning, denoising, clustering, and visualization of high-dimensional data. Diffusion condensation is constructed as a time-inhomogeneous process where each step first computes a diffusion operator and then applies it to the data. We theoretically analyze the convergence and evolution of this process from geometric, spectral, and topological perspectives. From a geometric perspective, we obtain convergence bounds based on the smallest transition probability and the radius of the data, whereas from a spectral perspective, our bounds are based on the eigenspectrum of the diffusion kernel. Our spectral results are of particular interest since most of the literature on data diffusion is focused on homogeneous processes. From a topological perspective, we show that diffusion condensation generalizes centroid-based hierarchical clustering. We use this perspective to obtain a bound based on the number of data points, independent of their location. To understand the evolution of the data geometry beyond convergence, we use topological data analysis. We show that the condensation process itself defines an intrinsic condensation homology. We use this intrinsic topology, as well as the ambient persistent homology, of the condensation process to study how the data changes over diffusion time. We demonstrate both types of topological information in well-understood toy examples. Our work

*Received by the editors March 28, 2022; accepted for publication (in revised form) January 6, 2023; published electronically May 22, 2023. The first, second, third, and fourth authors are equal contributors. The sixth, seventh, and eighth authors are equal senior contributors.

https://doi.org/10.1137/21M1462945

Funding: The sixth author was partially supported by NSF grant DMS-1845856. The seventh author was partially funded by IVADO Professor funds, CIFAR AI Chair, and NSERC Discovery grant 03267. The sixth, seventh, and eighth authors were partially supported by NIH grant NIGMS-R01GM135929. The content provided here is solely the responsibility of the authors and does not necessarily represent the official views of the funding agencies.

†Department of Mathematics and Statistics, Université de Montréal, Montréal, QC H3T 1J4, Canada, and Mila - Quebec AI Institute, Montréal, QC H2S 3H1, Canada (guillaume.huguet@mila.quebec).

‡Department of Computer Science and Operations Research, Université de Montréal, Montréal, QC H3T 1J4, Canada, and Mila - Quebec AI Institute, Montréal, QC H2S 3H1, Canada (alexander.tong@mila.quebec).

§Helmholtz Munich, Ingolstädter Landstr. 1, 85764 Neuherberg, Germany, and Technical University of Munich, Arcisstr. 21, 80333 Munich, Germany (bastian.rieck@helmholtz-muenchen.de).

¶Departments of Computer Science and Genetics, Yale University, New Haven, CT 06520 USA (jiexi.huang@yale.edu).

‖Department of Neuroscience, Yale University, New Haven, CT 06520 USA (manik.kuchroo@yale.edu).

#Departments of CMSE and Mathematics, Michigan State University, East Lansing, MI 48824 USA (matthew.hirn@gmail.com).

††Corresponding coauthor. Department of Mathematics and Statistics, Université de Montréal, Montréal, QC H3T 1J4, Canada, and Mila - Quebec AI Institute, Montréal, QC H2S 3H1, Canada (guy.wolf@umontreal.ca).

‡‡Corresponding coauthor. Departments of Computer Science and Genetics, Yale University, New Haven, CT 06520 USA, and Departments of CMSE and Mathematics, Michigan State University, East Lansing, MI 48824 USA (smita.krishnaswamy@yale.edu).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

gives theoretical insight into the convergence of diffusion condensation and shows that it provides a link between topological and geometric data analysis.

Key words. diffusion, time-inhomogeneous process, topological data analysis, persistent homology, hierarchical clustering

MSC codes. 57M50, 57R40, 62R40, 37B25, 68

DOI. 10.1137/21M1462945

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

1. Introduction.

Graph representations of high-dimensional data have proven useful in many applications, such as visualization, clustering, and denoising. Typically, a set of data points is described by a graph that uses a pairwise affinity measure, whose values are stored in an affinity matrix. With this matrix, one can define the random walk operator, or the graph Laplacian, and use numerous tools from graph theory to characterize the input data. Diffusion operators are closely related to random walks on a graph, as they describe how heat (or gas) propagates across the vertices. Using powers of this operator yields a time-homogeneous Markov process, which has been extensively studied. Most notably, Coifman and Lafon [9] proved that, under specific conditions, this operator converges to the heat kernel on an underlying continuous manifold. Manifold learning methods like diffusion maps [9] define an embedding via the eigendecomposition of the diffusion operator. Other methods, such as PHATE [27], embed a diffusion-based distance by multidimensional scaling. Various clustering algorithms rely on the eigendecomposition of this operator (or the resulting Laplacian) [24, 34]. However, this homogeneous process requires a bandwidth in order to fix and determine the scale of the captured data manifold. If we are interested in considering multiple scales of the data [2, 21], or if the data is sampled from a time-varying manifold [25], we need a time-inhomogeneous process.

In this paper, we focus on the time-inhomogeneous diffusion process for a given initial set of data points. This process is known as *diffusion condensation* [2] and yields a representation of the data by a sequence of datasets, each at a different granularity. This sequence is obtained by iteratively applying a diffusion operator. It has proven effective for tasks such as denoising, clustering, and manifold learning [2, 21, 25, 31, 32]. In this work, we study theoretical questions of diffusion condensation. Thus, we define conditions on the diffusion operators such that the process converges to a *single* point. The convergence to a point is a valuable characteristic, as it is a necessary condition for any process that sweeps a complete range of granularities of the data. We present this analysis from both a geometric and a spectral perspective, addressing different families of operators. We also study how the intrinsic shape of the condensed datasets evolves through condensation time using tools from topological data analysis. In particular, we define an intrinsic filtration based on the condensation process, resulting in the notions of persistent homology and condensation homology, for studying individual condensation steps or for summarizing the entire process, respectively. Making use of a topological perspective, we also prove the relation between diffusion condensation and various types of hierarchical clustering algorithms.

The paper is organized as follows. In section 2, we present an overview of diffusion condensation. In section 3, we develop a geometric analysis of the process; most importantly,

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

we prove its convergence to a point. In section 4, we study the convergence of the process from a spectral perspective. In section 5, we present a topological analysis of the process and relate diffusion condensation to existing hierarchical clustering algorithms.

2. Diffusion condensation.

In order to establish the setup and scope for our work, we first formalize the diffusion condensation framework and provide a unifying view of design choices and algorithms used to empirically evaluate its efficacy in previous and related work.

2.1. Notation and setup.

Let $\mathsf{X} = \{x(j) : j = 1, \ldots, N\} \subset \mathbb{R}^d$ be an input dataset of N data points in d dimensions. Given a symmetric nonnegative affinity kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, with $0 \leq k(x, y) = k(y, x) \leq 1$, $x, y \in \mathbb{R}^d$, we define an $N \times N$ kernel matrix $\mathbf{K}$ with entries $\mathbf{K}(i, j) := k(x(i), x(j))$, which can be regarded as a weighted adjacency matrix of a graph capturing the intrinsic geometry of the data. Furthermore, the kernel and resulting graph are often considered as providing a notion of locality in the data, which can be tuned by a kernel bandwidth parameter ϵ. We defer discussion of specific k dependent on ϵ to subsection 2.5 but mention that it can be regarded as a proxy for the size or (local) radius of the neighborhoods defined by the kernel. The diffusion framework for manifold learning [9, 27] uses this construction to define a Markov process over the intrinsic structure of the data by normalizing the kernel matrix with a diagonal degree matrix $\mathbf{D} := \mathrm{diag}(d(1), d(2), \ldots, d(N))$ where $d(i) := \sum_j \mathbf{K}(i, j)$, resulting in a row stochastic Markov matrix $\mathbf{P} := \mathbf{D}^{-1}\mathbf{K}$, known as the (discrete) *diffusion operator*. Traditionally, time-homogeneous diffusion processes leverage powers $\mathbf{P}^\tau$ of this diffusion operator, for diffusion times $\tau \in \mathbb{N}$, to capture underlying data-manifold structure in X and to organize the data along this structure [9, 27].

Here, on the other hand, we follow the diffusion condensation approach [2] and use a time-inhomogeneous process, where the diffusion operator (and underlying finite dataset) vary over time. We consider a sequence of datasets $\mathsf{X}_t = \{x_t(j) : j = 1, \ldots, N\}$, ordered along diffusion condensation time $t \in \mathbb{N}$, with corresponding diffusion operators $\mathbf{P}_t$, each constructed over the corresponding X_t. With a slight abuse of notation, we often refer to X_t as a set or as an $N \times d$ matrix, where $x_t(j)$ is the jth row or, equivalently, the jth element of the set. At time $t = 0$ we consider the input dataset, with its (traditional) diffusion operator, while for each $t > 0$ we take $\mathsf{X}_t := \mathbf{P}_{t-1}^\tau \mathsf{X}_{t-1}$, with the usual matrix multiplication. Then, instead of powers of a single diffusion operator, the t-step condensation process is defined via $\mathbf{P}^{(t-1)} := \mathbf{P}_{t-1}^\tau \cdots \mathbf{P}_0^\tau$, and thus we can also directly write $\mathsf{X}_t = \mathbf{P}^{(t-1)}\mathsf{X}_0$. Note that $\mathbf{P}^{(t-1)}$ is constructed from a collection of operators based on different datasets, and potentially different bandwidth parameters or kernels, which therefore makes the process time-inhomegeneous. For simplicity, we keep the diffusion time τ, but it could also depend on the condensation time t. Finally, we use the notation $\mathsf{X}^{(T)} := \mathsf{X}_0, \mathsf{X}_1, \ldots, \mathsf{X}_T$ for a sequence of datasets up to finite time T, and denote the diameter of the dataset at time t as $\mathrm{diam}(\mathsf{X}_t) := \max_{x,y \in \mathsf{X}_t} \|x - y\|_2$.

2.2. Related work using diffusion condensation for data analysis and open questions.

The diffusion condensation algorithm first proposed in Brugnone et al. [2] has been applied for data analysis in a number of areas. Moyle et al. [28] applied diffusion condensation to the study of neural connectomics between species and to identify biologically meaningful substructures. Kuchroo et al. [21] applied diffusion condensation to embedding and visualizing single-cell proteomic data to explore the effect of COVID-19 on the immune system. Kuchroo et al. [20] applied diffusion condensation to single-nucleus RNA sequencing data from human retinas with age-related macular degeneration (AMD) and found a potential

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

drug target by exploring the topological structure of the resulting diffusion condensation process. Van Dijk et al. [32] applied one step of diffusion condensation ($T = 1$) with high τ to single-cell RNA sequencing data to impute gene expression. They showed that high τ improves the quality of downstream tasks such as gene-gene relationships and visualization. These works demonstrate the empirical utility of diffusion condensation in a number of settings, specifically when multiscale clustering and visualization is needed and the data lies on a manifold.

The diffusion condensation process is a particular type of time-inhomogeneous diffusion process. General time-inhomogeneous diffusion processes over time-varying data were studied in [25], where it was proposed to use the singular value decomposition of the operator $\mathbf{P}^{(t)}$ to embed an arbitrary sequence of datasets $\mathsf{X}^{(T)}$ according to their space-time geometry. Additionally, in the case when those datasets $\mathsf{X}^{(T)}$ are sampled from a manifold $(\mathcal{M}, g(t))$ with time-varying metric tensor $g(t)$, it was shown in [25] that as $N, T \to \infty$, the operator $\mathbf{P}^{(t)}$ converges to the heat kernel of $(\mathcal{M}, g(t))$. We also note a resemblance to the mean shift algorithm [14, 7], which relies on a kernel-based estimation of $\nabla \log p(x)$, where $p(x)$ is the unknown density from which the points are sampled. The processed dataset is recursively updated via $x_{t+1}(i) = x_t(i) + \epsilon \nabla \log p(x)$, which effectively moves all points toward a mode of the distribution, which hence creates clusters.

Motivated by these empirical successes and inspired by the general theoretical results on time-inhomogeneous diffusion processes, we consider two open questions specific to the diffusion condensation process. Under what conditions does the diffusion condensation algorithm converge? How can the topology of the diffusion condensation process be understood?

2.3. Theoretical contributions. The main contribution of this paper is to address these open questions and establish the underpinnings of diffusion condensation. Our investigation is divided into three perspectives. First, we investigate the convergence properties of diffusion condensation under various parameter regimes from a geometric perspective in section 3, i.e., arrangement of data points in spatial coordinates. This geometric perspective gives an intuitive sense of convergence for a large family of kernels with minimum tail bounds. Next, in section 4 we investigate convergence from a spectral graph theory perspective and prove convergence in terms of the spectral properties of the kernel, viewing diffusion condensation as a nonstationary Markov process. A spectral perspective gives bounds in terms of the eigenvalues of the kernel, which can give better rates of convergence depending on considered data. Finally, in section 5 we investigate the topological characteristics of diffusion condensation. Here we describe both the structure of the dataset at each condensation step individually via its persistent homology and the topology of the condensation process itself, which we refer to as *condensation homology*. Additionally, we link the topology of the diffusion condensation process to hierarchical clustering and prove how it generalizes centroid linkage.

2.4. Algorithm. The diffusion condensation algorithm summarizes input data with a series of representations, organized by condensation time, with earlier representations providing low level, microscopic details, and later representations providing overall, macroscopic summarizations. Each time step of diffusion condensation can be broken up into five main steps as follows:

1. Construct a kernel matrix $\mathbf{K}_t$ summarizing similarities between points.
2. Construct a Markov normalized diffusion operator $\mathbf{P}_t$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Algorithm 2.1 Diffusion condensation.

Input: Dataset X_0, initial kernel parameter ϵ_0, diffusion time τ, and merge radius ζ
Output: Condensed datasets $\mathsf{X}^{(T)}$
for $t \in \{0, 1, \ldots, T-1\}$ **do**
  $\mathbf{K}_t \leftarrow \text{kernel}(\mathsf{X}_t, \epsilon_t)$
  $\mathbf{P}_t \leftarrow \mathbf{D}_t^{-1}\mathbf{K}_t$
  $\mathsf{X}_{t+1} \leftarrow \mathbf{P}_t^{\tau}\mathsf{X}_t$
  $\epsilon_{t+1} \leftarrow \text{update}(\epsilon_t, \mathsf{X}_{t+1})$
  for $x_t(i), x_t(j) \in \mathsf{X}_t$ **do**
    $\text{merge}(x_t(i), x_t(j))$ if $\|x_t(i) - x_t(j)\|_2 < \zeta$
  end for
end for
$\mathsf{X}^{(T)} \leftarrow \{\mathsf{X}_0, \mathsf{X}_1, \ldots, \mathsf{X}_T\}$

3. Diffuse the data coordinates for τ steps using $\mathbf{P}_t^{\tau}$.
4. Update the kernel bandwidth ϵ according to some `update` function.
5. (Optionally) merge points within distance ζ.

Algorithm 2.1 shows pseudocode for this process. At each time step, the positions of points are updated based on the predefined kernel through τ steps of diffusion. Intuitively, this can be thought of as moving each point to a kernel-weighted average of its neighbors, The condensation process will behave differently depending on the choice of kernel, the kernel bandwidth, the diffusion time, and the merging threshold.

Figure 1 depicts the differences between time-homogeneous condensation, time-inhomogeneous condensation, and a mixture of the two. Greater values of τ encourage the process to condense along the manifold, in contrast with other hierarchical clustering algorithms that are not able to do so. Comparing only inhomogeneous condensation $\mathbf{P}_{3i}$ (top row) with a mixture of homogeneous and inhomogeneous condensation $\mathbf{P}_i^3$ (middle row), we see that the mixture condenses the moon structures along the manifold rather than shattering them. Both of these are able to separate out the two clusters. In contrast, the fully time-homogeneous condensation process $\mathbf{P}^{3i}$ (bottom row) eventually mixes the two moons. For the rest of the paper, we let $\tau = 1$, but our results are valid for any $\tau \in \mathbb{N}$. It is only for the spectral part that we need to consider a slight nuance, which we discuss in Remark 4.8.

Remark 2.1. The *time-homogeneous* equivalent of the condensation process would be to recursively apply the same diffusion operator $\mathbf{P}_0$ on the initial dataset X_0. After t iterations, the new dataset would simply be $\mathbf{P}_0^t\mathsf{X}_0$. This constrasts with the *time-inhomegeneous* version where we create a new diffusion operator at each iteration. Concequently, after t iterations, the new dataset is $\mathbf{P}_{t-1}\ldots\mathbf{P}_1\mathbf{P}_0\mathsf{X}_0$. By using a time-inhomogeneous process, we gain more control over the convergence behavior. Indeed, since we allow for modifications of the diffusion probabilities, we can define a schedule for the parameters that could either promote or slow down the convergence of the process. Here, for simplicity, we assumed $\tau = 1$, but the same remark follows for any diffusion time $\tau \in \mathbb{N}$.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

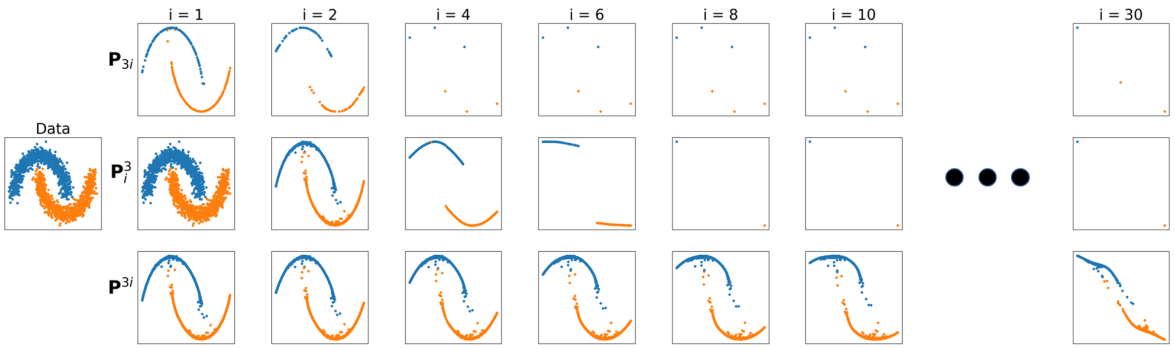


Figure 1. *Shown are the effects of powering the diffusion operator with a power τ before condensing on the moons dataset. Shown from top to bottom row are the time-inhomogeneous process with $\tau = 1$ and with $\tau = 3$, and the time-homogeneous process with $T = 1$ while varying τ. For step i this corresponds to comparing the applications of $\mathbf{P}_{3i}$, $\mathbf{P}_i^3$, and $\mathbf{P}^{3i}$ to the data. $\mathbf{P}_{3i}$ eventually merges but has a semistable state of six points after shattering the moons. $\mathbf{P}_i^3$ correctly identifies the two clusters of the data efficiently by first condensing along the moons individually into points. Time-homogeneous condensation $\mathbf{P}^{3t}$ mixes the two moons.*

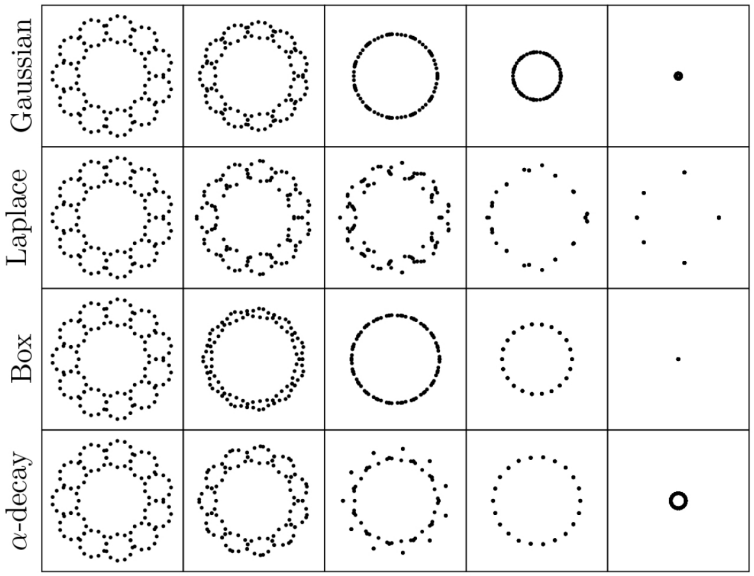


Figure 2. *An example of different kernels (rows) and how they affect convergence behavior for the "petals" dataset. Convergence speed is highest in the box kernel (15 iterations), followed by the Gaussian kernel (25 iterations), and the α-decay kernel (40 iterations, $\alpha = 10.0$), whereas the Laplace kernel requires 491 iterations to converge.*

2.5. Kernels for diffusion condensation. Here, we review specific kernel constructions used to study condensation properties in later sections. In Figure 2, we present a few iterations of the condensation process, depending on the choice of kernel used to construct the diffusion matrix.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Definition 2.2 (box kernel). *The box kernel of bandwidth ϵ is*

$$k_\epsilon(x,y) = \begin{cases} 1 & \text{if } \|x-y\|_2 \le \epsilon, \\ 0 & \text{else.} \end{cases} \tag{2.1}$$

The box kernel is arguably the simplest and most interpretable kernel, leading to interesting data summarizations, including an instance of agglomerate clustering depending on the bandwidth as a function of time. However, first we note some simple cases of bandwidth settings. Consider a case where $k_t(x,y) = 1$ for all $x, y \in \mathsf{X}_0$. This can be thought of as a box kernel with bandwidth greater than $\operatorname{diam}(\mathsf{X}_0)$. Using this kernel, after a single step of diffusion condensation, all points converge to the mean data point, $\frac{1}{n}\sum_i x_t(i)$. This mean data point is a useful, if trivial, summarization of the data. Next, consider the opposite extreme, a box kernel with infinitely narrow bandwidth, $k_t(x,y) = \{1 \text{ if } x = y \text{ else } 0\}$. In this case, we have $\mathsf{X}_t = \mathsf{X}_0$ for all $t > 0$, resulting in another trivial result, i.e., *no* data summarization over diffusion condensation time. Of more interest are bandwidths between these two extremes, providing hierarchical sets of summarizations. Next, we consider smoother kernels.

Definition 2.3. *The* α-decay kernel [27] *of bandwidth ϵ is* $k_{\epsilon,\alpha}(x,y) = \exp(-\|x-y\|_2^\alpha/\epsilon^\alpha)$.

The α-decay kernel was used in [21] along with anisotropic density normalization (see Definition 2.6), which was shown to empirically speed up convergence of diffusion condensation.

Definition 2.4. *The* Gaussian kernel *of bandwidth ϵ is* $k_\epsilon(x,y) = \exp(-\|x-y\|_2^2/\epsilon)$.

The Gaussian kernel was used in [2], employing density normalization and a merging threshold of 10^{-4}, with a bandwidth of ϵ_t doubling whenever the change in position of points between $t-1$ and t dropped below a separate threshold. This kernel and setting of ϵ_t ensure that the datasets converge to a single point in a reasonable amount of time in practice.

Another kernel that exhibits interesting behavior is the Laplace kernel; it is noteworthy since it is positive definite for all conditionally negative definite metrics [13].

Definition 2.5. *The* Laplace kernel *of bandwidth ϵ is* $k_\epsilon(x,y) = \exp(-\|x-y\|_2/\epsilon)$.

Note that this is the same as the α-decay kernel, with $\alpha = 1$. In fact, the Gaussian and Laplace kernels can be generalized to the α-decay kernel, which interpolates between the Gaussian kernel when $\alpha = 2$ and the box kernel as $\alpha \to \infty$.

Definition 2.6 (anisotropic density normalized kernel [9]). *For a rotation invariant kernel $k_\epsilon(x,y)$, let $q(x) = \int_{\mathsf{X}} k_\epsilon(x,y)q(y)\mathrm{d}y$; then a density normalized kernel with normalization factor β is given by* $k_{\epsilon,\beta}(x,y) = \frac{k_\epsilon(x,y)}{q^\beta(x)q^\beta(y)}$.

3. Geometric properties of the condensation process.

We first examine the time-varying nature of the data geometry along the diffusion condensation process. Since this process results in a sequence of finite datasets X_t, organized along condensation time, pertinent questions are how their underlying geometric structure changes as local variability is eliminated by the diffusion process and whether it eventually converges to a stable one as $t \to \infty$. In this section, we study these questions by considering two geometric characteristics (namely, convex hull and diameter) of the data, establishing their monotonic convergence, and determining their relation to the tail behavior of the kernel utilized in the construction of the diffusion process.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Our main result here is that, with appropriate kernel choice, the condensation process converges to a point, with the time to convergence being dependent on the shape of the kernel. That is, for all $\zeta > 0$, there exists an $M \in \mathbb{N}$ such that for all $t \geq M$, we have $\|x - y\| < \zeta$ for all $x, y \in \mathsf{X}_t$. Intuitively, we can make all the points arbitrarily close by iterating the process. It is important to note that this result requires some assumptions on the kernel in order to avoid pathological cases where the process may converge, but not to a single point. For instance, using the box kernel on a dumbbell dataset, each sphere would converge to a point, but for a certain threshold these points would not be connected. Hence, the process would reach a stable state, i.e., there exists $M \in \mathbb{N}$ such that $\mathbf{P}_M \mathsf{X}_M = \mathsf{X}_M$, but it would not converge to a point (see Figure 3 for an illustration). One of our goals is to define the conditions on the kernels for the process to converge to a single point (see Figure 4 for an example).

3.1. Diameter and convex hull convergence. Intuitively, one can consider each diffusion condensation iteration as eliminating local variability in data [2, 21]. While empirical results presented in previous work indicate the condensation process can accentuate separation between weakly connected data regions, they also indicate that the process has a global contraction property due to the elimination of variability in the data. Thus, it appears that diffusion condensation coarse-grains data by sweeping through granularities, from each point being a separate entity to all data points being in a single cluster. To establish this contractive property and formulate a notion of data geometry (monotone) convergence associated with it, we characterize the geometry of each X_t via its diameter and convex hull, whose convergence under the condensation process is shown in the following theorem.

Remark 3.1. The diffusion condensation process is not a contractive mapping in the strict sense. During individual iterations, distances are not generally *all* decreasing, i.e., there exist points $x_t(i)$ and $x_t(j)$ such that $\|x_t(i) - x_t(j)\|_2 < \|x_{t+1}(i) - x_{t+1}(j)\|_2$.

Theorem 3.2. *Let $(\mathsf{X}_t)_{t\in\mathbb{N}}$ and $(\mathbf{P}_t)_{t\in\mathbb{N}}$ be, respectively, the sequence of datasets and diffusion operators generated by diffusion condensation. If the kernel used to construct each $\mathbf{P}_t$ is strictly (pointwise) positive, then*

1. *their convex hulls form a nested sequence with*

$$\lim_{t\to\infty} \operatorname{conv}(\mathsf{X}_t) = \bigcap_{t=1}^{\infty} \operatorname{conv}(\mathsf{X}_t) \neq \emptyset \quad \textit{and convex.}$$

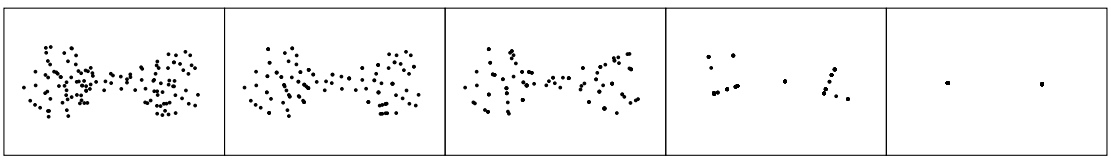


Figure 3. *Convergence behavior of a "dumbbell" dataset for different iterations of a box kernel with fixed bandwidth. The process converges to two points that are not connected.*

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

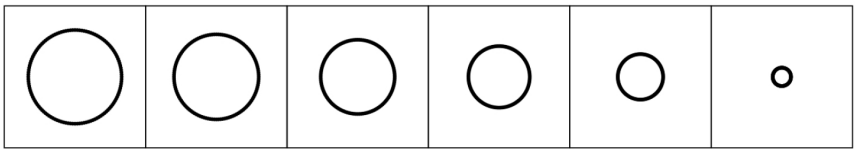


Figure 4. *The convergence behavior of a hyperuniform circle—a circle with equally spaced points around its circumference—using a Gaussian kernel.*

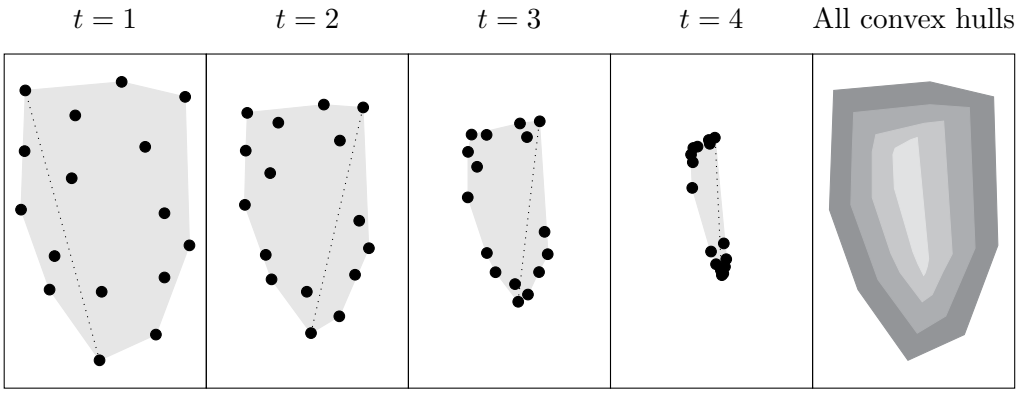


Figure 5. *Illustration of Theorem 3.2. We depict four time steps of the condensation process of a simple dataset. The convex hull of the points is shown in gray, with the diameter of X_t shown as a dotted line. As t progresses, convex hulls shrink, with $\operatorname{conv}(\mathsf{X}_{t+1}) \subsetneq \operatorname{conv}(\mathsf{X}_t)$. The rightmost figure shows all convex hulls of all time steps with lighter shades indicating later condensation time steps.*

2. *the diameters form a convergent monotonically decreasing sequence with*

$$\lim_{t\to\infty} \operatorname{diam}(\mathsf{X}_t) = \inf_{t\geq 1} \operatorname{diam}(\mathsf{X}_t) \geq 0.$$

   Further, $\operatorname{diam}(\mathsf{X}_{s+1}) = \operatorname{diam}(\mathsf{X}_s)$ *if and only if* $\operatorname{diam}(\mathsf{X}_s) = 0$*, i.e., for* $s \in \mathbb{N}$ *such that* $\operatorname{diam}(\mathsf{X}_s) > 0$*, we have* $\operatorname{diam}(\mathsf{X}_{s+1}) < \operatorname{diam}(\mathsf{X}_s)$*.*
3. *if there exists* $k \in \mathbb{N}$ *such that* $\mathbf{P}_k \mathsf{X}_k = \mathsf{X}_k$*, then* $\operatorname{conv}(\mathsf{X}_k) = \{x_k\}$*, i.e., the process converged to a single point.*

Prior to proving Theorem 3.2, we first require the following technical lemma about polytopes, which can also be found in standard literature [22]. Its proof is provided in the supplementary material (supplementary_material_dc.pdf [local/web 321KB]) for completeness.

Lemma 3.3. *Let* $\mathsf{X} \subset \mathbb{R}^d$ *be a set of points and* $C := \operatorname{conv}(\mathsf{X})$ *be their convex hull. Then every* extremal point $v_j \in C$ *satisfies* $v_j \in \mathsf{X}$*. Thus, the extremal points of* C *are a subset of* X*.*

With Lemma 3.3, we are now ready to prove Theorem 3.2 as follows (see Figure 5 for an illustration of the arguments in the proof).

Proof of Theorem 3.2. Denote the interior of the convex hull of X as $\operatorname{int}(\operatorname{conv}(\mathsf{X}))$. We start by proving the following:

(a) If $\operatorname{diam}(\mathsf{X}_t) > 0$, then $\operatorname{conv}(\mathsf{X}_{t+1}) \subseteq \operatorname{int}(\operatorname{conv}(\mathsf{X}_t))$;
(b) $\operatorname{diam}(\mathsf{X}_t) = 0$ if and only if $\mathbf{P}_t \mathsf{X}_t = \mathsf{X}_t$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

To prove (a), we note that since the entries of $\mathbf{P}_t$ are positive and its rows sum to 1, each element of X_{t+1} is a *convex combination* of the original data points. That is, $x_{t+1}(i) = \mathbf{P}_t(i,\cdot)\mathsf{X}_t$, with $\mathbf{P}_t(i,j) > 0$ for all $j \in \{1,\dots,N\}$, and $\sum_j \mathbf{P}_t(i,j) = 1$. As a consequence, X_{t+1} will be formed by convex combinations, so all points in X_{t+1} lie in the *interior* of $\mathrm{conv}(\mathsf{X}_t)$, which is not empty since $\mathrm{diam}(\mathsf{X}_t) > 0$. From Lemma 3.3, we know that the extremal points of $\mathrm{conv}(\mathsf{X}_{t+1})$ also lie in the interior of $\mathrm{conv}(\mathsf{X}_t)$. Hence $\mathrm{conv}(\mathsf{X}_{t+1}) \subseteq \mathrm{int}(\mathrm{conv}(\mathsf{X}_t))$, which proves (a). To prove (b), we assume $\mathrm{diam}(\mathsf{X}_t) = 0$. By construction of $\mathbf{P}_t$, we get $\mathbf{P}_t\mathsf{X}_t = \mathsf{X}_t$. Now if we assume $\mathbf{P}_t\mathsf{X}_t = \mathsf{X}_t$, we have $\mathrm{conv}(\mathsf{X}_{t+1}) = \mathrm{conv}(\mathbf{P}_t\mathsf{X}_t) = \mathrm{conv}(\mathsf{X}_t)$. If $\mathrm{diam}(\mathsf{X}_t) > 0$, this would contradict (a), and hence $\mathrm{diam}(\mathsf{X}_t) = 0$. Steps (a) and (b) show that the convex hulls are a nested sequence, so $\mathrm{conv}(\mathsf{X}_t) \to \cap_{t=1}^{\infty}\mathrm{conv}(\mathsf{X}_t)$. Since the intersection of convex sets is convex, the limiting set is also convex. Finally, we use Helly's theorem [22], which states that if an infinite collection of compact convex subsets in $\mathbb{R}^d$ has a nonempty intersection for every $d+1$ subsets, then the collection of *all* subsets has a nonempty intersection. Here, because of the nesting property, every subcollection has a nonempty intersection. Moreover, the convex hulls of finite sets are compact, and hence we conclude that $\cap_{t=1}^{\infty}\mathrm{conv}(\mathsf{X}_t)$ is not empty.

Finally, (a) and (b) imply $\mathrm{diam}(\mathsf{X}_{t+1}) < \mathrm{diam}(\mathsf{X}_t)$ if $\mathrm{diam}(\mathsf{X}_t) > 0$, and $\mathrm{diam}(\mathsf{X}_{t+1}) = \mathrm{diam}(\mathsf{X}_t)$ if $\mathrm{diam}(\mathsf{X}_t) = 0$. Thus, the diameters form a monotonically decreasing sequence, which converges since the sequence is nonnegative. ∎

3.2. Convergence rates. Theorem 3.2 applies for all strictly positive kernels. While it is only established in terms of the diameter and convex hull of the data, this result extends to showing pointwise convergence of the diffusion condensation process if we make further assumptions on the rate at which the diameter sequence decreases, or if we establish bounds on this rate based on the specific kernel used in the diffusion construction. We next proceed with such an in-depth analysis, focusing on strictly positive kernels, while noting that later, in subsection 5.4, we will also show a convergence result for a kernel with finite support (i.e., where the discussion here is not valid). We begin with the following result relating the rate of convergence to the minimum value of the kernel over the data.

Lemma 3.4. *If there exists a nonnegative constant δ, such that $0 < \delta \le \mathbf{K}_t(i,j) \le 1$ for all $t \in \mathbb{N}$, then the diameter sequence $(\mathrm{diam}(\mathsf{X}_t))_{t\in\mathbb{N}}$ decreases at a speed of at least $1-\delta$, i.e., $\mathrm{diam}(\mathsf{X}_{t+1}) \le (1-\delta)\mathrm{diam}(\mathsf{X}_t)$.*

Proof. Here we present the key ideas of the proof, and we refer the reader to the supplementary material (supplementary_material_dc.pdf [local/web 321KB]) for the detailed version. The assumption on $\mathbf{K}_t$ gives the elementwise lower bound $\mathbf{P}_t \ge \delta/N$, and we show that $d_{TV}(\mathbf{P}_t(i,\cdot),\mathbf{P}_t(j,\cdot)) \le 1-\delta$, where d_{TV} is the total variation distance. Next, using a coupling ξ with marginals $\mathbf{P}_t(i,\cdot)$ and $\mathbf{P}_t(j,\cdot)$, we can write

$$\begin{aligned}\|x_{t+1}(i) - x_{t+1}(j)\|_2 &= \|(\mathbf{P}_t(i,\cdot) - \mathbf{P}_t(j,\cdot))\mathsf{X}_t\|_2 = \|\sum_{i,j}\xi(i,j)(x_t(i) - x_t(j))\|_2 \\ &\le \sum_{i,j}\xi(i,j)\|x_t(i) - x_t(j)\|_2 \le \sum_{i\neq j}\xi(i,j)\mathrm{diam}(\mathsf{X}_t).\end{aligned}$$

We conclude with the coupling lemma, which guarantees the existence of a coupling such that $\sum_{i\neq j}\xi(i,j) = d_{TV}(\mathbf{P}_t(i,\cdot),\mathbf{P}_t(j,\cdot))$, and thus $\mathrm{diam}(\mathsf{X}_{t+1}) \le (1-\delta)\mathrm{diam}(\mathsf{X}_t)$. ∎

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Given this result for general kernels whose tails can be lower bounded by some constant, we can further state a union bound result on kernels that maintain this lower bound over the entire diffusion condensation process. We recall at this point the ϵ update step that is typically used to expedite the condensation process when it reaches a slow contraction metastable state. Previous work implemented this step with heuristics for updating the ϵ metaparameter. Here, we provide further insight into the impact of this update step to both justify it and suggest an update schedule that provides certain convergence guarantees via the following theorem.

Theorem 3.5. *For some nonnegative constant δ, if there exists an ϵ_t schedule such that $0 < \delta \leq \mathbf{K}_t(i,j) \leq 1$ for all $t \in \mathbb{N}$, then for any merge threshold $\zeta > 0$, diffusion condensation converges to a single point in $t^* = \left\lceil \frac{\log(\zeta) - \log(\mathrm{diam}(\mathsf{X}_0))}{\log(1-\delta)} \right\rceil$ steps.*

Proof. Repeated application of Lemma 3.4 yields $\mathrm{diam}(\mathsf{X}_{t+1}) \leq (1-\delta)^t \mathrm{diam}(\mathsf{X}_0)$. Solving for the t^* such that $\mathrm{diam}(\mathsf{X}_{t^*}) < \zeta$, we have that $\mathrm{diam}(\mathsf{X}_{t^*}) \leq (1-\delta)^{t^*} \mathrm{diam}(\mathsf{X}_0) < \zeta$. Since t^* is an integer, the ceiling suffices. ∎

Remark 3.6. Theorem 3.5 shows one advantage of using a time-inhomogeneous process, since for a given δ we can find a schedule for ϵ_t such that $0 < \delta \leq \mathbf{K}_t(i,j) \leq 1$ and thus control the rate of convergence. This is in contrast with the time-homogeneous process, where condensation would be defined using the same kernel, and hence the benefit of an adaptive ϵ would be lost.

For specific forms of kernels, this result can be translated to suggest concrete ways of setting the kernel parameters at each time step such that this bound holds and diffusion condensation achieves well-behaved linear convergence.

Proposition 3.7. *For the following kernels, the specific bandwidth update suffices for the result in Theorem 3.5 to hold:*

1. *For the α-decay kernel, $\exp(-\|x-y\|_2^\alpha / \epsilon^\alpha)$, $\epsilon_t \geq -\mathrm{diam}(\mathsf{X}_t)^\alpha / \log(\delta)$ suffices. For $\alpha = 2$, this defines the scheduling for the Gaussian kernel. For $\alpha = 1$, this defines a scheduling for the Laplace kernel.*
2. *For the density normalized kernel $k_{\epsilon,\beta}(x,y)$ combined with the α-decay, we define $\epsilon_t^\alpha \geq -\mathrm{diam}(\mathsf{X}_t)^\alpha / \log(N^{2\beta}\delta)$, and $\epsilon_t \geq -\mathrm{diam}(\mathsf{X}_t)^2 / \log(N^{2\beta}\delta)$ for the Gaussian kernel. Then, Theorem 3.5 holds for all $\delta \in (0, 1/N^{2\beta})$. The same holds if we replace N with $q_{max,t} := \max q(i)$.*

Proof. To show part 1 we need to define a scheduling of ϵ_t such that $\min_{x,y \in \mathsf{X}_t} k_\epsilon(x,y) \geq \delta > 0$. For the α-decay kernel, we have $\min_{x,y \in \mathsf{X}_t} k_\epsilon(x,y) = \exp(-(\mathrm{diam}(\mathsf{X}_t)/\epsilon_t)^\alpha) \geq \delta \implies \epsilon_t^\alpha \geq -\mathrm{diam}(\mathsf{X}_t)^\alpha / \log(\delta)$. To show part 2, we need ϵ_t such that $\min_{x,y} k_{\epsilon,\beta}(x,y) \geq \delta$. We remark that $q(i) \leq q_{max,t} \leq N$, and hence $\min_{x,y} k_{\epsilon,\beta}(x,y) \geq \min_{x,y} k_\epsilon(x,y)/N^{2\beta}$. Therefore, we find ϵ_t such that $\min_{x,y} k_\epsilon(x,y) \geq N^{2\beta}\delta$ and conclude in the same way as for part 1. ∎

Remark 3.8. We note that to ensure a constant rate of convergence in the diameter, the kernel bandwidth ϵ_t in Proposition 3.7 shrinks over time proportional to the square of the diameter. This is in contrast to previous work [2] where a doubling schedule was used.

Remark 3.9. The previous results also hold for $\mathbf{P}_t^\tau$ with $\tau > 1$, as long as there exists $\delta > 0$ such that all entries $\mathbf{P}_t^\tau(i,j) > \delta/N$. Moreover, the rate of convergence with $\mathbf{P}_t^\tau$ cannot be

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

slower than with $\mathbf{P}_t$, because we can always write $\mathsf{X}_{t+1} = \mathbf{P}_t^\tau \mathsf{X}_t = \mathbf{P}_t^{\tau-1}\mathbf{P}_t\mathsf{X}_t$. Finally, for some $\mathbf{P}_t$ that includes zero probabilities, it is possible for $\mathbf{P}_t^\tau$ to be strictly pointwise positive, and hence Theorem 3.2 could be used for the process defined with $\mathbf{P}_t^\tau$ instead of $\mathbf{P}_t$.

4. Spectral properties of the condensation process. We complement the geometric perspective of condensation from the previous section with a spectral one based on the idea of using an orthonormal basis to express any function $f\colon \mathsf{X} \to \mathbb{R}$ (abbreviated $f \in \mathbb{R}^N$) as a weighted sum of the eigenvectors of $\mathbf{P}_t$ for all t. This sum is then divided into two terms: a constant term and a nonconstant one. The former corresponds to the lowest frequency of a function; it is constant for all $x \in \mathsf{X}_t$ and for all t. The nonconstant term represents the rest of the frequencies; it can vary depending on the eigenvectors of $\mathbf{P}_t$. We extend this reasoning to the time-inhomogeneous diffusion $\mathbf{P}^{(t)}f$. Our main result is Theorem 4.4, which provides an upper bound on the norm of the nonconstant term. For a specific choice of kernel and by using the coordinate function, this bound will converge to zero, and hence in Corollary 4.6 we show how condensation converges to a single point. Before presenting the main theorem, in subsection 4.1 we introduce a simpler example to give insight into the structure and challenges of the proof.

4.1. A simple condensation process. We consider a symmetric transition matrix $\mathbf{A}_t$ based on X_t, and the coordinate functions $f_i(x)$ which return the ith coordinate of x. In that case, $\mathbf{A}_t$ is known as a *bistochastic* matrix, and its stationary distribution is the uniform distribution. Since $\mathbf{A}_t$ is symmetric, its eigenvectors $\{\phi_{t,i}\}_{i=1}^N$ form an orthonormal basis of $\mathbb{R}^N$. Moreover, its ordered eigenvalues $\{\lambda_{t,i}\}_{i=1}^N$ are less than or equal to one, with $\lambda_{t,1} = 1$. Because $\mathbf{A}_t$ is row stochastic, and $\lambda_{t,1} = 1$, we can define $\phi_{t,1} = N^{-1/2}\mathbb{1}$ where $\mathbb{1}$ is a vector of ones of size N. Given these properties, we can write any function $f \in \mathbb{R}^N$ as

$$f = \sum_{k=1}^{N} \langle f, \phi_{t,k}\rangle \phi_{t,k}.$$

By splitting this sum into two terms, we define the constant terms $L_t(f) := \langle f, \phi_{t,1}\rangle\phi_{t,1} = (1/N)\langle f, \mathbb{1}\rangle\mathbb{1}$ and $H_t(f) := \sum_{k\geq 2}\langle f, \phi_{t,k}\rangle\phi_{t,k}$, and hence $f = L_t(f) + H_t(f)$. After one condensation step, we get

$$\mathbf{A}_0 f = \langle f, \phi_{0,1}\rangle\phi_{0,1} + \sum_{k=2}^{N} \lambda_{0,k}\langle f, \phi_{0,k}\rangle\phi_{0,k},$$

since $\lambda_{0,1} = 1$. Moreover, we note $L_0(f) = L_0(\mathbf{A}_0 f)$, which is therefore invariant through the iterations of condensation. We note the resemblance to the graph Fourier transform, which uses the eigendecomposition of the Laplacian and treats the eigenvalues as frequencies and the eigenvectors as harmonics. Here, L_t can be thought of as the lowest frequency term of the function. Whereas H_t varies depending on the eigenvectors, it can be seen as the higher frequencies of the function. Since $L_t(f)$ is constant during condensation, showing the convergence of the process is equivalent to showing that $\|H_t(\mathbf{A}^{(t)}f)\|_2$ tends to zero as t tends to infinity. Indeed, if this is true, by using the coordinate function f_i we have

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

$$\lim_{t\to\infty} L_0(\mathbf{A}^{(t)} f_i) = \langle f_i, N^{-1/2}\mathbb{1}\rangle N^{-1/2}\mathbb{1} = \left[(1/N)\sum_{j=1}^{N} f_i(x_0(j))\right]\mathbb{1},$$

and thus the process converges to the mean of the N data points. What is left to show is that the norm of the term $H_t(\mathbf{A}^{(t)} f)$ indeed converges to zero as t goes to infinity. After the first condensation step, we have the bound

$$\|H_0(\mathbf{A}_0 f)\|_2^2 = \sum_{k=2}^{N} \lambda_{0,k}^2 |\langle f, \phi_{0,k}\rangle|^2 \le \lambda_{0,2}^2 \sum_{k=2}^{N} |\langle f, \phi_{0,k}\rangle|^2 = \lambda_{0,2}^2 \|H_0(f)\|_2^2 \le \lambda_{0,2}^2 \|f\|_2^2,$$

and it can be deduced that $\|H_t(\mathbf{A}^{(t)} f)\|_2^2 \le \prod_{i=0}^{t} \lambda_{i,2}^2 \|f\|_2^2$. Hence, by showing that $\prod_{i=0}^{t} \lambda_{i,2}^2$ tends to zero as t tends to infinity, we could conclude that the process converges to a point.

Our situation is more complex since many kernels are not symmetric, so their eigenvectors do not form an orthonormal basis of $\mathbb{R}^N$. Here, we also benefited from the fact that the kernels were bistochastic, and hence they all had the same (uniform) stationary distributions. Generally, each kernel $\mathbf{P}_t$ has a different stationary distribution. This will be reflected in the upper bound, as we have to consider the distance between two consecutive stationary distributions.

4.2. A general condensation process. In general, we consider the broader class of diffusion operators defined in subsection 2.1 by $\mathbf{P}_t = \mathbf{D}_t^{-1}\mathbf{K}_t$. We recall that $\mathbf{K}_t$ is symmetric with $0 \le \mathbf{K}_t(i,j) \le 1$, and the diagonal degree matrix is $\mathbf{D}_t := \mathrm{diag}(d_t)$ where $d_t(i) := \sum_j \mathbf{K}_t(i,j)$. Moreover, the stationary distribution associated to $\mathbf{P}_t$ is $\pi_t(i) = \|d_t\|_1^{-1} d_t(i)$, and $\mathbf{P}_t$ is d_t-reversible, i.e., $d_t(i)\mathbf{P}(i,j) = d_t(j)\mathbf{P}(j,i)$. Thus, its associated operator

$$\mathbf{P}_t f(x(i)) := \sum_{j=1}^{N} \mathbf{P}_t(i,j) f(x(j)) \tag{4.1}$$

is self-adjoint with respect to the dot product $\langle f, g\rangle_{d_t} = \sum_x f(x(i))g(x(i))d_t(i)$. Denote $\{\psi_{t,i}\}_{i=1}^N$ as the eigenvectors of $\mathbf{P}_t$ and $\{\lambda_{t,i}\}_{i=1}^N$ as its eigenvalues arranged in decreasing order. Because $\mathbf{P}_t$ is self-adjoint with respect to $\langle\cdot,\cdot\rangle_{d_t}$, its normalized eigenvectors are such that $\langle \psi_{t,i}, \psi_{t,j}\rangle_{d_t} = \delta_{ij}$, where $\delta_{ij} = 1$ if $i = j$ and 0 otherwise. Therefore, we can write any function $f \in \mathbb{R}^N$ as $f = \sum_{k=1}^{N} \langle f, \psi_{t,k}\rangle_{d_t} \psi_{t,k}$. Following the same steps as in subsection 4.1, we want to find a constant term of the function. Since $\mathbf{P}_t$ is row stochastic, and because $\lambda_{t,1} = 1$, we get $\psi_{t,1} = c\mathbb{1}$ where c is a constant. We can solve for c using $\langle \psi_{t,1}, \psi_{t,1}\rangle_{d_t} = \langle c\mathbb{1}, c\mathbb{1}\rangle_{d_t} = 1$, which yields $c^2 = [\sum d_t(i)]^{-1} = ||d_t||_1^{-1}$. Hence, we define the constant term as

$$L_t(f) := \langle f, \psi_{t,1}\rangle_{d_t} \psi_{t,1} = ||d_t||_1^{-1} \langle f, \mathbb{1}\rangle_{d_t} \mathbb{1} = \langle f, \mathbb{1}\rangle_{\pi_t} \mathbb{1},$$

and the nonconstant term as the rest of the sum, which varies depending on the eigenvectors,

$$H_t(f) := \sum_{k=2}^{N} \langle f, \psi_{t,k}\rangle_{d_t} \psi_{t,k}.$$

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

We can write $\mathbf{P}_t f = \langle f, \mathbb{1}\rangle_{\pi_t}\mathbb{1} + \sum_k \lambda_{t,k}\langle f, \psi_{t,k}\rangle_{d_t}\psi_{t,k}$ by using the facts that $\mathbf{P}_t$ is self-adjoint and $\lambda_{t,1} = 1$. Most importantly, we note that the constant term of the function is not affected by condensation, i.e., $L_t(\mathbf{P}_t f) = L_t(f)$. Consequently, to show that the condensation process converges to a single point, it is sufficient to show

$$\lim_{t\to\infty} \|H_t(\mathbf{P}^{(t)} f_i)\|_2 = 0. \tag{4.2}$$

Indeed, if (4.2) holds, then for a constant C, we get $\lim_{t\to\infty} \mathbf{P}^{(t)} f_i(x) = C$ for all coordinates $i \in \{1, \dots, d\}$, and every $x \in \mathsf{X}_0$. To achieve our goal, in Theorem 4.4 we find an upper bound on $\|H_t(\mathbf{P}^{(t)} f)\|_2$, which we use in Corollary 4.6 to show convergence of the condensation process. Before presenting these results, we introduce several lemmas, whose proofs are provided in the supplementary material (supplementary_material_dc.pdf [local/web 321KB]). Lemma 4.1 is the same as the upper bound we found in subsection 4.1 and will be beneficial when used recursively. Lemma 4.2 is necessary since each $\mathbf{P}_t$ possibly has a different degree d_t, and it enables a change of measure to $\|\cdot\|_{d_s}$ from $\|\cdot\|_{d_t}$. Lastly, Lemma 4.3 is beneficial when combined with the observation that $H_t(f) = L_s(f) - L_t(f) + H_s(f)$.

Lemma 4.1. *For the operator $\mathbf{P}_t$ and its second largest eigenvalue $\lambda_{t,2}$, we have the following bound on the norm $\|H_t(\mathbf{P}_t f)\|_{d_t} \le \lambda_{t,2}\|H_t(f)\|_{d_t}$ for all functions $f \in \mathbb{R}^N$.*

Lemma 4.2. *For all functions $f \in \mathbb{R}^N$, and two operators $\mathbf{P}_t$ and $\mathbf{P}_s$, the following inequalities hold: $\|f\|_{d_t}^2 \le \|d_t/d_s\|_\infty \|f\|_{d_s}^2 \le (\|d_t - d_s\|_2 + 1)\|f\|_{d_s}^2$.*

Lemma 4.3. *For all $f \in \mathbb{R}^N$, $\|L_t(\mathbf{P}_t f) - L_s(\mathbf{P}_t f)\|_{d_t} \le \lambda_{t,2} N^{1/2}\|d_s - d_t\|_2 \|H_t(f)\|_{d_t}$.*

We are now ready to state and prove the main theorem of this section, providing an upper bound on the norm of the nonconstant part of a function. The upper bound mainly depends on two terms: the second largest eigenvalue of each condensation operator, and the distance between their stationary distributions. The convergence proof relies on this theorem.

Theorem 4.4. *For a condensation step t and a collection of diffusion operators $\{\mathbf{P}_k\}_{k=0}^t$, we have the following bound on the norm of the nonconstant term of the function $\mathbf{P}^{(t)} f$:*

$$\|H_t(\mathbf{P}^{(t)} f)\|_2 \le \|d_0\|_\infty^{1/2} \left[\prod_{i=0}^{t-1} \lambda_{i,2}\right] \left[\prod_{i=0}^{t-1} (1 + N^{1/2}\|d_i - d_{i+1}\|_2)^2\right] \|f\|_2.$$

Proof. We start by proving the following inequality:

$$\|H_t(\mathbf{P}^{(t-1)} f)\|_{d_t} \le \left[\prod_{i=0}^{t-1} \lambda_{i,2}\right] \left[\prod_{i=0}^{t-1} (1 + N^{1/2}\|d_i - d_{i+1}\|_2)^2\right] \|f\|_{d_0}. \tag{4.3}$$

We will prove (4.3) by induction. For $t = 1$, using Lemma 4.2 we obtain

$$\|H_1(\mathbf{P}_0 f)\|_{d_1} \le (1 + N^{1/2}\|d_0 - d_1\|_2)\|H_1(\mathbf{P}_0 f)\|_{d_0}. \tag{4.4}$$

Moreover, using the fact that $H_1(f) = L_0(f) - L_1(f) + H_0(f)$, we write

$$\begin{aligned} \|H_1(\mathbf{P}_0 f)\|_{d_0} &\le \|L_0(\mathbf{P}_0 f) - L_1(\mathbf{P}_0 f)\|_{d_0} + \|H_0(\mathbf{P}_0 f)\|_{d_0} \\ &\le \lambda_{0,2} N^{1/2}\|d_0 - d_1\|_2 \|H_0(f)\|_{d_0} + \lambda_{0,2}\|H_0(f)\|_{d_0} \qquad (4.5) \\ &\le \lambda_{0,2}(1 + N^{1/2}\|d_0 - d_1\|_2)\|f\|_{d_0}, \qquad (4.6) \end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

where the inequality (4.5) is obtained by Lemmas 4.3 and 4.1. Combining (4.4) and (4.6) yields $\|H_1(\mathbf{P}_0 f)\|_{d_0} \leq \lambda_{0,2}(1+N^{1/2}\|d_0 - d_1\|_2)^2\|f\|_{d_0}$. We have shown that (4.3) is true for $t=1$. Now assume it is true up to $t-1$; we want to show that this implies that it is true for t. The proof is similar to the base case. From Lemma 4.2, we obtain

$$\|H_t(\mathbf{P}^{(t-1)} f)\|_{d_t} \leq (1+N^{1/2}\|d_{t-1} - d_t\|_2)\|H_t(\mathbf{P}^{(t-1)} f)\|_{d_{t-1}}. \tag{4.7}$$

Moreover, following the same steps as in (4.6) with Lemmas 4.3 and 4.1, we obtain $\|H_t(\mathbf{P}^{(t-1)} f)\|_{d_{t-1}} \leq \lambda_{t-1,2}(1+N^{1/2}\|d_{t-1} - d_t\|_2)\|H_{t-1}(\mathbf{P}^{(t-2)} f)\|_{d_{t-1}}$. Combining this last inequality with (4.7) yields $\|H_t(\mathbf{P}^{(t-1)} f)\|_{d_t} \leq \lambda_{t-1,2}(1+N^{1/2}\|d_{t-1} - d_t\|_2)^2\|H_{t-1}(\mathbf{P}^{(t-2)} f)\|_{d_{t-1}}$, and we prove (4.3) by applying the inductive hypothesis. We can now conclude the proof by showing

$$\|H_t(\mathbf{P}^{(t)} f)\|_2 \leq \|1/d_t\|_\infty^{1/2}\|H_t(\mathbf{P}^{(t)} f)\|_{d_t} \leq \|H_t(\mathbf{P}^{(t)} f)\|_{d_t} \tag{4.8}$$

$$\leq \lambda_{t,2}\|H_t(\mathbf{P}^{(t-1)} f)\|_{d_t} \leq \|H_t(\mathbf{P}^{(t-1)} f)\|_{d_t} \tag{4.9}$$

$$\leq \left[\prod_{i=0}^{t-1}\lambda_{i,2}\right]\left[\prod_{i=0}^{t-1}(1+N^{1/2}\|d_i - d_{i+1}\|_2)^2\right]\|f\|_{d_0} \tag{4.10}$$

$$\leq \|d_0\|_\infty^{1/2}\left[\prod_{i=0}^{t-1}\lambda_{i,2}\right]\left[\prod_{i=0}^{t-1}(1+N^{1/2}\|d_i - d_{i+1}\|_2)^2\right]\|f\|_2,$$

where the inequalities (4.8) and (4.9) are obtained by Lemma 4.2 and Lemma 4.1, and the inequality (4.10) is justified by (4.3), which we have just shown. Finally, the last inequality is due to Lemma 4.2. ∎

Remark 4.5. Theorem 4.4 is valid for a general collection of diffusion operators constructed from a kernel like the ones presented in subsection 2.5. In particular, this collection includes operators created by the diffusion condensation algorithm and the time-homogeneous process. For the latter, the product $\prod(1+N^{1/2}\|d_i - d_{i+1}\|_2)^2$ is equal to one, and thus the rate of convergence only depends on the second largest eigenvalue of the diffusion operator. Allowing for time-inhomogeneity enables controlling the eigenvalue *during* the process, for example, by defining an adaptive bandwidth parameter, but comes at the cost of having to consider the rate of change of the degrees.

We recall our initial argument that to show the convergence of the condensation process, it is sufficient to use the coordinate function f_i and to show that the norm of the nonconstant term $\|H_t(\mathbf{P}^{(t)} f_i)\|_2$ converges to zero. This is achieved in the next corollary, for which we require the following assumption on successive degree functions:

$$\sum_{k=0}^{\infty}\|d_k - d_{k+1}\|_2 < \infty. \tag{4.11}$$

Corollary 4.6. *For a family of diffusion operators $\{\mathbf{P}_t\}_{t\in\mathbb{N}}$ defined by (4.1) such that their second largest eigenvalues are all less than or equal to $1-\delta$, where $\delta \in (0,1)$, and that $(d_t)_{t\in\mathbb{N}}$ respects (4.11), the condensation process converges to a (single) point as t tends to infinity.*

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Proof. Using the coordinate function f_i and the upper bound from Theorem 4.4, we have

$$\|H_t(\mathbf{P}^{(t)} f_i)\|_2 \leq \|d_0\|_\infty^{1/2} [1-\delta]^t \left[\prod_{k=0}^{t-1} (1 + N^{1/2} \|d_k - d_{k+1}\|_2)^2 \right] \|f_i\|_2, \tag{4.12}$$

since $\lambda_{t,2} \leq 1-\delta$ for all t. Note that the quantities $\|d_0\|_\infty^{1/2}$ and $\|f_i\|_2$ are both finite. Furthermore, by assumption, the sequence $(d_t)_{t\in\mathbb{N}}$ satisfies (4.11), and thus

$$\lim_{t\to\infty} \prod_{k=0}^{t-1} (1 + N^{1/2} \|d_k - d_{k+1}\|_2)^2 < \infty.$$

The upper bound converges to zero since $\lim_{t\to\infty}[1-\delta]^t = 0$, and because $\lim_{t\to\infty} \mathbf{P}^{(t)} f_i = \langle f_i, \mathbb{1}\rangle_\pi \mathbb{1}$, we conclude that all points have the same ith coordinate for all $i \in \{1, \ldots, d\}$. ∎

We conclude this section by identifying kernels for which we can find analytic conditions that respect the assumptions of the previous corollary and hence produce a condensation process that converges to a single point. First, we introduce the following lemma regarding the degrees assumption (4.11).

Lemma 4.7. *If $\lim_{k\to\infty} d_k$ exists, and the degrees are such that $d_k(i) \leq d_{k+1}(i)$ except for a finite number of condensation steps, then assumption (4.11) is verified.*

Proof. We note $d_\infty := \lim_{k\to\infty} d_k$ and recall $1 \leq d_k(i) \leq N$. Without loss of generality, we assume that all degrees after ℓ condensation steps respect the monotonic assumption, since $\|\cdot\|_2 \leq \|\cdot\|_1$, we will show $\sum_{k=\ell}^\infty \|d_k - d_{k+1}\|_1 < \infty$ to complete the proof. We have

$$\sum_{k=\ell}^{\infty} \sum_{i=1}^{N} |d_k(i) - d_{k+1}(i)| = \sum_{i=1}^{N} \sum_{k=\ell}^{\infty} d_{k+1}(i) - d_k(i) = \sum_{i=1}^{N} d_\infty(i) - d_\ell(i) \leq \sum_{i=1}^{N} (N-1) \leq \infty,$$

where the first equality comes from the increasing degrees assumption and by interchanging the order of summation, and the second equality is due to the telescoping sum. ∎

In the following, we assume the conditions of Lemma 4.7 to be verified. This is consistent with our experiments, as, after a few condensation steps, we observe that all pairwise distances decrease, and hence each dimension of the degrees is increasing. For the assumption on $\lambda_{i,2}$ we analyze the diffusion operator.

Since $\mathbf{P}_t$ is reversible with respect to π_t, we can use Proposition 1 of Diaconis and Stroock [10] to find an upper bound on the second largest eigenvalue. They show that $\lambda_{t,2} \leq 1 - 1/\kappa_t$, where

$$\kappa_t := \max_{i,j} \frac{\pi_t(i)\pi_t(j)}{\pi_t(i)\mathbf{P}_t(i,j)} \leq \max_{i,j} \frac{d_t(i)}{\mathbf{K}_t(i,j)} \leq \frac{d_{max,t}}{\min_{i,j} \mathbf{K}_t(i,j)},$$

and $d_{max,t} := \max_i d_t(i)$. To respect the assumptions of Corollary 4.6, we define ϵ_t such that

$$\lambda_{t,2} \leq 1 - \frac{\min_{x,y\in \mathsf{X}_t} k_\epsilon(x,y)}{d_{max,t}} \leq 1 - \delta.$$

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Thus, for the α-decay kernel (Definition 2.3), we must define a schedule of the bandwidth parameter ϵ_t, such that $\epsilon_t^\alpha \geq -\text{diam}(\mathsf{X}_t)^\alpha/(-\log(\delta d_{max,t}))$, which extends schedules for the Gaussian and Laplace kernels. For these kernels, we always need $\delta \in (0, 1/d_{max,t})$, but we note that $d_{max,t} \leq N$, which thus avoids the case where δ tends to 0 as t tends to infinity. A similar result is obtained for the density normalized kernel $k_{\epsilon,\beta}$ (Definition 2.6), since

$$\max_{x,y\in\mathsf{X}_t} \sum_y k_{\epsilon,\beta}(x,y) \leq N \quad \text{and} \quad \min_{x,y\in\mathsf{X}_t} k_{\epsilon,\beta}(x,y) \geq \frac{\min_{x,y\in\mathsf{X}_t} k(x,y)}{q_{max,t}^{2\beta}}.$$

Combining these two bounds yields the following requirement for the density normalized kernel

$$\lambda_{t,2} \leq 1 - \frac{\min_{x,y} k_\epsilon(x,y)}{N q_{max,t}^{2\beta}} \leq 1 - \delta.$$

We can find a similar schedule for each of the previous kernels, since $\min_{x,y} k_t(x,y)$ can be lower bounded by a function of the diameter. For instance, we find $\epsilon_t \geq -\text{diam}(\mathsf{X}_t)^2/\log(\delta N q_{max,t}^{2\beta})$ for the anisotropic Gaussian kernel. This adaptive parametrization of the bandwidth parameter guarantees that the condensation process will converge to a point for these kernels.

Remark 4.8. These results can be generalized to $\mathbf{P}_t^\tau$, for any $\tau \in \mathbb{N}$. We can write $\mathbf{P}_t^\tau f = L_t(f) + \sum_k \lambda_{t,k}^\tau \langle f, \psi_{t,k}\rangle_{d_t} \psi_{t,k}$, and hence $\|H_t(\mathbf{P}_t^\tau f)\|_{d_t} \leq \|H_t(\mathbf{P}_t f)\|_{d_t} \leq \lambda_{t,2}\|H_t(f)\|_{d_t}$, since $|\lambda_{t,2}| \leq 1$. Thus, Theorem 4.4 can be used to prove convergence of the process.

Remark 4.9. Both Theorem 4.4 and Corollary 4.6 are valid for a broad class of diffusion operators, in particular those with finite support or a wider family of random walks on a graph. This differs from the geometric Theorem 3.2, which is restricted to strictly positive kernels. It is also possible to leverage information from the underlying structure of the data to characterize the convergence of the condensation process. For example, for a random walk on a graph, the second-largest eigenvalue is influenced by the connectivity of the graph; a highly connected graph would yield a small eigenvalue and hence converge faster. For the box kernel, assuming monotone convergence of degrees, Corollary 4.6 can be used to analyze overall convergence by evaluating the second-largest eigenvalue at different condensation times.

Remark 4.10. The degree convergence assumption (4.11) assumes that the process converges to a stable representation X_M, without any assumption on X_M. In practice, since transition operators are contractive, we observe that this assumption is easily respected (from Lemma 4.7). It is worth noting that (4.11) can be controlled for random walks on k-nearest-neighbor graphs (since $d_t(i) = k$). Corollary 4.6, bounding the second-largest eigenvalue, then guarantees that X_M is a single point.

5. Topological properties of the condensation process.

Having previously proved convergence properties, we now take on a coarser perspective and characterize topological, i.e., *structural*, properties of the diffusion condensation process. To this end, we note that the multiresolution structure provided by diffusion condensation naturally relates to recent advances in using computational topology to understand the "shape" of data geometry at varying scales. To elucidate this connection, subsections 5.2 and 5.3 introduce two perspectives for integrating topological information into the data geometry uncovered by the diffusion condensation process, i.e., (i) *condensation homology* for describing the topology of the diffusion

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

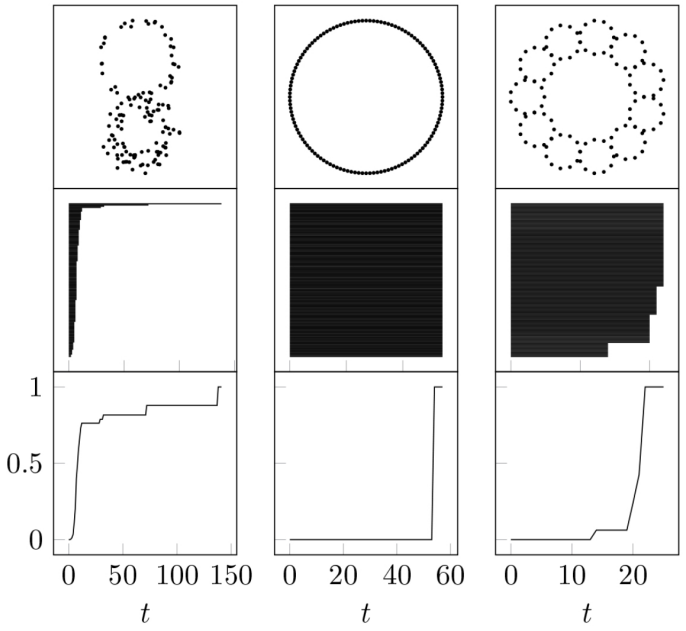


Figure 6. *An illustration of condensation homology for the "double annulus" dataset, the "hyperuniform circle" dataset where n points are evenly spaced around the circle with $\frac{2\pi}{n}$ radians between them, and the "petals" dataset. The top row depicts the original dataset at $t = 0$; the middle row depicts the condensation homology barcode, i.e., a summary of topological activity over all condensation iterations; and the bottom row depicts topological activity curves (cumulative sums of lengths in the condensation homology barcode).*

condensation process itself, and (ii) *persistent homology* based on Vietoris–Rips complexes for describing each step of the diffusion condensation process, thus closing the loop to the previously provided geometric notions. We provide a brief review of relevant topological data analysis (TDA) notions in subsection 5.1 and in the supplementary material (supplementary_material_dc.pdf [local/web 321KB]). Figures 6 and 7 depict the two types of topological descriptions. Readers familiar with TDA may recognize that our two perspectives may also be seen as slices of a special bifiltration, i.e., a filtration with two parameters. However, since bifiltrations are known to be computationally more challenging [23], we defer their treatment to future work.

5.1. A brief summary of persistent homology. Persistent homology [1, 12] is a method from the field of computational topology, which develops tools for obtaining and analyzing topological features of datasets. Given its beneficial robustness properties [8], persistent homology has received a large degree of attention from the machine learning community [17].

We first introduce the underlying concept of simplicial homology. For a simplicial complex K, i.e., a generalized graph with higher-order connectivity information in the form of cliques, simplicial homology employs matrix reduction algorithms to assign K a family of groups, the *homology groups*. The dth homology group $H_d(\mathrm{K})$ of K contains equivalence classes of d-dimensional topological features, such as connected components ($d = 0$), cycles/tunnels ($d = 1$), and voids ($d = 2$). These features are also known as homology classes. Homology groups are typically summarized by their ranks, thereby obtaining a simple invariant

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

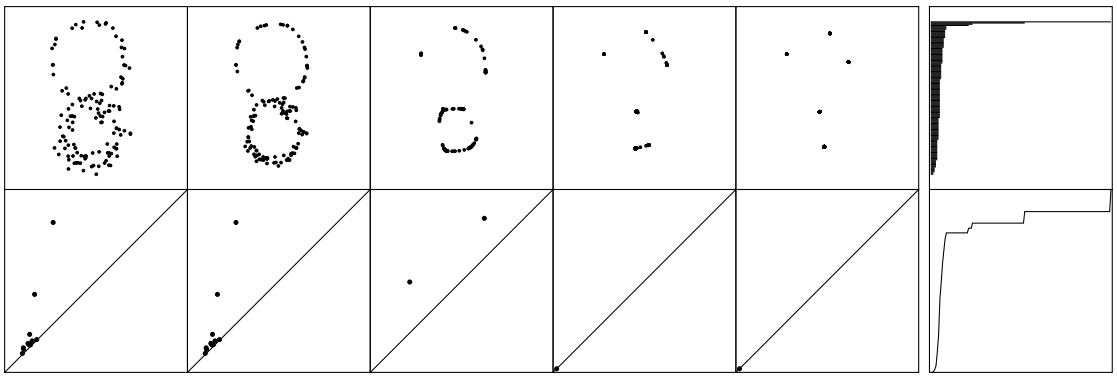


Figure 7. (a) *Persistent homology for the "double annulus" dataset. Following different steps in the diffusion condensation process (top row), we obtain a sequence of persistence diagrams (bottom row) that summarize the one-dimensional topological features, i.e., the cycles, in the dataset.* (b) *The condensation homology (top row) and the topological activity curve (bottom row) of the dataset for comparison purposes.*

"signature" of a manifold. For instance, a circle in $\mathbb{R}^2$ has one feature with $d = 1$, i.e., a cycle, and one feature with $d = 0$, i.e., a connected component. In practice, we are dealing with a point cloud X and a metric, such as the Euclidean distance. In this setting, *persistent homology* now creates a sequence of nested simplicial complexes, making it possible to track the changes in homology groups—and thus the changes in topology—over multiple scales (with the understanding that real-world datasets necessitate such a multiscale perspective, as a single scale is too restrictive). This is achieved by constructing a special simplicial complex, the Vietoris–Rips complex [33]. For $0 \leq \epsilon < \infty$, the Vietoris–Rips complex of X at scale ϵ, denoted by $\mathcal{V}_\epsilon(\mathsf{X})$, contains all simplices (i.e., subsets) of X whose elements $\{x_0, x_1, \ldots\}$ satisfy $\mathrm{d}(x_i, x_j) \leq \epsilon$ for all i, j. Calculating topological features of $\mathcal{V}_\epsilon$ results in a set of tuples of the form $(\epsilon_i, \epsilon_j, d)$, where $\epsilon_i \in \mathbb{R}$ refers to a threshold at which a topological feature was "created," i.e., the threshold at which it occurred for the first time in $\mathcal{V}_\epsilon$. Likewise, $\epsilon_j \in \mathbb{R}$ refers to the threshold at which the feature was destroyed. Lastly, d indicates the dimension of the respective feature. Together, the features of dimension d form the d-dimensional *persistence diagram*, a topological descriptor containing the point (ϵ_i, ϵ_j) for every such tuple above. For example, when $d = 0$, the threshold ϵ_j denotes at which distance two connected components in a dataset are merged into one.

5.2. Condensation homology. The formulation of the diffusion condensation process, with its merge step for close points, induces changes in the topological structure of the datasets. This will result in *one* topological descriptor summarizing each dataset. We first define a filtration, i.e., an ordering of subsets of the data, such that we obtain a sequence of nested simplicial complexes that is intrinsic to the diffusion condensation process and compatible with a citation of Algorithm 2.1. The filtration is based on the idea of first extracting subsets of the data that satisfy a pairwise distance requirement—similarly to the Vietoris–Rips filtration, which we shall describe in subsection 5.3—and assign them a weight based on the condensation time t. This weight is used to track topological changes during the condensation process.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Definition 5.1 (condensation homology filtration). *Given a merge threshold $\zeta \in \mathbb{R}_{>0}$, we define the intrinsic condensation filtration for $t \in \mathbb{N}$ as the filtration arising from the sequence of simplicial complexes*

$$\mathcal{V}_t(\mathsf{X},\zeta) := \{\sigma \subseteq \mathsf{X}_t \mid \mathrm{d}\left(x_t\left(i\right), x_t\left(j\right)\right) \leq \zeta \text{ for all } x(i), x(j) \in \sigma\} \bigcup_{t'=0}^{t-1} \mathcal{V}_{t'}(\mathsf{X},\zeta), \tag{5.1}$$

with $\mathcal{V}_0(\mathsf{X},\zeta) := \{\sigma \subseteq \mathsf{X} \mid \mathrm{d}(x(i),x(j)) \leq \zeta\}$. The weight function $\mathrm{w}\colon 2^{\mathsf{X}} \to \mathbb{N}$ for each $\mathcal{V}_t(\mathsf{X},\zeta)$ is defined by setting $\mathrm{w}(\{i\}) := 0$ for a 0-simplex $\{i\}$, and by setting $\mathrm{w}(\{i,j\}) := \min\{t \mid \{i,j\} \in \mathcal{V}_t(\mathsf{X},\zeta)\}$ for each 1-simplex $\{i,j\}$, i.e., we use the first t such that the two points are in a ζ-neighborhood. The weight function can be extended to higher-dimensional simplices inductively by taking the maximum.

Lemma 5.2. *Using (5.1) results in a nested sequence of simplicial complexes. We thus obtain a valid filtration from which we may calculate topological features.*

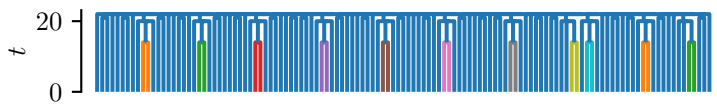
Proof. The nesting property is achieved by taking the union in (5.1). Hence, $\mathcal{V}_t(\mathsf{X},\zeta)$ can only grow, which ensures that consecutive complexes are nested. The weights are not guaranteed to be unique, but we obtain a consistent ordering by using the indices of the respective points. ∎

Intuitively, the condensation homology filtration measures at which iteration step t two points move into their ζ-neighborhood for the first time. There are two differences between this and the traditional Vietoris–Rips filtration used in subsection 5.3. First, we enforce the nesting condition of a filtration by taking the union of all simplicial complexes for previous time steps; this is necessary because, depending on the threshold ζ, we cannot guarantee that points remain within a ζ-neighborhood.[1] The second difference is that we filter over diffusion condensation iterations instead of distance thresholds, necessitating the use of an additional weight function (as opposed to using the distances between points). Since diffusion condensation results in changes of local distances, this filtration captures the intrinsic behavior of the process. For now, we only add 1-simplices and 0-simplices to every $\mathcal{V}_t(\mathsf{X},\zeta)$, but the definition generalizes to higher-order simplices. We define *condensation homology* to be the degree-0 persistent homology of $\mathcal{V}_t(\mathsf{X},\zeta)$ under the weight function defined above.

Intuitively, we initially treat each data point x_i as a 0-simplex, creating its own homology class, and identify homology classes over different time steps t, i.e., the homology class of $x_t(i)$ and $x_{t'}(i)$ for $t \neq t'$ is considered to be the same. As the geometry of the underlying point cloud changes during each iteration, points start to progressively cluster. Whenever a *merge* event happens (see line 9 in the diffusion condensation Algorithm 2.1), we let the homology class corresponding to the vertex with the lower index continue, while we *destroy* the other homology class. Given our weight function, such an event results in a tuple of the form $(0,t)$, where t denotes the diffusion condensation iteration. This can also be considered as a persistence diagram arising from a distance-based filtration of an abstract input dataset (hence, every tuple contains a 0; all homology classes—i.e., all vertices—are present at the

[1] For readers familiar with computational topology, we want to remark that using zigzag persistent homology [3], which does not require stringent nesting conditions for filtrations, would also be a possibility. We will consider such a perspective in future work.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.



(a) Gaussian kernel

(b) α-decay kernel

Figure 8. *Two dendrograms obtained on the "petals" dataset. The different condensation behaviors exhibited by different kernels (see Figure 2) also manifests in the dendrograms.*

start of the diffusion condensation process). Our weight function can be interpreted as a "temporal distance"; the distance between pairs of vertices (i, j) is given by calculating the value of t for which their spatial distance falls below the merge threshold ζ for the first time, i.e., $\mathrm{d}(i,j) := \min\{t \mid \mathrm{d}(x_t(i), x_t(j)) \leq \zeta\}$. A convenient representation can be obtained using a *persistence barcode* [15], i.e., a representation in which the lifespan of each homology class is depicted using a bar. Longer bars indicate more prominent clusters or groupings in data [20]. Figure 6 illustrates this for a "double annulus" dataset, which does not give rise to a complex set of clusters, as indicated by the existence of few long bars in such a barcode.

Notice that the persistence pairing $\mathcal{P}$ corresponding to the condensation homology carries all the information about the hierarchy of merges obtained during the diffusion condensation process. Specifically, $\mathcal{P}$ consists of pairs of the form $(\{u\}, \{v, w\})$, where $\{u\}$ is a vertex and $\{v, w\}$ is an edge between two vertices. We can use these edges to construct a tree of merges, i.e., a *dendrogram* (see Figure 8). This perspective will be useful later on when we show how diffusion condensation generalizes existing hierarchical clustering methods.

5.3. Persistent homology of the diffusion process. As a more expressive—but also more complicated—description of topological features in the condensation process, we calculate persistent homology of the input dataset X at every condensation iteration. To this end, we calculate a Vietoris–Rips complex for each point cloud X_t of the diffusion condensation process, denoting the Vietoris–-Rips complex of X at diffusion time t as $\mathcal{V}_\zeta(\mathsf{X}, t) := \{\sigma \subseteq \mathsf{X}_t \mid \mathrm{d}(x_t(i), x_t(j)) \leq \zeta \text{ for all } x(i), x(j) \in \sigma\}$ (this notation was chosen to contrast with $\mathcal{V}_t(\mathsf{X}, \zeta)$ from (5.1), in which t is varied as the filtration parameter while ζ is kept fixed). In the following, we will prove that the topological features of $\mathcal{V}_\zeta(\mathsf{X}, t)$ converge as the diffusion condensation process converges. To this end, we make use of the *bottleneck distance* $\mathrm{d_b}(\cdot, \cdot)$, a distance metric between persistence diagrams, defined as

$$\mathrm{d_b}(\mathcal{D}, \mathcal{D}') = \inf_{\eta\colon \mathcal{D} \to \mathcal{D}'} \sup_{x \in \mathcal{D}} \|x - \eta(x)\|_\infty, \tag{5.2}$$

where $\eta\colon \mathcal{D} \to \mathcal{D}'$ denotes a bijection between the point sets of both diagrams, and $\|\cdot\|_\infty$ refers to the L_∞ metric between two points in $\mathbb{R}^2$. Using the results from section 3, we can bound

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

the topological activity and prove convergence in terms of topological properties. Specifically, for the 0-dimensional persistence diagram of our input dataset at diffusion time t, which we subsequently denote by $\mathcal{D}_{\mathsf{X}_t}$, we prove that the bottleneck distance $\mathrm{d_b}(\mathcal{D}_{\mathsf{X}_t}, \mathcal{D}_{\mathsf{X}_{t'}})$ to another time step t' is upper-bounded by the respective diameters of the point clouds.

Theorem 5.3. *Let $t \leq t'$ refer to two iterations of the diffusion condensation process with $\mathsf{X}_t, \mathsf{X}_{t'}$ denoting their corresponding point clouds. If $\operatorname{diam}(\mathsf{X}_t) \geq \operatorname{diam}(\mathsf{X}_{t'})$, then the persistence diagrams corresponding to X_t and $\mathsf{X}_{t'}$ satisfy*

$$\mathrm{d_b}(\mathcal{D}_{\mathsf{X}_t}, \mathcal{D}_{\mathsf{X}_{t'}}) \leq \operatorname{diam}(\mathsf{X}_t). \tag{5.3}$$

Proof. From Chazal et al. [4, 5], we obtain $\mathrm{d_b}(\mathcal{D}_{\mathsf{X}_t}, \mathcal{D}_{\mathsf{X}_{t'}}) \leq 2\mathrm{d_{GH}}(\mathsf{X}_t, \mathsf{X}_{t'})$, where $\mathrm{d_{GH}}(\cdot,\cdot)$ denotes the Gromov–Hausdorff distance. According to Mémoli [26, Proposition 5], we have $\mathrm{d_{GH}}(\mathsf{X}_t, \mathsf{X}_{t'}) \leq 1/2 \max\{\operatorname{diam}(\mathsf{X}_t), \operatorname{diam}(\mathsf{X}_{t'})\}$, so we can simplify the bound to $\mathrm{d_b}(\mathcal{D}_{\mathsf{X}_t}, \mathcal{D}_{\mathsf{X}_{t'}}) \leq \max\{\operatorname{diam}(\mathsf{X}_t), \operatorname{diam}(\mathsf{X}_{t'})\}$. As $\operatorname{diam}(\mathsf{X}_t) \geq \operatorname{diam}(\mathsf{X}_{t'})$, we have $\mathrm{d_b}(\mathcal{D}_{\mathsf{X}_t}, \mathcal{D}_{\mathsf{X}_{t'}}) \leq \operatorname{diam}(\mathsf{X}_t)$. ∎

Under the conditions of Corollary 4.6, i.e., for a large family of diffusion operators, we know that diffusion condensation converges to a point, thus implying $\lim_{t\to\infty} \operatorname{diam}(\mathsf{X}_t) = 0$. While we cannot guarantee that $\operatorname{diam}(\mathsf{X}_t) \geq \operatorname{diam}(\mathsf{X}_{t'})$ for $t \leq t'$ holds in general (in the setting of Corollary 4.6, the diameter can increase; in the more restrictive setting of Theorem 3.2, diameters would also be nonincreasing, but that theorem only applies to strictly pointwise positive kernels), we know that there exists a subsequence of condensation steps $\{\widetilde{t}\}$ such that the diameter is nonincreasing. For this subsequence, the bottleneck distance between consecutive datasets, i.e., $\mathrm{d_b}(\mathcal{D}_{\mathsf{X}_{\widetilde{t}}}, \mathcal{D}_{\mathsf{X}_{\widetilde{t+1}}})$, also converges to 0. By contrast, $\mathrm{d_{GH}}(\mathsf{X}_{\widetilde{t}}, \mathsf{X}_{\widetilde{t+1}}) \geq 1/2|\operatorname{diam}(\mathsf{X}_{\widetilde{t}}) - \operatorname{diam}(\mathsf{X}_{\widetilde{t+1}})|$ (the bound being tight in certain cases), implying that the bottleneck distance between consecutive time steps is never zero if the diameter changes. Since all point clouds are embedded into the same space, namely $\mathbb{R}^d$, all preceding statements apply, with the Hausdorff distance $\mathrm{d_H}(\cdot,\cdot)$ replacing the Gromov–Hausdorff distance [6]. This distance has the advantage that we can easily evaluate it. We require one auxiliary lemma to replace the diameter bound in the previous proof.

Lemma 5.4. *Let X, Y be subsets of the same metric space, e.g., $\mathbb{R}^d$, with $\operatorname{conv}(\mathsf{Y}) \subsetneq \operatorname{conv}(\mathsf{X})$. Then $\mathrm{d_H}(\mathsf{X}, \mathsf{Y}) \leq \operatorname{diam}(\mathsf{X})$.*

Proof. The Hausdorff distance is the smallest r-thickening required such that both X and Y become subsets of each other, i.e., $\mathrm{d_H}(\mathsf{X}, \mathsf{Y}) := \inf\{r > 0 \mid \mathsf{Y} \subseteq \mathsf{X}^{(r)} \text{ and } \mathsf{X} \subseteq \mathsf{Y}^{(r)}\}$. Since $\operatorname{conv}(\mathsf{Y}) \subsetneq \operatorname{conv}(\mathsf{X})$, we have $r \leq \operatorname{diam}(\mathsf{X})$. ∎

As a consequence of this lemma and the preceding proof, we obtain a bound in terms of the Hausdorff distance and the diameter. For $t \leq t'$, we have

$$\mathrm{d_b}(\mathcal{D}_{\mathsf{X}_t}, \mathcal{D}_{\mathsf{X}_{t'}}) \leq 2\mathrm{d_H}(\mathsf{X}_t, \mathsf{X}_{t'}) \leq \operatorname{diam}(\mathsf{X}_t).$$

Figure 9 shows empirical convergence behavior between consecutive time steps, illustrating how different condensation processes are characterized by different diameter shrinkages.

5.4. Hierarchical clustering. In contrast to existing work on multiscale diffusion-based clustering [29], diffusion condensation changes the underlying geometric-topological structure of the data to extract hierarchical information. A topological perspective helps us elucidate

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

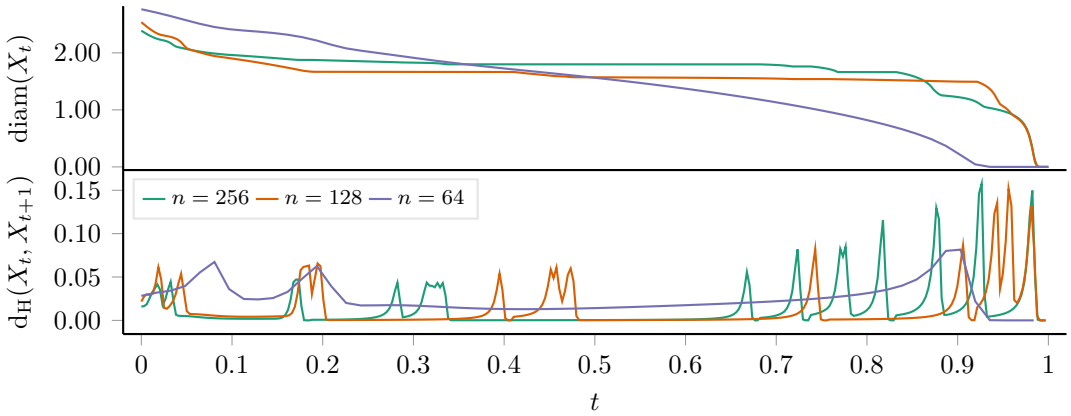


Figure 9. 2*Empirical convergence behavior of point cloud diameters (for the "petals" dataset at various sample sizes n) and the Hausdorff distance between consecutive steps $i, i+1$ of the condensation process (used as a proxy for the bottleneck distance). Convergence behavior with respect to the Hausdorff distance is not uniform and characterized by some "jumps", indicating that the datasets change considerably between certain time steps, before achieving a stable configuration.*

connections to hierarchical clustering, a clustering method based on measuring dissimilarities between clusters via *linkage methods*. While there are many linkage methods for measuring the association between clusters in agglomerative hierarchical clustering, we focus on the centroid method, as it is the most relevant to diffusion condensation. Agglomerative clustering, and the centroid method specifically, is widely applied in phylogeny [11], sequence alignment [18], and analysis of other types of data [19]. In the centroid method, the distance between any two clusters a and b is defined as the distance between the centroids of the clusters. There are two natural definitions of Euclidean centroids, leading to the unweighted pair group method with centroid mean (UPGMC) [30] and to the weighted version (WPGMC), also known as median linkage hierarchical clustering [16]. The unweighted centroid C_{UPGMC} is the centroid of all points in the cluster:

$$C_{\mathrm{UPGMC}}(a) = \frac{1}{|a|}\sum_{x\in a} x. \tag{5.4}$$

In contrast, the weighted version depends on the parent clusters: suppose cluster a is formed by the merging of clusters b and c; then the centroid of a is defined as

$$C_{\mathrm{WPGMC}}(a) = \frac{C_{\mathrm{WPGMC}}(b) + C_{\mathrm{WPGMC}}(c)}{2}. \tag{5.5}$$

In either case, the distance between two clusters a and b is defined as the squared Euclidean distance between their centroids, denoted by $D(a,b) := \|C(a) - C(b)\|^2$. This is detailed in Algorithm 5.1 (with $\oplus$ referring to sequence concatenation). For a given dataset, the UPGMC and WPGMC algorithms give a unique sequence of merges, provided that at each iteration there exists a unique choice of centroids a^* and b^* that achieve the minimum distance between clusters. These methods are similar to diffusion condensation and, in certain situations, equivalent; our next theorem makes this more precise.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Algorithm 5.1 Centroid hierarchical agglomerative clustering.

Input: set of points X_0
Output: the set of clusters at each level $(L_0, L_1, \ldots, L_{N-1})$
$L_0 \leftarrow \{\{x(1)\}, \{x(2)\}, \ldots, \{x(N)\}\}$ ▷ *Initially, every point is its own cluster*
for $t \in \{1, \ldots, N-1\}$ **do**
  $a^*, b^* \leftarrow \arg\min_{(a,b)\in(L_{t-1})^2} D(a,b)$ s.t. $a \neq b$ ▷ *Find centroids to merge*
  $L_t \leftarrow (L_{t-1} \setminus a^*) \setminus b^* \oplus (a^* \cup b^*)$ ▷ *Add new cluster with a^*, b^* merged*
end for

Theorem 5.5. *Let $\zeta = 0$, $\epsilon_t = \min_{x,y\in\mathsf{X}_t} \|x-y\|_2 > 0$, and $k_t(x,y)$ be the box kernel in Definition 2.2. In this case, the diffusion condensation produces topological features equivalent to centroid agglomerative clustering (UPGMC) in both diffusion homology and persistent homology (for $\zeta = 0$), i.e., $\mathcal{V}_0(\mathsf{X}, t) = L_t$ for all t. Further, if $\zeta : 0 < \zeta < \epsilon_t$, then diffusion condensation is similarly equivalent in both condensation homology and persistent homology to median linkage agglomerative clustering (WPGMC).*

Proof. To show the equivalence of diffusion condensation to UPGMC algorithms, we show that (i) centroids in the UPGMC algorithm correspond to points in the diffusion condensation algorithm, and (ii) the same clusters—represented by their respective centroids—are merged in each iteration, i.e.,$\mathcal{V}_0(\mathsf{X}, t)$ is a representation of the hierarchy of UPGMC at iteration t.

(i) We show the first claim by induction. For the condensation algorithm at $t = 0$, the claim is trivially true, since all points are singletons and therefore centroids. By the induction hypothesis, all points are centroids at time t, and we show that it still holds at time $t+1$. Without loss of generality, we assume that only a single pair of points achieves the minimal pairwise distance at time t, say $(x_t(k), x_t(l))$.[2] We have $\epsilon_t = \min_{x,y\in\mathsf{X}_t} \|x-y\|_2 > 0$, and by construction of the box kernel $\mathbf{K}_t(k,l) = \mathbf{K}_t(l,k) = 1$ and zero otherwise. Thus $x_{t+1}(k) = \mathbf{P}_t(k,\cdot)\mathsf{X}_t = \mathbf{P}_t(l,\cdot)\mathsf{X}_t = x_{t+1}(l)$, and hence only the two points with minimum distance will be merged at their midpoint (centroids), creating a new centroid. Since $\zeta = 0$, this will create a sequence of merges.

(ii) Just like UPGMC, only centroids with minimal distance are merged at every iteration. For each merge in the condensation algorithm, a tuple of the form $(0, t)$ is created in the condensation homology persistence diagram (and the respective pairs are created in its persistence pairing). Therefore, points in the diffusion condensation process are equivalent to centroids in the UPGMC algorithm, and the same merges happen in each iteration. Hence, $\mathcal{V}_0(\mathsf{X}, t)$, the Vietoris–Rips complex of X_t at scale 0, is a representation of the hierarchy of UPGMC at iteration t.

For the setting of $\zeta : 0 < \zeta < \epsilon_t$, the only difference is in the setting of the merge threshold. The proof follows the same logic as in the previous theorem (assuming again that pairwise distances in each step are unique) except that the centroid locations are updated as the average of two

[2]This is equivalent to assuming that all pairwise distances in the current diffusion condensation step are unique. Said assumption also ensures that the selection of a^* and b^* in Algorithm 5.1 is unique, so it is a useful requirement. It does *not* decrease the generality of our argumentation (in fact, a consistent ordering of merges can always be achieved), but it simplifies notation and discussion.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

points instead of a weighted average of all points in the two clusters, and hence we have the equivalence with WPGMC. ∎

Remark 5.6. Theorem 5.5 implies that diffusion condensation converges to a point in $N-1$ iterations, as each iteration reduces the number of unique point locations by one. Extending this logic to more general settings (where the number of unique points might not strictly decrease in each iteration) is not trivial and is left to future work.

Remark 5.7. This theorem motivates interpreting diffusion condensation as a soft hierarchical clustering method, particularly with other kernels and in situations where the general position assumption does not hold. When points are equally spaced and not naturally clusterable, we find diffusion condensation more appealing; for instance, consider the corners of a k-dimensional simplex, with all distances between points being equal. The only two "sensible" clusterings are k clusters of single points, or one cluster with k points. Performing agglomerative clustering on this dataset will result in an arbitrary binary tree over the data, where all levels of the tree result in meaningless clusters. Diffusion condensation with any radial kernel, ϵ schedule, and merge threshold will result in exactly these two clusterings.

Remark 5.8. This soft clustering interpretation also hints at a convergence result with potentially tighter bounds for general kernels. The geometric results in section 3 rely on a pointwise lower bound of the kernel; this can lead to pessimistic convergence results on kernels similar to the box kernel (for example, consider the α-decay kernel with large α), which act more like hierarchical clustering but have poor tail bounds. An interesting future direction would be to explore geometric convergence for general kernels in terms of the number of unique points rather than the diameter, following the line of reasoning in Theorem 5.5.

6. Discussion. Diffusion condensation is a process that alternates between computing a data diffusion operator and reapplying the operator back on the data to gradually eliminate variation. In this paper, we analyzed the diffusion condensation process from two main perspectives: its convergence and the evolution of its shape through condensation steps. We found conditions guaranteeing the convergence of the process using both geometric and spectral arguments. The geometric argument shows that the convex hull of each iteration of data after condensation shrinks in comparison to the previous iteration. The spectral argument reasons that the second largest eigenvalues of the data graph bounds the result of any function multiplied by the diffusion operator. Our spectral results are of particular interest since they are valid for a broad family of diffusion operators creating a time-inhomogeneous process.

Further, we used and extended tools from topological data analysis to characterize the evolution of the shape of the datasets during the condensation process. In particular, we defined the condensation homology filtration that operates on the data manifold and studied the resulting condensation homology. This provides us with a summary of the topological features during the entire process. Since the process is guaranteed to converge, the filtration will sweep through the different resolutions of the data, hence providing meaningful details. With the persistent diffusion homology, we studied the topological features for a given condensation step, resulting in snapshots of topological characteristics of the process. Furthermore, we provided experiments showcasing the relevance of our analysis, specifically comparing the condensation and persistent homologies, and the usage of condensation for clustering purposes.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

We also showed that instances of diffusion condensation with the box kernel are equivalent to hierarchical clustering algorithms. In future work we would like to extend this equivalence result to other "softer" kernels. This could potentially give a tighter convergence bound dependent on the concentration of a kernel and the number of points rather than its tail, which can lead to pessimistic bounds on k-nearest-neighbor random walks. Additionally, we would like to extend the definition of the intrinsic condensation filtration to multidimensional filtrations, for instance, by identifying the cycles or considering path probabilities defined by the diffusion kernel.

REFERENCES

[1] S. A. Barannikov, *The framed Morse complex and its invariants*, Adv. Soviet Math., 21 (1994), pp. 93–115.

[2] N. Brugnone, A. Gonopolskiy, M. W. Moyle, M. Kuchroo, D. van Dijk, K. R. Moon, D. Colon-Ramos, G. Wolf, M. J. Hirn, and S. Krishnaswamy, *Coarse graining of data via inhomogeneous diffusion condensation*, in Proceedings of the 2019 IEEE International Conference on Big Data, 2019, pp. 2624–2633.

[3] G. Carlsson, V. de Silva, and D. Morozov, *Zigzag persistent homology and real-valued functions*, in Proceedings of the Annual Symposium on Computational Geometry, 2009, pp. 247–256.

[4] F. Chazal, D. Cohen-Steiner, M. Glisse, L. J. Guibas, and S. Y. Oudot, *Proximity of persistence modules and their diagrams*, in Proceedings of the Twenty-Fifth Annual Symposium on Computational Geometry, Association for Computing Machinery, 2009, pp. 237–246.

[5] F. Chazal, V. de Silva, and S. Oudot, *Persistence stability for geometric complexes*, Geom. Dedicata, 173 (2014), pp. 193–214.

[6] F. Chazal, B. Fasy, F. Lecci, B. Michel, A. Rinaldo, and L. Wasserman, *Subsampling methods for persistent homology*, in Proceedings of the 32nd International Conference on Machine Learning, 2015, pp. 2143–2151.

[7] Y. Cheng, *Mean shift, mode seeking, and clustering*, IEEE Trans. Pattern Anal., 17 (1995), pp. 790–799.

[8] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer, *Stability of persistence diagrams*, Discrete Comput. Geom., 37 (2007), pp. 103–120.

[9] R. R. Coifman and S. Lafon, *Diffusion maps*, Appl. Comput. Harmon. Anal., 21 (2006), pp. 5–30.

[10] P. Diaconis and D. Stroock, *Geometric bounds for eigenvalues of Markov chains*, Ann. Appl. Probab., (1991), pp. 36–61.

[11] R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press, 1998.

[12] H. Edelsbrunner, D. Letscher, and A. J. Zomorodian, *Topological persistence and simplification*, Discrete Comput. Geom., 28 (2002), pp. 511–533.

[13] A. Feragen, F. Lauze, and S. Hauberg, *Geodesic exponential kernels: When curvature and linearity conflict*, in Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3032–3042.

[14] K. Fukunaga and L. Hostetler, *The estimation of the gradient of a density function, with applications in pattern recognition*, IEEE Trans. Inform. Theory, 21 (1975), pp. 32–40.

[15] R. Ghrist, *Barcodes: The persistent topology of data*, Bull. Amer. Math. Soc., 45 (2008), pp. 61–75.

[16] J. C. Gower, *A comparison of some methods of cluster analysis*, Biometrics, 23 (1967), pp. 623–637.

[17] F. Hensel, M. Moor, and B. Rieck, *A survey of topological machine learning methods*, Frontiers in Artificial Intelligence, 4 (2021).

[18] K. Katoh, *MAFFT: A novel method for rapid multiple sequence alignment based on fast Fourier transform*, Nucleic Acids Res., 30 (2002), pp. 3059–3066.

[19] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley Ser. Probab. Statist. 344, John Wiley & Sons, 2009.

[20] M. Kuchroo, M. DiStasio, E. Calapkulu, M. Ige, L. Zhang, A. H. Sheth, M. Menon, Y. Xing, S. Gigante, J. Huang, R. M. Dhodapkar, B. Rieck, G. Wolf, S. Krishnaswamy, and

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

B. P. Hafler, *Topological Analysis of Single-Cell Data Reveals Shared Glial Landscape of Macular Degeneration and Neurodegenerative Diseases*, preprint, bioRxiv, 2021.

[21] M. Kuchroo, J. Huang, P. Wong, J.-C. Grenier, D. Shung, A. Tong, C. Lucas, J. Klein, D. B. Burkhardt, S. Gigante, A. Godavarthi, B. Rieck, B. Israelow, M. Simonov, T. Mao, J. E. Oh, J. Silva, T. Takahashi, C. D. Odio, A. Casanovas-Massana, J. Fournier, Yale I. M. P. A. C. T. Team, A. Obaid, A. Moore, A. Lu-Culligan, A. Nelson, A. Brito, A. Nunez, A. Martin, A. L. Wyllie, A. Watkins, A. Park, A. Venkataraman, B. Geng, C. Kalinich, C. B. F. Vogels, C. Harden, C. Todeasa, C. Jensen, D. Kim, D. McDonald, D. Shepard, E. Courchaine, E. B. White, E. Song, E. Silva, E. Kudo, G. DeIuliis, H. Wang, H. Rahming, H.-J. Park, I. Matos, I. M. Ott, J. Nouws, J. Valdez, J. Fauver, J. Lim, K.-A. Rose, K. Anastasio, K. Brower, L. Glick, L. Sharma, L. Sewanan, L. Knaggs, M. Minasyan, M. Batsu, M. Tokuyama, M. C. Muenker, M. Petrone, M. Kuang, M. Nakahata, M. Campbell, M. Linehan, M. H. Askenase, M. Simonov, M. Smolgovsky, N. D. Grubaugh, N. Sonnert, N. Naushad, P. Vijayakumar, P. Lu, R. Earnest, R. Martinello, R. Herbst, R. Datta, R. Handoko, S. Bermejo, S. Lapidus, S. Prophet, S. Bickerton, S. Velazquez, S. Mohanty, T. Alpert, T. Rice, W. Schulz, W. Khoury-Hanold, X. Peng, Y. Yang, Y. Cao, Y. Strong, S. Farhadian, C. S. Dela Cruz, A. I. Ko, M. J. Hirn, F. P. Wilson, J. G. Hussin, G. Wolf, A. Iwasaki, and S. Krishnaswamy, *Multiscale PHATE identifies multimodal signatures of COVID*-19, Nat. Biotechnol., 40 (2022), pp. 681–691.

[22] S. R. Lay, *Convex sets and their applications*, Courier Corporation, 2007.

[23] M. Lesnick and M. Wright, *Interactive Visualization of* 2*-D Persistence Modules*, preprint, https://arxiv.org/abs/1512.00180, 2015.

[24] M. Maggioni and J. M. Murphy, *Learning by unsupervised nonlinear diffusion*, J. Mach. Learn. Res., 20 (2019), pp. 1–56.

[25] N. F. Marshall and M. J. Hirn, *Time coupled diffusion maps*, Appl. Comput. Harmon. Anal., 45 (2018), pp. 709–728.

[26] F. Mémoli, *On the use of Gromov-Hausdorff distances for shape comparison*, in Proceedings of the Eurographics Symposium on Point-Based Graphics, M. Botsch, R. Pajarola, B. Chen, and M. Zwicker, eds., The Eurographics Association, 2007, pp. 81–90.

[27] K. R. Moon, D. van Dijk, Z. Wang, S. Gigante, D. B. Burkhardt, W. S. Chen, K. Yim, A. van den Elzen, M. J. Hirn, R. R. Coifman, N. B. Ivanova, G. Wolf, and S. Krishnaswamy, *Visualizing structure and transitions in high-dimensional biological data*, Nat. Biotechnol., 37 (2019), pp. 1482–1492.

[28] M. W. Moyle, K. M. Barnes, M. Kuchroo, A. Gonopolskiy, L. H. Duncan, T. Sengupta, L. Shao, M. Guo, A. Santella, R. Christensen, A. Kumar, Y. Wu, K. R. Moon, G. Wolf, S. Krishnaswamy, Z. Bao, H. Shroff, W. A. Mohler, and D. A. Colón-Ramos, *Structural and developmental principles of neuropil assembly in C. elegans*, Nature, 591 (2021), pp. 99–104.

[29] J. M. Murphy and S. L. Polk, *A multiscale environment for learning by diffusion*, Appl. Comput. Harmon. Anal., 57 (2022), pp. 58–100, https://doi.org/10.1016/j.acha.2021.11.004.

[30] R. R. Sokal and C. D. Michener, *A statistical method for evaluating systematic relationships*, Univ. Kansas Sci. Bull., 28 (1958), pp. 1409–1438.

[31] A. D. Szlam, M. Maggioni, and R. R. Coifman, *Regularization on graphs with function-adapted diffusion processes*, J. Mach. Learn. Res., 9 (2008), pp. 1711–1739.

[32] D. Van Dijk, R. Sharma, J. Nainys, K. Yim, P. Kathail, A. J. Carr, C. Burdziak, K. R. Moon, C. L. Chaffer, D. Pattabiraman, B. Bierie, L. Mazutis, G. Wolf, and S. Krishnaswamy, *Recovering gene interactions from single-cell data using data diffusion*, Cell, 174 (2018), pp. 716–729.

[33] L. Vietoris, *Über den höheren Zusammenhang kompakter Räume und eine Klasse von zusammenhangstreuen Abbildungen*, Math. Ann., 97 (1927), pp. 454–472.

[34] U. Von Luxburg, *A tutorial on spectral clustering*, Stat. Comput., 17 (2007), pp. 395–416.

Downloaded 07/23/23 to 73.119.35.26 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.