Chapter 6
Fast-TDA Implementation for High Rate Dynamic Systems in Noisy Environment and Introduction to Chaotic System

Arman Razmarashooli, Daniel A. Salazar Martinez, Simon Laflamme, Chao Hu, Paul T. Schrader, Gurcan Comert, Negash Begashaw, and Jacob Dodson

Abstract High-rate systems (HRS), defined as engineering systems experiencing substantial disturbances/variability over short periods of time, can be significantly empowered by implementing feedback mechanisms, but these mechanisms are severely impeded by the complexity of the high-rate dynamics and tight reaction time constraints. In particular, the HRS time scale is constrained to 1 ms for whole closed-loop mechanism, including data acquisition, assessment execution, and decision-making. A compelling tool to analyze complex dynamic systems is topological data analysis (TDA). Yet, TDA requires a high computational cost and is not applicable to the HRS time scale. Here, the authors introduce the concept of *Fast-TDA*. Fast-TDA is a set of computationally efficient algorithms constructed to extract TDA-inspired features from time series for targeting HRS analytics. This study investigates methods mimicking the capabilities of persistent homology (PH) to extract the topological characteristics of one-dimensional (H_1) voids for dynamic systems embedded in two dimensions. This pseudo-PH process dynamically fits the phase space point cloud with ellipses using a peak identification method and fitting error analysis, subsequently identifying intersections between the ellipses. This approach is demonstrated on synthetic datasets embedded in two dimensions with added noise and the Lorenz attractor. Results show that Fast-TDA can accurately extract key topological features in noisy and chaotic environments over comparative state-of-the-art TDA methods.

Keywords High rate dynamic systems · Topological data analysis · Persistent homology · Chaotic systems · Lorenz attractor

---

Arman Razmarashooli · Daniel A. Salazar Martinez · Simon Laflamme
Department of Civil, Construction, and Environmental Engineering, Iowa State University, Ames, IA 50011
e-mail: shooli@iastate.edu; daniel94@iastate.edu; laflamme@iastate.edu

Simon Laflamme
Department of Electrical and Compu ter Engineering, Iowa State University, Ames, IA 50011
e-mail: laflamme@iastate.edu

Chao Hu
Department of Mechanical Engineering, University of Connecticut, Storrs, CT 06269
e-mail: chao.hu@uconn.edu

Paul T. Schrader
Air Force Research Laboratory, Information Directorate, Rome, NY 13441
e-mail: paul.schrader.1@us.af.mil

Gurcan Comert
School of Science and Engineering, Benedict College, Columbia, SC 29204
e-mail: gurcan.comert@benedict.edu

Negash Begashaw
Department of Computational Data Science and Engineering, North Carolina A&T State University, Greensboro, NC 27411
e-mail: negash.begashaw@benedict.edu

Jacob Dodson
Air Force Research Laboratory, Munitions Directorate, Eglin AFB, FL 32542
e-mail: jacob.dodson.2@us.af.mil

Thomas Matarazzo et al. (eds.), *Data Science in Engineering, Vol. 11*, Conference Proceedings
of the Society for Experimental Mechanics Series, https://doi.org/10.13052/97887-438-0156-6_6

Introduction

Topological data analysis (TDA) is a fast-growing field that leverages algebraic topology and modern mathematics to analyze the shape of data sets [1]. TDA's primary goal is finding topological features such as connected components, loops, and voids, etc., in data by constructing *simplicial complexes* from point clouds [2]. These simplicial complexes are multi-dimensional forms of *graphs* that capture the local relationships between data points [1] while simultaneously inheriting the topology of its underlying data set. TDA is particularly useful for analyzing complex, noisy datasets. Persistent homology (PH) algebraically represents how topological features (multi-dimensional holes) in a dataset change over different scales. Thus, the 0^{th} dimension of PH, or H_0, captures the connected components of a data set, the 1^{st} dimension of PH, or H_1, captures the 1-dimensional (1D) holes of a data set, the 2^{nd} dimension of PH, or H_2, captures the 2D holes or voids of a data set, etc. For instance, assuming a two-dimensional point cloud, a circle of diameter $\epsilon > 0$ is drawn around each point and the diameter of the circles dilates for all time. As the diameters grow, points connect when the edges of two circles overlap [3]. The connected points may eventually form loops or one-dimensional holes (birth), which eventually close (death). This process, known as a *filtration* measures, tracks, and records a given data's topological features lifespans as ordered pairs $(birth, death) = (b, d)$ in high fidelity, and the difference between these times, known as *persistence*, helps us to understand the stability of the features within the data. the results are often visualized in a persistence diagram, which provides insight into the dataset's overall structure, such as periodicity and clustering [3].

In many real applications, data that are collected from systems are time-dependent and represented as time series $\mathbf{x}(t) = [x_1, x_2, \cdots, x_m]$. To apply TDA to time series, it must first be converted into a point cloud. This conversion is typically done using Taken's embedding theorem, which reconstructs the state space of a system from its time series data [4, 5]. By creating a delay vector with an optimally chosen dimension d and time delay τ $\boldsymbol{\chi}(t) = [\mathbf{x}(t), \mathbf{x}(t-\tau), \mathbf{x}(t-2\tau), \cdots, \mathbf{x}(t-(d-1)\tau)]$, one can represent the current state of the system and its past values, createing a point cloud embedded in finite dimensional Euclidean space that preserves the underlying dynamics of the original system topologically equivalent manifold of the same dimension [6, 7]. The embedding dimension d (number of past values to include) and time delay τ (the gap between the successive values) are key parameters typically are chosen by false nearest neighbor test and mutual information methodologies [5]. Once this transformation is complete, TDA can be applied to the embedded data to extract topological invariants. TDA captures key system features by using Taken's embedding theorem. Persistent homology and simplicial complexes are the main tools to find these features. Previous studies have demonstrated that combining TDA features with machine learning algorithms is effective for solving classification problems in various fields, including electrocardiogram classification, time series analysis, and the exploration of large datasets in biology and chemistry [8, 9].

The application domain of this paper is high-rate systems (HRS), defined as dynamic systems experiencing accelerations greater than 100 g_n for durations under 100 milliseconds, such as those found in blast mitigation, advanced weaponry, and hypersonic vehicles [10]. HRS could greatly benefit from feedback systems to ensure optimal operations, but the complexity of their dynamics combined with the tight time constraint complicates the integration of any closed-loop mechanism. This paper investigates the applicability of TDA for real-time state estimation of HRS. TDA features have shown promise for state estimation in such systems, particularly when used as inputs to Gated Neural Networks [11]. However, the computational burden of forming simplicial complexes and persistent diagrams limits TDA's real-time applicability, especially in systems with high sampling rates and chaotic environments. Any closed-loop mechanisms applicable to HRS must function under 1 ms [12]. As a result, the scalability of TDA scalability in high-rate systems becomes a significant obstacle [13]. To address these challenges, we study the concept of *Fast-TDA*, a method inspired by traditional TDA but optimized for efficiency. Fast-TDA avoids the computational cost of constructing simplicial complexes by using geometric methods to extract one-dimensional homology features (H_1) through inscribed ellipses and intersections. This approach not only reduces computational time but also enhances the robustness of feature extraction, particularly in noisy environments.

The rest of the paper is organized as follows. The next section describes the Fast-TDA algorithm to estimate H_1. Following that, we present the synthetic and laboratory datasets used in validating the algorithm. The subsequent section discusses the results, and the final section provides the conclusion.

Algorithm

Our approach uses TDA to determine the birth and death of topological features without relying on constructing a simplicial complex filtration. We focus on analyzing two-dimensional (2D) point clouds. In Fast-TDA, we estimate the persistence of the feature H_1 by first calculating the average distance between neighboring points to determine the birth time. Then, we fit an ellipse through the point cloud, the length of the ellipse's whose minor axis corresponds to the death time of H_1. Various techniques exist to fit one ellipse through a set of points; we adopt the least squares method for ellipse fitting. To account for the nonstationarity in our system, we used a sliding window to extract local features, assuming the system remains stationary within each window [11]. The window is defined to include one period of the system $w_1 = 1/f_{min} + \tau$ where f_{min} represents the system's minimum frequency, ensuring the point cloud forms a loop that corresponds to H_1. However, the problem becomes more complicated when there is more than one frequency in the system since the point cloud should be fitted by multiple ellipses. We investigate dynamic ellipse identification extracted by using a second sliding window nested within the first window. As the first sliding window moves along the time series, the second sliding window with the size of $w_2 = 1/f_{max} + \tau$ where f_{max} represents the system's maximum frequency, captures all inner ellipses. When the fitted error for the ellipses reaches a local minimum, it has found a fitted ellipse. Once the ellipses are fitted/captured in both cases. The intersection of pairs of ellipses are calculated by simultaneously solving their parametric equations.

The objective is to find the set of points where both ellipses overlap, which involves determining if the boundary of one ellipse lies inside the other. In cases where ellipses do not directly intersect but overlap geometrically, the space between the outer boundary of the smaller ellipse and the inner boundary of the larger one is also included as part of the overlap. The regions are combined into a unified composite shape that represents all areas where multiple ellipses overlap. This is achieved by taking the union of the previously identified intersection areas with any new ellipses introduced into the system. The process carefully retains only the portions of the ellipses that are common between them, ensuring that the composite region accurately reflects the overlapping areas. By continuously updating this composite region as new ellipses are added, the system maintains a comprehensive map of all the overlaps between ellipses within the time series. To analyze the properties of the overlapping regions, the largest inscribed circle is computed for each intersection. This is done by evaluating the distance from points inside the region to the boundary and identifying the point that maximizes this distance. The radius of the largest inscribed circle is determined by the distance between this central point and the nearest boundary, ensuring the circle fits entirely within the region without extending beyond it. To optimize this process, a *Voronoi diagram* is employed, where each Voronoi vertex provides a candidate for the circle's center [14]. The vertex with the maximum distance to the boundary is selected, and the corresponding circle is inscribed within the region, revealing the largest possible enclosed area within the overlapping ellipses.

Datasets

To investigate the efficiency and sensitivity of the algorithm in noisy and chaotic environments, three datasets are selected.

Dropbear: The DROPBEAR testbed was developed to validate state estimation algorithms for HRS. As shown in Figure 1, this testbed consists of a cantilever beam subjected to a fast-moving boundary condition using a moving roller [15]. The moving roller has been used to model a sudden change in the beam's stiffness. DROPBEAR can be simplified as a cantilever beam of variable length. The beam typically experiences free vibration and it can be simplified to [11].

$$x(t) = A\cos(wt) + n_p = A\cos(2\pi f t) + n_p \tag{1}$$

where $x(t)$ is the vertical acceleration at the tip, A is the amplitude, f is the natural cyclic frequency in Hertz (Hz), t is time in seconds, and n_p is noise. In the context of free vibration, the fundamental frequency of the beam can be estimated by using Equation 2 [16]:

$$f = \frac{E}{L^2} \tag{2}$$

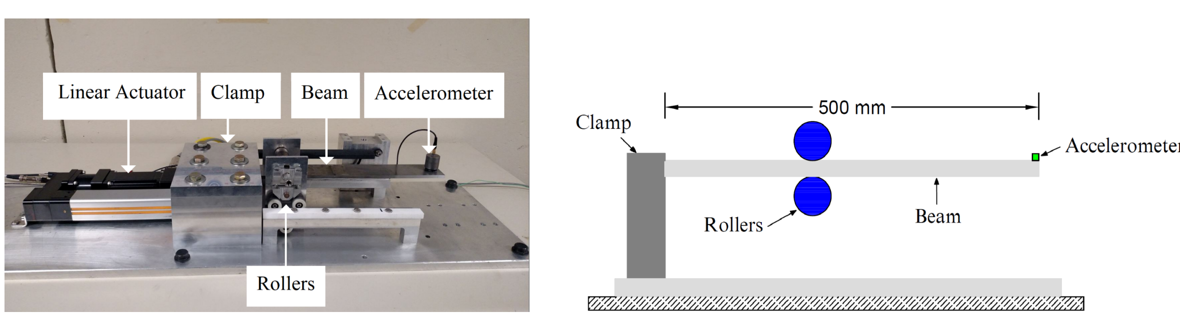

Fig. 1 Picture of the DROPBEAR testbed [11]

where E is a constant and L is length of the beam. Literature has shown that the ideal embedding dimension of a single harmonic signal is 2 [17]. This results in a 2-dimensional point cloud, from which a persistence diagram can be generated, containing information about connected components or zero-dimensional holes, known as the TDA persistent homology (PH) feature representation H_0, and loops or one-dimensional holes, known as TDA PH feature representation H_1.

Chaotic system: To explore chaotic behavior, we use the Lorenz system, a well-known set of ordinary differential equations (ODEs) that exhibits chaos under specific conditions [6]. The system is defined by the following equations:

$$\frac{dx}{dt} = \sigma(y - x) \tag{3}$$

$$\frac{dy}{dt} = x(\rho - z) - y \tag{4}$$

$$\frac{dz}{dt} = xy - \beta z \tag{5}$$

Here, σ, ρ, and β are parameters that influence the system's dynamics. For this study, we set $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$ to induce chaotic behavior. The system is solved numerically, starting from the initial condition $X_0 = [0, 1, 1.05]$, which sets the values for $x(t)$, $y(t)$, and $z(t)$ at $t = 0$. The integration is performed over a time span with 4000 points to accurately capture the system's dynamics. The result is a numerical solution that reflects the chaotic nature of the Lorenz system, providing a time series for $x(t)$ (as shown in Figure 3, and 4).

After obtaining the numerical solution, Taken's embedding theorem is applied to reconstruct the phase space in two dimensions using a time delay of $\tau = 0.1$ seconds [18]. This approach helps in properly unfolding the system's dynamics, effectively capturing the characteristic 'butterfly effect' of the Lorenz attractor.

Synthetic dual harmonic signal: Due to the unavailability of a realistic laboratory dataset, we consider the dual harmonic signal with noise, which is expressed as follows:

$$x(t) = \cos(2\pi f_1 t) + \cos(2\pi f_2 t) + n_p \tag{6}$$

where f_1 and f_2 represents the system's frequencies, with $f_1 > f_2$, and n_p represents noise. The system is embedded using a time delay of $\tau = 0.01$. To ensure that the point cloud consists of multiple ellipses, the time delay must be less than $\tau < 0.25/f_2$. The point cloud is plotted using arbitrary values of f_1 = 3.65 Hz and f_2 = 1.9 Hz to increase the complexity.

Results

In this section, we investigate how well the minor axis in the fitted ellipse can be utilized to find the right cart location compared to maximum persistence and short time Fourier transform (STFT). Data obtained from Fast-TDA, TDA, and STFT are post-processed to map results to the cart location by assuming a quadratic relationship in Equation 2. We conduct the following linear regression.

$$x^2 = a_0 + a_1 F_i \tag{7}$$

where x is the cart location in millimeters (mm), a_0 and a_1 are regression parameters estimated using a least squares estimator, and F_i is the feature obtained from Fast-TDA (the minor axis), TDA (the maximum persistence of H_1), and STFT (the frequency of the system). Table 1 lists regression parameters for various features. Results indicate that Fast-TDA provides a better estimation of the cart location relative to TDA based on the value of R^2. The DROPBEAR test corresponds to Dataset 2 [19]. The dataset contains three distinct cart movement patterns, each with different velocities and accelerations. For better comparison, the results were divided into three sections: Section 1 covers the cart movement from 0.7 seconds to 17.8 seconds, Section 2 from 17.8 seconds to 32.5 seconds, and Section 3 from 32.5 seconds to 41.8 seconds. Data is embedded using $\tau = 0.0008$ seconds, and the size of the moving window is $w = 0.025$ seconds.

Table 1 Results from LSE

Method	a_0	a_1	R^2
TDA	-0.92	2.47	0.87
STFT	-0.034	0.93	0.95
Fast-TDA	0.46	2.16	0.91

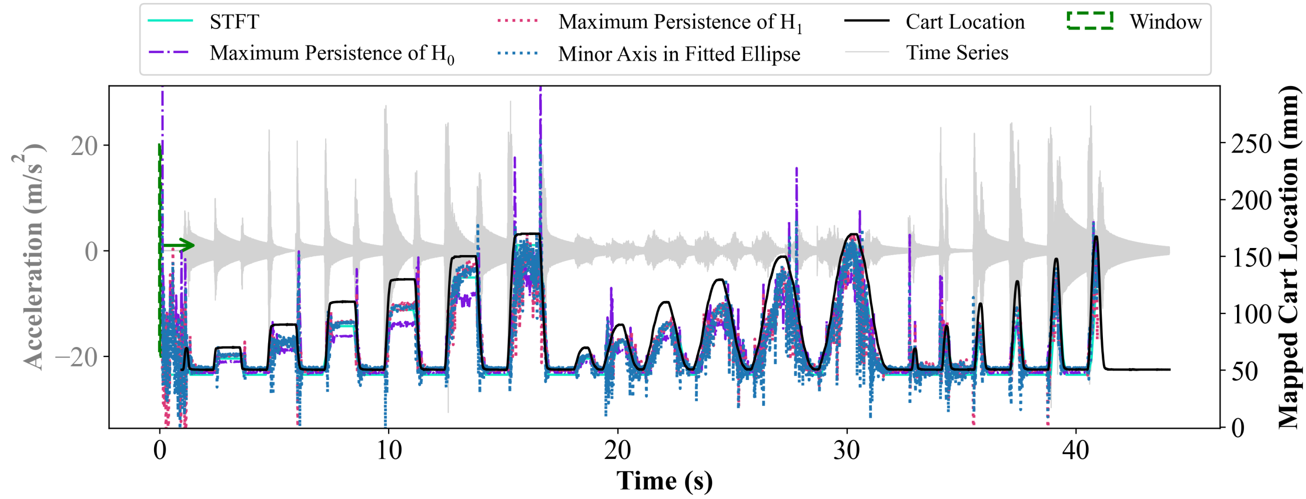

Fig. 2 Cart localization results on DROPBEAR using various features

We evaluate performance through three performance metrics (J_1, J_2, and J_3) [11]. Metric J_1 is the mean absolute error between the actual cart location and the mapped cart location for each section of the excitation. Metric $J_{2,i}$ is the number of instances over the signal that the estimation does not remain below a given distance i (in mm, here taken as $i = 5, 10,$ and 20 mm, with 20 mm arbitrarily taken as an acceptable target). Metric $J_{3,i}$ is the mean absolute error of the cart location when the estimated cart position is within the given distance i (in mm). Figure 2 plots the time series results of the car localization using various features. Table 2 list results in terms of performance metrics A study of the metrics shows that, in terms of J_1, Fast-TDA performs similar to TDA in section 3, but outperform TDA in the other sections. However, STFT outperforms both Fast-TDA and TDA in sections 1 and 2. Metric J_2 demonstrates that Fast-TDA accurately maps within 20 mm of the correct cart location over 90% of the time, outperforming TDA in all cases. It also performs similarly to the SFTF or even better, except notably under J_3.

We analyze the performance of windowing as a pre-defined hyperparameter in both TDA and Fast-TDA for chaotic behavior, as the correct window size in such systems is generally unknown. The study is conducted using four different window sizes. Figures 3 and 4 show the time series for $x(t)$ and the evolution of various window sizes for TDA and Fast-TDA. It can be observed that a window size of 1.5 seconds produces the most stable results, with performance deteriorating as the window size decreases, becoming unstable at 0.5 seconds. In contrast, results in Figure 4 indicate that Fast-TDA performs similarly for window sizes of 1 and 1.5 seconds. Although there is some instability with the smaller window sizes, Fast-TDA maintains the trend with increased noise. The results suggest that Fast-TDA is less sensitive to window size compared to TDA.

Lastly, we examine the sensitivity of Fast-TDA and TDA results in noisy environments. Results for both noise-free and noisy dual harmonic signals are plotted in Figures 5 and 6. Figure 5 (a) and 6 (a) shows the time series and sliding windows,

Table 2 Performance results for cart localization on DROPBEAR.

Section	Method	J_1 (mm)	$J_{2,5}$ (%)	$J_{2,10}$ (%)	$J_{2,20}$ (%)	$J_{3,5}$ (mm)	$J_{3,10}$ (mm)	$J_{3,20}$ (mm)
Section 1	TDA	8.7	46	29	12	1.4	2.9	4.9
	STFT	6.3	38	13	3	3.4	4.6	5.4
	Fast-TDA	8.3	42	30	10	1.9	4.6	5.0
Section 2	TDA	10.2	56	34	15	1.9	3.8	6.0
	STFT	7.3	50	23	6	3.5	4.6	6.4
	Fast-TDA	9.2	54	31	11	2.1	3.9	6.2
Section 3	TDA	5.7	30	18	8	1.3	2.1	3.4
	STFT	5.9	21	12	5	4	4.4	5.1
	Fast-TDA	5.7	28	17	7	1.9	2.6	3.8

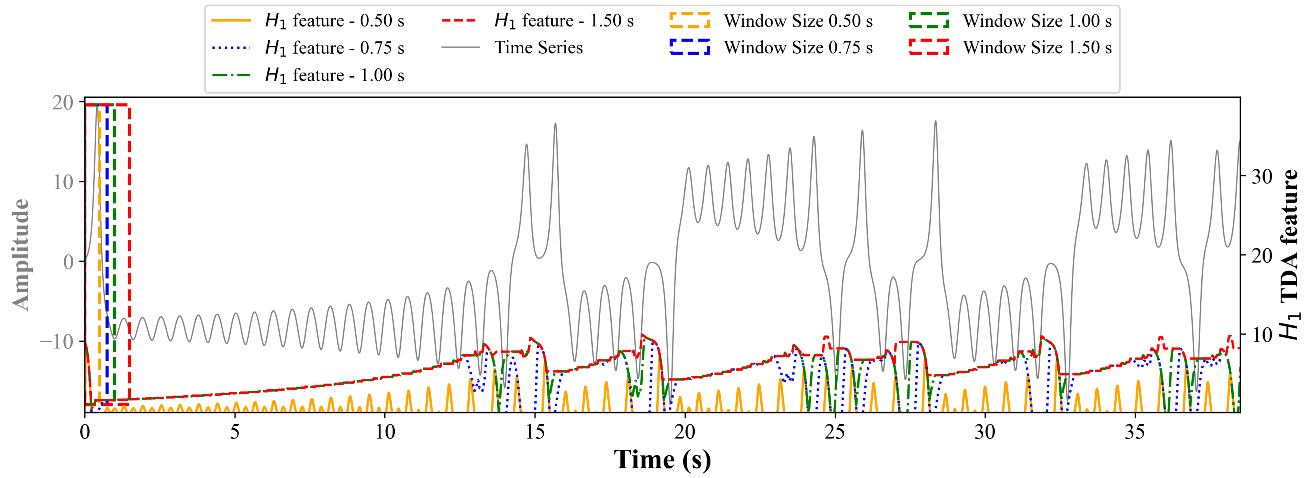

Fig. 3 TDA feature evolution on Lorenz system under various window sizes.

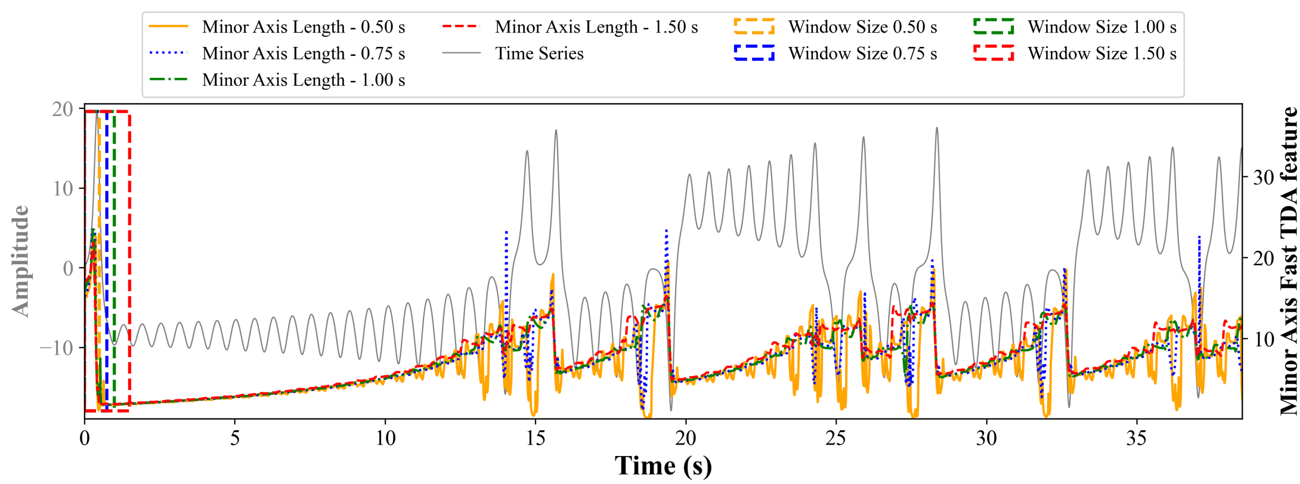

Fig. 4 Fast-TDA feature evolution on Lorenz system under various window sizes.

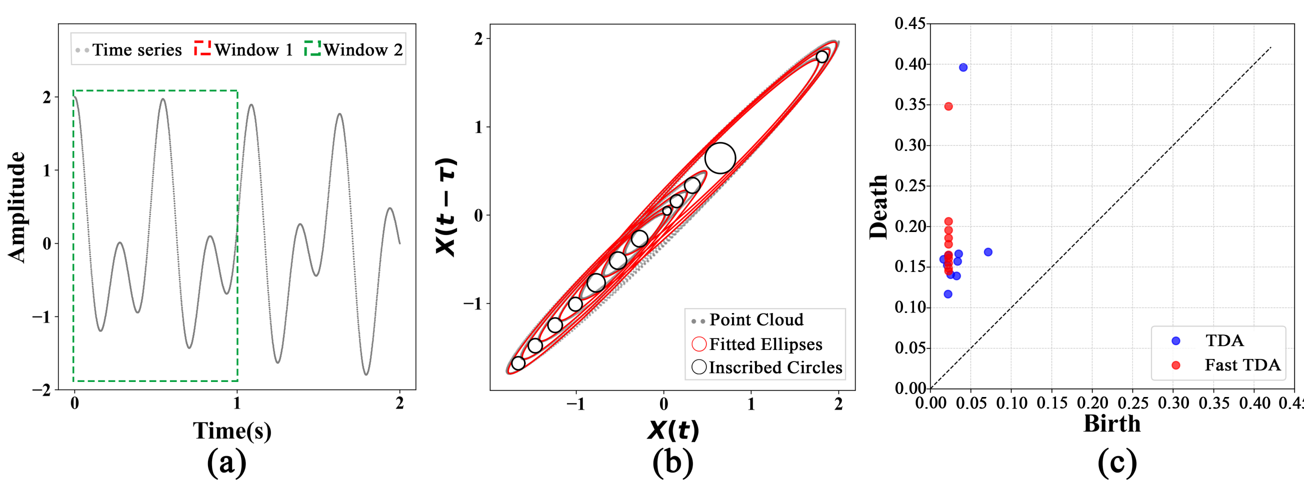

Fig. 5 TDA and Fast-TDA feature comparison for noise-free dual harmonic signal.

Figure 5 (b) and 6 (b) shows the fitted ellipses and inscribed circles, and Figure 5 (c) and 6 (c) compares the H_1 persistence diagram for the H_1 feature from TDA and Fast-TDA. For better comparison, the birth and death times for the first 10 points with longer persistence are listed in Tables 3 and 4. The results indicate that adding noise to the system causes a 60% change in the maximum persistence of H_1 from TDA but only a 1% change in the maximum persistence value from Fast-TDA. For points with shorter persistence, the changes are still significant in TDA compared to Fast-TDA. The number of off-diagonal points increases from 13 to 25 in TDA when moving from a noise-free to a noisy environment, whereas in Fast-TDA the number of off-diagonal points decreases from 11 to 10. Results show that adding noise to the system has a much smaller impact on the features extracted by Fast-TDA compared to TDA.

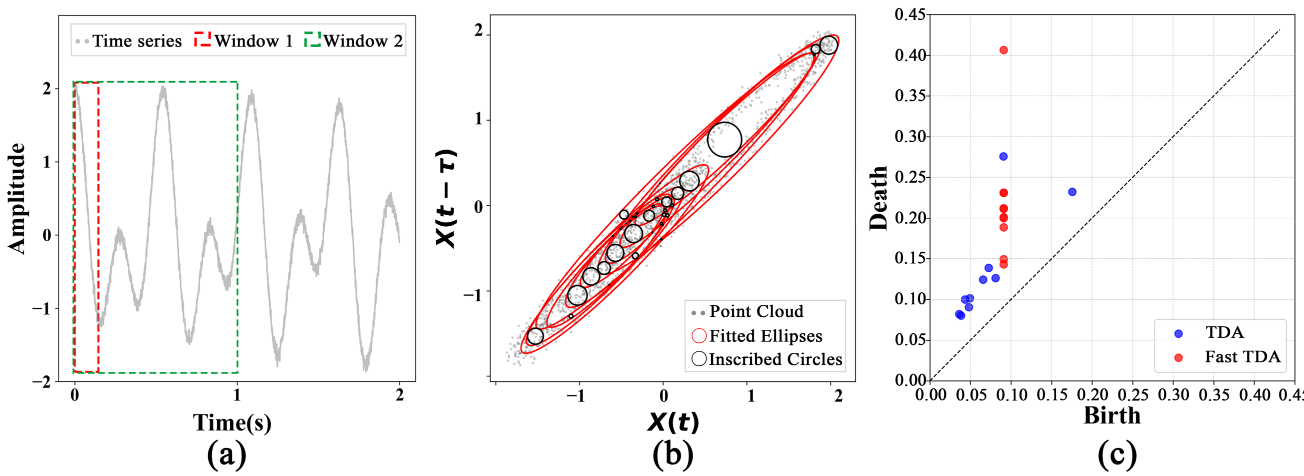

Fig. 6 TDA and Fast-TDA feature comparison in noisy dual harmonic signal.

Table 3 Birth and death values of the first 10 points using TDA and Fast-TDA for the noise-free dual harmonic signal

Method	Feature	Point 1	Point 2	Point 3	Point 4	Point 5	Point 6	Point 7	Point 8	Point 9	Point 10
TDA	Birth	0.040	0.016	0.022	0.020	0.035	0.033	0.024	0.032	0.071	0.021
(H_1)	Death	0.396	0.159	0.164	0.152	0.166	0.157	0.140	0.139	0.168	0.116
Fast-TDA	Birth	0.022	0.022	0.022	0.022	0.022	0.022	0.022	0.022	0.022	0.022
(Diameter)	Death	0.348	0.206	0.195	0.185	0.178	0.165	0.161	0.154	0.148	0.145

Table 4 Birth and death values of the first 10 points using TDA and Fast-TDA for the noisy dual harmonic signal.

Method	Feature	Point 1	Point 2	Point 3	Point 4	Point 5	Point 6	Point 7	Point 8	Point 9	Point 10
TDA	Birth	0.090	0.072	0.065	0.175	0.043	0.049	0.036	0.080	0.047	0.038
(H_1)	Death	0.275	0.138	0.124	0.232	0.099	0.101	0.082	0.126	0.090	0.080
Fast-TDA	Birth	0.090	0.090	0.090	0.090	0.090	0.090	0.090	0.090	0.090	0.090
(Diameter)	Death	0.406	0.231	0.230	0.212	0.211	0.201	0.199	0.188	0.149	0.143

Conclusion

In this paper, we propose the concept of *Fast-TDA* for noisy and chaotic signals, which estimates TDA features directly from a point cloud of data without necessitating the computation of filtrations or persistence diagrams. The development of the Fast-TDA approach was motivated by the deployment of TDA at the sub-millisecond realm, targeting applications to high-rate systems. The results showed that Fast-TDA could be a powerful method to extract topological features inspired by TDA, significantly improving computing time outperforming traditional TDA in a noisy and chaotic environment while being less sensitive to pre-TDA known hyperparameters (window size).

Acknowledgments The authors would like to acknowledge the financial support from the Defense Established Program to Stimulate Competitive Research (DEPSCoR) award number FA9550-22-1-0303, the Air Force Office of Scientific Research (AFOSR) award number FA9550-23-1-0033, and the National Science Foundation awards numbers CCF-1937460 and CCF-2234919. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors/organizations, the United States Air Force, or the US government.

Disclosure of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Gowdridge, T., Dervilis, N., and Worden, K. “On topological data analysis for structural dynamics: an introduction to persistent homology”. *ASME Open Journal of Engineering*, 1:011038 (2022)
2. Carlsson, G. “Topology and data”. *Bulletin of the American Mathematical Society*, 46(2):255–308 (2009)
3. Carlsson, G. and Vejdemo-Johansson, M. *References*. Cambridge University Press (2021)
4. Takens, F. “Detecting strange attractors in turbulence”. In *Lecture Notes in Mathematics*, pages 366–381. Springer Berlin Heidelberg (1981)
5. Alexander, Z. *A topology-based approach for nonlinear time series with applications in computer performance analysis*. PhD thesis, University of Colorado at Boulder (2012)
6. Kantz, H. and Schreiber, T. *Nonlinear Time Series Analysis*. Cambridge University Press (2003)
7. Stark, J. “Delay embeddings for forced systems. i. deterministic forcing”. *Journal of Nonlinear Science*, 9:255–332 (1999)
8. Dlugas, H. “Electrocardiogram feature extraction and interval measurements using optimal representative cycles from persistent homology”. *bioRxiv*, pages 2022–02 (2022)
9. Umeda, Y. “Time series classification via topological data analysis”. *Information and Media Technologies*, 12:228–239 (2017)
10. Nelson, M., Barzegar, V., Laflamme, S., Hu, C., Downey, A.R., Bakos, J.D., Thelen, A., and Dodson, J. “Multi-step ahead state estimation with hybrid algorithm for high-rate dynamic systems”. *Mechanical Systems and Signal Processing*, 182:109536 (2023)
11. Razmarashooli, A., Martinez, D.A.S., Chua, Y.K., Laflamme, S., and Hu, C. “Real-time state estimation using recurrent neural network and topological data analysis”. In *Nondestructive Characterization and Monitoring of Advanced Materials, Aerospace, Civil Infrastructure, and Transportation XVIII*, volume 12950, pages 106–114. SPIE (2024)
12. Dodson, J., Downey, A., Laflamme, S., Todd, M.D., Moura, A.G., Wang, Y., Mao, Z., Avitabile, P., and Blasch, E. “High-rate structural health monitoring and prognostics: An overview”. In *Data Science in Engineering, Volume 9: Proceedings of the 39th IMAC, A Conference and Exposition on Structural Dynamics 2021*, pages 213–217. Springer (2022)
13. Schrader, P.T. “Topological multimodal sensor data analytics for target recognition and information exploitation in contested environments”. In *Signal Processing, Sensor/Information Fusion, and Target Recognition XXXII*, volume 12547, pages 114–143. SPIE (2023)
14. Aurenhammer, F. “Voronoi diagrams—a survey of a fundamental geometric data structure”. *ACM Computing Surveys (CSUR)*, 23(3):345–405 (1991)
15. Nelson, M., Laflamme, S., Hu, C., Moura, A.G., Hong, J., Downey, A., Lander, P., Wang, Y., Blasch, E., and Dodson, J. “Generated datasets from dynamic reproduction of projectiles in ballistic environments for advanced research (DROPBEAR) testbed”. *IOP SciNotes*, 3(4):044401 (2022)
16. Kausel, E. *Advanced Structural Dynamics*. Cambridge University Press (2017)
17. Small, M. *Applied nonlinear time series analysis: applications in physics, physiology and finance*, volume 52. World Scientific (2005)
18. Buzug, T. and Pfister, G. “Optimal delay time and embedding dimension for delay-time coordinates by analysis of the global static and local dynamical behavior of strange attractors”. *Phys. Rev. A*, 45:7073–7084 (1992)
19. Downey, A., Hong, J., Dodson, J., Carroll, M., and Scheppegrell, J. “Dataset-2-dropbear-acceleration-vs-roller-displacement” (2020)