Chatter Classification in Turning using Machine Learning and Topological Data Analysis [⋆]

Firas A. Khasawneh [*] **Elizabeth Munch** [†] **Jose A. Perea** [‡]

[*] *Dept. of Mechanical Engineering (e-mail: khasawn3@egr.msu.edu)*
[†,‡] *Dept. of Computational Mathematics, Science, and Engineering;*
Dept. of Mathematics
(e-mails: [†] muncheli@egr.msu.edu; [‡] joperea@msu.edu)
Michigan State University, East Lansing, MI 48824 USA

Abstract: Chatter identification and detection in machining processes has been an active area of research in the past two decades. Part of the challenge in studying chatter is that machining equations that describe its occurrence are often nonlinear delay differential equations. The majority of the available tools for chatter identification rely on defining a metric that captures the characteristics of chatter, and a threshold that signals its occurrence. The difficulty in choosing these parameters can be somewhat alleviated by utilizing machine learning techniques. However, even with a successful classification algorithm, the transferability of typical machine learning methods from one data set to another remains very limited. In this paper we combine supervised machine learning with Topological Data Analysis (TDA) to obtain a descriptor of the process which can detect chatter. The features we use are derived from the persistence diagram of an attractor reconstructed from the time series via Takens embedding. We test the approach using deterministic and stochastic turning models, where the stochasticity is introduced via the cutting coefficient term. Our results show a 97% successful classification rate on the deterministic model labeled by the stability diagram obtained using the spectral element method. The features gleaned from the deterministic model are then utilized for characterization of chatter in a stochastic turning model where there are very limited analysis methods.

Keywords: chatter, machine learning, machining, time delay systems, stability, stochastic equations, topological data analysis, transfer learning, turning

1. INTRODUCTION

Cutting processes such as turning and milling, which are often described using delay differential equations, constitute a major part of discrete manufacturing. One of the most intriguing problems that face the industry in these processes is chatter. Chatter is characterized by large amplitude oscillations that have detrimental effects on the workpiece quality, cutting tool, and the spindle. Therefore, a large body of research has emerged focused on predicting and modeling chatter.

In addition to predictive models for chatter, prior works have studied in-situ methods for chatter detection (Smith and Tlusty (1992); Altintas and Chan (1992); Choi and Shin (2003); Kuljanic et al. (2009); Yao et al. (2010); Elias and Narayanan Namboothiri (2014)). These methods are often combined with active control strategies to suppress chatter (Faassen (2007); van Dijk et al. (2010)). In-process chatter detection methods attempt to alleviate the process modeling difficulties by investigating the cutting signal itself. Currently available in-process methods for studying chatter typically rely on comparing the characteristics of the acoustic, vibration, or force signals against certain predefined features indicative of chatter (Tlusty and Andrews (1983); Delio et al. (1992); Gradisek et al. (1998); Schmitz et al. (2002); Choi and Shin (2003); Bediaga et al. (2009); Sims (2009); Nair et al. (2010); van Dijk et al. (2010); Tsai et al. (2010); Kakinuma et al. (2011); Ma et al. (2013)). Often a metric or an index and a threshold are defined on the time series and chatter is detected if this threshold is exceeded. However, online chatter detection is associated with another set of challenges. For example, the value of an adequate threshold is difficult to identify and the majority of tools used to define the necessary metrics on the data are linear despite the inherently nonlinear nature of machining processes. Further, each specific case of cutting conditions would require performing preliminary cutting tests to train the used algorithms (Hino et al. (2006)). Another difficulty is the need for accurately identifying threshold values to compare certain features in the signal against those associated with chatter (Choi and Shin (2003)). Even if a suitable threshold is found, many online chatter detection methods indicate the occurrence of chatter only after it has fully developed, hence damaging the workpiece or the tool (Faassen (2007)). Chatter-detection using currently avai-

[⋆] This material is based upon work supported by the National Science Foundation under Grant Nos. CMMI-1759823 and DMS-1759824 with PI FAK, and CMMI-1800466 and DMS-1800446 with PI EM. JAP acknowledges the support of the NSF under grant DMS-1622301 and DARPA under grant HR0011-16-2-003.

lable classification and machine learning methods requires a training set for each setup of the process. Consequently, if data is obtained for the same type of cutting process but with a different configuration, a new training set is generally required.

In contrast, this paper seeks to apply learning on features of a topological descriptor of the signal, rather than to features of the signal itself, thus basing the learning on intrinsic process characteristics which can be generalized and adapted to different situations. The tools we use come from Topological Data Analysis (TDA) and nonlinear time series analysis. Specifically, we embed the time series as a point cloud using tools from time series analysis, and then represent the data using topological features extracted from the shape of the resulting point cloud. The features are extracted from persistence diagrams (a tool from TDA) which provide a summary of the topological features in the point cloud. We use the described approach to obtain a classification for chatter in a deterministic turning model, then apply the trained algorithm to a stochastic turning model. The method's accuracy in classifying the chatter/chatter-free cuts in the deterministic case is determined using the stability calculations of the spectral element approach (Khasawneh and Mann, 2011). For the stochastic model, we qualitatively comment on the resulting classification in light of the expected behavior of stochastic systems since the tools for the stability analysis of time-periodic, stochastic delay differential equations are slim or non-existent. This inductive learning is especially useful when it is desirable to train the signal on an abstract system model where the ground truth is known, and then hope to transfer the acquired knowledge to a more complicated system where it is more difficult to classify the resulting behavior.

This work expands upon the model and the framework described in Khasawneh and Munch (2016) by providing a machine learning approach for training a classifier that provides insight about the underlying dynamics and enables a decision making mechanism when other available tools fall short.

2. MECHANICAL MODEL

Figure 1 shows the turning process that will be investigated in this study. The tool is modeled as a linear oscillator with a single degree of freedom in the y direction while the workpiece is assumed to be rigid. The cutting force F acts at the tool tip and the resulting equation of motion reads

$$\ddot{y} + 2\zeta\omega_n\dot{y} + \omega_n^2 y = \frac{F}{m} = \frac{Kwh^\alpha}{m} \tag{1}$$

where ζ is the damping ratio, ω_n is the natural angular frequency of the tool, and m is the modal mass; K is the mechanistic cutting coefficient with units of $\mathrm{N/m^2}$ which relates the cutting force to the area of the uncut chip $A = w \times h$, w is the depth of cut shown in Fig. 1, h is the uncut chip thickness, and the exponent is typically chosen as $\alpha = 0.75$ (Tlusty (2000); Stepan (2001)).

The tool will oscillate under the influence of the cutting force leaving behind an undulated surface on the workpiece. When the tool then advances into the workpiece during the following revolution at the nominal feedrate h_0 meter per spindle revolution (m/rev), it will encounter the wavy surface from the previous revolution. This dynamic variation of the dynamic chip thickness is thus influenced by both the current and the previous tool oscillations represented by $y(t)$ and $y(t-\tau)$, respectively. Here $\tau = 2\pi/\Omega$ is the duration of one spindle revolution corresponding to the spindle speed Ω rad/sec. The coupling between the cutting forces and the tool oscillations can lead to large amplitude, self-regenerative vibrations referred to as chatter. If the tool oscillations are severe, then the tool will leave the surface of the workpiece and experience damped oscillations, i.e., the right hand side of Eq. (1) becomes zero.

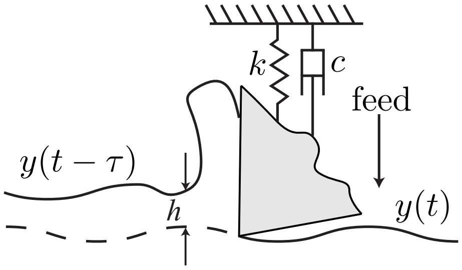

Fig. 1. The turning model described in Sec. 2.

To reduce the number of parameters in Eq. (1), we use the rescaling described in Insperger et al. (2008): let $y(t) = h_0\tilde{y}(t)$ and let $\tilde{t} = \omega_n t$, and $\tilde{\tau} = \omega_n\tau$. After dropping the tildes we obtain

$$\begin{aligned}\ddot{y} + 2\zeta\dot{y} + y &= \frac{Kw(2\pi R)^{\alpha-1}}{m\omega_n^2}\rho^{\alpha-1}(1 + y(t-\tau) - y(t))^\alpha \\ &= b\rho^{\alpha-1}(1 + y(t-\tau) - y(t))^\alpha \end{aligned}\tag{2}$$

where R is the radius of the workpiece, $\rho = h_0/(2\pi R)$, b is the dimensionless depth of cut, and $\rho < 0.01$. Equation (2) is used in the simulation to generate time series for the deterministic system.

We can capture the effect of the variations in the workpiece material, shear angle, or temperature effects, by introducing the stochastic non-dimensional cutting coefficient $\hat{b}$. If we allow the dimensionless cutting coefficient to become a stochastic variable

$$\hat{b} = \bar{b} + \delta\frac{dB}{dt} \tag{3}$$

where $\bar{b}$ is the average or nominal value of the stochastic non-dimensional cutting coefficient $\hat{b}$, δ is the diffusion coefficient, and B is standard Brownian motion. The coefficient δ represents a measure of the stochasiticity in the system. The resulting stochastic delay differential equation for the tool oscillations then reads

$$\begin{aligned}d\dot{Y} = &\left(-2\zeta\dot{Y} - Y + \bar{b}\rho^{\alpha-1}(1 + Y(t-\tau) - Y(t))^\alpha\right)dt \\ &+ \delta\left(\rho^{\alpha-1}(1 + Y(t-\tau) - Y(t))^\alpha\right)dB\end{aligned}\tag{4}$$

where Eq. (4) is interpreted in the Itô sense (Oksendal, 2007). We use Eq. (4) in this paper to generate the noisy time series.

3. TOPOLOGICAL SIGNAL PROCESSING

In order to automate the classification of the time series arising from our model, we turn to the new field of topological signal processing, which combines methods

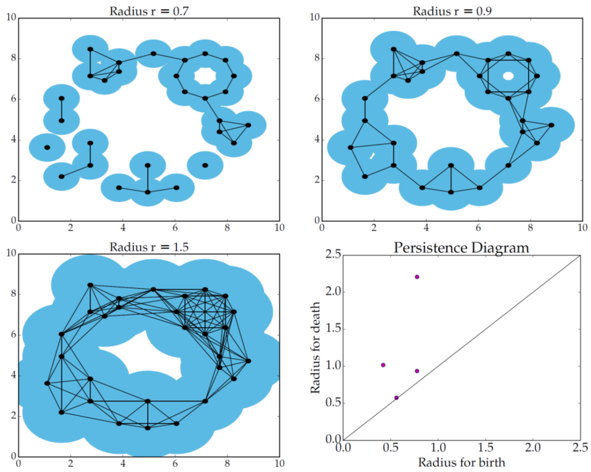

Fig. 2. A point cloud shown with collections of blue disks for several different radii (top left, top right and bottom left); the persistence diagram for this point cloud is given in the bottom right. The circular structure of the point cloud is reflected in the single point far from the diagonal in the persistence diagram.

from Topological Data Analysis (TDA) with tools from time series analysis to analyze streams of data.

3.1 Persistent Homology

We begin with an informal introduction to the subject of persistent homology, and direct the reader to Munch (2017) or Edelsbrunner and Harer (2010) for a more in depth discussion. Suppose we are given a point cloud drawn from a manifold and want to understand something about the underlying structure. To do this, we consider expanding a collection of discs of the same radius centered at each point. We can then study the structure of the union of these discs for a changing radius. In the example of Fig. 2, we start with a point cloud sampled from an annulus and see that at a very small radius (around $r = 0.3$), the collection of blue disks consists of a set of disconnected components. At a slightly larger radius ($r = 0.7$), these discs start to intersect, possibly forming small circular structures which fill in at a still slightly larger radius. What is very interesting is that at a relatively small radius (approximately $r = 0.9$), we form a circular structure representing the full annulus which takes a much longer time to fill in than the small circular structures seen previously. The goal of persistent homology is to quantify this intuition in a rigorous manner and use it to "measure" the size of the annulus.

A hole is called a "class" in the homology. While homology gives information about a static topological space, persistent homology quantifies the changing homology of a changing topological space. In particular, persistent homology measures how long each class persists in the changing space. Having a class with a long life quantifies our intuition that the point cloud appears to have circular structure.

To make this mathematical construction computable, we approximate the structure of the union of disks by a simplicial complex. A simplicial complex is a topological space built up from discrete, combinatorial building blocks. It can be thought of as a generalization of a graph since the lowest dimensional building blocks are vertices and edges; however, it can also contain higher dimensional simplices such as triangles, tetrahedra, and their higher dimensional analogues. The edge set of this simplicial complex is drawn in Fig. 2 as black lines between the points; we assume that higher dimensional simplices are included whenever all their lower dimensional faces appear. By representing the changing space as a combinatorial structure, we have the ability to store its data in a computer as well as to compute its persistent homology and keep track of the appearance and disappearance of classes. The persistence diagram (i.e. bottom right of Fig. 2) plots the lifetime of each of these classes by placing a point for each class at (b, d) if the class appeared (was born) at radius b and was filled in (died) at radius d. In the example of Fig. 2, small loops that appear and disappear quickly in the set of expanding discs, correspond to points close to the diagonal in the persistence diagram. The large circular structure appears as a point far from the diagonal. In this way, we have a visual representation of the difference between classes which live a long time (and could be considered important), and the points representing short lived classes (and are often assumed to be noise). For this reason, we usually draw the diagonal $\{(x, x) \mid x \geq 0\}$ into the persistence diagram along with the off-diagonal points to remind the observer that points always appear above the diagonal, and that proximity to the diagonal implies that an off-diagonal point is more likely to be noise. While this example shows that our intuition is valid in the case of a 1-dimensional structure (the circle) embedded in visualizeable 2-dimensions, the persistence diagram can give information about a structure of any dimension embedded in any high dimensional space, while still remaining as a 2-dimensional output.

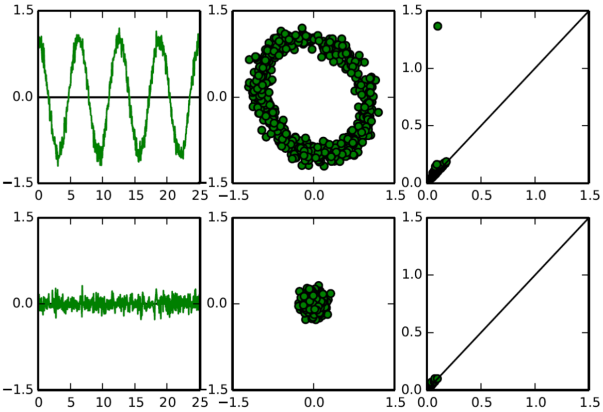

Fig. 3. An example of the relationship between periodicity and maximum persistence. The columns correspond to the signal, the Takens embedding of the signal, and the persistence diagram of the Takens embedding, respectively. Note that the periodic signal has a prominent off-diagonal point in the persistence diagram unlike the signal that only contains noise.

3.2 Delay Embedding and Persistence

The delay embedding is a standard tool for time series analysis (Kantz and Schreiber, 2004). Given a time series $X(t)$, which in practice is a set of samples $s_n = X(t_n)$, fix a delay $\eta > 0$ and choose a dimension $m \in \mathbb{Z}_{>0}$ in which to embed the data. Then the delay embedding is a lift of the time series to the map to $\mathbb{R}^m$

$$\psi_\eta^m : t \longmapsto (X(t), X(t+\eta), \cdots, X(t+(m-1)\eta)).$$

Through an important theorem of Takens (Takens, 1981), we are justified in using the term "embedding" since, under the correct parameter choices, this mapping preserves the structure of the underlying manifold and the dynamics of the system.

The discrete representation of the time series $X(t_n)$ leads to a point cloud $\{\psi_\eta^m(t_n)\}$ called the Takens embedding. In the case that a time series contains periodicity, the Takens embedding is a point cloud with a circular structure. Of course, we can compute the 1-dimensional persistence of the generated point cloud in order to measure the size of this structure. Circularity in the point cloud is quantified in the persistence diagram as a point far from the diagonal as shown in the first row of Fig. 3. If the time series is just noise, there is no prominent off-diagonal point which can be seen in the second row of Fig. 3. For a theoretical treatment of persistence and delay embeddings see Perea and Harer (2015) and Perea (2016).

3.3 Featurization of Persistence Diagrams

Persistence diagrams encode a great deal of information about the structure of a point cloud, making it very useful for data analysis. However, because the space of persistence diagrams does not have an inner product structure, direct application of most machine learning algorithms is not possible. Thus, we turn to the featurization of persistence diagrams for help; that is, the construction of a map which returns a point $\phi(D) \in \mathbb{R}^N$ for a given persistence diagram D stored as a list of its off diagonal points.

For this work, we choose the features of Adcock et al. (2016) combined with maximum persistence. The basic idea of this method of featurization is to work with polynomials defined on the collection of off-diagonal points, but which respect the structure inherent in the persistence diagram. First, these polynomials must be able to accept persistence diagrams with different numbers of off-diagonal points since even beginning with input data containing the same number of points, the persistence diagrams can be vastly different in size. Secondly, while a persistence diagram can be stored as its list of off-diagonal points, $D = \{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}$, the order of presentation does not change the diagram, so any given function must present the same output no matter the input order. To this end, we choose the following collection of functions,

$$\begin{aligned}
f_1(D) &= \sum x_i(y_i - x_i),\\
f_2(D) &= \sum (\overline{y} - y_i)(y_i - x_i),\\
f_3(D) &= \sum x_i^2(y_i - x_i)^4,\\
f_4(D) &= \sum (\overline{y} - y_i)^2(y_i - x_i)^4,\\
f_5(D) &= \max\{(y_i - x_i)\}
\end{aligned}$$

where $\bar{y}$ is the maximum death time, and the summations and maximum are each taken over all points in D.

4. NUMERICAL SIMULATION

Two sets of simulations were used: MATLAB's DDE23 command for the deterministic case in Eq. (1), and an Euler-Maruyama simulation for the stochastic system of Eq. (4). For the stochastic case, multiple realizations of the solution path were generated for use in the analysis.

The system parameters that were used in the simulations are $\zeta = 0.03$, $\rho = 0.01$, and $\alpha = 0.75$, and the simulation was performed over a 100×100 grid in the $(\Omega/\omega_n, b)$ space. To provide a basis of comparison for the analysis of both the deterministic an the stochastic models, we also include the results of the linearized stability analysis of the noise-free model Eq. (1). There are several methods available in the literature to do this, e.g. zero-order approximation (Altintaş and Budak, 1995), semi-discretization (Insperger and Stépán, 2004), Chebyshev collocation (Butcher and Bobrenkov, 2011), extended Tlusty's law method Otto et al. (2014), or spectral element method (Khasawneh and Mann, 2011). In this study, we use the latter approach with one temporal element, a polynomial of degree 12, and a 100×100 grid in the $(\Omega/\omega_n, b)$ plane (see Khasawneh and Mann (2011) more details). The resulting stability diagram is the gray line in Fig. 4. The region below the line represents the chatter-free regime while the region above marks the chatter regime.

The stochastic case, Eq. (4) was simulated using the Euler-Maryuama method (Buckwar, 2000), while the Brownian path was created using the approach described in Higham (2001). For more details on the method and parameters, please refer to Khasawneh and Munch (2016).

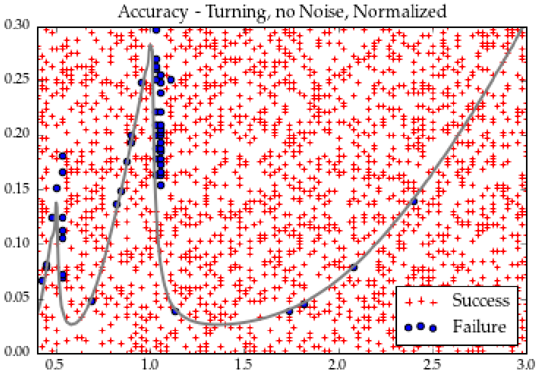

Fig. 4. Accuracy plot in turning with no noise.

5. METHODS, RESULTS, AND DISCUSSION

Following the computations of Khasawneh and Munch (2016), we used persistent homology to analyze the time series generated by the numerical simulation. For every experiment, each time series was given as a series of samples $s_n = X_{\Omega/\omega_n,b}(t_n)$ for $t_n \in [0, T]$, where T depends on the value of Ω/ω_n according to $\frac{2^5 \times 2\pi}{\Omega/\omega_n}$. The second half of the time series was retained to avoid transients, and was sub-sampled to speed up persistence computations. This resulted in 264 remaining data points evenly distributed in $[T/2, T]$ for all the time series, both with and without noise.

The time lag η was determined for each time series using the first zero of the autocorrelation function. This was used to construct the Takens embdding as a point cloud in $\mathbb{R}^3$. We computed the 0- and 1-dimensional persistence diagrams for each Takens embedding using the

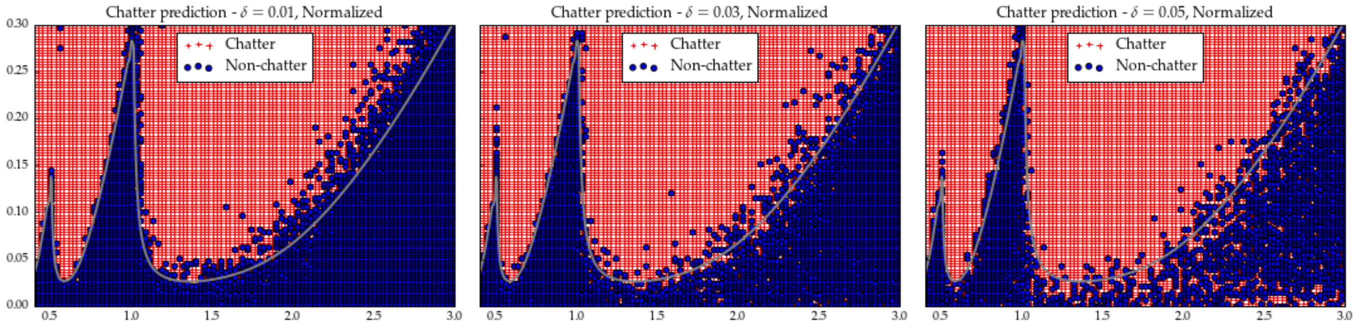

Fig. 5. The results of classification on the data from the stochastic model using the classifier trained on the deterministic model. The stability boundary is included in gray for comparison, but is not included in the calculations.

M12 package (Harer et al., 2014). For each of the diagrams, we computed the five functions described in Sec. 3.3. Because the 0-dimensional persistence diagram for the Rips complex of a point cloud has all birth times equal to 0 in the 0-dimensional persistence diagram, f_1 and f_3 were entirely 0, so these features were discarded. This resulted in a total of 8 input features computed for each parameter value which were normalized prior to the analysis. Machine learning was done using the `sklearn` python package. Specifically, a logistic regression classifier[1] was trained on 80% of the time series and tested using the remaining 20%. The labels for classification were determined using the stability diagram of the deterministic model (Eq. 1).

Using the featurization of the persistent homology for machine learning resulted in a 97% classification rate. The confusion matrix for the classification test on the noise-free data set is

$$\text{Pred.}\quad \begin{array}{rcc} & \multicolumn{2}{c}{\text{True}} \\ & \text{Chatter} & \text{Not-Chatter} \\ \text{Chatter} & 1250 & 46 \\ \text{Not-Chatter} & 6 & 698 \end{array}$$

and the locations of the misclassified time series can be seen in the blue dots of Fig. 4. Notice that not only is the classification rate incredibly high, the locations of the misclassified time series are restricted to the boundary of the stability matrix. As the stability diagram was calculated numerically on a discrete mesh, we suspect that issues with meshing and possibly with machine precision have led to a slight lowering of the two sharp peaks in the boundary. If this is true, the machine learning model is actually recreating this peak but interpreting it as a misclassification since our labels are constructed using the calculated boundary.

We then used the trained logistic model on the data sets arising from the stochastic model (Eq. (4)). That is, using the same procedure for generating the Takens embedding and computing the normalized 8-dimensional feature vector, we classified the resulting output. The results of the classification can be seen in Fig. 5.

Notice in particular that the labeling implies that small amounts of noise (i.e., $\delta = 0.01$) actually increases the stability of the system since more parameters are considered to have a non-chatter output. This can be attributed to stochastic resonance (Kuske (2010)). Meanwhile, we can see that as the noise increases, (i.e., $\delta = 0.03$ and 0.05) the chatter labeled parameters increasingly infiltrate the non-chatter region, thus leading to more instability. Nevertheless, the approach we describe can provide chatter classification for signals produced by complicated and noisy manufacturing systems especially when other methods cannot be applied.

REFERENCES

Adcock, A., Carlsson, E., and Carlsson, G. (2016). The ring of algebraic functions on persistence bar codes. *Homology, Homotopy and Applications*, 18(1), 381–402. doi:10.4310/HHA.2016.v18.n1.a21.

Altintaş, Y. and Budak, E. (1995). Analytical prediction of stability lobes in milling. *CIRP Annals*, 44(1), 357–362.

Altintas, Y. and Chan, P.K. (1992). In-process detection and suppression of chatter in milling. *International Journal of Machine Tools and Manufacture*, 32(3), 329–347. doi:10.1016/0890-6955(92)90006-3.

Bediaga, I., noa, J.M., Hernández, J., and de Lacalle, L.L. (2009). An automatic spindle speed selection strategy to obtain stability in high-speed milling. *International Journal of Machine Tools and Manufacture*, 49(5), 384–394. doi:10.1016/j.ijmachtools.2008.12.003.

Buckwar, E. (2000). Introduction to the numerical analysis of stochastic delay differential equations. *Journal of Computational and Applied Mathematics*, 125(1-2), 297–307. doi:10.1016/S0377-0427(00)00475-1. Numerical Analysis 2000. Vol. VI: Ordinary Differential Equations and Integral Equations.

Butcher, E.A. and Bobrenkov, O.A. (2011). On the chebyshev spectral continuous time approximation for constant and periodic delay differential equations. *Communications in Nonlinear Science and Numerical Simulation*, 16(3), 1541–1554. doi:10.1016/j.cnsns.2010.05.037.

Choi, T. and Shin, Y.C. (2003). On-line chatter detection using wavelet-based parameter estimation. *Journal of Manufacturing Science and Engineering*, 125(1), 21–28. doi:10.1115/1.1531113.

Delio, T., Tlusty, J., and Smith, S. (1992). Use of audio signals for chatter detection and control. *Journal of Manufacturing Science and Engineering*, 114(2), 146–157. doi:10.1115/1.2899767.

Edelsbrunner, H. and Harer, J. (2010). *Computational Topology: An Introduction.* American Mathematical Society.

Elias, J. and Narayanan Namboothiri, V. (2014). Cross-recurrence plot quantification analysis of input and out-

[1] `sklearn.linear_model.LogisticRegression` with default inputs

put signals for the detection of chatter in turning. *Nonlinear Dynamics*, 76(1), 255–261. doi:10.1007/s11071-013-1124-0.

Faassen, R. (2007). *Chatter prediction and control for high-speed milling: modelling and experiments.* printed by University Press Facilities, Eindhoven, the Netherlands.

Gradisek, J., Govekar, E., and Grabec, I. (1998). Using coarse-grained entropy rate to detect chatter in cutting. *Journal of Sound and Vibration*, 214(5), 941–952. doi:10.1006/jsvi.1998.1632.

Harer, J., Slaczedek, J., and Bendich, P. (2014). Ripscollapse: discrete morse theory and fast computation of one-dimensional persistence. *Manuscript, Duke University.*

Higham, D. (2001). An algorithmic introduction to numerical simulation of stochastic differential equations. *SIAM Review*, 43(3), 525–546. doi:10.1137/S0036144500378302.

Hino, J., Okubo, S., and Yoshimura, T. (2006). Chatter prediction in end milling by fnn model with pruning. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 49(3), 742–749. doi:10.1299/jsmec.49.742.

Insperger, T. and Stépán, G. (2004). Updated semi-discretization method for periodic delay-differential equations with discrete delay. *International Journal for Numerical Methods*, 61, 117–141.

Insperger, T., Barton, D.A., and Stepan, G. (2008). Criticality of hopf bifurcation in state-dependent delay model of turning processes. *International Journal of Non-Linear Mechanics*, 43(2), 140–149. doi:10.1016/j.ijnonlinmec.2007.11.002.

Kakinuma, Y., Sudo, Y., and Aoyama, T. (2011). Detection of chatter vibration in end milling applying disturbance observer. *{CIRP} Annals - Manufacturing Technology*, 60(1), 109 – 112. doi:10.1016/j.cirp.2011.03.080.

Kantz, H. and Schreiber, T. (2004). *Nonlinear Time Series Analysis.* Cambridge University Press.

Khasawneh, F.A. and Mann, B.P. (2011). A spectral element approach for the stability of delay systems. *International Journal for Numerical Methods in Engineering*, 87(6), 566–592. doi:10.1002/nme.3122.

Khasawneh, F.A. and Munch, E. (2016). Chatter detection in turning using persistent homology. *Mechanical Systems and Signal Processing*, 70-71, 527–541. doi:http://dx.doi.org/10.1016/j.ymssp.2015.09.046.

Kuljanic, E., Totis, G., and Sortino, M. (2009). Development of an intelligent multisensor chatter detection system in milling. *Mechanical Systems and Signal Processing*, 23(5), 1704–1718. doi:10.1016/j.ymssp.2009.01.003.

Kuske, R. (2010). Competition of noise sources in systems with delay: The role of multiple time scales. *Journal of Vibration and Control*, 16(7-8), 983–1003. doi:10.1177/1077546309341104.

Ma, L., Melkote, S.N., and Castle, J.B. (2013). A model-based computationally efficient method for on-line detection of chatter in milling. *Journal of Manufacturing Science and Engineering*, 135(3), 031007. doi:10.1115/1.4023716.

Munch, E. (2017). A user's guide to topological data analysis. *Journal of Learning Analytics*, 4(2). doi:10.18608/jla.2017.42.6.

Nair, U., Krishna, B., Namboothiri, V., and Nampoori, V. (2010). Permutation entropy based real-time chatter detection using audio signal in turning process. *IJAMT*, 46(1-4), 61–68. doi:10.1007/s00170-009-2075-y.

Oksendal, B. (2007). *Stochastic differential equations.* Springer, 6th edition.

Otto, A., Rauh, S., Kolouch, M., and Radons, G. (2014). Extension of tlusty's law for the identification of chatter stability lobes in multi-dimensional cutting processes. *Int. J. Mach. Tools Manuf.*, 82-83, 50 – 58.

Perea, J.A. (2016). Persistent homology of toroidal sliding window embeddings. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, 6435–6439. IEEE.

Perea, J.A. and Harer, J. (2015). Sliding windows and persistence: An application of topological methods to signal analysis. *Foundations of Computational Mathematics*, 15(3), 799–838. doi:10.1007/s10208-014-9206-z.

Schmitz, T.L., Medicus, K., and Dutterer, B. (2002). Exploring once-per-revolution audio signal variance as a chatter indicator. *Machining Science and Technology*, 6(2), 215–233. doi:10.1081/MST-120005957.

Sims, N.D. (2009). *Dynamics Diagnostics: Methods, Equipment and Analysis Tools*, chapter 4, 85–115. Springer Series in Advanced Manufacturing. Springer.

Smith, S. and Tlusty, J. (1992). Stabilizing chatter by automatic spindle speed regulation. *{CIRP} Annals - Manufacturing Technology*, 41(1), 433–436. doi:10.1016/S0007-8506(07)61238-4.

Stepan, G. (2001). Modelling nonlinear regenerative effects in metal cutting. *Philosophical Transactions of the Royal Society of London. Series A:Mathematical, Physical and Engineering Sciences*, 359(1781), 739–757. doi:10.1098/rsta.2000.0753.

Takens, F. (1981). Detecting strange attractors in turbulence. In D. Rand and L.S. Young (eds.), *Dynamical Systems and Turbulence, Warwick 1980*, volume 898 of *Lecture Notes in Mathematics*, 366–381. Springer Berlin Heidelberg. doi:10.1007/BFb0091924.

Tlusty, J. (2000). *Manufacturing Processes and Equipment.* Prentice Hall, Upper Saddle River, NJ, 1 edition.

Tlusty, J. and Andrews, G. (1983). A critical review of sensors for unmanned machining. *{CIRP} Annals - Manufacturing Technology*, 32(2), 563–572. doi:10.1016/S0007-8506(07)60184-X.

Tsai, N.C., Chen, D.C., and Lee, R.M. (2010). Chatter prevention for milling process by acoustic signal feedback. *The International Journal of Advanced Manufacturing Technology*, 47(9-12), 1013–1021. doi:10.1007/s00170-009-2245-y.

van Dijk, N.J.M., Doppenberg, E.J.J., Faassen, R.P.H., van de Wouw, N., Oosterling, J.A.J., and Nijmeijer, H. (2010). Automatic in-process chatter avoidance in the high-speed milling process. *Journal of Dynamic Systems, Measurement, and Control*, 132(3), 031006. doi:10.1115/1.4000821.

Yao, Z., Mei, D., and Chen, Z. (2010). On-line chatter detection and identification based on wavelet and support vector machine. *Journal of Materials Processing Technology*, 210(5), 713–719.