Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Tension Monitoring of Toothed Belt Drives Using Interval-Based Spectral Features

Moritz Fehsenfeld * **Johannes Kühn** ** **Mark Wielitzka** *
Tobias Ortmaier *

* *Leibniz University Hannover, Institute of Mechatronic Systems, An der Universität 1, 30823 Garbsen, Germany (e-mail: fehsenfeld@imes.uni-hannover.de).*
** *Lenze Automation GmbH, Am Alten Bahnhof 11, 38122 Braunschweig, Germany*

Abstract: Toothed belt drives are used in manifold automation applications. But only if the belt tension is properly adjusted, optimal working conditions are ensured. A loss of efficiency or even breakdowns might be the consequences otherwise. For this reason, tension monitoring reduces operation costs and may prevent failures. In order to meet industrial requirements, the monitoring is supposed to rely on standard sensor data. From this data, features are extracted in time and frequency domain which are passed on to a random forest. For further improvement, a segmentation of the frequency spectrum is performed beforehand. In this way, interval-based spectral features can be extracted to capture small distinctive parts in the frequency domain. For this purpose, two different segmentation procedures are compared in a random forest regression. A belt drive powered by a 1.9 kW synchronous servomotor is used to evaluate the proposed approaches in two different industrial scenarios. The experimental results show that both segmentation methods enhance the performance of a tree-based regression and offer a reliable tension prediction.

Keywords: Fault detection and diagnosis, Machine learning, Industrial production systems, Time series modeling, Segmentation

1. INTRODUCTION

Condition monitoring is a widespread solution for many industrial applications to detect failures during operation. Monitoring of electrical drives mainly focuses on the motor where extensive work has been done to identify faulty conditions including bearing faults (e.g. Prieto et al. (2013)), rotor faults (e.g. Gyftakis et al. (2013)) and winding faults (e.g. Filho et al. (2014)). An overview of the field is given by Riera-Guasp et al. (2015). However, failures in belt drives can also force an entire production line to stop resulting in high expenses. But preventing downtimes is not the only issue. To keep the wear of involved components at a minimum and to increase efficiency to a maximum, it is necessary to tension the belt properly. If pretension is set too high, cord fatigue and tooth wear might be the consequence. If the belt is too loose, the tooth load increases. Both cases lead to early failures and non-optimal operation conditions. For this reason, a tension monitoring system helps to keep belt drives close to their optimal working point.

1.1 Effects of Belt Tension on Measurement Data

The basis for distinction of different tension conditions is finding patterns in the sensor data that are changing as the belt tension is increased or decreased. In industrial applications, the proper belt tension is determined by evaluating the first mode natural frequency of the transverse belt oscillations. When considering the belt as a vibrating string, the belt tension F is given by

$$F = 4f^2l^2m. \tag{1}$$

Besides the frequency f, the tension, therefore, depends on the belt's mass per length m and its length l. This relationship is often used in fault detection and tension monitoring, leading to the need to measure the belt's transverse frequency during operation. Khazaee et al. (2017) and Ucar et al. (2014) obtained vibration signals by employing an optical laser that captures the belt's transverse motion to classify different failures. Musselman and Djurdjanovic (2012) utilized strain gauges to monitor belt tension. Hu et al. (2016) capture belt oscillations using an electrostatic sensor. All these approaches require external sensors which are not always available in industrial applications due to cost and place restrictions. Therefore, it is desirable to build a monitoring system that only requires data from standard sensors. Motor current signature analysis (MCSA) is a very popular procedure in fault diagnosis because the motor current is commonly available. It is used to classify belt failures like crack and wear by Kang et al. (2018). Picot et al. (2017) distinguish four different belt tension levels based on motor current.

1.2 Principles of Belt Tension Monitoring

This work presents a data-driven procedure using conventional machine learning techniques to monitor the pretension of a toothed belt drive relying only on measurement

data that is accessible in common servomotors. To get a better understanding of the system's behavior under changing belt tension, a frequency spectrum, which is averaged over multiple velocity measurements is shown in Fig. 1. It highlights different belt states ranging from "critically loose" to "strongly tightened". At critically low tension the magnitude constantly decreases as the frequency grows and no peaks are visible. If the belt is properly tensioned, several peaks shifting dependent on the tension can be observed. This differs from other fault diagnosis applications where a certain failure changes the system's behavior significantly. Following this consideration, the algorithm must be capable of detecting small changes in the frequency spectrum in order to achieve a high quality. Thus, one might assume that it is beneficial to focus only on distinguishable parts. During feature extraction, which is described in section 2, an effort is made to take frequency shifts into account by using interval-based features. A distinction is made between a random segmentation and an optimized segmentation. Section 3 provides an overview of the experimental setup. The procedure to determine all relevant hyperparameters is presented in section 4 and subsequently the results are shown in section 5.

2. FEATURE-BASED BELT MONITORING

In this work, belt tension monitoring is realized by a random forest regression. With a view to capture even small changes, supplementary features are extracted from intervals of the frequency domain. The segmentation procedure is done in two different ways compared throughout this work. First all subsequences are chosen randomly and second, a global optimization algorithm is run to find an optimal placement.

2.1 Random Forest Regression

Bagnall et al. (2016) recommend among others a random forest as a benchmark predictive model. Although it achieves high quality in many applications it is a comparatively simple model, which is advantageous over copious advanced black box techniques. Therefore a random forest is chosen in order to meet industrial requirements.

In this study binary decision trees are used to construct the random forest based on the "Classification and Regression Tree" (CART) algorithm described by Breiman (2001). Each tree is created on a bootstrap sample of the original data combined with a random feature selection at every node. Finally the overall decision is done by averaging over all trees in the ensemble. The mean squared error (MSE) between actual output $\boldsymbol{y}$ and predicted output $\hat{\boldsymbol{y}}$ averaged over N observations

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 \tag{2}$$

is chosen as quality criterion.

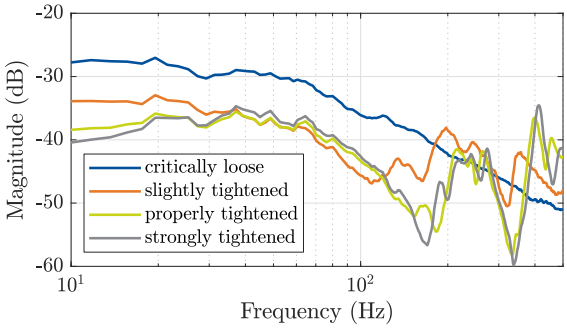


Fig. 1. Comparison of averaged velocity frequency spectrums for different belt tensions.

2.2 Feature Extraction

Time series data can be expressed as a pair of input $\boldsymbol{x}_i$ and related output value y_i for each observation i. Transformed to frequency domain each observation consists of amplitudes $\boldsymbol{a}_i$ and the related frequencies $\boldsymbol{f}_i$. The obtained phase data is omitted. The raw data in time and frequency domain is not appropriate for conventional machine learning algorithms like a random forest. A common way to overcome this problem is to extract features from the raw data, which are passed to the learning algorithm. In this way, the prediction is done by evaluating a feature-based representation of the data rather than the raw data. Hence, the quality of a machine learning algorithm crucially depends on the feature engineering performed beforehand. During this stage, domain knowledge can be incorporated to improve the results. Therefore, features that aim at capturing small displacements in the frequency domain are created. For that reason, the usage of interval-based spectral features is investigated in this work. Unlike global features that are affected by the entire data length, local features are extracted from smaller subsequences.

There are numerous off-the-shelf functions proposed in the literature to extract features from time-series data (see e.g. Fulcher and Jones (2014)). An overview of selected functions that are applied to time and frequency domain data is shown in table 1. These are mainly statistical functions quantifying properties of the data distribution like skewness ($\mathcal{F}_3$ and $\mathcal{F}_{10}$) or kurtosis ($\mathcal{F}_4$ and $\mathcal{F}_{11}$). Functions like the weighted mean frequency ($\mathcal{F}_{14}$) are selected to cover among others a shift of peaks. Note that each time series is first standardized to zero mean and unit standard deviation before extracting features.

In literature, segmentation is discussed in order to omit noisy data parts and focus on distinctive ones to enhance feature quality. These methods are mainly applied to time series data but they are not restricted to it and can be used for any kind of sequential data (Fu (2011)). The segmentation requires to find suitable split points first to define the sections. Second, a procedure to extract features needs to be defined once all segments are selected.

For the sake of simplicity, all segments are treated the same and every function introduced before is used to create summary features. Numerous procedures are applicable to obtain a segmentation of the spectrum. In the following two ways are analyzed concerning their performance in the given application. They mainly distinguish in the way the interval limits are chosen. First a random segmentation adapted from the *time series forest* proposed by Deng et al. (2013) is presented. Subsequent the location of subsequences is further investigated and all boundaries are optimized utilizing a Bayesian optimization algorithm. In the following, the first approach of random sequences is

Table 1. Functions to extract features from a signal of L and K samples respectively.

Time domain	Frequency domain
$\mathcal{F}_1 = \frac{1}{L}\sum_{i=1}^{L} x_i$	$\mathcal{F}_8 = \frac{1}{K}\sum_{i=1}^{K} a_i$
$\mathcal{F}_2 = \frac{1}{L}\sum_{i=1}^{L}(x_\mathrm{i} - \mathcal{F}_1)^2$	$\mathcal{F}_9 = \frac{1}{K}\sum_{i=1}^{K}(a_\mathrm{i} - \mathcal{F}_8)^2$
$\mathcal{F}_3 = \frac{\sum_{i=1}^{L}(x_\mathrm{i}-\mathcal{F}_1)^3}{L\cdot\mathcal{F}_2^3}$	$F_{10} = \frac{\sum_{i=1}^{K}(a_\mathrm{i}-\mathcal{F}_{11})^3}{k\cdot\mathcal{F}_9^3}$
$\mathcal{F}_4 = \frac{\sum_{i=1}^{L}(x_\mathrm{i}-\mathcal{F}_1)^4}{L\cdot\mathcal{F}_2^4}$	$\mathcal{F}_8 = \frac{\sum_{i=1}^{K}(a_\mathrm{i}-\mathcal{F}_9)^4}{K\cdot\mathcal{F}_9^4}$
$\mathcal{F}_6 = \frac{1}{L}\sum_{i=1}^{L} \lvert x_\mathrm{i}\rvert$	$\mathcal{F}_{12} = \sum_{i=1}^{K} a_i^2$
$\mathcal{F}_7 = \left(\frac{1}{L}\sum_{i=1}^{L}\sqrt{\lvert x_i\rvert}\right)^2$	$\mathcal{F}_{13} = \frac{\sum_{i=1}^{K} a_i\cdot f_i}{\sum_{i=1}^{K} a_i}$

referred to as *random segmentation forest* (RSF) and the latter as *optimized segmentation forest* (OSF).

2.3 Random Segmentation Forest

The biggest issue when using interval-based features is where to split the sequential data. One easy way to do so is to randomly select boundaries. Deng et al. (2013) describe an approach to create a *time series forest* (TSF) based on features from random subsequences. As the name implies it is a random forest approach where each tree is trained on an individual set of interval-based features. During training stage, a set of $\lfloor\sqrt{m}\rfloor$ subsequences is defined individually for each tree where m is the length of dataset. All split points are chosen randomly leading to different features as well. Summary functions are applied to every subsequence to obtain local features. For comparability the same feature functions described in table 1 are chosen. This leads to a set of 14 times $\lfloor\sqrt{m}\rfloor$ features which are used to train each tree. For further information on the algorithm refer to Deng et al. (2013).

There are two main hyperparameters to tune a TSF namely the number of trees N_T and the minimum interval length L_min. The number of trees influences the amount of intervals and by this the size of the related feature sets. Therefore, the computational time for feature extraction and regression learning increases as the number of trees grows. To ensure a level of information in every subsequence its minimum length L_min can be set externally.

2.4 Optimized Segmentation Forest

A random splitting leads to a vast amount of different intervals which are only evaluated during the random forest training. In order to find an optimal solution with only a few good intervals, the segmentation is expressed as an optimization problem. Each interval is defined by a starting point and an endpoint giving two parameters per section that need to be optimized. All points are only constrained by the frequency range of the sampled data. Since the location of a subsequence highly affects the features, the prediction result directly depends on it as well. Therefore, the quality of segmentation is evaluated based on the performance in a tree-based regression. For this reason, a random forest is trained and the mean squared error of a 3-fold cross-validation is calculated to determine the loss of a parameter combination. A Bayesian optimization algorithm (BOA) is used to run the optimization. For information on the algorithm refer for example to Snoek et al. (2012). To avoid a high-dimensional parameter space only one subsequence is optimized per run. The number is then gradually increased by keeping previous subsequences constant. By this, the optimization problem is split into numerous smaller ones leading to lower computational costs. On the other hand the performance is not found to be significantly lower than a one step optimization.

3. EXPERIMENTAL SETUP

The validation is conducted on a test bench depicted in Fig. 2. It comprises of three triangularly arranged belt pulleys connected by a toothed belt (AT-5 profile) with a total length of $l = 0.975\,\mathrm{m}$. The lower right pulley is linked to a synchronous servomotor with a rated torque of $M_0 = 5.5\,\mathrm{Nm}$ and a rated power of $P_0 = 1.9\,\mathrm{kW}$. The setup's control topology is characteristic of applications in automation consisting of a programmable logic controller (PLC) and a servo inverter which feeds the motor. The servomotor is equipped with an encoder to measure its position.

A roller can be moved up and down by a spindle drive allowing a continuous adjustment of the belt tension. A potentiometer is attached to the spindle drive to measure the roller's position and thereby the distance used for pretension. At this point, it should be noted that measurement data obtained from the potentiometer is not used to predict the belt tension but for labeling purposes only.

3.1 Measurement Data

The data acquisition is done by a PLC at a frequency of $f_\mathrm{s} = 1000\,\mathrm{Hz}$ which is a typical setup in many industrial applications. The employed servomotor is not equipped with additional sensors and thus only provides data that is available in off-the-shelf motors, which is:

- position (and derivatives),
- torque and
- temperature.

Because the belt tension can not be measured directly it is represented by the spindle drive's position. The first mode natural frequency of the belt transverse oscillation is acoustically measured to determine the tension based on (1). Although the belt's behavior is non-linear the relation between position and belt tension can be linearly described

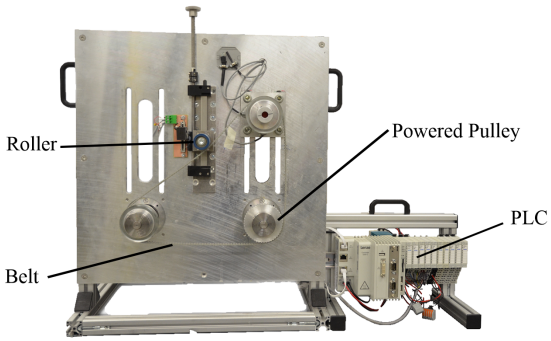
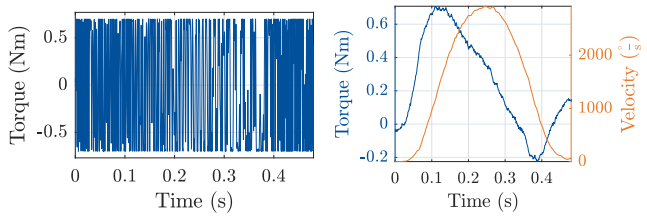


Fig. 2. Front view of the test bench.



(a) MFE (b) Jerk-limited trajectory

Fig. 3. Time domain representation of the excitations.

in the considered range. The position is be varied from $y = 0\,\text{cm}$ ($F \approx 0\,\text{N}$) at the upper end of the potentiometer where the roller barely touches the belt to a maximum of $y = 5\,\text{cm}$ ($F = 230\,\text{N}$) where a tension higher than desired is induced.

An excitation is needed to predict the actual belt tension. In general, any random motion that is applicable in an automation industry environment can serve for this purpose. But the recorded time series data and by that the prediction quality highly depends on it. Furthermore the excitation determines the operation conditions in practice.

Fault detection approaches can generally be divided into active and passive (see e.g. Punčochář and Škach (2018)). The excitations assessed throughout this work are chosen in a way that they cover both groups:

1. jerk-limited positioning trajectory (JLT)
2. multifrequency sinusoidal excitation (MFE).

A jerk-limited trajectory is a typical positioning movement in industrial settings and is therefore available during normal operation. It consists of matching set values of position, velocity, and acceleration, that are generated during path planning. As a result, all available sensor signals are logged during motion and the belt monitoring can be done without disturbing the normal operation.

In the second scenario an auxiliary signal is injected into the system to evaluate the belt tension. For this purpose a MFE signal is chosen, which is often used for parameter identification (see e.g. Evans et al. (1992)). It is a superposition of N_F sinusoidal functions with different angular frequency ω_i and phase shift φ_i. The torque set value $M(t)$ which is applied can be described by

$$M(t) = \sum_{i=1}^{N_\text{F}} \sin(\omega_i t + \varphi_i). \tag{3}$$

During the excitation the system's velocity response is examined leading to a time series with a fixed number of samples that is used for feature extraction. Due to its design it excites many frequencies in a certain range. By that it is assured that the system characteristics are comprehensively visible. A time-domain representation of the resulting torque for both excitations is depicted in Fig. 3(a) and 3(b) to highlight the differences.

4. HYPERPARAMETER TUNING

This section focuses on the adaption of hyperparameters that are directly connected to the extraction of interval-based features. All other parameters especially those of underlying decision trees are not tuned and globally kept constant.

4.1 Random Segmentation Forest

The main parameters of a TSF are the number of trees and the minimum length of a subsequence. The maximum number of trees is set to $N_\text{T} = 20$ to limit the computational costs. A grid search over both parameters with a step size of $\Delta L_\text{min} = 5\,\text{samples}$ and $\Delta N_\text{T} = 2\,\text{trees}$ is carried out. The results are shown in Fig. 4. In general, a small number of trees leads to a slightly higher prediction error while no significant tendency can be observed for the minimum length. The minimum MSE is at a number of $N_\text{T} = 14\,\text{trees}$ and a minimum length of $L_\text{min} = 30\,\text{samples}$ and thus chosen as hyperparameter combination.

4.2 Optimized Segmentation Forest

The total number of subsequences is determined by gradually increasing it during segmentation optimization as described in section 2.4. The quality is evaluated based on the MSE that is achieved in a random forest regression. It can be seen from Fig. 5 that the MSE decreases as the number of subsequences grows. The maximum number of subsequences is set to $N_\text{seq} = 10$ because for both excitations no significant improvement is observed beyond that point.

Since the loss function depends on the train-test split it is subject to scattering. Hence, the segment boundaries slightly vary from run to run but the overall performance is almost constant. To illustrate the general optimization results a characteristic example segmentation is shown in Fig. 6 for the MFE. The frequency spectrum shown at the beginning in Fig. 1 is plotted to clarify the findings. The segment width ranges from small to almost across the entire spectrum. A concentration of starting and endpoints can be seen close to peaks in the spectrum which supports the initial intention to recognize their shift.

5. RESULTS OF TENSION MONITORING

The tension monitoring algorithms are tested under two aforementioned excitations. In order to obtain meaningful results, the training, validation and test dataset are treated separately. Training and validation dataset are used for model creation and determination of hyperparameters. The training dataset contains 900 measurements for both excitation movements. A uniform distribution over the output y is intended to eliminate data imbalances as a

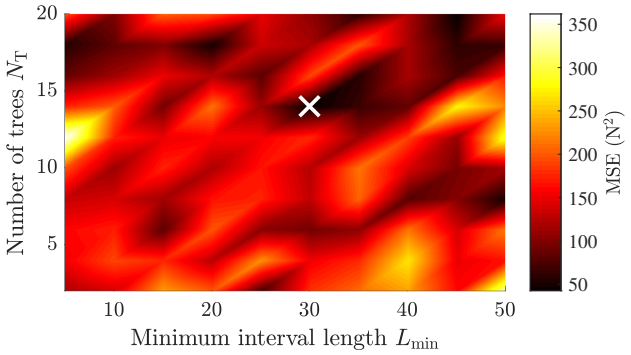
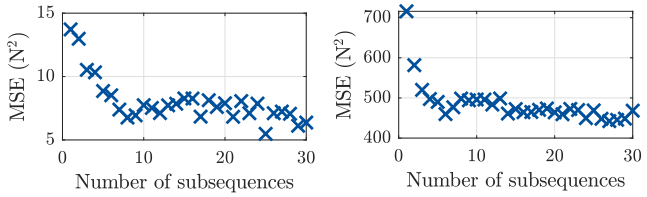


Fig. 4. Result of the TSF hyperparameter grid search where X marks the minimum.



(a) MFE (b) Jerk-limited Trajectory

Fig. 5. Investigation of the number of subsequences for interval-based features.

reason for differences in performance. All results shown in this section are based on the test dataset which is recorded separately from the training. A random forest where all interval features are omitted and only global features are kept is used as a reference to evaluate the interval-based approaches.

For the MFE a comparison of the actual output with the predicted output for each sample in the test dataset is shown in Fig. 7. The reference results which are depicted in Fig. 7(a) show an occasional offset where the predicted value differs from the actual output. A further improvement can be observed for both segmentation forests shown in Fig. 7(b). Almost no deviation between predicted and actual output is visible resulting in a low MSE.

The MFE can be considered more suitable for monitoring since it is optimized to excite a broad spectrum of frequencies while the JLT is simply a typical movement during operation in the automation industry. This is reflected in the achieved results depicted in Fig. 8. Especially the reference performance drops compared to the MFE as demonstrated in Fig. 8(a). Many predicted samples have a clear deviation from the actual value. Especially at lower belt tension a larger improvement compared to the reference is accomplished for a JLT by employing a segmentation. At higher belt tension several outliers remain in either case.

As pointed out earlier the quality varies due to differences in the segmentation. Therefore, the results of 30 runs are determined and the mean and standard deviation are given in table 2 for both excitations. The active fault detection approach of using a MFE leads to significantly better regression results in all cases. Even a global feature approach yields a lower MSE than every procedure for a JLT. Furthermore it can be concluded that both segmentation procedures lead to an improvement in both scenarios. The optimization (OSF) does not lead to a significant

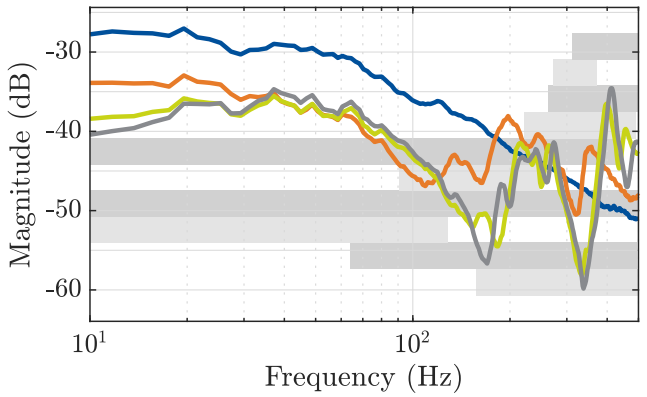
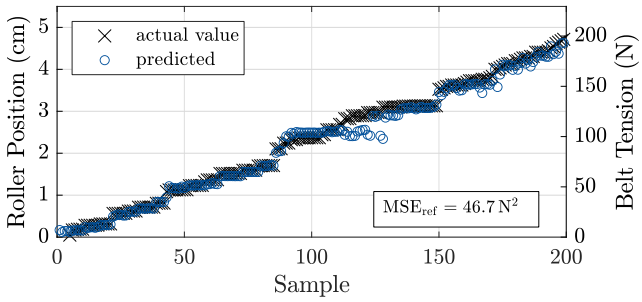
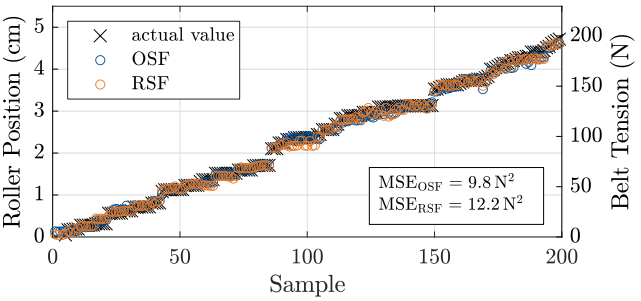


Fig. 6. Distribution of optimized subsequences (gray) for a MFE.



(a) Reference algorithm with global features



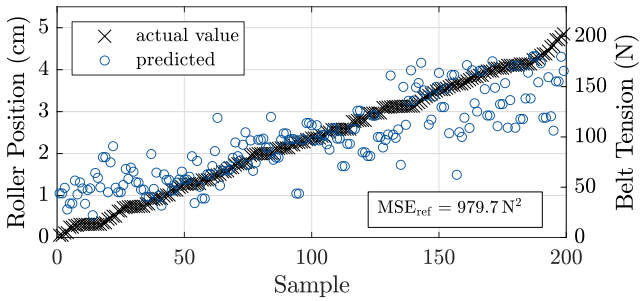
(b) Segmentation forest with interval-based features

Fig. 7. Results of tension monitoring during MFE excitation.

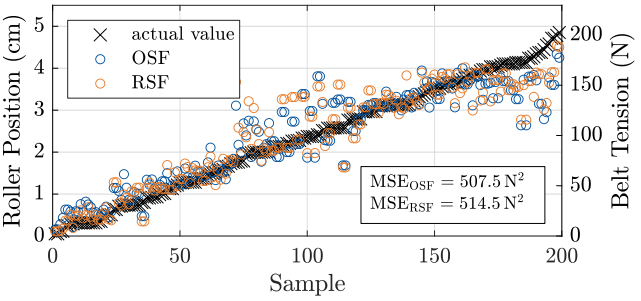
performance gain but reduces the number of intervals ($N_{\text{seq,OSF}} = 10$) compared to the RSF ($N_{\text{seq,RSF}} = 286$) and the standard deviation of the MSE.

6. CONCLUSIONS

In this work, a segmentation procedure is proposed, which was tested in two industrial scenarios. A test bench where the belt tension can be adjusted is utilized to validate the approaches. By applying a MFE the entire frequency range is excited and even a basic approach like a random forest



(a) Reference algorithm with global features



(b) Segmentation forest with interval-based features

Fig. 8. Results of tension monitoring during JLT excitation.

Table 2. Mean μ and standard deviation σ of the MSE over multiple runs.

	MFE	JLT
Global features	$\mu = 87.2\,\mathrm{N}^2$	$\mu = 982.9\,\mathrm{N}^2$
	$\sigma = 8.5\,\mathrm{N}^2$	$\sigma = 27.6\,\mathrm{N}^2$
Random segmentation	$\mu = 9.9\,\mathrm{N}^2$	$\mu = 478.9\,\mathrm{N}^2$
	$\sigma = 1.1\,\mathrm{N}^2$	$\sigma = 50.7\,\mathrm{N}^2$
Optimized segmentation	$\mu = 9.5\,\mathrm{N}^2$	$\mu = 472.2\,\mathrm{N}^2$
	$\sigma = 1.1\,\mathrm{N}^2$	$\sigma = 33.5\,\mathrm{N}^2$

based on global features leads to an acceptable prediction quality. A JLT as excitation results in a much higher prediction error for all approaches. It can be concluded that the excitation has a larger impact on the monitoring quality than the algorithm itself. The active approach (MFE), where an auxiliary signal is injected into the system, is considerably superior to the passive approach (JLT) using measurements form normal operation. However, we could demonstrate that a segmentation of the frequency spectrum yields an improvement regardless of the procedure. On average the optimized segmentation yields a slightly higher prediction quality over a random segmentation.

A passive approach offers the possibility to monitor the belt tension by collecting data during normal operation. In this way, there is no interaction between the monitoring system and the plant. On the contrary, an active approach intervenes in the normal operation to apply the auxiliary signal and analyze the belt tension. In the present case this leads to a significantly higher prediction accuracy as shown before. However, in industrial applications, it might not be necessary to predict the tension with high accuracy, but a warning if it drops below a certain threshold is sufficient. In this case, a trade-off needs to be done to select the appropriate monitoring procedure.

ACKNOWLEDGEMENTS

The authors of the Institute of Mechatronic Systems would like to thank Lenze Automation GmbH for enabling the cooperative project.

REFERENCES

Bagnall, A., Lines, J., Bostrom, A., Large, J., and Keogh, E. (2016). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.

Deng, H., Runger, G., Tuv, E., and Vladimir, M. (2013). A time series forest for classification and feature extraction. *Information Sciences*, 239, 142 – 153.

Evans, D.C., Rees, D., and Jones, D.L. (1992). Design of test signals for identification of linear systems with nonlinear distortions. *IEEE Transactions on Instrumentation and Measurement*, 41(6), 768–774.

Filho, P.C.M.L., Pederiva, R., and Brito, J.N. (2014). Detection of stator winding faults in induction machines using flux and vibration analysis. *Mechanical Systems and Signal Processing*, 42(1), 377 – 387.

Fu, T.C. (2011). A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1), 164 – 181.

Fulcher, B.D. and Jones, N.S. (2014). Highly comparative feature-based time-series classification. *IEEE Transactions in Knowledge and Data Engineering*, 3026–3037.

Gyftakis, K.N., Spyropoulos, D.V., Kappatou, J.C., and Mitronikas, E.D. (2013). A novel approach for broken bar fault diagnosis in induction motors through torque monitoring. *IEEE Transactions on Energy Conversion*, 28(2), 267–277.

Hu, Y., Yan, Y., Wang, L., and Qian, X. (2016). Non-contact vibration monitoring of power transmission belts through electrostatic sensing. *IEEE Sensors Journal*, 16(10), 3541–3550.

Kang, T., Yang, C., Park, Y., Hyun, D., Lee, S.B., and Teska, M. (2018). Electrical monitoring of mechanical defects in induction motor-driven v-belt–pulley speed reduction couplings. *IEEE Transactions on Industry Applications*, 54(3), 2255–2264.

Khazaee, M., Banakar, A., Ghobadian, B., Mirsalim, M.A., Minaei, S., and Jafari, S.M. (2017). Detection of inappropriate working conditions for the timing belt in internal-combustion engines using vibration signals and data mining. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 231(3), 418–432.

Musselman, M. and Djurdjanovic, D. (2012). Tension monitoring in a belt-driven automated material handling system. *CIRP Journal of Manufacturing Science and Technology*, 5(1), 67 – 76.

Picot, A., Fournier, E., Régnier, J., TientcheuYamdeu, M., Andréjak, J., and Maussion, P. (2017). Statistic-based method to monitor belt transmission looseness through motor phase currents. *IEEE Transactions on Industrial Informatics*, 13(3), 1332–1340.

Prieto, M.D., Cirrincione, G., Espinosa, A.G., Ortega, J.A., and Henao, H. (2013). Bearing fault detection by a novel condition-monitoring scheme based on statistical-time features and neural networks. *IEEE Transactions on Industrial Electronics*, 60(8), 3398–3407.

Punčochář, I. and Škach, J. (2018). A survey of active fault diagnosis methods. *IFAC-PapersOnLine*, 51(24), 1091–1098.

Riera-Guasp, M., Antonino-Daviu, J.A., and Capolino, G. (2015). Advances in electrical machine, power electronic, and drive condition monitoring and fault detection: State of the art. *IEEE Transactions on Industrial Electronics*, 62(3), 1746–1759.

Snoek, J., Larochelle, H., and Adams, R.P. (2012). Practical bayesian optimization of machine learning algorithms. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 25*, 2951–2959. Curran Associates, Inc.

Ucar, M., Ergun, R.E., and Cengiz, A. (2014). A novel failure diagnosis system design for automotive timing belts. *Experimental Techniques*, 38(5), 48–53.