

Received: 30 September 2020 / Accepted: 07 January 2021 / Published online: 29 March 2021

*data provision, anomaly detection,
machine learning, manufacturing,
condition monitoring*

Markus NETZER^{1*}

Yannic PALENGA¹

Philipp GOENNHEIMER¹

Juergen FLEISCHER¹

OFFLINE-ONLINE PATTERN RECOGNITION FOR ENABLING TIME SERIES ANOMALY DETECTION ON OLDER NC MACHINE TOOLS

Intelligent IoT functions for increased availability, productivity and component quality offer significant added value to the industry. Unfortunately, many old machines and systems are characterized by insufficient, inconsistent IoT connectivity and heterogeneous parameter naming. Furthermore, the data is only available in unstructured form. In the following, a new approach for standardizing information models from existing plants with machine learning methods is described and an offline-online pattern recognition system for enabling anomaly detection under varying machine conditions is introduced. The system can enable the local calculation of signal thresholds that allow more granular anomaly detection than using only single indexing and aims to improve the detection of anomalous machine behaviour especially in finish machining.

1. INTRODUCTION

As manufacturing companies implement more digitalization in an effort for their transformation towards Industry 4.0, problems arise when older machines have to be connected for data access. Many of these machines do not offer modern acquisition interfaces, making it difficult to integrate them into IoT solutions. Replacing this equipment is not a viable option, as manufacturers often fear that costs may outweigh the benefits offered by IoT integration. This leads to a slowdown of digitalization, which hinders industry-wide development and may also lead to competitive disadvantages compared to competing companies further ahead in digital transformation. Consequently, this leads to an increased need for new and standardized data acquisition methods when dealing with old machines. However, even older machine models in brownfield applications offer some ways of data access, often in the form of TCP/IP connectivity.

A central application leveraging this data is the condition monitoring of production and handling machines. By using recorded time-series data such as positional- or motor

¹ Institute of Production Science (wbk), Karlsruhe Institute of Technology (KIT), German

* E-mail: markus.netzer@kit.edu

<https://doi.org/10.36897/jme/132248>

current data, anomaly detection based on deviations from reference time series can be applied. The main advantage of detecting these anomalies is the avoidance of unplanned machine shutdowns by recognizing abnormal behavior in signals before a machine or component crash occurs. However, in finish machining, conventional condition monitoring or anomaly detection systems mainly implemented with single indexing are unable to detect small signal deviations due to noise on a global level. In this paper, a novel approach for supplying data to apply anomaly detection on a local level for older NC steered machine tools by using an offline-online pattern recognition is described. The approach segments time series into different subsequences corresponding to machining processes and does not require any long algorithm training or prior knowledge about the specific machine operations. This machining process segmentation is based on space condition which is recognized by positional data of the feed axes.

2. RELATED WORK

As heterogeneous production plants consist of different types of machines and new machines are introduced continually, the need to simplify and standardize this environment is greater than ever. Some research has been done on the characterization of these production plants by Ahmed Ismail [1], who view the manufacturing ecosystem as a multi-dimensional grouping of systems designed to support a variety of different business, operations, engineering, maintenance, and training functions involved in the manufacturing process. Due to its varied characteristics, the data that is generated during these different processes is heterogeneous in nature as well. This poses a challenge for data analysis, as data may be scattered across different databases adapted to the specific processes and may also be available in different formats. Data sources reach from applications to sensors and cyber-physical devices. Due to these circumstances, data acquisition itself is a major challenge, as without clean and structured data, no analysis can be performed at all. As per Ahmed Ismail [1], data analysis pipelines often use varying implementations of SQL databases (PostgreSQL, MySQL, MongoDB) for storage of the generated data. In cases where data is especially big, distributed storage implementations like HDFS are commonly used. However, these storage solutions are optimized for structured data which is often not available directly from machine tools. Even in developed manufacturing landscapes, a lot of data is saved in unstructured form, as producers only slowly adapt to the needs of Industry 4.0. Data can be extracted from logs or time series in order to be used for manufacturing prognostics and health management [2]. However, unified approaches for this are largely missing as difficulties arise case-wise, since manufacturing equipment is used for a vast variety of different tasks. One effort to create transferable connectivity was made by the MTConnect Project [3] that aims at providing a royalty free industry standard for data access via TCP/IP.

Based on this standard, Edrington [4] created a web-based RMMS solution for access to data from machine tools of different companies and with proprietary data communication protocols. When such standardized access is not possible, some approaches to reduce

complexity of data acquisition exist: when machines of the same type are used in different processes, their grouping into fleets may be beneficial for data acquisition and analysis, as data pipelines and algorithms may be transferable when connectivity is similar among them [5]. When data from machine tools is in addition generated at high frequencies and in great quantity, traditional storage becomes impossible. This may require the use of alternative techniques, such as complex event processing (CEP). Here, data is accessed as a stream before or even completely without being stored in a database.

Especially time series are a common machine data output and different approaches exist for structuring them in order to gain insights from their analysis. Many existing studies on extracting patterns from time series data are based on early breakthroughs of Lin [6] & Keogh [7], which form the foundations of motif recognition in this domain. There are different examples for algorithms that deal with time series data by imposing a standardized structure through transformations and data engineering. Sakurai et al. [8] introduced the SPRING-Algorithm for matching subsequences in times series with a reference, which allowed similarity matching in real-time fashion. Using this algorithm, data in time series format can be accessed quickly and allows for use in IOT applications. To further extend this traditional CEP approach, Emec [9] modified the Algorithm by improving the initialization procedure and the unsupervised training. Their proposed framework uses hall-effect sensors for data acquisition which allows a flexible usage on a variety of different machine tools, independent of their specific interfaces.

3. DATA ACQUISITION

Machines used in industry differ not only by machine type and variation but also by the type of controller (e.g. NC or PLC). In addition, many industrial steering types such as SIEMENS, FANUC, BOSCH REXROTH, BECKHOFF, HEIDENHAIN or other more uncommonly used ones exist. All these controllers provide controller data like motor current or positional data via their respective proprietary information models. However, the data accessibility based on different hardware or software structure will not be further focused upon in the following. The data required for the presented approach includes time series data like feed motor currents, spindle and position data. Extracting this data is not always trivial, as apart from the used information model, the parameter naming of the time series could vary with different controller types. The algorithm focuses on this data, as it is often readily available even on older machines, where external sensors may not be present and the costs of adding them may outweigh the benefits. In addition, positional data is useful for identifying subsequences that correspond to machine movements while motor currents (e.g. spindle current for milling tools) according to experience often contain anomalies. However, through the modularity of the approach, up to 200 signals can be used per axis for eventual anomaly detection.

The positional and motor current data is available in the controller unit of the respective machine and is either provided by the PLC or by infrastructure such as an OPC UA server that improves the accessibility of the data for the user. However, even if data is

provided via OPC UA, the identification of the required parameters can be a major challenge due to tens of thousands of nodes with static, binary or dynamic parameters available on the server.

OPC UA represents a key data interface and therefore serves as a foundation for the described approach of an intelligent and automated data provision of the required parameters. If an OPC UA server already exists, it can be read out directly by a corresponding client application. In case no OPC UA server exists, it must be implemented on the controller unit. Software such as the PLC Engine by TANI can be used for this, which allows an automated readout of Ethernet-capable controllers in connection with the implementation of an OPC UA server [10]. Based on the OPC UA Server as the basic interface, an intelligent identification of the parameters was implemented, which scans the data sources, recognizes the searched parameters based on their characteristics and assigns them to the required parameters. The identification of the parameters is achieved by machine learning and rule-based approaches. Here, a feature-based approach is used for the basic detection of the parameters, such as motor currents, position signals or spindle speeds. Three binary decision trees with 100 splits each with data from a 5-axis machining center were trained. If further differentiations are necessary, rule-based methods are used in a further stage based on the results. On the one hand, to differentiate between a nominal and an actual position, for example, a time offset or signal noise can be examined. On the other hand, to differentiate between parameters of different axes in a machine, deviating behavior between horizontal and vertical axes can be detected. The result of the identification is an assignment of the required parameters for the specific usage of these, e.g. for segmentation and anomaly detection described in the following chapter. The data provision can either be realized in the form of information on which node of the OPC UA Server which parameter data can be retrieved, or the corresponding time series data can be written to a defined structure in an SQL database.

4. SEGMENTATION & ANOMALY DETECTION

The approach described in the following provides a way to group signals into smaller subsequences, which can then allow anomaly detection on a local level. In finish machining, this provides a significant advantage over regular anomaly detection using single indexing: here, signals disguise in noise, and looking at global thresholds does not allow for the detection of anomalous behavior if signal deviation is not high enough to exceed global noise variance. By creating local thresholds, such deviations can be identified.

After data acquisition as described in Chapter 3, a time series is available, which describes the signal course over time using the extracted positional values. From this time series, representative subsequences are to be generated in an offline training approach (Offline Pattern Recognition) visualized in Fig. 1.

In this case, the position values were focused, which are assigned to points in time with equal intervals based on a constant sampling frequency. This data is accompanied by so-called information signals like motor current or torque, which are time series containing possible anomalies. For each timestamp in the positional time series, multiple other

information signals may be available. In order to split the time series into subsequences, the approach focuses on minima in the positional data. However, as these series are usually noisy, the search for local minima cannot be executed without applying smoothing beforehand (Fig. 2).

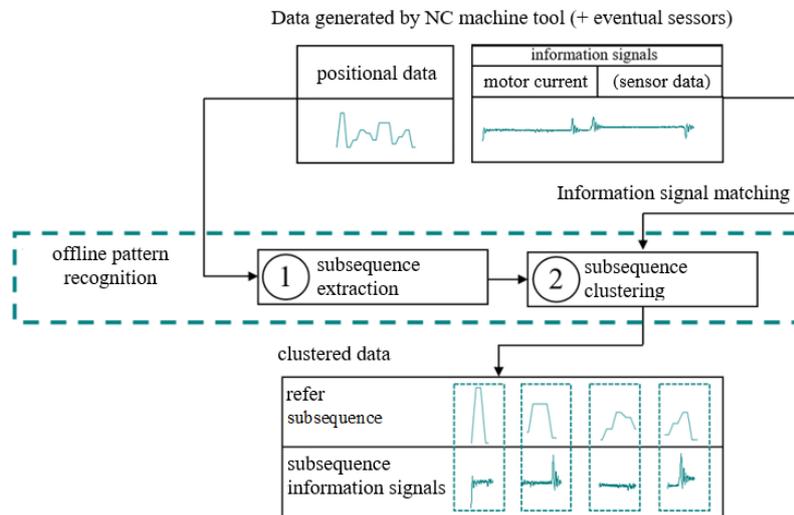


Fig. 1. Overview of the Offline Pattern Recognition. The system extracts subsequences from positional time series data and clusters the subsequences to calculate reference positional and information signals

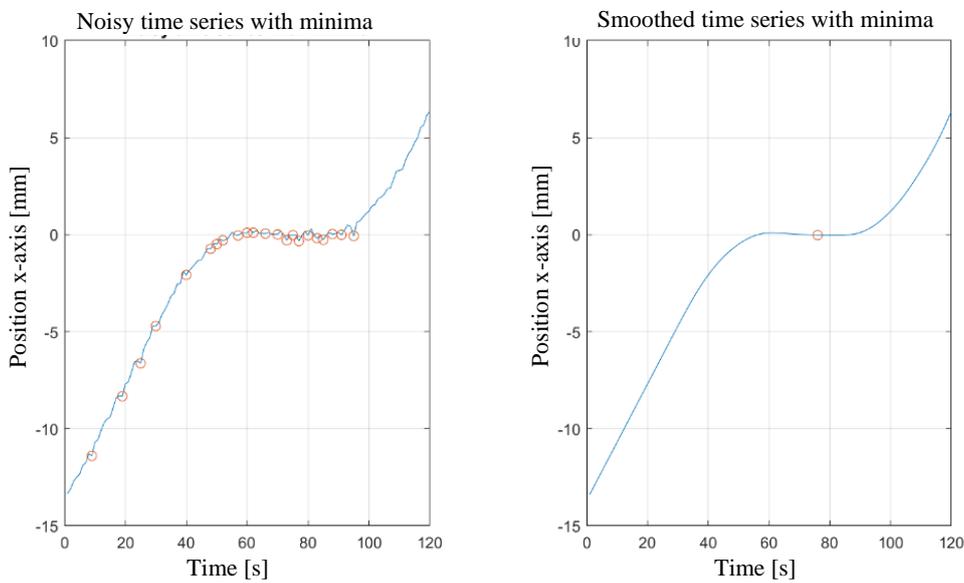


Fig. 2. Noise making the search for local minima impossible. On the left, local minima found when not accounting for noise are shown, on the right, the desired local minimum on which the time series will be split when correcting for noise is shown. Red circles denote a minimum or extremum

By applying smoothing, noise is eliminated, and the data is segmented using the defined criterion. As the time series that is split into patterns may not be one-dimensional (e.g., if the machine moves along multiple axes, usually along the x-, y- and z-axis), a dominant axis will be selected, based on which the split into subsequences follows. This

creates multiple different subsequences if positional movement along the non-dominant axes differs. Once the time series is split into these subsequences, a clustering approach is applied. During clustering, sparse patterns are eliminated while dominant patterns that occur often are saved for matching them with the respective information signals, essentially creating labels that can then be used for detecting anomalies in this sensor data.

For the clustering, a mean shift approach is used. To enable full comparability of the identified subsequences, a position offset adjustment is necessary, which is achieved via a discrete Fourier transformation (DFT) of the subsequences whereby the offsets are removed, and the representation can be transformed back to a position-time signal. Afterwards the point-to-point distance between subsequences is calculated using a length relative variation of the Euclidean distance. The resulting $n \times n$ distance matrix then acts as input for the mean shift cluster algorithm [11]. The output generated by the clustering contains representative subsequences that are matched to timestamps in the original input time series.

When new data is generated by a machine for which such representative sequences were calculated, the information signals that accompany the new data can be compared to prior signals that were observed during an earlier training cycle.

For this goal, an online recognition of the representative patterns is needed. This approach is called Online Cycle Detection (subsequences in the online data are also called cycles from now on, as they often define recurring patterns). Fig. 3 gives an overview of this system.

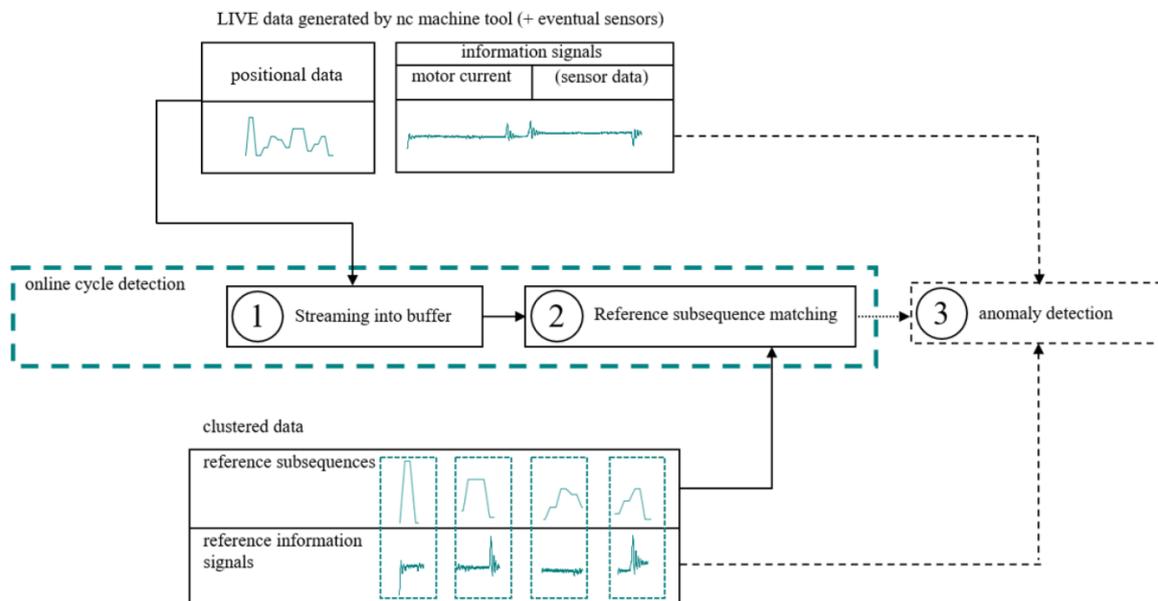


Fig. 3. Overview of the Online Cycle Detection. The system recognizes previous subsequences that were stored by the Offline Pattern Recognition system. This can then act as the basis for anomaly detection, as live and reference information signals can be compared

The Online Cycle Detection task is finding the most probable subsequence that was identified by the Offline Pattern Recognition for any subsequence that is found in the online

data. Hereby it has to deal with data of high volume and velocity in the process, which makes a streaming approach necessary.

As problems regarding main memory size and computation times can arise, a FIFO-buffer structure with a manually defined length handles the incoming data signals. This signal buffer B is a set of signal data sequences x^s starting at t_1 and ending at t_2 .

The Online Cycle Detections task is to identify and localize the right representative subsequence from the set of representatives S , when a new signal subsequence is detected inside the buffer. This corresponds to the following set:

$$\{s | s \in S \text{ and } s \in B\} \quad (1)$$

As the signal buffer can contain multiple signals, the subsequences identified by the Offline Pattern Recognition have to be matched to the signal buffer. Therefore, both identification and localization have to be considered. Identification includes finding all representatives s that could appear in the buffer, while localization describes finding their position in order to extract the information signals that are tracked in parallel. This is computationally expensive and requires tuning of the signal buffer size in order to reduce execution times, as all representatives have to be considered at different positions.

To localize the subsequence, the position of subsequence s inside the signal buffer B at time t has to be detected. This is done by finding the position p of subsequence s at which the mean absolute error between the signal x^s and the representative subsequence s is minimized:

$$\arg \min_{p \in [t_1, \dots, t_2]} \frac{1}{n} \sum_{i=t_1+p}^n |x_i^s - s_i| \text{ where } n = \min(t_1 + p + |s|, t_2) \quad (2)$$

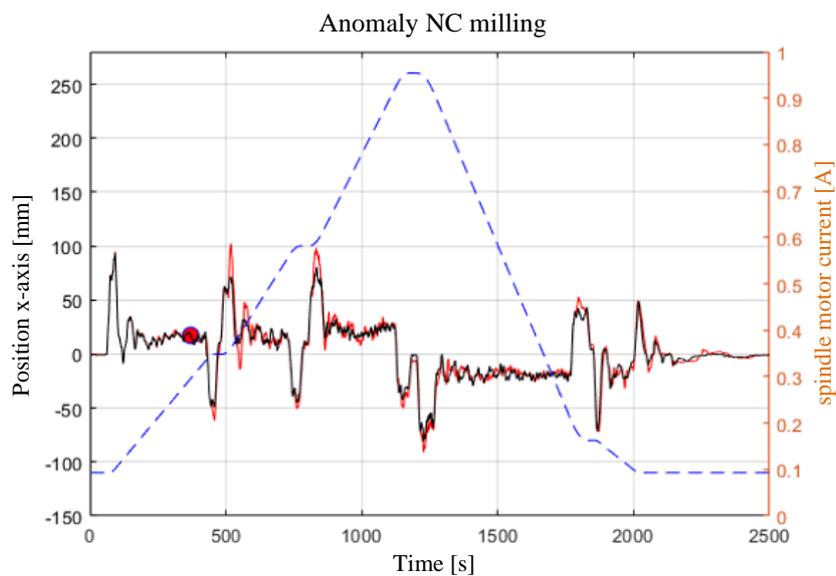


Fig. 4. Example of information signals for a positional subsequence (blue): the live signal (red) deviates from the reference information signal (black). On a global level, small deviations like these are difficult to detect. By calculating signal thresholds locally, anomaly detection can be applied more efficiently.

Problematic cases where only a limited number of sample points can be used to calculate the difference between online data and representative subsequence data occur as soon as new signals enter the signal buffer. In such cases, the distance measure only holds little information, because most of the signal is still unknown. To prevent misclassifications, a threshold that excludes signals with less than the defined amount of signal points is introduced.

After a subsequence is found in the online data, anomaly detection algorithms for condition monitoring can be applied traditionally. For this, the information signals of the streamed data can be compared to representative information signals of the matched representative subsequences (Fig. 4). As arbitrary information signals can be compared, the solution is compatible with any sensor that generates data at specific time stamps.

5. EXPERIMENTAL FUNCTION VALIDATION

The introduced approach for finding time series subsequences offline and recognizing these sequences online during machine operation can act as a standardized solution for applying anomaly detection based on the generated data. At wbk – Institute of Production Science in Karlsruhe, Germany, a test bench which is reduced to one feed axis was set up (illustration in Fig. 5). The feed axis consists of a standardized ball screw drive for milling machines and a steered drive.



Fig. 5. Reduced test bench of a milling machine, feed axis realized by ball screw drive with a steered drive

An OPC UA server was implemented (see Chapter 3) to acquire the needed data from the controller unit. To find the required data sources on the OPC UA server and to validate the described identification approach the developed algorithm was implemented with an OPC client. The time series data was recorded with a time resolution of 10 ms and features were extracted with sample times of 500 ms. A total of 163 test series were taken. Using the trained decision trees, the required parameters regardless of their parameter naming scheme could be identified successfully, as shown in Fig. 6. The motor current was detected most reliably with 100%, followed by the position signal with 97.1% and the torque with 96.7%,

both of which were also classified as motor current in individual cases. Based on the assignment, the corresponding node information of the selected parameters can be provided to the data segmentation application in form of an xml file.

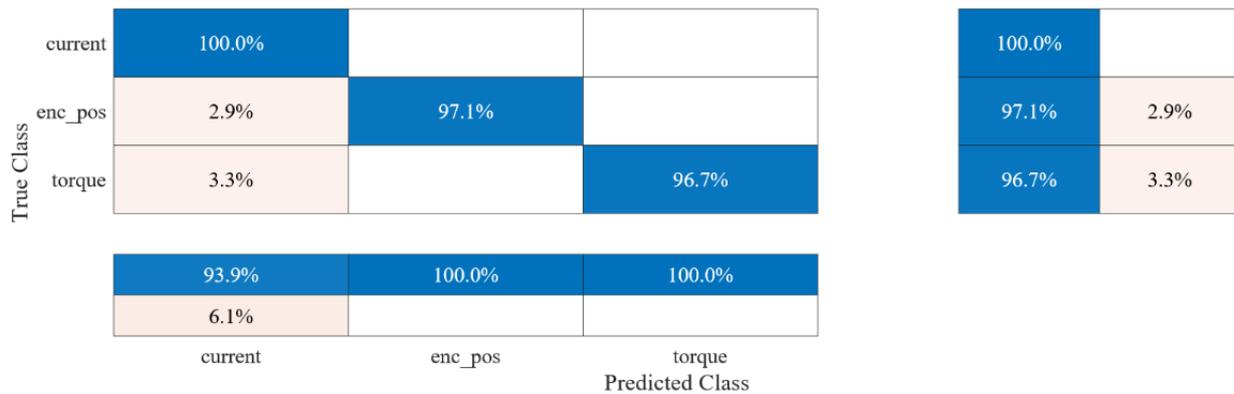


Fig. 6. Classification on presented test bench using binary decision trees

To validate the described approach of data segmentation, a normal milling process reduced to the x-axis is applied. In case of milling, different traverse movements characterized by different velocity and end positions are present. In a first run, the algorithm learns the cycle movements through the Offline Pattern Recognition. The results are shown in Fig. 7.

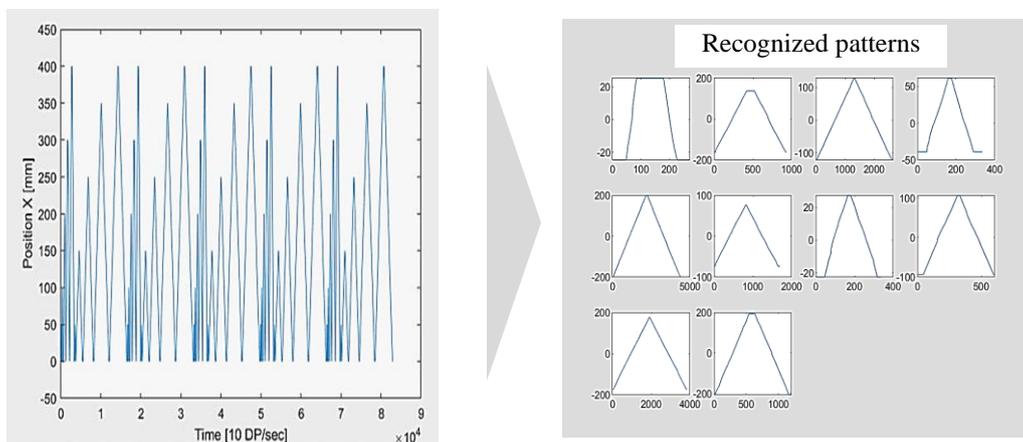


Fig. 7. On the left: input data time series of feed axis position measured through OPC UA. There are 10 different movements, simulated by 5 subsequent parts. On the right: result after the Offline Pattern Recognition step: there are 10 calculated patterns

As seen above, there are 10 different movements calculated by the clustering algorithm. These learned subsequences act as the input for the following Online Cycle Detection that aims to redetect these recurring movements. Fig. 8 shows the detected subsequences for the online buffer data. Based on the Offline Pattern Recognition the algorithm is able to find the recurring movements by calculating the matching score. On

the right side (blue diagram) the detected sequences are displayed. In the green diagram on the bottom, the information signals extracted from the controller unit (current, torque) are shown to the user. Based on this standardized method, condition monitoring solutions, e.g. anomaly detection in drive signals or data provisioned from external sensors can be applied to older machine tools. In order to detect these anomalies, deviations from the clustered subsequence can be detected using self-learned thresholds. When clusters are updated with more sequences recognized by the pattern recognition, the thresholds may change (as they could be based on different variables such as signal variance).

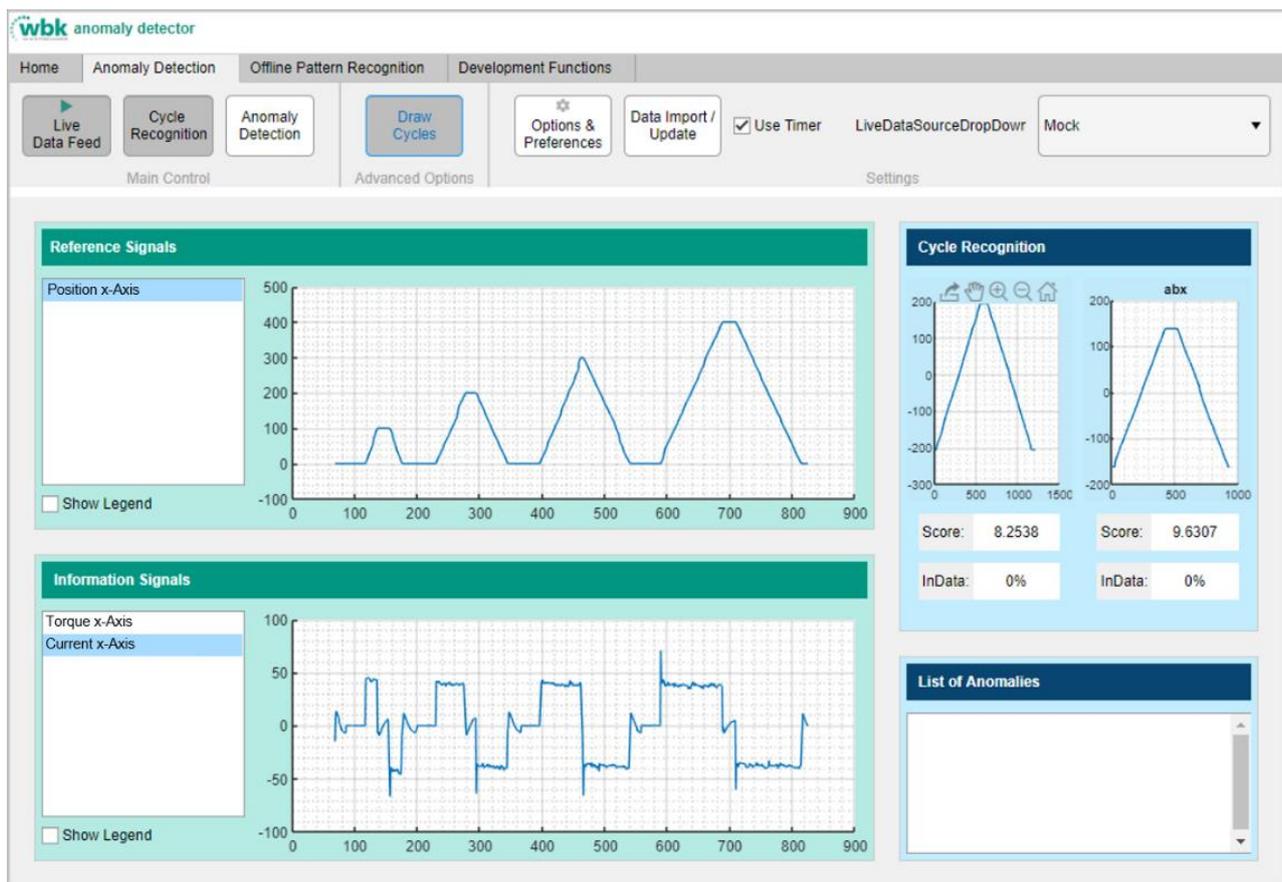


Fig. 8. Cycle Detection & Anomaly Detection GUI

6. CONCLUSION AND OUTLOOK

Using the described data acquisition, it is possible to provide and identify machine parameters from older machine tools and further accessible data sources. This serves as the basis for extracting useful data from any machine that generates time series and sensor data that can be matched to the attached timestamps. The presented approach focuses on two main components: An Offline Pattern Recognition which identifies representative subsequences and an Online Cycle Detection, which searches for the most probable representative subsequence once a subsequence (also called cycle in this instance) is found

inside a streaming buffer. This standardized method is applicable to most industrial machines where access to a time series that describes their behavior is possible. The generation of information signals that can be used in anomaly detection for condition monitoring can then be handled by accessory sensors, making the application highly modular. This modularity can be exploited by manufacturers, allowing them to evade expensive hardware swaps.

One area where further developments are needed is the improvement of the Online Cycle Recognitions execution time, as currently, expensive distance calculations between subsequences are still necessary. With an increase in calculation efficiency, it could be possible to introduce higher sampling rates of up to 500 kHz at lower latencies. Additionally, by focusing on edge or cloud computing, bottlenecks in data transmission could be avoided. Future investigations should also focus on the combination with newer standardized Industry 4.0 hardware in brownfield applications with positive cost-benefit relationships.

REFERENCES

- [1] ISMAIL A., TRUONG H.L., KASTNER W., 2018, *Manufacturing Process Data Analysis Pipelines: a Requirements Analysis and Survey*, Journal of Big Data, 6, 1–26.
- [2] GITTNER T., GONTARZ A., WEISS L., WEGENER K., 2019, *A Fundamental Approach for Data Acquisition on Machine Tools as Enabler for Analytical Industrie 4.0 Applications*. Procedia CIRP, 79, 586–591, DOI: 10.1016/j.procir.2019.02.088.
- [3] SOBEL W., 2014, *MTCConnect Standard*, MTCConnect Institute, Online available <https://github.com/mtconnect/standard>.
- [4] BEN E., BINGYAN Z., HANSEL A., MASAHIKO M., FUJISHIMA M., 2014, *Machine Monitoring System Based on MTCConnect Technology*, Procedia CIRP, 22, 92–97, DOI: 10.1016/j.procir.2014.07.148.
- [5] LEE J., KAO H.A., YANG S., 2014, *Service Innovation and Smart Analytics for Industry 4.0 and Big Data Environment*, Procedia CIRP, 16, 3–8, DOI: 10.1016/j.procir.2014.02.001.
- [6] LIN J., KEOGH E., LONARDI S., PATEL P., 2002, *Finding Motifs in Time Series*, Proceedings of the Second Workshop on Temporal Data Mining, 53–68.
- [7] KEOGH E., LIN J., 2005, *Clustering of Time-Series Subsequences is Meaningless: Implications for Previous and Future Research*, Knowl. Inf. Syst., 8/2, 154–177, DOI: 10.1007/s10115-004-0172-7.
- [8] SAKURAI Y., FALOUTSOS Ch., YAMAMURO M., 2007, *Stream Monitoring Under the Time Warping Distance*, IEEE 23rd International Conference on Data Engineering, Istanbul, 1046–1055.
- [9] EMEC S., KRÜGER J., SELIGER G., 2016, *Online Fault-monitoring in Machine Tools Based on Energy Consumption Analysis and Non-Invasive Data Acquisition for Improved Resource-Efficiency*, Procedia CIRP, 40, 236–243, DOI: 10.1016/j.procir.2016.01.111.
- [10] TANI GmbH Networks for Industry, Copyright 2013–2019, Tani GmbH, Nürnberg, Deutschland, Online available <https://www.tanindustrie.de/de/index.php>.
- [11] NETZER M., MICHELBERGER J., FLEISCHER J., 2020, *Intelligent Anomaly Detection of Machine Tools Based on Mean Shift Clustering*, Procedia CIRP, 93, 1448–1453.