Manja Mai-Ly PFAFF[1*]
Uwe FRIEß[1]
Andreas OTTO[1]
Matthias PUTZ[1]

PREDICTIVE TECHNOLOGY ASSESSMENT
BY MEANS OF A STRUCTURE-BASED METHOD OF MACHINE LEARNING

From a user perspective, the current development of the generic term Industry 4.0 increasingly moves its orientation towards flexible production. Due to increasingly variable products with small quantities and the resulting high degree of adaptability of a plant over its entire operating phase, the need for rapid production commissioning gives rise to the demand for live commissioning support and technology evaluation of induced production start-ups. Classification axioms can be formed by 1-class learning procedures for the predictive state evaluation of subsequent production start-ups based on collected machine and process data from past production start-ups. The starting point is an adaptive algorithm that performs a dynamic tolerance band formation based on different criteria, emphasizing on adaptive characteristic segmentation. This first step represents comprehensive condition monitoring. Based on this algorithm, correlation considerations can be performed on the data structure, the measured variables, and the diagnostic parameters. Moreover, the structure of production systems can and should be included in the analyzation, so that probabilistic causalities can be postulated and then be added to the underlying data sets for quantification. Using these adaptive structure-based segmentations is the first step to interpret data sets of new production systems without the need for complex pre-configuration.

1. INTRODUCTION

The generic term Industry 4.0 is characterized by buzzwords such as customized products, Big Data applications, and the assisted support of human beings as a creative decision-makers through cyber physical systems [1]. The applications in the industrial areas are manifold and often result from the growing demands on the industry to ensure competitiveness. Users focus more and more on flexible production and, as a result, demand adaptive and (if possible real-time) maintenance based on machine condition [2].

The intention is to identify potential ad hoc failures in good time and reduce maintenance measures to a minimum. In order to evaluate the condition of a machine,

an evaluation basis is required, which is normally a complex comparative database. The demand for adaptive maintenance is growing given increasingly complex products in small quantities.

High plant availability is also of importance not only for small and medium-sized companies, which may only have a small number of production plants [3], but also for the complex production of large-scale plants such as shipbuilding plants [4] and because of the increasing importance of maintenance as a service itself [5]. The growing digitalization and progress in information and communication technology make it possible to obtain an ever-larger volume of data – a field with new, digital business models [6].

The following requirements are elementary in maintenance: plant availability, reliability, or reduction of maintenance costs, among other things. The inherent potential here must be tapped by exploiting the new opportunities that arise during the development [7].

For process-parallel condition assessment, essential influencing variables must be defined, analyzed, and classified. This requires algorithmic adaptive models for the (preferably real-time) processing and information acquisition of the underlying characteristic values. This work aims at developing the basis of a fully comprehensive application that analyzes, classifies, and evaluates (new) technologies autonomously (with the meaning of machine learning) without the need of a complex database for comparison. Rather, the state is derived from "good" data, i. e., data from error-free machine operations.

The procedure can be divided into three parts: data pre-processing, adaptive segmentation, and parameter interpretation. First, the data must be extracted and brought into a compatible condition with the application (data pre-processing). Subsequently, the data-based state evaluation takes place in which the first step is to subdivide the data into parts/phases according to their curve structure (adaptive segmentation). In the following step, the data is analyzed in a phase-related manner (parameter interpretation).

As the aim is to build a comprehensive application, this evaluation should apply to different data curves, without defining the phases manually. This paper will describe the second part mentioned previously: the adaptive subdivision/segmentation of data into parts, which continuously show a similar characteristic profile, curve over curve. The software environment MATLAB was used for this purpose.

2. CYCLICAL PROCESSES

Mechatronic components and systems already provide a large amount of data. Current developments increasingly strive to gain information from large datasets [8]. However, individual values are not very significant. Therefore, several data values from the same action must be provided in order to examine them for correlations or irregularities.

The following consideration is based on cyclical processes, recurring processing steps/operations, which usually characterize a production system. At different processing stations, specific work processes are performed in a determined rhythm, leading to characteristic data curves, marked by concise extremes or (almost) constant level, among other things.

However, it is the overall machining of a component during one rotation that represents a cyclic process and the individual work processes, including generally cyclic activities. In order to investigate more precisely whether anomalies occur in the overall machining process, which could indicate a machine failure, it is recommended to compare the individual areas with their specific behavior. Not only could cyclic phase-limited faults be detected, but also wear processes caused by specific machine operations could be identified.

A phase is defined as a section in a data graph with a characteristic cyclic curve in the following chapters. Figure 1 shows the average motor current curve of a processing station during the full rotation of the turntable of a rotary transfer machine. Various machining operations such as countersinking, crimping, milling or drilling operations can be carried out at the processing stations on this type of machine [9], which leads to individual characteristic curves.
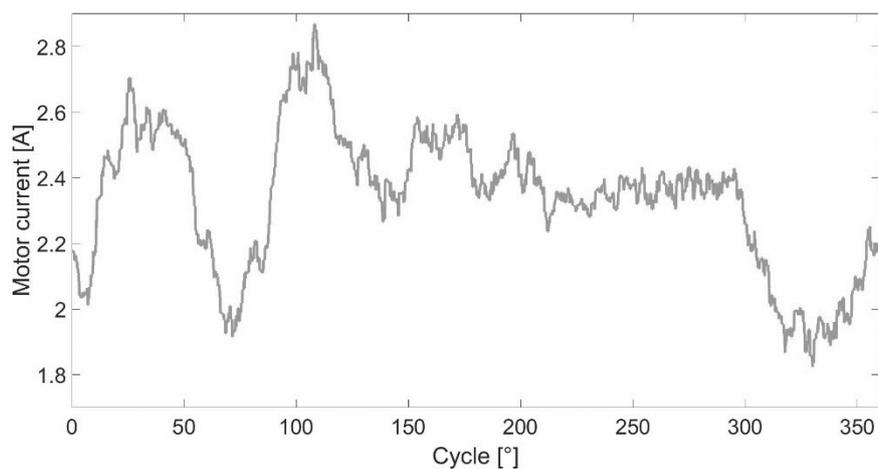
Fig. 1. Motor current curve of the main drive of a rotary transfer machine

Figure 1 shows areas that have increasing, decreasing, or steady characters. Usually, cyclic processes are now examined for analytical-technical relationships in order to subdivide the data curve meaningfully, as shown in Fig. 2. However, an agile, continually changing production environment would require constantly new parameterization. Since each parameterization requires complex detailed knowledge of the production process [e.g., 10, 11], this would be an excellent task given current developments (e.g., flexible production). Moreover, the time required for each parameterization [12] also needs to be considered and the feasibility.

This gives rise to the demand for a transferable application for monitoring the essential functions and the key components. The challenge is to implement an algorithm that adapts to the changing production environment, including changing machines and changing processes.

As illustrated in Fig. 2 (upper part), the subdivision of the data curve correlates with the data structure: increasing character in Section 1, 3, 5 and 9, decreasing character in Section 2, 4, 6 and 8, and (nearly) steady character in Section 7.
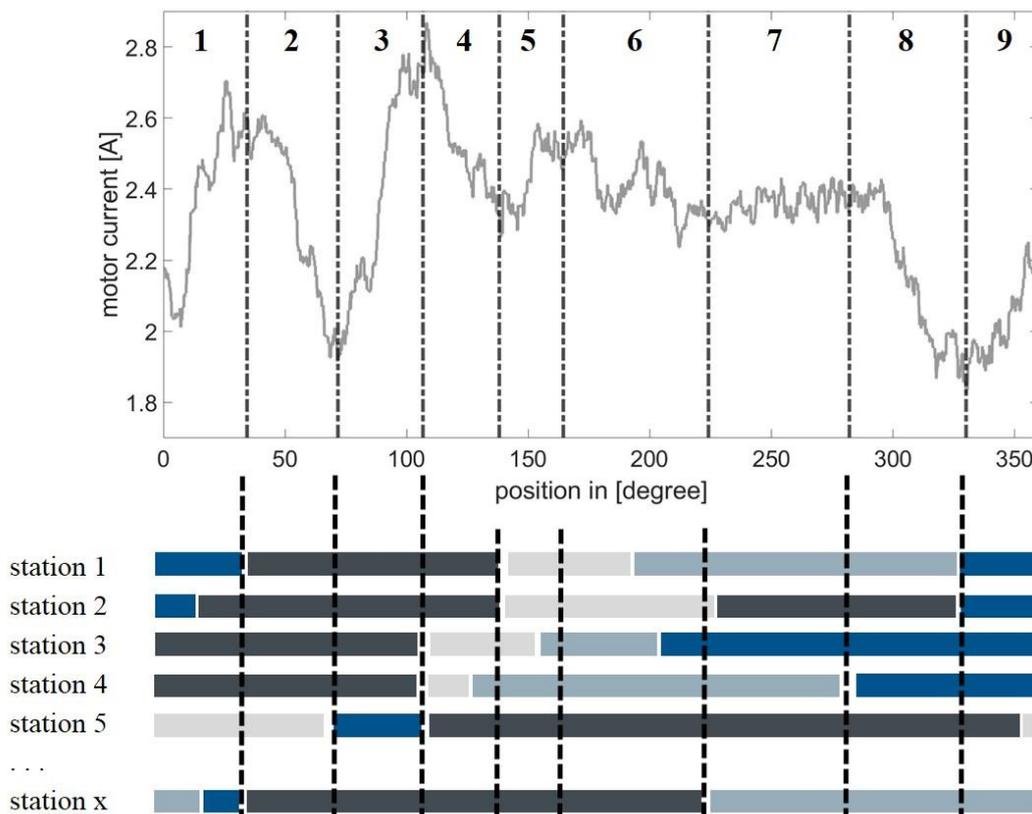
Fig. 2. Derived phase boundaries

Instead of the segmentation based on the mechanical effects chain, the incentive is to automatically subdivide the signals based on the data structure, as described in the following Section 3. Thus, it would not be necessary for the machine operators themselves to continually change the parameters. Instead, the characteristic sections in the data curve of changing production starts are identified automatically. Based on the segmentation, a phase-specific analysis can then be performed, using e.g., statistical calculation methods such as Gaussian normal distribution.

A further advantage lies in the fact that even complex characteristic values (e.g., a sufficient vibration level) can be evaluated. Technical interdependencies are not always directly apparent or the data progression may be "shifted" to the executed processes (e.g., due to elasticities). Nevertheless, it is important to analyze the machine's long-term behavior to answer questions such as the following: Does the machine still have the same performance as it did yesterday at this time? Even if the cause of irregularities is not immediately apparent at first, phase-specific analyses can provide information about the concrete processing procedures. There are different components running during the machining operations in the individual phases. Therefore, it is necessary to narrow down the irregularities to those phases affected in order to react cause-related without the need for component-specific sensors.

Thus, the overall idea consists of data-based adaptive interpretation without pre-estimations and pre-definitions of hard limits.

3. ADAPTIVE DATA/SIGNAL SEGMENTATION APPROACH

1-Data smoothing

Although the data in the phases shown in Fig. 3 generally show a characteristic course structure (e.g., rising, falling), they are only discernible in the overall view. On closer inspection, the progression is a sequence of rising and falling straight lines between two data points. Therefore, for segmentation, it is necessary to indicate the basic structure of the data curve.
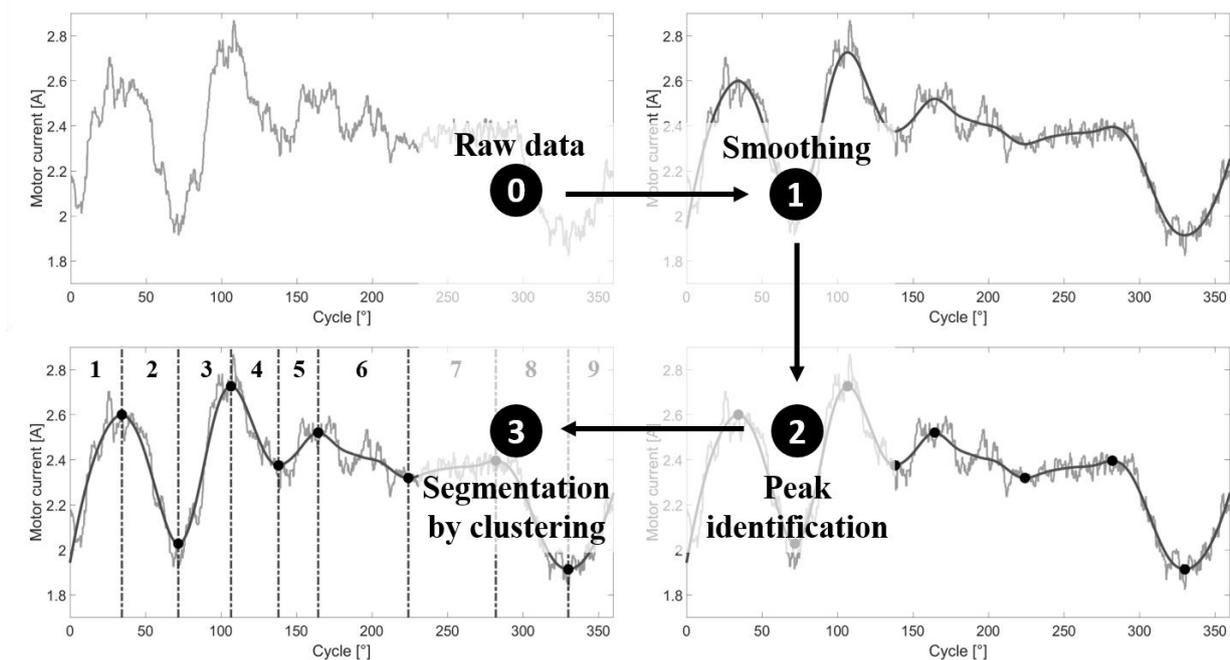
Fig. 3. Overall processing of the characteristic value
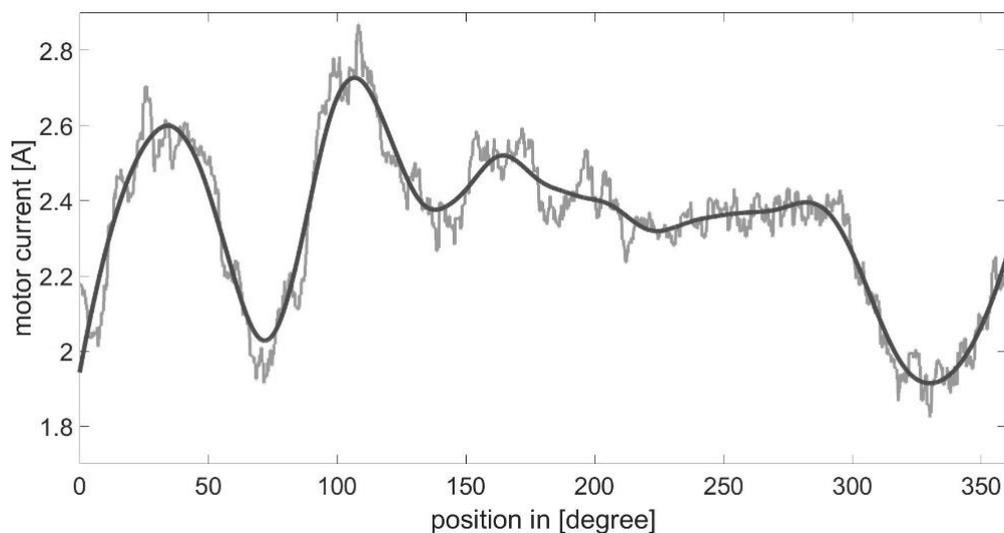
Fig. 4. Smoothed data curve

For this reason, the data was filtered. In MATLAB, there are different filter methods such as *Savitzky-Golay*, *lowpass,* or *local regression (including rloess). Rloess* was chosen for this work. This is a local regression that applies the least-squares method and the 2nd-degree polynomial, which gives outliers a lower weighting, thus weighting values outside the 3 boundaries with 0.

Figure 4 shows the smoothing of the motor current curve data from Fig. 1. It is evident that the extremes of the smoothed data curve characterize the start and the end of a phase.

2-Determine segmentation relevant extremes

The extreme values are then determined based on the smoothed characteristic curve. MATLAB provides the *findpeaks*-function in the Signal Processing Toolbox. This enables the determination of x- and y-coordinates, and dominance and prominence (further details see below).

The *findpeaks*-function is used for maxima. The same function can also be applied to minimum values by mirroring the motor current curve on the x-axis. Subsequently, it is necessary to reverse the y-values (multiplication by minus one) [13, 14].

The fundamental idea of the adaptive segmentation is to perform clustering based on phase-marking peaks, thus determining the phase boundaries (= cluster centers). Peaks result from changes in the monotony of the data curve, even slight changes (see Fig. 5). However, smaller peaks distort the cluster result. Therefore, it is required to select non-phase-marking peaks.
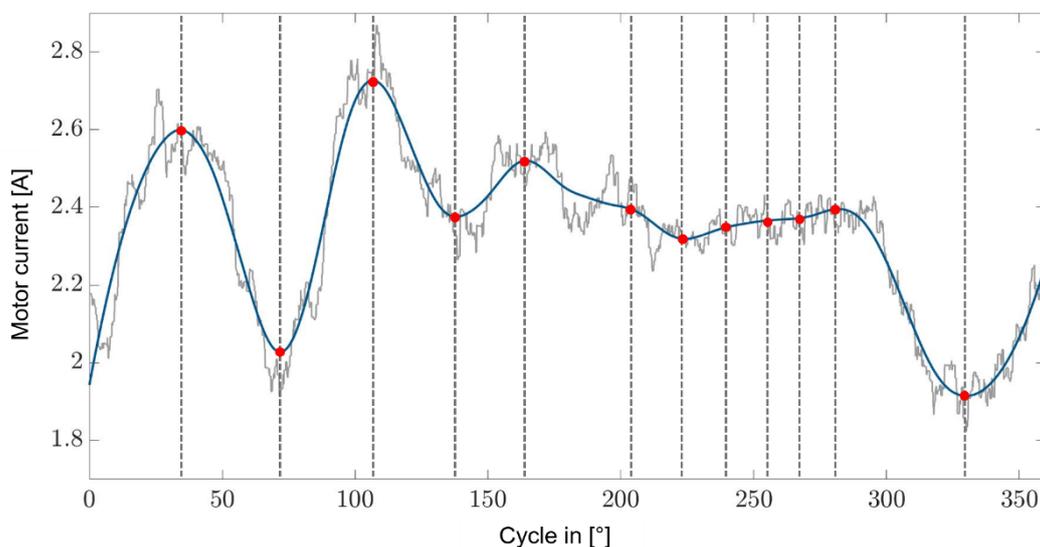
Fig. 5. Possible phase boundaries without selection of non-phase-marking peaks

A similar topic has been applied to determine phase-marking peaks: the categorization of mountains, which uses the "dominance" and "prominence" of a peak. "Dominance" indicates the distance to the next higher elevation. Therefore, mountains are dominant since they are higher than others in their surroundings [15]. "Prominence" is also known as the difference between the ridge and the reference saddle, the lowest point between this peak

and the higher one [16]. Therefore, the prominence of the highest mountain of a continent would be equal to its own height [15]. Figure 6 illustrates an example of a peak's prominence and dominance.
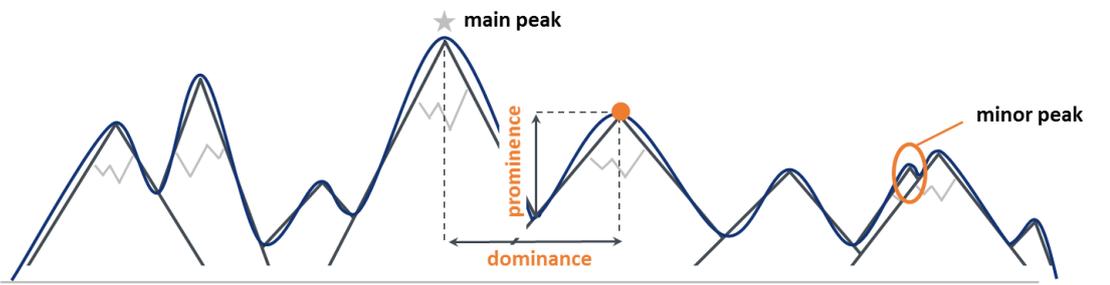
Fig. 6. Prominence and dominance of a peak

The classification of mountains differs according to literature and country. The approach to the selection of peaks found here goes back to [17] in which it calculates the summit autonomy based on three factors, see Table 1. The third factor, the altitude, however, is not transferable to this context. Not only is the reference axis missing for determining minima, but it would also not be possible to categorize them in the approach of a fully comprehensive application (e.g., peaks from a filtered data course with generally high y-values would tend to be categorized higher).

Table 1. Calculating the indices for peak selection

Factor	Index referred to [17]	Index for selection of peaks
Prominence	$E_P = -log_2\left(\dfrac{relative\ prominence}{100}\right)$	$E\_Peak_P = -log_2\left(\dfrac{prominence\ [A]}{max.prominence\ [A]}\right)$
Dominance	$E_D = -log_2\left(\dfrac{dominance\ [m]}{100,000}\right)$	$E\_Peak_D = -log_2\left(\dfrac{dominance\ [°]}{max.prominence\ [°]}\right)$
Altitude	$E_H = -log_2\left(\dfrac{altitude\ peak\ [m]}{8,848}\right)$	-

Similarly, a prominence concerning one's own height shall be considered. This is reasonable if, theoretically, the data value is subject to extreme fluctuations. Otherwise, relative values with relatively small fluctuations in the numerator and denominator would have no significance. In nature, mountains can be as high as the Himalaya or can have a reference saddle at sea level, meaning that the data value could be unusually large or even 0. Consequently, a prominence relative to one's own altitude may make sense regarding the classification of mountains, but not necessarily for the classification of phase-marking peaks on which this analysis is based. Furthermore, it would be questionable to what extent the comparability of maxima and minima would be given if minima were set in

relation to their own height. For this reason, the prominence of a peak was compared with the maximum prominence over the entire period (in this case the unit is given in [A]).

Leonhard W. [17] determines the dominance factor using the ratio of the respective prominence value to 100,000 km. Once again, it is questionable to what extent such a fixed value can be defined in the sense of a fully comprehensive application. For this reason, the two factors were set concerning the respective maximum value throughout the entire period (here 360°).

Summits can have a high prominence with low dominance or vice versa, but according to [17], these two values have a more significant impact the lower the summit is. Furthermore, significant peaks (characterized by high prominence and dominance value) would never occur together with a high burr and low saddle. In conclusion, a phase-marking peak would have both the highest possible value of prominence and dominance.

Based on the prominence and the dominance of a peak, phase-marking could be distinguished from non-phase-marking peaks.

3-Determine number of clusters for k-means clustering

These phase-marking peaks can then be clustered, followed by the determination of the phase boundaries. This is the principal idea. The data is clustered according to the k-means approach (for advantages of this method, see below). The number of clusters must be specified. Despite the preceding selection of phase-marking peaks, the number of identified peaks can deviate over several data curves. So, the number of peaks over a data curve cannot be defined as the number of clusters ($\triangleq$ number of phases). For clustering according to the k-means principle, the number of clusters must be specified. There is no specific solution concept [18], but several approaches exist, such as the elbow method [19], gap method, or silhouette method [20, 21]. All these methods intend to find the number of clusters where adding another one does not improve data modeling (regarding the original purpose/intention) [22]. For the presented case, the silhouette approach has been used and extended. The silhouette coefficient is a measure of how well a value fits into the cluster assigned to it (compared to others) and is defined as follows [23].

3.1. SILHOUETTE COEFFICIENT

$$s(i) = \frac{b(i) - w(i)}{max\{b(i), w(i)\}} \tag{1}$$

where:
    s(i) – silhouette coefficient of object *i*,
    b(i) – average distance of object *i* to objects of the nearest cluster,
    w(i) – average distance of object *i* to objects of the same cluster.

The silhouette value is maximum, and at best 1, for objects (in the following also called data nodes) equivalent to the cluster center. A value of 0 indicates that data points cannot be uniquely assigned to either one or another cluster.

If the number of clusters is determined solely based on the silhouette coefficient, the "fit" of these data points to the cluster is taken into account, but not the data quantity (and in this case, peaks) that has been assigned to the clusters (in this paper also defined as

cluster size). This may be irrelevant for some applications, but not for this one. The extreme values on which the clustering is based are distributed equally over all cycles considered, when possible, and only then can they be regarded as representative phase boundaries. Thus, it is not sufficient to merely consider the "fitting accuracy" without taking into account an even distribution of the values over the clusters. Clusters that differ significantly in their size would indicate a grouping of phase segments – assuming that the number of peaks per cluster is almost equal.

For this reason, not only the silhouette coefficient is considered, but also the number of peaks per cluster. Four indices are set up. If possible, I_silhouette, I_cluster$_{min}$, and I_cluster$_{max}$ are equal to 1 and I_width equal to 0; see Eq. (2) to (4) below. The first index value indicates the average "accuracy of fit" of all data points to their clusters; see Eq. (2).

3.2. INDEX I_SILHOUETTE

$$I\_silhouette = \frac{1}{I}\sum_{i=1}^{I} \frac{b(i)-w(i)}{max\{b(i),w(i)\}} \tag{2}$$

where:

I – total number of objects

$b(i)$ – average distance of object i to objects of the nearest cluster

$w(i)$ – average distance of object i to objects of the same cluster

As previously mentioned, ideally, peaks are equally significant across all curves. This would result in an equal number of data nodes per cluster. A median value can be derived from the number of objects per cluster. The number of values assigned to the "smallest cluster" is set in relation to the median, see Eq. (3).

3.3. INDEX I_CLUSTER$_{MIN}$

$$I\_cluster_{min} = \frac{num_{min}}{num_{med}} \tag{3}$$

where:

num_{min} – number of objects within a cluster with the minimum number of objects over all clusters,

num_{med} – median number of objects over all clusters.

Assuming analogy with the previous index, the "largest cluster" can also be related to the median value, see Eq. (4).

3.4. INDEX I_CLUSTER$_{MAX}$

$$I\_cluster_{max} = \frac{num_{med}}{num_{max}} \tag{4}$$

where:

num_{med} – median number of objects over all clusters,

num_{max} – number of objects within a cluster with the maximum number of objects over all clusters.

This index is an indicator of the maximum deviation of objects within a cluster. For each cluster, the difference is formed from its highest and lowest silhouette coefficient. It can be assumed that a cluster's objects are usually close together (and thus an index value of 0 at best). If this is not the case, it is characterized by a high index value (maximum 1).

3.5. INDEX I_WIDTH

$$I\_cluster = \frac{1}{C}\sum_{c=1}^{C} S_{max}(c) - S_{min}(c) \tag{5}$$

where:

C – number of clusters,

$S_{max}(c)$ – maximum silhouette coefficient of cluster c,

$S_{min}(c)$ – minimum silhouette coefficient of cluster c.

Note: As it is the intention to find the delta between the highest and the lowest silhouette coefficient within a cluster, this equation has to be adjusted for negative coefficients (by switching the position of $S_{max}(c)$ and $S_{min}(c)$).

A total index is then calculated.

3.6. TOTAL INDEX

$$Index = I\_silhouette + I\_cluster_{min} + I\_cluster_{max} - I\_cluster \tag{6}$$

where:

$I\_silhouette$ – average silhouette coefficient over all objects,

$I\_cluster_{min}$ – cluster size, relation "smallest cluster" to the median cluster,

$I\_cluster_{max}$ – cluster size, relation "largest cluster" to the median cluster,

I_width – average maximum delta of the silhouette coefficient within the clusters.

Within a loop, by clustering and steadily increasing the number of clusters, it can then be determined how many clusters the phase-marking peaks should be assigned to groups of (almost) the same size.

4. CLUSTERING USING THE K-MEANS APPROACH

The purpose of clustering, finding a structure within a data set, is ambiguous from a mathematical point of view, and can essentially be attributed to the following characteristics: disjunctive, overlapping, or fuzzy clustering. For the last two characteristics, multiple allocation/overlapping is possible. For fuzzy clustering, e.g., the data node is measured by cluster similarity given the probability with which the data nodes can be assigned to clusters [24].

Although probability values about the cluster classification are informative, in the end, there must be specific phase boundaries (≜ specific cluster classification). Furthermore, in comparison with other approaches, clustering using the k-means principle is a suitable

method for large amounts of data [25]. Other studies, especially on the segmentation of large data sets, are also based on the k-means approach [26–28]. Clustering based on the k-means approach was chosen, considering a possible real-time application and unambiguous assignment requirement.

Based on the previously determined number of clusters, the data was clustered according to the chosen approach. The cluster centers were then defined as phase boundaries. Figure 7 shows the positions of the phase-marking peaks (the peaks after the filtering of non-phase-marking peaks, step 2) in degrees over 112 data curves. There are 8 phases. The phase boundaries are shown as dotted lines.

After the first two phases mentioned at the beginning (data pre-processing and adaptive segmentation), a characteristic parameter interpretation can now be carried out. Within the determined phases, probabilistic prognoses can be formulated, see [29]. Furthermore, the data for the state evaluation can be analyzed for causalities.
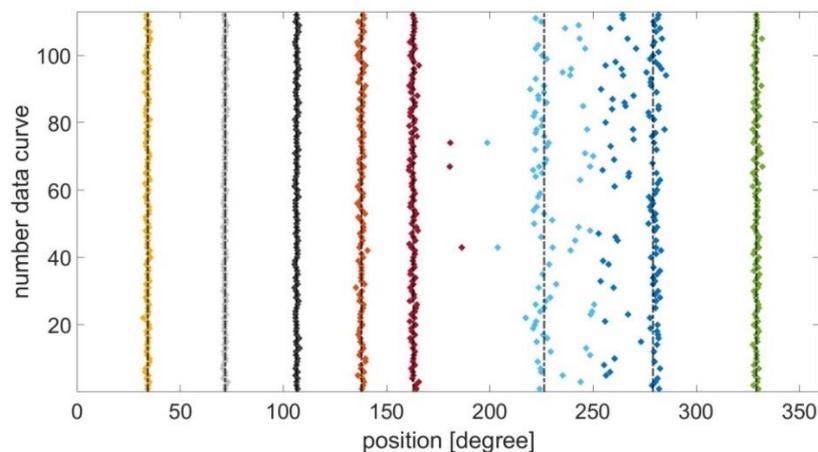
Fig. 7. Phase-marking peaks over 112 data curves

5. INTERPRETATION

Numerous applications are based on the same few characteristic values, such as the motor current or the displacement difference of measuring systems. This approach described here is based on auto-segmentation and the statistical evaluation of adaptive dynamic tolerance bands. As already mentioned, due to an increasingly flexible production environment, it is crucial to build an application that can analyze data adaptively. With constantly changing productions, it is not possible to permanently build up a new comparative database. In addition, there is usually no empirical knowledge. It is also challenging to define analysis parameters if no empirical values are given.

Therefore, the aim was to create an application for automatically segmenting the data progression before phase-specific analysis could be carried out in the next step. In order to trace irregularities back to the causal machining components, it is necessary to detect affected sections in the data curve. Ultimately, the data flow areas indicate which operations are performed, thus pointing to the cause of any irregularity in the system.

However, by considering each phase separately and combining specific phases, the machine operator's fault detection becomes much more straightforward and easier. While typical machine operators may not be able to pre-define the exact position in degrees a specific operation is proceeded, they often can "basically" distinguish operations if they are provided with demonstrative data as depicted in Fig. 8.
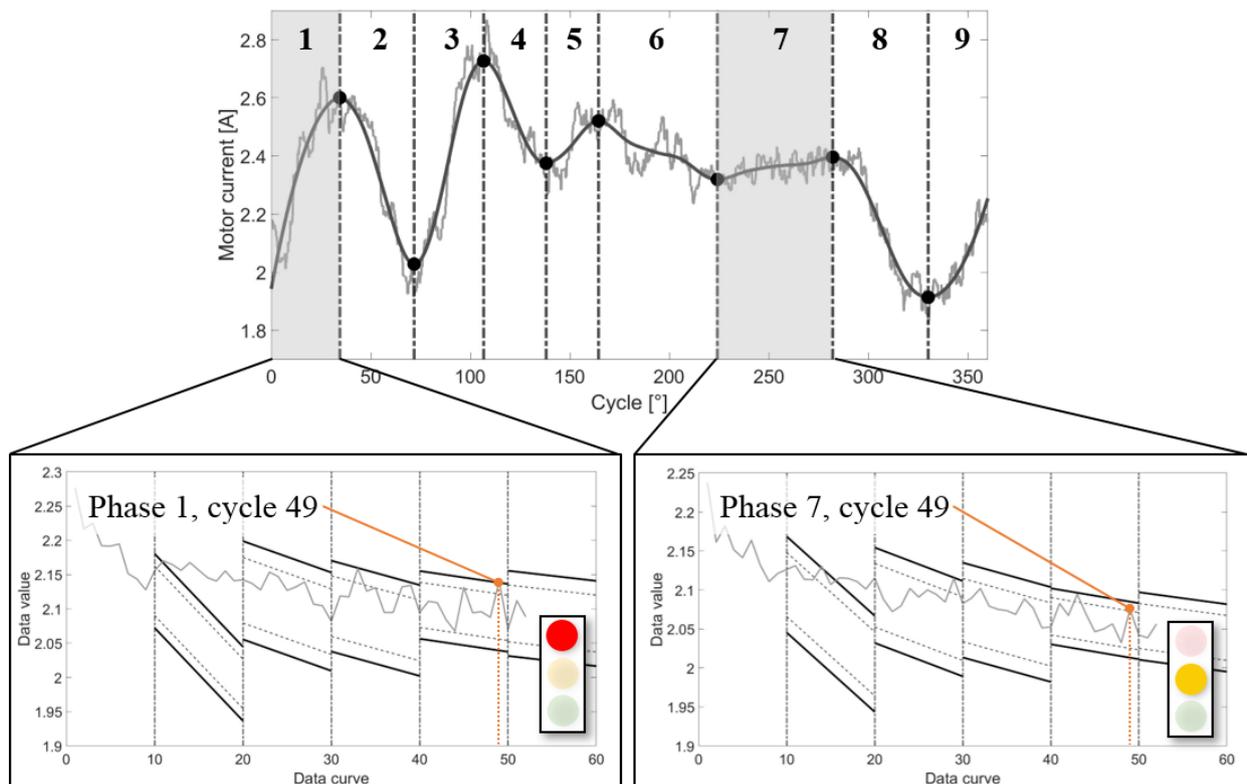
Fig. 8. Phase-marking peaks over 52 data curves

Based on the adaptive phase segmentation, a trend-interpretation of every single phase is proceeded based on principle statistic approaches such as forming an adaptive dynamic tolerance band based on the normal distribution. The trend-interpretation applied is classical fingerprint based on statistics.

Using this approach, the data is analyzed immediately without any prior training phase. This is advantageous since no faulty machine states have to be forced to obtain representative amounts of data (which is often difficult in practice). Subsequently, a dynamic tolerance band (see bottom part of Fig. 8) is formed, based on "good" values. This means that an evaluation is performed using the last error-free (and therefore "good" data) machine states. The underlying principle of dynamic tolerance band formation is as follows: iteratively calculating the 2- and 3-σ limits based on a certain number of last data curves and applying them as tolerance limits to future data. In this way, data from new production start-ups (whether due to new machines or a new production) can be analyzed right from the start, and the machine condition can be evaluated.

6. CONCLUSION

Today, there are already numerous applications that intend to gain information from a large dataset. Many of them are based on a comparison database. However, given increasingly flexible production environments, this is a prerequisite that can hardly be fulfilled.

The intention was to develop an adaptive algorithm that can be applied to various (and especially changing) productions and characteristic values that can hardly be evaluated due to complex technical effect correlations. The approach comprises two core aspects: adaptive phase identification and phase-specific analysis (using dynamic tolerance band formation). On the one hand, machine data can be evaluated directly from the start and without a previous training phase. On the other hand, there is no need to pre-analyze the machining steps in order to define phases. Instead, in the first step, the characteristic curve (in this paper, the motor current curve) was filtered. As a second step, phase-marking peaks were detected. Finally, these peaks were clustered according to the k-means approach to determine the phase boundaries. Within these phases, more specific analyses could then be performed in order to identify irregularities more quickly and in a cause-related manner. Using statistical methods, an adaptive dynamic tolerance band was formed, indicating critical machine data.

The adaptive phase segmentation presented here can form the basis for developing a flexible methodology that can be used to analyze, classify, and autonomously evaluate data. It would be necessary to subsequently apply this to more numerous and, above all, other data curves in order to confirm the reliable differentiation between smaller and phase-marking peaks.

As previously mentioned, further correlation considerations can be carried out based on adaptive phase identification, for example, between the measured variables, the diagnostic parameters, and the general conditions. Probabilistic causalities (cf. Bayesian networks, among other things) are to be assumed and used for data set interpretation of new production start-ups. Further research is required in the investigation of characteristic values relevant for predictions.

It is not only essential to avoid unforeseen breakdowns, but also to increase plant availability or to reduce maintenance measures. The methodology presented here provides a valuable contribution to self-sufficient, i.e. "operator-free", and adaptive maintenance. This could be important for production scenarios with generally low production volumes. Furthermore, against the background of current development efforts, the demand for flexible production is growing, increasing the need for adaptive maintenance.

It is also conceivable that failure is not only a consequence of wear and tear but the result of a weakness in the system. A different approach in the data-based state assessment of a machine could lie in applying the algorithms created for the cause-related analysis of potential ad hoc failures.

REFERENCES

[1] LIDONG W., GUANGHUI W., 2016, *Big Data in Cyber-Physical Systems, Digital Manufacturing and Industry 4.0*. IJEM 6/4, 1–8, https://doi.org/10.5815/ijem.2016.04.01.

[2] HIRSCH-KREINSEN H., ITTERMANN P., NIEHAUS J., 2018, *Digitalisierung Industrieller Arbeit, Die Vision Industrie 4.0 und Ihre Sozialen Herausforderungen* (2nd ed.), Nomos Verlagsgesellschaft, Baden-Baden, https:// doi.org/10.5771/9783845283340.

[3] FASSNACHT P., GRIGIN C., NICKEL R., 2009, *Prognostische Fähigkeiten*, QZ – Qualität und Zuverlässigkeit, 54, 58–59.

[4] BEHRENS B.A., OVERMEYER L., NYHUIS P., NICKEL R., 2009, XXL-*Produkte – ein Trend in der Produktionstechnik*, VDI-Z Integrierte Produktion, Springer VDI Verlag, 151, 56–58.

[5] EILERT B., KERSTIN D., OVERMEYER L., 2010, *Hier Gesucht Heißt Schon Gefunden, Erfahrungswissen Verbessert Servicequalität*, QZ – Qualität und Zuverlässigkeit, 55, 56–57.

[6] REICHEL I., 2008, *Maschine Detektiert Kritische Zustände vor Einem Ausfall*, VDI-Nachrichten, 62/35, 13.

[7] REICHEL J., MUELLER G., HAEFFS J., 2018, *Betriebliche Instandhaltung* (*2. Auflage*), Springer Verlag GmbH.

[8] *Digitalisierung kommt in den deutschen Unternehmen an* (2019), Bitkom – Bundesverband Informations wirts chaft, Telekommunikation und neue Medien e.V., https://www.bitkom.org/Presse/Presseinforma tion/Digitalisie-rung-kommt-den-deutschen-Unternehmen, (Retrieved December 11, 2019).

[9] MANFRED W., BRECHER C., 2018, *Werkzeugmaschinen Fertigungssysteme 1*, Springer-Verlag GmbH.

[10] SCHMIRGEL H., KRAH J,. 2008, *Automatische Parametrierung der Regelkreise von Servoreglern Über systematische Auswertung einer Frequenzanalyse*, Conference, PCIM 2007, Nuremberg, Germany.

[11] RUDOLF T.M., 2014, *Adapterbare Parametrierung von Diagnosesystemen Durch Verwendung Digitaler Antriebssignale in der Prozessüberwachung*, Aachen, Techn. Hochsch., Diss., Ergebnisse aus der Produktionstechnik, Aachen.

[12] JAEKEl J., SCHULZE R., RAUCHFUSS T., HUBER R, SCHOENEBERG F., 2014. *Virtuelle Inbetrieb-nahme von Verdichter- und Turbinensteuerung*, Conference, AUTOMATION 2014, Baden-Baden, Germany.

[13] SMOLÍK L., 20018, *Post-Processing Techniques for Response of Non-Linear Systems*, 20th International Conference Applied Mechanics, Applied Mechanics, 143–148.

[14] LI C., BI L., 2015, *Intelligent Recognition Method of Infant Sleeping Position Based on Thermal Infrared Imaging*, Inter. J. of Control and Automation, 8/7, 133–140, https://doi.org/10.14257/ijca.2015.8.7.14.

[15] PRANTL D., 2012, *Gipfelbuch*, Süddeutsche Zeitung, Goldmann Verlag im Netz.

[16] JURGALSKI E., 2016, *Das Erweiterte Prominenzkonzept. Ein mathematisches Einteilungssystem für alle Berge und Gebirge, Weltweit Anwendbar vom Hochgebirge bis zu Heimathügeln*, Mitteilungen der Fränkischen Geographischen Gesellschaft, 61/62, 105–110.

[17] LEONHARD W., 2019, *Eigenständigkeit von Gipfeln*, https://www.thehighrisepages.de/bergtouren/na_oro gr. htm, (retrieved 06.07.2019).

[18] KODINARIYQM T., MAKWANA P., 2013, *Review on Determining Number of Cluster in k-means Clus-tering*, Inter. J. of Advance Research in Computer Science and Management Studies, 1/6, 90–95.

[19] BHOLOWALIA P., KUMAR A., 2014, *EBK-means, A Clustering Technique Based on Elbow Method and k-means in WSN*, International Journal of Computer Applications, 105, 9.

[20] TIBSHIRANI R., WALTHER G., HASTIE T., 2001, *Estimating the Number of Clusters in a Data Set via the Gap Statistic*, Journal of the Royal Statistical Society, Series B (Statistical Methodology), 63/2, 411–423, https://doi.org/10.1111/1467-9868.00293.

[21] CHAKI N., CORTESI A., DEVARAKONDA N., (Eds.), 2018, *Proceedings of International Conference on Computational Intelligence and Data Engineering*, Springer Singapore, https://www.springer.com/gp/book/9789811364587.

[22] KINGRANI S.K., LEVENE M., ZHANG D., 2017, *Estimating the Number of Clusters Using Diversity*. AIR 7, 1, 15, https://doi.org/10.5430/air.v7n1p15.

[23] LLETI R., ORTIZ M.C., SARABIA L.A., SANCHEZ M.S., 2004, *Selecting Variables for k-means Cluster* Analysis by Using a Genetic Algorithm that Optimizes the silhouettes, Analytica Chimica Acta, 515/1, 87–100, https://doi.org/10.1016/j.aca.2003.12.020.

[24] KAPPES A., 2015, *Engineering Graph Clustering Algorithms,* Karlsruher Institut für Technologie (KIT).

[25] MAIMON O., ROKACH L., 2005, *Data Mining and Knowledge Discovery Handbook*, Springer-Verlag GmbH.

[26] CHEN K., LIU L., 2006, *iVIBRATE, Interactive Visualization-Based Framework for Clustering Large Datasets*, ACM Transactions on Information Systems, 24/2, 245–294, https://doi.org/10.1145/1148020.11 48024.

[27] BU Y., HOWE B., BALAZINSKA M., ERNST M.D., 2010, *HaLoop: Efficient Iterative Data Processing on Large Clusters*, Proceedings of the VLDB Endowment, 3/1-2, 285–296, https://doi.org/ 10.14778/19208 41.1920881.

[28] GANTI V., RAMAKRISHNAN R., GEHRKE J., POWELL A., FRENCH J., 1999, *Clustering Large Datasets in Arbitrary Metric Spaces, Proceedings 15th International Conference on Data Engineering* (Cat. No.99CB36337)*, IEEE, https://doi.org/10.1109/icde.1999.754966.

[29] FRIEß U., KOLOUCH M., FRIEDRICH A., ZANDER A., 2018, *Fuzzy-Clustering of Machine States for Condition Monitoring*, CIRP Journal of Manufacturing Science and Technology*, 23, 64–77. https://doi.org /10.1016/j.cirpj.2018.09.001.