Anomaly Detection for Hydraulic Systems under Test

1st Deniz Neufeld
Cognitive Systems Group
University of Bamberg
Bamberg, Germany
deniz.neufeld@uni-bamberg.de

2nd Ute Schmid
Cognitive Systems Group
University of Bamberg
Bamberg, Germany
ute.schmid@uni-bamberg.de

2021 IEEE 26th International Conference on Emerging Technologies and Factory Automation (ETFA) | 978-1-7281-2989-1/21/$31.00 ©2021 IEEE | DOI: 10.1109/ETFA45728.2021.9613265

***Abstract*—This work focuses on computationally efficient difference metrics of time series and compares two different unsupervised methods for anomaly classification. It takes place in the domain of hardware systems testing for reliability, where several structurally identical devices are tested at the same time with a load expected in their lifetime use. The devices perform different maneuvers in predefined testing cycles. It is possible that rare, unexpected system defects appear. They often show up in the measured data signals of the system, for example as a decrease in the output pressure of a pump. Due to the intended aging of the parts under load, the measured data also exhibits a concept drift, i.e. a shift in the data distribution. It is of interest to detect anomalous behavior as early as possible to reduce cost, save time and enable accurate root-cause-analysis. We formulate this problem as an anomaly detection task on periodic multivariate time series data. Experiments are evaluated using an open access hydraulic test bench data set by Helwig et al. [1]. The method's performance under concept drift is tested by simulating an aging system using the same data set. We find that Mean Squared Error towards the median in combination with the Modified z-Score is the most robust method for this use case. The solution can be applied from the beginning of a hardware testing cycle. The computations are intuitive to understand, and the classification results can be visualized for better interpretability and plausibility analysis.**

***Index Terms*—Hardware testing, anomaly detection, time series, signal processing, semi-supervised algorithms.**

I. Introduction

Reliability testing is an important tool in hardware development to build long-lasting systems. It means that the functionality over a lifetime of a device is verified using statistical methods [2]. It is part of research and development of new systems and essential in domains focussed on safety, like in automotive, aerospace or medical engineering. In hardware reliability testing, the goal is to verify mechanically that a product will retain its intended function during its whole life cycle and despite material aging. In the automotive or aerospace industry for example, such tests are often performed in hardware-in-the-loop simulations. To make a statistically significant evaluation, several devices are tested at the same time. Often, the load is applied to the test devices in the shape of periodic, pre-defined maneuvers. These depend on the device and are designed to mimic load similar to its real world usage.

978-1-7281-2989-1/21/$31.00 2021 European Union IEEE

During the tests several measurement signals are recorded. For a hydraulic system, for example, relevant signals are output pressure, input motor current and voltage and system temperatures. Data is collected centrally and evaluated during the test bench run for engineers and technicians (see Fig. 1). Testing is time intensive, expensive and demands specialized equipment. An anomaly in these tests is for example a bursting hydraulic pipe, which would be a critical error that must not occur in a product life cycle.

It is of interest to detect and analyze such unexpected defects as early as possible: First, the root cause of the malfunction is more likely to be found before additional parts get damaged. Second, a need for subsequent system improvement can be identified sooner, thereby shortening development feedback cycles. Automating the process of detecting anomalous behavior in a test bench saves time, because visual inspection of the recorded maneuver signals is time intensive and can only be done by trained personnel. Even with adequate visualization of the time series, a visual regular inspection is expensive.

There are challenges for an anomaly detection algorithm in this use case: The test benches yield large amounts of data due to the continuous data recording with high sampling rates (in this work up to 100 Hz, but up to 10 kHz can be common). Additionally, often no labeled training data for the training of anomaly detection systems exist: During research and development, the tested systems are first-of-their-kind prototypes and often simulations do not model the system's dynamic properties accurately enough to produce all measurement signals. Furthermore, due to the aging resulting from the test bench cycles, the output data will show a concept drift, i.e. the normal behavior of the systems changes with time, even if no anomaly is present.

The goal of this paper is to demonstrate a robust and easily adjustable method which can be applied without extensive prior feature engineering from the start of the test bench. To solve this problem, we assume that several parts are tested synchronously using repeated load maneuvers and that the systems, on average, work normally. Therefore, the problem is in the scope of anomaly detection on periodic, multivariate time series data. It takes place in an unsupervised setting, because there is no labeled data prior to its execution [3]. We compute the normal maneuver of a certain time interval

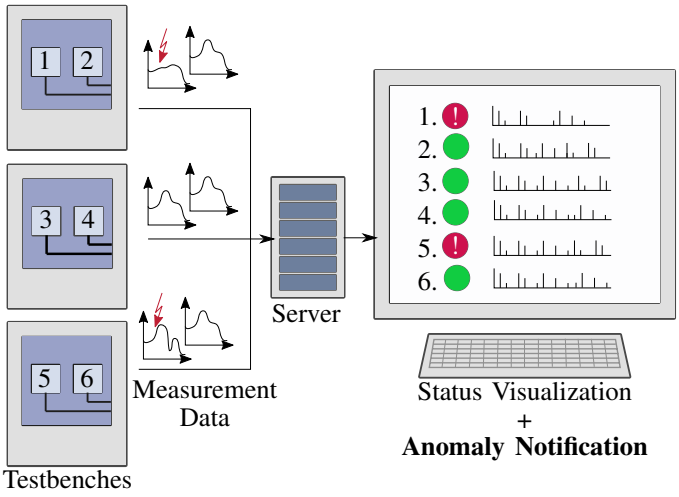


Fig. 1: Schematic of multiple test benches with six total tested systems. The data is collected centrally and processed for easy access to technicians and engineers.

from the median of all repetitions over time. The distance of every maneuver towards the normal is then classified using an unsupervised anomaly detection algorithm. We evaluate six time series distance metrics and two different anomaly classification methods. We show that our method is, to a certain degree, robust towards wear related concept drifts in the data.

This paper first presents related work. Afterwards, the data set and the evaluated methods will be explained. Finally, we will show the results of our experiments and conclude which algorithm is the best in this use case.

II. Related Work

We focus on multivariate, cyclical measurement signals under concept drift. The load applied to the system is pre-programmed by an engineer in one or several maneuvers, which means that all time series are time aligned and there are no periodicity changes between repetitions. Therefore, we evaluate difference metrics for periodic time series and outlier detection algorithms in combination.

The data set stems from work by Helwig et al. and has been analyzed and described by them in [1], [4] and [5]. Their work focuses on accurate supervised classification of hydraulic system defects based on extensive feature engineering and feature reduction for effective search in time series data. This use case differs from the one in this work, which is to detect anomalies with as little prior feature engineering as possible, in an unsupervised way. In our case, premature feature selection can cause errors in a system to go undetected, if they appear in feature spaces that were omitted before.

Anomaly detection for time series is a very common topic in research (see e.g. the survey by Chandola et al. [6]). In terms of unsupervised anomaly detection, for example in periodic ECG data, Chakraborty et al. [7] first reconstructed a ”normal” signal as an average of several repetitions using Dynamic Time Warp. The distance between a new repetition and the ”normal” is computed and if it exceeds a pre-defined threshold, it is classified as an outlier. Manually determining a threshold is not desirable for our use case, because it reduces the flexibility of the approach towards new device types. In contrast, Twitter published an approach for univariate detection on periodic data with an automatic threshold, which is specialized on point anomalies [8]. The time series is first decomposed with Seasonal Trend Decomposition (STD) into a trend, a seasonal and a residual component and the residual is then classified for anomalies using the Generalized Extreme Studentized Deviate Test (ESD Test) by Rosner [9], which depends on Students t-distribution. Our use case deals with complete, multivariate time series instead. In experiments, Rosner shows that the ESD test shows reasonable accuracy at a sample size above $n = 15$. For point anomalies, this is not an issue, but it can be a limit for anomaly detection on complete time series for test benches with lesser samples. Therefore, we use evaluate a different statistical measure, the Modified z-Score by Iglewicz and Hoaglin [10]. Additionally, Local Outlier Factor, a nearest neighbor based method, is evaluated. It has the advantage over other methods like the One Class Support Vector machine, that it can be applied without prior labeled training data [11].

Anomaly detection algorithms rely upon distance metrics of data points for classification. For time series distance metrics, several prior reviews exist, for example by Serr and Arcos [12] or Toller et al. [13]. Because of the high sampling rate and size of the data set, we focus on computationally efficient metrics, which is why Dynamic Time Warp distance, for example, will not be part of this work. Rather, metrics like the Mean Absolute Error and Mean Squared Error will be used.

With the rising popularity of deep learning, other anomaly detection methods, for example using Auto-Encoder networks have been applied either on time series or on time series features. Chalapathy et al. list several in their survey, among which use cases like industrial and IoT (internet of things) anomaly detection methods [14]. Additionally, Braei and Wagner [16] surveyed several neural network architectures for anomaly detection in univariate series.

Audibert et al. [15] show deep learning methods for the unsupervised multivariate time series anomaly detection for IT systems, and Yin et al. [17] for IoT time series. Ding et al., for example, demonstrate the use of Auto-Encoders for anomaly detection in cyber physical systems using LSTM (Long Short-Term Memory) Auto-Encoders [18]. Auto-encoders act as unsupervised, non-linear feature extractors. Once trained, these neural networks can be used efficiently on modern hardware. For anomaly detection, the reconstruction error of the network is used as the distance metric. In general, these methods need a pre-set threshold for outlier classification. Alternatively, Hundman et al. demonstrate the use of a dynamic threshold in combination with LSTM auto-encoders [19].

If there are continuous changes in the system that cause a concept drift in the data, an Auto-Encoder model needs to be re-trained or the anomaly threshold has to be adjusted. For non-periodic data sets in more complex use cases, Auto-Encoder networks would be beneficial. Since test bench data often is

periodic, a more direct approach for anomaly quantification can be considered, as shown in this work.

III. Anomaly Detection for Test Bench Data

First, we give an overview over the data set. Afterwards, the anomaly detection process is explained in two parts: The difference metrics between a maneuver repetition and the average of all related maneuvers, followed by the outlier detection method used to classify the repetitions into normal and anomaly.

A. Hydraulic Test Bench Data Set

The hydraulic test bench data set consists of labeled multivariate measurements of a hydraulic system under test. The system has four different components (cooler, valve, pump, hydraulic accumulator) at three to four discrete levels of wear, with ten repetitions of each combination. It consists of 17 different signal channels (amongst others six pressure, two volume flow and four temperature) with different sampling rates. In the following all channels are re-sampled to the highest sampling rate (100 Hz) using linear interpolation and rescaled to an amplitude of $[0, 1]$. Not every type of anomaly is visible in all channels equally, as shown in Fig. 2 and 3. The degradation of the cooler is clearly visible across several channels, while the deteriorating valve is mostly visible in the "Temperature 1" channel, and not clearly discernible in the other channels.
For the simulation of the concept drift, we constructed a data set by taking different component wear levels in a continuously degrading order, starting with the normal for all components. This will be described in detail later.

B. Difference Metrics

We assume that the maneuver repetitions are synchronized over time, as shown in the used data set. Should this not be the case, for example because of measurement hardware limitations, the methods presented in this paper can be applied after re-fitting the data on the time axis, for example using Dynamic Time Warp or cross-correlation.

To deal with the unsupervised setting of the problem, the median maneuver of all repetitions is assumed as the normal; and repetitions with unusually high deviation from it are counted as abnormal. Therefore, the anomaly detection process is divided into the following steps: First, the median maneuver is computed out of all repetitions. Then, the deviation from the normal is computed per channel for all repetitions, which are then used as input for the anomaly classification algorithm. This means that for this data set with 17 channels, each repetition yields a feature vector with 17 elements comprised of the differences towards the median, which is used as input for the classifier. The median was chosen as averaging metric due to its robustness towards outliers.

In the following, the evaluated distance metrics are described:

1) Mean Absolute Error and Mean Squared Error: The Mean Absolute Error (MAE) and Mean Squared Error (MSE) are common difference metrics in statistics and machine learning. MSE is less sensitive to smaller errors than MAE. This is of interest because of sensor noise often prevalent in data with a higher sampling rate. Both distance metrics towards the normal $\hat{x}$ with the length n are defined as follows:

$$MAE = \frac{1}{n}\sum_{t=1}^{n}|\hat{x}_t - x_t| \tag{1}$$

,

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(\hat{x}_t - x_t)^2 \tag{2}$$

2) Difference of the Cumulative Sum: The difference of the cumulative sum (in the following MAE Med. Sum, described in [20]) is a distance metric where the signals are cumulatively summed over time before their difference (in MAE) is computed. Its aim is to be able to take the distance of peaks into account (shown in Fig. 4), which also makes it more robust towards noise. It is evaluated because its benefit for noisy signals (especially sensor noise) is plausible. It is implemented using the cumsum function in NumPy [21].

3) MSE on the Fast Fourier Transform: In order to see, whether frequency features of the signals $s(t)$ are influenced by changes in the system, we also evaluate the MSE based on the log-scaled Fast Fourier Transform (FFT) of the signals. The scaling is performed to enhance the higher frequencies of the spectrum. We define the resulting FFT' as:

$$FFT'(s(t)) = log(|FFT(s(t))|) \tag{3}$$

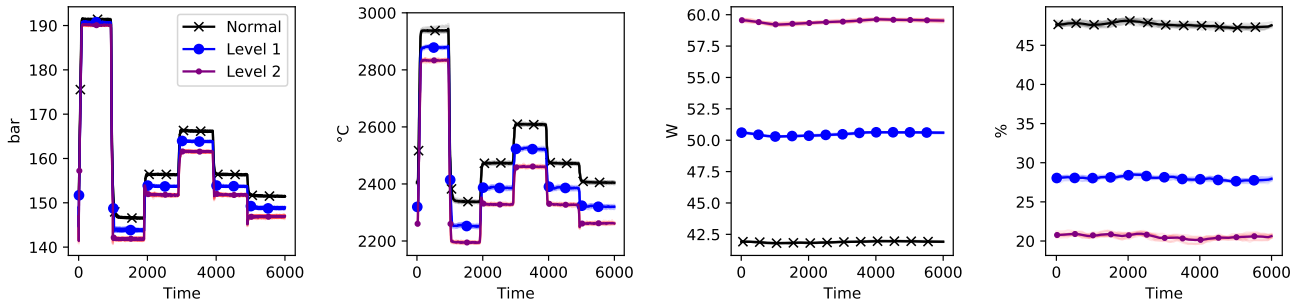
The distance between the FFT' is computed using MSE.

4) Correlation: Correlation is a measure of similarity between two vectors. The implementation in the Scipy library [22] of the correlation z between two vectors x and y, with $||x||$ as the length of x and $N = \max(||x||, ||y||)$ is defined as:

$$z[k] = (x * y)(k - N + 1) = \sum_{l=0}^{||x||-1} x_l y^*_{l-k+N-1}. \tag{4}$$

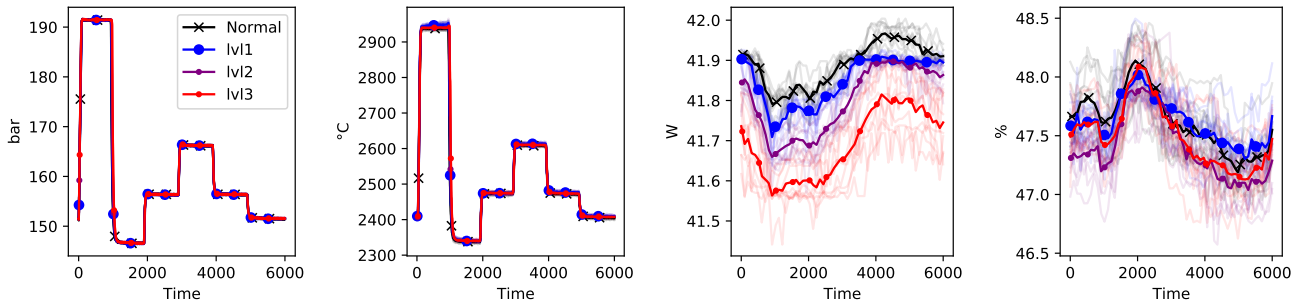
This yields an array z with the same length as the inputs. The used value for evaluation in this case is the last element of z, which, if x and y are identical, is the point of its greatest correlation.

5) Distance towards the Function Envelope: The Hilbert transform yields an upper and lower bound for an oscillating function. This can be used to compute a 'tolerance envelope' around a noisy time series to make anomaly detection more robust. To calculate the envelope for a non-oscillating function, first the median-smoothed time series is subtracted from the signals. The average tolerance envelope of all repetitions is computed as the median of all upper bounds and the median of all lower bounds, to yield the average upper and lower limits s_u and s_l, as shown in Fig. 5. All signal points between these bounds are counted as zero, the remainder is counted in their



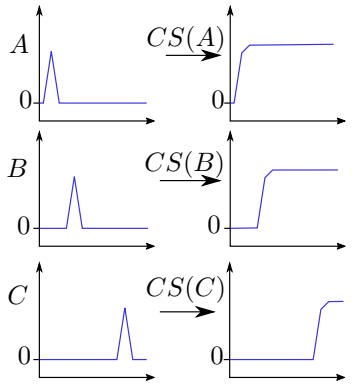
(a) Pressure 1 (b) Motor Power (c) Temp. 1 (d) Cooling Efficiency

Fig. 2: Plots at different levels of cooler degradation. The median measurement is visualized in opaque with line-markers. The differences in amplitude are clearly visible in all channels.



(a) Pressure 1 (b) Motor Power (c) Temp. 1 (d) Cooling Efficiency

Fig. 3: Plots of measurements at different levels of valve degradation. Measurements are shown as semi-transparent, while median measurement is shown opaque with line markers. The difference in amplitude is not as clearly visible as in Fig. 2, especially for Temp. 1 and Cooling Efficiency.



$$|A - B| = |A - C|$$
$$|CS(A) - CS(B)| < |CS(A) - CS(C)|$$

Fig. 4: Visualization of the cumulative sum (CS) distance. A and C are supposed to be closer in distance, which is achieved when first computing the cumulative sum of the time series. Example derived from [20].

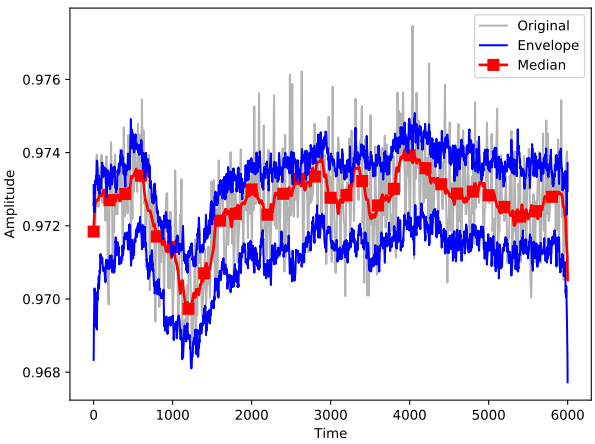


Fig. 5: Envelope function computation using Hilbert function on a function that does not oscillate around 0. Before the envelope computation the windowed median of the signal is subtracted.

distance towards the envelope. Therefore, this difference $\text{diff}(t)$ of a signal $s(t)$ is computed as:

$$\text{diff}(t) = \begin{cases} s(t) - s_u(t) & \text{if } s(t) > s_u(t), \\ 0 & \text{if} s_l < s(t) < s_u \\ s_l(t) - s(t) & \text{if} s(t) < s_l(t) \end{cases} \quad (5)$$

Again, diff(t) is squared and averaged using the mean to yield the anomaly score per channel.

C. Multivariate Outlier Algorithms

Having shown the difference metrics, we now explain the evaluated anomaly detection methods. They were chosen to work without prior training data. We evaluate the following anomaly classification methods instead:

- Local Outlier Factor (LOF)
- Modified z-Score

LOF is a nearest-neighbor based anomaly classification method. Its main parameter is the neighbor count. When one data point is the nearest neighbor of n neighbors, this means that it is classified as normal. If the point is not amongst its neighbors' nearest neighbors it is classified as an outlier, as shown in Fig. 6. This means that LOF can be used in a multivariate setting, by computing it based on all distances per signal channel per maneuver.

The Modified z-Score by Iglewicz and Hoaglin [10] is a statistical score based on the median deviation of a data set. We adjust it for the multivariate case by computing the average Modified z-score for all channel differences in a maneuver repetition. If the average z-score is above the set threshold value (3.5), the complete maneuver counts as an anomaly. The Modified z-score for a maneuver repetition with c channels based on the differences d of the repetition is defined as:

$$score = \frac{1}{c}\sum_{i=1}^{c} |0.6745(d_i - \tilde{d})/\text{MAD}(d)|, \qquad (6)$$

with $\tilde{d}$ as median of all repetitions and MAD as the median absolute deviation. The threshold is chosen according to the original authors, who equate a Modified z-score above 3.5 to an anomaly.

D. Interpretability of results

Since the outlier classification is based on the difference between the signal types of a maneuver repetition, it is possible to visualize the result of the anomaly detection for further inspection. Fig. 7 demonstrates this. Plotting the difference metrics with one line per maneuver in parallel coordinates shows the channels with the most deviation from the normal. This can help technicians and engineers find the channels with the most influence on the classification decision, and support them in a faster verification.

E. Real-Time Anomaly Detection and Concept Drifts

The shown methods can be used for real time anomaly detection by comparing the output of one tested system to all others performing the same maneuver at the same progress in the test run in a sliding window. It is especially important to reset the window for changing operating points, e.g. in temperature modulated test benches. Once a significant temperature change is performed, old measurements can cause false positive anomaly classifications. Therefore, the maneuver repetitions for comparison have to be recorded at the same temperature. As a rule of thumb, our experiments have shown that eight to ten repetitions are sufficient, though less might work as well given that only a minority of parts fail at a time. This also works for the anomaly detection under concept drift. Since there is no explicit model trained, it is possible to detect anomalies even under system changes.

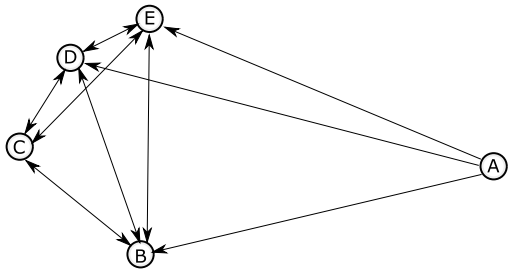


Fig. 6: Visualization of the LOF algorithm with a neighbor-count of 3. Arrows show the nearest neighbors of a node. Point A has several other points in the left cluster as its nearest neighbors, while the nearest neighbors of the points in the left group are in the group. Therefore, point A would be classified as an outlier.

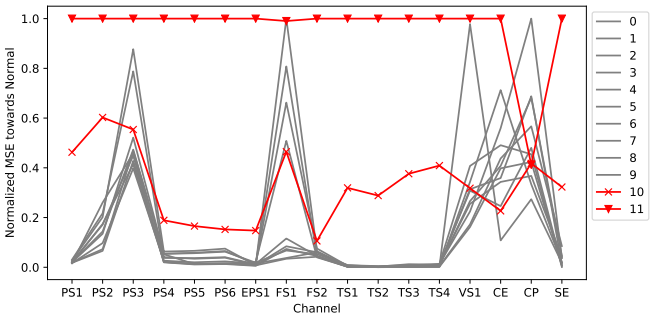
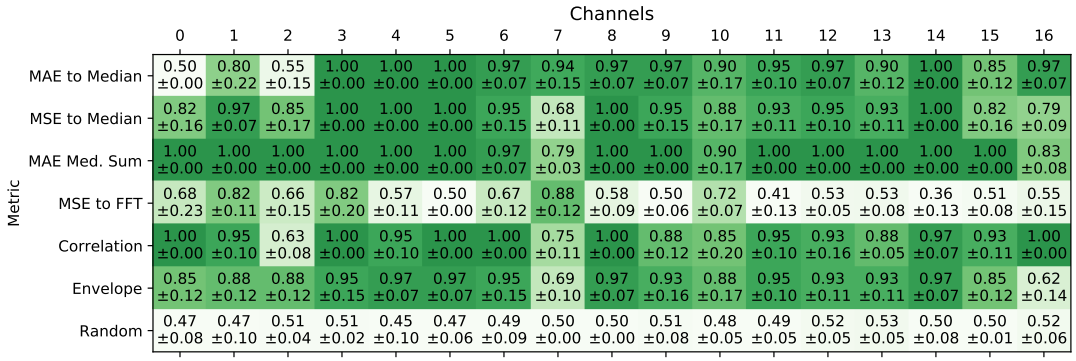


Fig. 7: Visualization of a classification result with parallel coordinates. Each line symbolizes one measurement. For each channel, the anomaly score is shown. Repetitions 10 and 11 are marked as classified as anomalous. This way, the relevant channels can be selected for further investigation.

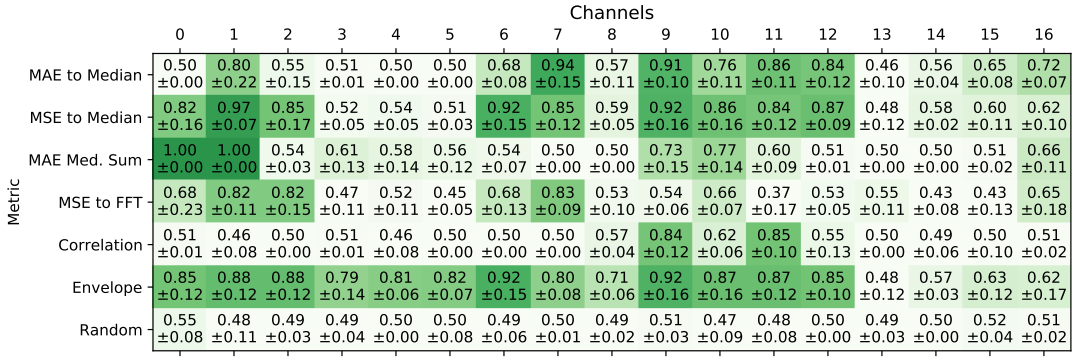
IV. Experiments

Having described the process of difference computation between the maneuver repetitions and the subsequent anomaly classification, we now describe the experiments used to evaluate the different methods.

Our experiments are structured in three parts. First, we evaluate the distance metrics for different defect parts per channel. Then, we evaluate the distance metrics from Section III-B with different anomaly detection algorithms for anomalies in the different subcomponents (cooler, valve, pump, hydraulic accumulator) of the system. In the end, we use a derived concept drift data set to evaluate whether subcomponent anomalies are detectable over the system life cycle. The accuracy in all experiments is evaluated in 10-fold cross validation using the area under the ROC curve statistic in Scikit-learn. During all



(a) Cooler



(b) Valve

Fig. 8: Experiments Part 1: Results of difference metrics after anomaly classification using the standard Modified z-Score. Better values are highlighted in green.

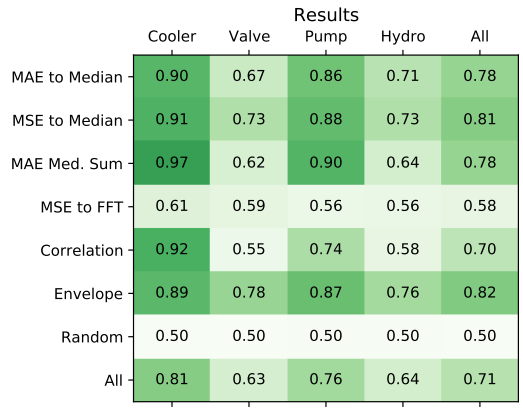


Fig. 9: Experiments Part 1: Comparison of all distance metrics over all different parts.

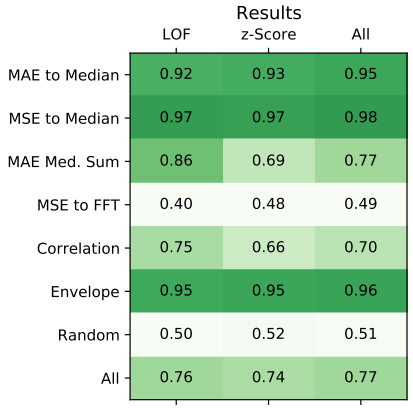


Fig. 10: Experiments Part 2: Comparison of distance metrics in combination with the two different anomaly detection algorithms for the multivariate case.

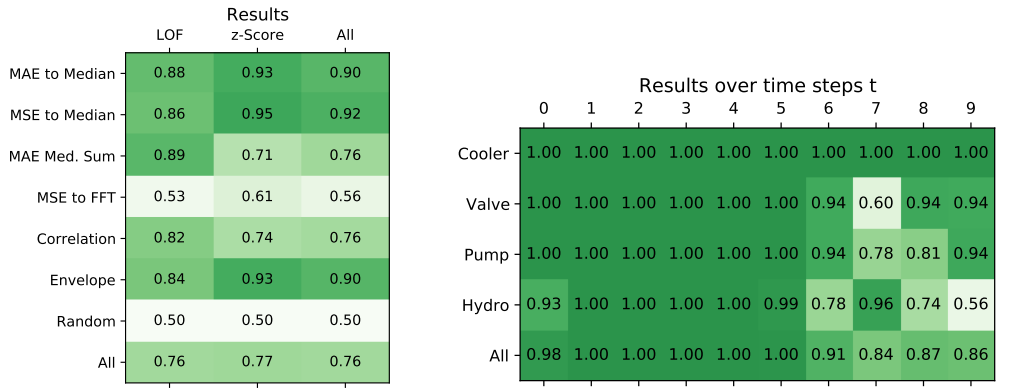
experiments, a randomized distance function is also computed as a comparative metric.

A. *Per Channel Anomaly Detection*

To evaluate distance measures, the Modified z-Score is used in a semi-supervised setting per measurement channel. The mean and MAD of the normal state was modelled using 8

TABLE I: Test bench measurements taken for the simulation of the concept drift dataset. To create the data set for each sub-component, the normal trend over time was used and the respective column was replaced with the components' normal and abnormal columns to create the specific dataset. The percentages stem from the discrete wear levels in the data set.

	Normal Integrity over Time				Cooler Test		Valve Test		Pump Test		Hydr. Test	
t	Cooler	Valve	Pump	Hydr.	Normal	Anomaly	Normal	Anomaly	Normal	Anomaly	Normal	Anomaly
0	100%	100%	100%	100%	100%	50%	100%	67%	100%	50%	100%	67%
1	50%	100%	100%	100%	50%	0%	100%	67%	100%	50%	100%	67%
2	50%	67%	100%	100%	50%	0%	67%	33%	100%	50%	100%	67%
3	50%	67%	100%	100%	50%	0%	67%	33%	100%	50%	100%	67%
4	50%	67%	50%	67%	50%	0%	67%	33%	50%	0%	100%	67%
5	50%	33%	50%	67%	50%	0%	67%	33%	50%	0%	67%	33%
6	0%	33%	50%	67%	50%	0%	67%	33%	50%	0%	67%	33%
7	0%	0%	50%	67%	50%	0%	33%	0%	50%	0%	67%	33%
8	0%	0%	0%	33%	50%	0%	33%	0%	50%	0%	33%	0%
9	0%	0%	0%	0%	50%	0%	33%	0%	50%	0%	33%	0%



(a) Comparison of all difference scores and outlier algorithms for concept drift.

(b) Comparison of detection accuracy over time for MSE and Modified z-Score.

Fig. 11: Experiments Part 3: Results of experiments concerning the concept drift of the system.

samples from the normal data set, where every subcomponent is in best condition. The testing was done using 2 from the normal, and 20 abnormal repetitions (test set). The distance measures were computed per signal channel. Measurements with a score above 3.5 were classified as anomalies. Results can be seen in Fig. 8 for wear on the cooler and the valve. In these experiments, the detection accuracy for wear in the cooler is generally better than for the valve. This can be explained by the data set plots shown before (Fig. 2 and 3), where the cooler wear was detectable easily for the human eye. Changes in the cooler, for example, were the most difficult to detect in the signal channel 7. Fig. 9 shows that the general accuracy of the distance metrics per part of the Envelope-Diff, MSE, MAE and MAE Med. Sum is similar, with Envelope and MSE being the best. The MSE-FFT performs worst of all, followed by correlation.

B. Different Outlier Classification Methods

The outlier classification methods were then evaluated in a semi-supervised, multivariate setting, to evaluate their accuracy for the wear of single components. This means that the algorithm's normal mode is extracted based on 8 samples of the normal data set, and is then evaluated by classifying anomalies from 2 remaining normal and 20 abnormal samples from each wear level per part. Based on this, the Local Outlier Factor with n=5 and the Modified z-Score with a threshold of 3.5 were evaluated.

Fig. 10 shows the results: In this use case, LOF works the best in combination with MAE. Notice how this is different from the channel-wise anomaly detection case, where Envelope-Diff and MSE performed best.

C. Concept Drift Analysis

For the concept drift evaluation using unsupervised anomaly detection, we simulated the increasing change in the system with time series data at ten different time steps. For this, a system life cycle from intact to maximum wear of all subcomponents was simulated. The taken measurements can be seen in Table I. The normal wear over time was adjusted by replacing the corresponding column with the subcomponents normal and abnormal. For example, at $t = 3$, the accuracy for Cooler anomalies were tested with 10 normal samples of $C = 50\%$, $V = 67\%$, $P = 100\%$ and $H = 100\%$ combined with 2 abnormal measurements with $C = 0\%$. This way, a comparable evaluation was achieved.

In general, for this use case, the Modified z-Score in combination with the MSE works best, as shown in Fig. 11a. Fig. 11b shows the different accuracies per component over

time for this constellation. It is noticeable that the accuracies for all parts except the cooler decreases with $t \geq 6$. Nonetheless, this experiment is a proof of concept that, up to a certain degree, accurate anomaly detection under concept drift is possible using this method. Generally, changes in the cooler are detectable with the most reliability. The accuracy of the model over time degrades after $t = 6$, which coincides with the maximum wear level of the cooler.

V. Discussion

In this work, we showed a framework for test bench anomaly detection on the example of one hydraulic data set. Nonetheless, the model's accuracy can decrease with the raising wear of the complete system. This is especially the case, if the wear of various subcomponents influences the system's signal amplitudes differently, as shown in our example.

We focused on one data set that was taken from one hydraulic system. The great benefit of this data set were the different wear levels in the subcomponents, the diverse measurement signals and the accurate labeling. This enabled us to do a first of a kind multivariate anomaly detection on periodic time series data. Nonetheless, a larger amount of measurements would be beneficial for showing the statistical significance of our method, especially in the concept drift analysis.

We see several areas of future research. We only evaluated one distance measure per channel at a time. A combination of multiple feature types, like envelope function with the correlation function, might make predictions more accurate. Using the presented methods in an ensemble training approach would also be of interest. Additionally, evaluating these methods on more systems and use cases is required. Other types of preprocessing can also be evaluated in combination with the presented anomaly classification algorithms, like the step-wise Fourier transform or the discrete Wavelet transform.

VI. Conclusion

We demonstrated methods to evaluate periodic time series anomaly detection for hardware test benches. We first evaluated several distance measures (Hilbert function envelope, MSE, MAE) and used them to evaluate two outlier detection methods. Finally, a concept drift was simulated to emulate wear on different components during test and the presented variants were demonstrated to perform well in this more complex anomaly detection task. The great advantage of our method is that it can be employed from the start of a test bench and with a small sample size. The shown method is computationally efficient and can be re-performed frequently, making it robust when dealing with changing operating points of a system that is aging under load.

References

[1] N. Helwig, E. Pignanelli, and A. Schütze, "Condition monitoring of a complex hydraulic system using multivariate statistics," in *2015 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*. IEEE, 11.05.2015 - 14.05.2015, pp. 210–215. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Condition+monitoring+of+hydraulic+systems

[2] *IEEE standard computer dictionary: A compilation of IEEE standard computer glossaries, 610*. New York, NY, USA: Institute of Electrical and Electronics Engineers, 1990.

[3] M. Goldstein and S. Uchida, "A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data," *PloS one*, no. 11, p. e0152173, 2016. [Online]. Available: https://doi.org/10.1371/journal.pone.0152173

[4] T. Schneider, N. Helwig, and A. Schütze, "Automatic feature extraction and selection for classification of cyclical time series data," *tm - Technisches Messen*, vol. 84, no. 3, 2017.

[5] N. Helwig, E. Pignanelli, and A. Schütze, Eds., *D8.1 - Detecting and Compensating Sensor Faults in a Hydraulic Condition Monitoring System: AMA Service GmbH, P.O. Box 2352, 31506 Wunstorf, Germany*, 2015.

[6] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, 2009.

[7] G. Chakraborty, T. Kamiyama, H. Takahashi, and T. Kinoshita, "An Efficient Anomaly Detection in Quasi-Periodic Time Series Data—A Case Study with ECG," in *Time Series Analysis and Forecasting*, ser. Contributions to Statistics, I. Rojas, H. Pomares, and O. Valenzuela, Eds. Cham: Springer International Publishing, 2018, pp. 147–157.

[8] A. Kejariwal, "Introducing practical and robust anomaly detection in a time series," 2015. [Online]. Available: https://web.archive.org/web/20210506134624/https://blog.twitter.com/engineering/en_us/a/2015/introducing-practical-and-robust-anomaly-detection-in-a-time-series.html

[9] B. Rosner, "Percentage Points for a Generalized ESD Many-Outlier Procedure," *Technometrics*, vol. 25, no. 2, p. 165, 1983.

[10] B. Iglewicz and D. C. Hoaglin, *How to detect and handle outliers*, ser. ASQC basic references in quality control. Milwaukee, Wis: ASQC Quality Press, 1993, vol. v. 16.

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[12] J. Serrà and J. L. Arcos, "An Empirical Evaluation of Similarity Measures for Time Series Classification," *Knowledge-Based Systems*, vol. 67, pp. 305–314, 2014. [Online]. Available: http://arxiv.org/pdf/1401.3973v1

[13] M. Toller, B. C. Geiger, and R. Kern, "A Formally Robust Time Series Distance Metric." [Online]. Available: http://arxiv.org/pdf/2008.07865v1

[14] R. Chalapathy and S. Chawla, "Deep Learning for Anomaly Detection: A Survey." [Online]. Available: http://arxiv.org/pdf/1901.03407v2

[15] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, "USAD," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, R. Gupta, Y. Liu, J. Tang, and B. A. Prakash, Eds. New York, NY, USA: ACM, 08232020, pp. 3395–3404.

[16] M. Braei and S. Wagner, "Anomaly Detection in Univariate Time-series: A Survey on the State-of-the-Art." [Online]. Available: http://arxiv.org/pdf/2004.00433v1

[17] C. Yin, S. Zhang, J. Wang, and N. N. Xiong, "Anomaly Detection Based on Convolutional Recurrent Autoencoder for IoT Time Series," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–11, 2020.

[18] K. Ding, S. Ding, A. Morozov, T. Fabarisov, and K. Janschek, "On-Line Error Detection and Mitigation for Time-Series Data of Cyber-Physical Systems using Deep Learning Based Methods," in *2019 15th European Dependable Computing Conference (EDCC)*. IEEE, 17.09.2019 - 20.09.2019, pp. 7–14.

[19] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom, "Detecting Spacecraft Anomalies Using LSTMs and Nonparametric Dynamic Thresholding," vol. 60, pp. 387–395, 2018. [Online]. Available: http://arxiv.org/pdf/1802.04431v3

[20] paolof89, "Time series distance metric," 2021. [Online]. Available: https://web.archive.org/web/20210507081734/https://stackoverflow.com/questions/48497756/time-series-distance-metric

[21] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen *et al.*, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, 2020.

[22] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy *et al.*, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.