List of Changes and Answers to Reviewers

Adaptable and Interpretable Framework for Anomaly Detection in SCADA-based Industrial Systems
M. Wadinger, M. Kvasnica

December 5, 2023

1 List of Changes

List of main changes:

1. Introduction was enriched to address reviewers' comments.
2. Figure 2 was added to illustrate the definition of anomalies to address reviewers' comments.
3. Captions in Figures 5,6,7 (labeled as Figures 4,5,6 in the previous manuscript) were reworded to make them more illuminating.
4. Table 2 was transposed to allow for new metrics to be added.
5. False Alarm Rate, AUC, and Mean of Rolling AUC were added to the results in Table 2 to address reviewers' comments.
6. Section 4.4 was added to address reviewers' suggestion on scalability analysis.

2 Answers to Reviewers and to the Associate Editor

We would like to thank all reviewers and to the associate editor for encouraging comments and hints. We have tried to address all of them appropriately.

Associate Editor

Reviewers have now commented on your paper. You will see that they are advising that you revise your manuscript. If you are prepared to undertake the work required, I would be pleased to reconsider my decision.

For your guidance, reviewers' comments are appended below.

If you decide to revise the work, please submit a list of changes or a rebuttal against each point raised by the reviewers. You can upload this as the 'Detailed Response to Reviewers' when you submit the revised manuscript.

Response: We would like to thank the associate editor for his/her evaluation. We believe that the modifications, described in more detail below, address all issues pointed out by the reviewers.

Reviewer 1

This paper proposes a new online anomlay detection method and verifies its effectiveness on real-world datasets. However, there are some limitations as follows:

1. *There is no clear definitions of point anomaly, collective anomalies and concept changes. Figure 2 does not illustrate their differences either.*

   Response: Thank you for your suggestion. We have added Figure 2 in the revision to illustrate the distinctions between these anomaly types. Figure 2, previously in question, is now labeled as Figure 3 in the revision.

   Figure 2 depicts a sample of each of the three scenarios: point anomaly (measurement with significant dissimilarity), collective anomaly (cluster of abnormal points), and change point (initial sequence of changed operation) detection.

2. *The captions of Figures 4,5,6 are similar but the labels are different. It is a bit confused as which one is the ground truth.*

   Response: The reviewer is correct that the captions are too similar. We have updated the captions in the revised manuscript to eliminate any confusion. Please note that Figures 4, 5, and 6 in the initial submission are now labeled as Figures 5, 6, and 7 in the revised manuscript due to the addition of Figure 2 to address the reviewer's previous comment.

   The changes made to the captions include a description of two model setups: without adaptation to change points in Figure 5 and with adaptation to change points in Figures 6 and 7. Though an identical model setup is depicted in Figures 6 and 7, Figure 7 shows the accompanying task of sampling anomaly detection in addition. This way, we could better highlight specific features that build up our proposed AID method.

   It is important to note that the challenge of obtaining precise ground truth information remains. Operators did not inform us about the exact time of abnormal events, introducing ambiguity. While we refer to the dates of the events in Section 4.1, selecting the time of the anomaly for plotting purposes would be arbitrary and compromise the objectivity of

the results. Therefore, Figures 5, 6, and 7 do not bear information about ground truth.

3. *There are other self-supervised change-point detection method, such as [DSXS21].*

   Response: We appreciate the reviewer's suggestion and acknowledgment of the reference [DSXS21]. We have incorporated the reference into the Introduction section of the paper to enrich the discussion.

   We found the suggested reference highly relevant in the paragraph discussing the need for an early change point detection mechanism. After mentioned remark of Tartakovsky from 2013 [TPS13], that the immediate change point detection is not a feasible option unless there is a high tolerance for false alarms, we refer to the contrastive learning approach of reviewer's suggested reference [DSXS21] as a promising balance between early transition detection and low false alarm rate. We believe this helps to reflect the current state of the research better.

4. *It seems that using ARIMA or moving average can easily detect the anomalies or change points on the real-world datasets.*

   Response: We acknowledge this observation of ARIMA and moving averages usage for anomaly and change point detection tasks on real-world datasets. To address this observation, we included in the Introduction two recent publications dealing with offline vectorized adaptation of ARIMA for complex systems. Moreover, the reviewer's suggestion motivated our effort to apply methods from this family to data from the first case study. We were unable to find a suitable implementation of online-trained ARIMA for vectorized usage, resulting in the selection of an online univariate implementation. Due to the univariate nature and limitations discussed below, we did not include the results in the revised manuscript.

   Referring to Figure 1, it seems that unique challenges posed by online anomaly detection in evolving data streams within real-world datasets present some obstacles for ARIMA and moving average methods.

   In an effort to achieve the best performance, we performed Bayesian optimization of the Seasonal Non-linear AutoRegressive Integrated Moving-Average with eXogenous inputs model (SNARIMAX) [AHMS13] and its special parametrization cases for the data from the first case study. Various metrics, with and without running mean, were tested in the cost function, selecting Median Absolute Error (MedAE) as one with the best convergence and highest robustness toward anomalies. Five step ahead forecast of normalized temperature was used for evaluation, showing a good balance between anomaly discrimination and accurate tracking. We made several observations on the performance of the methods.

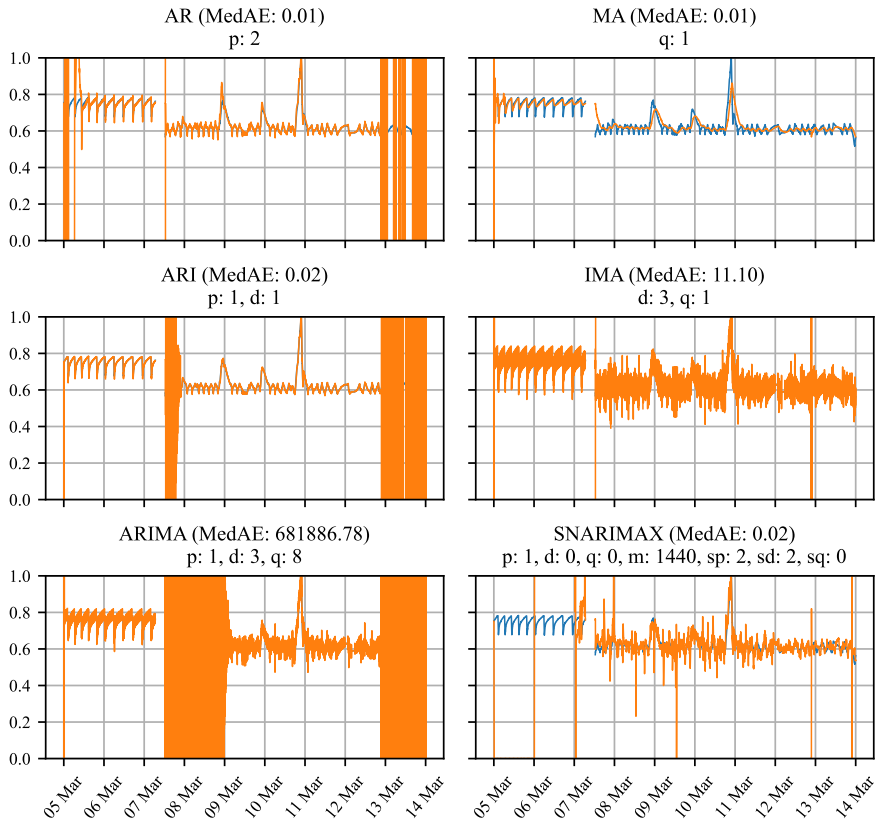

Figure 1: Comparison of five-step ahead forecast (orange) of normalized temperature (blue) using various parametrization of SNARIMAX model and special parametrization cases (AR, MA, ARI, IMA, ARIMA) with optimized parameter values. The forecast is aligned with the original data.

Firstly, the moving average (MA) method's forecast alone discriminates outliers well. Nevertheless, it shows little sensitivity to abrupt regular changes of temperatures from 5th to 7th March, which would be falsely marked as anomalous, along with true anomalies by a simple threshold filter. Secondly, the autoregressive (AR) method exhibits high sensitivity to point anomaly before 13th March, resulting in ongoing instability (though barely visible, sensor reports 0.0 on 12th March from 21:15 to 21:22 in Figure 1).

Using differencing to remove trends and seasonality with the AR method (ARI) further decreased stability during the formerly mentioned event starting on 12th March at 21:15 and, combined with the moving average method (IMA), added significant noise to predictions during whole evaluated history. Despite hyperparameter optimization and evaluation with engineered time-based features, we did not observe satisfactory results with ARIMA and SNARIMAX models.

Protection of the model from learning anomalous data through threshold filters, to mitigate their effect on the model's performance, and intelligent adaptation to change points and non-stationary may improve the performance. Nevertheless, due to the extent of the work required, we decided to leave the inclusion of these features as a potential avenue for future work.

A critical feature of our proposed method within SCADA-based systems is the ability to dynamically render process limits for individual signals, enhancing diagnostic capabilities. While a similar task could be achieved with ARIMA, it requires extensive fine-tuning of an ensemble of ARIMA models for each signal in the system. Vectorization of this task may significantly speed up this process.

Our literature review indicates that Vector Autoregression, the multivariate extension of ARIMA, could efficiently model multivariate time series, as explored in papers such as [MBMO16, ZZQ23]. However, our research focuses on online anomaly detection for evolving data streams, and the need for a vectorized implementation for online-trained ARIMA limits its usage in our comparisons. Future research could explore extending ARIMA to its multivariate counterpart for online training.

Integrating our proposed method with forecasting models during feature engineering, as demonstrated in Section 4.1 for physics-based model utilization, presents a promising direction for enhancing performance with the aid of the ARIMA method.

5. *It would be better to use AUC rather than F1 to measure the anomaly detection performance. Moreover, range-based AUC is even better and more fair for streaming or sliding window-based method.*

   Response: We agree with the reviewer that AUC, in general, is a better

metric for imbalanced datasets. In response to the reviewer's recommendation, we have included AUC in the results presented in Table 2 of the revised paper. Adding AUC provides an alternative perspective on performance that may interest the reader.

We tried to implement hyperparameter optimization with AUC. Nevertheless, due to the poor convergence of the reference methods on benchmark data, we decided to use the F1 score, which showed better convergence for all three compared methods.

Additionally, we were unaware of the range-based AUC metric during paper writing and result collection. After this suggestion, we computed the mean value of range-based AUC using the implementation from [BS17] and enriched the results in Table 2.

We also attempted to use the mean value of range-based AUC for hyperparameter optimization. Due to minimal improvement in performance compared to regular AUC, we decided to retain the F1 score as the optimized metric. The results obtained using the range-based AUC metric in the hyperparameter optimization cost function are provided in Table 1 for reference. We highlight that the false positive rate dropped by 10 % compared to the model optimized on the F1 score. This is a significant improvement in the context of anomaly detection, where false alarms have higher priority than precision or recall.

We hope these additions to the revised manuscript and explanations in this response enhance the transparency and completeness of our evaluation.

Table 1: Evaluation of models optimized for Rolling AUC score on SKAB dataset. The best-performing model is highlighted in bold. Values in brackets represent macro values of the metric.

Algorithm	AID	HS-Trees	OC-SVM
Precision [%]	**47** (60)	30 (47)	32 (48)
Recall [%]	**55** (61)	4 (50)	3 (50)
F1 [%]	**51** (60)	7 (42)	6 (42)
AUC [%]	**61**	50	50
Mean Rolling AUC [%]	**60**	50	49
FPR [%]	38	**37**	**37**

Reviewer 2

This paper presents an interesting and potentially useful framework called AID for anomaly detection and root cause diagnosis in industrial internet-of-things

(IoT) systems. It incorporates dynamic conditional probability distribution modeling to adapt to non-stationary data streams, which is crucial for industrial systems. And industrial case studies demonstrate capabilities on real systems. However, i still have following concerns.

1. *More analysis of computational complexity and scalability limitations for high-dimensional industrial systems would strengthen the work.*

   Response: Thank you for bringing attention to the importance of computational complexity and scalability in the context of high-dimensional industrial systems. To address this suggestion, we have incorporated Section 4.4 into our revised manuscript, providing a dedicated analysis of scalability and time complexity.

   In Section 4.4, we delve into the computational complexities and scalability limitations of our proposed method in the context of high-dimensional industrial systems. The section aims to offer insight into the scalability of the proposed method for anomaly detection tasks and in the context of SCADA-based systems where dynamic limits are of interest. Evaluation is performed on the data from the first case study, enriched with additional signals from other modules in the system, resulting in a total of 60 signals.

   Table 2 (Table 4 in the revised manuscript) presents the latency analysis of the proposed method AID with a varying number of features. The results indicate that the latency of the proposed method grows increasingly with the number of features. The minimum latency reflects the time required to process a single sample during the grace period, while maximum latency reflects the time required when multiple samples are processed at once.

Table 2: Latency analysis of the proposed method AID with varying number of features.

Number of Features	**Detection** $\mu \pm \sigma$ (min, max) [ms]	**Detection + Limits** $\mu \pm \sigma$ (min, max) [ms]
1	0.37 ± 0.26 $(0.05, 31.7)$	0.63 ± 0.38 $(0.23, 35.9)$
10	2.25 ± 0.92 $(0.10, 13.6)$	5.24 ± 0.98 $(0.80, 15.1)$
20	5.46 ± 2.16 $(0.26, 30.6)$	14.7 ± 2.27 $(1.10, 47.5)$
30	10.9 ± 4.31 $(0.52, 42.4)$	34.3 ± 4.50 $(2.59, 72.4)$
40	20.7 ± 8.15 $(0.89, 52.7)$	69.5 ± 8.57 $(2.84, 140)$
50	97.3 ± 47.4 $(1.36, 1010)$	297 ± 59.4 $(3.94, 1330)$
60	142 ± 71.2 $(1.95, 1640)$	468 ± 111 $(7.08, 3710)$

Figure 2 showing the distributions of the latency for a varying number of features indicate that the mean latency is cubic in the number of features in the detection task while inclines to quartic in the combined task of detection and dynamic operating limits setting.

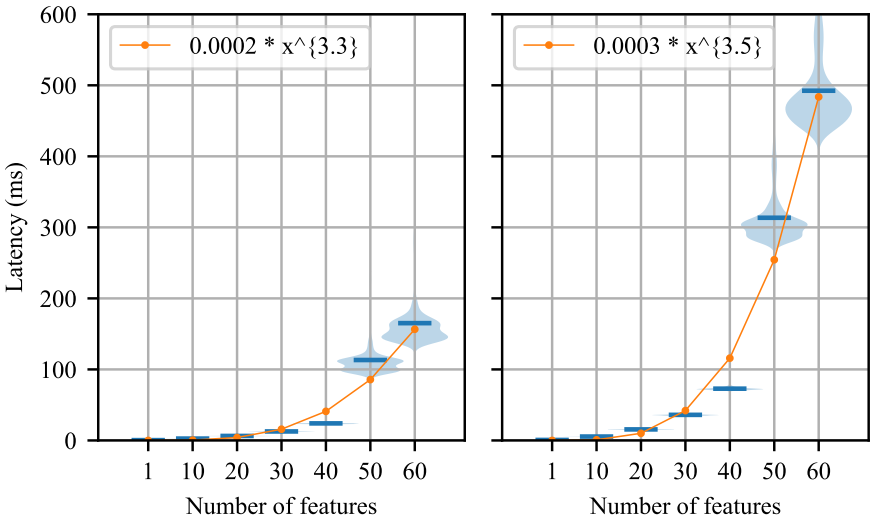

Figure 2: Analysis of latency distribution of the proposed method AID. Violin plots depict the distribution of the latency for varying number of features, while horizontal bars show mean latency.

We hope that this addition will enhance the completeness of our work.

2. *While the paper mentions comparisons with other methods, it lacks detailed benchmarking data, such as false positive rates.*

   Response: We appreciate the reviewer's insightful comment regarding the need for more detailed benchmarking data, particularly including false positive rates. In response, we have added a False Positive Rate (FPR) to the results in Table 2 of the revised manuscript (Table 3 in this document), addressing this specific concern.

   Additionally, to provide a more comprehensive view of the model's performance on imbalanced datasets while addressing other reviewers' comments, we have included AUC and mean of range-based AUC in Table 2 of the revised manuscript. These metrics offer further insights into the detection capabilities of our proposed method. In Table 3 we observe that our proposed AID method outperforms the reference methods in terms of added metrics on the benchmark dataset.

3. *Comparing diagnosis accuracy for root causes against other interpretable methods could better highlight capabilities.*

Table 3: Evaluation of models optimized for F1 score on SKAB dataset. The best-performing model is highlighted in bold. Values in brackets represent macro values of the metric.

Algorithm	AID	HS-Trees	OC-SVM
Precision [%]	**41** (59)	36 (51)	39 (54)
Recall [%]	**80** (59)	74 (51)	63 (54)
F1 [%]	**54** (53)	48 (44)	48 (52)
AUC [%]	**59**	51	54
Mean Rolling AUC [%]	**57**	50	53
FPR [%]	**47**	56	48
Avg. Latency [ms]	1.45	**0.05**	**0.05**

Response: We appreciate the reviewer's suggestion regarding the comparison of diagnosis accuracy for root causes against other interpretable methods. We recognize the importance of addressing this aspect, resulting in a comparison with the diagnostic method DBStream. We have tried hard to convey the results of this comparison in a clear and concise manner in the paper. However, we could not manage without significantly extending the page range, which we found inappropriate at this stage of writing the paper.

As we have recognized the challenges of adapting analogous interpretable methods from cited publications in our paper [YRB+22, SDV+21] due to undisclosed or non-adaptable code, we decided to compare our proposed method with DBStream [HB16], an online version of DBScan designed for evolving data streams.

It is an unsupervised method with the ability to disregard anomalies and capture clusters of various sizes and densities. It has been successfully used throughout the literature for diagnostic purposes. In [LZLW19], DBScan was used for thermal runaway diagnosis of battery systems in electric vehicles, relying on engineered features reflecting battery performance. Other applications include power transformer fault diagnosis [LSW+20] and fault diagnosis of rolling bearing [LWHL20]. Its usage in these applications, with expert knowledge of the system, involves mapping clusters to root causes, making it a promising method for our comparison.

The comparison was performed on the Controlled Anomalies Time Series (CATS) Dataset, a simulated complex dynamical system with 200 injected anomalies. This dataset is publicly available and suitable for root cause analysis, as disclosed in the dataset's description. We performed two types of comparisons.

Firstly, we assessed clustering performance, a key feature of DBStream. This analysis demonstrated the overlap of clusters and various groups

of root causes in data. Both methods were optimized to maximize the adjusted mutual information score, a metric that showed the highest importance on synthetic root cause data. The results of this comparison are presented in Table 2.

Secondly, we compared the ability to detect root causes. Since DBStream does not inherently provide information about root causes, we introduced expert knowledge a posteriori by mapping clusters to ground truth root causes based on the highest overlap. While this approach is artificial, it facilitated a comparison of the methods in root cause detection, with a slight benefit given to DBStream.

It is important to note that the cluster-to-root cause mapping was also performed while optimizing DBStream's hyperparameters, resulting in a slightly unfair comparison of our proposed method.

The detailed analysis revealed that our proposed method exhibits a higher precision in detecting root causes compared to DBStream. We have disclosed the results of this comparison in Table 3.

Table 4: Evaluation of AID and DBStream models optimized for adjusted mutual information score on CATS dataset. The best-performing model is highlighted in bold. Perfect clustering achieves 100% in each metric.

Algorithm	AID	DBStream
Adjusted Mutual Info [%]	**13**	2
Adjusted Rand [%]	**20**	1
Completeness [%]	**9**	1
Fowlkes-Mallows [%]	**87**	67
VBeta [%]	**14**	2

Table 5: Found optimal hyperparameters and searched ranges of AID and DBStream models optimized for adjusted mutual information score on CATS dataset.

Algorithm	**Hyperparameters**	**Found**	**Ranges**
AID	Threshold	0.99973	(0.95, 0.99994)
	t_e	22035	(150, 30000)
	t_a	3844	(50, 10000)
	t_g	16667	16667
DBStream	Cleanup Interval	691	(1, 1000)
	Clustering Threshold	15.68	(0.01, 100)
	Fading Factor	0.3547	(0.0001, 1.0)
	Interstection Factor	1.37	(0.03, 3.0)
	Minimum Weight	4.4	(0.1, 10)

Table 6: Evaluation of AID and DBStream models optimized for macro of F1 score on CATS dataset. The best-performing model is highlighted in bold. Perfect clustering achieves 100% in each metric. Values in brackets represent macro values of the metric.

Algorithm	AID	DBStream
Weighted Precision [%]	**96** (**46**)	93 (27)
Weighted Recall [%]	86 (**21**)	**96** (16)
Weighted F1 [%]	90 (**26**)	**95** (18)
FPR [%]	**1**	**1**

Table 7: Found optimal hyperparameters and searched ranges of AID and DBStream models optimized for macro of F1 score on CATS dataset.

Algorithm	**Hyperparameters**	**Found**	**Ranges**
AID	Threshold	0.99981	(0.95, 0.99994)
	t_{e}	20699	(150, 30000)
	t_{a}	6130	(50, 10000)
	t_{g}	16667	16667
DBStream	Cleanup Interval	605	(1, 1000)
	Clustering Threshold	9.25	(0.01, 100)
	Fading Factor	0.8090	(0.0001, 1.0)
	Interstection Factor	0.45	(0.03, 3.0)
	Minimum Weight	3.7	(0.1, 10)

References

[AHMS13] Oren Anava, Elad Hazan, Shie Mannor, and Ohad Shamir. Online learning for time series prediction. *CoRR*, abs/1302.6927, 2013.

[BS17] Dariusz Brzezinski and Jerzy Stefanowski. Prequential auc: properties of the area under the roc curve for data streams with concept drift. *Knowledge and Information Systems*, 52(2):531–562, Aug 2017.

[DSXS21] Shohreh Deldari, Daniel V. Smith, Hao Xue, and Flora D. Salim. Time series change point detection with self-supervised contrastive predictive coding. In *Proceedings of the Web Conference 2021*, WWW '21, pages 3124–3135, New York, NY, USA, 2021. Association for Computing Machinery.

[HB16] Michael Hahsler and Matthew Bolaos. Clustering data streams based on shared density between micro-clusters. *IEEE Trans. on Knowl. and Data Eng.*, 28(6):1449–1461, jun 2016.

[LSW+20] Yongxin Liu, Bin Song, Linong Wang, Jiachen Gao, and Rihong Xu. Power transformer fault diagnosis based on dissolved gas analysis by correlation coefficient-dbscan. *Applied Sciences*, 10(13), 2020.

[LWHL20] Hai Li, Wei Wang, Pu Huang, and Qingzhao Li. Fault diagnosis of rolling bearing using symmetrized dot pattern and density-based clustering. *Measurement*, 152:107293, 2020.

[LZLW19] Da Li, Zhaosheng Zhang, Peng Liu, and Zhenpo Wang. Dbscan-based thermal runaway diagnosis of battery systems for electric vehicles. *Energies*, 12:2977, 08 2019.

[MBMO16] Igor Melnyk, Arindam Banerjee, Bryan Matthews, and Nikunj Oza. Semi-markov switching vector autoregressive model-based anomaly detection in aviation systems. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1065–1074, New York, NY, USA, 2016. Association for Computing Machinery.

[SDV+21] Bram Steenwinckel, Dieter De Paepe, Sander Vanden Hautte, Pieter Heyvaert, Mohamed Bentefrit, Pieter Moens, Anastasia Dimou, Bruno Van Den Bossche, Filip De Turck, Sofie Van Hoecke, and Femke Ongenae. Flags: A methodology for adaptive anomaly detection and root cause analysis on sensor data streams by fusing expert knowledge with machine learning. *Future Generation Computer Systems*, 116:30–48, 2021.

[TPS13] Alexander G. Tartakovsky, Aleksey S. Polunchenko, and Grigory Sokolov. Efficient computer network anomaly detection by change-point detection methods. *IEEE Journal of Selected Topics in Signal Processing*, 7(1):4–11, Feb 2013.

[YRB+22] Wei-Ting Yang, Marco S. Reis, Valeria Borodin, Michel Juge, and Agnès Roussy. An interpretable unsupervised bayesian network model for fault detection and diagnosis. *Control Engineering Practice*, 127:105304, 2022.

[ZZQ23] Ruiyao Zhang, Ping Zhou, and Junfei Qiao. Anomaly detection of nonstationary long-memory processes based on fractional cointegration vector autoregression. *IEEE Transactions on Reliability*, pages 1–12, 2023.