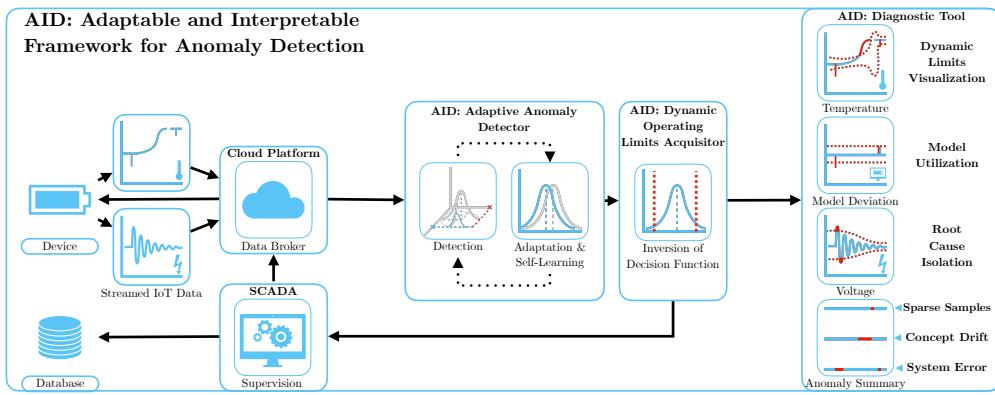


Graphical Abstract

Adaptable and Interpretable Framework for Anomaly Detection in SCADA-based Industrial Systems

Marek Wadinger, Michal Kvasnica



Highlights

Adaptable and Interpretable Framework for Anomaly Detection in SCADA-based Industrial Systems

Marek Wadinger, Michal Kvasnica

- Interpretable anomaly detector with self-supervised adaptation
- Demonstrates interpretability by providing dynamic operating limits
- Leverages self-learning approach on streamed IoT data
- Utilizes existing SCADA-based industrial infrastructure
- Offers faster response time to incidents due to root cause isolation

Adaptable and Interpretable Framework for Anomaly Detection in SCADA-based Industrial Systems

Marek Wadinger^{a,*}, Michal Kvasnica^a

^a*Institute of Information Engineering, Automation and Mathematics, Slovak University of Technology in Bratislava, Radlinskeho 9, 812 37, Bratislava, Slovakia*

Abstract

In this paper, we introduce an Adaptable and Interpretable Framework for Anomaly Detection (AID) designed for industrial systems utilizing IoT data streams on top of well-established SCADA systems. AID leverages dynamic conditional probability distribution modeling to capture the normal operation of dynamic systems and isolate the root causes of anomalies at the level of individual inputs. The self-supervised framework dynamically updates parameters of underlying model, allowing it to adapt to non-stationarity. AID interprets anomalies as significant deviations from conditional probability, encompassing interactions as well as both spatial and temporal irregularities by exposing them as features. Crucially, AID provides dynamic operating limits to integrate with existing alarm handling mechanisms in SCADA-based IoT systems. Two industrial-scale case studies demonstrate AID's capabilities. The first study showcases AID's effectiveness on energy storage system, adapting to changes, setting context-aware limits for SCADA, and ability to leverage a physics-based model. The second study monitors battery module temperatures, where AID identifies hardware faults, emphasizing its relevance to energy storage safety. A benchmark evaluation on real data shows that AID delivers comparable performance to other self-learning adaptable anomaly detection methods, with the significant advancement in diagnostic capabilities for improved system reliability and performance.

Keywords: Anomaly detection, Root cause isolation, Iterative learning, Statistical learning, Self-supervised learning

*Phone numbers: +421 902 810 324 (Marek Wadinger)

Email addresses: marek.wadinger@stuba.sk (Marek Wadinger), michal.kvasnica@stuba.sk (Michal Kvasnica)

¹ 1. Introduction

² Anomaly detection systems play a critical role in risk-averse systems by
³ identifying abnormal patterns and adapting to novel expected patterns in
⁴ data. These systems are particularly vital in the context of Internet of Things
⁵ (IoT) devices that continuously stream high-fidelity data to control units.

⁶ In this rapidly evolving field with long-spanning roots, Chandola et al.
⁷ (2009) conducted an influential review of prior research efforts across diverse
⁸ application domains. Recent studies have underscored the need for holis-
⁹ tic and tunable anomaly detection methods accessible to operators (Laptev
¹⁰ et al., 2015; Kejariwal, 2015; Cook et al., 2020).

¹¹ Cook et al. denote substantial aspects that pose challenges to anomaly
¹² detection in IoT, including the temporal, spatial, and external context of
¹³ measurements, multivariate characteristics, noise, and nonstationarity (Cook
¹⁴ et al., 2020). To address these complexity issues, Zhang et al. (2024) have
¹⁵ successfully employed spatially distributed sensors and time-relative modu-
¹⁶ lation. Their approach has proven effective, particularly in the context of
¹⁷ complex non-linear systems, offering potential solutions to some of the chal-
¹⁸ lenges posed by IoT data. Huang et al., on the other hand, tackled the
¹⁹ problems of detecting global outliers, local outliers, and outlier clusters si-
²⁰ multaneously. Their proposed approach, based on density estimation, relies
²¹ on the notion that density distributions should exhibit minimal variations
²² in local areas. To achieve this, they introduce a novel turning ratio metric,
²³ which reduces reliance on hyperparameters and enhances anomaly detection
²⁴ (Huang et al., 2023).

²⁵ Additionally, feature engineering techniques play a crucial role in cap-
²⁶ turing contextual properties and enhancing anomaly detection performance
²⁷ (Fan et al., 2019). However, it is worth noting that feature engineering
²⁸ may introduce categorical variables and significantly increase the dimen-
²⁹ sionality of the data, requiring specific methods for handling large data, size-
³⁰ able data storage, and substantial computational resources (Talagala et al.,
³¹ 2021). Recently, Li et al. introduced an attribute-weighted outlier detection
³² algorithm, designed for high-dimensional datasets with mixtures of categor-
³³ ical and numerical data. Their approach assigns different weights to indi-
³⁴ vidual attributes based on their importance in anomaly detection and uses
³⁵ these weights to calculate distances between data points. Notably, Li et

al. demonstrated the superior performance of their algorithm compared to state-of-the-art methods (Li and Liu, 2024). Another strategy for handling high-dimensional data involves using deep learning methods with synthetic normal data to enhance the detection of outliers with subtle deviations, as proposed in Du et al. (2024).

Nevertheless, the presence of nonstationarity, often stemming from concept drift (a shift in data patterns due to changes in statistical distribution) and change points (permanent alterations in system state), presents a substantial challenge (Salehi and Rashidi, 2018). In practical scenarios, those changes tend to be unpredictable in both their spatial and temporal aspects. Consequently, they require systems with solid outlier rejection capabilities of intelligent tracking algorithms (Barbosa Roa et al., 2019). This underscores the critical importance of an anomaly detection method’s ability to adapt to evolving data structures, especially in long-term deployments. Nevertheless, as (Tartakovsky et al., 2013) remarked, immediate detection is not a feasible option unless there is a high tolerance for false alarms.

The adaptation of batch models at scale introduces a significant latency in detector adaptation (Wu et al., 2021). Incremental learning methods allowed adaptation while restraining the storage of the whole dataset. The supervised operator-in-the-loop solution offered by Pannu et al. (2012) showed the detector’s adaptation to data labeled on the flight. Others approached the problem as sequential processing of bounded data buffers in univariate signals (Ahmad et al., 2017) and multivariate systems (Bosman et al., 2015).

1.1. Related Work

Recent advances in anomaly detection have broadened its scope to include root cause identification governed by the development of explanatory methods capable of diagnosing and tracking faults across the system. Studies can be split into two groups of distinct approaches. The first group approaches explainability as the importance of individual features (Carletti et al., 2019; Nguyen et al., 2019; Amarasinghe et al., 2018). Those studies allow an explanation of novelty by considering features independently. The second group uses statistical learning creating models explainable via probability. For instance, the integration of variational Bayesian inference probabilistic graph neural network allowed Zhang et al. to model the posterior distribution of sensor dependency for gas leakage localization on unlabeled data (Zhang et al., 2023). Yang et al. recently proposed a Bayesian network (BN) for fault

72 detection and diagnosis. In this BN, individual nodes of the network represent
73 normally distributed variables, whereas the multiple regression model defines weights and relationships. Using the predefined structure of the BN,
74 the authors propose offline training with online detection and diagnosis (Yang et al., 2022).

75 Given the infrequent occurrence of anomalies and their potential absence in training data, the incorporation of synthetic data or feature extraction for various detected events emerges to assist diagnosis of the system. Brito et al. designed synthetic faults based on expert knowledge and introduced them into a transfer learning classifier to exploit faults in rotating machinery, with a subsequent explanation layer (Brito et al., 2023). Conversely, We et al. leveraged feature selection to expose various types of abnormal behavior. The team presents competitive performance while using change in relationships to provide causal inference (Wu et al., 2024).

76 However, it is crucial to underscore that offline training, as previously emphasized, is inherently inadequate when it comes to adapting to anticipated novel patterns, rendering it unsuitable for sustained, long-term operation on IoT devices.

77 This paper emphasizes the importance of combining adaptability in interpretable anomaly detection and proposes a method that addresses this challenge in real industrial systems. Here we report the discovery and characterization of an adaptive anomaly detection method for existing supervisory control and data acquisition (SCADA) systems, employing streaming IoT data. The ability to diagnose multivariate data while providing root cause isolation via statistical learning, extends our previous contribution to the field as presented in (Wadinger and Kvasnica, 2023). The proposed algorithm aims to represent a general method that aids a range of existing safety-critical systems where anomaly diagnosis and identification are paramount.
78 The schematic overview of the proposed method’s integration is presented in Figure 1.

79 1.2. Novelty of proposed approach

80 The idea of using statistical outlier detection is well-established. We highlight the impactful contributions of Yamanishi et al. in (Yamanishi and Takeuchi, 2002; Yamanishi et al., 2004). The authors propose a method for detecting anomalies in a time series. The method is based on the assumption that the continuous data is generated by a mixture of Gaussian distributions, while discrete data is modeled as histogram density. The authors solve the

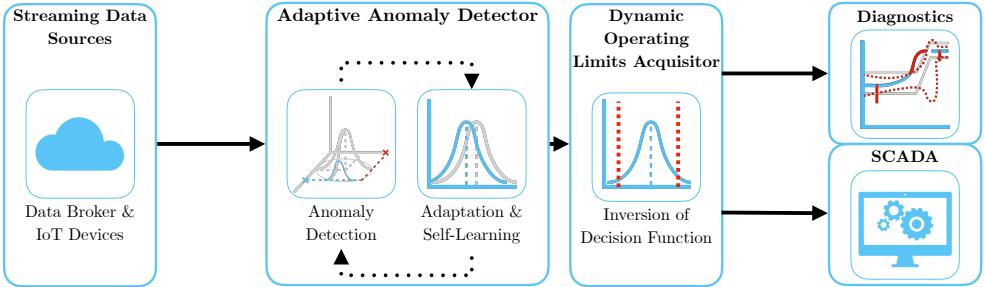


Figure 1: Schematic representation of the proposed method AID.

109 problem of change point detection as well. However, the adaptation sys-
 110 tem is unaware of such changes, making the moving window the only source
 111 of adaptation. Our self-supervised approach facilitates intelligent adaptation
 112 concerning detected change points, to increase the speed of adaptation where
 113 the probability of concept drift is high. By leveraging its ability to adapt to
 114 changes in operational states, our proposed method operates autonomously
 115 when such changes occur. Moreover, Yamanishi et al. (2004) does not at-
 116 tempt to isolate the root cause of the anomaly. Our approach extends statis-
 117 tical outlier detection by incorporating interpretability. This is achieved by
 118 evaluating the inverse cumulative distribution function of the latest condi-
 119 tional probabilities for each measurement, considering the remainder of the
 120 measurements, and establishing limits that define the threshold for normal
 121 event probabilities.

122 A limited number of studies have focused on adaptation and interpretabil-
 123 ity within the framework of anomaly detection. Two recent contributions in
 124 this area are made by Steenwinckel et al. as reported in (Steenwinckel, 2018;
 125 Steenwinckel et al., 2021). In Steenwinckel (2018), the authors emphasize
 126 the importance of combining prior knowledge with a data-driven approach
 127 to achieve interpretability, particularly concerning root cause isolation. They
 128 propose a novel approach that involves extracting features based on knowl-
 129 edge graph pattern extraction and integrating them into the anomaly de-
 130 tection mechanism. This graph is subsequently transformed into a matrix,
 131 and adaptive region-of-interest extraction is performed using reinforce-
 132 learning techniques. To enhance interpretability, a Generative Adversarial
 133 Network (GAN) reconstructs a new graphical representation based on se-
 134 lected vectors. However, it is important to note that the validation of this

idealized approach is pending further investigation. Lately, Steenwinckel et al. (2021) introduced a comprehensive framework for adaptive anomaly detection and root cause analysis in data streams. While the adaptation process is driven by user feedback, the specific mechanism remains undisclosed. The authors present an interpretation of their method through a user dashboard, featuring visualizations of raw data. This dashboard is capable of distinguishing between track-related problems and train-related issues, based on whether multiple trains at the same geographical location approach the anomaly. Meanwhile, our efforts are directed towards the development of a self-supervised method that can learn autonomously, reducing the reliance on human supervision, which is often constrained by time limitations and can lead to significant delays in adaptation. Our method is distinguished by its straightforward statistical reasoning and the ability to isolate the root cause of anomalies. The interpretability of our method is demonstrated through the establishment of dynamic operating limits for each signal, leveraging conditional probabilities derived from the signal and other system measurements and features. This provides operators with a clear understanding of the system's state and the underlying causes of anomalies and offers interoperability with existing alarm handling mechanisms in SCADA which utilize operating limits. To the best of our knowledge, this study appears to be one of the initial attempts to introduce a self-supervised approach for adaptive anomaly detection and root cause isolation in SCADA-based systems utilizing IoT data streams.

1.3. Validation

Two real-world industrial-scale case studies showcase that our proposed method has the capacity to explain anomalies, isolate the root cause, and allow adaptation to change points, allowing long-term deployment at the end users of energy storage systems. We observe similar detection performance, albeit with lower scalability, on benchmark data when comparing our approach to well-established unsupervised anomaly detection methods in streamed data which create a bedrock for many state-of-the-art contributions, such as One-Class SVM (Amer et al., 2013; Liu et al., 2014; Krawczyk and Woźniak, 2015; Miao et al., 2019; Gözüaçık and Can, 2021), and Half-Space Trees (Wetzig et al., 2019; Lyu et al., 2020).

169 *1.4. Practical Impact*

170 Potential applications of the proposed method are in the field of energy
171 storage systems, where the ability to detect anomalies and isolate their root
172 causes while adapting to changes in operation and environment, is crucial
173 for the system safety. The proposed method is designed to be integrated
174 into the existing infrastructure of the systems, utilizing IoT data streams on
175 top of well-established SCADA systems. SCADA systems continuously mon-
176 itor these process data in real-time, embodying alarm handling mechanisms,
177 which are designed to notify operators of the system's abnormal behavior
178 and drive attention to the root of the problem. By comparing the current
179 values to the upper and lower operating limits, they take action when a
180 variable exceeds or falls below these limits. However, safe operating limits
181 are often established based on a combination of equipment design limits and
182 the dynamics of the process (Stauffer and Chastain-Knight, 2021). Those
183 are indifferent to the actual state of the system and environmental condi-
184 tions. The proposed method allows the establishment of dynamic operating
185 limits, based on the current state of the system and its environment, with
186 direct utilization in SCADA systems expecting minimal intervention to exist-
187 ing infrastructure. This allows the system to operate closer or further from
188 its design limits, increasing its safety and profitability. The dynamic op-
189 erating limits allow operational metrics monitoring, making potential early
190 detection and prevention easier. Using general adaptable methods without
191 interpretability, on the other hand, may pose safety risks and lower total
192 financial benefits, as the triggered false alarms may need to be thoroughly
193 analyzed, resulting in prolonged downtimes.

194 The main contribution of the proposed solution to the developed body of
195 research is that it:

- 196 • Interpretable anomaly detector with self-supervised adaptation
- 197 • Demonstrates interpretability by providing dynamic operating limits
- 198 • Leverages self-learning approach on streamed IoT data
- 199 • Utilizes existing SCADA-based industrial infrastructure
- 200 • Offers faster response time to incidents due to root cause isolation

201 1.5. Paper Organization

202 The rest of the paper is structured as follows: We begin with the problem
203 and motivation in **Section 1**, providing context. Next, in **Section 2**, we
204 lay the theoretical groundwork. Our proposed adaptive anomaly detection
205 method is detailed in **Section 3**. We then demonstrate real-world industrial-
206 scale applications in **Section 4**. Finally, we conclude the paper in **Section 5**,
207 summarizing findings and discussing future research directions.

208 2. Preliminaries

209 In this section, we present the fundamental ideas that form the basis
210 of the developed approach. Subsection 2.1 explains Welford's online algo-
211 rithm, which can adjust distribution to changes in real-time. Subsection 2.2
212 proposes a two-pass implementation that can reverse the impact of expired
213 samples. The math behind distribution modeling in Subsection 2.3 estab-
214 lishes the foundation for the Gaussian anomaly detection model discussed in
215 Subsection 2.5, followed by conditional probability computation in Subsec-
216 tion 2.4. The last subsection of the preliminaries is devoted to the definition
217 of anomalies.

218 2.1. Welford's Online Algorithm

219 Welford introduced a numerically stable online algorithm for calculating
220 mean and variance in a single pass through data. Therefore, the algorithm
221 allows the processing of IoT device measurements without the need to store
222 their values (Welford, 1962).

223 Given measurement x_i where $i = 1, \dots, n$ is a sample index in sample
224 population n , the corrected sum of squares S_n is defined as

$$S_n = \sum_{i=1}^n (x_i - \bar{x}_n)^2, \quad (1)$$

225 with the running mean \bar{x}_n defined as previous mean \bar{x}_{n-1} weighted by pro-
226 portion of previously seen population $n - 1$ corrected by current sample as

$$\bar{x}_n = \frac{n-1}{n} \bar{x}_{n-1} + \frac{1}{n} x_n = \bar{x}_{n-1} + \frac{x_n - \bar{x}_{n-1}}{n}. \quad (2)$$

227 Throughout this paper, we consider the following formulation of an update
228 to the corrected sum of squares:

$$S_n = S_{n-1} + (x_n - \bar{x}_{n-1})(x_n - \bar{x}_n), \quad (3)$$

as it is less prone to numerical instability due to catastrophic cancellation, significant loss of precision due to subtracting two nearly equal numbers. Finally, the corresponding unbiased variance is

$$s_n^2 = \frac{S_n}{n - 1}. \quad (4)$$

This implementation of the Welford method requires the storage of three scalars: \bar{x}_{n-1} ; n ; S_n .

2.2. Inverting Welford's Algorithm

Based on (2), it is clear that the influence of the latest sample over the running mean decreases as the population n grows. For this reason, regulating the number of samples used for sample mean and variance computation has crucial importance over adaptation. Given access to the instances used for computation and expiration period $t_e \in \mathbb{N}_0^{n-1}$, reverting the impact of x_{n-t_e} can be written as follows

$$S_{n-1} = S_n - (x_{n-t_e} - \bar{x}_{n-1})(x_{n-t_e} - \bar{x}_n), \quad (5)$$

where the reverted mean is given as

$$\bar{x}_{n-1} = \frac{n}{n-1}\bar{x}_n - \frac{1}{n-1}x_{n-t_e} = \bar{x}_n - \frac{x_{n-t_e} - \bar{x}_n}{n-1}. \quad (6)$$

Finally, the unbiased variance follows the formula:

$$s_{n-1}^2 = \frac{S_{n-1}}{n-2}. \quad (7)$$

Notably, inversion allows the algorithm to keep a constant rate of adaptation at the cost of storing a bounded data buffer.

2.3. Statistical Model of Multivariate System

Multivariate normal distribution generalizes the multivariate systems to the model where the degree to which variables are related is represented by the covariance matrix. Gaussian normal distribution of variables is a reasonable assumption for process measurements, as it is a common distribution that arises from stable physical processes measured with noise (Mishra and Datta-Gupta, 2018). The general notation of multivariate normal distribution is:

$$\mathbf{X} \sim \mathcal{N}_k(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad (8)$$

253 where k -dimensional mean vector is denoted as $\boldsymbol{\mu} = (\bar{x}_1, \dots, \bar{x}_k)^T \in \mathbb{R}^k$
 254 and $\boldsymbol{\Sigma} \in \mathbb{R}^{k \times k}$ is the $k \times k$ covariance matrix, where k is the index of last
 255 random variable.

256 The probability density function (PDF) $f(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ of multivariate normal
 257 distribution is denoted as:

$$f(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{k/2} |\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}, \quad (9)$$

258 where \mathbf{x} is a k -dimensional vector of measurements x_i at time i , $|\boldsymbol{\Sigma}|$
 259 denotes the determinant of $\boldsymbol{\Sigma}$, and $\boldsymbol{\Sigma}^{-1}$ is the inverse of $\boldsymbol{\Sigma}$.

260 The cumulative distribution function (CDF) of a multivariate Gaussian
 261 distribution describes the probability that all components of the random
 262 vector \mathbf{X} take on a value less than or equal to a particular point q in space,
 263 and can be used to evaluate the likelihood of observing a particular set of
 264 measurements or data points. In other words, it gives the probability of
 265 observing a random vector that falls within a certain region of space. The
 266 standard notation of CDF is as follows:

$$F(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \int_{-\infty}^q f(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) d\mathbf{x}, \quad (10)$$

267 where $d\mathbf{x}$ denotes the integration over all k dimensions of \mathbf{x} .

268 As the equation (10) cannot be integrated explicitly, an algorithm for
 269 numerical computation was proposed in Genz (2000).

270 Given the PDF, we can also determine the value of \mathbf{x} that corresponds to a
 271 given quantile q using a numerical method for inversion of CDF (ICDF) often
 272 denoted as percent point function (PPF) or $F(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})^{-1}$. An algorithm that
 273 calculates the value of the PPF is part of standard statistical software tools.

274 2.4. Conditional Probability Distribution

275 Considering that we observe particular vector \mathbf{x}_i , we can update probability
 276 distributions, calculated according to the rules of conditional probability,
 277 of individual measurements within the vector given the rest of the measurements
 278 in \mathbf{x}_i . Let's assume multivariate normal distribution (8) and without
 279 loss of generality, that the vector \mathbf{x}_i can be partitioned into subset variable
 280 x_a , and complement vector \mathbf{x}_b as follows

$$\mathbf{x}_i = \begin{bmatrix} x_a \\ \mathbf{x}_b \end{bmatrix} \text{ with dimensions } \begin{bmatrix} 1 \times 1 \\ (k-1) \times 1 \end{bmatrix}, \quad (11)$$

281 where $a = 1, \dots, k$ and $\mathbf{b} = \{1, 2, \dots, k\}$ where $a \notin \mathbf{b}$. This partitioning
 282 allows us to define block-wise mean and covariance as follows:

$$\boldsymbol{\mu} = \begin{bmatrix} \mu_a \\ \boldsymbol{\mu}_{\mathbf{b}} \end{bmatrix} \text{ with dimensions } \begin{bmatrix} 1 \times 1 \\ (k-1) \times 1 \end{bmatrix}, \quad (12)$$

283 and

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_{aa}^2 & \boldsymbol{\Sigma}_{ab} \\ \boldsymbol{\Sigma}_{ba} & \boldsymbol{\Sigma}_{bb} \end{bmatrix} \text{ with dimensions } \begin{bmatrix} 1 \times 1 & 1 \times (k-1) \\ (k-1) \times 1 & (k-1) \times (k-1) \end{bmatrix}. \quad (13)$$

284 Subsequently, we can derive the conditional distribution of any subset
 285 variable x_a , given the complementary vector \mathbf{x}_b . This conditional distribution
 286 conforms to a univariate normal distribution, characterized by:

$$X_a | \mathbf{X}_b \sim \mathcal{N}(\mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2). \quad (14)$$

287 where $\mu_{a|\mathbf{b}}$ denotes the conditional mean and $\sigma_{a|\mathbf{b}}^2$ represents the conditional variance.
 288 These crucial parameters can be computed by applying the Schur complement as follows:
 289

$$\sigma_{a|\mathbf{b}}^2 = \sigma_{aa}^2 - \boldsymbol{\Sigma}_{ab}\boldsymbol{\Sigma}_{bb}^{-1}\boldsymbol{\Sigma}_{ba}, \quad (15)$$

290 for the conditional variance $\sigma_{a|\mathbf{b}}^2$, while the conditional mean, denoted as
 291 $\mu_{a|\mathbf{b}}$, is determined by:

$$\mu_{a|\mathbf{b}} = \mu_a + \boldsymbol{\Sigma}_{ab}\boldsymbol{\Sigma}_{bb}^{-1}(\mathbf{x}_b - \boldsymbol{\mu}_b). \quad (16)$$

292 The conditional variance $\sigma_{a|\mathbf{b}}^2$ essentially represents the Schur complement
 293 of $\boldsymbol{\Sigma}_{bb}$ within the overall covariance matrix $\boldsymbol{\Sigma}$.

294 2.5. Gaussian Anomaly Detection

295 From a viewpoint of statistics, outliers are commonly denoted as values
 296 that significantly deviate from the mean. Under the assumption that the
 297 spatial and temporal characteristics of a system, observed over a moving
 298 window, can be suitably represented as normally distributed features, we
 299 assert that any anomaly can be identified as an outlier.

300 In empirical fields like machine learning, the three-sigma rule (3σ) provides
 301 a framework for characterizing the region of a distribution within which
 302 normal values are expected to fall with high confidence. This rule renders
 303 approximately 0.265% of values in the distribution as anomalous.

304 The 3σ rule establishes the probability that any sample x_a of a random
 305 vector X falls within a given CDF over a semi-closed interval as the distance
 306 from the conditional mean $\mu_{a|\mathbf{b}}$ of 3 conditional variances $\sigma_{a|\mathbf{b}}^2$ and gives an
 307 approximate value of q as

$$q = P\{|x_a - \mu_{a|\mathbf{b}}| < 3\sigma_{a|\mathbf{b}}^2\} = 0.99735. \quad (17)$$

308 Utilizing a probabilistic model of normal behavior, we can determine
 309 threshold values x_l and x_u corresponding to the closed interval of the CDF
 310 where this probability is established. The inversion of Equation (10) facili-
 311 tates this calculation, yielding:

$$x_l = F((1 - P\{|x_a - \mu_{a|\mathbf{b}}| < 3\sigma_{a|\mathbf{b}}^2\}); \mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2)^{-1}, \quad (18)$$

312 for the lower limit, and

$$x_u = F((P\{|x_a - \mu_{a|\mathbf{b}}| < 3\sigma_{a|\mathbf{b}}^2\}); \mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2)^{-1}, \quad (19)$$

313 for the upper limit. These lower and upper limits together form vectors
 314 \mathbf{x}_l and \mathbf{x}_u , respectively, defining the region of normal system operation. This
 315 region is conceptualized as a hypercube in the feature space, with each di-
 316 mension bounded by the corresponding feature limits, as computed using
 317 Equations (18) and (19) for all $a = 1, \dots, k$; $\mathbf{b} = \{1, 2, \dots, k\}$ where $a \notin \mathbf{b}$.
 318 The approximation of a confidence ellipse as a hypercube can be employed
 319 to represent the region of normal system operation for individual variables
 320 of a multivariate system, rendering it as an aid for visual representation.

321 The predicted state of the system, denoted as y_i , and the normality of
 322 signals $\mathbf{y}_{s,i}$ at time i are determined based on the maximum distance of
 323 observations from the center of the probabilistic density. The center of the
 324 probabilistic density corresponds to the vector of conditional means $\mu_{a|\mathbf{b}}$ with
 325 respect to other features. The calculation of this distance involves the cumu-
 326 lative distribution function (CDF) of observations and conditional distribu-
 327 tions, as follows:

$$F(x_a; \mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2) : a = 1, \dots, k; \mathbf{b} = \{1, 2, \dots, k\} \text{ where } a \notin \mathbf{b}. \quad (20)$$

328 Subsequently, operation states of individual inputs are defined as follows:

$$\mathbf{y}_{s,i} = \begin{cases} 0 & \text{if } T \leq (20) \\ 1 & \text{if } T > (20), \end{cases} \quad (21)$$

329 where T represents a threshold that distinguishes between normal signal
330 measurement ($\mathbf{y}_{s,i} = 0$) and abnormal ($\mathbf{y}_{s,i} = 1$).

331 For the overall abnormality of the system, any anomaly in signals $\mathbf{y}_{s,i}$ is
332 considered, resulting in:

$$y_i = \begin{cases} 1 & \text{if } 1 \in \mathbf{y}_{s,i} \\ 0 & \text{otherwise,} \end{cases} \quad (22)$$

333 defining the discrimination boundary between system operation where
334 $y_i = 0$ indicates normal system operation, and $y_i = 1$ indicates anomalous
335 operation.

336 2.6. Anomaly Definition

337 This subsection provides an overview of the definition of anomalies in
338 data analysis and their categorization, setting conventions for this paper.

339 In the realm of data analysis, anomalies are conspicuous deviations from
340 the anticipated patterns within a dataset. Traditionally, the task of anomaly
341 detection has relied upon unsupervised methodologies, wherein the identifi-
342 cation of "outliers" entails the comparison of data points in both temporal
343 and spatial contexts. This approach, often referred to as point-wise anomaly
344 detection, classifies a data point as an anomaly when it exhibits significant
345 dissimilarity from its neighboring data points (Iglesias Vázquez et al., 2023).

346 The concept of point anomalies, influenced by factors such as temporal
347 and spatial aspects, can be further categorized into conditional and contex-
348 tual anomalies (Ruff et al., 2021).

349 Nevertheless, this conventional method may not be suitable for scenarios
350 characterized by collective anomalies, where clusters of abnormal data points
351 coexist. A more pragmatic approach defines anomalies as deviations from
352 established "normal" patterns, resembling the principles of semi-supervised
353 learning. Change point detection, in a similar vein, can be regarded as a
354 relative approach that takes into account the varying dynamics of changes,
355 whether they occur gradually or abruptly (Iglesias Vázquez et al., 2023).

356 It is imperative to recognize that the interpretation of anomalies, outliers,
357 and novelties can vary upon the application. Anomalies typically garner
358 significant attention, while outliers are often treated as undesirable noise
359 and are typically excluded during data preprocessing. Novelties, on the other
360 hand, signify new observations that necessitate model updates to adapt to
361 an evolving environment (Ruff et al., 2021).

362 Notwithstanding the differences in terminology, methods employed for the
363 identification of data points residing in low-probability regions, irrespective of
364 whether they are referred to as "anomaly detection," "outlier detection," or
365 "novelty detection," share fundamental similarities (Iglesias Vázquez et al.,
366 2023).

367 **3. Adaptive Anomaly Detection and Interpretation Framework**

368 In this section, we present an adaptive and interpretable detection frame-
369 work (AID) designed for SCADA-based industrial systems with streaming
370 IoT devices. Our approach is rooted in the foundational concepts discussed
371 in Preliminaries 2. We systematically leverage these theoretical building
372 blocks to introduce our method in a coherent manner.

373 Our approach begins by modeling the system as a dynamic multivariate
374 normal distribution, allowing it to effectively handle pervasive nonstationary
375 effects and interactions that impact industrial processes. We address several
376 critical factors, such as change points, concept drift, and seasonal effects.
377 Our primary contribution is the integration of an adaptable self-supervised
378 system with root cause identification and dynamic operating limits setting.
379 This unique combination empowers our online statistical model to diagnose
380 anomalies through three distinct mechanisms.

381 Firstly, we employ conditional probability calculations to assess the nor-
382 mality of the system’s operating conditions. This step ensures that our
383 method identifies outliers within individual signal measurements and inter-
384 prets the root causes of anomalies, facilitating faster and more precise diag-
385 noses. Secondly, we detect abrupt changes due to concept drift, serving for
386 faster adaptation to new operating conditions without human intervention.
387 Thirdly, we harness interpretability as a tool to establish dynamic operating
388 limits. These adaptive limits enable our framework to seamlessly integrate
389 with existing SCADA-based infrastructure, a substantial advantage over ex-
390 isting solutions.

391 We have structured the subsequent sections to delve into the details of our
392 proposed methodology by the logical flow of data. The upcoming subsection
393 will cover the anomaly detection mechanism, followed by sections on online
394 training and adaptation. The next subsection will describe dynamic operat-
395 ing limits setting, followed by diagnostic capabilities. Lastly, we describe how
396 those parts converge into a diagnostic tool. For a schematic representation of
397 our proposed method, with a highlighted subsection attribution, please refer
398 to Figure 2. For a concise technical representation of our proposed method,
399 please refer to Algorithm 1.

400 *3.1. Online detection*

401 In the online detection phase, AID distinguishes between normal and
402 anomalous observations based on the model of the system’s normal behavior.

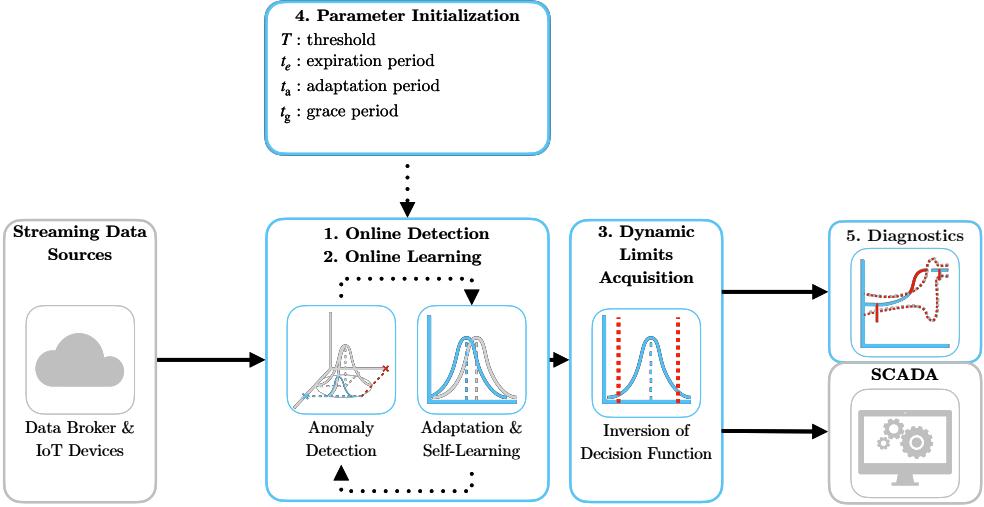


Figure 2: Schematic representation of the proposed method AID with parameter initialization. Colored boxes represent steps described within the subsection.

403 The detection pipeline is event-triggered upon the arrival of a new set of
 404 measurements.

405 To initiate the process, AID computes the properties of the conditional
 406 distribution based on the current observations given the dynamic joint nor-
 407 mal distribution. These calculations are performed for each element of the
 408 process observation vector \mathbf{x}_i at time instance i . Specifically, we calculate the
 409 conditional mean using (16) and the conditional variance using (15) for ele-
 410 ments of \mathbf{x}_i . These computations yield univariate conditional distributions
 411 for individual signals and features. These conditional distributions play a
 412 crucial role in assessing the abnormality of signals and features concerning
 413 their relationships with other elements of \mathbf{x}_i . Consequently, AID inherently
 414 considers the interactions between input signals and features.

415 The determination of anomalous behavior is influenced by the parame-
 416 ter T , which is a user-defined hyperparameter representing a probabilistic
 417 threshold that sets the boundary between normal and anomalous behavior.
 418 Details regarding the selection of an appropriate value for T are discussed
 419 in Subsection 3.5. Whenever an anomaly is detected within one of the sig-
 420 nals or features, it triggers an alert regarding the overall system's anomalous
 421 behavior, as described in (22). Nevertheless, individual determinations of

422 anomalies serve as a diagnostic tool for isolating the root causes of anomalies,
423 as further discussed in Subsection 3.4.

424 The proposed mechanism is applicable to both point anomalies and collective
425 anomalies. In the case of collective anomalies, their duration and deviation
426 may serve as precursors to concept drift in the system. To identify concept drift,
427 we introduce a parameter adaptation period t_a . Given the predicted system anomaly
428 state from (22) as y_i over a window of past observations $\mathbf{y}_i = \{y_{i-t_a}, \dots, y_i\}$ bounded by t_a ,
429 the following test determines anticipated change points:
430

$$\frac{\sum_{y \in \mathbf{y}_i} y}{n(\mathbf{y}_i)} > T. \quad (23)$$

431 Here, $n(\mathbf{y}_i)$ denotes the dimensionality of \mathbf{y}_i . The logic behind (23) is
432 that over an adaptation period t_a , change points can be distinguished from
433 collective anomalies and point anomalies due to their minimum duration,
434 while T allows for some overlap with previous normal conditions.

435 Our framework anticipates unexpected novel behavior, including non-uniformities
436 in sampling. Assuming that the distribution of sampling times
437 remains stable over the long term, we can employ equivalent steps on the
438 observed time between samples to discriminate signal loss from long-term
439 anomalous network events.

440 3.2. Online learning

441 AID's training process follows an incremental self-learning approach, allowing
442 for online model updates as new samples arrive. Self-learning, in this context,
443 focuses on selecting only relevant data for training to maintain the
444 model's long-term relevancy and stability. This approach proves particularly
445 valuable in handling streaming data, where human supervision can introduce
446 significant computational delays, affecting response time in a sequential
447 setting.

448 In online anomaly detector training, regardless of the type of supervision,
449 the learning is typically built upon observations of the normal state.
450 We introduce a grace period denoted as t_g to enable model calibration in
451 the initial stages after deployment. During this period, when normality in
452 samples is expected, the model learns from all observations. Subsequently,
453 self-supervised and unsupervised detectors are expected to make autonomous
454 decisions.

455 However, in the case of industrial systems, the drifts in the concept might
456 often render the normal state anomalous, slowing down or preventing adap-
457 tation completely. This is particularly true for the case of seasonal effects,
458 where the system is expected to operate in a different mode for a certain
459 period of time. To address this issue, AID’s adaptation incorporates two
460 self-supervised mechanisms.

461 Firstly, the model is updated if the observation at time instance i is
462 marked normal in the detection phase. In the case of a dynamic multivari-
463 ate probability distribution, the updated parameters are μ_i and Σ_i at time
464 instance i . Update of the mean vector μ_i and covariance matrix Σ_i is gov-
465 erned by Welford’s online algorithm using equation (2) and (4) respectively.
466 Samples beyond the expiration period t_e , discussed further in Subsection 3.5,
467 are disregarded during the second pass. The effect of expired samples is
468 reverted using inverse Welford’s algorithm for mean (6) and variance (7),
469 accessing the data in the bounded internal buffer. For more details, refer to
470 Subsection 2.2.

471 The second mechanism, which enables adaptation to anomalous samples,
472 relies on changepoint detection. This mechanism operates under the as-
473 sumption that detected changepoints represent new operational states with
474 limited overlap with the previous ones, as specified in Equation 23. It facili-
475 tates rapid adaptation to evolving data patterns without the need for human
476 intervention. The selection of the adaptation period t_a , as discussed further
477 in Subsection 3.5, is thus crucial for determining the speed of adaptation or
478 the potential mitigation of the second adaptation mechanism.

479 To anticipate potential deviations from sampling uniformity, we calculate
480 the cumulative distribution function (CDF) over the univariate normal dis-
481 tribution of sampling. We operate under the assumption that, over the long
482 term, the distribution of sampling times remains stable, employing a one-
483 pass update mechanism of (2) and (4), for efficiency. To proactively detect
484 subtle changes in sampling patterns, self-supervised learning is employed,
485 leveraging anomalies weighted by the deviation from $(1 - F(x_i; \mu, \sigma^2))$ for
486 training.

487 3.3. *Dynamic limits acquisition*

488 As we wrote in the subsection 1.4 Practical Impact, the monitoring mech-
489 anisms of SCADA readily depend on the upper and lower operating limits
490 of individual parameters of the system. In the case of industrial systems,

491 these limits are often defined by the sensor's designed limits and the sys-
492 tem dynamics. These limits are typically static and do not account for the
493 dynamically changing conditions. Our proposed method AID is capable of
494 setting dynamic operating limits, thus allowing integration into the existing
495 SCADA-based infrastructure.

496 The threshold T applied on the dynamic multivariate normal distribution
497 creates a confidence hyperellipse at T probability level. Such a hyperellipse
498 would not allow to effectively bound individual signals as it depends on val-
499 ues that other jointly distributed variables take. Nevertheless, by computing
500 the conditional for process observation vector \mathbf{x}_i at time instance i , we can
501 compute the conditional density function for individual signals. By applying
502 threshold T on individual conditional probabilities, we establish a hyper-
503 cube defined by lower and upper threshold values, denoted as \mathbf{x}_l and \mathbf{x}_u ,
504 respectively. These thresholds are derived from (18) and (19), incorporating
505 updated model parameters. Lower and upper thresholds play a pivotal role
506 as dynamic operating limits. They may be used as an addition to static op-
507 erating limits used by monitoring systems in SCADA, accounting for spatial
508 factors, such as multipoint measurements, temporal factors, such as aging,
509 and actual environmental conditions that influence sensor operation. More-
510 over, any violation of the limits is also detected as an anomaly.

511 *3.4. Diagnostics*

512 One of the crucial aspects of diagnostics is root cause isolation. Using the
513 ability to detect anomalies in individual signals and features, AID is capable
514 of isolating the root cause of anomalies with consideration of their mutual
515 relationships. This is achieved by computing the conditional probability of
516 individual signals and features given the rest of the process observation vec-
517 tor \mathbf{x}_i at time instance i . The dynamic process limits further enhance the
518 diagnosis by providing the context of the anomaly, including the extent of
519 deviation from normal operation and the direction of the deviation. The
520 proposed diagnostic mechanism is particularly useful in the case of collec-
521 tive anomalies, where the unified direction of deviations is expected. AID's
522 interpretability is an asset for domain experts to understand why certain
523 anomalies are flagged and enables operators to assess the system's state by
524 visualizing limits and deviations, thus detecting the speed at which the pro-
cess variable approaches the limits before an anomaly occurs.

526 3.5. Model Parameters Initialization

527 The model initialization is governed by defining two required hyperparameters of the model: the expiration period (t_e) and the threshold (T). The
528 expiration period determines the window size for time-rolling computations,
529 impacting the proportion of outliers within a given timeframe and directly in-
530 fluencing the relaxation (with a longer expiration period) or tightening (with
531 a shorter expiration period) of dynamic signal limits. Additionally, we intro-
532 duce a grace period t_g , which defaults to $uite$, allowing for model calibration.
533 During this grace period, system anomalies are not flagged to prevent false
534 positives and speed up self-supervised learning, introduced in Subsection 3.2.
535 t_g can take any value smaller than $uite$, if the detection must be delivered fast
536 after intergration. The length of the expiration period inversely correlates
537 with the model’s ability to adapt to sudden changes. The adaptation and
538 detection of significant drifts in the data-generating process, such as changes
539 in central tendency, is managed through the adaptation period t_a . A shorter
540 t_a results in faster adaptation to new operating conditions, while making the
541 system vulnerable to prolonged collective anomalies. A longer t_a results in
542 slower adaptation to significantly deviating new operations, but allows longer
543 alerts regarding collective anomalies. In most cases, $t_a = 1/4t_e$ offers optimal
544 performance.

546 As a general rule of thumb, expiration period t_e should be determined
547 based on the slowest observed dynamics within the multivariate system. The
548 threshold T defaults to the three-sigma probability of q in (17). Adjusting
549 this threshold can fine-tune the trade-off between precision and recall. A
550 lower threshold boosts recall but may lower precision, while a higher thresh-
551 old enhances precision at the cost of recall. We recommend starting with
552 the default values of other parameters and making adjustments based on
553 real-time model performance, as the model’s interpretability can reduce the
554 time and effort required for fine-tuning. The presence of one non-default
555 interpretable hyperparameter facilitates quick adaptation of AID in a broad
556 range of use cases.

Algorithm 1 Online Detection and Identification Workflow

Input: expiration period t_e

Output: system anomaly y_i , signal anomalies $\mathbf{y}_{s,i}$, sampling anomaly $y_{t,i}$, change-point $y_{c,i}$, lower thresholds $\mathbf{x}_{l,i}$, upper thresholds $\mathbf{x}_{u,i}$,

Initialisation :

- 1: $i \leftarrow 1; n \leftarrow 1; T \leftarrow (17); \boldsymbol{\mu} \leftarrow \mathbf{x}_0; \boldsymbol{\Sigma} \leftarrow \mathbf{1}_{k \times k}; \mu_t \leftarrow 0; \sigma_t^2 \leftarrow 1;$
 - 2: compute $F(\mathbf{x}_0; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ using algorithm in Genz (2000);
LOOP Process
 - 3: **loop**
 - 4: $\mathbf{x}_i, t_i \leftarrow \text{RECEIVE}();$
 - 5: $\mathbf{y}_{s,i} \leftarrow \text{PREDICT}(\mathbf{x}_i, T)$ using (21);
 - 6: $y_i \leftarrow \text{PREDICT}(\mathbf{y}_{s,i})$ using (22);
 - 7: $\mathbf{x}_{l,i}, \mathbf{x}_{u,i} \leftarrow \text{GET}(T, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ using (18), (19);
 - 8: $y_{t,i} \leftarrow \text{PREDICT}(t_i - t_{i-1})$ using (21);
 - 9: $\mu_t, \sigma_t^2 \leftarrow \text{UPDATE}(t_i - t_{i-1}, \mu_t, \sigma_t^2)$ using (2), (4);
 - 10: **if** (22) = 0 **or** (23) **then**
 - 11: $\boldsymbol{\mu}, \boldsymbol{\Sigma} \leftarrow \text{UPDATE}(\mathbf{x}_i, \boldsymbol{\mu}, \boldsymbol{\Sigma}, n)$ using (2), (4);
 - 12: **if** (23) **then**
 - 13: $y_{c,i} \leftarrow 1;$
 - 14: **else**
 - 15: $y_{c,i} \leftarrow 0;$
 - 16: **end if**
 - 17: $n \leftarrow n + 1;$
 - 18: **for** \mathbf{x}_{i-t_e} **do**
 - 19: $\boldsymbol{\mu}, \boldsymbol{\Sigma} \leftarrow \text{REVERT}(\mathbf{x}_{i-t_e}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, n)$ using (6), (7);
 - 20: $n \leftarrow n - 1;$
 - 21: **end for**
 - 22: **end if**
 - 23: $i \leftarrow i + 1;$
 - 24: **end loop**
-

557 **4. Case Study**

558 This section presents two case studies on real industrial-scale energy stor-
559 ages and a real data benchmark to demonstrate the effectiveness and appli-
560 cability of our proposed approach. We investigate the properties and perfor-
561 mance of the approach using signals from IoT devices in an energy system
562 and streamed benchmark system data. The successful deployment demon-
563 strates that this approach is suitable for existing industrial systems utilizing
564 IoT data streams on top of well-established SCADA systems.

565 *4.1. Battery Energy Storage System TERRA*

566 In the first case study, we demonstrate our proposed method on real
567 industrial-scale battery energy storage system (BESS) TERRA, depicted in
568 Fig 3. TERRA has an installed capacity of 151 kWh distributed among 10
569 modules with 20 Li-ion NMC cells. The Inverter’s nominal power is 100 kW.
570 The TERRA reports measurements of State of Charge (SoC), supply/draw
571 energy set-points, and inner temperature, at 6 positions (channels) of each
572 battery module. A substantial size of the system, which is 2.4x2.4x1.2m
573 (HxWxD), requires a proper cooling mechanism. The cooling is handled by
574 forced air from the HVAC system and inner fans, while the fire safety system
575 is passive. Tight battery temperature control is needed to optimize perfor-
576 mance and maximize the safety and battery’s lifespan. Identifying anomalous
577 events and removal of corrupted data might yield significant improvement in
578 the process control level and increase the reliability and stability of the sys-
579 tem.

580 The AID is integrated into the existing software infrastructure of the
581 system, allowing detection and diagnosis of the system using streamed IoT
582 data. Here we replay a 9-day stream of historical measurements of the device,
583 to demonstrate key features of AID.

584 For demonstration purposes, the expiration period t_e is set to 4 days, as
585 the system is expected to adapt to the new behavior, due to the transfer of
586 the module to the outside. The grace period was reduced to 1 day, to observe
587 the reaction to concept drift. The threshold T is set to 3.5σ to reduce the
588 number of alarms. The frequency will be higher as the detector is protected
589 and self-supervised. The adaptation period t_a is changed to 3 hours as this
590 is the time constant of the temperature to the unit change of supply/draw
591 power demand.



Figure 3: Photograph of the actual studied TERRA energy storage unit with open doors (left), and closed doors (right).

592 Figure 4 depicts the average cell temperature measurement of the TERRA
 593 for all 10 modules. The data are normalized to the range [0, 1] to protect
 594 the sensitive business value. The light red area represents the region out of
 595 dynamic operating limits as provided by AID. On 7th March 2022, the system
 596 was relocated from the inside of the building to the outside power socket. The
 597 system was expected to adapt to the new behavior within 4 days as specified
 598 by t_e . Nevertheless, due to the protection of the model from learning the
 599 anomalous data, the new behavior could not be captured as the system was
 600 not operating within the safe limits. The adaptation started three days later,
 601 as only some of the measurements within the safe region after transfer were
 602 learned. Therefore, the importance of self-supervised adaptation to changes
 603 in data is crucial. As we can see, the change points detection according to
 604 (23) alerted such change shortly after the TERRA was connected to a data
 605 broker, while the length of the adaptation period enabled discrimination from
 606 collective anomaly.

607 In Figure 5 we depict the same measurement with a changepoint adap-
 608 tation mechanism in place. The mechanism speeds up the adaptation to the
 609 new behavior, as the system is allowed to learn from anomalous data when
 610 they represent the changed behavior. The adaptation took approximately 6
 611 times shorter.

612 The default sampling rate of the incoming signal measurements is 1
 613 minute. However, network communication of the IoT devices is prone to

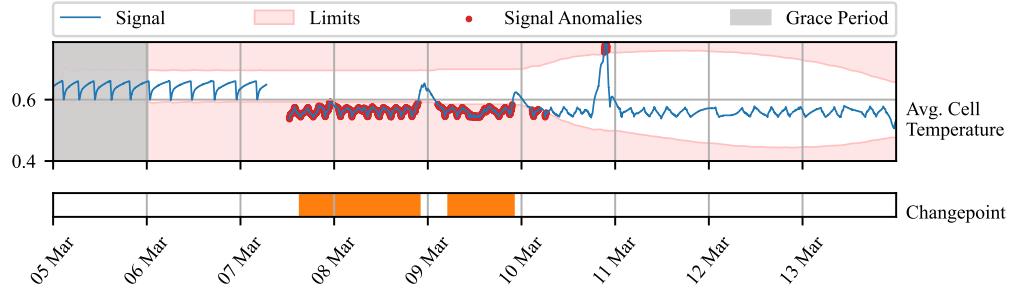


Figure 4: Time series of TERRA measurements observed over 9 days (blue line). The y-axis renders the average temperature of all cells and modules after the normalization to the range of [0, 1]. The light red area represents an area out of dynamic operating limits for individual signals. Observations out of the limits are marked by a red dot. Orange bars represent the times, at which changepoints were detected. Grace period is grayed out.

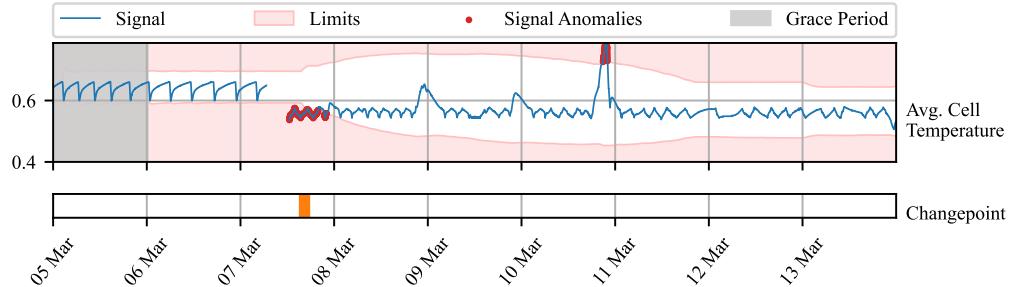


Figure 5: Time series of TERRA measurements observed over 9 days (blue line). The y-axis renders the average temperature of all cells and modules after the normalization to the range of [0, 1]. The light red area represents an area out of dynamic operating limits for individual signals. Observations out of the limits are marked by a red dot. Orange bars represent the times, at which changepoints were detected. Grace period is grayed out.

614 packet dropout, which results in unexpected non-uniformities in sampling
 615 from the perspective of the SCADA system. The transfer of TERRA was
 616 accompanied by the disconnection of IoT sensors from the data broker which
 617 might be considered an anomaly. The system can detect such anomalies as
 618 well, as depicted in Figure 6. Along with known disconnection, the system
 619 alerted two more non-uniformities of shorter extend, scaled in the figure for
 620 better visibility. The short loss of signal was caused by the packet drop, as
 621 it impacted only a few consecutive measurements. Various confidence levels
 622 could be used to further analyze and map potential causes to the duration
 623 of the outage.

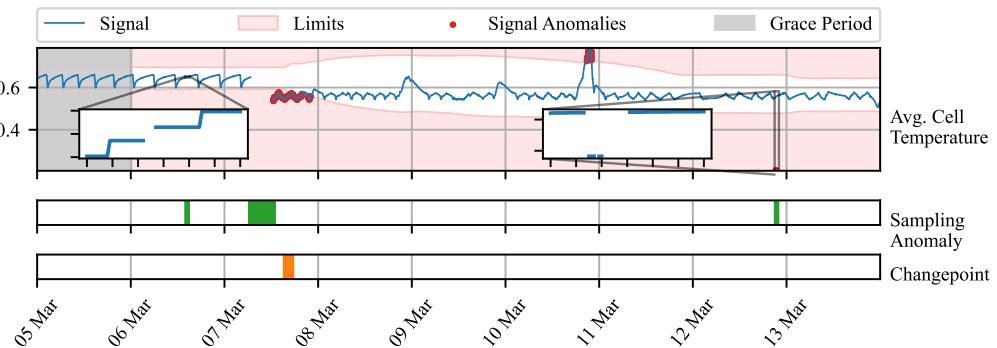


Figure 6: Time series of TERRA measurements observed over 9 days (blue line). The y-axis renders the average temperature of all cells and modules after the normalization to the range of [0, 1]. The light red area represents an area out of dynamic operating limits for individual signals. Observations out of the limits are marked by a red dot. Orange bars represent the times, at which changepoints were detected. Green bars represent periods where sampling anomaly was alerted. Grace period is grayed out.

624 Lastly, we want to acknowledge the outlier, left uncaptured due to in-
 625 creased variance of the distribution in a period of adaptation. Observing
 626 multiple variables, where some might be influenced less by the change in be-
 627 havior, might be beneficial in such cases. The industrial partner provided a
 628 physics-based model of the battery module temperature, defined as follows:

$$\begin{aligned}
 T_{\text{bat},i+1} = & T_{\text{bat},i} + T_s(q_{\text{fan}} V_{\text{b,max}} \rho c_p (T_{\text{out}} - T_{\text{bat},i}) + V_{\text{c,max}} q_{\text{circ,fan}} \rho c_p T_{\text{bat},i} \\
 & + q_{\text{circ,fan}} (P_{\text{cool}} q_{\text{cool}} P_{\text{heat}} q_{\text{heat}}) + c_{\text{scale}} Q_{\text{bat}} + q_{\text{inner fans}} \\
 & - (V_{\text{b,max}} q_{\text{fan}} V_{\text{c,max}} q_{\text{circ,fan}}) \rho c_p T_{\text{bat},i}) / (m_{\text{bat}} c_{\text{p,b}})
 \end{aligned} \tag{24}$$

When combined with an averaged measurement of battery module temperature, we could compute the difference between real and predicted temperature. Such deviation can be useful in detecting unexpected patterns in temperature due to the impact of external disturbance and aging. Nevertheless, it may be inaccurate as the physics-based model is simplified and does not account for spatial aspects, like temperature gradients as well as different dynamic effects of charging and discharging on temperature. For instance, in Fig. 3 during the first two days we see, that the cooling dynamic is not captured well, resulting in a subtle positive difference between average cell temperature and the temperature predicted by the model. In combination with the raw measured average of the temperature, the AID captures the outlier on 9th March which could not be captured in a univariate setting. The physics-based model exposes temporal aspects of the behavior as it considers the dynamics of its inputs. The rapid increase in temperature w.r.t the modeled dynamics due to environmental conditions will draw a sharp positive peak in the difference between the real and predicted temperature, which will slowly vanish. Based on the significance of the deviation, the peak will be notified as a single-point anomaly or collective anomaly.

This case study demonstrated AID’s effectiveness within the context of the energy storage system, specifically the TERRA system. The AID system exhibited adaptability to changes in the operational environment, contributing to its versatility and robustness. Additionally, it facilitated the establishment of dynamic operating limits for SCADA systems, considering context of the device such as environmental conditions or aging. Furthermore, the AID system showcased its capability to operate with a physics-based model, enhancing the precision of anomaly detection processes. This highlights the potential of AID as a valuable tool within complex industrial systems. The validity of our proposed approach was verified by our industrial partner, who confirmed that the detected anomalies were indeed caused by the aforementioned events.

4.2. Kokam Battery Module

A second case study presents temperature profile monitoring of individual modules of battery pack TERRA deployed at the premises of the end user. During the operation, a hardware fault of module’s 9 cooling fan occurred on 23rd August 2023 at 17:12:30. Our industrial partner was interested in finding out, whether such an event could be captured by an anomaly detection system. Each of the 10 modules, embodies 20 cells measured by 6 spatially

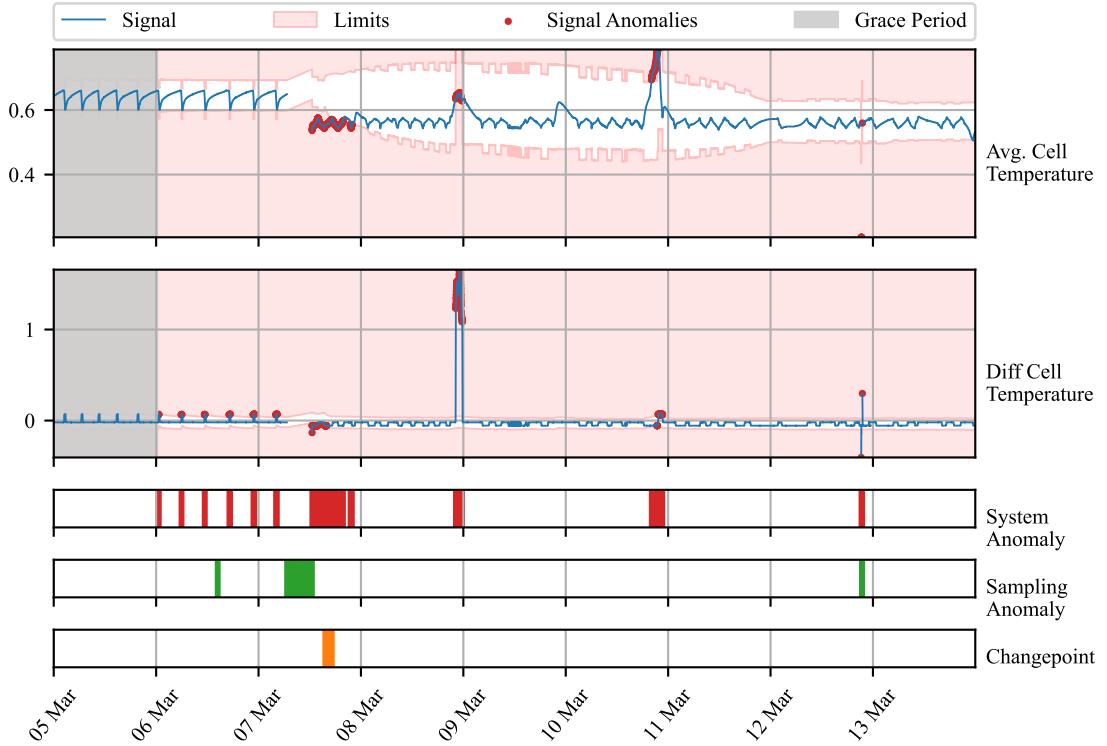


Figure 7: Time series of TERRA measurements observed over 9 days (blue line). The y-axis renders the average temperature of all cells and modules after the normalization to the range of $[0, 1]$. The light red area represents an area out of dynamic operating limits for individual signals. Observations out of the limits are marked by a red dot. Orange bars represent the times, at which changepoints were detected. Green bars represent periods where sampling anomaly was alerted. Red bars denote the period where any of the signals contained anomaly. Grace period is grayed out.

666 distributed sensors as shown in Figure 8. The measurements are sent in 30-
 667 second intervals and processed in a streamed manner by SCADA. With the
 668 availability of the temperature profiles for all the modules, we computed the
 669 deviation of the observed value from the average of all the modules' temper-
 670 ature measurements. The ground truth information about the fan fault was
 671 provided to the best of the operator's knowledge. However, this information
 672 serves for evaluation only, as the system operates in a self-supervised manner.

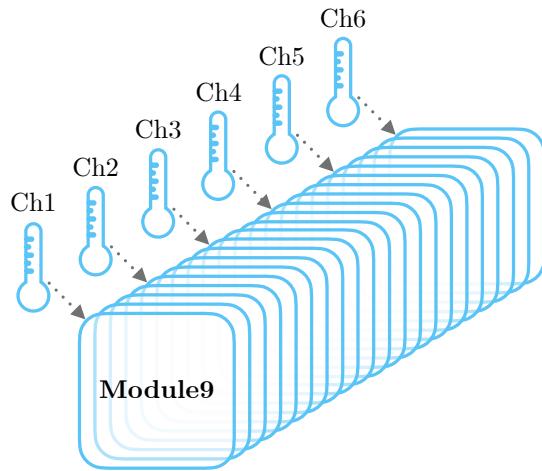


Figure 8: Module 9 with 20 cells and 6 sensors measuring the temperature at each 4th cell.

673 Our anomaly detection system was, in this case, initialized for the op-
 674 eration in production. The expiration period of 7 days, allowed the system
 675 to adapt to weekly seasonality due to the usage of the battery following
 676 work week. The grace period was kept at the default value, equal to t_e . The
 677 threshold value was shifted to a 4 sigma value of 99.977% which makes the the
 678 frequency of anomalous events approximately once a week given 30-second
 679 sampling. The adaptation period was held constant as the deployed system
 680 is not expected to change its behavior dramatically on a daily basis.

681 In Figure 9 we observe 4 days of operation around the period of fan
 682 fault occurrence. The deviations between the observed temperature mea-
 683 sured by channels of module 9 and the average temperature of all modules
 684 are displayed. The dynamic operating limits tightly envelop temperatures
 685 measured by the sensors in the middle of the module (refer to Figure 8),

686 while measurements at both sides deviate more due to the proximity to the
687 walls and sources of disturbance. We observed multiple alarms raised by var-
688 ious channels individually before the fan fault. These anomalies, while not
689 addressed here further, could be subjects of interest for further investigation
690 by system operators. Meanwhile, the fan fault at the center of our focus is
691 alarmed based on three measurements, namely channels 1, 2, and 3. From
692 the zoomed views, we can observe a sharp increase in the temperature devia-
693 tion. The alarm is on until 24th August at noon, when significant fluctuations
694 vanish followed by temporary settling of the temperature. On 25th August
695 at 11:21, increased temperature fluctuations are followed by an increase of
696 temperature similar to the initial one. AID alerts this fault again based on
697 measurements by channels 1, 2, and 3.

698 Time series of TERRA measurements observed over 9 days (blue line).
699 The y-axis renders the average temperature of all cells and modules after the
700 normalization to the range of [0, 1]. The light red area represents an area out
701 of dynamic operating limits for individual signals. Observations out of the
702 limits are marked by a red dot. Orange bars represent the times, at which
703 changepoints were detected. Green bars represent periods where sampling
704 anomaly was alerted. Red bars denote the period where any of the signals
705 contained anomaly. Grace period is grayed out.

706 Interestingly, during the presence of a fault in the fan, two more peri-
707 ods where the fan started operating again followed as depicted in Figure 10.
708 Periods of operation were interrupted again on 27th and 28th August respec-
709 tively in the early morning hours. In both of the cases, AID detected the
710 presence of the fault at the moment of occurrence. In the first case, channel
711 3 reported an anomaly slightly before the increase in temperature, due to
712 abnormal fluctuation happening prior to faults.

713 This case study demonstrates the effectiveness of the AID framework in
714 identifying hardware faults within the context of energy storage systems. It
715 showcases the system’s ability to harness spatially distributed sensors that
716 measure the same process variable. The AID system successfully pinpointed
717 a fault in a cooling fan during real-world production operations, underlining
718 its practical utility and its relevance in enhancing the safety of energy storage
719 systems. Furthermore, the incorporation of adaptation mechanisms ensures
720 that the system can be deployed over extended periods without necessitating
721 resource-intensive retraining. Additionally, the concept of dynamic operating
722 limits introduced in this study holds promise for integration with Supervisory
723 Control and Data Acquisition (SCADA) monitoring systems, enabling proac-

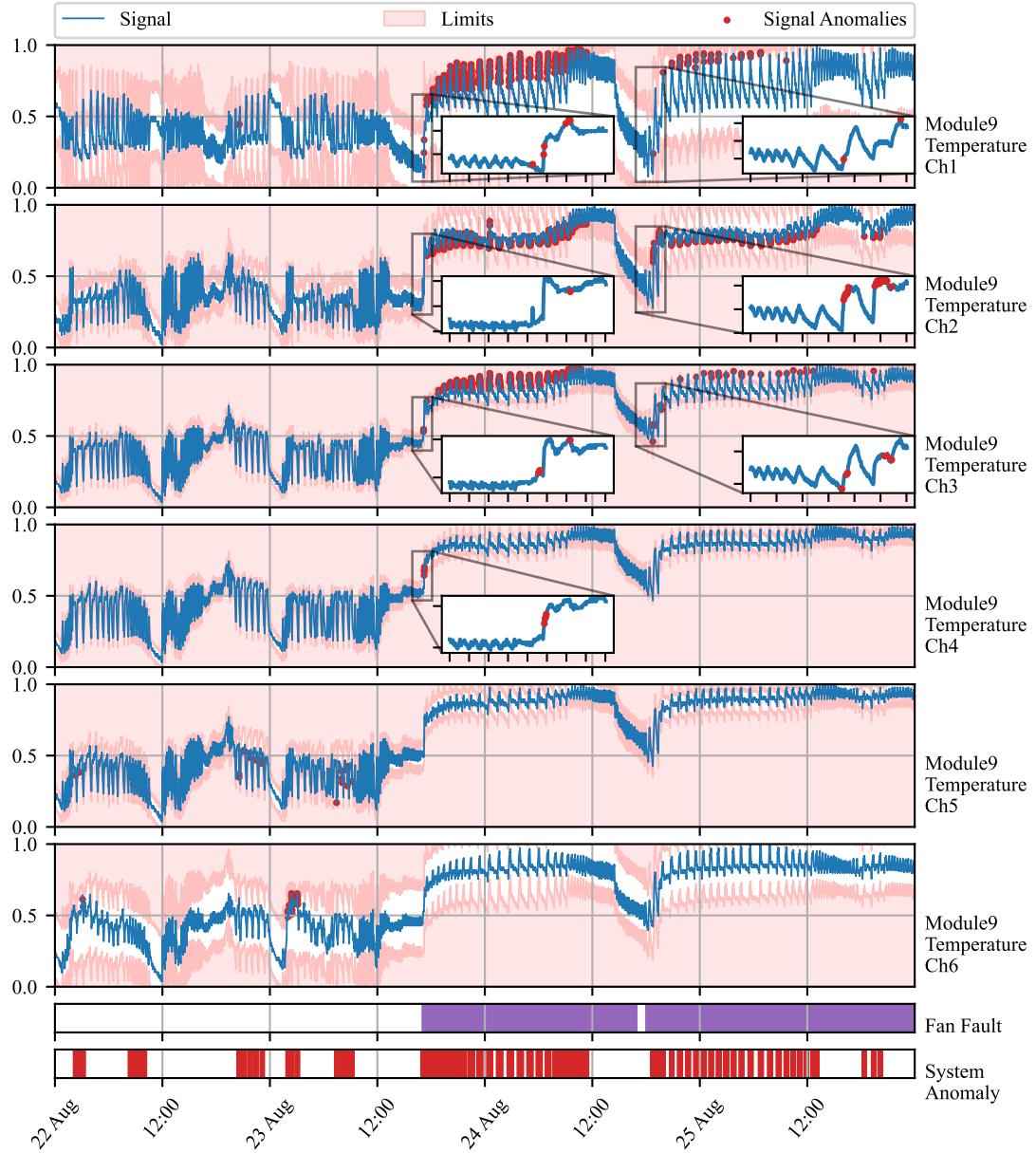


Figure 9: Time series of battery module 9 measurements (blue line). The y-axis renders the normalized deviations of temperature from the average of all modules. Signal anomalies are marked as red dots. The light red area represents an area out of dynamic operating limits. True fan faults are marked by purple bars. Green bars represent the times of changepoint detection. Red bars denote the period where any signal anomaly.

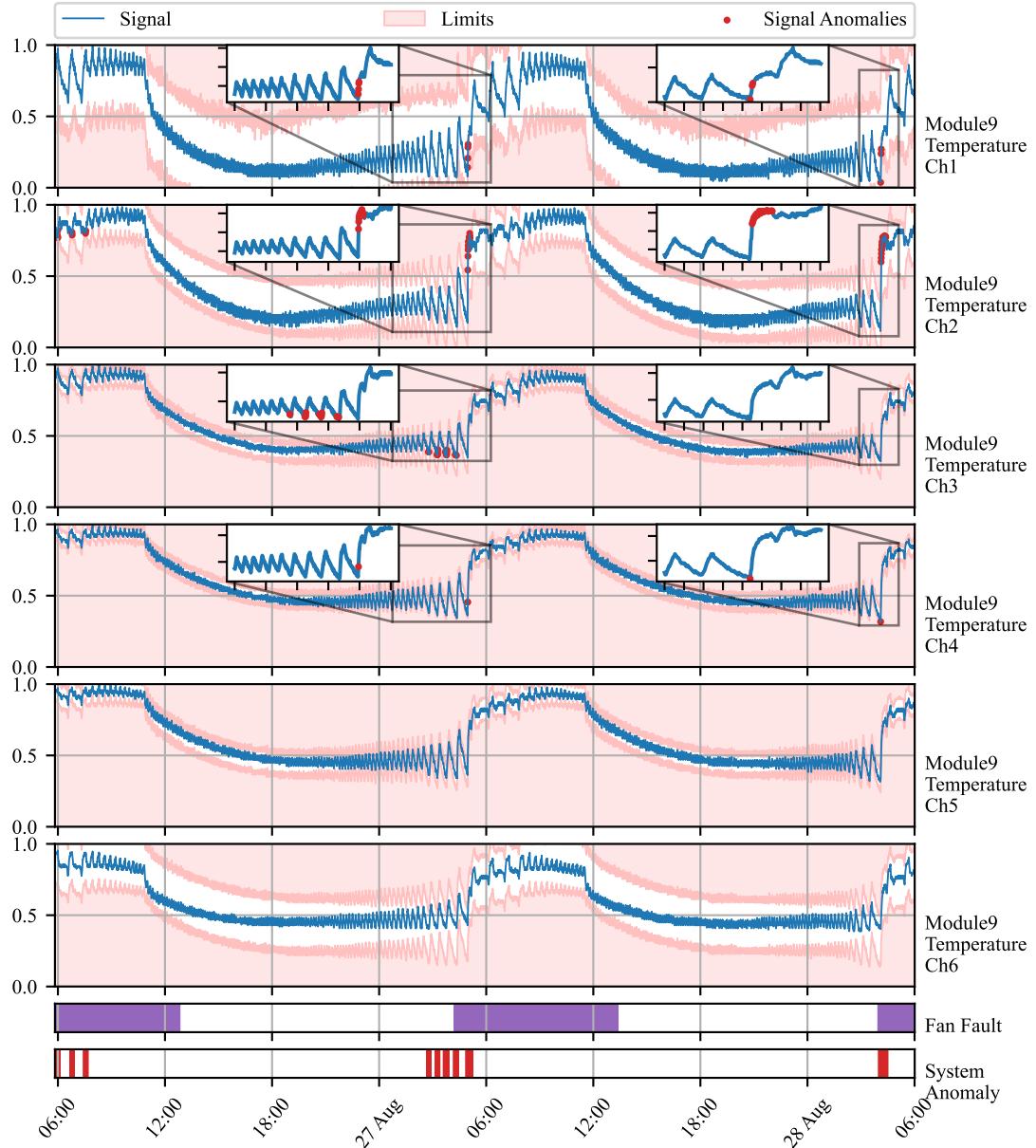


Figure 10: Time series of battery module 9 measurements (blue line). The y-axis renders the normalized deviations of temperature from an average of all modules. Signal anomalies are marked as red dots. The light red area represents an area out of dynamic operating limits. True fan faults are marked by purple bars. Green bars represent the times of changepoint detection. Red bars denote the period where any of the signals contained anomaly.

724 tive responses in situations where human life, equipment, or the environment
725 may be at risk.

726 *4.3. Real Data Benchmark*

727 The benchmarking comparison in this subsection evaluates the AID frame-
728 work against adaptive unsupervised detection methods, specifically One-
729 Class Support Vector Machine (OC-SVM) and Half-Space Trees (HS-Trees).
730 These methods are widely recognized for their iterative learning capabilities
731 on multivariate time-series data, making them suitable for anomaly detection
732 in dynamic systems, as previously discussed in the Introduction 1.3.

733 The comparison is based on the Skoltech Anomaly Benchmark (SKAB)
734 dataset, a real-world dataset with annotated labels distinguishing between
735 anomalous and normal observations (Katser and Kozitsin, 2020). SKAB
736 is used for this purpose, as no established benchmarking multivariate data
737 were found regarding energy storage systems similar to the ones studied in
738 Subsection 4.1 and Subsection 4.2. The SKAB dataset involves experiments
739 related to rotor imbalance, where various control actions and changes in
740 water volume are introduced to the system. It encompasses eight features
741 and exhibits both gradual and sudden drifts.

742 To ensure fairness in the benchmark, data preprocessing adheres to best
743 practices for each method. OC-SVM employs standard scaling, while HS-
744 Trees use normalization. Our proposed AID method requires no scaling.
745 Preprocessing is performed online, simulating a real production environment,
746 with running mean and variance for standard scaling and running peak-to-
747 peak distance for normalization, as supported by the online machine learning
748 library "river" (Montiel et al., 2021).

749 The optimal hyperparameters for both reference methods are found us-
750 ing Bayesian Optimization. Due to no further knowledge about the data
751 generating process, and equity in benchmark, the hyperparameters of our
752 proposed method were optimized using Bayesian Optimization as well. 20
753 steps of random exploration with 100 iterations of Bayesian Optimization
754 were used, increasing default values set in the Bayesian Optimization library,
755 to allow thorough exploration and increase the possibility of finding global
756 optima in each case (Nogueira, 2014). The hyperparameters are optimized
757 with the F1 score as a cost function first, to maximize both precision and
758 recall on anomalous samples.

759 As adaptation is required and anticipated within benchmark datasets,
760 the performance is evaluated iteratively, similarly to the operation after de-

761 ployment. The metric is updated with each new sample and its final value is
 762 used to drive Bayesian Optimization. The performance is evaluated using the
 763 best-performing model, found by Bayesian Optimization. The performance
 764 of the proposed method is evaluated on the same data as the models are
 765 optimized for.

766 Hyperparameter search ranges are specified, with values centered around
 767 default library values for OC-SVM and HS-Trees. The ranges are intention-
 768 ally set wide to facilitate comprehensive exploration. The quantile filter
 769 threshold used in OC-SVM and HS-Trees aligns with the threshold used in
 770 AID. These hyperparameter ranges are presented in Table 1.

Table 1: Hyperparameter Ranges for Detection Algorithms

Algorithm	Hyperparameters	Default	Ranges
AID	Threshold	0.99735	(0.85, 0.99994)
	t_e	-	(150, 10000)
	t_a	t_e	(50, 2000)
	t_g	t_e	(50, 1000)
OC-SVM	Threshold	-	(0.85, 0.99994)
	Learning Rate	0.01	(0.005, 0.02)
HS-Trees	Threshold	-	(0.85, 0.99994)
	N Trees	10	(0, 20)
	Max Height	8	(2, 14)
	Window Size	250	(100, 400)

771 The results for models optimized for the F1 score are summarized in Ta-
 772 ble 2, which includes precision, recall, F1 score, and average latency. Macro
 773 values are enclosed in brackets, representing the mean of the metric for both
 774 anomalies and normal data. A perfect detection achieves 100% in each met-
 775 ric. According to the Scoreboard for various algorithms on SKAB’s Kaggle
 776 page, all iterative approaches perform comparably to the batch-trained iso-
 777 lation forest and autoencoder, validating the optimization process. Notably,
 778 the proposed AID method outperforms both reference methods in terms of
 779 F1 score, recall, and precision, despite having a 30-fold higher latency per
 780 sample. This highlights the scalability as a candidate for further develop-
 781 ment. Nevertheless, in this case, sampling of the benchmark data still offers
 782 enough time to deliver predictions with sufficient frequency.

783 Optimal hyperparameters found during Bayesian Optimization are de-
 784 tailed in Table 3. None of the parameters are at the edge of the provided

Table 2: Evaluation of models optimized for F1 score on SKAB dataset (Katser and Kozitsin, 2020). The best-performing model is highlighted in bold. Values in brackets represent macro values of the metric.

Algorithm	Precision [%]	Recall [%]	F1 [%]	Avg. Latency [ms]
AID	41 (59)	80 (59)	54 (53)	1.45
HS-Trees	36 (51)	74 (51)	48 (44)	0.05
OC-SVM	39 (54)	63 (54)	48 (52)	0.05

ranges, serving as necessary proof of ranges being broad enough. Nevertheless, sufficient proof is not possible as multiple parameter ranges are not bounded by designed limits.

Table 3: Optimal hyperparameters of methods optimized for F1 score

Algorithm	Hyperparameters	Found
AID	Threshold	0.96442
	t_e	1136
	t_a	396
	t_g	546
OC-SVM	Threshold	0.86411
	Learning Rate	0.01956
HS-Trees	Threshold	0.99715
	N Trees	1
	Max Height	7
	Window Size	283

5. Conclusion

In this paper, we demonstrate the capacity of adaptive conditional probability distribution to model the normal operation of dynamic systems employing streaming IoT data and isolate the root cause of anomalies. AID dynamically adapts to non-stationarity by updating multivariate Gaussian distribution parameters over time. Additionally, self-supervision enhances the model by protecting it from the effects of outliers and increasing the speed of adaptation in response to autonomously detected changes in operation.

797 Our statistical model isolates the root causes of anomalies as extreme
798 deviations from the conditional means vector, considering spatial and tem-
799 poral effects encoded in features, as demonstrated in our case studies. This
800 approach establishes the system’s operational state by analyzing the dis-
801 tribution of signal measurements, computing the distance from the mean
802 of conditional probability, and setting dynamic operating limits based on
803 multivariate distribution parameters. Additionally, the detector alerts for
804 non-uniform sampling due to packet drops and sensor malfunctions. These
805 adaptable limits can be seamlessly integrated into SCADA architecture, en-
806 hancing context awareness and enabling plug-and-play compatibility with
807 existing infrastructure.

808 The ability to detect and identify anomalies in the system, isolate the
809 root cause of anomaly to specific signal or feature, and identify signal losses
810 is shown in two case studies on data from operated industrial-scale energy
811 storages. These case studies highlight the model’s ability to adapt, diagnose
812 the root cause of anomalies, and leverage both physics-based models and
813 spatially distributed sensors. Unlike many anomaly detection approaches,
814 the proposed AID method does not require historical data or ground truth
815 information about anomalies, alleviating the general limitations of detection
816 methods employed in the energy industry.

817 The benchmark performed on industrial data indicates that our model
818 provides comparable results to other self-learning adaptable anomaly detec-
819 tion methods. This is an important property of our model, as it also allows
820 for root cause isolation.

821 AID represents a significant advancement in the safety and profitability
822 of evolving systems that utilize well-established SCADA architecture and
823 streaming IoT data. By providing dynamic operating limits, AID seamlessly
824 integrates with existing alarm mechanisms commonly employed in SCADA
825 systems. To the best of our knowledge, this study appears to be one of the
826 initial attempts to introduce a self-supervised approach for adaptive anomaly
827 detection and root cause isolation in SCADA-based systems utilizing IoT
828 data streams.

829 Future work on this method will include improvements to the change point
830 detection mechanism, reduction in latency for high-dimensional data, and
831 minimizing the false positive rate, which is a challenge for general plug-and-
832 play models. We will also explore the ability to operate with non-parametric
833 models, in contrast to Gaussian distribution.

834 **Additional information**

835 Our framework is openly accessible on GitHub at the following URL:
836 https://github.com/MarekWadinger/online_outlier_detection.

837 **CRediT authorship contribution statement**

838 **Marek Wadinger:** Conceptualization; Data curation; Formal analysis;
839 Investigation; Methodology; Resources; Software; Validation; Visualization;
840 Writing - original draft; and Writing - review & editing. **Michal Kvasnica:**
841 Conceptualization; Funding acquisition; Project administration; Resources;
842 Supervision; Validation.

843 **Declaration of Competing Interest**

844 The authors declare that they have no known competing financial inter-
845 ests or personal relationships that could have appeared to influence the work
846 reported in this paper.

847 **Acknowledgements**

848 This work was supported by the Horizon Europe [101079342]; the Slovak
849 Research and Development Agency [APVV-20-0261]; and the Scientific Grant
850 Agency of the Slovak Republic [1/0490/23].

851 **References**

- 852 Ahmad, S., Lavin, A., Purdy, S., Agha, Z., 2017. Unsuper-
853 vised real-time anomaly detection for streaming data. Neuro-
854 computing 262, 134–147. URL: <https://www.sciencedirect.com/science/article/pii/S0925231217309864>, doi:<https://doi.org/10.1016/j.neucom.2017.04.070>. online Real-Time Learning Strategies for
856 Data Streams.
858 Amarasinghe, K., Kenney, K., Manic, M., 2018. Toward explainable deep
859 neural network based anomaly detection, in: 2018 11th International Con-
860 ference on Human System Interaction (HSI), pp. 311–317. doi:[10.1109/HSI.2018.8430788](https://doi.org/10.1109/HSI.2018.8430788).

- 862 Amer, M., Goldstein, M., Abdennadher, S., 2013. Enhancing one-class sup-
863 port vector machines for unsupervised anomaly detection, in: Proceedings
864 of the ACM SIGKDD Workshop on Outlier Detection and Description,
865 Association for Computing Machinery, New York, NY, USA. pp.
866 8–15. URL: <https://doi.org/10.1145/2500853.2500857>, doi:10.1145/
867 2500853.2500857.
- 868 Barbosa Roa, N., Travé-Massuyès, L., Grisales-Palacio, V.H., 2019. Dy-
869 clee: Dynamic clustering for tracking evolving environments. Pattern
870 Recognition 94, 162–186. URL: <https://www.sciencedirect.com/science/article/pii/S0031320319301992>, doi:<https://doi.org/10.1016/j.patcog.2019.05.024>.
- 873 Bosman, H.H., Iacca, G., Tejada, A., Wörtche, H.J., Liotta, A., 2015. En-
874 sembles of incremental learners to detect anomalies in ad hoc sensor net-
875 works. Ad Hoc Networks 35, 14–36. URL: <https://www.sciencedirect.com/science/article/pii/S1570870515001481>, doi:<https://doi.org/10.1016/j.adhoc.2015.07.013>. special Issue on Big Data Inspired Data
876 Sensing, Processing and Networking Technologies.
- 879 Brito, L.C., Susto, G.A., Brito, J.N., Duarte, M.A.V., 2023. Fault diag-
880 nosis using explainable ai: A transfer learning-based approach for ro-
881 tating machinery exploiting augmented synthetic data. Expert Systems
882 with Applications 232, 120860. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423013623>, doi:<https://doi.org/10.1016/j.eswa.2023.120860>.
- 885 Carletti, M., Masiero, C., Beghi, A., Susto, G.A., 2019. Explainable machine
886 learning in industry 4.0: Evaluating feature importance in anomaly detec-
887 tion to enable root cause analysis, in: 2019 IEEE International Conference
888 on Systems, Man and Cybernetics (SMC), pp. 21–26. doi:10.1109/SMC.
889 2019.8913901.
- 890 Chandola, V., Banerjee, A., Kumar, V., 2009. Anomaly detection: A sur-
891vey. ACM Comput. Surv. 41. URL: <https://doi.org/10.1145/1541880.1541882>, doi:10.1145/1541880.1541882.
- 893 Cook, A.A., Misirlı, G., Fan, Z., 2020. Anomaly detection for iot time-
894 series data: A survey. IEEE Internet of Things Journal 7, 6481–6494.
895 doi:10.1109/JIOT.2019.2958185.

- 896 Du, X., Chen, J., Yu, J., Li, S., Tan, Q., 2024. Generative adversarial nets
897 for unsupervised outlier detection. Expert Systems with Applications 236,
898 121161. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423016639>, doi:<https://doi.org/10.1016/j.eswa.2023.121161>.
- 901 Fan, C., Sun, Y., Zhao, Y., Song, M., Wang, J., 2019. Deep learning-
902 based feature engineering methods for improved building energy predic-
903 tion. Applied Energy 240, 35–45. URL: <https://www.sciencedirect.com/science/article/pii/S0306261919303496>, doi:<https://doi.org/10.1016/j.apenergy.2019.02.052>.
- 906 Genz, A., 2000. Numerical computation of multivariate normal probabili-
907 ties. Journal of Computational and Graphical Statistics 1. doi:[10.1080/10618600.1992.10477010](https://doi.org/10.1080/10618600.1992.10477010).
- 909 Gözüaçık, Ö., Can, F., 2021. Concept learning using one-class classi-
910 fiers for implicit drift detection in evolving data streams. Artificial
911 Intelligence Review 54, 3725–3747. URL: <https://doi.org/10.1007/s10462-020-09939-x>, doi:[10.1007/s10462-020-09939-x](https://doi.org/10.1007/s10462-020-09939-x).
- 913 Huang, J., Cheng, D., Zhang, S., 2023. A novel outlier detecting algorithm
914 based on the outlier turning points. Expert Systems with Applications 231,
915 120799. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423013015>, doi:<https://doi.org/10.1016/j.eswa.2023.120799>.
- 918 Iglesias Vázquez, F., Hartl, A., Zseby, T., Zimek, A., 2023. Anomaly detec-
919 tion in streaming data: A comparison and evaluation study. Expert Sys-
920 tems with Applications 233, 120994. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423014963>, doi:<https://doi.org/10.1016/j.eswa.2023.120994>.
- 923 Katser, I.D., Kozitsin, V.O., 2020. Skoltech anomaly benchmark
924 (skab). <https://www.kaggle.com/dsv/1693952>. doi:[10.34740/KAGGLE/DSV/1693952](https://doi.org/10.34740/KAGGLE/DSV/1693952).
- 926 Kejariwal, A., 2015. Introducing practical and ro-
927 bust anomaly detection in a time series. URL:

- 928 https://blog.twitter.com/engineering/en_us/a/2015/introducing-practical-and-robust-anomaly-detection-in-a-time-series.
- 930 Krawczyk, B., Woźniak, M., 2015. One-class classifiers with incremental learning and forgetting for data streams with concept drift. Soft Computing 19, 3387–3400. URL: <https://doi.org/10.1007/s00500-014-1492-5>.
- 934 Laptev, N., Amizadeh, S., Flint, I., 2015. Generic and scalable framework for automated time-series anomaly detection, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery, New York, NY, USA. pp. 1939–1947. URL: <https://doi.org/10.1145/2783258.2788611>.
- 940 Li, J., Liu, Z., 2024. Attribute-weighted outlier detection for mixed data based on parallel mutual information. Expert Systems with Applications 236, 121304. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423018067>, doi:<https://doi.org/10.1016/j.eswa.2023.121304>.
- 945 Liu, B., Xiao, Y., Yu, P.S., Cao, L., Zhang, Y., Hao, Z., 2014. Uncertain one-class learning and concept summarization learning on uncertain data streams. IEEE Transactions on Knowledge and Data Engineering 26, 468–484. doi:[10.1109/TKDE.2012.235](https://doi.org/10.1109/TKDE.2012.235).
- 949 Lyu, Y., Li, W., Wang, Y., Sun, S., Wang, C., 2020. Rmhsforest: Relative mass and half-space tree based forest for anomaly detection. Chinese Journal of Electronics 29, 1093–1101. doi:<https://doi.org/10.1049/cje.2020.09.010>.
- 953 Miao, X., Liu, Y., Zhao, H., Li, C., 2019. Distributed online one-class support vector machine for anomaly detection over networks. IEEE Transactions on Cybernetics 49, 1475–1488. doi:[10.1109/TCYB.2018.2804940](https://doi.org/10.1109/TCYB.2018.2804940).
- 956 Mishra, S., Datta-Gupta, A., 2018. Chapter 3 - distributions and models thereof, in: Mishra, S., Datta-Gupta, A. (Eds.), Applied Statistical Modeling and Data Analytics. Elsevier, pp. 31–67. URL: <https://www.sciencedirect.com/science/article/pii/B9780128032794000031>, doi:<https://doi.org/10.1016/B978-0-12-803279-4.00003-1>.

- 961 Montiel, J., Halford, M., Mastelini, S.M., Bolmier, G., Sourty, R., Vaysse,
962 R., Zouitine, A., Gomes, H.M., Read, J., Abdessalem, T., Bifet, A., 2021.
963 River: machine learning for streaming data in python. *Journal of Ma-*
964 *chine Learning Research* 22, 1–8. URL: <http://jmlr.org/papers/v22/20-1380.html>.
- 966 Nguyen, Q.P., Lim, K.W., Divakaran, D.M., Low, K.H., Chan, M.C., 2019.
967 Gee: A gradient-based explainable variational autoencoder for network
968 anomaly detection, in: 2019 IEEE Conference on Communications and
969 Network Security (CNS), pp. 91–99. doi:10.1109/CNS.2019.8802833.
- 970 Nogueira, F., 2014. Bayesian Optimization: Open source constrained
971 global optimization tool for Python. URL: <https://github.com/fmfn/BayesianOptimization>.
- 973 Pannu, H.S., Liu, J., Fu, S., 2012. Aad: Adaptive anomaly detection system
974 for cloud computing infrastructures, in: 2012 IEEE 31st Symposium on
975 Reliable Distributed Systems, pp. 396–397. doi:10.1109/SRDS.2012.3.
- 976 Ruff, L., Kauffmann, J.R., Vandermeulen, R.A., Montavon, G., Samek, W.,
977 Kloft, M., Dietterich, T.G., Müller, K.R., 2021. A unifying review of deep
978 and shallow anomaly detection. *Proceedings of the IEEE* 109, 756–795.
979 doi:10.1109/JPROC.2021.3052449.
- 980 Salehi, M., Rashidi, L., 2018. A survey on anomaly detection in evolving
981 data: [with application to forest fire risk prediction]. *SIGKDD Explor.*
982 *Newsl.* 20, 13–23. URL: <https://doi.org/10.1145/3229329.3229332>,
983 doi:10.1145/3229329.3229332.
- 984 Stauffer, T., Chastain-Knight, D., 2021. Do not let your safe oper-
985 ating limits leave you s-o-l (out of luck). *Process Safety Progress*
986 40, e12163. URL: <https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/prs.12163>,
987 doi:<https://doi.org/10.1002/prs.12163>,
988 arXiv:<https://aiche.onlinelibrary.wiley.com/doi/pdf/10.1002/prs.12163>.
- 989 Steenwinckel, B., 2018. Adaptive anomaly detection and root cause analy-
990 sis by fusing semantics and machine learning, in: Gangemi, A., Gentile,
991 A.L., Nuzzolese, A.G., Rudolph, S., Maleshkova, M., Paulheim, H., Pan,
992 J.Z., Alam, M. (Eds.), *The Semantic Web: ESWC 2018 Satellite Events*,
993 Springer International Publishing, Cham. pp. 272–282.

- 994 Steenwinckel, B., De Paepe, D., Vanden Hautte, S., Heyvaert, P., Bente-
995 frit, M., Moens, P., Dimou, A., Van Den Bossche, B., De Turck, F.,
996 Van Hoecke, S., Ongena, F., 2021. Flags: A methodology for adap-
997 tive anomaly detection and root cause analysis on sensor data streams
998 by fusing expert knowledge with machine learning. Future Generation
999 Computer Systems 116, 30–48. URL: <https://www.sciencedirect.com/science/article/pii/S0167739X20329927>, doi:<https://doi.org/10.1016/j.future.2020.10.015>.
- 1002 Talagala, P.D., Hyndman, R.J., Smith-Miles, K., 2021. Anomaly
1003 detection in high-dimensional data. Journal of Computational
1004 and Graphical Statistics 30, 360–374. URL: <https://doi.org/10.1080/10618600.2020.1807997>,
1005 doi:<https://doi.org/10.1080/10618600.2020.1807997>,
1006 arXiv:<https://doi.org/10.1080/10618600.2020.1807997>.
- 1007 Tartakovsky, A.G., Polunchenko, A.S., Sokolov, G., 2013. Efficient computer
1008 network anomaly detection by changepoint detection methods. IEEE Jour-
1009 nal of Selected Topics in Signal Processing 7, 4–11. doi:[10.1109/JSTSP.2012.2233713](https://doi.org/10.1109/JSTSP.2012.2233713).
- 1011 Wadinger, M., Kvasnica, M., 2023. Real-time outlier detection with dynamic
1012 process limits, in: 2023 24th International Conference on Process Control
1013 (PC), pp. 138–143. doi:[10.1109/PC58330.2023.10217717](https://doi.org/10.1109/PC58330.2023.10217717).
- 1014 Welford, B.P., 1962. Note on a method for calculating corrected sums of
1015 squares and products. Technometrics 4, 419–420. doi:[10.1080/00401706.1962.10490022](https://doi.org/10.1080/00401706.1962.10490022).
- 1017 Wetzig, R., Gulenko, A., Schmidt, F., 2019. Unsupervised anomaly alert-
1018 ing for iot-gateway monitoring using adaptive thresholds and half-space
1019 trees, in: 2019 Sixth International Conference on Internet of Things: Sys-
1020 tems, Management and Security (IOTSMS), pp. 161–168. doi:[10.1109/IOTSMS48152.2019.8939201](https://doi.org/10.1109/IOTSMS48152.2019.8939201).
- 1022 Wu, H., He, J., Tömösközi, M., Xiang, Z., Fitzek, F.H., 2021. In-network
1023 processing for low-latency industrial anomaly detection in softwarized net-
1024 works, in: 2021 IEEE Global Communications Conference (GLOBECOM),
1025 pp. 01–07. doi:[10.1109/GLOBECOM46510.2021.9685489](https://doi.org/10.1109/GLOBECOM46510.2021.9685489).

- 1026 Wu, Z., Yang, X., Wei, X., Yuan, P., Zhang, Y., Bai, J., 2024. A self-
1027 supervised anomaly detection algorithm with interpretability. Expert Sys-
1028 tems with Applications 237, 121539. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423020419>, doi:<https://doi.org/10.1016/j.eswa.2023.121539>.
- 1031 Yamanishi, K., Takeuchi, J.i., 2002. A unifying framework for detecting out-
1032 liers and change points from non-stationary time series data, in: Proceed-
1033 ings of the Eighth ACM SIGKDD International Conference on Knowledge
1034 Discovery and Data Mining, Association for Computing Machinery, New
1035 York, NY, USA. pp. 676–681. URL: <https://doi.org/10.1145/775047.775148>,
1036 doi:[10.1145/775047.775148](https://doi.org/10.1145/775047.775148).
- 1037 Yamanishi, K., Takeuchi, J.i., Williams, G., Milne, P., 2004. On-line
1038 unsupervised outlier detection using finite mixtures with discounting
1039 learning algorithms. Data Mining and Knowledge Discovery 8, 275–
1040 300. URL: <https://doi.org/10.1023/B:DAMI.0000023676.72185.7c>,
1041 doi:[10.1023/B:DAMI.0000023676.72185.7c](https://doi.org/10.1023/B:DAMI.0000023676.72185.7c).
- 1042 Yang, W.T., Reis, M.S., Borodin, V., Juge, M., Roussy, A., 2022. An
1043 interpretable unsupervised bayesian network model for fault detection
1044 and diagnosis. Control Engineering Practice 127, 105304. URL: <https://www.sciencedirect.com/science/article/pii/S0967066122001502>,
1045 doi:<https://doi.org/10.1016/j.conengprac.2022.105304>.
- 1047 Zhang, K., Chen, J., Lee, C.G., He, S., 2024. An unsupervised spa-
1048 tiotemporal fusion network augmented with random mask and time-
1049 relative information modulation for anomaly detection of machines with
1050 multiple measuring points. Expert Systems with Applications 237,
1051 121506. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423020080>, doi:<https://doi.org/10.1016/j.eswa.2023.121506>.
- 1054 Zhang, X., Shi, J., Huang, X., Xiao, F., Yang, M., Huang, J., Yin,
1055 X., Sohail Usmani, A., Chen, G., 2023. Towards deep probabilistic
1056 graph neural network for natural gas leak detection and localization
1057 without labeled anomaly data. Expert Systems with Applications 231,
1058 120542. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423010448>, doi:<https://doi.org/10.1016/j.eswa.2023.120542>.