



Knowledge-Enhanced Spatiotemporal Analysis for Anomaly Detection in Process Manufacturing

Louis Allen^a, Haiping Lu^{b,c}, Joan Cordiner^{a,*}

^a Department of Chemical and Biological Engineering, The University of Sheffield, S10 2TN, Sheffield, United Kingdom

^b Department of Computer Science, The University of Sheffield, 211 Portobello Street, S1 4DP, Sheffield, United Kingdom

^c Centre for Machine Intelligence, The University of Sheffield, 211 Portobello Street, S1 4DP, Sheffield, United Kingdom

ARTICLE INFO

Keywords:

Anomaly detection
Explainable modeling
Knowledge-enhanced modeling
Predictive maintenance
Generative AI

ABSTRACT

Effective fault detection and diagnosis (FDD) is crucial for proactively identifying irregular states that could jeopardize operator well-being and process integrity. In the era of Industry 4.0, data-driven FDD techniques have received particular attention, driven by the proliferation of stored manufacturing sensor data. While these methods have proven adept at categorizing established process fault scenarios, there remains an imperative to identify and explain anomalies stemming from uncharted faults or the interplay of consecutive anomalies. To address this we present a knowledge-enhanced FDD approach that integrates well-defined chemical engineering knowledge with cutting-edge deep learning techniques. We apply our methodology, named Knowledge-Enhanced Spatiotemporal Analysis (KESA), to identify abnormal process conditions that may be a precursor to failure. Furthermore, we utilize the knowledge of the fundamental relationships governing the process to explain why this fault case has occurred. This type of in-depth fault analysis is only possible through leveraging domain expertise and marks a step forward in FDD technology in comparison to current literature. Using the benchmark Tennessee Eastman process dataset, we establish superiority in the accuracy and efficiency of our KESA model against state-of-the-art FDD algorithms. This work highlights the importance of a knowledge-enhanced approach to deep learning in complex environments, emphasizing the critical role of timely and interpretable fault detection. By providing explanations for model results, our KESA framework not only aids in effective decision-making but also has the potential to significantly reduce the time between fault detection and the implementation of proactive mitigation actions. This capability is paramount for improving overall safety, minimizing downtime, and ultimately contributing to substantial cost savings in industrial processes.

1. Introduction

Poorly planned maintenance is detrimental to process manufacturers, with poorly maintained equipment leading to increased safety risks, product quality defects, and significant impacts on overall equipment effectiveness (OEE). Process manufacturing industries typically embrace a preventative maintenance approach in which a regular maintenance period is set for each piece of equipment, based either on time elapsed or the number of cycles through the process. The equipment is run between these set intervals without further maintenance, except for corrective work that occurs following breakdowns. The condition of the machine is recorded during maintenance, and the maintenance time interval is adjusted according to the deterioration of the equipment when inspected. While these types of scheduled maintenance systems are a vast improvement on run-to-failure (RTF) maintenance, they operate under the assumption that equipment reliability varies linearly with the operating age, an assumption which is known to be false for many

complex systems (Nowlan and Heap, 1978). The result is misdirected maintenance efforts leading to equipment in a comparatively good state being maintained while equipment close to failure is left to run until complete breakdown.

Moubray (2001) highlighted this fundamental flaw in schedule-based maintenance in his seminal work and called for the move to a fourth evolution of maintenance methodologies. Moubray called for a greater focus on the causes of equipment failure with a view to failure elimination rather than prevention. Failure is defined as a permanent interruption of a system to perform a function under given conditions, i.e. the point at which the process no longer meets a specified performance standard. A process fault is defined as an unpermitted deviation of at least one property of the system from normal operating conditions (Isermann and Ballé, 1997). Nowlan and Heap extend the definition of a fault to mean a physical condition that indicates a functional failure is imminent. Finally, we define a process disturbance

* Corresponding author.

E-mail address: j.cordiner@sheffield.ac.uk (J. Cordiner).

<https://doi.org/10.1016/j.compind.2024.104111>

Received 8 December 2023; Received in revised form 22 April 2024; Accepted 21 May 2024

Available online 31 May 2024

0166-3615/© 2024 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC license (<http://creativecommons.org/licenses/by-nc/4.0/>).

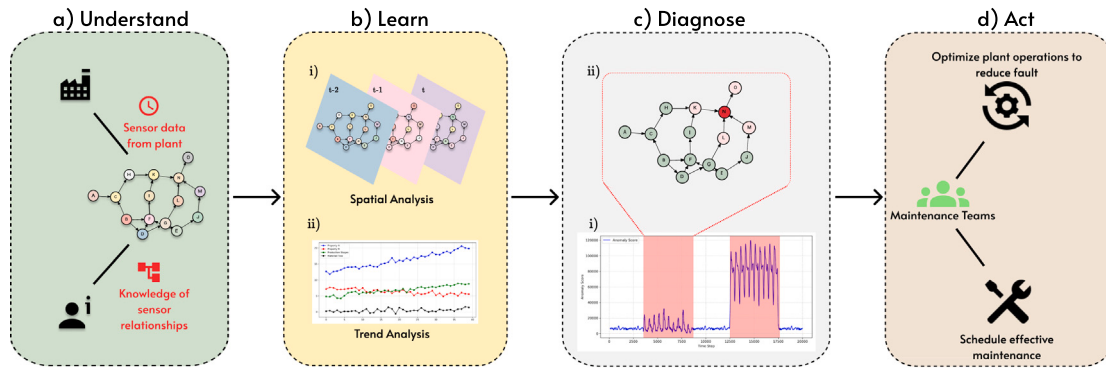


Fig. 1. Overview of the Knowledge-Enhanced Spatiotemporal Analysis for Anomaly Detection (KESA-AD) framework.

as an unknown input acting on the system, whereas a deviation is any departure from approved processes, procedures or accepted standards (Munro, 2017). It is vital for the process industries that we consider the deviation of product quality and process conditions as fault cases to be identified as well as the typically examined mechanical fault of equipment.

The process of eliminating failure starts with understanding the root cause of the problems observed. This root cause analysis (RCA) is a process that requires the input of experienced plant personnel, including maintenance professionals, plant engineers, and operators (Chemweno et al., 2016). The reliance on experienced staff to perform RCA is a challenge since this knowledge cannot easily be passed between people or across to different sites (Lokrantz et al., 2018). Furthermore, with an aging workforce nearing retirement this vital knowledge and experience are soon to leave the industry (Schramm and Wessels, 2015). Methods to capture and preserve this wealth of knowledge should be considered of vital importance over the coming years, especially as the industry undergoes its fourth industrial revolution (Vaidya et al., 2018).

Fig. 1 shows the scope of the work presented in this paper — Knowledge-Enhanced Spatiotemporal Analysis for Anomaly Detection (KESA-AD). This method utilizes chemical engineering knowledge as a foundation for applying machine learning, a computational approach where algorithms improve their performance at tasks by analyzing and learning from data. (a) The first step involves the combination of this knowledge with sensor data from the manufacturing process, giving a *spatio-temporal* representation of the process that can be read by the graph ML algorithms. (b (i)) From this, changes in the relationships between sensors are tracked over time to identify contextual faults (b (ii)) while long-term trends are analyzed to detect degradation. (c (i)) The knowledge graph representation of the process allows detection of the faulty timesteps (c (ii)) as well as diagnosis of the origin of the fault. (d) This information can be used by operational and maintenance teams to proactively intervene in fault cases to reduce the likelihood of unplanned downtime events.

2. Background and related work

Fault detection and diagnosis (FDD) is an important field of research for industrial applications, enabling early intervention in abnormal process events to ensure safe and efficient operations (Isermann, 1984). Consequently, FDD is a well-researched topic with methods developed for a range of industrial applications ranging from semiconductors, automobiles, and aerospace to name a few. The literature for FDD can be broadly divided into model-based, data-driven, and knowledge-based modeling techniques (Park et al., 2020). Model-based techniques do not generalize well to the process industries due to the often high levels of noise, auto-correlation, and non-stationarity which lead to imprecise models (Botre et al., 2017). Therefore, this paper considers only data-driven and knowledge-based modeling approaches.

2.1. Data-driven FDD methods

Data-driven methods require the collection of historical data depicting both normal and abnormal process conditions to build correlation-based models to detect faults. The most well-studied of these methods in FDD literature concern the use of principal component analysis (PCA), partial least squares (PLS), and independent component analysis (ICA) along with derivatives of these methods. These multivariate statistical process monitoring (MSPM) techniques allow the extraction of key information from highly voluminous manufacturing data, reducing the dimensions to be analyzed and thus simplifying the problem. The resulting lower dimensional representation can be used to identify abnormal process operations. Ding et al. explore the use of PCA techniques on industrial fault detection, concluding that the standard application of PCA is insufficient for accurately detecting all fault scenarios and all conditions, requiring an amended test statistic to be able to detect off-set faults (Ding et al., 2010). Russell et al. compared the use of PCA against a variant called dynamic PCA (DPCA) and canonical variate analysis (CVA) for fault detection in an industrial process and concluded that CVA lacked robustness while DPCA performed similarly to PCA in detecting most faults. Lee et al. used PCA to approximate the variance of independent components (ICs), where ICs the variance of the IC selected is the same as the PC. Dominant ICs were calculated using the FastICA algorithm and used to detect faults in a range of industrial processes from wastewater treatment to the Tennessee Eastman Process (TEP) (Lee et al., 2006). This approach outperformed both PCA and traditional ICA for fault detection. While PCA-based algorithms have been shown successful in detecting faults, they are limited in their application to process manufacturing systems since they do not take into account information between classes when creating the lower-dimensional space representation (Chiang et al., 2000). Therefore, when it comes to discriminating between different faults they often cannot correctly identify the root of the disturbance, instead pointing to response variables (Zhang et al., 1995; Gharahbagheri et al., 2017b). This is a vital flaw given Venkatasubramanian et al. highlight ‘novelty identifiability’ and ‘multiple fault identifiability’ as two of ten crucial features for effective process fault detection (Venkatasubramanian et al., 2003).

To overcome these limitations, authors have explored combining statistical FDD methods. One noteworthy method explores the combination of Kernel PCA (KPCA) with causal discovery algorithms (Gharahbagheri et al., 2017a). Here, Gharahbagheri et al. use KPCA to detect and identify faults, overcoming the limitations of PCA for non-linear processes. The authors used both Granger Causality and Transfer Entropy to identify fault propagation pathways amongst affected variables. While this method proved effective for detecting random disturbance faults, the stationarity assumption of the causal discovery methods limits the application to slow drift and step disturbances. Nonetheless, combination methods such as this show promise for accurate FDD in the process industries.

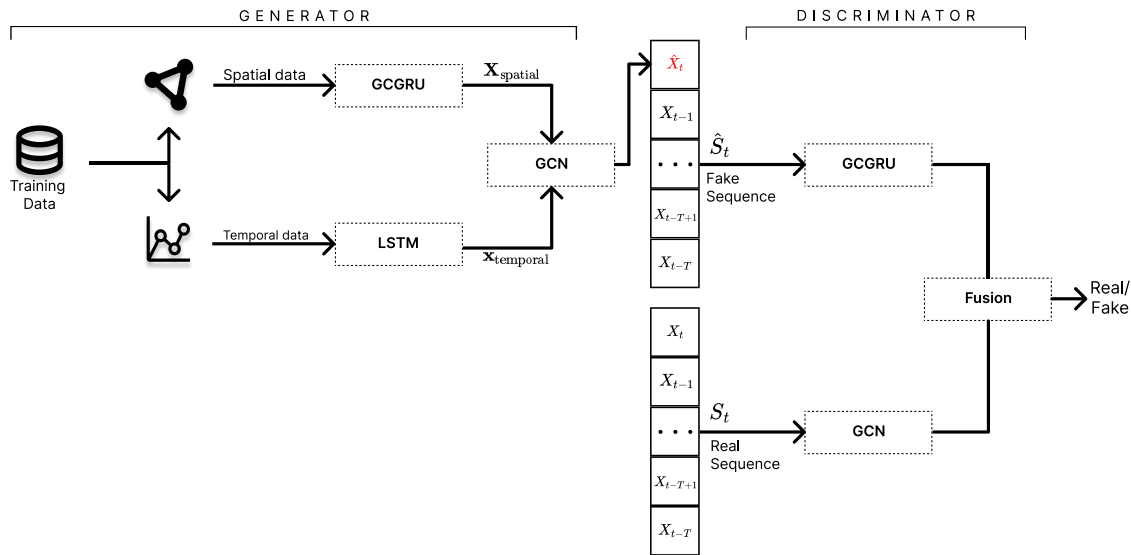


Fig. 2. Training KESA-AD framework.

There have been multiple attempts to detect and diagnose faults in manufacturing processes using deep learning methods based on artificial neural networks (ANNs) (Shin et al., 2005; Onel et al., 2019). These methods have been shown to accurately handle complex nonlinearities and highly variable data characteristic of manufacturing environments, however, they are classified as black-box methods and cannot accurately diagnose fault origins (Xie and Bai, 2016). Identifying this, Becraft et al. integrated neural networks with expert systems to aid in fault diagnosis (Becraft et al., 1991). The integration of domain knowledge with analytical tools for FDD is a promising area of research, yielding high detection rates with enhanced diagnostic ability (Park et al., 2020). There are, however, problems when combining domain knowledge with FDD methodologies, specifically with what constitutes the best domain knowledge to use. One could argue that using expert systems is a safe approach, however, most expert systems are rule-based and so struggle to diagnose new fault conditions for which there are no rules (Becraft et al., 1991). Integrating knowledge directly from plant engineers or operators introduces bias based on the experience of the personnel (He et al., 2014). Some thought must be given to how knowledge can be integrated to best serve the purpose of detecting and diagnosing faults in process manufacturing.

2.2. Knowledge-based FDD

Knowledge-based FDD can be thought of as an aspect of data-driven FDD in which ‘knowledge’ can be defined from historical data, or as qualitative input from process experts. Authors such as Hong et al. who introduce knowledge from historical plant data only overlook a wealth of operational knowledge to the potential detriment of fault diagnosis since only historical faults can be diagnosed (Hong et al., 2009). Don and Khan (2019) extend this approach by using a hidden Markov model (HMM) to extract process information from the historical data taken at normal operating conditions (NOC). This is augmented with process knowledge. A log-likelihood value is monitored and used to detect fault which is diagnosed using a BN. While this method can learn information from the process to detect a fault, it requires historical data for the occurrence of each of the fault cases, which is difficult to achieve for a real-world manufacturing facility. Other authors have had success in combining MSPM techniques with BNs or fault detection, where the network is built from some version of domain knowledge outside of historical data. Yu et al. propose a two-step approach where modified ICA is integrated with BNs for FDD. This has an advantage in large-scale processes where it is not economical to monitor and analyze

all variables. Gharahbagheri et al. propose a combination of KPCA with BNs, leveraging the fault detection of KPCA with diagnosis ability of BN to provide information on root causes. The authors discovered a limitation in modeling cyclic processes using BNs since they are inherently acyclic graphs, but were able to overcome this using pseudo-nodes (Gharahbagheri et al., 2017b). This cyclic constraint is a problem in using BNs to represent manufacturing systems since processes containing recycle streams will be cyclic by nature. Furthermore, with this type of method, it is not clear how these methods would deal with unobserved faults.

Some authors have addressed this issue by using graphical deep learning in place of BNs. Wu and Zhao define a process topology convolutional network (PTCN) to detect fault using graph convolutional networks built on a process topology of the TEP. Using deep learning in place of BN removes the acyclicity constraint since the PTCN is not a probabilistic graphical model, thus enabling a more general process topology to be adopted without use of pseudo-nodes (Wu and Zhao, 2021). However, while the authors showed this method was adept at detecting and classifying fault, they did not consider the potential of topology to *explain* a detected fault. This is crucial since process manufacturing faults occur along a delineated pathway where a failure may manifest in a particular unit despite originating from another separate downstream unit. The ability to *not only detect a fault but also explain its origin* is crucial to prevent repeated downtime occurrences by treating the root of the fault, not the symptom.

The objective of this research is to bridge the gap between the flexibility and fault detection ability of graphical deep learning-based methods, and the diagnostic ability of probabilistic graph methods such as BNs. Moreover, we aim to explore the ability of these models to detect and diagnose *unobserved* fault i.e. faults which the model has not been trained to detect. This has been highlighted by multiple authors as essential future learning for the research area (Gharahbagheri et al., 2017b; Venkatasubramanian et al., 2003). The novel contributions of this work are threefold:

- Development of a knowledge-enhanced spatiotemporal analysis (KESA) framework to perform accurate fault detection in a multi-component process manufacturing system even without prior familiarity with existing fault conditions.
- Harnessing domain expertise in chemical engineering to explain identified faults and suggest underlying root causes.
- Evaluating the performance of the KESA framework model, and benchmarking its effectiveness against relevant literature using the Tennessee Eastman Process dataset.

The paper is structured as follows. Section 3 introduces the KESA framework and describes how it can be used for anomaly detection. Section 3.4 proposes how this method can be used for fault diagnosis in the process industries. Section 4 introduces the TEP and applies the KESA framework to the benchmark dataset. Section 5 presents results from the case study and discusses the effectiveness of the method. Section 6 provides a conclusion to this paper and comments on future work.

3. Methods

3.1. Problem definition

The problem of fault detection and diagnosis in complex multi-component processes is difficult owing to the complex and volatile relationships that exist between variables. The state of a system variable at a given time will be correlated with other elements of the same system in the spatial and temporal dimensions (Zhang et al., 2022). Since manufacturing equipment is interconnected, a fault in one sensor or piece of equipment will change the conditions of the neighboring equipment. It is therefore important to consider the proximity of connected equipment when detecting faults. Knowledge of how faulting equipment impacts its neighborhood is also important for explaining fault propagation pathways. Moreover, a fault exhibited at a given timestep will be reflected in the adjacent timesteps. This is especially the case in manufacturing processes which have a residence time for material processing since disturbances may be carried forward and impact later stages of the process. Assessment across short and long-term windows in the temporal domain is important to capture multiple fault dynamics, including point faults, cyclical faults, and long-term degradation (Paolanti et al., 2018). The inclusion of the spatial domain will allow for the identification of contextual faults (Yang et al., 2023).

In this section, we first introduce the mathematical background for each of the elements of the framework depicted in Fig. 2. Then, we describe the training and evaluation of the fault detection model and how this can be used for diagnosing the root cause. Unless stated otherwise, we use bold uppercase letters for matrices (e.g., $\mathbf{A}$), bold lower case letters for vectors (e.g., $\mathbf{x}$), upper case letters for sets (e.g., G), and lower case letters to represent scalar values (e.g., k). We use superscript $\top$ for matrix transpose and $^{-1}$ for matrix inversion. We use $\mathbf{A}_{i,j}$ to represent the value of matrix $\mathbf{A}$ at the i th row and the j th column.

3.2. Mathematical background

This work looks to diagnose faults in multi-component manufacturing systems by capturing the spatiotemporal relationships present in data collected from process sensors. We express a manufacturing process as a graph of plant sensors $G = (V, E, \mathbf{W})$ where V is a finite set of nodes where $n = |V|$ is the number of sensors in the available data. The focus here is on detecting process faults, so we use process sensors such as temperature, pressures, and flows of each stream in a manufacturing process. E is a set of edges such that $e_{i,j} \in E$ denotes an edge between nodes v_i and v_j . In this work, an edge between nodes exists where there are mechanistic relationships between the sensor variables (Chen et al., 2020). For example, a temperature and pressure sensor may have an edge connecting them since we know the two properties are linked by the ideal gas law. $\mathbf{W}$ represents the weighted adjacency matrix of graph G , showing the spatial correlations between the graph nodes. The weights assigned to each edge are calculated using a thresholded Gaussian kernel for the edge weight between nodes v_i and v_j (Shuman et al., 2013):

$$\mathbf{W}_{i,j} = \begin{cases} \exp\left(-\frac{\text{dist}(v_i, v_j)^2}{2\sigma^2}\right), & \text{if } i \neq j \text{ and } e_{i,j} = 1 \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

where $\mathbf{W}_{i,j}$ represents the weight of the edge between nodes v_i and v_j . The term $\text{dist}(v_i, v_j)$ represents the distance between nodes v_i and v_j , and σ is the standard deviation of the distances. Calculating the weighting via Eq. (1) assigns a larger weighting to edges where nodes have greater proximity to one another, since the closer they are the more information they are likely to have about one another.

The problem can therefore be phrased simply: given (a) a graph, G , of a process and (b) a sequence of sensor data $\mathbf{X} \in \mathbb{R}^{n \times m}$ where m represents the total number of timesteps the data is collected across, can the spatiotemporal relationships be learned such that we can predict the future state of the system and detect anomaly?

3.2.1. Graph convolutions

Graphs are important to this work since they allow us to capture the spatial layout of the process, and can help provide detail to the model about the underlying mechanics defining the system. However, they can be difficult to learn from directly. This is because graphs are a non-Euclidean data structure — i.e. they do not follow the rules of Euclidean geometry (Kipf and Welling, 2017). Therefore, we require an operation to be able to process graph data in the KESA framework called graph convolution. This differs from a regular convolution, which can be applied to images or regular grid structures since they are Euclidean structures (Yu et al., 2017). To perform a convolution on a graph structure, Kipf and Welling propose the spectral graph convolution, which generalizes the convolution operator for graph structures, leading to the definition of the graph convolutional operator, denoted by $*_G$ (Bruna et al., 2014). In this formulation, spectral convolution is considered as the multiplication of a graph signal with a filter $g_\theta = \text{diag}(\theta)$ parameterized by $\theta \in \mathbb{R}^n$ in the Fourier domain (Kipf and Welling, 2017):

$$g_\theta *_G \mathbf{X} = g_\theta(\mathbf{L})\mathbf{X} = g_\theta(\mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top)\mathbf{X}, \quad (2)$$

where $\mathbf{U} \in \mathbb{R}^{n \times n}$ is the matrix of eigenvectors and $\mathbf{\Lambda} \in \mathbb{R}^{n \times n}$ is the diagonal matrix of eigenvalues of the normalized graph Laplacian $\mathbf{L} = \mathbf{I}_n - \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top \in \mathbb{R}^{n \times n}$, $\mathbf{I}_n$ is the identity matrix and $\mathbf{D}$ is the diagonal degree matrix $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{i,j}$. Therefore, utilizing the graph convolutional approach enables us to leverage the structural information contained in the graph data structure when constructing machine learning models. One example of this type of machine learning model used in this work is a graph convolutional gated recurrent unit (GCGRU).

3.2.2. GCGRU

Gated recurrent units (GRUs) are recurrent neural network (RNN) architectures common in time series forecasting problems (Torres et al., 2021). GRUs work by using a gating mechanism to control the flow of information through the network (Chung et al., 2014). This gating mechanism allows GRUs to learn which information is important to retain and which information can be discarded. This is particularly important for time series forecasting problems, where the data can be noisy and irrelevant information can obscure the underlying patterns (Zheng and Chen, 2021). To make the GRU architecture applicable to graph data, we must employ the above graph convolutions to create a GCGRU. To do so the matrix multiplications in the reset and update gates of the GRU are replaced with the graph convolutional operator (Cho et al., 2014):

$$\mathbf{r}^{(t)} = \sigma_r(g_{\theta,r} *_G [\mathbf{X}^{(t)}, \mathbf{H}^{(t-1)}] + \mathbf{b}_r), \quad (3)$$

$$\mathbf{u}^{(t)} = \sigma_u(g_{\theta,u} *_G [\mathbf{X}^{(t)}, \mathbf{H}^{(t-1)}] + \mathbf{b}_u), \quad (4)$$

$$\mathbf{C}^{(t)} = \tanh(g_{\theta,C} *_G [\mathbf{X}^{(t)}, (\mathbf{r}^{(t)} \odot \mathbf{H}^{(t-1)})] + \mathbf{b}_C), \quad (5)$$

$$\mathbf{H}^{(t)} = \mathbf{u}^{(t)}\mathbf{H}^{(t-1)} + (1 - \mathbf{u}^{(t)}) \odot \mathbf{C}^{(t)}, \quad (6)$$

where $\mathbf{X}^{(t)}$ is the input and $\mathbf{H}^{(t)}$ is the output at a timestep t , $\mathbf{r}^{(t)}$, $\mathbf{u}^{(t)}$ are the reset and update gates at time t , σ_r , σ_u are sigmoid activation functions and $g_{\theta,r}$, $g_{\theta,u}$, $g_{\theta,C}$ are filters with the corresponding parameters (Li

et al., 2018). Throughout this work, GCGRUs have been employed to capture spatial dynamics of the manufacturing system by applying forecasts using the graph defined by the mechanistic equations of the system. This spatial analysis forms one element of the spatiotemporal fault detection defined in the wider framework.

3.3. KESA framework

The proposed method is based on the concept of a generative adversarial network with graph convolution (Jia et al., 2023). The KESA framework consists of two major components, a generator and a discriminator. The generator attempts to produce an accurate forecast of the sensor time series data based on relationships learned during the training phase such that the generated forecast is indistinguishable from the real data. The discriminator attempts to differentiate the real data from the generated forecasts. Over the training process, the generator learns to produce very accurate forecasts as it tries to fool the discriminator. As the discriminator becomes better at distinguishing real data from synthetic data, the generator is forced to become better and better, in the end generating very realistic time series based on learned relationships (Goodfellow et al., 2014). It is important for fault detection that we can accurately capture system dynamics to produce accurate forecasts, using a GAN helps us achieve this.

3.3.1. Graph subgraphs

The KESA-AD framework leverages the concept of subgraphs to capture the relationships between sensor nodes. A subgraph is defined below (Deng et al., 2022):

$$G_v = (V_v, E_v, W_v), \quad (7)$$

where V_v is a list of nodes comprising node v and the closest $k - 1$ nodes where k is a tunable hyperparameter representing the number of nodes in a subgraph. The parameter k controls the scope of the spatial influence considered for a particular node. Nodes within a subgraph are likely to be highly correlated due to their proximity within the process.

3.3.2. Generator

The generator section of the generative adversarial network (GAN) is responsible for generating forecasts of sensor time series data. It is divided into two distinct modules — a spatial module and a temporal module. The spatial module captures the spatial relationships between different sensors, while the temporal module captures the temporal relationships between different timesteps. The spatial module looks to capture the spatial arrangements of the system, by processing time series for the subgraph of each node. As noted, the state of a node is strongly correlated with nodes in its surrounding neighborhood. The spatial module aims at capturing the spatial correlations of the data across a shortened period and consists of a GCGRU. The graph convolution is based on the spectral approach discussed in Section 3.2.1. This spatial module captures a shortened period, s , from each subgraph:

$$\mathbf{X}_{\text{spatial},v} = (\mathbf{X}_{v,t-s}, \mathbf{X}_{v,t-s+1}, \dots, \mathbf{X}_{v,t-1}) \in \mathbb{R}^{s \times k}, \quad (8)$$

$\mathbf{X}_{v,t} \in \mathbb{R}^k$ denotes the dynamics of a subgraph of node v at time t .

A second module is defined within the generator to capture the temporal dynamics of each node in the system and forecast the long-term trend of each node. Here a long short-term memory (LSTM) architecture is used to help identify cyclical anomaly or degradation. LSTM architectures in particular have a strong ability to capture and predict accurately different degradation patterns linked to manufacturing equipment (Ahmed et al., 2022). Since the spatial correlations are captured by the spatial module we focus only on capturing temporal dynamics across a longer time:

$$\mathbf{x}_{\text{temporal},v} = (x_{v,t-l}, x_{v,t-l+1}, \dots, x_{v,t-1}) \in \mathbb{R}^l, \quad (9)$$

where l denotes the length of the period across which we assess the temporal dynamics of each node. The output from both the spatial and

temporal generators are concatenated in a graph convolutional layer to give the final prediction for the process dynamics of the subgraph of the nodes, $\hat{\mathbf{X}}_v$:

$$\hat{\mathbf{X}}_v = \tanh(g_\theta *_{\mathcal{G}} [\mathbf{X}_{\text{spatial},v}, \mathbf{x}_{\text{temporal},v}]), \quad (10)$$

where $*_{\mathcal{G}}$ is the graph convolutional operator. To enable the generator to create accurate forecasts, we must define a suitable adversary to train against. This is the discriminator.

3.3.3. Discriminator

The purpose of the discriminator is to determine whether a given signal is a real signal from the process or a fake signal from the generator. During training, this helps to improve the accuracy of the prediction from the generator. During detection, the discriminator can be used alongside the generator to create a combined abnormality score to improve the detection performance (Lee et al., 2018). This is described in the following section.

3.3.4. Training KESA-AD

To effectively train the KESA-AD framework, an appropriate loss function must be selected for both the generator and discriminator. A forecasting error and a realism loss are defined for the generator. Minimizing the forecasting error in the loss function ensures the generated series resembles the true state of the system. By minimizing the realism loss, the generator can encourage the discriminator to fail to classify a fake sequence as fake. The forecasting loss (ℓ_{forecast}) and realism loss (ℓ_{real}) are combined to give an overall generator loss L_G :

$$\ell_{\text{forecast}} = \|G_\theta(\mathbf{X}_{t-s}, \dots, \mathbf{X}_t) - \mathbf{X}_t\|_2, \quad (11)$$

$$\ell_{\text{real}} = -\log(D_\phi(\hat{\mathbf{S}}_t)), \quad (12)$$

$$L_G(\theta) = \sum_{t \in \text{batch}} \ell_{\text{real}} + \lambda_G \ell_{\text{forecast}}, \quad (13)$$

where G_θ is the generator function with parameter θ , D_ϕ is the discriminator with parameter ϕ , and λ_G is a hyperparameter to balance the forecasting loss with the realism loss. $\hat{\mathbf{S}}_t = \{\mathbf{X}_{t-s}, \mathbf{X}_{t-s+1}, \dots, \hat{\mathbf{X}}_t\}$ is the fake sequence including the generated value $\hat{\mathbf{X}}_t$.

As well as the generator loss, we also define an adversarial loss for the discriminator. Minimizing the adversarial loss ensures the discriminator can accurately differentiate the real sequence from the fake sequence from the generator:

$$L_D(\phi) = \sum_{t \in \text{batch}} \ell_{\text{real}} - \log(D_\phi(S_t)), \quad (14)$$

where $S_t = \{\mathbf{X}_{t-s}, \mathbf{X}_{t-s+1}, \dots, \mathbf{X}_t\}$ is the real sequence taken from the training data. Once trained, the framework can be used to detect anomalies based on incoming data.

3.4. Score-based fault detection and diagnosis

3.4.1. Anomaly detection score

Fault detection can, in theory, be achieved using either the forecasting error from the generator, s_G or by using the adversarial loss from the discriminator, s_D . Lee et al. (2018) showed that the robustness of the anomaly detection is increased by using both the generator and discriminator in a combined anomaly score:

$$s_G = \|G_\theta(\mathbf{X}_{t-s}, \dots, \mathbf{X}_t) - \mathbf{X}_t\|_2 \quad (15)$$

$$s_D = D_\phi(S_t) - D_\phi(\hat{\mathbf{S}}_t) \quad (16)$$

$$\text{anom}(v, t) = s_G - \frac{s_D}{\lambda_s} \quad (17)$$

where λ_s is a tunable parameter to balance the generator and discriminator loss.

An anomaly score is calculated for each node at each timestep based on Eq. (17). These are summed to give an overall anomaly score for each timestep. A threshold value, τ is defined based on the prediction

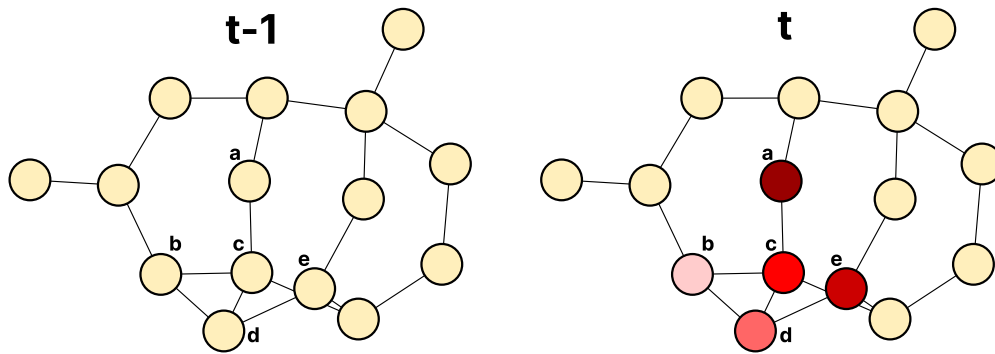


Fig. 3. A process graph undergoing a disturbance both before ($t-1$) and after (t). The deepness of the color represents the magnitude of the anomaly score quantified in Table 1. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

of anomaly scores for a standardized set of non-faulty data and is set three standard deviations above the mean anomaly score for the dataset. When testing, if the anomaly score per predicted timestep is greater than or equal to τ , that timestep is assigned a 1 to show fault is detected. If the anomaly score is less than τ value the timestep is assigned a 0 to show there is no fault:

$$\text{anom}(t) = \sum_{v=1}^n \text{anom}(v, t) \quad (18)$$

$$\text{Fault} = \begin{cases} 0, & \text{anom}(t) < \tau \\ 1, & \text{anom}(t) \geq \tau \end{cases} \quad (19)$$

Once a fault has been detected, it is important to identify the root cause of the fault. This is because the root cause analysis can help us to take corrective action to prevent the fault from happening again.

3.4.2. Root cause analysis

The KESA-AD framework can assist with fault diagnosis and explanation by identifying the nodes and timesteps most associated with the faults. To achieve this, we note that the interconnected dynamics of a manufacturing process create a network wherein faults in a specific node manifest not only in that node but also in its surrounding node subgraph. This complexity can make explaining a fault's root cause challenging, as the anomaly scores of adjacent nodes are also likely to rise. To address this, we analyze anomaly scores for entire subgraphs of each node before and after a fault. By summing anomaly scores across subgraphs, we identify the most significant changes in system relationships. This approach aids in process fault investigation, helping to explain the fault by narrowing down potential root causes.

Consider a simple anonymous process shown in Fig. 3 that undergoes a disturbance. The figure depicts the process as a knowledge graph at two points, both before the disturbance at time $t-1$ and after at time t . The color of the node represents the anomaly score calculated via Eq. (17), between the actual value and the KESA-AD model prediction which is shown numerically in Table 1. Five nodes (a–e) exhibit anomalies due to the disturbance.

At first glance, node a seems the most problematic with an anomaly score of 0.8. However, a more nuanced picture emerges when we examine the *subgraph anomaly impact* of each of the affected nodes, defined as the sum of the anomaly scores of each of the nodes subgraphs, shown in Table 1. Here, node c's subgraph experiences a larger anomaly (1.94) compared to node a's subgraph (1.49). This suggests node c is more likely to be the root cause of the fault. Therefore, focusing investigation starting with node c is more likely to lead to fast and effective intervention.

4. Case study — Tennessee Eastman process

4.1. Overview

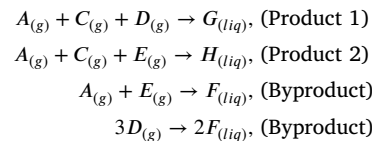
To demonstrate the KESA-AD framework, we utilize the benchmark Tennessee Eastman Process (TEP) simulation (Downs and Vogel, 1993).

Table 1

Anomaly score and subgraph anomaly score of affected nodes in Fig. 3.

Node	Anomaly score	Node subgraph	$\sum$ Anomaly Score
a	0.80	{a, c, b}	1.49
b	0.15	{b, c, d}	0.92
c	0.54	{c, e, a}	1.94
d	0.23	{d, c, e}	1.37
e	0.60	{e, c, d}	1.37

This simulation was first introduced for testing process control technologies and is based on the actual process of the Eastman Chemical Company in Tennessee, USA. It has since been widely to compare the efficacy of fault detection and diagnosis algorithms. Using simulation data for model testing is important since it allows us to know exactly which faults are propagating through the system at a given time, when they were induced, and what the original cause of the fault is. This would not be possible to the same extent using data from a real manufacturing facility since this information is not as readily available. Furthermore, simulations do not suffer from the influence of external factors (e.g., weather conditions, operator changes) as a real process would. Therefore, by using simulation data we gain visibility of all factors influencing potential faults, which allows us to test for the true root cause with the knowledge that there are no external factors contributing. The process produces two products from four reactants via the reactions shown below:



where all the reactions are both irreversible and exothermic. The process briefly comprises five major unit operations: a reactor, a product stripper, a vapor–liquid separator, and a recycle condenser. The process and instrumentation diagram (P&ID) for the revised TEP is shown in Fig. 4 (Bathelt et al., 2015).

4.2. Data description

The process shown above concerning five major units is monitored through 41 measurement variables, shown in Table 2 where XMEAS (1)–(22) are continuous process variables and XMEAS (23)–(41) relate to composition measurements. There are 12 additional manipulated variables, shown in Table 3. Each measured and manipulated variable is monitored throughout the simulation, with samples taken every 3 min. The modern TEP simulation allows the introduction of 20 process disturbances, where faults 1–15 are known faults and faults 15–20 are unknown (Downs and Vogel, 1993).

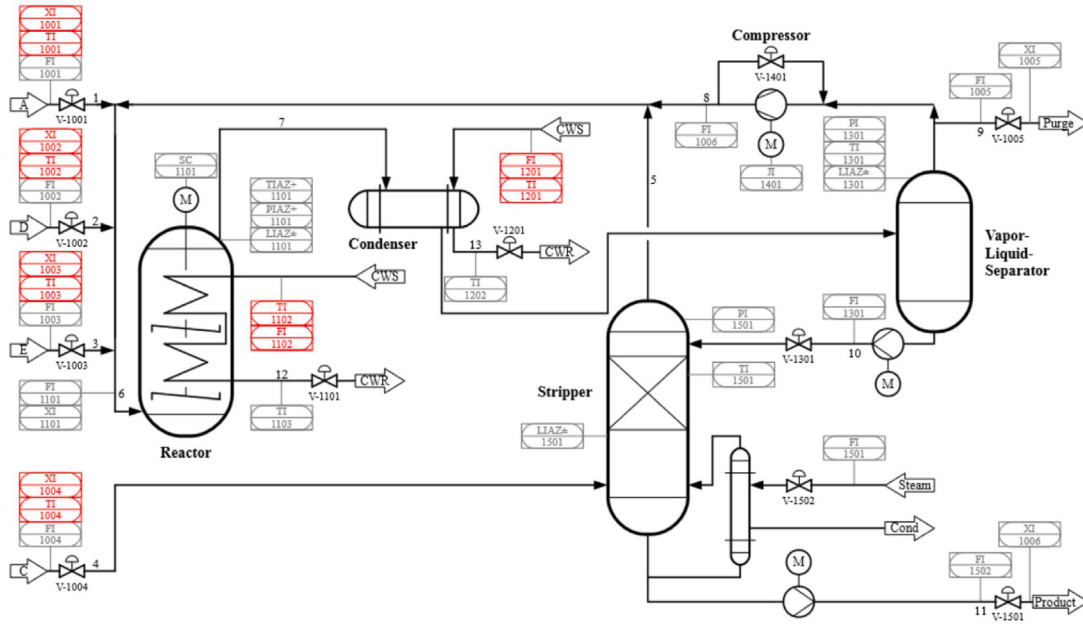


Fig. 4. A process and instrumentation diagram (P&ID) of the revised TEP model (Bathelt et al., 2015).

Table 2

Measured variables in the TEP dataset including process variables XMEAS (1)–(22) and composition variables XMEAS (23)–(41).

Measured variable	Description	Units
XMEAS (1)	A feed (stream 1)	kscmh
XMEAS (2)	D feed (stream 2)	kg h ⁻¹
XMEAS (3)	E feed (stream 3)	kg h ⁻¹
XMEAS (4)	A & C feed (stream 4)	kscmh
XMEAS (5)	Recycle flow (stream 5)	kscmh
XMEAS (6)	Reactor feed rate (stream 6)	kscmh
XMEAS (7)	Reactor pressure	kPa gauge
XMEAS (8)	Reactor level	%
XMEAS (9)	Reactor temperature	°C
XMEAS (10)	Purge rate (stream 9)	kscmh
XMEAS (11)	Product separator temperature	°C
XMEAS (12)	Product separator level	%
XMEAS (13)	Product separator pressure	kPa gauge
XMEAS (14)	Product separator underflow (stream 10)	m ³ h ⁻¹
XMEAS (15)	Stripper level	%
XMEAS (16)	Stripper pressure	kPa gauge
XMEAS (17)	Stripper underflow (stream 11)	m ³ h ⁻¹
XMEAS (18)	Stripper temperature	°C
XMEAS (19)	Stripper steam flow	kg h ⁻¹
XMEAS (20)	Compressor work	kW
XMEAS (21)	Reactor cooling water outlet temperature	°C
XMEAS (22)	Separator cooling water outlet temperature	°C
Reactor feed analysis (stream 6)		
XMEAS (23–28)	Component A–F conc	mol %
Pure gas analysis (stream 9)		
XMEAS (29–36)	Component A–H conc	mol %
Product analysis (stream 11)		
XMEAS (37–41)	Component D–H conc	mol %

4.3. Process graph from TEP dataset

The framework proposed in Fig. 2 requires a process graph to learn the spatial relationships. Since in this case study we are interested in detecting fault from the sensor data, we can construct a graph $G_{TEP} = (V, E, W)$ where $|V| = n$ is the number of sensors available. 52 sensors from the TEP dataset are used; 41 measured variables and 11 manipulated variables. XMV12 is not considered since the agitator speed

Table 3

Manipulated variables in the TEP dataset.

Manipulated Variable	Description	Units
XMV (1)	D feed flow (stream 2)	kg h ⁻¹
XMV (2)	E feed flow (stream 3)	kg h ⁻¹
XMV (3)	A feed flow (stream 1)	kscmh
XMV (4)	A & C feed flow (stream 4)	kscmh
XMV (5)	Compressor recycle valve	%
XMV (6)	Purge valve (stream 9)	%
XMV (7)	Separator pot liquid flow (stream 10)	m ³ h ⁻¹
XMV (8)	Stripper liquid product flow (stream 11)	m ³ h ⁻¹
XMV (9)	Stripper steam valve	%
XMV (10)	Reactor cooling water flow	m ³ h ⁻¹
XMV (11)	Condenser cooling water flow	m ³ h ⁻¹
XMV (12)	Agitator speed	rpm

remains constant throughout all simulations and therefore provides no information for fault diagnosis.

Edges are drawn between nodes according to the causal structure of the system under the constraints of mechanical knowledge as determined by Chen et al. (2021). The original structure proposed contains only process variables XMEAS (1)–(22) and the 11 manipulated variables XMV (1)–(11), excluding agitator speed. For this paper, the causal structure is revised to include composition variables XMEAS (23)–(41). The final process representation is shown in Fig. 5.

4.3.1. Training data

The purpose of the KESA framework is to detect and diagnose faults for which there is insufficient labeled data to use supervised learning approaches. The training data for the model, therefore, does not include any fault data. Instead, the training data is taken from the process in normal operating conditions (NOC) so that the model can learn the relationships between sensor variables and what they should look like when no fault is present. The training dataset comprises 1000 h of sensor data taken from the TEP simulation. This equates to 20 000 sampled timesteps taken every 3 min across approximately 41 days.

4.3.2. Testing data

Testing data is created for each fault from IDV (1)–(15). IDV (15)–(20) are not assessed since there is no information about the fault type or root cause. For each assessed fault, the process was simulated 10

Table 4

Process faults for the TEP dataset including the fault number, a description of the fault, the nature of the fault, and the original root cause of the fault as identified by [Gharahbagheri et al. \(2017b\)](#).

Fault	Description	Type	Root cause
IDV (1)	A/C feed ratio, B composition constant (stream 4)	Step	XMEAS (4)
IDV (2)	B composition, A/C ratio constant (stream 4)	Step	
IDV (3)	D feed temperature (stream 2)	Step	
IDV (4)	Reactor cooling water inlet temperature	Step	XMEAS (9)
IDV (5)	Condenser cooling water inlet temperature	Step	XMEAS (11)
IDV (6)	A feed loss (stream 1)	Step	XMEAS (1)
IDV (7)	C header pressure loss - reduced availability (stream 4)	Step	
IDV (8)	A, B, C feed composition	Random variation	
IDV (9)	D feed temperature (stream 2)	Random variation	
IDV (10)	C feed temperature (stream 4)	Random variation	XMEAS (18)
IDV (11)	Reactor cooling water inlet temperature	Random variation	XMEAS (9)
IDV (12)	Condenser cooling water inlet temperature	Random variation	XMEAS (11)
IDV (13)	Reaction kinetics	Slow drift	
IDV (14)	Reactor cooling water valve	Sticking	XMEAS (9)
IDV (15)	Condenser cooling water valve	Sticking	
IDV (16)	Unknown	—	—
IDV (17)	Unknown	—	—
IDV (18)	Unknown	—	—
IDV (19)	Unknown	—	—
IDV (20)	Unknown	—	—

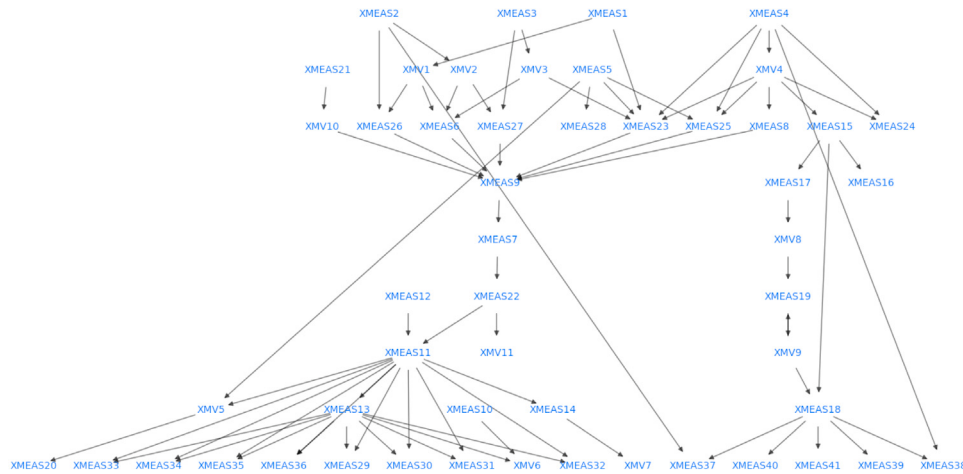


Fig. 5. Knowledge graph of the TEP based on causal structure proposed by [Chen et al. \(2021\)](#) and updated to include composition variables.

times. Each of the 10 runs generated approximately 25 h of data, totaling around 250 h of fault data generated for each fault case sampled every 3 min, equating to 5000 samples per fault. A test containing data under normal operating conditions was generated to validate the model's performance against a simulation with no fault. This test case comprised around 200 h of normal operating data.

To assess the fault detection delay, an additional testing data set is generated. This test contains a sequence of randomly selected fault cases in a single data set separated by normal operating conditions (NOC) data. The goal is to see how quickly the model can go from identifying NOC to detecting fault, and back again since latency is an important factor. This data set contains 178 840 samples (approx 372 days). Each fault occurs 3 times in a random series and is separated by NOC data in each case.

4.4. Preprocessing

Since data is produced using a simulation there is no missing data to deal with in the TEP data. The different sensor readings differ in scale, so the data is normalized before processing as:

$$X_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}, \quad (20)$$

4.5. Evaluation criteria

It is essential to choose evaluation metrics that reflect the application of the model. Precision is considered an essential metric for fault detection in an industrial case since the penalty for misdiagnosing NOC as a fault is severe — unnecessary maintenance that costs production and puts personnel at risk. Therefore precision is considered in the evaluation of each model. Fault detection rate (FDR) is an important metric also defined and used to assess the model ([Wu and Zhao, 2021](#)). The FDR concerns the ability of the model to classify each faulting sample as a fault and is sometimes referred to as the true positive rate (TPR) in literature:

$$\text{FDR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (21)$$

where TP and FN refer to the count of true positives, and false negatives respectively. Fault detection delay will also be measured. This is a vital metric representing how quickly the model can detect an induced fault. The detection delay is of vital importance for the real-world adoption of FDD algorithms.

Finally, we define an F1 score as a standard metric in the field of anomaly detection to allow easy comparison to other work in this field:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (22)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (23)$$

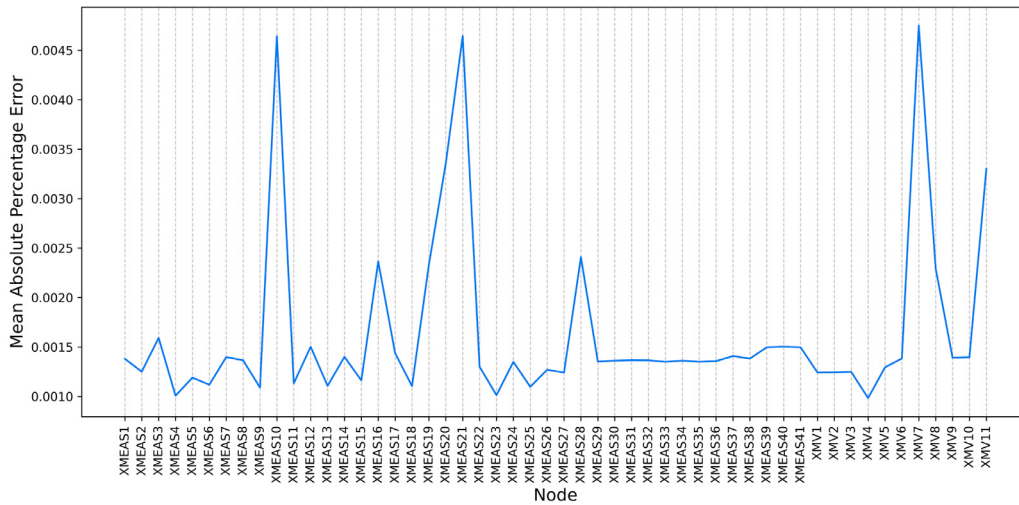


Fig. 6. MAPE per node for each predicted sensor (from 1–53 corresponding to XMEAS (1)–(41) and XMV (1)–(11)) under normal operating conditions.

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (24)$$

where FP refers to the count of false positive predictions of anomalous data.

5. Results and discussion

The KESA model was trained over 5 epochs, in which the data was divided into mini-batches of 128 samples. The Adam optimizer was used to tune the hyperparameters with a learning rate of 0.001 (Kingma and Ba, 2014). The value of k is set to 5 for this work.

5.1. Model prediction accuracy

We first look at the model's ability to reproduce data at NOC since this ability forms the baseline for fault detection. Since there are no faults to detect here, mean absolute percentage error (MAPE) is used to assess the ability of the model to accurately reproduce the dynamics of the system. Fig. 6 shows the MAPE per node for the prediction under normal conditions. The first thing to note is the low error for each node, with an average MAPE of 0.17% across all nodes. The accuracy of the generator in forecasting the system dynamics shows the ability of the framework to capture and replicate the complex nonlinear relationships that link the process sensors in the system under normal conditions. This is important for fault detection as it will allow identification by comparison of times when these relationships are disturbed, which can be classed as faults.

Four particular nodes relating to XMEAS (10), XMEAS (20), XMEAS (21), and XMV (7) exhibit a high MAPE compared to other nodes. From Fig. 5 it can be seen that these poor-performing nodes all exhibit either an in-degree of 1 and an out-degree of 0, or conversely, an in-degree of 0 and an out-degree of 1. Therefore, the higher error on these nodes in particular could point towards a case for a *more connected graph being used to increase the accuracy of performance*. When applying these methodologies to other processes, consideration should be given to how the graph is constructed such that less connected nodes are avoided since these may result in reduced model performance. In particular, future research could look at supplementing the mechanistic relationships used to build the graph with physical layout or topology information. An alternative approach could be found through the use of pseudo-nodes to increase the connectivity of all nodes. Conversely, this method could be used in the future to identify areas of a process that require greater visibility for effective FDD, highlighting where sensor installation might be most effective to allow for influential conclusions from process analysis.

5.2. Fault detection ability

To show the efficacy of the framework proposed, we compared our model against top-performing unsupervised anomaly detection models (Hartung et al., 2023). In each case, the models were trained using the same training data with no existing faults, and given the same 10 simulations for each of the 15 faults tested. The models were all trained on the same system equipped with an AMD Ryzen 7 5800H with Radeon Graphics, 3201 MHz, 8 Cores, 16 Logical Processors, 32 GB RAM, with access to an NVIDIA GeForce RTX 3080 16GBGDDR6 GPU. Models were evaluated using the code provided in the TimeSeAD library¹ (Wagner et al., 2023). A threshold value for determining anomaly was calculated by taking three standard deviations above the mean anomaly score from the NOC data. This was set at $\tau = 0.043$. Fig. 7 shows a comparison of the performance of each tested algorithm against each test case IDV (1)–(15) with Fig. 7(a) showing the fault detection rate scores and Fig. 7(b) showing the F1 scores. These results have been averaged and summarized in Table 5 for easy comparison.

KESA-AD performs well, outperforming the state-of-the-art unsupervised detection algorithms with a high average FDR of $0.962 (\pm 0.063)$ across all 15 fault cases. The lowest FDR shown is by IDV (3) with the lowest average and highest deviation ($FDR = 0.921 \pm 0.086$). This fault relates to the temperature of the D feed in stream 2 from Fig. 4. This is because there is no explicit measurement of temperature for this stream, with the closest temperature data being from the reactor stage (XMEAS (9)). Therefore, there is a difference between the induction of the fault in stream 2 and the position where that fault is first recorded in the reactor. This lag results in multiple timesteps where the fault has occurred undetected. In a real-world scenario, this would suggest that greater visibility of the inlet streams to the reactor is needed to be able to detect such a fault.

Our model outperforms other FDD models with a high F1 score of 0.979 ± 0.035 . Importantly, KESA-AD is the only one to maintain high performance over 'challenging' fault cases IDV (3), (9) and (15). The performance of all non-knowledge-enhanced models drops significantly when presented with hard-to-detect faults, shown by a drop in both F1 score and FDR. For instance, the BeatGAN model manages to detect IDV (3) well but struggles with IDV (9) and (15). This distinction highlights the impact of integrating domain knowledge into fault detection, fostering robustness across a broad spectrum of fault cases. In a manufacturing context, characterized by the complexity and multifaceted nature of process failures, this demonstrated enhancement

¹ <https://github.com/wagner-d/TimeSeAD>.

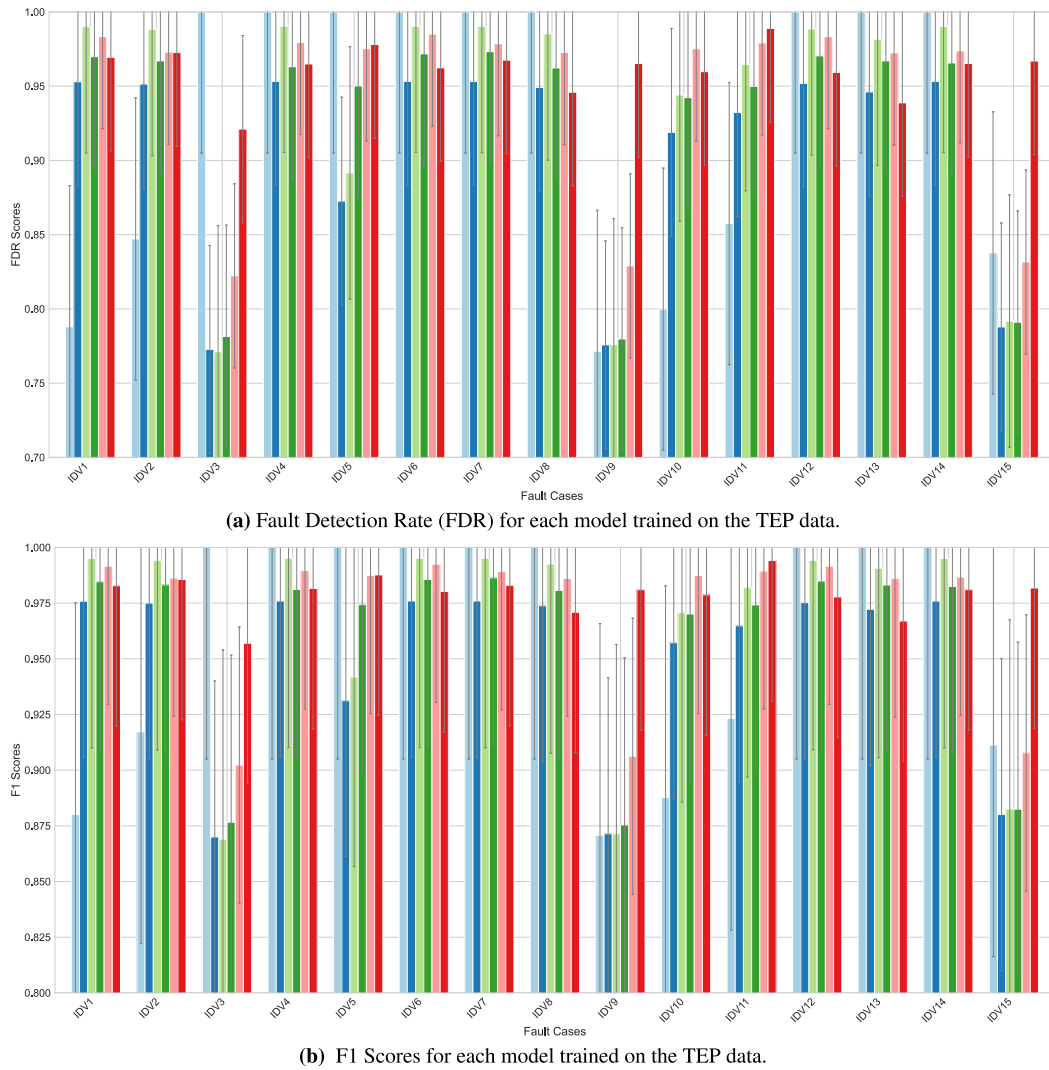


Fig. 7. Comparison of FDR and F1 scores of different models tested on the 15 fault cases and averaged across 10 different simulation runs. Error bars show the ± 1 standard deviation across the 10 runs.

is of paramount significance. It empowers manufacturers with a tool capable of *effectively identifying and addressing faults that would otherwise remain elusive and challenging to pinpoint*. It is important to note that, when applying this methodology to other FDD problems, given the model has no constraints on stationarity or linearity, KESA-AD can detect a wide range of fault types from random disturbances to step changes and slow drift faults without visible effect on performance. This highlights the importance of generative AI methods in this space to alleviate the research problems faced by authors using MSPM or probabilistic methods in this space.

The F1 scores and FDR results have been averaged across each fault and tabulated in Table 5 for ease of comparison. We see that our model outperforms the state-of-the-art unsupervised fault detection algorithms when compared using the benchmark TEP dataset. This performance increase shows the potential of the model to tackle more complex and dynamic datasets. Not only did our model outperform others in the average value of both the F1 score and FDR, but it did so while maintaining a lower variability in the prediction than other models shown by the lower standard deviation except STORN. This is important since a model with lower variability in the prediction is less likely to trigger false alarms, so when the model detects a fault it is more likely to be real.

As all tested models consistently achieved a precision score of 1.00 for simulated faults, this metric has been excluded from the

comparative analysis. Nevertheless, it is important to acknowledge the operational significance of maintaining such high precision in the model. This is vital because erroneously identifying a non-existent fault can incur unnecessary production costs and elevate the risk to personnel safety.

5.3. Root cause analysis

The most important advancement shown by our work compared to the other anomaly detection algorithms is the ability to diagnose the root cause. The graph-based approach allows the identification of where the fault exists in the process by analyzing the anomaly scores of each node. Other models that do not leverage the graph structure lack this capability. Moreover, because we construct a graph using chemical engineering knowledge and mechanistic relationships, we can utilize the underlying structure of the node sub-graphs to begin to identify the reason why the faulting nodes might be exhibiting such behavior. This is a clear distinction between KESA-AD and the other unsupervised fault detection algorithms that offer a wealth of operational benefits. It allows maintenance teams to promptly understand the underlying cause of the issue, facilitating improved maintenance planning and resource management. The above demonstrates the ability of the KESA-AD framework to identify when a fault condition occurs in a process

Table 5

Comparison of F1 scores and FDR for unsupervised anomaly detection models tested on the Tennessee Eastman Process Simulation data. Bold values show the best results.

Model name	F1 score ($\pm$)	FDR ($\pm$)	Explain cause of fault?	Source
BeatGAN	0.959 (± 0.053)	0.927 (± 0.095)	$\times$	Zhou et al. (2019)
USAD	0.950 (± 0.041)	0.908 (± 0.070)	$\times$	Audibert et al. (2020)
GenAD	0.960 (± 0.043)	0.927 (± 0.075)	$\times$	Hua et al. (2023)
Donut	0.964 (± 0.049)	0.934 (± 0.085)	$\times$	Xu et al. (2018)
STORN	0.971 (± 0.034)	0.948 (± 0.062)	$\times$	Sölch et al. (2016)
KESA-AD	0.979 (± 0.035)	0.962 (± 0.063)	$\checkmark$	–

manufacturing system. Here, we go beyond fault detection by leveraging the inbuilt process knowledge from the construction of the graph shown in Fig. 5 to explain individual fault cases by suggesting the root cause. This is a crucial function for process manufacturing. Within complex multi-component systems, it is often difficult to pinpoint the exact reason for a given failure since connected units may all exhibit a single fault at the same time. Because of this, root cause investigation can be a time-consuming process and often may struggle to uncover the true cause of a given fault or failure. Therefore, this work can aid users by reducing investigation time through the explanation of detected faults using the dynamics of the subgraphs of each of the nodes to focus the investigation on the largest contributors to a given fault.

We can review the efficacy of this by reviewing the faults from the TEP case study. To analyze the performance of the framework in this case, we will focus specifically on faults IDV (1), (4)–(6), (11)–(12), (14)–(15) since these are faults with a known root cause as shown in Table 4. No comparison can be drawn to other unsupervised anomaly detection algorithms shown in Table 5 since they do not provide a similar breakdown of potential fault causes. Fig. 8 shows the top 5 identified causes for the fault cases tested. Given the high variability of the data and the complex nature of the system, the model is considered successful if it can identify the true cause (from Table 4) within the top 5 causes based on the root cause analysis.

Fig. 8 shows the breakdown of the change in subgraph anomaly scores before and after each fault was induced. It can be seen that KESA-AD correctly identifies the root cause in the top 5 for 75% of fault cases (red bars), and in 62.5% of cases, this is identified with the highest absolute change in anomaly score. Furthermore, in cases where the overall root cause is not identified, the model was still able to identify nodes that lie on the fault propagation pathway (orange bars). This shows a clear ability for the model to identify and extract nodes that are having a profound impact on the system and generating fault. It should be noted that these nodes are identified based on the change in the dynamic of their subgraphs — not explicitly on their behavior. This is important since fault-generating nodes can often appear within allowable or normal limits, it is only when their context concerning surrounding nodes is observed that it becomes clear where the problem lies.

The challenge in pinpointing the overall root cause becomes evident when examining the cases of IDV (1) and IDV (6). This challenge is closely linked to the configuration of the initial graph, as depicted in Fig. 5, where the two root cause nodes for the faults, XMEAS (4) and XMEAS (1) respectively, occupy the uppermost positions in the graph. It is noteworthy that both these nodes exhibit limited connectivity, characterized by low out-degrees (two and five, respectively) and a complete absence of in-degrees. Such deficient connections might reduce the model's ability to detect changes within the surrounding subgraph of these nodes, thereby impeding its ability to identify the root cause of the fault. In contrast, nodes with robust connectivity, like XMEAS (9) and XMEAS (11), emerge as readily recognizable root causes for faults IDV (4), IDV (11), IDV (12), and IDV (14). Hence, it becomes imperative for future research to delve into the influence of graph construction on the model's fault detection and diagnostic capabilities. This analysis underscores the significant impact of node degrees on the model's aptitude to identify potential root causes.

Table 6

Fault propagation pathways for IDV (1) and IDV (6) showing in bold the nodes detected as a cause of the fault by KESA-AD (Ji et al., 2021; Gharahbagheri et al., 2017b).

Fault	Fault Pathway
IDV (1)	(4, 16, 20) $\rightarrow$ (7, 13) $\rightarrow$ 1
IDV (6)	1 $\rightarrow$ 6 $\rightarrow$ 7 $\rightarrow$ 9 $\rightarrow$ 21

It is important to examine the cases of IDV (1) and IDV (6) more closely to understand why the model may have underperformed. While the primary root cause is not detected, the model does manage to identify nodes situated along the fault propagation pathway for both of these faults. The propagation pathway delineates the sequential impact of nodes in the system, commencing from the root node (identified in Table 4) and culminating at the affected node where the fault manifests. Table 6 outlines the fault pathways for IDV (1) and (6) (Ji et al., 2021; Gharahbagheri et al., 2017b). IDV (6) concerns the loss of Feed A to the reactor. We can see from the propagation pathway that when the immediate impacts of this are a reduction in the reactor feed rate, XMEAS (6), leading to a subsequent drop in pressure of the reactor (XMEAS (7)). Since pressure and temperature are inherently linked, the reactor temperature XMEAS (9) is also affected which causes a change in the reactor cooling water outlet temperature, XMEAS (21). KESA-AD was able to successfully detect the fault, however misdiagnosed the root cause. The model did, however, detect that XMEAS (9), the reactor temperature was one of the main affected nodes that lie along the propagation pathway and therefore KESA-AD is still able to aid in reducing investigation time.

IDV (1) concerns a fault in the A/C feed ratio which is not directly measured but is carried to the stripper via stream 4 (XMEAS (4)). The stripper pressure (XMEAS (16)) is the first measured change to the system, which affects the compressor work (XMEAS (20)). This results in a change of the reactor pressure through a change in the flow of stream A, as well as a change in product separator pressure (XMEAS (13)). Once the fault occurs with the stripper pressure and the compressor work, it propagates through the system and causes the reactor pressure (XMEAS (7)) and the product separator pressure. Lastly, the fault is propagated to the A feed flow (XMEAS (1)). The root cause of IDV (1), a step change in the A/C ratio of stream 4, is not directly measurable through the dataset which explains the misdiagnosis of the root cause by the KESA-AD model.

The key learnings from root cause diagnosis with KESA-AD are two-fold. Firstly, we observe that the extent of connectivity of the knowledge graph used impacts the diagnostic ability of the model. Where nodes have few connections, there is little information that can be gained about the node from its subgraph which makes accurate diagnosis challenging. Consideration should be given to the knowledge graph construction methods for future work to allow more comprehensive diagnosis. Secondly, we learn that the model struggles to identify the effect of external influences on the system. This can be addressed by exploring methods to deal with latent confounders. This factor also impacts the fault detection delay, as evidenced by Fig. 9.

5.4. Fault detection delay

Rapid fault detection is imperative for the use of FDD algorithms in real industrial settings. Fig. 9 shows the average fault detection delay

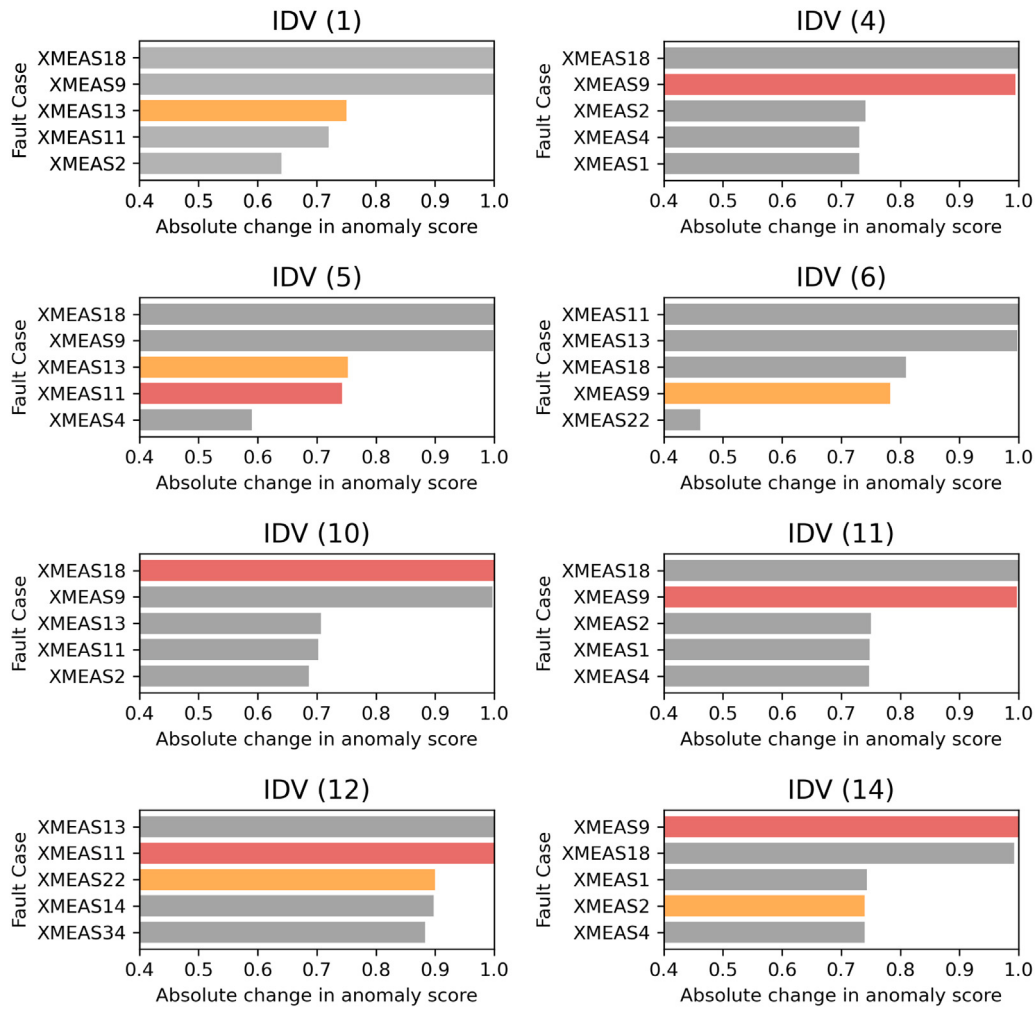


Fig. 8. Contribution plots showcasing the top five probable factors influencing the manifested fault. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

for each of the 15 tested fault cases representing the time between the induction of the fault, and the detection by the model. The model responds rapidly to detect faults, on average detecting the fault within 11.3 timesteps, providing a significant improvement over comparable deep learning FDD algorithms (Lomov et al., 2021). A large number of the faults are detected within a timestep, meaning a fault detection time of less than 3 min. IDV (6) is rapidly detected, despite the misdiagnosis of the root cause mentioned above. IDV (1), however, shows a high fault detection delay of 121 timesteps, which could be problematic for real-world applications. The root cause of IDV (1) is a step change in the A/C feed ratio, which is not a measured variable within the dataset. Therefore, there is a lag between the step change in the ratio and a subsequent change in stripper pressure. This lag is reflected in the detection delay.

Overall, we see that our model exhibits rapid fault detection for all cases, improving upon the detection delay of other deep-learning FDD models. We note that the fault detection delay is increased when the fault originates from a cause external to the dataset. This finding highlights the importance of incorporating external influences into future research, as it reflects the broader spectrum of faults encountered in real-world manufacturing environments.

6. Conclusion

This work explores the use of knowledge-enhanced machine learning methods for FDD in the process industries. Specifically, we propose

Knowledge-Enhanced Spatiotemporal Analysis for Anomaly Detection (KESA-AD), applied here to industrial anomaly detection. The importance of this method is in the ability to accurately detect process faults based on a combination of plant sensor data and specific domain knowledge and to highlight key root causes behind the existence of the fault. By providing this information to maintenance and operational teams we can reduce the time to effective intervention which ultimately reduces the likelihood of process failure and therefore promotes cost savings and improved process safety.

The proposed methodology was compared against state-of-the-art unsupervised fault detection models using the benchmark Tennessee Eastman process dataset. The results show the ability of KESA-AD to detect a range of complex manufacturing faults, including step changes and slow drift faults as well as random disturbances. The model demonstrates a high F1 score and fault detection rate in comparison to other models. Furthermore, in challenging fault detection cases where the performance of other models suffered, KESA-AD maintained high scores. These results underscore the significance of incorporating domain-specific knowledge as the key to achieving robust fault detection for the complex fault scenarios inherent to process manufacturing. By leveraging domain knowledge through the graph structure, we also demonstrate the ability to identify the root cause of detected faults by analyzing the dynamics of the subgraphs of each of the nodes.

This work highlights the importance of knowledge-enhanced AI frameworks as a future avenue for researchers applying FDD to process manufacturing. The flexibility of the graph convolutional approach

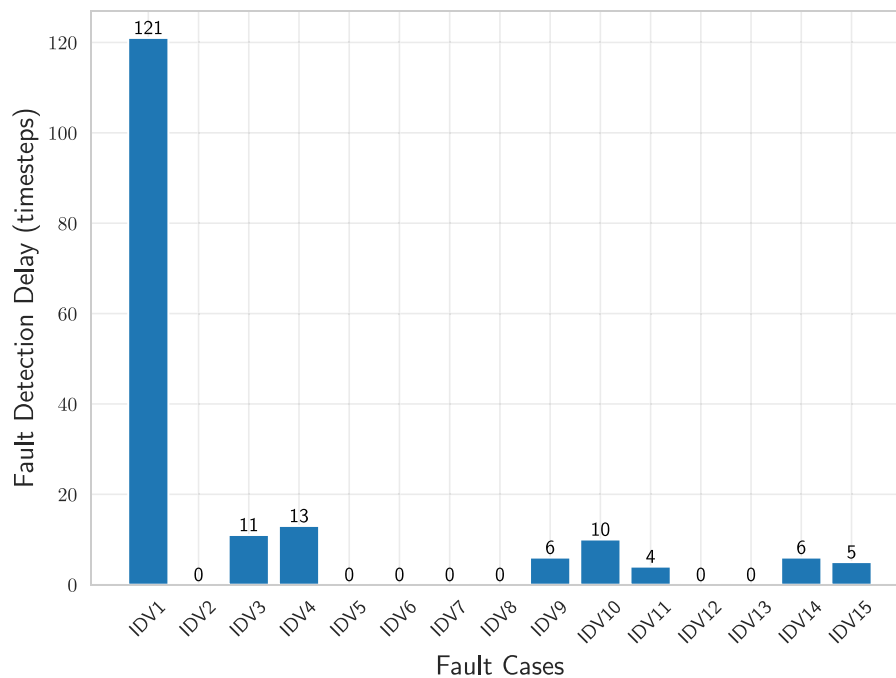


Fig. 9. The average fault detection delay between an induced fault and detection.

means process knowledge from a wide range of sources can be included without acyclicity requirements. It can therefore be easily applied equally to processes where cycles are inherent, such as processes with recycle streams. This factor, along with the inherent ability of deep neural networks to deal with non-linear data makes this solution a good fit for FDD in the process industries.

Future avenues for research should concern the construction of the process knowledge graph used to diagnose faults. In this work, we have used mechanistic profiles of the process to build knowledge graphs. This approach removes the bias of using operator input knowledge and circumvents the use of rule-based expert systems which could lead to misdiagnosis. However, it was shown that in cases where the mechanistic profiles are not as well defined leading to nodes with fewer connections, fault diagnosis suffers. This could be solved with pseudo-nodes to increase connectivity, or with additional inference techniques to augment the process knowledge. Furthermore, for applications in wider processes where mechanistic profiles for each unit are not available due to process complexity, for example, wastewater treatment, our approach would be difficult. However, our study demonstrates a methodology that is broadly applicable across process manufacturing and could aid in advancing FDD technology. This method could be further enhanced by applying causal discovery to affected nodes upon fault detection to aid in understanding the direction of fault propagation pathways to further aid in root cause diagnosis.

Overall, this paper has developed a framework aimed at supporting and assisting day-to-day maintenance operations by accurately identifying faults, and by providing insights to help determine and eliminate the root cause of the fault. This allows for timely and effective maintenance and operational intervention to prevent the fault case from propagating to failure. In turn, this increases both the profitability and safety of the process.

CRedit authorship contribution statement

Louis Allen: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Haiping Lu:** Writing – review & editing, Supervision, Conceptualization. **Joan Cordiner:** Writing – review & editing, Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

- Ahmed, M., Allen, L., Cordiner, J., 2022. Investigating machine learning techniques for effective predictive maintenance in industrial systems. In: 2022 AIChE Annual Meeting. AIChE.
- Audibert, J., Michiardi, P., Guyard, F., Marti, S., Zuluaga, M.A., 2020. USAD: Un-supervised anomaly detection on multivariate time series. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. KDD '20, Association for Computing Machinery, New York, NY, USA, pp. 3395–3404. <http://dx.doi.org/10.1145/3394486.3403392>.
- Bathelt, A., Ricker, N.L., Jelali, M., 2015. Revision of the Tennessee Eastman process model. IFAC-PapersOnLine 28, 309–314. <http://dx.doi.org/10.1016/J.IFACOL.2015.08.199>.
- Becraft, W.R., Lee, P.L., Newell, R.B., 1991. Integration of neural networks and expert systems for process fault diagnosis. ICJAI 832–837.
- Botre, C., Mansouri, M., Karim, M.N., Nounou, H., Nounou, M., 2017. Multiscale PLS-based GLRT for fault detection of chemical processes. J. Loss Prev. Process Ind. 46, 143–153. <http://dx.doi.org/10.1016/J.JLP.2017.01.008>, Difficulty of applying model-based techniques of process data due to highly correlated and non-Gaussian and non-stationary data. Use a TEP case study..
- Bruna, J., Zaremba, W., Szlam, A., Lecun, Y., 2014. Spectral networks and locally connected networks on graphs. In: International Conference on Learning Representations (ICLR2014), CBLS, April 2014.
- Chemweno, P., Pintelon, L., Jongers, L., Muchiri, P., 2016. I-RCAM: Intelligent expert system for root cause analysis in maintenance decision making. In: 2016 IEEE International Conference on Prognostics and Health Management, ICPHM 2016. Institute of Electrical and Electronics Engineers Inc., <http://dx.doi.org/10.1109/ICPHM.2016.7542830>.
- Chen, X., Wang, J., Ding, S.X., 2020. Complex system monitoring based on distributed least squares method. IEEE Trans. Autom. Sci. Eng. 18, 1892–1900. <http://dx.doi.org/10.1109/TASE.2020.3022924>.
- Chen, X., Wang, J., Ding, S.X., 2021. Complex system monitoring based on distributed least squares method. IEEE Trans. Autom. Sci. Eng. 18, 1892–1900. <http://dx.doi.org/10.1109/TASE.2020.3022924>.

- Chiang, L.H., Russell, E.L., Braatz, R.D., 2000. Fault diagnosis in chemical processes using Fisher discriminant analysis, discriminant partial least squares, and principal component analysis. *Chemometr. Intell. Lab. Syst.* 50, 243–252. [http://dx.doi.org/10.1016/S0169-7439\(99\)00061-1](http://dx.doi.org/10.1016/S0169-7439(99)00061-1).
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. EMNLP*, Association for Computational Linguistics, Doha, Qatar, pp. 1724–1734. <http://dx.doi.org/10.3115/v1/D14-1179>, URL <https://aclanthology.org/D14-1179>.
- Chung, J., Gulcehre, C., Cho, K., Bengio, Y., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.
- Deng, L., Lian, D., Huang, Z., Chen, E., 2022. Graph convolutional adversarial networks for spatiotemporal anomaly detection. *IEEE Trans. Neural Netw. Learn. Syst.* 33, 2416–2428. <http://dx.doi.org/10.1109/TNNLS.2021.3136171>.
- Ding, S., Zhang, P., Ding, E., Yin, S., Naik, A., Deng, P., Gui, W., 2010. On the application of PCA technique to fault diagnosis. *Tsinghua Sci. Technol.* 15, 138–144. [http://dx.doi.org/10.1016/S1007-0214\(10\)70043-2](http://dx.doi.org/10.1016/S1007-0214(10)70043-2), Comments on the use of PCA for fault detection..
- Don, M.G., Khan, F., 2019. Dynamic process fault detection and diagnosis based on a combined approach of hidden Markov and Bayesian network model. *Chem. Eng. Sci.* 211, 82–96. <http://dx.doi.org/10.1016/J.CES.2019.01.060>.
- Downs, J.J., Vogel, E.F., 1993. A plant-wide industrial process control problem. *Comput. Chem. Eng.* 17 (3), 245–255.
- Gharahbagheri, H., Imtiaz, S., Khan, F., 2017a. Combination of KPCA and causality analysis for root cause diagnosis of industrial process fault. *Can. J. Chem. Eng.* 95, 1497–1509. <http://dx.doi.org/10.1002/CJCE.22852>, <https://onlinelibrary.wiley.com/doi/full/10.1002/cjce.22852> <https://onlinelibrary.wiley.com/doi/abs/10.1002/cjce.22852> <https://onlinelibrary.wiley.com/doi/10.1002/cjce.22852>.
- Gharahbagheri, H., Imtiaz, S.A., Khan, F., 2017b. Root cause diagnosis of process fault using KPCA and Bayesian network. *Ind. Eng. Chem. Res.* <http://dx.doi.org/10.1021/acs.iecr.6b01916>, URL <https://pubs.acs.org/sharingguidelines>.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* 27.
- Hartung, F., Franks, B.J., Michels, T., Wagner, D., Liznerski, P., Reithermann, S., Fellenz, S., Jirasek, F., Rudolph, M., Neider, D., Leitte, H., Song, C., Kloepper, B., Mandt, S., Bortz, M., Burger, J., Hasse, H., Kloft, M., 2023. Deep anomaly detection on Tennessee Eastman process data. *Chem. Ing. Tech.* 95, 1077–1082. <http://dx.doi.org/10.1002/CITE.202200238>, URL <https://onlinelibrary.wiley.com/doi/10.1002/cite.202200238>.
- He, B., Chen, T., Yang, X., 2014. Root cause analysis in multivariate statistical process monitoring: Integrating reconstruction-based multivariate contribution analysis with fuzzy-signed directed graphs. *Comput. Chem. Eng.* 64, 167–177. <http://dx.doi.org/10.1016/J.COMPCHEMENG.2014.02.014>.
- Hong, W., Tian-You, C., Jin-Liang, D., Brown, M., 2009. Data driven fault diagnosis and fault tolerant control: Some advances and possible new directions. *Acta Automat. Sinica* 35 (6), 739–747.
- Hua, X., Zhu, L., Zhang, S., Li, Z., Wang, S., Deng, C., Feng, J., Zhang, Z., Wu, W., 2023. GenAD: General unsupervised anomaly detection using multivariate time series for large-scale wireless base stations. *Electron. Lett.* 59, e12683. <http://dx.doi.org/10.1049/ELL2.12683>, <https://onlinelibrary.wiley.com/doi/full/10.1049/ell2.12683> <https://onlinelibrary.wiley.com/doi/abs/10.1049/ell2.12683> <https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/ell2.12683>.
- Isermann, R., 1984. Process fault detection based on modeling and estimation methods—A survey. *Automatica* 20, 387–404. [http://dx.doi.org/10.1016/0005-1098\(84\)90098-0](http://dx.doi.org/10.1016/0005-1098(84)90098-0).
- Isermann, R., Ballé, P., 1997. Trends in the application of model-based fault detection and diagnosis of technical processes. *Control Eng. Pract.* 5, 709–719. [http://dx.doi.org/10.1016/S0967-0661\(97\)00053-1](http://dx.doi.org/10.1016/S0967-0661(97)00053-1).
- Ji, C., Ma, F., Wang, J., Wang, J., Sun, W., Cravotto, G., 2021. Real-time industrial process fault diagnosis based on time delayed mutual information analysis. *Processes* <http://dx.doi.org/10.3390/pr9061027>.
- Jia, N., Tian, X., Gao, W., Jiao, L., 2023. Deep graph-convolutional generative adversarial network for semi-supervised learning on graphs. *Remote Sens.* 15 (12), 3172.
- Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Kipf, T.N., Welling, M., 2017. Semi-supervised classification with graph convolutional networks. In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24–26, 2017, Conference Track Proceedings*. OpenReview.net, pp. 1–9, URL <https://openreview.net/forum?id=SJU4ayYgl>.
- Lee, S., Kim, H., Ro, Y., 2018. STAN: Spatio-temporal adversarial networks for abnormal event detection. In: *IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE*, pp. 1323–1327, URL https://ieeexplore.ieee.org/abstract/document/8462388/?casa_token=YyHsRm-4MSkAAAAA:zY-F2KHROpZ2AHOKYCa24ol_IWWW_sxE2RixKic6P_28yRHVoleBVSZrtnJJpAs4cu_yuGMGcg. Providing reference on combination of anomaly score from generator and discriminator.
- Lee, J.M., Qin, S.J., Lee, I.B., 2006. Fault detection and diagnosis based on modified independent component analysis. *AIChE J.* 52, 3501–3514. <http://dx.doi.org/10.1002/AIC.10978>, <https://onlinelibrary.wiley.com/doi/full/10.1002/aic.10978> <https://onlinelibrary.wiley.com/doi/abs/10.1002/aic.10978> <https://aiche.onlinelibrary.wiley.com/doi/10.1002/aic.10978>.
- Li, Y., Yu, R., Shahabi, C., Liu, Y., 2018. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In: *International Conference on Learning Representations*. URL <https://openreview.net/forum?id=SJIHXGWAZ>.
- Lokrantz, A., Gustavsson, E., Jirstrand, M., 2018. Root cause analysis of failures and quality deviations in manufacturing using machine learning. *Proc. CIRP* 72, 1057–1062. <http://dx.doi.org/10.1016/J.PROCIR.2018.03.229>.
- Lomov, I., Lyubimov, M., Makarov, I., Zhukov, L.E., 2021. Fault detection in Tennessee Eastman process with temporal deep learning models. *J. Ind. Inf. Integr.* 23, 100216. <http://dx.doi.org/10.1016/J.JII.2021.100216>.
- Moubray, J., 2001. *Reliability-Centered Maintenance*. Industrial Press Inc..
- Munro, G., 2017. ICH Q7 good manufacturing practice guide for active pharmaceutical ingredients (APIs). *ICH Qual. Guide.: Implement. Guide* 509–534.
- Nowlan, F.S., Heap, H.F., 1978. *Reliability-Centered Maintenance*. Technical Report, United Air Lines Inc San Francisco Ca.
- Onel, M., Kieslich, C.A., Pistikopoulos, E.N., 2019. A nonlinear support vector machine-based feature selection approach for fault detection and diagnosis: Application to the Tennessee Eastman process. *AIChE J.* 65, 992–1005. <http://dx.doi.org/10.1002/aic.16497>, URL <https://aiche.onlinelibrary.wiley.com/doi/10.1002/aic.16497>.
- Paolanti, M., Romeo, L., Felicetti, A., Mancini, A., Frontoni, E., Loncarski, J., 2018. Machine learning approach for predictive maintenance in industry 4.0. In: *2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, MESA 2018*. Institute of Electrical and Electronics Engineers Inc., <http://dx.doi.org/10.1109/MESA.2018.8449150>, Condition Monitoring

Decision Tree based classification with high recall on single cutting operation..
- Park, Y.J., Fan, S.K.S., Hsu, C.Y., 2020. A review on fault detection and process diagnostics in industrial processes. *Processes* 8, 1123. <http://dx.doi.org/10.3390/PR8091123>, <https://www.mdpi.com/2227-9717/8/9/1123/htm> <https://www.mdpi.com/2227-9717/8/9/1123>. Vol. 8, Page 1123.
- Russell, E.L., Chiang, L.H., Braatz, R.D., 2000. Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis. *Chemometr. Intell. Lab. Syst.* 51, 81–93. [http://dx.doi.org/10.1016/S0169-7439\(00\)00058-7](http://dx.doi.org/10.1016/S0169-7439(00)00058-7).
- Schramm, J., Wessels, K., 2015. *Preparing for an Aging Workforce: Oil, Gas and Mining Industry Report*. Technical Report, SHRM Research, URL www.shrmfoundation.org.
- Shin, H.J., Eom, D.H., Kim, S.S., 2005. One-class support vector machines - An application in machine fault detection and classification. *Comput. Ind. Eng.* 48, 395–408. <http://dx.doi.org/10.1016/J.CIE.2005.01.009>.
- Shuman, D., Narang, S., Frossard, P., Ortega, A., Vandergheynst, P., 2013. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Process.* URL <https://ieeexplore.ieee.org/abstract/document/6494675/>. Discussions on graph spectral theory and graph convolutions. Ideas used for GCGRU architecture.
- Sölch, M., Bayer, J., Ludersdorfer, M., van der Smagt, P., 2016. Variational inference for on-line anomaly detection in high-dimensional time series. In: *4th International Conference on Learning Representations*. Puerto Rico, URL https://www.researchgate.net/publication/301878310_Variational_Inference_for_On-line_Anomaly_Detection_in_High-Dimensional_Time_Series.
- Torres, J.F., Hadjout, D., Sebaa, A., Martínez-Álvarez, F., Troncoso, A., 2021. Deep learning for time series forecasting: A survey. *Big Data* 9 (1), 3–21.
- Vaidya, S., Ambad, P., Bhosle, S., 2018. Industry 4.0 - A glimpse. In: *Procedia Manufacturing*. Vol. 20, Elsevier B.V., pp. 233–238. <http://dx.doi.org/10.1016/j.promfg.2018.02.034>.
- Venkatasubramanian, V., Rengaswamy, R., Yin, K., Kavuri, S.N., 2003. A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. *Comput. Chem. Eng.* 27, 293–311. [http://dx.doi.org/10.1016/S0098-1354\(02\)00160-6](http://dx.doi.org/10.1016/S0098-1354(02)00160-6), Comment on 10 things that FDD software should contain..
- Wagner, D., Michels, T., Schulz, F.C., Nair, A., Rudolph, M., Kloft, M., 2023. Timesead: benchmarking deep multivariate time-series anomaly detection. *Transactions on Machine Learning Research*.
- Wu, D., Zhao, J., 2021. Process topology convolutional network model for chemical process fault diagnosis. *Process Saf. Environ. Prot.* 150, 93–109. <http://dx.doi.org/10.1016/J.PSEP.2021.03.052>.
- Xie, D., Bai, L., 2016. A hierarchical deep neural network for fault diagnosis on Tennessee-Eastman process. In: *Proceedings - 2015 IEEE 14th International Conference on Machine Learning and Applications, ICMLA 2015*. Institute of Electrical and Electronics Engineers Inc., pp. 745–748. <http://dx.doi.org/10.1109/ICMLA.2015.208>, Applied ANNs for fault classification in the Tennessee Eastman process dataset..
- Xu, H., Chen, W., Zhao, N., Li, Z., Bu, J., Li, Z., Liu, Y., Zhao, Y., Pei, D., Feng, Y., Chen, J., Wang, Z., Qiao, H., 2018. Unsupervised anomaly detection via variational auto-encoder for seasonal KPIs in web applications. In: *Proceedings of the 2018 World Wide Web Conference. WWW '18*, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, pp. 187–196. <http://dx.doi.org/10.1145/3178876.3185996>.

- Yang, W., Zhang, K., Hoi, S.C.H., 2023. A causal approach to detecting multivariate time-series anomalies and root causes. In: *International Conference on Learning Representations*.
- Yu, H., Khan, F., Garaniya, V., 2015. Modified independent component analysis and Bayesian network-based two-stage fault diagnosis of process operations. *Ind. Eng. Chem. Res.* 54, 2724–2742. <http://dx.doi.org/10.1021/IE503530V>, URL <https://pubs.acs.org/doi/abs/10.1021/ie503530v>.
- Yu, B., Yin, H., Zhu, Z., 2017. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In: *IJCAI International Joint Conference on Artificial Intelligence*. 2018-July, International Joint Conferences on Artificial Intelligence, pp. 3634–3640. <http://dx.doi.org/10.24963/ijcai.2018/505>.
- Zhang, J., Martin, E., Morris, A.J., 1995. Fault detection and classification through multivariate statistical techniques. *Proc. Am. Control Conf.* 1, 751–755. <http://dx.doi.org/10.1109/ACC.1995.529351>.
- Zhang, H., Zhao, S., Liu, R., Wang, W., Hong, Y., Hu, R., 2022. Automatic traffic anomaly detection on the road network with spatial-temporal graph neural network representation learning. *Wirel. Commun. Mob. Comput.* 2022, 1–12. <http://dx.doi.org/10.1155/2022/4222827>.
- Zheng, W., Chen, G., 2021. An accurate GRU-based power time-series prediction approach with selective state updating and stochastic optimization. *IEEE Trans. Cybern.* 52 (12), 13902–13914.
- Zhou, B., Liu, S., Hooi, B., Cheng, X., Ye, J., 2019. BeatGAN: Anomalous rhythm detection using adversarially generated time series. In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. ICJAI, pp. 4433–4439. <http://dx.doi.org/10.24963/ijcai.2019/616>.