Fast Fourier Transform-Based Synthetic Method for Chemical Process Data Augmentation and Fault Classification

Jie-Ning Chen,a Jia-Lin Kang,b\* Yuan Yaoa\*

aDepartment of Chemical Engineering, National Tsing Hua University, Hsinchu, 30013, Taiwan, ROC

bDepartment of Chemical and Materials Engineering, National Yunlin University of Science and Technology, Yunlin, 64002, Taiwan, ROC

jlkang@yuntech.edu.tw (Kang); [yyao@mx.nthu.edu.tw](mailto:yyao@mx.nthu.edu.tw) (Yao)

Abstract

Faults in industrial processes can lead to significant financial losses and severe safety concerns. Thus, timely identification of faults is crucial. Fault classification, usually based on supervised machine learning approaches, uses historical process data to extract fault characteristics and classify them accurately. However, the limited availability of fault data in real-world plants hampers effective model training, as conventional supervised learning depends on sufficient training data. Consequently, creating synthetic fault data that captures the temporal evolution of variable trajectories can be a viable solution to enrich the dataset. In this study, we proposed a synthetic method using Fast Fourier Transform (FFT) to generate fault data, applied to the Tennessee Eastman Process (TEP) dataset to augment fault data and improve classification accuracy. Besides, we use the gradient boosting classifier as our classifier to determine whether our synthetic fault data can assist the detection post fault occurred by comparing the testing accuracy of the classifiers training with and without fault data augmentation. The case study results illustrate the feasibility of the proposed method.

**Keywords**: fast Fourier transform, fault classification, imbalanced data, data augmentation.

* 1. Introduction

In chemical industry operations, it is essential to maintain the normal functioning of process variables, including the concentration, temperature, and flow rates of materials flowing into and out of each unit. This is crucial for achieving mass production and preventing malfunctions. Fault classification, typically reliant on supervised machine learning techniques, plays a significant role in enhancing fault detection and diagnostics. This approach leverages historical process data to accurately identify and categorize fault characteristics. However, a major challenge in this approach is the dependency on extensive training data, which is often scarce in real-world plants due to the infrequent nature of historical fault occurrences (Jiang and Ge, 2020). Consequently, generating synthetic fault data to augment the training dataset emerges as a practical strategy to improve the accuracy of fault classification.

Industrial processes involve the operation of multiple units and the interaction of diverse components, resulting in process data that is typically represented as multivariable time series. Generally, the emergence of a process issue is not attributable to the abnormality of a single variable. Consequently, it is essential to generate synthetic fault data that accurately reflects the temporal evolution of faults. The generative adversarial network (GAN) is a popular method for dataset augmentation, prized for its ability to generate a wealth of data from random noise. However, training GAN models can be challenging due to the min-max competition between the generator and the discriminator. Often, the discriminator rapidly outpaces the generator, leading to the latter's underperformance and a tendency to produce overly conservative data outputs (Klopries and Schwung, 2024). This imbalance can be particularly problematic when generating time series data akin to industrial process data, as GANs may struggle to capture the temporal evolution of the series, instead generating data that represents an average of the entire series.

we introduce a novel data synthesis method utilizing the fast Fourier transform (FFT) to address the challenges associated with augmenting multivariable time series data. By applying FFT, we transform our time series fault data into the frequency domain. This transformation reveals the amplitude compositions of sinusoidal waves across various frequencies, simplifying the analysis process. We then generate synthetic fault data by applying the Inverse Fast Fourier Transform (IFFT) to these processed frequency-domain data, effectively reconstructing time series data that retain the essential characteristics of the original faults.

* 1. Methodology
     1. Fast Fourier transform-based data synthesis method

FFT is an algorithm designed to compute the discrete Fourier transform (DFT) with enhanced efficiency. It transforms data from the time domain into the frequency domain (Rapuano and Harris, 2007). The DFT is mathematically represented as shown in Eq. (1), where *W* denotes , with *N* being the size of the data set. In this equation, *X*(*k*) represents the frequency domain data obtained post-transformation, while *x*(*n*) corresponds to the original time domain data collected at sampling time *n*.

fault or normal (1)

(2)

(3)

(4)

The framework of the proposed fault data synthesis method is depicted in Figure 1. The process initiates by applying FFT to each process variable, as per Eq. (1). This step transforms the data from the time domain (*x*fault(*n*) or *x*normal(*n*)) to the frequency domain (*X*fault(*k*) or *X*normal(*k*)). Next, we analyse the differences between normal and fault operation data in the frequency domain (Eq. (2)), identifying what termed “waveform variations” (). These variations not only describe the disturbance post fault occurs but also capture the evolution of the fault. Subsequently, we randomly overlay these waveform variations onto other normal operation data sets in the frequency domain (), as depicted in Eq. (3). This procedure generates the frequency

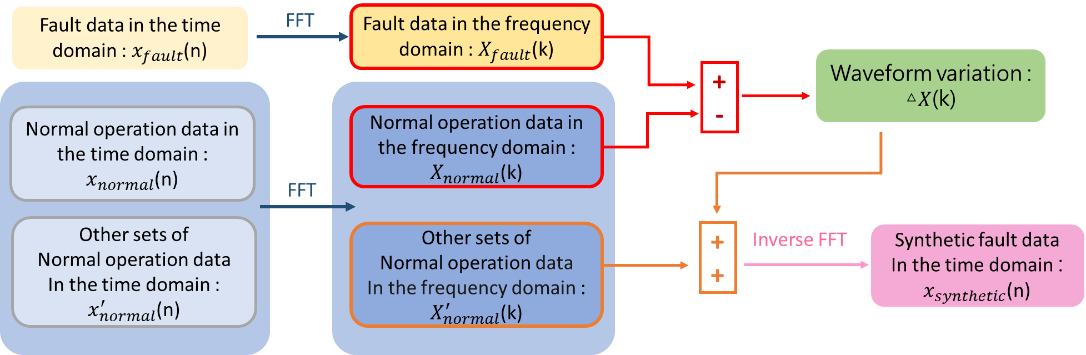


Figure1. The flow chart of our fault data synthetic method based on FFT

compositions for synthetic fault data (). The final step involves applying inverse FFT (as per Eq. (4)) to convert these compositions back into the time domain, resulting in synthetic fault data () that aligns in the same format with the original process data.

* + 1. Gradient boosting

After integrating the measured process data with the synthetic fault data, we employ gradient boosting (Friedman, 2002) as the classifier for fault classification. Gradient boosting is a powerful algorithm that sequentially combines multiple “weak learners” to form a stronger model. Each learner in the sequence is trained using the residual errors from the previous model as its target. The optimal model we aim to construct can be described by Eq. (5), where *F* represents the cumulative model, with the subscript *m* denoting the sequence of the models. The term *h* refers to the new model being trained, which will be added to *F*,and *ρ* is a small positive coefficient, typically ranging between 0 and 1.

(5)

Let represent the observed value and is the predict value from the model *Fm*. The objective of gradient boosting is to fit the new model ℎ*m*​ to the residuals of *Fm* (as shown in Eq. (6)). This approach allows each new model to correct the errors made by the previous models, progressively improving the overall accuracy of the classifier.

(6)

In our study, we utilized decision trees as the weak learners within the gradient boosting framework, while the total number of weak learners was chosen to be 100. The loss function selected for optimization is the multinomial deviance. We set the learning rate at 0.01 to ensure gradual model improvement. During the boosting process, a maximum of 52 features were randomly chosen at each split for every tree, providing a balance between diversity and model complexity. Additionally, we constrained the size of each decision tree by limiting the number of nodes to a maximum of 21.

* 1. Case Study
     1. Tennessee Eastman Process

The Tennessee Eastman process (TEP) (Rieth et al., 2017) dataset was used as the case study to illustrate the feasibility of the proposed method. The TEP encompasses five key units: a reactor, a product condenser, a vapour-liquid separator, a recycle compressor, and a product stripper. The dataset encompasses 21 distinct types of faults, including step changes, random variations, slow drifts, sticking, and some unknown types. A comprehensive description of each variable and fault type is available in reference (Downs and Vogel,1993). In each run, there are 52 variables recorded over 500 sampling points, with the exception of fault 6. Fault 6 involves a shutdown occurring 6 hours after the fault onset, resulting in a total of 140 sampling points. Faults are introduced after an hour of normal operation. Consequently, the initial twenty sampling points represent normal operational conditions, while the subsequent 480 points correspond to data during the fault condition.

* + 1. Fault data synthesis by FFT

We applied FFT to the TEP dataset for the purpose of synthesizing fault data. Given that FFT is designed to process one variable at a time, we systematically applied FFT to the data of each variable sequentially. To account for the fact that the frequency components identified by FFT can vary with different lengths of input data, we implemented a moving window approach. This approach involved slicing the data using a window of 50 sampling points and a step size of 10 sampling points. The next step involved calculating the waveform variations for each fault type. This was achieved by subtracting the amplitude composition of the normal operational data from that of the fault data. We then integrated these waveform variations with the amplitude compositions of normal operational data, which had not been previously used in the waveform variation calculation, to create new amplitude compositions representative of the faults. Finally, we generated time-series synthetic fault data by applying inverse-FFT to these newly formed amplitude compositions.

In this study, we assumed that the known historical data for each fault type was represented by the first run of data in the TEP dataset. In this context, 140 sampling points were available for fault 6 and 500 sampling points for other fault types as well as normal operational data. To generate the synthetic fault data, five runs of normal operational data were randomly selected.

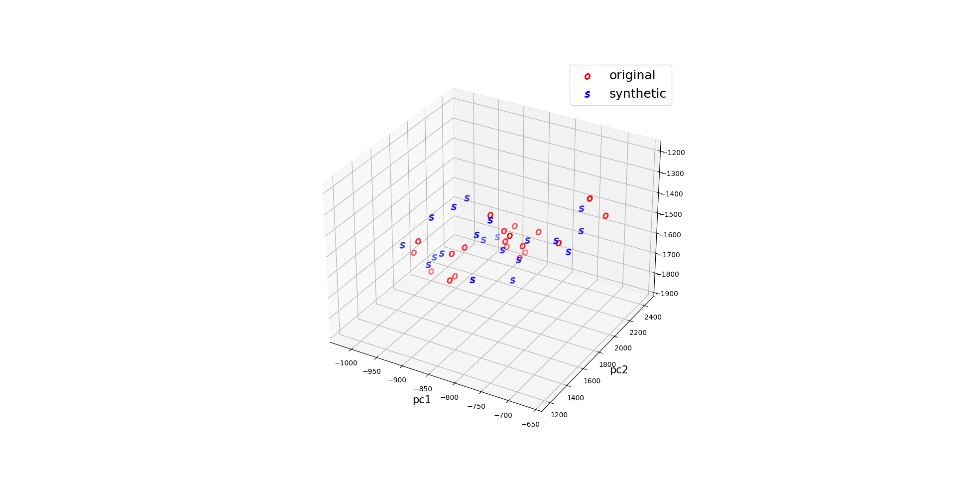
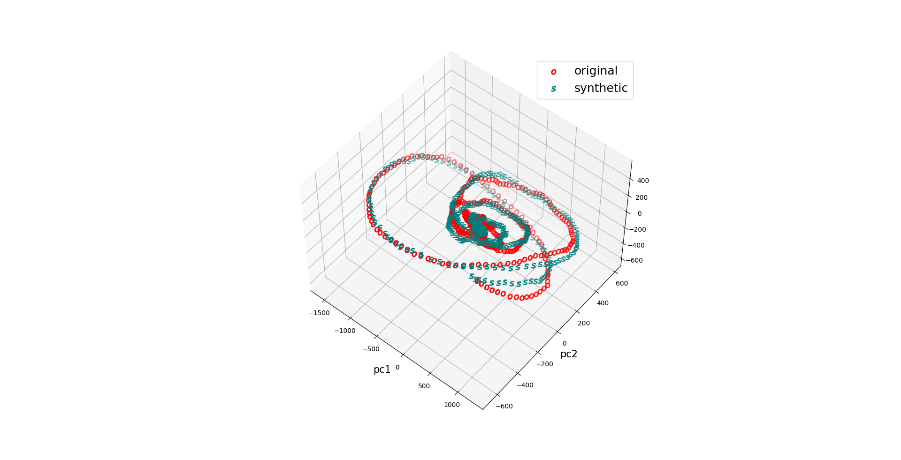
* + 1. Classifier training

Before training the classifier, the process data was standardized and then pre-processed by applying a moving window technique which involves a window length of 20 sampling points and a step size of 1 sampling point. Consequently, for each run of normal operation data, this process yielded 461 data windows. For fault 6, which has a shorter duration, we obtained 101 data windows. For other fault types, we generated 181 data windows representing the transition state and 261 windows for the steady state. The first ten hours of data following a fault occurrence was categorized as transition data, with the subsequent data classified as steady state. For training purposes, we included all windows of transition data and a selection of 90 windows from the steady state data. For testing, we used all available data windows.

In terms of training the classifier, we initially used five runs of normal operation data and one run of data for each fault type, prior to data augmentation. Post-augmentation, we enriched the training dataset with an additional 10 runs of synthetic data for each fault type.

* 1. Results And Discussions

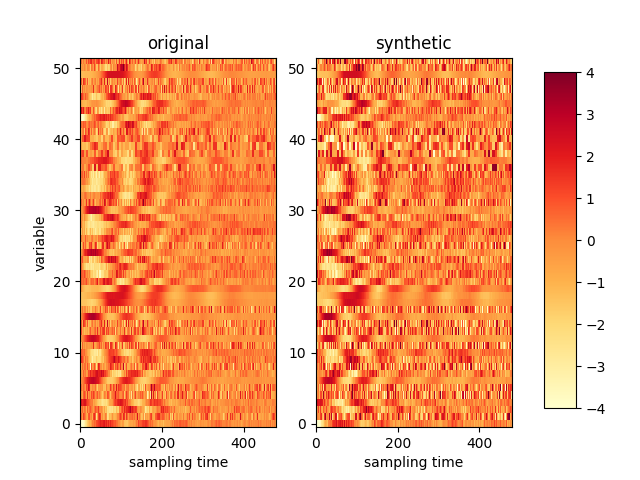
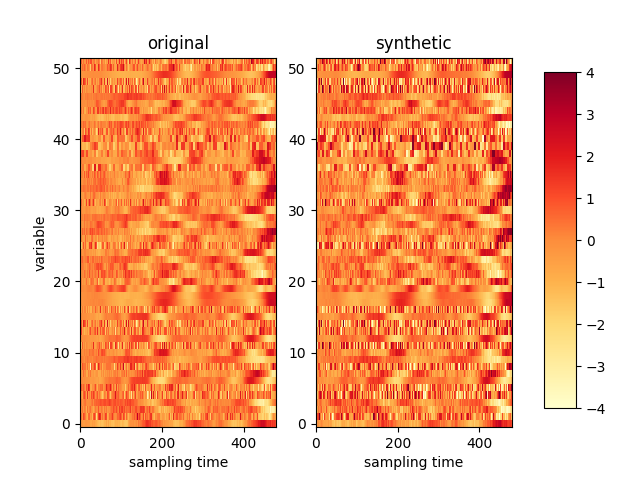
For data visualization purposes, we projected both the original fault data from the TEP dataset and the synthetic fault data into a common coordinate system defined by three principal component axes: pc1, pc2, and pc3. First, we represented each run of fault process data as a single point in the PCA score plot. This was done to assess whether the

(a) (b)

Figure 2. PCA score plot of fault 7:

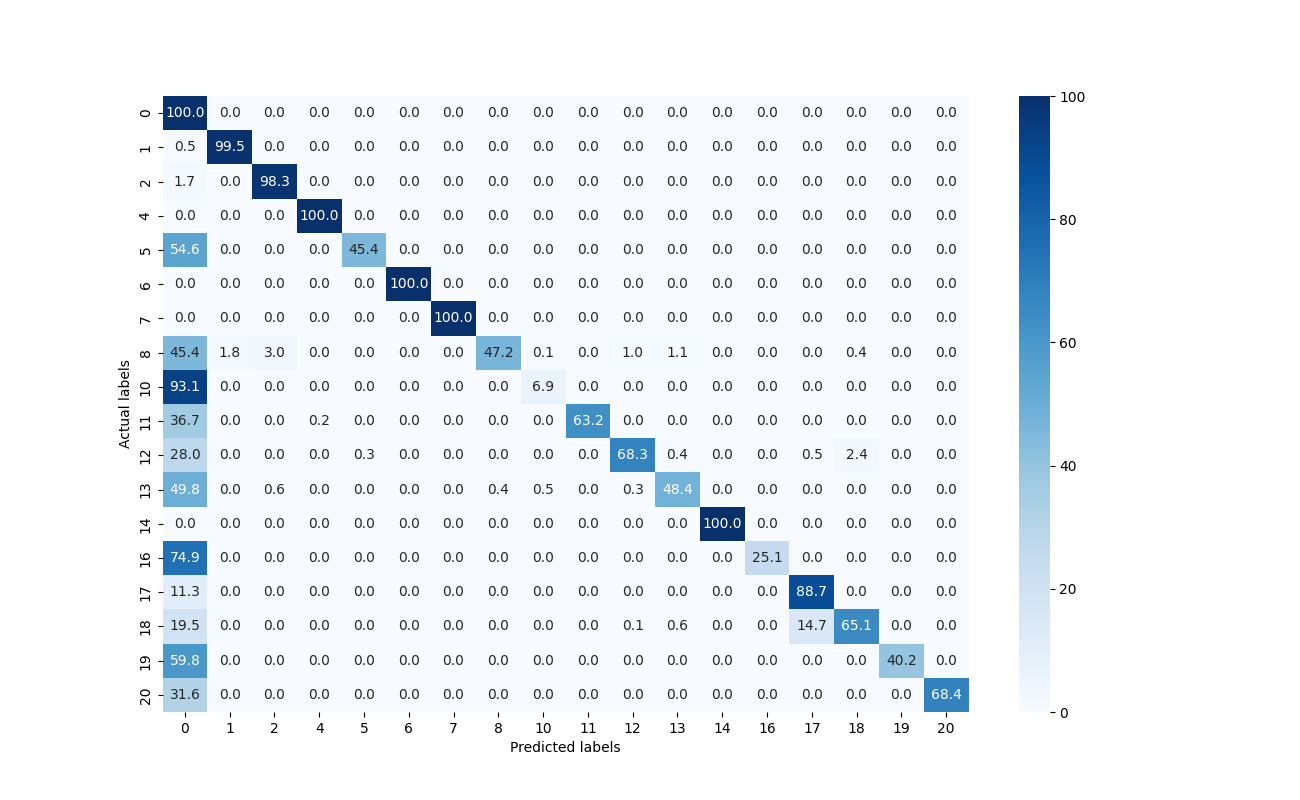
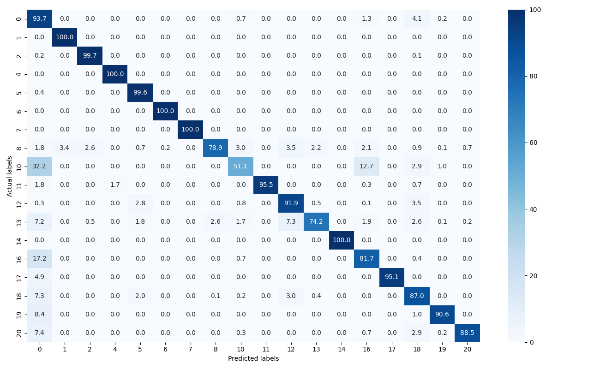
(a) each point represents one run; and (b) each point represents one window

(a) (b)

Figure 3. Heatmaps comparing original and synthetic fault data:

(a) fault 1; and (b) fault 8

(a) (b)

Figure 4. Confusion matrices of classification accuracy (%):

(a) before and (b) after data augmentation

overall information and diversity of our synthetic fault data were comparable to that of the original fault data. The results, as depicted in Figure 2 (a), indicated that the distributions of most synthetic faults closely resembled those of the original faults. This similarity demonstrates the effectiveness of our method in capturing the essential characteristics of faults while maintaining diversity. Subsequent analysis involved representing each moving window of data as a point in the PCA score plot. This approach allowed us to determine whether our synthetic data successfully captured the temporal evolution of fault trends. The results, shown in Figure 2 (b), revealed that our synthetic fault data closely followed the trajectory of the original fault data, indicating our method’s capability to replicate fault trends in time series relationships.

We also evaluated our synthetic data using heatmaps, which visually represent the numerical fluctuations of the 52 variables through color variations. The results indicated that the numerical fluctuations in our synthetic fault data closely mirrored those in the original data, confirming the successful extraction of fault trends (Figure 3).

To ascertain the impact of our synthetic data on fault classification effectiveness, we trained two gradient boosting classifiers. The first classifier was trained using the original, imbalanced TEP dataset, while the second was trained with our augmented dataset, which included the synthetic data. The results demonstrated a significant improvement in testing accuracy, from 75.08% to 91.40%, before and after data augmentation, respectively. This marked improvement, as shown in Figure 4, validates the efficacy of our data augmentation approach in addressing the challenges posed by imbalanced datasets in fault classification problems.

* 1. Conclusions

In this study, we investigated the effectiveness of time series data augmentation in chemical industrial processes, particularly focusing on datasets with imbalanced fault distribution, and examined whether the inclusion of generated data could enhance fault classification accuracy. We applied FFT to analyse the waveform compositions of various faults and generate synthetic fault data, the characteristics of which were visualized using PCA score plots and heatmaps. These visualizations demonstrated that the synthetic data successfully captured the dynamic and sequential properties of the fault progression over time. Additionally, the overall variable fluctuations in the synthetic data closely mirrored those in the original dataset, encompassing both the transition and steady states following a fault occurrence. To quantify the impact of data augmentation on fault classification, we compared the confusion matrices of classifiers trained with and without the augmented data. These comparisons clearly showed that the inclusion of the synthetic data led to a significant improvement in classification accuracy.

Acknowledgement

This work was supported in part by the National Science and Technology Council under project numbers NSTC 112-2221-E-007-015

References

Downs, J.J. and E.F. Vogel, A plant-wide industrial process control problem. Computers &

chemical engineering, 1993. 17(3): p. 245-255.

Friedman, J.H., Stochastic gradient boosting. Computational statistics & data analysis, 2002.

38(4): p. 367-378.

Jiang, X. and Z. Ge, Data augmentation classifier for imbalanced fault classification. IEEE

Transactions on Automation Science and Engineering, 2020. 18(3): p. 1206-1217.

Klopries, H. and A. Schwung, ITF-GAN: Synthetic time series dataset generation and

manipulation by interpretable features. Knowledge-Based Systems, 2024. 283: p. 111131.

Rapuano, S. and F.J. Harris, An introduction to FFT and time domain windows. IEEE

instrumentation & measurement magazine, 2007. 10(6): p. 32-44.

Rieth, C.A., et al., Additional tennessee eastman process simulation data for anomaly detection

evaluation. Harvard Dataverse, 2017. 1: p. 2017.