Journal of Building Engineering 76 (2023) 107127

Contents lists available at ScienceDirect

Journal of Building Engineering

journal homepage: www.elsevier.com/locate/jobe

Full length article

A data-driven approach for fault diagnosis in multi-zone HVAC systems: Deep neural bilinear Koopman parity

Fatemeh Negar Irani [a], Mohammadhosein Bakhtiaridoust [a], Meysam Yadegar [a], Nader Meskin [b,*]

[a] Department of Electrical and Computer Engineering, Qom University of Technology, Qom, Iran
[b] Department of Electrical Engineering, Qatar University, Doha, Qatar

ARTICLE INFO

Keywords:
HVAC system
AHU
Koopman operator
Sensor fault detection and isolation
Bilinear system
Data-driven
Parity-space method
Deep learning

ABSTRACT

Sensor faults in heating, ventilation, and air conditioning (HVAC) systems are inevitable and result in significant energy waste. This paper presents an innovative data-driven approach for sensor fault detection and isolation in multi-zone HVAC systems. The proposed solution integrates bilinear Koopman model realization, deep learning, and bilinear parity-space. A deep neural network realizes a bilinear model, enabling bilinear parity-space sensor fault detection and isolation. This yields a reliable, accurate, and interpretable data-driven framework. The method requires no prior HVAC dynamics knowledge, relying solely on normal operation data. It diagnoses additive, multiplicative, and complete failure sensor faults while minimizing false alarms, even with severe faults. A four-zone HVAC system is simulated in TRNSYS as a case study to demonstrate the performance and efficacy of the proposed approach. The proposed bilinear deep Koopman model realization is utilized to develop a bilinear model for the four-zone HVAC system. The bilinear model is then used for designing the bilinear parity-space. Further, considering various failure scenarios, the proposed sensor fault detection and isolation framework demonstrates promising diagnosis performance. Finally, a comparison is conducted to showcase the advantages of the proposed method over earlier works based on PCA and neural networks.

1. Introduction

Since the building construction sector is responsible for a significant proportion of the world's energy consumption, it is of great importance to maintain the thermal comfort of occupants with the minimum energy usage. The heating, ventilation, and air conditioning (HVAC) system is a major source of energy consumption in any residential and commercial building [1]. Consequently, any malfunction in this system can lead to a substantial loss of energy as well as thermal discomfort, poor indoor air quality, and low productivity [2]. Hence, the fault diagnosis and isolation (FDI) in the HVAC system is crucial to improve its reliability, efficiency, and performance and to provide preventive maintenance.

Sensor faults are pervasive in building operations, and accordingly, sensor fault detection and isolation (SFDI) methods are among the most commonly investigated research fields. In general, SFDI methods emerged for the building HVAC system can be classified into model-based and data-driven approaches [3]. The model-based SFDI techniques use the analytical model of the process [4–6] and hence, they require an adequate and precise understanding of the HVAC system mathematical model. The HVAC system is generally composed of mechanical and electrical systems, including cold and heat source systems, air handling systems,

* Corresponding author.
E-mail address: nader.meskin@qu.edu.qa (N. Meskin).

https://doi.org/10.1016/j.jobe.2023.107127
Received 7 April 2023; Received in revised form 12 June 2023; Accepted 15 June 2023
Available online 20 June 2023

Nomenclature

Symbol Description Units

T_A	temperature of zone A [°C]
T_B	temperature of zone B [°C]
T_C	temperature of zone C [°C]
T_D	temperature of zone D [°C]
T_O	outlet air temperature of the AHU [°C]
T_{amb}	ambient temperature [°C]
U_A	zone A inlet air control signal [%]
U_B	zone B inlet air control signal [%]
U_C	zone C inlet air control signal [%]
U_D	zone D inlet air control signal [%]
U_W	inlet water to cooling coil control signal [%]
n_s	number of snapshots
k	time step
$\tilde{\mathcal{K}}$	approximated Koopman operator
$\mathbf{R}$	lifting encoder
s	parity space order
ℓ_D	detection threshold
ℓ_I	isolation threshold
η	detection residual
σ_A	zone A isolation residual
σ_B	zone B isolation residual
σ_C	zone C isolation residual
σ_D	zone D isolation residual
σ_O	outlet air temperature of the AHU isolation residual

Abbreviation Description

HVAC	heating, ventilation, and air conditioning
AHU	air handling unit
VAV	variable air volume
TRNSYS	transient system simulation
PCA	principle component analysis
AANN	auto associative neural network
FDI	fault diagnosis and isolation
SFDI	sensor fault detection and isolation
EDMD	extended dynamic mode decomposition
PID	proportional integral derivative
LOE	loss of effectiveness
PD	precision degradation
CF	complete failure
RMSE	root mean square error
SPE	squared prediction error
SVDD	support vector data description
SVM	support vector machine
ICA	independent component analysis
EEMD	empirical mode decomposition
AFT	active functional testing

terminal devices, and multi-sensor monitoring systems, which are distributed throughout the building [7]. Hereupon, establishing an explicit model could be challenging for large-scale HVAC systems and may result in inaccurate modeling which could deteriorate the performance of model-based FDI methods.

NNs	neural networks
WPM	weather and schedule-based pattern matching
FPCA	feature based principal component analysis
RACNN	rule-based method and convolutional neural network
DisBN	discrete Bayesian Network
RF	random forest

Table 1
Summary of the literature review on data-driven SFDI for HVAC systems.

Ref.	Method	Data source	Training with healthy data	Fault detection	Fault isolation	Fault types				
						Bias	Drift	PD	CF	LOE
[8]	SVM	Simulation (TRANSYS)	✓	✓		✓				
[9]	SVM & RBF & KPCA	Simulation (HVACSIM+)	✓	✓	✓	✓				
[10]	ICA	Operational	✓	✓	✓	✓				
[11]	Probabilistic PCA	-		✓		✓				
[12]	PCA & pattern matching	Operational		✓		✓				
[13]	PCA & clustering analysis	Simulation	✓	✓	✓	✓				
[14]	Satizky-Golay PCA	Operational	✓	✓	✓	✓				
[15]	PCA	Simulation (TRANSYS)	✓	✓	✓	✓				
[16]	PCA & AFT	Simulation (Modelica)	✓	✓	✓	✓				
[17]	PCA & clustering	Operational	✓	✓	✓	✓				
[18]	PCA	Operational	✓	✓		✓				
[19]	PCA & EEMD	Simulation (Matlab) & operational	✓	✓		✓	✓			
[20,21]	WPM-FPCA	operational	✓	✓		✓				
[22]	ANN & Elman NN	Simulation		✓	✓	✓				
[23]	NNs	Simulation (TRANSYS)		✓	✓	✓	✓		✓	
[24]	NNs & clustering	Simulation (TRANSYS)		✓	✓	✓	✓		✓	
[25]	Fuzzy NN	Simulation (Matlab)		✓	✓	✓				
[26]	NN & fractal analysis	Simulation		✓		✓	✓			
[27]	Recurrent NN	Simulation (Modelica) & Operational	✓	✓	✓	✓				
[28,29]	Auto-Associative Neural Network	Simulation (TRNSYS)	✓	✓	✓	✓	✓			
[30]	RACNN	Operational		✓	✓	✓	✓			
[31]	Decentralized Boltzmann	Operational	✓	✓	✓	✓	✓			
[32]	WPM-DisBN	Operational	✓	✓	✓	✓				
[33]	PCA & RF	Operational	✓	✓	✓	✓				
This paper	Deep Neural Bilinear Koopman Parity (proposed method)	Simulation (TRNSYS)	✓	✓	✓	✓	✓	✓	✓	✓

Data-driven SFDI methods have been receiving considerable interest in recent years since they do not require the system analytical model and their performance only depends on process data, which enables them to investigate various solutions for the fault diagnosis problem. Hence, there is a vast amount of literature on data-driven SFDI in HVAC systems such as support vector machine (SVM) [8,9], support vector data description (SVDD) [34], independent component analysis (ICA) [10], principle component analysis (PCA) [11–21], neural networks (NNs) [22–32], and hybrid methods [33].

Regarding the PCA-based approaches, in [16], an SFDI method for the air handling unit (AHU) is proposed by combining PCA with active functional testing (AFT). In [13], PCA with clustering analysis is used for SFDI in AHU. Also, a mixture of probabilistic PCA models is used for sensor fault diagnosis in [11]. The work in [17] develops a combination of density-based clustering with PCA for SFDI in screw chillers. Further, in [19], a combination of empirical mode decomposition (EEMD) and PCA is used for sensor fault diagnosis in the chiller, in which EEMD is used for denoising. In [20], an innovative fault detection approach based on a weather and schedule-based pattern matching (WPM) procedure and a feature based principal component analysis (FPCA) procedure is produced and then the experimental results are provided in [21]. These PCA-based methods are not capable of detecting and isolating all types of HVAC faults and are likely to have missed alarms, especially in the case of small-magnitude faults [2].

On the other hand, neural networks have shown great potential in recent studies [35] particularly for detecting and isolating faulty sensors in HVAC systems, ensuring effective building operation [36]. A self-adaptive SFDI approach is proposed in [22] for the local system of AHU using back-propagation neural network models and the Elman neural network. In [23], the dual neural networks combined strategy is proposed for SFDI in the supply air temperature of the AHU, where the basic and auxiliary neural networks are developed and combined by allocating the weighting factors of the two neural networks using PCA. Further, a robust FDI method for AHU is presented using integrating the neural network and subtracting the clustering analysis in [24]. In [25], the artificial neural networks and fuzzy logic are used for FDI in variable air volume (VAV) boxes. The work in [26] presents a method based on a neural network pre-processed by wavelet and fractal for SFDI in AHU. In [27], a fault diagnosis approach is developed using recurrent neural networks based on local FDI agents designed for HVAC subsystems. The works in [28,29] use a semi-supervised data-driven method for sensor fault diagnosis in HVAC systems based on the auto associative neural network (AANN). In [30], an FDI method is proposed using the rule-based method and convolutional neural network (RACNN) for sensor and complicated fault diagnosis in AHU. In [31], a decentralized Boltzmann-machine-based method is developed for SFDI in AHU. The work in [32], presents an anomaly diagnosis framework based on a discrete Bayesian Network (DisBN) and a weather and schedule information-based pattern matching (WPM) method. Also, a hybrid method based on the combination of the PCA, time series anomaly detection, and random forest (RF) classifier is presented in [33] for fault diagnosis in AHUs.

The summary of the aforementioned literature on SFDI in the HVAC system is presented in Table 1. One of the drawbacks of the reviewed works, as in [11,12,22–26,30] is the requirement of a sufficient amount of faulty data for SFDI, which can be costly in practice. Moreover, some other methods that require only healthy data address only the fault detection problem and are not concerned with fault isolation, as in [8,18,19]. Furthermore, some of the SFDI solutions are computationally complex and time-consuming, and the majority of them are considered to have a detectable fault range, which has an upper and lower limit in most cases as in [28,29]. This results in performance and reliability degradation of the fault diagnosis when a fault with a magnitude over the detectable range occurs. Moreover, some of these methods suffer from having false or missed alarms even when the fault is within their detectable range, as in [22,28,29,31].

In this paper, a novel data-driven SFDI method is proposed for multi-zone HVAC systems by integrating the bilinear Koopman model realization, deep learning, and bilinear parity-space. The main goal of this work is to provide a reliable, accurate, and interpretable SFDI method that does not require any prior knowledge regarding the HVAC dynamics and only uses the data collected from the normal operation of the system. As shown in Table 1, the majority of earlier works considered only bias and drift sensor faults. However, the method presented in this paper is capable of detecting and isolating all types of sensor faults with various severity levels, including the bias, drift, precision degradation (PD), complete failure (CF), and loss of effectiveness (LOE). As stated above, false/missed alarms triggered within the detectable fault range are one of the main drawbacks of the majority of the earlier solutions to data-driven SFDI in HVAC systems. As a result, the reliability of these solutions for fault diagnosis is dependent on the type and severity level of the fault. Nevertheless, our proposed method does not lead to false/missed alarms within the detectable fault range, and by assigning appropriate thresholds, the residuals are sensitive to each type of sensor fault with a magnitude over the minimum diagnosable fault range. Further, in contrast to some other works, as in [28,29], our method can handle more severe faults in terms of the fault's amplitude. The main distinction between the current work and those in [37,38] is that in this paper, an architecture is proposed to obtain the bilinear Koopman model realization instead of the linear model and the superiority of the bilinear Koopman model realization is demonstrated over the linear Koopman model realization proposed in [37,38] in terms of accuracy and dimension of the predictor. The key features of this study are as

1. A global bilinear model realization scheme is proposed for the multi-zone HVAC system using the Koopman operator and deep neural network. The proposed scheme does not require the system dynamics to be known and uses only the healthy data of the system's normal operation to obtain an interpretable model realization. In addition, the proposed realization is extendable and can be used for more complicated large-scale HVAC systems.
2. The bilinear parity-space SFDI method is formulated according to the realized bilinear model. Then, a data-driven solution is provided for the HVAC SFDI problem, which is capable of detecting and isolating all types of sensor faults and is completely accurate and reliable within the detectable fault range.
3. The proposed method leads to a minimum false alarm rate within the detectable range. Furthermore, the performance of the proposed method remains reliable in the case of the occurrence of severe faults in terms of the fault's amplitude.
4. To demonstrate the capabilities of the proposed method, a four-zone HVAC system with a cooling application is simulated using the transient system simulation (TRNSYS) program, which is a versatile, graphical-based software widely used in the design and optimization of energy systems.

The remainder of this paper is structured as follows. In Section 2, the under-study 4-zone HVAC system and its simulation using TRNSYS are elaborated. Brief preliminaries on the Koopman operator theory, the Koopman bilinear model realization, and the problem formulation are provided in Section 3. The proposed deep neural Koopman model realization scheme and the parity-space SFDI method are described in Section 4. In Section 5, the deep neural bilinear model realization is performed for the 4-zone HVAC system, and a comparison is then conducted to demonstrate the capability of the bilinear model realization over the linear model realization. Then, the bilinear model realization is used to construct the parity-space, and the proposed SFDI method results are provided for all types of sensor faults. In Section 6, a comparison study is conducted to show the efficiency and benefits of the proposed approach over other earlier works based on PCA and AANN. Finally, the conclusions are drawn in Section 7.

2. System description and simulation

A typical HVAC system includes various sub-systems, such as an air handling unit (AHU), air distribution system, fluid chiller/heater system, etc., that couple closely in operation. The AHU is mainly composed of a cooling coil, water valve, air valve, air supply fan, return fan, and various sensors, and is dedicated to bringing the conditioned space temperature to a desired set point.

In this study, a variable air volume (VAV) based HVAC system for a multi-zone building is considered to demonstrate the effectiveness of the proposed method. In this system, the temperature of each zone is controlled using a proportional integral derivative (PID) controller by changing the inlet air flow rate of the corresponding zone while keeping the outlet air temperature of the cooling coil unchanged. Meanwhile, a PID controller controls the inlet water flow rate to the cooling coil.

The building under study, shown in Fig. 1, is a single-floor, 4-zone office, assumed to be located in Doha city in Qatar, with a total floor area of 168 m^2, which has four identical rooms with a ground area of 42 m^2 and a volume of 147 m^3. Each zone has two windows on its external walls, with a 4 m^2 area. The system is simulated using the TRNSYS program, a flexible graphically based software widely accepted in the design and optimization of energy systems recently [8,13,15,23,24]. The accuracy and fidelity of TRNSYS models can be attributed to the modules' development by an authoritative department, the thermal energy systems specialists of the United States, and the software's adoption of component object method (COM) technology, which allows it to replicate the HVAC system to a large extent [7].

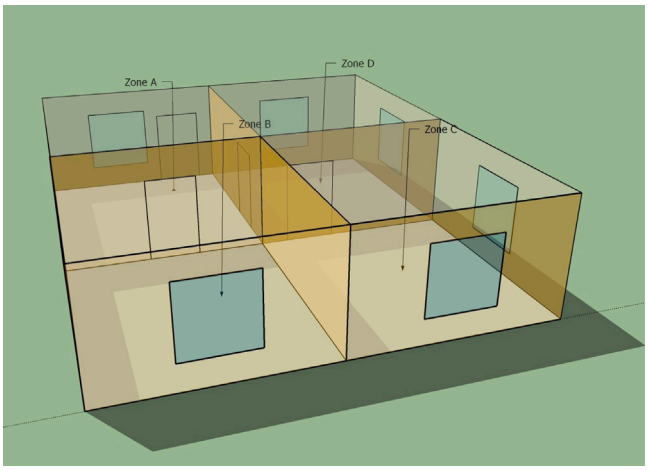

Fig. 1. Sketch of the simulated 4-zone building.

The office is equipped with a VAV-based HVAC system for cooling application. The four zones are supplied from the AHU with the cold air of temperature 16 °C and variable flow rate controlled by the VAV box terminal. The inlet cold water into the cooling coil is considered to have a constant temperature of 10 °C. The water flow rate is controlled using the PID controller, which regulates the water open position valve. Six sensors are considered in the system to measure the temperature of the zones denoted by T_{A}, T_{B}, T_{C}, and T_{D}, the outlet air temperature of the AHU T_{O}, and the ambient temperature T_{amb}.

3. Preliminaries and problem formulation

This section presents a brief overview of the Koopman operator theory and how it can be used for bilinear model realization. Further, four typical types of sensor faults in the HVAC system are described and modeled.

3.1. Overview of koopman operator theory

In 1931, B. O. Koopman demonstrated the existence of an operator with an infinite dimension capable of describing the evolution of states of a nonlinear system in an infinite-dimensional space [39]. Consider the nonlinear dynamical system governed by the following differential equation

$$\mathbf{x}(k+1) = \mathbf{F}(\mathbf{x}(k), \mathbf{u}(k)), \tag{1}$$

where $\mathbf{x}(k) = [x_1(k), \dots, x_n(k)]^{\mathrm{T}} \in \mathcal{X} \subset \mathbb{R}^n$ is the system state, $\mathbf{u}(k) = [u_1(k), \dots, u_m(k)]^{\mathrm{T}} \in \mathcal{U} \subset \mathbb{R}^m$ is the control input, $\mathcal{X}$, $\mathcal{U}$ are compact subsets, and $\mathbf{F} : \mathcal{X} \times \mathcal{U} \to \mathcal{X}$ is a continuously differentiable function that defines the flow map of the dynamical system.

Let $\mathcal{F}$ be the infinite-dimensional function space with compact domain $\mathcal{X} \times \mathcal{U} \subset \mathbb{R}^{n \times m}$, spanned by all the observables $\boldsymbol{\Phi} : \mathbb{R}^{n \times m} \to \mathbb{R}^N$, where $\boldsymbol{\Phi} = [\phi_1, \dots, \phi_N]^{\mathrm{T}}$, $N > n \times m$, and $N \in \mathbb{N} \cup \infty$, then the Koopman operator $\boldsymbol{\mathcal{K}} : \mathcal{F} \to \mathcal{F}$ is a linear infinite operator that can act on functions $\boldsymbol{\Phi} \in \mathcal{F}$ and is defined as follows

$$\boldsymbol{\mathcal{K}}\boldsymbol{\Phi}(\mathbf{x}(k), \mathbf{u}(k)) \triangleq \boldsymbol{\Phi}(\mathbf{F}(\mathbf{x}(k), \mathbf{u}(k))). \tag{2}$$

Since this operator is infinite-dimensional, using the extended dynamic mode decomposition (EDMD) [40,41], a finite-dimensional matrix approximation of the Koopman operator can be obtained via a linear regression applied to the observed data. To this end, a countable set of P linearly independent observables $\{\phi_{\mathrm{xu},i} : \mathcal{X} \times \mathcal{U} \to \mathbb{R}\}_{i=1}^{P}$ should be chosen and used to construct the lifting function as follows

$$\boldsymbol{\Phi}_{\mathrm{xu}}(\mathbf{x}, \mathbf{u}) = [\phi_{\mathrm{xu},1}(\mathbf{x}, \mathbf{u}), \phi_{\mathrm{xu},2}(\mathbf{x}, \mathbf{u}), \dots, \phi_{\mathrm{xu},P}(\mathbf{x}, \mathbf{u})]^{\mathrm{T}}. \tag{3}$$

In order to obtain the Koopman approximation matrix form, n_{s} discrete measurements in the form of $\{(\mathbf{u}(k), \mathbf{x}(k), \mathbf{x}(k+1))\}_{k=1}^{n_{\mathrm{s}}}$ is collected satisfying (1). These measurements are also called snapshots. The best matrix approximation of Koopman operator, denoted by $\tilde{\boldsymbol{\mathcal{K}}}$, can be expressed as a solution to the following least squares problem

$$\min_{\tilde{\boldsymbol{\mathcal{K}}}} \{ \sum_{k=1}^{n_{\mathrm{s}}} \| \tilde{\boldsymbol{\mathcal{K}}} \boldsymbol{\Phi}_{\mathrm{xu}}(\mathbf{x}(k), \mathbf{u}(k)) - \boldsymbol{\Phi}_{\mathrm{xu}}(\mathbf{x}(k+1), \mathbf{u}(k+1)) \|_{\mathrm{F}}^2 \}, \tag{4}$$

where $\| \cdot \|_{\mathrm{F}}$ denotes the Frobenius norm.

3.2. Model realization using koopman operator

A model realization of (1) is a dynamical system that realizes the state $\mathbf{x}$ that (1) specifies under any input signal $\mathbf{u}$; hence the Koopman operator can be used to establish a model realization [42]. To this end, a countable set of M observables $\{\psi_i \in \mathcal{F}\}_{i=1}^{M}$ is considered, from which the original state can be obtained through an inverse mapping $C : \mathbb{R}^M \to \mathbb{R}^n$. Various model realizations can be obtained for a dynamical system, including linear, bilinear, and nonlinear model realizations. These realizations have a linear behavior concerning observables. However, they do not necessarily have a linear behavior with respect to the original state and input. The type of approximated Koopman model depends on the choice of observables.

3.2.1. Linear model realization

The linear model can be realized using a suitable choice of observables $\{\psi_i\}_{i=1}^{N+m}$ constructing the lifting function as

$$\boldsymbol{\Psi} = \begin{bmatrix} \boldsymbol{\Psi}_{\mathrm{x}}{}^{\mathrm{T}} & \boldsymbol{\Psi}_{\mathrm{u}}{}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}, \tag{5}$$

where each component of $\boldsymbol{\Psi}_{\mathrm{x}} \triangleq \{\psi_{\mathrm{x},i}(\mathbf{x},\mathbf{u}_1) = \psi_{\mathrm{x},i}(\mathbf{x},\mathbf{u}_2)\}_{i=1}^{N}, \forall \mathbf{u}_1, \mathbf{u}_2 \in \mathbb{R}^m$ depends on the state only, and each component of $\boldsymbol{\Psi}_{\mathrm{u}} \triangleq \{\psi_{\mathrm{u},i}(\mathbf{x},\mathbf{u}) = u_i\}_{i=1}^{m}$, projects onto components of the input, i.e., $\boldsymbol{\Psi}_{\mathrm{u}} = \mathbf{u}$.

By collecting the snapshots from the input-state of the dynamics (1), and obtaining the lifted states using the suitable lifting function, the Koopman operator can be approximated using EDMD. The linear realization matrices are embedded within the first N rows of the approximated matrix as follows

$$\tilde{\boldsymbol{\mathcal{K}}} = \begin{bmatrix} A & B \\ \times & \times \\ \vdots & \vdots \\ \times & \times \end{bmatrix}, \tag{6}$$

where $\tilde{\boldsymbol{\mathcal{K}}} \in \mathbb{R}^{(N+m)\times(N+m)}$, $A \in \mathbb{R}^{N\times N}$, $B \in \mathbb{R}^{N\times m}$, and the linear realization can be represented as

$$\boldsymbol{\Psi}_{\mathrm{x}}(k+1) = A\boldsymbol{\Psi}_{\mathrm{x}}(k) + B\boldsymbol{\Psi}_{\mathrm{u}}(k), \tag{7}$$

where $\boldsymbol{\Psi}_{\mathrm{x}}(k)$ and $\boldsymbol{\Psi}_{\mathrm{u}}(k)$ stand for $\boldsymbol{\Psi}_{\mathrm{x}}(\mathbf{x}(k),\mathbf{u}(k))$ and $\boldsymbol{\Psi}_{\mathrm{u}}(\mathbf{x}(k),\mathbf{u}(k))$ in shorthand, respectively.

3.2.2. Bilinear model realization

Bilinear model realizations retain some of the computing advantages of linear Koopman models while being more probable to exist for arbitrary dynamical systems, and they might also be able to provide a more precise model with a lower dimensionality in comparison with linear models. The appropriate observables $\{\psi_i\}_{i=1}^{N(m+1)+m}$ that are required to generate a bilinear model realization constituting the lifting function as

$$\boldsymbol{\Psi} = \begin{bmatrix} \boldsymbol{\Psi}_{\mathrm{x}}^{\mathrm{T}} & \boldsymbol{\Psi}_{\mathrm{xu}}^{\mathrm{T}} & \boldsymbol{\Psi}_{\mathrm{u}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}, \tag{8}$$

where $\boldsymbol{\Psi}_{\mathrm{x}}$ and $\boldsymbol{\Psi}_{\mathrm{u}}$ can be obtained as suitable observables for linear model realization, and each component of $\boldsymbol{\Psi}_{\mathrm{xu}} \triangleq \boldsymbol{\Psi}_{\mathrm{u}} \otimes \boldsymbol{\Psi}_{\mathrm{x}}$ depends on both state and input. The symbol $\otimes$ denotes the Kronecker product.

Now, having collected the snapshots from the input-state of the dynamics (1), and obtaining the lifted states using the suitable lifting function, the Koopman operator can be approximated using EDMD. The bilinear realization coefficients are embedded within the first N rows of the approximated matrix as follows

$$\tilde{\boldsymbol{\mathcal{K}}} = \begin{bmatrix} A & H_1 & \cdots & H_m & B \\ \times & \times & \cdots & \times & \times \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \times & \times & \cdots & \times & \times \end{bmatrix}, \tag{9}$$

where $\tilde{\boldsymbol{\mathcal{K}}} \in \mathbb{R}^{(N(m+1)+m)\times(N(m+1)+m)}$, $A \in \mathbb{R}^{N\times N}$, $B \in \mathbb{R}^{N\times m}$, $H_i \in \mathbb{R}^{N\times N}, i = 1,\ldots,m$, and the bilinear realization can be represented as

$$\boldsymbol{\Psi}_{\mathrm{x}}(k+1) = A\boldsymbol{\Psi}_{\mathrm{x}}(k) + B\boldsymbol{\Psi}_{\mathrm{u}}(k) + H\boldsymbol{\Psi}_{\mathrm{xu}}(k), \tag{10}$$

where $H = [H_1,\ldots,H_m] \in \mathbb{R}^{N\times Nm}$, and $\boldsymbol{\Psi}_{\mathrm{x}}(k)$, $\boldsymbol{\Psi}_{\mathrm{u}}(k)$, and $\boldsymbol{\Psi}_{\mathrm{xu}}(k)$ stand for $\boldsymbol{\Psi}_{\mathrm{x}}(\mathbf{x}(k),\mathbf{u}(k))$, $\boldsymbol{\Psi}_{\mathrm{u}}(\mathbf{x}(k),\mathbf{u}(k))$, $\boldsymbol{\Psi}_{\mathrm{xu}}(\mathbf{x}(k),\mathbf{u}(k))$ in shorthand. For convenience, in both linear and bilinear model realizations, the first n observables can be considered as the original state, i.e.

$$\psi_i \triangleq x_i, \quad \text{for} \quad i = 1,\ldots,n, \tag{11}$$

so the inverse mapping from the lifted space to the original space can be obtained easily as

$$\hat{\mathbf{x}}(k) = C\boldsymbol{\Psi}_{\mathrm{x}}(k), \tag{12}$$

where $C = \begin{bmatrix} I_n & 0_{n\times(N-n)} \end{bmatrix}$, $\hat{\mathbf{x}}(k)$ is the approximation of the original states, I_n denotes an n-dimensional identity matrix, and $0_{n\times N}$ is a zero matrix of dimension $n \times (N-n)$.

3.3. Sensor fault models

The performance of the proposed SFDI method is verified considering all sensor fault types, i.e., bias, drift, complete failure, precision degradation, and LOE with different severity levels and they are defined as follows.

3.3.1. Bias

The bias sensor fault is occurred when there is a constant bias between the faulty sensor output $y_{\mathrm{f}}(k)$ and the true sensor value $y_{\mathrm{h}}(k)$. This fault can be defined as

$$y_{\mathrm{f}}(k) = y_{\mathrm{h}}(k) \pm f_{\mathrm{b}}, \quad k > k_{\mathrm{f}}, \tag{13}$$

where f_{b} is the bias value and k_{f} denotes the fault occurrence time.

3.3.2. Drift

The drift sensor fault leads to a gradually increasing bias between the faulty sensor output and the actual value. Thus, it can be described as

$$y_{\mathrm{f}}(k) = y_{\mathrm{h}}(k) \pm f_{\mathrm{d}}k, \quad k > k_{\mathrm{f}}, \tag{14}$$

where f_{d} denotes the drift gain.

3.3.3. Precision degradation

Under precision degradation fault, there is a dynamic change in the faulty sensor output, while its average remains the same. The fault can be expressed as

$$y_{\mathrm{f}}(k) = y_{\mathrm{h}}(k) + f_{\mathrm{p}}(k), \quad k > k_{\mathrm{f}}, \tag{15}$$

where $f_{\mathrm{p}}(k)$ is a zero mean while noise.

3.3.4. Complete failure

This kind of fault freezes the sensor output, and can be defined as

$$y_{\mathrm{f}}(k) = f_{\mathrm{f}}, \quad k > k_{\mathrm{f}}, \tag{16}$$

where f_{f} is a constant value.

3.3.5. Loss of effectiveness (LOE)

This kind of fault leads to a decreased sensor efficiency and can be defined as

$$y_{\mathrm{f}}(k) = f_{\mathrm{m}}y_{\mathrm{h}}(k), \quad k > k_{\mathrm{f}}, \tag{17}$$

where $0 \leq f_{\mathrm{m}} < 1$.

3.4. Problem formulation

Sensor faults can be categorized into additive faults such as bias, drift, and precision degradation, and multiplicative faults such as LOE, and complete failure. Consider a nonlinear discrete-time that allows for external control inputs as

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{F}(\mathbf{x}(k), \mathbf{u}(k)), \\ \mathbf{y}(k) &= \mathbf{G}(\mathbf{x}(k), \mathbf{u}(k)), \end{aligned} \tag{18}$$

where $\mathbf{x}(k) \in \mathcal{X} \subset \mathbb{R}^n$ is the system state, $\mathbf{u}(k) \in \mathcal{U} \subset \mathbb{R}^m$ is the control input, and $\mathbf{y}(k) \subset \mathbb{R}^q$ is the system output. The faults corrupted the output of the system can be described as

$$\tilde{\mathbf{y}}(k) = \tilde{A}_{\mathrm{f}}(k)\mathbf{y}(k) + \tilde{\mathbf{B}}_{\mathrm{f}}(k), \tag{19}$$

where $\tilde{A}_{\mathrm{f}}(k) \triangleq \mathrm{diag}\{\alpha_{\mathrm{f},1}(k), \alpha_{\mathrm{f},2}(k), \ldots, \alpha_{\mathrm{f},q}(k)\}$, $0 \leq \alpha_{\mathrm{f},i}(k) \leq 1$, $i = 1, \ldots, q$, and $\tilde{\mathbf{B}}_{\mathrm{f}}(k) \triangleq [\beta_{\mathrm{f},1}(k), \ldots, \beta_{\mathrm{f},q}(k)]^{\mathrm{T}}$ are unknown arbitrary function of time. With this formulation, all types of sensor faults can be modeled. Additive sensor faults such as bias, drift, and complete failure can be modeled through $\tilde{\mathbf{B}}_{\mathrm{f}}(k)$ and multiplicative sensor faults such as LOE and calibration errors can be modeled through $\tilde{A}_{\mathrm{f}}(k)$. By integrating (10), (12) and (19), the Koopman bilinear model realization after the fault occurrence can be written as

$$\begin{aligned} \boldsymbol{\Psi}_{\mathrm{x}}(k+1) &= A\boldsymbol{\Psi}_{\mathrm{x}}(k) + B\boldsymbol{\Psi}_{\mathrm{u}}(k) + H\boldsymbol{\Psi}_{\mathrm{xu}}(k), \\ \hat{\mathbf{x}}_{\mathrm{f}}(k) &= C\boldsymbol{\Psi}_{\mathrm{x}}(k) + Q\mathbf{f}_{\mathrm{s}}(k), \end{aligned} \tag{20}$$

where $\hat{\mathbf{x}}_{\mathrm{f}}(k)$ denotes the faulty measurements of the system, $\mathbf{f}_{\mathrm{s}}(k) \triangleq (\tilde{A}_{\mathrm{f}}(k) - I)C\boldsymbol{\Psi}_{\mathrm{x}}(k) + \tilde{\mathbf{B}}_{\mathrm{f}}(k)$, and $Q = I_q$. A, B, C, and H are the system constant matrices of compatible sizes. The problem is now to design a bank of residuals for detecting and isolating each sensor fault. This is mainly achieved by having each residual to be insensitive to its corresponding fault and sensitive to the other faults.

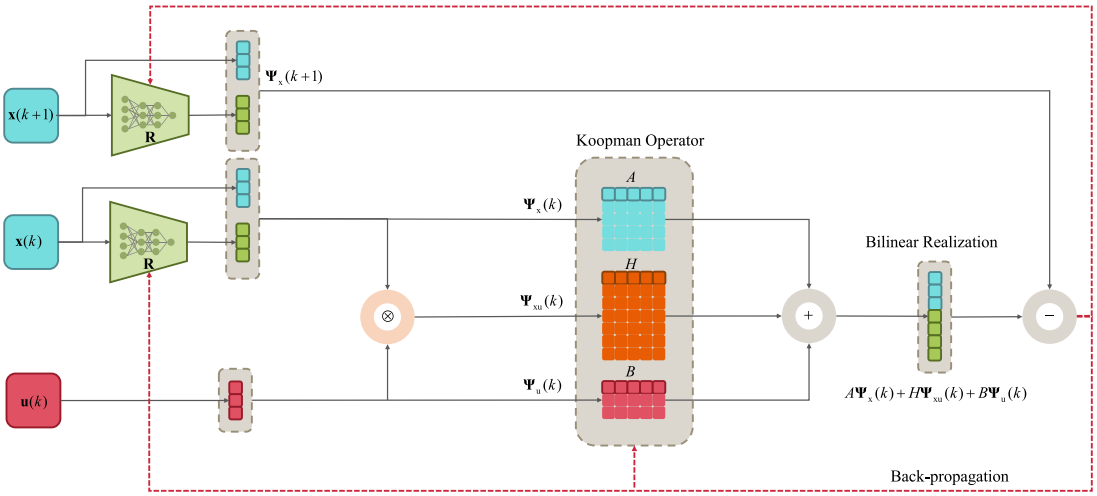

Fig. 2. The deep neural-network architecture used for Koopman bilinear model realization. The lifting encoder **R** and the bilinear model realization A, B, and H are learned simultaneously using the proposed architecture. Both of the lifting encoders in this architecture are the same.

4. Methodology

In this section, the proposed SFDI method for the multi-zone HVAC system is introduced. First, the deep Koopman bilinear model realization scheme is described, and then the parity-space is calculated using the realized model matrices. Next, the detection and isolation residuals are generated accordingly.

4.1. Bilinear deep koopman model realization

As discussed in the previous section, using EDMD to provide a closed-form approximation of the Koopman operator requires a suitable choice of observables to establish an intrinsic coordinate that stays invariant under the operator $\tilde{\mathcal{K}}$. In practice, the choice of observables is non-systematic, time-demanding, and requires a great deal of trials and errors for high-dimensional and complex dynamics [43]. The overarching purpose of this section is to leverage the capability of deep neural networks to simultaneously discover and generate an appropriate Koopman basis along with the bilinear model realization of the form (10). Further, learning the basis functions could often lead to lower-dimensional model realization and no knowledge of the underlying dynamics is required.

The schematic of the proposed deep neural network is demonstrated in Fig. 2. This structure consists of a linear layer that forms the Koopman operator, and a deep section so-called lifting encoder that lifts the states to a new basis, and they can be trained synchronously. The observable functions are parameterized using the encoder **R**, and the original states of the system are augmented to the output of the encoder **R** to compose the state-dependent observables $\boldsymbol{\Psi}_x(k;\theta) \in \mathbb{R}^N$, where θ denotes the trainable parameters of the network, i.e., weights and biases. The Kronecker product of the state-dependent observables and the input components is obtained to perform the $\boldsymbol{\Psi}_{xu}(k;\theta) \in \mathbb{R}^{Nm}$, where $N = n + n_{\mathrm{d}}$, and n_{d} denotes the number of neurons in the output layer of the encoder **R**. The bilinear evolution of the system states is guaranteed by concatenation of them to the observables. Therefore, a decoder is not required for the inverse mapping from the lifted states to the original ones, since the computation of matrix C is trivial as mentioned in the previous section, given by (12).

As shown in Fig. 2, the Koopman operator $\tilde{\mathcal{K}}$ is decomposed to the bilinear realization matrices A, B, and H. The activations $\bar{\mathbf{y}}_l \in \mathbb{R}^{n_l}$ of the deep part of the network at any hidden processing layer l can be expressed as

$$\bar{\mathbf{y}}_l = \vartheta(W_l \bar{\mathbf{x}} + b_l), \quad l = 1, \ldots, L, \tag{21}$$

where $\bar{\mathbf{x}} \in \mathbb{R}^{n_{l-1}}$ is the input to layer, ϑ is the activation function, $W_l \in \mathbb{R}^{n_l \times n_{l-1}}$ is the weights matrix of the layer l, $b_l \in \mathbb{R}^{n_l}$ is the bias, n_l is the width of the hidden layer, and L denotes the number of hidden layers. The bilinear model realization can be obtained by minimizing the following cost function

$$\min_{A,B,H,\theta} \|\boldsymbol{\Psi}_{\mathrm{x}}(k+1;\theta) - A\boldsymbol{\Psi}_{\mathrm{x}}(k;\theta) - H\boldsymbol{\Psi}_{\mathrm{xu}}(k;\theta) - B\boldsymbol{\Psi}_{\mathrm{u}}(k)\|_{\mathrm{F}}. \tag{22}$$

Due to the non-convex characteristic of (22), the dataset is partitioned into two parts, and the training procedure is assumed to have three phases that are performed iteratively. The training manner is illustrated in Fig. 3, and one can refer to [37,38] for more details. Next, a model selection based on a long term prediction is carried out and the model realization with the best prediction performance is chosen, and then used for the FDI scheme in the next section. Indeed, the realized model not only exhibits the same input–output behavior as the actual dynamics but also provides an analytical bilinear structure, known as a deep bilinear Koopman model, for the underlying dynamics, and therefore, enables the application of structured fault diagnosis methods such as parity-space.

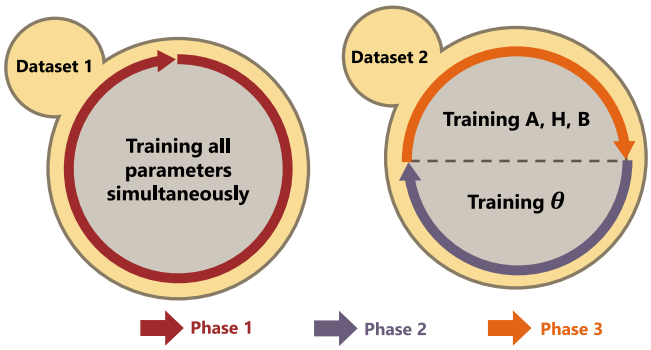

Fig. 3. Deep Koopman training procedure. The training is performed in 3 phases.

4.2. Model realization performance metrics

The root mean square error (RMSE) and Taylor diagram are determined to evaluate the reliability of the realized model. The RMSE formula is as follows

$$\text{RMSE} = 100\frac{\sqrt{\sum_k \|\hat{\mathbf{x}}(k) - \mathbf{x}(k)\|_2^2}}{\sqrt{\sum_k \|\mathbf{x}(k)\|_2^2}}, \tag{23}$$

where $\hat{\mathbf{x}}(k)$ denotes the predicted state and $\mathbf{x}(k)$ is the true measurements of the system state at step k.

The Taylor diagram is one of the most effective methods for graphically evaluating the accuracy of prediction models [44]. This method can assess models' accuracy by their standard deviations, correlation coefficients, and centered RMSE values (i.e., RMSE for zero mean data points). These statistics are computed for the real data and the models' predictions, and a point is assigned to each one of the models and the real data. Each point's radial distance from the origin determines the standard deviation, its azimuthal position represents the correlation coefficient, and its distance from the reference point indicates the centered RMSE. The optimal model is represented by the point closest to the real data point.

4.3. Bilinear parity-space SFDI

The central principle of the parity space method is to check the consistency of the parity relations by using the measurement of outputs and control inputs collected over a finite window. A fault is declared to have occurred once the balance of the parity equations is corrupted, and the resulting error exceeds the preassigned threshold [45]. In this section, the extension of the parity-space method for bilinear systems [46] is used to generate the residuals corresponding to the sensor fault of the system. To this end, the bilinear model realization obtained in the previous section is utilized to obtain the parity equations and construct the parity-space.

The realized bilinear model with the sensor fault in output is represented as follows

$$\begin{aligned} \boldsymbol{\Psi}_{\mathrm{x}}(k+1) &= A\boldsymbol{\Psi}_{\mathrm{x}}(k) + B\mathbf{u}(k) + \sum_{i=1}^{m} H_i \boldsymbol{\Psi}_{\mathrm{x}}(k)u_i(k), \\ \hat{\mathbf{x}}(k) &= C\boldsymbol{\Psi}_{\mathrm{x}}(k) + Q\mathbf{f}_{\mathrm{s}}(k), \end{aligned} \tag{24}$$

where $\boldsymbol{\Psi}_{\mathrm{x}}(k) \in \mathbb{R}^N$, $\hat{\mathbf{x}}(k) \in \mathbb{R}^n$, $\mathbf{u}(k) \in \mathbb{R}^m$, and $\mathbf{f}_{\mathrm{s}}(k) = [f_{\mathrm{s},2}, \ldots, f_{\mathrm{s},q}]^{\mathrm{T}} \in \mathbb{R}^q$ denotes the sensor fault. A, B, C, Q, and $H_i, i = 1, \ldots, m$ are the system constant matrices of compatible sizes.

By including the bilinear terms of the model (24) into the system matrix, the bilinear system can be represented in a time-varying linear system form as follows

$$\begin{aligned} \boldsymbol{\Psi}_{\mathrm{x}}(k+1) &= \mathcal{A}(k)\boldsymbol{\Psi}_{\mathrm{x}}(k) + B\mathbf{u}(k), \\ \hat{\mathbf{x}}(k) &= C\boldsymbol{\Psi}_{\mathrm{x}}(k) + Q\mathbf{f}_{\mathrm{s}}(k), \end{aligned} \tag{25}$$

where

$$\mathcal{A}(k) \triangleq A + \sum_{i=1}^{m} H_i u_i(k). \tag{26}$$

Let s is the parity-space order, the outputs over a finite window of $s+1$ sampling intervals can be expressed as

$$\mathbf{Y}(k) = \mathcal{M}(k)\boldsymbol{\Psi}_{\mathrm{x}}(k-s) + \mathcal{P}(k)\mathbf{U}(k) + \mathcal{Q}\mathbf{F}_{\mathrm{s}}(k), \tag{27}$$

where $\mathbf{Y}(k) = \left[\hat{\mathbf{x}}(k-s)^{\mathrm{T}}, \ldots, \hat{\mathbf{x}}(k)^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{n(s+1)}$, and $\mathbf{U}(k) = \left[\mathbf{u}(k-s)^{\mathrm{T}}, \ldots, \mathbf{u}(k)^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{m(s+1)}$, $\mathbf{F}_{\mathrm{s}}(k) = [\mathbf{f}_{\mathrm{s}}(k-s)^{\mathrm{T}}, \ldots, \mathbf{f}_{\mathrm{s}}(k)^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{q(s+1)}$, and $\mathcal{Q} = \mathrm{diag}\{Q, \ldots, Q\} \in \mathbb{R}^{n(s+1)\times q(s+1)}$. Also $\mathcal{M}(k) \in \mathbb{R}^{n(s+1)\times N}$ and $\mathcal{P}(k) \in \mathbb{R}^{n(s+1)\times m(s+1)}$ are time-varying matrices constructed using $\mathcal{A}(k)$, B, and C, as follows

$$\mathcal{M}(k) = \begin{bmatrix} C \\ C\mathcal{A}(k-s) \\ C\mathcal{A}(k-s+1)\mathcal{A}(k-s) \\ \vdots \\ C\prod_{i=1}^{s}\mathcal{A}(k-i) \end{bmatrix}, \tag{28}$$

$$\mathcal{P}(k) = \begin{bmatrix} 0_{n\times m} & 0 & 0 & \cdots & 0 \\ CB & 0_{n\times m} & 0 & \cdots & 0 \\ C\mathcal{A}(k-s+1)B & CB & 0_{n\times m} & & \\ \vdots & \vdots & & \ddots & \vdots \\ C\prod_{i=1}^{s-1}\mathcal{A}(k-i)B & C\prod_{i=2}^{s-1}\mathcal{A}(k-i)B & \cdots & CB & 0_{n\times m} \end{bmatrix}. \tag{29}$$

A parity-space $\mathcal{S}(k)$ can be defined as

$$\mathcal{S}(k) \triangleq \{\lambda(k) \mid \lambda(k)\mathcal{M}(k) = 0\}, \tag{30}$$

that is equivalent to obtaining the left null space of matrix $\mathcal{M}(k)$. Thus, the parity check $\boldsymbol{\gamma}(k)$ can be described as

$$\boldsymbol{\gamma}(k) = \lambda(k)(\mathbf{Y}(k) - \mathcal{P}(k)\mathbf{U}(k)), \tag{31}$$

satisfies

$$\boldsymbol{\gamma}(k) = \begin{cases} 0 & \text{if} \quad \mathbf{f}_{\mathrm{s}}(k) = 0, \quad \forall k, \\ \lambda(k)\mathcal{Q}\mathbf{F}_{\mathrm{s}}(k) & \text{if} \quad \mathbf{f}_{\mathrm{s}}(k) \neq 0, \quad \forall k. \end{cases} \tag{32}$$

Consequently, the parity check can be used for sensor fault detection, and the fault detection residual is defined as

$$\eta(k) \triangleq \|\boldsymbol{\gamma}(k)\|. \tag{33}$$

The parity order s should be chosen so that the parity vector $\lambda(k)$ exists. Hence, the inequality $sn > N - n$ should be satisfied [46].

Fault isolation is then implemented by generating q residuals for q sensors of the system, where each residual is insensitive to the corresponding sensor fault and sensitive to the other sensor faults. These residuals can be generated by processing the parity vector $\lambda(k)$. To this end, the fault distribution matrix $\mathcal{T}(k) = \lambda(k)\mathcal{Q}$ should be regrouped so that the effects of each fault at different sample times are relocated in one group. Considering $\lambda(k)$ has maximum linear independent rows, the parity check $\boldsymbol{\gamma}(k)$ can be reformulated as

$$\boldsymbol{\gamma}(k) = [\mathcal{R}_1(k), \ldots, \mathcal{R}_q(k)] \begin{bmatrix} \mathbf{f}_1(k) \\ \vdots \\ \mathbf{f}_q(k) \end{bmatrix}, \tag{34}$$

where $\mathcal{R}_i(k) = [\mathbf{t}_i(k), \mathbf{t}_{q+i}(k), \mathbf{t}_{2q+i}(k), \ldots, \mathbf{t}_{sq+i}(k)]$, $\mathbf{f}_i(k) = [f_{\mathrm{s},i}(k-s), \ldots, f_{\mathrm{s},i}(k)]^{\mathrm{T}}, \quad i = 1, \ldots, q$, $\mathbf{t}_i(k)$ is the ith column of matrix $\mathcal{T}(k)$, and $f_{\mathrm{s},i}(k)$ denotes the ith element of $\mathbf{f}_{\mathrm{s}}(k)$. The fault isolation vector is defined as

$$\boldsymbol{\varsigma}(k) \triangleq \delta(k)\boldsymbol{\gamma}(k), \tag{35}$$

where $\delta(k)$ denotes the isolation matrix and can be obtained such that

$$\delta(k)[\mathcal{R}_1(k), \ldots, \mathcal{R}_q(k)] = \mathcal{O}(k), \tag{36}$$

where

$$\mathcal{O}(k) = \begin{bmatrix} 0_{1\times s+1} & \times & \cdots & \times \\ \times & 0_{1\times s+1} & & \vdots \\ \vdots & & \ddots & \times \\ \times & \cdots & \times & 0_{1\times s+1} \end{bmatrix}. \tag{37}$$

Here $\times$ is the non-zero row vector of compatible dimension and $0_{1\times s+1}$ denotes the zero vector of dimension $s+1$. Combining (35)–(37) yields

$$\boldsymbol{\varsigma}(k) = \mathcal{O}(k) \begin{bmatrix} \mathbf{f}_1(k) \\ \vdots \\ \mathbf{f}_q(k) \end{bmatrix}. \tag{38}$$

Taking advantage of (38), the fault isolation vector $\boldsymbol{\varsigma}(k) = [\sigma_1(k), \ldots, \sigma_q(k)]^{\mathrm{T}}$ can be obtained in each sample k. Note that the fault in the ith sensor $\mathbf{f}_i(k), i = 1, \ldots, q$, modifies the $\boldsymbol{\varsigma}(k)$ such that the ith element of the isolation vector $\sigma_i(k), i = 1, \ldots, q$ is zero while the other components have non-zero values. Each entry of the fault isolation vector is referred to as fault isolation residuals in the following sections.

As discussed above, the detection and isolation residuals are designed using the bilinear Koopman approximation (25), and then the SFDI method is applied to the implemented HVAC system in TRNSYS. To this end, the outputs and the control inputs

Table 2
RMSE values of the realized model.

Metric	$\mathbf{T}_A$	$\mathbf{T}_B$	$\mathbf{T}_C$	$\mathbf{T}_D$	$\mathbf{T}_O$
RMSE	2.26%	2.21%	2.10%	2.06%	0.02%

of the system are collected and fault detection and isolation residuals are generated using the aforementioned equations in each sample time. To alleviate the computation demand, the time-dependent matrices $\mathcal{M}(k)$ and $\mathcal{P}(k)$ can be obtained using the recursive algorithm [46], which results in a faster residual computation at each sample time.

Since the accuracy of the approximated Koopman bilinear realization might suffer from approximation error, the residuals in the fault-free mode of the system are not exactly at zero. Therefore, to prevent false alarms, a predetermined threshold should be considered for fault detection and isolation. The threshold specifies the safe margin that the residuals must exceed to declare that fault happened. Once a fault occurs in the jth sensor ($j = 1, \dots, q$), the residuals indicate the fault as follows

$$\begin{cases} \eta(k) > \ell_D, & \\ \sigma_j(k) \le \ell_I & j = 1, \dots, q, \\ \sigma_i(k) > \ell_I & i = 1, \dots, q, i \neq j, \end{cases} \tag{39}$$

where ℓ_D and ℓ_I denote the detection and isolation thresholds respectively and q is the number of sensors. According to (39), when the jth sensor ($j = 1, \dots, q$) is subjected to a fault, the detection residual exceeds the detection threshold $\eta(k) > \ell_D$, the isolation residual corresponding to the faulty sensor remains below the isolation threshold $\sigma_j(k) < \ell_I$, and other isolation residuals overstep the isolation threshold $\sigma_i(k) > \ell_I$, $i = 1, \dots, q, i \neq j$.

Remark 1. It is essential to emphasize that, since the parity residual generators are analytically designed using the Koopman model matrices rather than being trained on a specific faulty dataset, there are no constraints imposed on the fault properties, such as fault type or amplitude, that might not be included in the dataset. Specifically, the design of the Koopman parity residual generators is based on the approximation of the Koopman predictor, which is obtained during the healthy operation of the dynamics. Therefore, no restrictions are imposed regarding the upper bound of the fault, as faults clearly adversely affect the input–output behavior of the dynamics, and the distorted data are not included in the parity space and will be indicated using generated residuals. However, the threshold is determined by applying the SFDI framework during the healthy operation of the system and is set sufficiently high so that the approximation error does not lead to having false alarms. Hence, as long as the input–output of the Koopman model has good accuracy, the fault diagnosis procedure is valid for the underlying nonlinear system and can isolate faults without triggering false alarms.

5. Multi-zone HVAC SFDI results

In this section, first, the network structure used for HVAC system is described. Next, the long term prediction of the proposed bilinear Koopman model realization is validated, and a comparison between the bilinear and linear Koopman model realization is presented. Further, the performance of the proposed data-driven SFDI method is demonstrated for all sensor fault types with various severity levels.

5.1. Multi-zone HVAC bilinear model realization

The bilinear model realization of the under-study 4-zone HVAC system is obtained using a 4-layer feed-forward neural network as the lifting encoder. The encoder has 2 hidden layers with 3 neurons followed by the output layer with one neuron. The input layer has 4 neurons, and the activation function of each layer is chosen to be rectified linear unit (ReLU). Adam optimizer is used for training the network parameters, and the training is carried out using PyTorch.

The data are collected from the simulation in TRNSYS for the normal operation modes of the HVAC system. The temperature set-points of the zones are considered to be 20 °C in the office working time and 25 °C when the office is closed. The main measurements of the system are as follows, the zones temperatures T_A, T_B, T_C, and T_D, the outlet air temperature of the AHU T_O, the ambient temperature T_{amb}, the zones inlet air control signals U_A, U_B, U_C, and U_D, and the inlet water to cooling coil control signal U_W. Each control signal determines the percentage of the open position of its connected valve. The training data are collected for 1 month of the normal system operation and sampled every 6 min in the summer season from June 1, at 00:00 to June 30, at 23:59. Normalization is performed on the collected data to enhance the training performance.

Since the output layer of the lifting encoder is chosen to have one neuron, and the under study four-zone HVAC system has five states, the obtained bilinear model realization will have an order of 6 and can provide us with the dynamic of the HVAC system cooling operation in hot months of the year (from June to August). The prediction validation of the obtained bilinear realization is demonstrated for the normal operation of the HVAC system on working days from July 17, at 08:00 to July 22, at 00:00 in Fig. 4. As shown in Fig. 4, this model has a promising ability to predict the future states of the implemented HVAC system (the temperature of each zone and the outlet air temperature of AHU) in TRNSYS. Further, the RMSE values of the zones temperatures and the outlet air temperature of the AHU is provided in Table 2.

Table 3
Prediction comparison based on RMSE between various Koopman model realizations for the HVAC system.

Predictor model	Method	N_{lift}/Encoder architecture	Dimension of the model realization	Lifting/Activation function	RMSE
Linear	EDMD	1	6	RBF	16.4%
Linear	EDMD	10	15	RBF	4.41%
Deep Linear	NNs	[5 1]	6	RBF	16.51%
Deep Linear	NNs	[5 4 3 3 1]	6	ReLU	67.5%
Deep Linear	NNs	[5 4 3 3 5]	10	ReLU	60.71%
Deep Bilinear	NNs	[5 4 3 3 1]	6	Sigmoid	12.58%
Deep Bilinear	NNs	[5 4 3 3 1]	6	Tanh	3.78%
Deep Bilinear	NNs	[5 4 3 3 1]	6	ReLU	2.26%

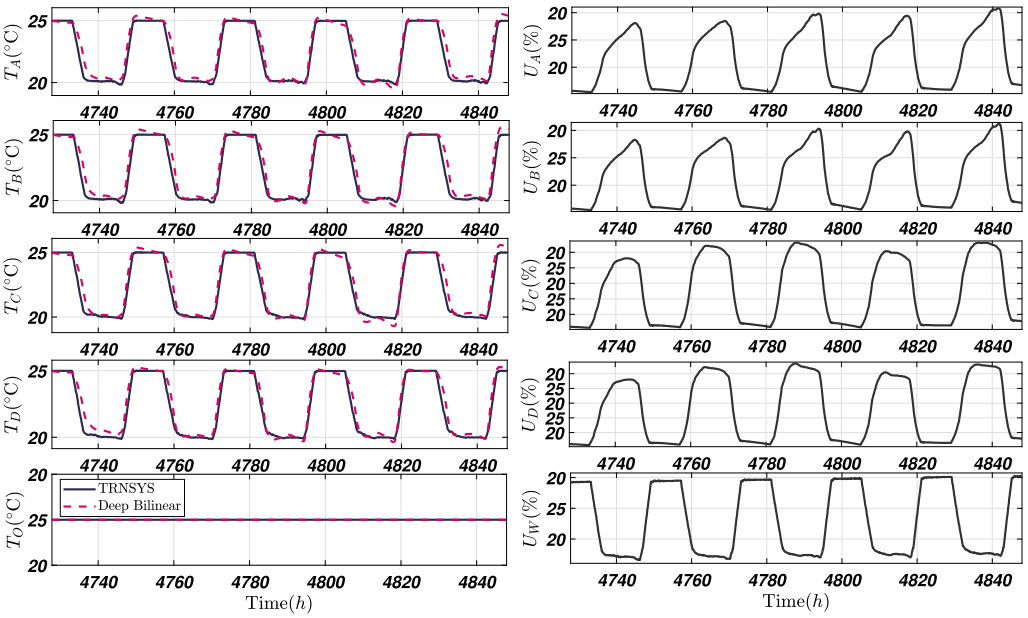

Fig. 4. Prediction validation of the bilinear model realization "Deep Bilinear" on working days July 17, at 00:00 to July 21, at 23:59, that is, from the 4728th hour to the 4848th hour in the whole year.

Remark 2. The realized model is trained on the generated data for one month (from June 1, at 00:00, to June 30, at 23:59); nevertheless, due to the global characteristics of the Koopman operator, the prediction performance of the realized model is valid for the entire hot months of the year, i.e., June, July, and August.

5.2. HVAC predictor comparison

In this section, a comparison is conducted between the bilinear and linear model realizations. The predictors are compared from two aspects, the capability for capturing the system dynamics and the dimension of the realized model. Realizing a high-dimensional model leads to more computation demand for the SFDI process. Therefore, we are interested in obtaining a reliable, precise, and concise model realization. The RMSE metric and the Taylor diagram are used to compare the prediction performance of models.

The linear model realization is performed using both EDMD algorithm with different number of observable functions N_{lift} and deep neural network framework used in [37], considering different architectures for the lifting encoder. The fixed observable functions for EDMD-based model realization are chosen to be the thin plate radial basis functions (RBF) with centers selected uniformly distributed on a unit box. Also, the deep bilinear Koopman model is trained with various activation functions and architectures and the best model is adopted. Fig. 5 illustrates the prediction comparison between various predictors with different dimensions. The test dataset is considered the same for all predictors, and each model's long-term prediction is assessed using RMSE index and the Taylor diagram. The characteristics of each predictor and the RMSE values are summarized in Table 3. Also, the Taylor diagram is illustrated in Fig. 6 for the best realized models using linear and bilinear Koopman model realizations. This is evident from the provided results that the deep Koopman bilinear model with ReLU activation function has the best long-term prediction, and

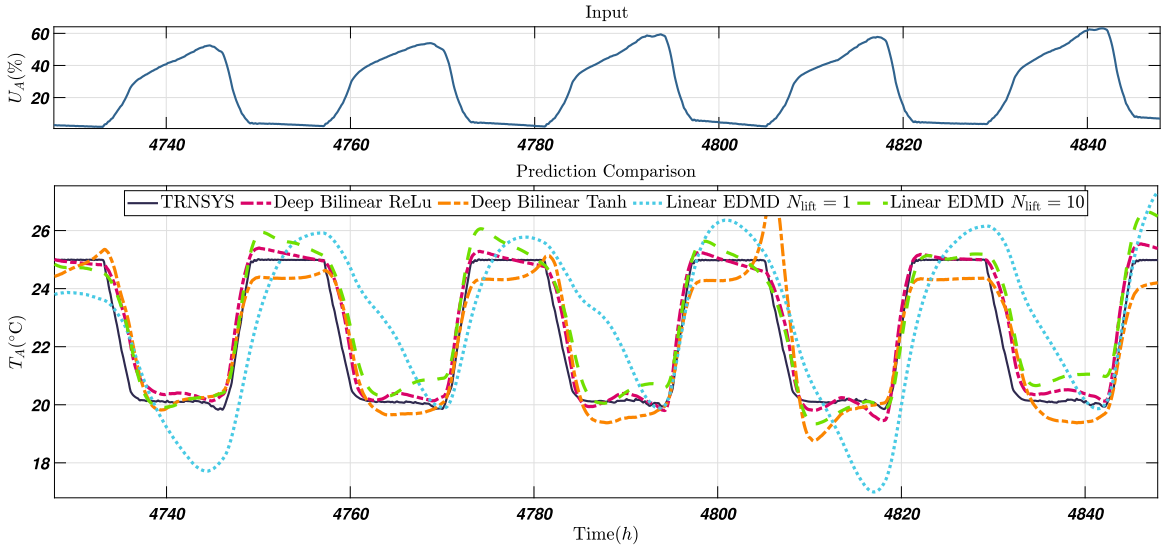

Fig. 5. Prediction comparison between various model realization using Koopman operator on working days July 17, at 00:00 to July 21, at 23:59, that is, from the 4728th hour to the 4848th hour in the whole year.

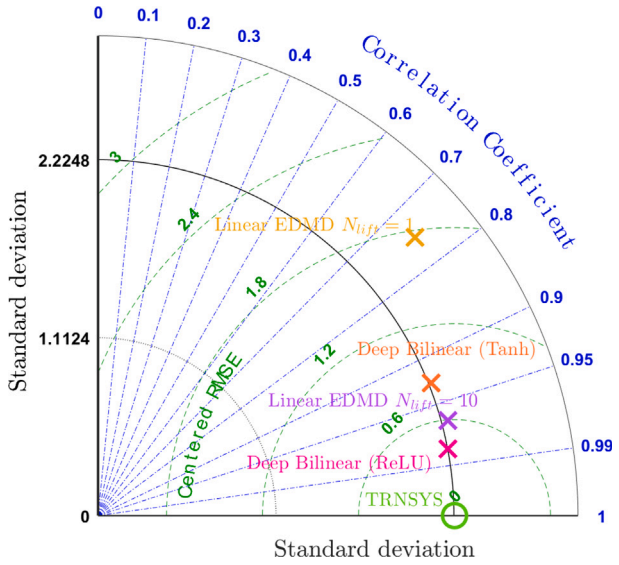

Fig. 6. Taylor diagram of realized models.

the adopted bilinear model outperforms the linear model realization in terms of long-term prediction with even lower dimension. The EDMD-based linear model realization can hardly achieve the same precision as the bilinear model, even when $N_{\text{lift}} = 10$ and the linear model has an order of 15. It should be noted that the prediction performance of the EDMD-based linear model does not improve by increasing the N_{lift} and we provided its best result here. As well, the deep linear model realization with one neuron in the output layer of the lifting encoder and the RBF activation function has about the same performance as the EDMD-based linear model and is not acceptable. Further, the deep linear model is compared with the bilinear one, using the same lifting encoder as the deep bilinear model. The RMSE values, in this case, reveal that the deep linear model failed to provide a proper long-term prediction. Considering these results, it can be concluded that the proposed bilinear model realization framework has the best precision and reliability with the minimum dimension.

5.3. Multi-zone HVAC sensor fault detection and isolation

The 4-zone HVAC system is equipped with a total of six sensors, four sensors for measuring the temperature of zones T_A, T_B, T_C, and T_D, and two for measuring the outlet air temperature of AHU T_O and the ambient temperature T_{amb}. Reliable measurements

Table 4
The summary of sensor fault occurrence scenarios.

Scenario	Sensor	Fault type	Severity (Fault magnitude)	Fault occurrence time	Fault duration	Detection time
Scenario. **1**	T_A	Bias	2 °C	July 18, at 04:00	09:00 h	Immediate
	T_B	Bias	4 °C	July 18, at 17:00	09:00 h	Immediate
	T_C	Bias	6 °C	July 19, at 06:00	09:00 h	Immediate
	T_D	Bias	8 °C	July 19, at 19:00	09:00 h	Immediate
	T_O	Bias	10 °C	July 20, at 08:00	09:00 h	Immediate
Scenario. **2**	T_A	Drift	0.4 °C/h	July 18, at 04:00	06:00 h	03:12 h
	T_B	Drift	0.8 °C/h	July 18, at 16:00	06:00 h	01:24 h
Scenario. **3**	T_D	Precision degradation	[−2 2] °C	July 18, at 04:00	06:00 h	Immediate
	T_O	Precision degradation	[−4 4] °C	July 18, at 16:00	06:00 h	Immediate
Scenario. **4**	T_A	Complete failure	-	July 18, at 05:00	05:00 h	00:48 h
	T_C	Complete failure	-	July 18, at 18:00	05:00 h	01:30 h
Scenario. **5**	T_A	LOE	90% efficiency	July 18, at 04:00	09:00 h	Immediate
	T_B	LOE	70% efficiency	July 18, at 17:00	09:00 h	Immediate
	T_C	LOE	50% efficiency	July 19, at 06:00	09:00 h	Immediate
	T_D	LOE	30% efficiency	July 19, at 19:00	09:00 h	Immediate

* The term "immediate" detection time means detecting a fault immediately one sample after its occurrence.

of each zone's temperature and outlet air temperature of the AHU are vital for the closed-loop control of the HVAC system. In this section, the performance of the proposed deep neural bilinear Koopman parity method is demonstrated through several sensor fault occurrence scenarios.

Taking into account the realized model utilizing a network with a single neuron in the output layer of the lifting encoder, we have $n = 5, N = 6$. By substituting these values in $sn > N - n$ the parity order is determined as $s \geq 1$. Therefore, the parity order is considered to be $s = 1$ resulting in low computation demand. Now, appropriate values should be obtained for the thresholds to ensure the best performance of the method and to prevent false alarms within the detectable fault range. The detection and isolation thresholds are assigned by executing the SFDI method for the healthy cooling operation of the HVAC system on different hot days and determining the false alarm rate considering different threshold values. Then, the diagnosable fault range is specified by injecting faults with various magnitudes and assessing the sensitivity of the residuals and the missed alarm rate. We are interested in choosing threshold values in a way that there are no false/missed alarms according to the detectable fault magnitude.

The detection and isolation thresholds are chosen to be $\ell_D = 1$, $\ell_I = 0.2$, respectively. Considering these values, there is no false alarm during the healthy operation of the system, and the residuals are sensitive to the faults with a magnitude $f > 1.5$ °C, which is acceptable in practice. This means that when the difference between the real temperature and the output of the faulty sensor is greater than 1.5 °C, the fault will be detected and isolated immediately. In contrast with some earlier methods, such as [28,29], there is no upper limit for the detectable range of the fault.

After designing the thresholds, several fault scenarios are considered to demonstrate the capability of the proposed method to detect and isolate all types of sensor faults, which are bias, drift, precision degradation, complete failure, and LOE. In addition, various aspects of the proposed SFDI approach, including detection time, sensitivity, reliability, and isolability are analyzed for each type of fault. As stated before, when the jth sensor ($j = 1, \ldots, q$) is subjected to a fault, the detection residual exceeds the detection threshold $\eta(k) > \ell_D$, the isolation residual corresponding to the faulty sensor remains below the isolation threshold $\sigma_j(k) < \ell_I$, and other isolation residuals overstep the isolation threshold $\sigma_i(k) > \ell_I$, $i = 1, \ldots, q, i \neq j$.

5.3.1. Additive faults

Three additive fault occurrence scenarios are considered to demonstrate the reliability of the proposed method to diagnose additive faults, which are bias, drift, and precision degradation. The summary of fault occurrence scenarios are provided in Table 4.

Scenario.* 1 *(bias): To demonstrate the significant capability of the proposed method to diagnose faults with a magnitude beyond 1.5 °C, this fault scenario, shown in Fig. 7, is performed. The bias fault in the whole five sensors of the system is injected according to (13), considering various severity levels. The faults occurred during different hours from July 18, at 00:00, to July 20, at 23:59. As can be seen, the detection time is fast, and faulty sensor is immediately detected and isolated. Also, when the system comes back to the normal conditions, the residuals fall under the threshold. As expected, by increasing the severity level of the fault, the proposed approach reveals a promising performance without any false alarms triggered.

Scenario.* 2 *(drift): In this scenario, as shown in Fig. 8, two drift faults are occurred in the sensors related to zones A and B on July 18. The first drift fault with the gain of 0.4 °C/h is started at 04:00 in the zone A sensor. The fault is detected and isolated once its magnitude reaches the diagnosable range. Thus the detection time is nearly 3 h in this case. This is because the abnormal effect caused by the drift (gradual fault) is too small to be detected at the beginning. Then, the zone B sensor is subjected to the drift fault with the gain of 0.8 °C/h at 16:00. The diagnosis time, in this case, is about 01:20 h.

Scenario.* 3 *(precision degradation): In this scenario, as shown in Fig. 9, the precision degradation fault raised in the sensors of zone D and AHU with the range of [−2 2] °C and [−4 4] °C, respectively. It can be seen that faulty sensors are detected and isolated immediately and accurately.

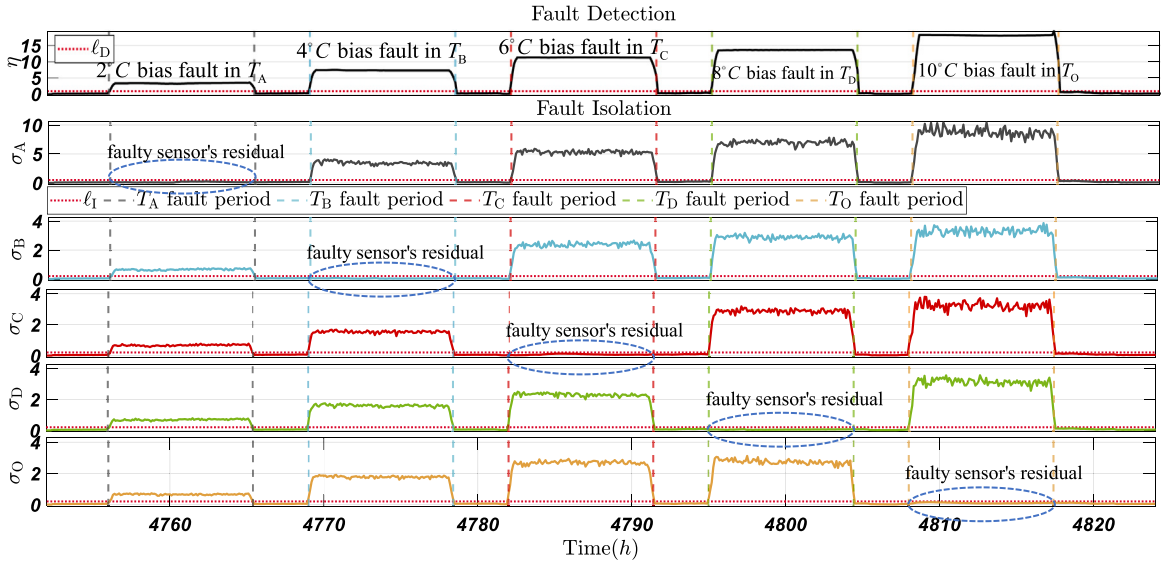

Fig. 7. ***Scenario.*** **1**: Bias faults with different ranges are occurred in five sensors during days July 18, 00:00 - July 20, 23:59, that is, from the 4752th hour to the 4824th hour in the whole year.

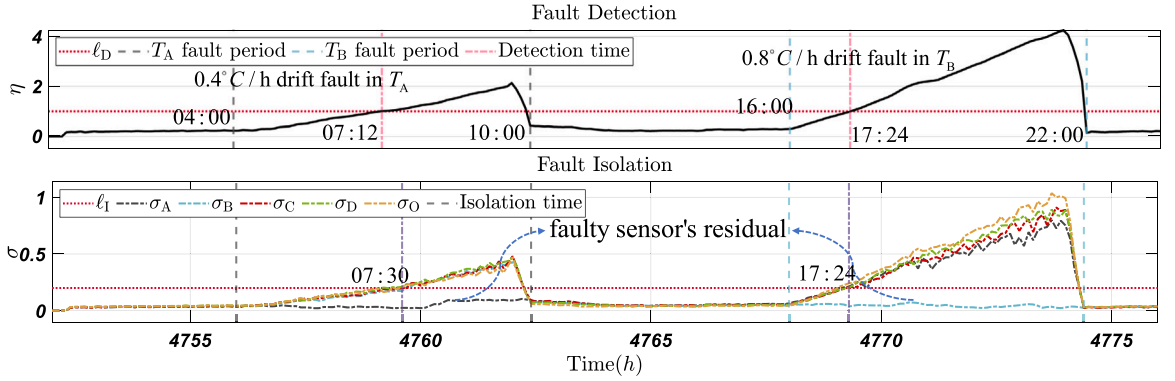

Fig. 8. ***Scenario.*** **2**: Drift faults with different gains are occurred in two sensors on July 18, from 00:00 to 23:59, that is, from the 4752th hour to the 4756th hour in the whole year.

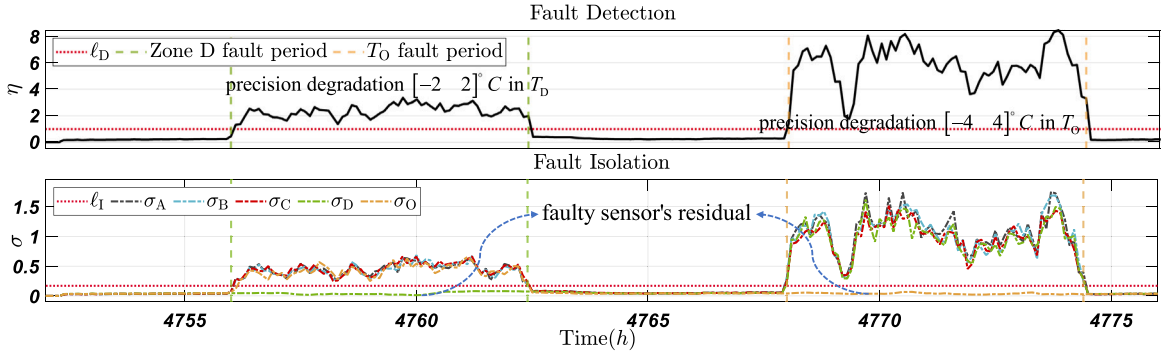

Fig. 9. ***Scenario.*** **3**: Precision degradation faults with different magnitudes are occurred in two sensors on July 18, from 00:00 to 23:59, that is, from the 4752th hour to the 4756th hour in the whole year.

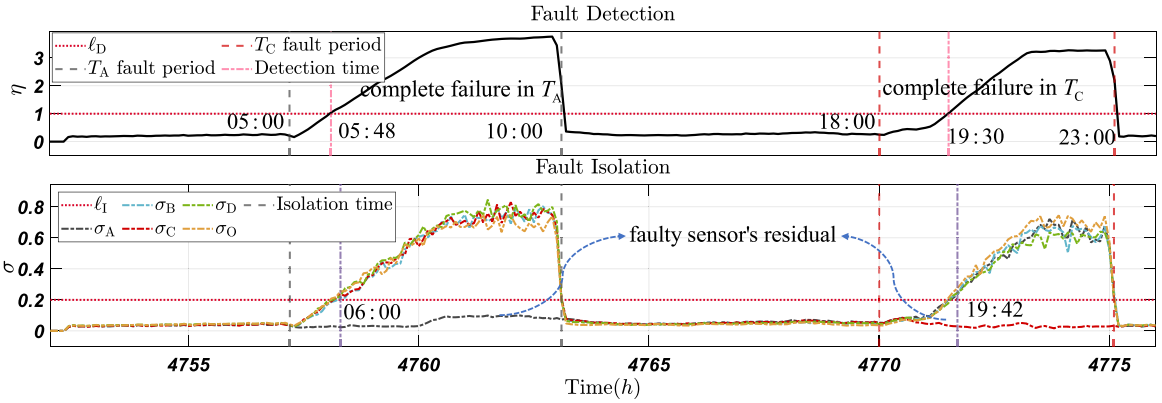

Fig. 10. ***Scenario.*** **4**: The temperature sensors of zone A and C are subjected to complete failure fault on July 18, from 00:00 to 23:59, that is, from the 4752th hour to the 4756th hour in the whole year.

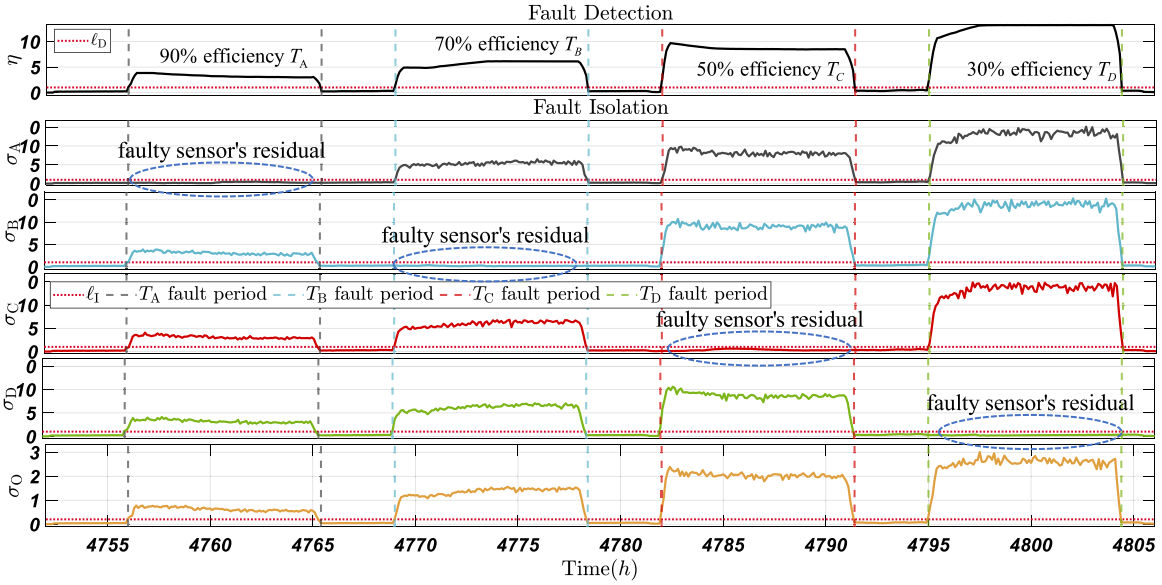

Fig. 11. ***Scenario.*** **5**: The efficiency of the sensors of the zones temperature are dropped down with various severity levels during days July 18, 00:00 - July 20, 06:00, that is, from the 4752th hour to the 4806th hour in the whole year.

5.3.2. Complete failure fault

Scenario.* 4 *(complete failure): In this scenario, as shown in Fig. 10, sensors related to zones A and C are subjected to a complete failure fault at 5:00 and 18:00 on July 18, respectively. As illustrated, it took the faults 00:48 and 1:30 h to be diagnosed after their occurrence. This is because the zone temperature changes slowly, and it takes time for the abnormal effect of the complete failure fault to exceed 1.5 °C.

5.3.3. Multiplicative fault

Scenario.* 5 *(LOE): In this scenario, as shown in Fig. 11, the efficiency of the sensors measuring the zones temperature are dropped with various severity levels from July 18, at 00:00, to July 20, at 06:00. As can be seen, the detection time is fast, and the faulty sensor is immediately detected and isolated. Also, when the system returns to normal conditions, the residuals fall under the threshold.

5.4. Discussion

As demonstrated in multiple fault scenarios, the proposed method effectively diagnosed all types of sensor faults. However, it should be noted that the SFDI performance of the proposed method is heavily dependent on the prediction performance of the realized model. The detectable fault range is also determined based on the model's accuracy. Therefore, the key factor to enhancing

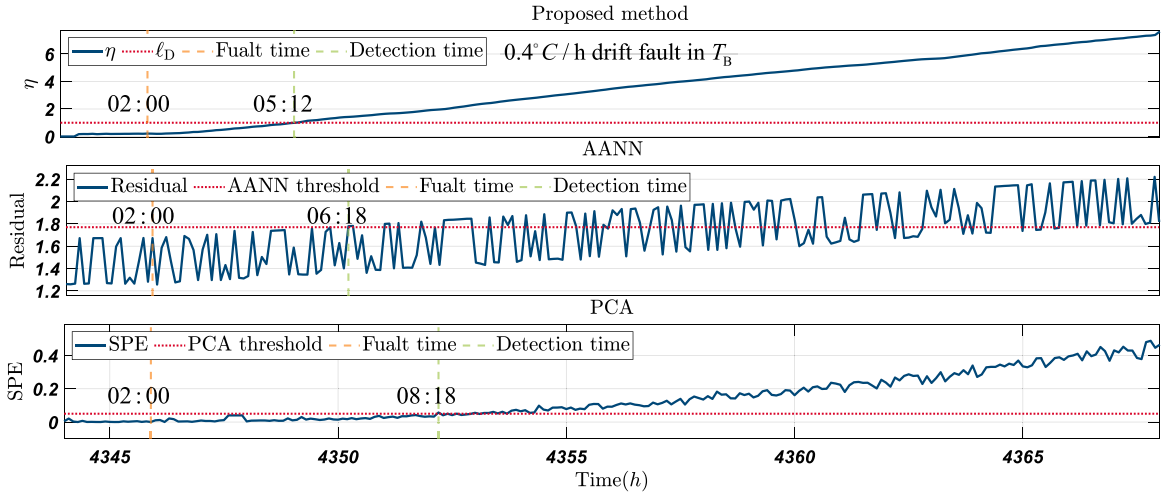

Fig. 12. The temperature sensor of zone B is subjected to drift fault on July 1, from 00:00 to 23:59, that is, from the 4344th hour to the 4368th hour in the whole year.

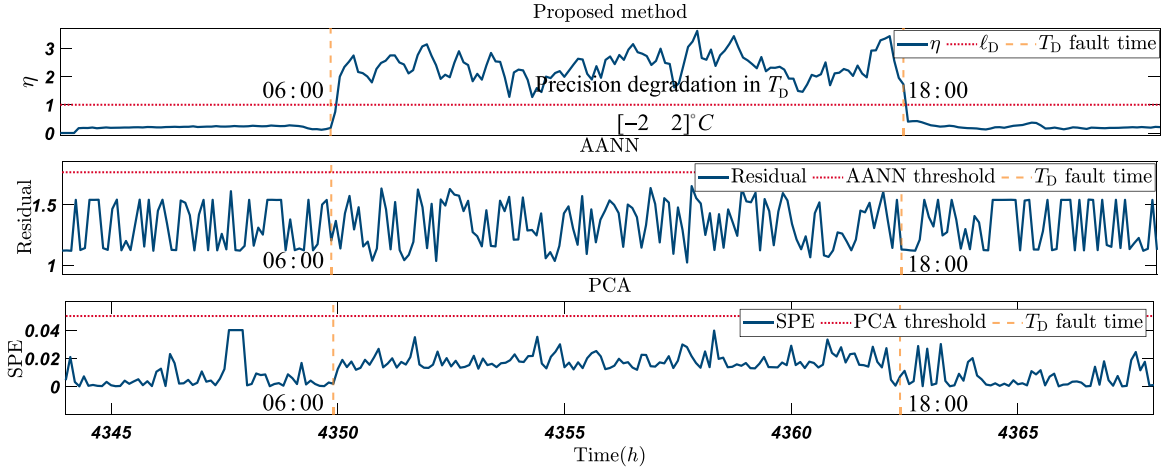

Fig. 13. The temperature sensor of zone D is subjected to precision degradation fault on July 1, from 00:00 to 23:59, that is, from the 4344th hour to the 4368th hour in the whole year.

Table 5
The summary of sensor fault occurrence scenarios for comparison.

Scenario	Method	Sensor	Fault type	Severity (Fault magnitude)	Fault occurrence time	Fault duration	Detection time
Scenario. **1**	Proposed method AANN PCA	T_B	Drift	0.4 °C/h	July 1, at 02:00	22:00 h	03:12 h 04:18 h 06:18 h
Scenario. **2**	Proposed method AANN PCA	T_D	Precision degradation	[−2 2] °C	July 1, at 06:00	12:00 h	Immediate Failed Failed

* The term "immediate" detection time means detecting a fault immediately one sample after its occurrence.
* The term "Failed" means that the detection process was not successful and the fault was not detected.

SFDI performance and diagnosing faults of smaller magnitude precisely is realizing an accurate and reliable model. The accuracy would be improved by increasing the number of observables. In other words, by considering more neurons in the output layer of the lifting encoder, the long-term prediction of the realized model can be enhanced. This leads to realizing a model with a higher dimension. Subsequently, considering the inequality $sn > N - n$, the parity order will be increased by increasing the dimension of the realized model. However, the higher parity order results in more computation demand, which is undesirable in practice. Hence, there is a trade-off between the SFDI performance and the computation demand of the method.

As shown in the previous section, by adopting a model with one neuron in the lifting encoder's output layer, we achieved significant performance for SFDI in the 4-zone HVAC system. In addition, all types of additive and multiplicative sensor faults are investigated, and it is demonstrated that once a fault with a magnitude within the detectable range occurs, the generated residuals will indicate the fault immediately. However, in more complex systems, such as large-scale HVAC systems, it may be necessary to develop a model with a higher dimension in order to obtain acceptable prediction and SFDI performance.

6. Comparison study

In this section, a comparison is conducted between our proposed approach and two earlier works [15,29], which are based on PCA and AANN. The training dataset described in Section 5.1 is used for training process of the PCA-based and AANN approaches, and the required preprocessing is applied to them. Two fault occurrence scenarios are considered to compare different aspects of the aforementioned methods with our proposed method. The summary of the fault occurrence scenarios and the performance of different methods are provided in Table 5.

Scenario.* 1 *(drift): In this scenario, as shown in Fig. 12, the sensor in zone B is subjected to the drift fault with the gain of 0.4 °C/h. The detection residual of our proposed method, the AANN residual corresponding to the faulty sensor, and the squared prediction error (SPE) of the PCA output is depicted in Fig. 12. As can be seen, the detection time of our proposed method is faster than PCA-based and AANN approaches. In addition, our proposed method leads to zero missed alarm rates. However, the PCA-based and AANN approaches lead to missed alarms after the fault is detected.

Scenario.* 2 *(precision degradation): In this scenario, as shown in Fig. 13, the precision degradation fault with the range of [−2 2] °C is raised in the sensor of zone D. The detection residual of our proposed method, the AANN residual corresponding to the faulty sensor, and the (SPE) of the PCA output is depicted in Fig. 13. As can be seen, the fault is detected immediately and without any missed alarms using our proposed method. However, the PCA-based and AANN methods failed to diagnose the fault.

7. Conclusion

In this work, a data-driven solution is presented for the SFDI problem in the multi-zone HVAC system by integrating the bilinear Koopman model realization, deep learning, and bilinear parity-space. A novel deep neural network scheme is proposed to realize the bilinear model using the normal operational data from the target HVAC system. Then, the realized model is employed in designing the bilinear parity space and generating the residuals for diagnosing sensor malfunctions in the system. The proposed method does not require any prior knowledge regarding the HVAC dynamics and only uses the data collected from the normal operation of the system. Also, it is capable of diagnosing all types of sensor faults, including bias, drift, precision degradation, LOE, and complete failure, with various severity levels. Due to the global characteristics of the Koopman model realization, this method has no constraints imposed on the fault properties, such as fault type or amplitude. Hence, the proposed framework leads to a minimum false alarm rate. The significant capability of the proposed method is demonstrated for diagnosing all types of sensor faults with various severity levels in a four-zone HVAC system simulated in the TRNSYS software. Furthermore, our proposed approach is compared with two other methods based on PCA and AANN to demonstrate its advantages and improvements in terms of detection time and the capability for diagnosing the precision degradation fault. The results indicate that once a fault with a magnitude within the detectable range occurs, it will be diagnosed immediately and precisely. The generalizability of the proposed framework for large-scale buildings with additional zones has to be further examined. In addition, an energy performance assessment of the HVAC system to demonstrate the method's capability for improving the system's efficiency and devising an algorithm to automatically obtain the detection and isolation threshold values remain to be explored in the future.

CRediT authorship contribution statement

Fatemeh Negar Irani: Software, Writing – original draft. **Mohammadhosein Bakhtiaridoust:** Software, Writing – original draft. **Meysam Yadegar:** Conceptualization, Resources, Supervision, Writing – review & editing. **Nader Meskin:** Conceptualization, Resources, Supervision, Writing – review & editing.

Declaration of competing interest

All authors declare that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

Data availability

Data will be made available on request.

Acknowledgement

Open Access funding is provided by the Qatar National Library.

References

[1] M.E.S. Trothe, H.R. Shaker, M. Jradi, K. Arendt, Fault isolability analysis and optimal sensor placement for fault diagnosis in smart buildings, Energies 12 (9) (2019) 1601.
[2] M.S. Mirnaghi, F. Haghighat, Fault detection and diagnosis of large-scale HVAC systems in buildings using data-driven methods: A comprehensive review, Energy Build. 229 (2020) 110492.
[3] V. Singh, J. Mathur, A. Bhatia, A comprehensive review: Fault detection, diagnostics, prognostics, and fault modelling in HVAC systems, Int. J. Refrig. (2022).
[4] R. Isermann, Model-based fault-detection and diagnosis–status and applications, Annu. Rev. Control 29 (1) (2005) 71–85.
[5] M. Du, P. Mhaskar, Isolation and handling of sensor faults in nonlinear systems, Automatica 50 (4) (2014) 1066–1074.
[6] V. Reppa, P. Papadopoulos, M.M. Polycarpou, C.G. Panayiotou, A distributed architecture for HVAC sensor fault detection and isolation, IEEE Trans. Control Syst. Technol. 23 (4) (2014) 1323–1337.
[7] A. Qiu, Z. Yan, Q. Deng, J. Liu, L. Shang, J. Wu, Modeling of HVAC systems for fault diagnosis, IEEE Access 8 (2020) 146248–146262.
[8] P.M. Van Every, M. Rodriguez, C.B. Jones, A.A. Mammoli, M. Martínez-Ramón, Advanced detection of HVAC faults using unsupervised SVM novelty detection and Gaussian process models, Energy Build. 149 (2017) 216–224.
[9] A. Montazeri, S.M. Kargar, Fault detection and diagnosis in air handling using data-driven methods, J. Build. Eng. 31 (2020) 101388.
[10] M. Kim, H. Liu, J.T. Kim, C. Yoo, Sensor fault identification and reconstruction of indoor air quality (IAQ) data using a multivariate non-Gaussian model in underground building space, Energy Build. 66 (2013) 384–394.
[11] R. Sharifi, R. Langari, Nonlinear sensor fault diagnosis using mixture of probabilistic PCA models, Mech. Syst. Signal Process. 85 (2017) 638–650.
[12] S. Li, J. Wen, Application of pattern matching method for detecting faults in air handling unit system, Autom. Constr. 43 (2014) 49–58.
[13] R. Yan, Z. Ma, G. Kokogiannakis, Y. Zhao, A sensor fault detection strategy for air handling units using cluster analysis, Autom. Constr. 70 (2016) 77–88.
[14] Y. Guo, G. Li, H. Chen, Y. Hu, H. Li, L. Xing, W. Hu, An enhanced PCA method with Savitzky-Golay method for VRF system sensor fault detection and diagnosis, Energy Build. 142 (2017) 167–178.
[15] S. Wang, Q. Zhou, F. Xiao, A system-level fault detection and diagnosis strategy for HVAC systems involving sensor faults, Energy Build. 42 (4) (2010) 477–490.
[16] M. Padilla, D. Choiniere, A combined passive-active sensor fault detection and isolation approach for air handling units, Energy Build. 99 (2015) 214–219.
[17] G. Li, Y. Hu, Improved sensor fault detection, diagnosis and estimation for screw chillers using density-based clustering and principal component analysis, Energy Build. 173 (2018) 502–515.
[18] Y. Hu, G. Li, H. Chen, H. Li, J. Liu, Sensitivity analysis for PCA-based chiller sensor fault detection, Int. J. Refrig. 63 (2016) 133–143.
[19] S. Wang, J. Xing, Z. Jiang, Y. Dai, A novel sensors fault detection and self-correction method for HVAC systems using decentralized swarm intelligence algorithm, Int. J. Refrig. 106 (2019) 54–65.
[20] Y. Chen, J. Wen, J. Lo, Using weather and schedule-based pattern matching and feature-based principal component analysis for whole building fault detection—Part I development of the method, ASME J. Eng. Sustain. Build. Cities 3 (1) (2022).
[21] Y. Chen, J. Wen, L.J. Lo, Using weather and schedule based pattern matching and feature based PCA for whole building fault detection—Part II field evaluation, ASME J. Eng. Sustain. Build. Cities (2021) 1–16.
[22] B. Fan, Z. Du, X. Jin, X. Yang, Y. Guo, A hybrid FDD strategy for local system of AHU based on artificial neural network and wavelet analysis, Build. Environ. 45 (12) (2010) 2698–2708.
[23] Z. Du, B. Fan, J. Chi, X. Jin, Sensor fault detection and its efficiency analysis in air handling unit using the combined neural networks, Energy Build. 72 (2014) 157–166.
[24] Z. Du, B. Fan, X. Jin, J. Chi, Fault detection and diagnosis for buildings and HVAC systems using combined neural networks and subtractive clustering analysis, Build. Environ. 73 (2014) 1–11.
[25] W.H. Allen, A. Rubaai, R. Chawla, Fuzzy neural network-based health monitoring for HVAC system variable-air-volume unit, IEEE Trans. Ind. Appl. 52 (3) (2015) 2513–2524.
[26] Y. Zhu, X. Jin, Z. Du, Fault diagnosis for sensors in air handling unit based on neural network pre-processed by wavelet and fractal, Energy Build. 44 (2012) 7–16.
[27] H. Shahnazari, P. Mhaskar, J.M. House, T.I. Salsbury, Modeling and fault diagnosis design for HVAC systems using recurrent neural networks, Comput. Chem. Eng. 126 (2019) 189–203.
[28] M. Elnour, N. Meskin, M. Al-Naemi, Sensor fault diagnosis of multi-zone HVAC systems using auto-associative neural network, in: 2019 IEEE Conference on Control Technology and Applications, CCTA, IEEE, 2019, pp. 118–123.
[29] M. Elnour, N. Meskin, M. Al-Naemi, Sensor data validation and fault diagnosis using Auto-Associative Neural Network for HVAC systems, J. Build. Eng. 27 (2020) 100935.
[30] H. Liao, W. Cai, F. Cheng, S. Dubey, P.B. Rajesh, An online data-driven fault diagnosis method for air handling units by rule and convolutional neural networks, Sensors 21 (13) (2021) 4358.
[31] Y. Yan, J. Cai, Y. Tang, Y. Yu, A decentralized Boltzmann-machine-based fault diagnosis method for sensors of air handling units in HVACs, J. Build. Eng. 50 (2022) 104130.
[32] Y. Chen, J. Wen, O. Pradhan, L.J. Lo, T. Wu, Using discrete Bayesian networks for diagnosing and isolating cross-level faults in HVAC systems, Appl. Energy 327 (2022) 120050.
[33] P. Movahed, S. Taheri, A. Razban, A bi-level data-driven framework for fault-detection and diagnosis of HVAC systems, Appl. Energy 339 (2023) 120948.
[34] G. Li, Y. Hu, H. Chen, H. Li, M. Hu, Y. Guo, S. Shi, W. Hu, A sensor fault detection and diagnosis strategy for screw chiller system using support vector data description-based D-statistic and DV-contribution plots, Energy Build. 133 (2016) 230–245.
[35] H. Li, J. Li, V. Farhangi, Determination of piers shear capacity using numerical analysis and machine learning for generalization to masonry large scale walls, Structures 49 (2023) 443–466.
[36] J. Chen, L. Zhang, Y. Li, Y. Shi, X. Gao, Y. Hu, A review of computing-based automated fault detection and diagnosis of heating, ventilation and air conditioning systems, Renew. Sustain. Energy Rev. 161 (2022) 112395.
[37] M. Bakhtiaridoust, F.N. Irani, M. Yadegar, N. Meskin, Data-driven sensor fault detection and isolation of nonlinear systems: Deep neural-network Koopman operator, IET Control Theory Appl. 17 (2) (2023) 123–132.
[38] M. Bakhtiaridoust, M. Yadegar, N. Meskin, Data-driven fault detection and isolation of nonlinear systems using deep learning for Koopman operator, ISA Trans. 134 (2023) 200–211.
[39] B.O. Koopman, Hamiltonian systems and transformation in Hilbert space, Proc. Natl. Acad. Sci. 17 (5) (1931) 315–318.
[40] M.O. Williams, I.G. Kevrekidis, C.W. Rowley, A data–driven approximation of the Koopman operator: Extending dynamic mode decomposition, J. Nonlinear Sci. 25 (6) (2015) 1307–1346.
[41] A. Mauroy, J. Goncalves, Koopman-based lifting techniques for nonlinear systems identification, IEEE Trans. Automat. Control 65 (6) (2019) 2550–2565.
[42] D. Bruder, X. Fu, R. Vasudevan, Advantages of bilinear Koopman realizations for the modeling and control of systems with unknown dynamics, IEEE Robot. Autom. Lett. 6 (3) (2021) 4369–4376.

[43] M. Bakhtiaridoust, M. Yadegar, N. Meskin, M. Noorizadeh, Model-free geometric fault detection and isolation for nonlinear systems using Koopman operator, IEEE Access 10 (2022) 14835–14845.
[44] M. Kioumarsi, H. Dabiri, A. Kandiri, V. Farhangi, Compressive strength of concrete containing furnace blast slag; optimized machine learning-based models, Clean. Eng. Technol. 13 (2023) 100604.
[45] P.M. Frank, Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey and some new results, Automatica 26 (3) (1990) 459–474.
[46] D. Yu, D.N. Shields, Extension of the parity-space method to fault diagnosis of bilinear systems, Internat. J. Systems Sci. 32 (8) (2001) 953–962.