Contents lists available at YXpublications

International Journal of Applied Mathematics in Control Engineering

Journal homepage: http://www.ijamce.com

HVAC System Modeling and Fault Simulation Research Based on Modelica

Hui Zhu[a], Aiping Pang [a,*], Huan Feng[a]

[a] Guizhou University, School of Electrical Engineering, Guiyang 550025, China

ARTICLE INFO

Article history:
Received 21 January 2021
Accepted 28 March 2021
Available online 30 March 2021

Keywords:
HVAC system
Fault simulation
Modelica modeling

ABSTRACT

HVAC is a multifunctional air conditioning system that integrates several subsystems such as cooling subsystem and air handling subsystem interacting simultaneously. The causal modeling approach is very complex and difficult to establish an accurate physical model for HVAC system. In addition, the actual HVAC is generally not allowed to operate in a fault state. Therefore, it is difficult to obtain comprehensive and adequate fault sample data directly from the actual HVAC system. In order to solve the shortcomings of causal modeling and the inadequate fault samples of the actual system, an acausal modeling method based on the modelica language is proposed and the faults are simulated in the simulation platform to generate fault samples based on the fault types of the actual system. The results show that this method is not only able to build HVAC system models with more similar topology to the actual physical system but also has better scalability for effective fault simulation of the whole system compared to the traditional causal modeling approach.

1. Introduction

HVAC (Heating Ventilating and Air Conditioning) systems are multifunctional air conditioning systems that are applied at important locations such as data centers and communication stations [1-2]. There is an urgent need to explore the fault characteristics of the system for accurate and effective fault detection to ensure and improve the reliability and safety of HVAC systems [3-4]. However, the data of fault samples to be used for the research of system fault diagnosis methods and Multi-source alarm information fusion are very limited due to the limited number of actual HVAC system sensors and the inability to perform dedicated fault experiments[5]. Therefore, the study of system faults simulation analysis plays a very important role, and the prerequisite for the study of HVAC system faults is the development of the system model [6]. In the paper [7], a residential HVAC system model was developed based on Simulink, in which different physical characteristics of the building and weather conditions were described through mathematical equations to establish a physical model that can accurately predict the energy consumption of the air conditioning system. In the literatures [8-9], the chiller components were described mathematically, which was based on partial differential equations to establish the chiller model of air conditioning system, as well as the control strategy research with the model. The literature [10] proposes a dynamic modeling method for absorption chillers, which includes evaporator, condenser, and indirect heat exchanger, etc. The simulation results show that the model exhibits excellent representation of the characteristic parameters regardless of the steady-state or transient parameter changes of the chiller unit. The literature [11] proposes a modeling approach that combines Modelica with Python. The method uses the equation-based Modelica programming language to construct simulation models of typical components of air conditioning systems, and calibrates the models with actual operational data. Finally, the Modelica models are integrated into the Python environment resulting in HVAC system models that are computationally fast and suitable for real-time applications. The literature [12] discusses two different modeling approaches, chunked Simulink and bonding diagrams, and describes the advantages and disadvantages of both approaches. In the paper, a fault detection and diagnosis model of HVAC system is developed for an office building air conditioning room by developing a modeling program using a combination of Simulink and bonding diagrams. The literature [13] shows that existing studies have developed various fault detection and diagnosis methods to identify and analyze operational faults in components or subsystems of HVAC systems. However, the current research lacks a comprehensive approach to identify the overall impact on the HVAC system from the failure of each individual component. For this problem, the authors outline that operational fault can be modeled and simulated using Energy Plus to quantify the impact of faults on the system. Comprehensive research found that there is

* Corresponding author.
E-mail addresses: appang@gzu.edu.cn (A. Pang)

still less research on HVAC system fault simulation, and most of the research is usually only for a single component or a subsystem of the HVAC system, so the overall system fault simulation analysis still needs to continue to deepen.

An acausal modeling approach was used to establish a complete HVAC system model in order to study how a fault in the HVAC system would affect the operation of the entire system. The acausal modeling division of problems based on system functions is different from the causal modeling division of problems by system steps, so it has the advantages of accurate and fast modeling and good reusability [14]. The Modelica language is an acausal modeling and open modeling language that is well suited for modeling HVAC systems [15]. There have been studies with results using its powerful multi-domain complex system modeling capabilities for power systems, aero-engines, vehicles, etc [16-19]. Therefore, this paper uses OpenModelica, a Modelica language simulation platform, to modularly model the HVAC system, then simulates the typical faults of the whole system based on the simulation model.

2. Modelica language and simulation platform

2.1 Modelica

Modelica language is a directly object-oriented modeling computer simulation language based on the C programming language. It can build the physical processes of complex systems by using mathematical equations to describe the various subsystems of different domains. The Modelica language achieves the integration of multi-domain complex systems based on the actual physical system topology with its intrinsic component connection principle. Finally it realizes the simulation of the integrated system by using differential algebraic equations for each subsystem. Table 1 describes the basic classes and features of the Modelica language. The most prominent feature of the language is the concept of components and connections, where system components interact with information through an intrinsically defined connection mechanism. This feature both distinguishes the Modelica language from general programming languages and is superior to block-based Simulink simulation modeling with one directional data transfer.

Tab. 1. Modelica types and features

Types	Role
class	general class
connector	interfaces between components
package	model level
block	block diagram style modeling
function	algorithmic modeling
record	data structure

There are two general Modelica modelling methods: one is to use a text box based on Modelica syntax to model in the form of text code; the other is to model through a graphical tool by importing the Modelica standard library, dragging and dropping the model from the library into the component view area of the simulation platform, then giving the parameters of the component and connecting the individual components to build the model.

The syntactic structure of Modelica is now illustrated by the creation of a Newtonian cooling formula. For example, a connector class FluidPort, given below, defines an interface for one-dimensional fluid flow in a pipe network and contains two variables, the pressure P as a potential variable and the mass flow rate m_flow as a flow variable.

$$\begin{array}{l}
\mathrm{model} \quad newtoncooling \\
parameter \quad \mathrm{Real} \quad T_\inf \\
parameter \quad \mathrm{Real} \quad T_0 \\
parameter \quad \mathrm{Real} \quad h \\
parameter \quad \mathrm{Real} \quad A \\
parameter \quad \mathrm{Real} \quad m \\
parameter \quad \mathrm{Real} \quad c_p \\
\mathrm{Real} \quad T \\
Initial \quad equation \\
T = T_0 \\
equation \\
m \times c_p \times der(T) = h \times A \times (T_\inf - T) \\
\quad end \quad newtoncooling
\end{array}$$

In modelling in the form of text code, the first is the "class", which is the basic unit of the Modelica model and consists of three constituent members: variables, equations and member classes. Variables represent the properties of a class, usually representing a physical quantity. Equations specify the behaviour of a class, expressing the numerical constraints between variables. Classes can also be members of other classes. Members of a class can be defined directly or obtained from a base class through inheritance. General classes are modified by the keyword class and specific classes are modified by specific keywords such as model, connector, record, block and type. A specific class is simply a specialised form of the general class concept, and the specific class keyword can be replaced by the general class keyword class in a model without changing the behaviour of the model.

In the Modelica language, the interfaces of components are called connectors and Modelica connectors are instances of the connector class. The coupling relationships established on component connectors are called connections and connections can be classified as causal or non-causal depending on the causal coupling relationship between them. The main purpose of the connector class is to define the properties and structure of the component interface. The variables defined in connector classes can be divided into two types: stream variables and potential variables. Flow variables are "through" variables such as currents, forces, moments, etc. and are defined by the keyword flow. The coupling between flow variables is represented by an equation of the form "and zero". Potential variables are "crossing" variables such as voltage, displacement, angle, etc. The coupling between the potential variables is represented by an equation in the form of "equivalence".

$$\begin{array}{l}
connector \quad Fluidport \\
\mathrm{Real} \quad P; \\
flow \quad \mathrm{Real} \quad m_flow; \\
end \quad Fluidport
\end{array}$$

For example, a connector class FluidPort, given below, defines an interface for one-dimensional fluid flow in a pipe network and contains two variables, the pressure P as a potential variable and the mass flow rate m_flow as a flow variable.

A Modelica connection must be built on two connection interfaces of the same type to express the coupling between the different components. This coupling is semantically realised through equations, so Modelica connections are transformed into equations when the model is compiled. According to Kirchhoff's two laws, the coupling between flow variables is represented by an equation in the form of a "sum of zeros", i.e. the sum of the flow variables at the intersection of the connections is zero. The coupling between the potential variables is represented by an equation in the form of "equal values", i.e. the potential variables at the intersection of the connections are of equal value. The equation of connection reflects the power balance, momentum balance or mass balance at the actual physical connection point. Suppose there is a connection connect(FluidPort1,FluidPort2), where FluidPort1 and FluidPort2 are two instances of the connection class FluidPort. The connection is equivalent to the following two equations.

$$\begin{cases} Fluidport1.p = Fluidport2.p \\ Fluidport1.m\_flow + Fluidport2.m\_flow = 0 \end{cases} \tag{1}$$

2.2 OpenModelica simulaton platform

OpenModelica is one of the simulation platforms for the Modelica language as an integrated environment for modeling and simulation for complex systems in multiple domains. It enables users to model complex multidisciplinary domain systems and perform steady-state and dynamic performance studies of any component or entire system on its simulation platform. There are two methods of modeling based on OpenModelica. One is to model the system in the form of text code based on Modelica. The other method is to import the Modelica standard library and drag the component modules from the library to the component view interface to connect the components for modeling. Users can change the original modules or create new ones as needed since the code of each of its modules is publicly available.

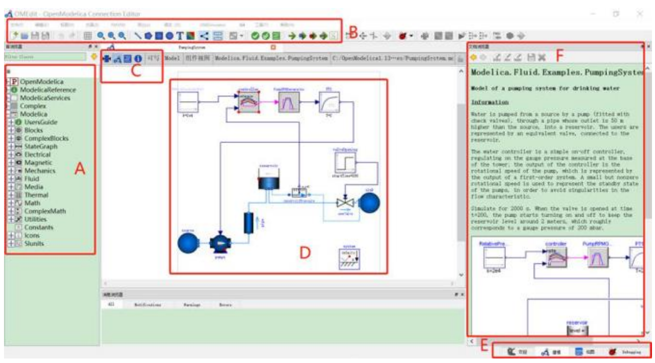

Fig. 1. The modeling work interface of OpenModelica software.

Figure 1 shows the modelling work interface of OpenModelica software, in which area A is the display area of the system standard model library; area B is the toolbar and menu bar; area C has a toolbar to switch between modelling windows; area D is the modelling window, which is divided into four sub-windows: Icon, Diagram, Modelica text and Documentation The modeling window is divided into four sub-windows: Icon, Diagram, Modelica text and Documentation, which are used to describe and build models and model components, to accumulate and reuse models and their accompanying knowledge; the toolbar in area E is used to switch between the modelling and simulation environments; area F is the documentation browser, which is used to explain the models in the modelling area.

The OpenModelica simulation environment consists of the following interconnected subsystems, as shown in Figure 2. The arrows indicate data and control flows. Interactive session handlers receive commands and display the evaluated translated and executed the results of the commands and expressions. The rest of the subsystem provides different forms of Modelica code browsing and text editing. The debugger currently provides debugging of a subset of Modelica's extended algorithms, and uses Eclipse for display and positioning. The Graphical Model Editor provides graphical model editing, plotting and browsing of the Modelica standard library. By default, OpenModelica converts Modelica models into ODE or DAE representations to perform simulations using numerical integration methods. DASSL is used as the default solver for OpenModelica, it is an implicit, high order multi-step solver with step control and has these properties and is very stable for a wide range of models.

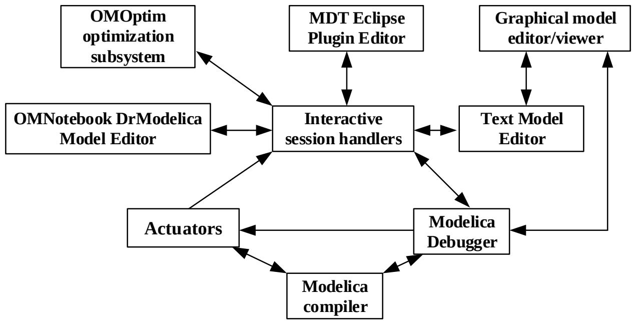

Fig. 2. Architecture of the OpenModelica simulation environment.

3. HVAC system model

The studied HVAC system is a water-cooled air conditioning system for a factory. The system is mainly responsible for supplying cooling to the factory in order to maintain the factory temperature at about 20℃ at all times. The structure of the HVAC system is shown in Fig. 3.

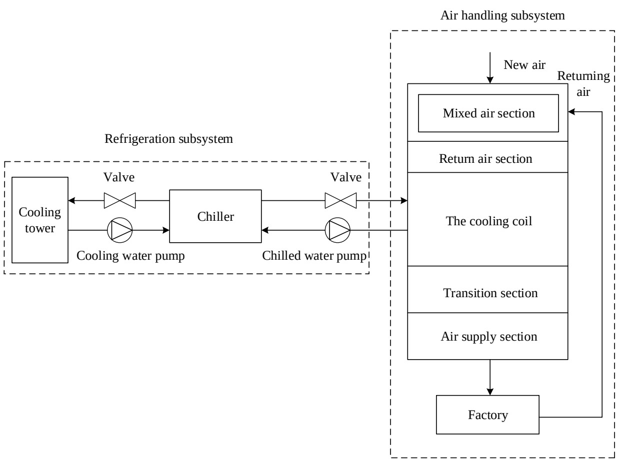

Fig. 3. The structure of the HVAC system.

3.1 Chiller model

The chiller is the core equipment of HVAC system and its working principle is mainly through the evaporation of refrigerant to realize heat absorption for cooling. The compressor compresses the refrigerant gas into a high temperature and high pressure gas and sends it to the condenser, then the condenser cools the refrigerant gas into a high pressure liquid and sends it to the evaporator. The refrigerant evaporates and absorbs the heat of the chilled water in the evaporator to make the temperature of the chilled water drop and the evaporated refrigerant returns to the compressor for the refrigeration cycle to achieve the purpose of refrigeration. The mathematical description of the chiller cooling process is as follows.

Equations (2) and (3) are the evaporator energy balance equation of the chiller.

$$\eta_e = \beta_{ei} S_{ei} (T_{wo} - T_e) \tag{2}$$

$$F_r (Q_1 - Q_2) = \beta_{eo} S_{eo} (T_e - T_{eo}) \tag{3}$$

The energy balance equation of the chiller condenser is as follows.

$$\eta_c = \beta_{ci} S_{ci} (T_w - T_{cwo}) \tag{4}$$

$$F_r (Q_3 - Q_4) = \beta_{co} S_{co} (T_{ci} - T_w) \tag{5}$$

3.2 Cooling tower model

Cooling towers use the contact of water and air to cool the cooling water of chillers by evaporation.

Equations (6), (7) are the mathematical description of the cooling tower heat dissipation process.

$$\eta_t = \mu Q_{\min} (T_{cw} - T_a) \tag{6}$$

$$Q_{\min} = \min(C_w F_w, C_a F_a) \tag{7}$$

3.3 Cooling coil model

The cooling coil of the air handling unit mainly cools the air flowing outside the tube through the chilled water flowing inside the tube, and the fan transports the cold air to the room to supply the room with cold. The chilled water is brought back to the chiller from the return pipe of the cooling coil and the cooled chilled water is then sent back to the fan coil. The energy balance equations of the hot and cold fluids of the cooling coil based on the energy balance theorem are as follows.

Heat transfer equation on the cold water side:

$$C_w m_w \frac{dT_{wo}}{dt} = Aq - F_w C_w (T_{wi} - T_{wo}) \tag{8}$$

Air-side heat transfer equation:

$$C_a m_a \frac{dT_{ao}}{dt} = F_a C_a (T_{ai} - T_{ao}) - Aq \tag{9}$$

3.4 Factory model

The factory temperature is affected by various factors such as outdoor temperature, air supply volume and air supply temperature. According to the principle of energy conservation, the difference between the energy entering the room and the energy leaving the room per unit time is the rate of change of energy storage in the factory.

The model of the factory can be represented by equations (10), (11).

$$C \frac{dT}{dt} = W + \rho C_a U (T_\alpha - T) - \frac{T - T_0}{R} \tag{10}$$

$$C = C_m + C_a \rho V \tag{11}$$

The variables and parameters throughout this paper are defined in Table 2.

Tab. 2. Parameters in this paper

Parameters	Description
η_e	evaporator heat transfer rate
β_{ei}	heat transfer coefficient of the refrigerant into the evaporator
S_{ei}	evaporator inlet area
T_{wo}	return water temperature
T_e	evaporator wall temperature
F_r	mass flow rate of refrigerant
Q_1	refrigerant enthalpy
Q_2	refrigerant enthalpy at evaporator inlet
β_{eo}	heat transfer coefficient of refrigerant leaving the evaporator
S_{eo}	evaporator outlet area
T_{eo}	the refrigerant temperature at the evaporator inlet
η_c	condenser heat transfer rate
β_{ci}	heat transfer coefficient of refrigerant entering the condenser
S_{ci}	condenser inlet area
T_{cwo}	condenser return temperature
T_w	condenser wall temperature
Q_3	refrigerant enthalpy at the condenser outlet
Q_4	the refrigerant enthalpy at the condenser inlet
β_{co}	heat transfer coefficient of the refrigerant leaving the condenser
S_{co}	condenser outlet area
T_{ci}	refrigerant temperature at the condenser inlet
η_t	cooling tower heat dissipation rate
μ	heat transfer efficiency
T_{cw}	cooling tower heat dissipation rate
T_a	ambient weather temperature
F_w	cooling water mass flow rate
F_a	air mass flow rate
C_w	specific heat capacity of water
m_w	mass of water stored in the coil

T_{wo}	temperature of the exported water
A	characteristic parameter of the surface cooler
q	convection heat exchange
F_w	mass flow of chilled water
T_{wi}	temperature of the imported chilled water
C_a	specific heat capacity of air
m_a	mass of air stored in the coil
T_{ao}	temperature of the exported air
F_a	mass flow of air
T_{ai}	temperature of the imported air
T	indoor air temperature
T_0	outdoor air temperature
T_α	supply air temperature
C	heat capacity in the room
C_m	wall heat capacity
ρ	supply air density
R	total heat transfer thermal resistance of the room wall structure
W	indoor heat dissipation
V	volume of the factory

3.5 System model and calibration

The modules of the HVAC system consist of two parts： the interface and the implementation. In the Modelica language, the two types of quantities included in the interface are potential variables and flow variables.The potential variable represents the quantity that has equal values at both ends of the interface, while the flow variable represents the quantity that sums to zero. According to the system theory, the product of the flow and potential variables of the same interface should be equal to the work transferred by the port. Therefore, the pressure is selected as the potential variable and the flow rate is the flow variable in the refrigeration subsystem and the air handling subsystem. The simulation model of the HVAC system is built on the OpenModelica platform based on the above modeling ideas and the models of each component.

After the modeling was completed, the model parameters were set and calibrated based on the sensor measurement data of an actual HVAC system in a factory. According to the field data, the actual HVAC system maintains the temperature of the factory at about 20°C. Therefore, the factory temperature of the established simulation model in the stabilization phase will be similar to the actual factory temperature.The factory temperature simulation curve of the simulated system after calibration is shown in Figure 4, which shows that the factory temperature of the system is maintained at about 293 K (20°C). In order to check the difference between the established model and the real system model, the factory temperature of the actual system and the corresponding factory temperature of the simulation model were used as monitoring indicators for comparison. The actual system factory temperature was sampled every half hour to obtain 500 samples. A comparison of the simulated system factory temperature and the actual system factory temperature is shown in Figures 5 and Figure 6.

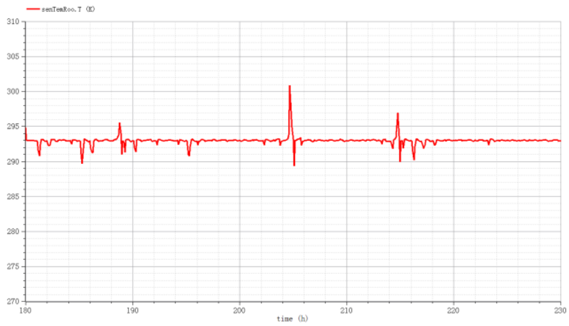

Fig. 4. Diagram of factory temperature change.

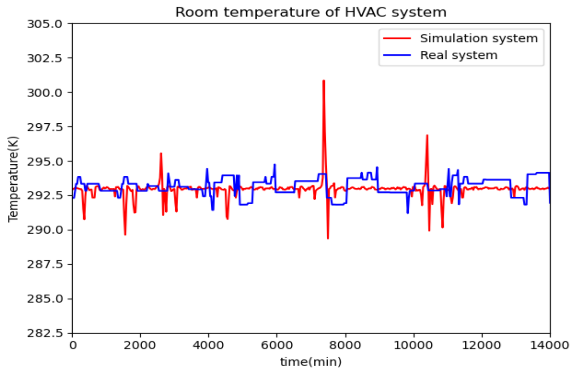

Fig. 5. Comparison of factory temperature between simulated system and actual system.

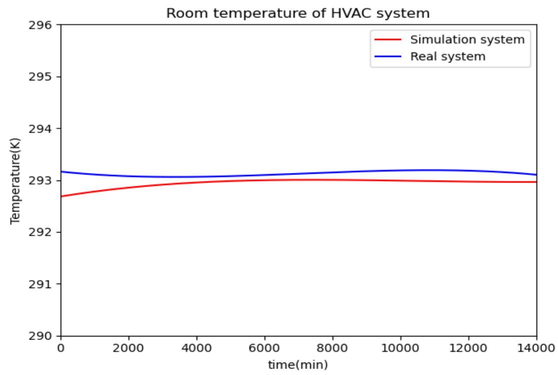

Fig. 6. Comparison graph of sampling points fitted for temperature.

From Figure 5 and Figure 6, it can be seen that the factory of the simulated system and the real system both fluctuate around 293K. Therefore, the established simulation system differs little from the real system and it is relatively realistic and accurate.

4. Typical faults simulation

Typical failures of HVAC systems are summarized in the literature [16], which shows that HVAC system failures are divided into sensor failures and equipment failures. Among them, sensor faults mainly include four types of faults, such as deviation, drift, decrease in accuracy and complete failure, while equipment faults include chiller refrigerant leakage, chilled water pump blockage, evaporator fouling, etc.In this paper, three representative typical faults are selected and manually injected into the simulation system, namely sensor deviation fault, chiller evaporator scale and air supply pipe rupture.

4.1 The fault of sensor deviation

The sensor deviation fault is mainly a type of fault where the measured value of the fault differs from the correct measured value by a constant constant. The mathematical description is given in the following equation.

$$x = x + b + \varepsilon \tag{12}$$

where , x is the measured value of the measured variable, x is the true value, b is the constant, and ε is the random error of the measurement.

The chilled water sensor of the HVAC system is simulated to operate in normal operation and with a deviation fault. After the system is stabilized in normal operation, a positive deviation fault of 5℃ is set for the chilled water sensor at a certain point of stable operation, and the sensor readings are monitored in the simulation. The simulation curves are shown in Figure 7. In figure 7, the red curve is the monitored value with normal sensor, while the blue curve is the monitored value when the sensor has a positive deviation.

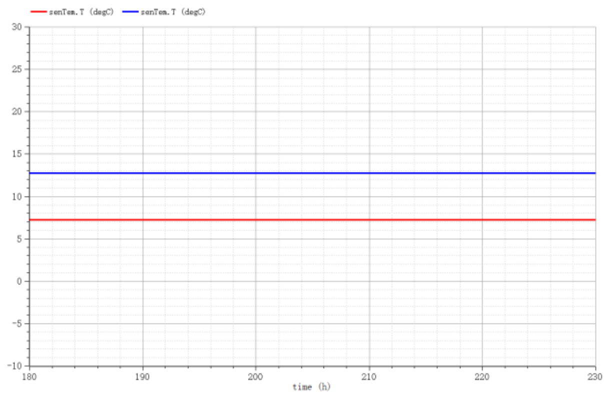

Fig. 7. Sensor measurement data curve in operation.

4.2 Evaporator Fouling of Chiller

Evaporator fouling will lead to a reduction in heat transfer between chilled water and refrigerant. As the heat exchange effect of evaporator decreases, its heat transfer coefficient COP value will also decrease. It can be concluded that the COP value of the chiller is an important thermodynamic parameter to detect whether the evaporator is scaled or not. The chiller COP value monitoring curve is shown in Figure 8 and Figure 9.

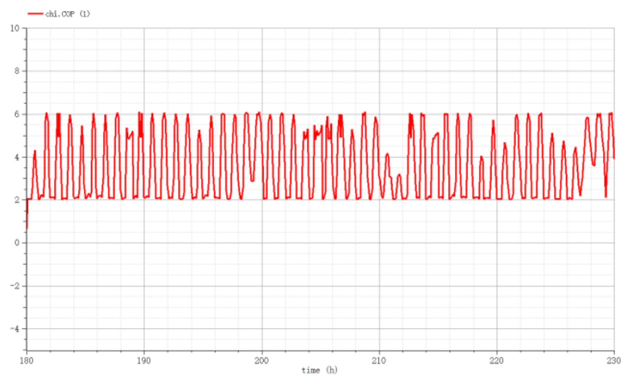

Fig. 8. The COP curve of chiller in normal condition of evaporator.

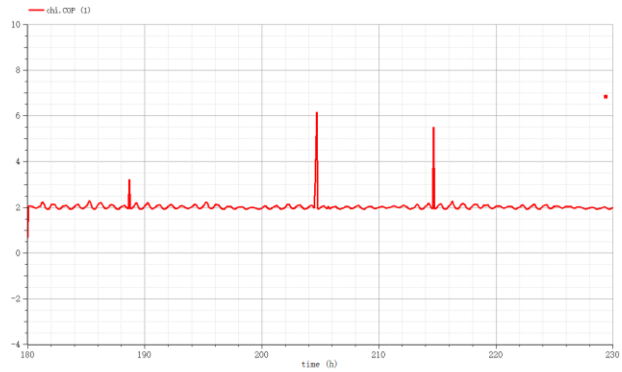

Fig. 9. The COP curve of chiller with evaporator fouling state.

4.3 Rupture of air supply duct

The rupture of a supply air duct changes the temperature of a room mainly by affecting the amount of air supplied to the room for cooling. The rupture of the duct will reduce the air supply volume resulting in an increase in the room temperature. The room temperature monitoring curves are shown in Figure 10 and Figure 11.

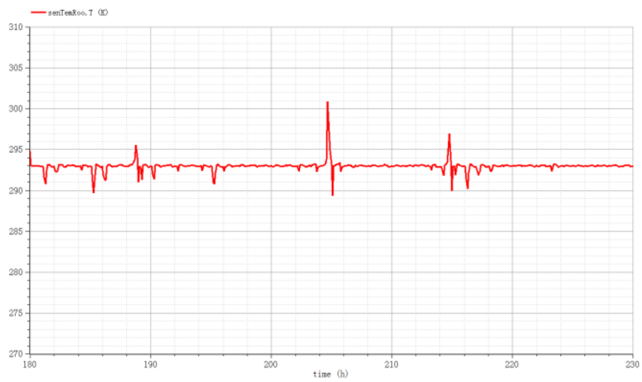

Fig. 10. Room temperature of normal air supply pipe.

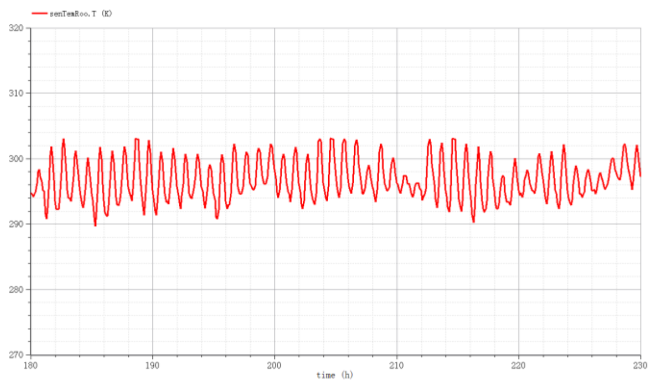

Fig. 11. Room temperature of air supply pipe rupture .

According to the above three types of fault simulation, the rest of the HVAC system can be simulated by adjusting the module where the fault is located or by changing the parameters of the module. However, the fault simulations for chilled water sensor accuracy degradation and action valve failure require to add the corresponding controller.

5. Conclusion

Acausal modeling techniques are used in this paper to model the HVAC system by using the modular modeling tool OpenModelica.

Furtherlly, the calibration of the model based on the measurement data of the real system can effectively reflect the physical characteristics of the real system. Finally, the typical fault simulation is performed on the calibration simulation system. This experiment does not require changing the mathematical model of the system, but only adjusting the system module or changing the system module parameters. The results show that the acausal modeling approach is more scalable enabling it to simulate more adequate fault types and have wider engineering applications than the traditional causal modeling approach. Meanwhile, the method can provide a reference for fault diagnosis research of HVAC systems when the fault samples are lacking.

Acknowledgements

This work is supported by Science and Technology Foundation of Guizhou Province (QiankeHe[2020]1Y273), Guizhou University Incubation Program, Guizhou University Incubation [2019] 60.

References

[1] Zhu X,Du Z,Jin X,et al.Fault diagnosis based operation risk evaluation for air conditioning systems in data centers[J]. Building and Environment, 2019, 163(Oct.):106319.1-106319.14.

[2] Sun, L., L. Sun, and Y. Li, Development of Air Conditioning Fault Detection Technology in Existing Buildings. International Journal of Applied Mathematics in Control Engineering, 2019. 2(2): p. 170-175.

[3] Afroz Z,Shafiullah G,Urmee T,et al.Modeling techniques used in building HVAC control systems: A review[J]. Renewable & Sustainable Energy Reviews, 2018:S1364032117314193.

[4] Du, W., et al., Fault Diagnosis of Sensors based on Empirical Mode Decomposition and Neural Network. International Journal of Applied Mathematics in Control Engineering, 2019. 2(1): p. 66-73.

[5] Zheng, G., P. Huang, and G. Li, Multi-source Alarm Information Fusion Considering Space-time Characteristics. International Journal of Applied Mathematics in Control Engineering, 2019. 2(1): p. 9-16.

[6] Afram A,Janabi-Sharifi F.Review of modeling methods for HVAC systems[J]. Applied Thermal Engineering, 2014, 67(1-2):507-519.

[7] Kassas M.Modeling and Simulation of Residential HVAC Systems Energy Consumption.[C]// Workshop. 2015.

[8] Zhang W J , Zhang C L , Ding G L . Transient modeling of an air-cooled chiller with economized compressor. Part I: Model development and validation[J]. Applied Thermal Engineering, 2009, 29(11-12):2396-2402.

[9] Zhang W J , Ding S F , Zhang C L . Transient modeling of an air-cooled chiller with economized compressor. Part II: Application to control design[J]. Applied Thermal Engineering, 2009, 29(11-12):2403-2407.

[10] Misenheimer C T , Terry S D . The development of a dynamic single effect, lithium bromide absorption chiller model with enhanced generator fidelity[J]. Energy Conversion and Management, 2017, 150(oct.):574-587.

[11] Zabala L,Febres J,Sterling R,et al.Virtual testbed for model predictive control development in district cooling systems[J]. Renewable and Sustainable Energy Reviews, 2020, 129:109920.

[12] Yu B,Paassen A.Simulink and bond graph modeling of an air-conditioned room[J]. Simulation Modelling Practice & Theory, 2004, 12(1):61-76.

[13] Zhang R,Hong T.Modeling of HVAC operational faults in building performance simulation[J]. Applied Energy, 2017.

[14] Li X,Yao R,Li Q,et al. An object-oriented energy benchmark for the evaluation of the office building stock[J]. Utilities Policy, 2018, 51(APR.):1-11.

[15] Sodja A,Skrjanc I,Zupancic B.Cyber-physical modelling in Modelica with model-reduction techniques[J]. The Journal of Systems and Software, 2020, 163(May):110517.1-110517.13.

[16] Vanfretti,Li W,Bogo D,Orova T,et al.Unambiguous power system dynamic modeling and simulation using modelica tools[C]// Power and Energy Society General Meeting (PES), IEEE, 2013.

[17] Zhang L.Convergence Approach to Model Physical World and Cyber World of Aviation Cyber Physical System[C]//IEEE International Conference on Dependable. IEEE, 2014.

[18] Andreasson J,Andersson D,Batteh J,et al.Integrated simulation of a e4WD vehicle using Modelica[J].Ifac Proceedings Volumes, 2013, 46(21):440-445.

[19] Du Z,Bo F,Jin X,et al.Fault detection and diagnosis for buildings and HVAC systems using combined neural networks and subtractive clustering analysis[J].Building & Environment, 2014, 73(mar.):1-11.

Hui Zhu is currently pursuing the master degree in School of Electrical Engineering at Guizhou University, Guiyang, China, and works with Aiping Pang as a Student Researcher. His research interest includes the robust control, H-inf control, Spacecraft control, system fault diagnosis and intelligent algorithms.et.

Aiping Pang received the M.S. degrees in Mathe-matics College of Science from Harbin Institute of Technology, China in 2013. And the Ph.D. degree in Control Science and Engineering College of Aerospace from Harbin Institute of Technology, China, in 2018. She is currently a Associate Professor in the Department of Automation, School of Electrical Engineering, Guizhou University, Guiyang, China. Her major field of study are robot control, H-inf control, delay system control, Spacecraft control, et.

Huan Feng is currently pursuing the masters degrees in Guizhou University at Guiyang, Guizhou, China. He works with Aiping Pang as a Student Researcher. His current research interests include Structured synthesized and intelligent algorithms.