Fault Injection Framework for Demand-Controlled Ventilation and Heating Systems Based on Wireless Sensor and Actuator Networks

Ali Behravan
Chair for Embedded Systems
University of Siegen
Siegen, Germany
Ali.behravan@uni-siegen.de

Roman Obermaisser
Chair for Embedded Systems
University of Siegen
Siegen, Germany
Roman.obermaisser@uni-siegen.de

Mohammad Abboush
University of Siegen
Siegen, Germany
Mohammad2.abboush@student.uni-siegen.de

***Abstract*—The demand-controlled ventilation as an advanced control approach is one of the recent developments in smart building technologies. The aim is the optimization of energy consumption, occupant comfort and air quality based on cost-effective, flexible, scalable, and low-power wireless sensor and actuator networks that facilitate monitoring and control of the building automation system. However, the device nodes and communication routes are error-prone due to various types of faults. When a fault arises in the network or in the nodes, the sensor nodes may produce erroneous data and the actuator nodes' behavior can differ from their expected action on the physical plant. Therefore, this study successfully explicates a novel fault injection framework as a tool that scholars can simply monitor the behavior of this system in the occurrence of different types of faults which are artificially injected or add their own desired type of fault to this framework. Then, authors indicate the fault-error-failure propagation model in component level and system level. The final aim of authors is to use this framework for their future research of testing fault detection and diagnosis methods. This demand-controlled ventilation and heating system is created based on wireless sensor and actuator networks which is more compatible with reality as the wireless communication is very prevalent nowadays and this wireless model is validated by the previous cabled model. The literature research by the authors indicates the excellence of the ZigBee protocol for building automation. In the result section, some samples from the behavior of the system in healthy-mode and faulty-mode in the format of temperature signals as the controlled variable and the comparison of energy consumption of heating system in healthy mode and different faulty modes are shown.**

Keywords—wireless sensor and actuator network, simulation, ZigBee protocol, fault injection, TrueTime

I. Introduction

Internet of Things (IoT) refers to an advanced connection of various devices and systems to connect them for data exchange and coordination. Its noteworthy applications such as building automation systems or smart homes with the evolvement of embedded devices such as sensors, actuators with wireless communication capability absorb lots of research interests nowadays. The ultimate goal of the smart building application is to minimize energy consumption as the building sector consumes a vast part of the energy, e.g. 40% of the total energy in the European Union [1, 2]. Another goal is to increase the occupants' comfort to improve the productivity, health, and quality of their lives by achieving the intelligent control of buildings. The goal of IoT in smart buildings is a building with user-friendly functionality which is equipped with easy monitoring, control, and maintenance systems that support simple installation and configuration, and further expansions and updates. One important application of the building automation systems is heating, ventilation and air conditioning (HVAC). In addition, the demand-controlled ventilation (DCV) is an automatic adjustment of ventilation according to the fresh air demand based on sensor measurements of air quality parameters (CO_2 measurements, occupancy, or in some cases, humidity measurements) or need in every zones at any time. Damper actuators play the role of adjustment. DCV increases the potential energy saving in heating systems by decreasing the heating load of system. Studies demonstrate that 15% to 25% of the HVAC system's energy can be saved by setting the ventilation rates based on the occupancy's fresh air requirement [3]. Behravan et al. describe thermal dynamic modeling and simulation of a heating system for a multi-zone office building equipped with demand controlled ventilation using MATLAB/Simulink [3].

On the other hand, the new technologies in the field of data communication will furnish an efficient system architecture, especially in the case that the cabled network has some shortcomings such as difficulty in device access. Therefore, the HVAC industry focuses on the application of wireless technology in buildings for better control of the indoor climate [4]. The wireless sensor and actuator network (WSAN) as a characteristic technology of the building automation system is a composition of distributed networked embedded devices such as sensors, actuators, and controllers that are placed freely in a plug-and-play fashion. These devices communicate through a wireless network instead of conventional wired connections for monitoring, information gathering, tracking, and control applications. Wireless systems are able to offer building owners and facility managers more choices and fewer constraints, including ease of deployment, cost benefits, scalability of the network, simpler and more flexible system design, faster and less disruptive installations and retrofits [5, 6]. Additionally, the cabling failures are automatically omitted in WSANs and robust communication over different network topologies can be established. From the viewpoint of control theory, traditional WSNs are open-loop systems that only detect the physical world, whereas WSANs are closed-loop systems that can further interact with it automatically [7]. In WSAN, sensor nodes attached to the physical plant, sample and transmit their measurements to the controller over a wireless channel; controllers compute control commands based on these sensor data, which are then forwarded to the actuators in order to influence the dynamics of the physical plant [8, 9]. The development of DCV and heating systems based on WSAN is a challenging task since it is a hybrid solution of an advanced-control continuous time-driven system on one hand, and a wireless network discrete event-driven system on the other hand. This is the integration of the

978-1-5386-7266-2/18/$31.00 ©2018 IEEE

cyber world with the physical systems which is known as Cyber-Physical System (CPS). Due to the limitations of the practical development of CPSs, simulation is a promising solution to design the system, to monitor the system behavior, and to test the performance. In literature, there are many works study the WSNs but without taking actuators into account or they didn't use in Simulink environment [10-12]. Guinard et al. developed a scalable single-hop WSN software tool using Borland C++ to gather sensor information for building energy management but they didn't consider the control system and actuators [13]. Song et al. proposed a real-time simulation model of a WSAN control system that nodes communicate by IEEE 802.15.4 protocol using TrueTime but for a different plant which is a DC servo [14]. Therefore, the first achievement of this study is the newfound simulation of the demand-controlled ventilation and heating system based on wireless sensor and actuator networks in MATLAB/Simulink.

Moreover, the WSANs are prone to hardware, software and communication failures. A hardware failure may occur due to a fault affecting a sensor or actuator hardware modules such as physical damage, a power module such as lack of power, or a processing module. Software failures can arise due to an issue of programming. Communication failures can happen due to issues in data transmission of the network such as interference fault. Faults in the wireless communication networks can result in the degradation of the control system performance, decrease of quality of service (QoS), and even more serious economic losses or decreased human safety. In a survey of buildings in the UK, the data showed 25–50% of energy is wasted due to faults in building HVAC systems. This range could be reduced below 15% whenever those faults are detected and identified early before unacceptable damages occur [15]. Therefore, this paper presents a novel fault injection framework to trace the behavior of the system in occurrence of different of faults and to realize the effect of different type of faults on energy consumption of the system or occupancy comfort by comparing them in non-faulty mode and different faulty modes which is described in the result section. Also, this study will be used in future research of authors to test various techniques of fault detection and diagnosis. The remainder of this study is organized as follows: Section II briefly describes the system model description and simulation. The details of the fault injection framework are presented in Section III. In Section IV, the results are presented.

II. System Model Description and Simulation

The simulation is often more effective than an experimental setup from cost, time, and risk points of view. In the simulated wireless system, the sensors associated with the physical plant, monitor and send their ambient measurements (e.g. temperature) or air quality measurements (e.g. CO_2 concentration) to the controller over a wireless network channel via a communication protocol and the commands after processing which are the output of the controller, will be forwarded to the actuators (e.g. dampers or heater thermostats) in order to influence the dynamics of the physical plant (e.g. building). Every connection between devices are wireless. This study was implemented in MATLAB/Simulink 2018a using the TrueTime 2.0 Block Library.

A. TrueTime 2.0

TrueTime was introduced by Cervin et al. in 2010 [16]. TrueTime simulates the real-time behavior of multi-tasking kernels containing controller tasks, network transmissions, and continuous plant dynamics. Tasks are used to simulate both periodic (time-driven) activities and aperiodic (event-driven) activities. Event-driven tasks are executed when network blocks such as the actuators, routers, and coordinator nodes receive a signal otherwise they do not execute. Only for the sensor nodes, the authors use Kernel block to measure the physical attributes periodically. More information and instructions related to TrueTime 2.0 Block Library are available in the guide [16].

B. Communication Protocol

A suitable communication protocol is essential to solve the integration problem of connecting heterogeneous devices with different brands. Authors reviewed many literature works to compare these different protocols regarding their advantages, functionality, and compatibility with TrueTime. The results are gathered in table 1 [17-20]. Based on this research, the ZigBee protocol is selected as the most suitable communication protocol for the wireless sensor and actuator networks in building automation and HVAC control applications.

The IEEE 802.15.4 standard has been extracted by the ZigBee Alliance for reliable and cost-effective wireless networked devices. The standard was integrated with the BACnet protocol with a specialty in HVAC and building automation around the year 2008. The main characteristics of this protocol are low power, low range, low data rate, and low complexity. As defined, they can be used in constrained, embedded environments, running on batteries [21]. The ZigBee technology features are low cost, easy implementation, reliability, low power, and high security [22]. The nodes (devices) are defined based on their role and capabilities [23]:

1. *Network coordinator*: a unique full-function device (FFD) is responsible for choosing key parameters of the network configuration and for starting the network. It also stores information about the network or acts as a bridge to the other networks if they are available. In this study, the network coordinator has been placed in the coordinator.
2. *Router*: a FFD device that supports the data routing functionality, including acting as an intermediate device to link different components of the network and forwarding messages between remote devices across multi-hop paths. A router can communicate with other routers and end devices. In this research, every room has been assumed as a cluster and the router plays the role of the cluster head.
3. *End devices*: a reduced-function device (RFD) that contains limited functionality to communicate with its parent node: the network coordinator or a router. The end nodes can go to the sleep mode because of their limited functionalities and profit from a long operating life. The end devices in this study are three sensors of CO_2 concentration, temperature, and occupancy and two actuators of damper and thermostat.

TABLE 1. WIRELESS COMMUNICATION TECHNOLOGIES

	ZigBee	*Bluetooth*	*Wi-Fi*	*WirelessHART*	*MiWi*	*Z-Wave*
Standard/Protocol	IEEE 802.15.4,ZigBee	IEEE 802.15.1	IEEE 802.11 (a-b-n-g)	IEEE 802.15.4	IEEE 802.15.4	Z-Wave
Network Topology	Star, Peer-to-Peer, and Mesh	Star, Peer-to-Peer	Star, Peer-to-Peer	Star, Peer-to-Peer, and Mesh	Star, Peer-to-Peer	Mesh
Power Consumption	Very Low	Medium	High	low	Low	Low power
Battery life (Days)	100 to +1000	1 to 10	0.5 to 5	Depends on Battery Specifications	Depends on Battery Specifications	Depends on Battery Specifications
Range (meters)	10–300	10	10–100	100	20–50	30 -100
Market Adoption	High	High	Extremely high	High	Medium	Medium
Network size (nodes)	64000	8	2007	100	1024	232
Application Areas	Demand Response, remote control and automation in residential and commercial buildings	Wireless connectivity between personal devices such as headphones, medical, sport & fitness, mobile phones or laptops	Wireless LAN connectivity, broadband Internet access	Industrial Control, building control the sensory data conveying temperature, pressure or speed	Industrial monitoring and control, home and building automation, remote control lighting control and automated meter reading	Remote control lighting and automation, control, security systems, thermostats, windows, locks, swimming pools
Advantages	Low Power consumption, several application profiles (home automation, smart energy) and topology flexibility	Speed and flexibility	Speed and flexibility	Communication Security, reliability and Environment with wired HART infrastructure	Flexible, cost-effective platform	Controllers and slaves network, flexible network configuration

C. Network Topology

A ZigBee wireless personal area network (PAN) supports three possible topologies: star topology, peer-to-peer (mesh) topology and cluster-tree topology. Star networks are suitable for simple requirements with low power consumption. Peer-to-peer (mesh) networks have the capability of higher reliability and provide various paths in the network and there is a routing algorithm. The cluster-tree-mesh topology which is used in this study actually utilizes a hybrid cluster-tree and peer-to-peer (mesh) topology, benefits both for a high level of reliability and support for battery-powered nodes with a minimum of routing [24]. Figure 1 describes our network topology based on the building architecture with wireless connections. The main steps of data transmission are:

- Sending the measured values of temperature/CO_2 from the sensor node to the cluster head of a room (i.e., the router)
- Receiving the measured values by the router from the sensor nodes and forward them to the coordinator and controller.
- The controller will receive the feedback values and determine the corresponding command which will be sent to the actuator node through the coordinator.
- The receiver actuator node will receive the command and applies it to the plant (heater/damper subsystem).

D. System Model Validation

Authors use the result of reference model simulator that was established in a wired sensor and actuator network to validate the system behavior in wireless network [3]. The simulated results of models in wireless configurations overlap the results in wired model reasonably at the same operational condition which describes a good validation. Figure 2 shows that there is a reasonable match between the outputs of the system which are blue temperature signals for the wired network and dashed pink line for the wireless network for room number one that confirms the validity of the deployed system model based on the ZigBee protocol.

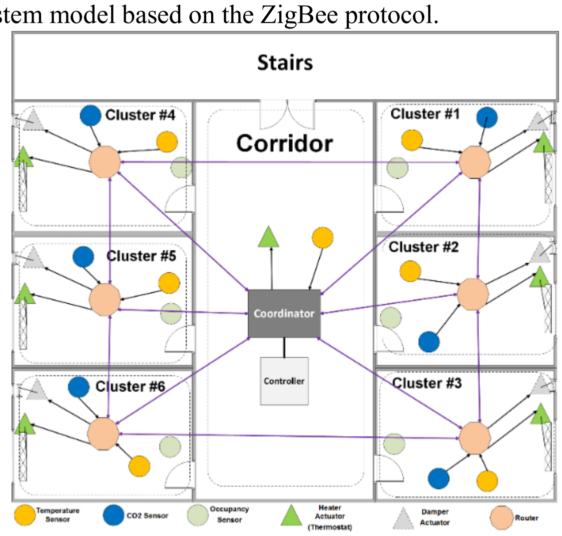

Fig. 1. Network Topology based on Building Architecture.

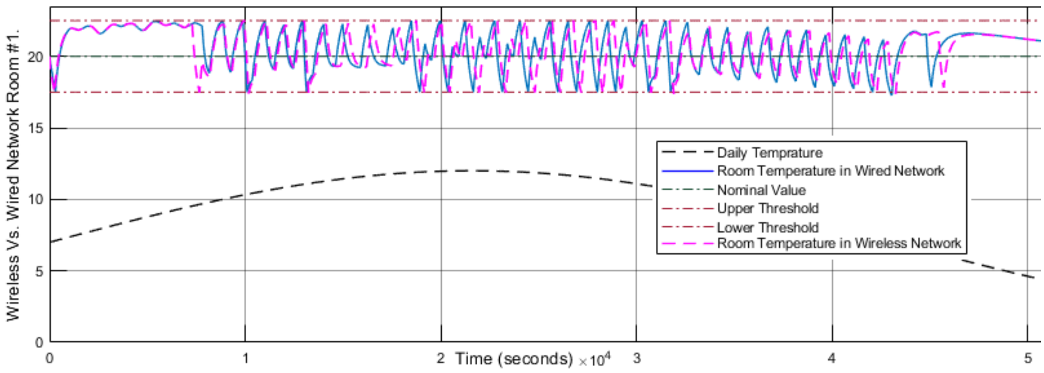

Fig. 2. System Model based on Wireless Network Validation.

III. FAULT INJECTION FRAMEWORK

A. Fault-Error-Failure Propagation

To assist in the understanding of the fault-error-failure relationship, figure 3 shows how an error propagates to one or more components, or even to one or more systems at a higher level. A fault is an unpermitted deviation of at least one characteristic property of the system from its normal, acceptable, usual and expected behavior that leads to an error [25]. An error corresponds to an incorrect (undefined) system state which may lead to a failure [26]. A failure is the (observable) manifestation of an error, which occurs when the system deviates from its specification and cannot deliver its intended functionality. When a fault is activated, it produces an error in a system component and the error can either propagate within the component, which is considered as internal propagation until it causes the component failure or it reaches the service interface of component A. Henceforth, the service is delivered by component A to component B becomes incorrect. Eventually, the service failure of component A appears as an external fault to component B and propagates as an error into the component B via its interface. This error can either propagate into component B and cause the failure in component B or propagate to another component. The failure of one or more components cause a permanent or transient fault in the system that consists of the components and it causes the system error. The system error either propagates into a system and causes the system failure or propagates to another system. The implemented framework of this study is a good example of this fault-error-failure relationship as when one type of fault is activated in one system, the behavior of the other system will change (from its normal behavior), however, no fault was originally activated in the second subsystem. Therefore, a very low sensor reading that is a fault in the damper subsystem causes the thermostat stuck at on-position in the heater subsystem.

B. Fault Injection Model

The user can activate/inject different types of faults using the graphical user interface (GUI) into the system model. Figure 4 indicates the developed GUI in this study. The faults are related to the components (devices) such as faults in the CO_2 concentration sensor, temperature sensor, damper actuator, and heater actuator (thermostat) or they especially target the wireless network communication. The established model has the capability to model lots of faults, but some are selected based on the references and modeled in this work [27-31]. For the data-centric faults, if x is the measured value of the sensors and the operation parameter for the actuators (e.g. damper position in fully opened with the value 1 or 0 for the fully closed position) and x' describes the faulty value of the parameters, offset faults, gain faults, stuck-at faults, and noise in network channel can be modeled using the following equation:

$$x' = Ax + B + n \quad (1)$$

where A is the gain value, B is the offset value, and n is noise value in network channel. Therefore, to inject the offset fault, parameters of A, B, and n must be inserted into the GUI. For activation of gain fault, A and n must be given. For the stuck-at fault which means the representative parameter of a device will get the same value over a period of time, just component B must get a value that shows that the sensor reports a faulty constant value. For the actuators, such as thermostat and ventilation damper, the stuck-at can be a value from 0 to 1 where 0 is the constantly closed position and 1 represents the constantly open position, and if the actuator is stuck-at the middle position, then it gets 0.5 which shows that it is 50% open.

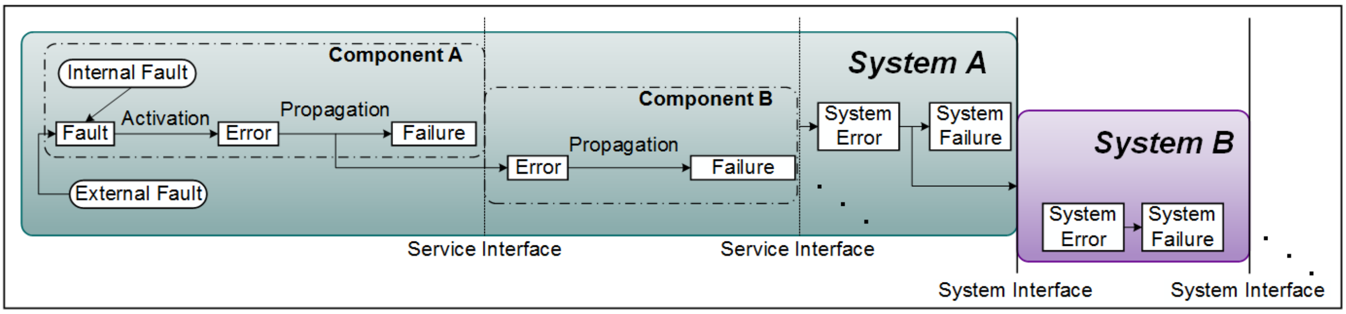

Fig. 3. Fault-Error-Failure Propagation Model in Component Level and System Level.

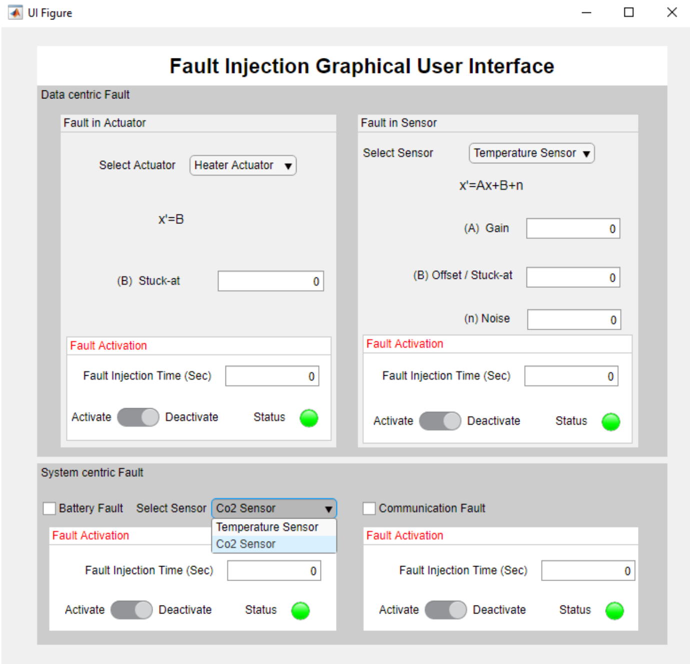

Fig. 4. The Fault Injection graphical User Interface (GUI).

For the system-centric faults, the user can also select the target device for the fault injection and then activate the battery fault which is due to battery depletion and can arise in the operational phase or a communication fault which is a routing fault. For the non-faulty mode (healthy mode) of router communication, the message will be sent to the coordinator over the shortest way. If this route is disconnected, then the message will be sent to the next nearest router in another cluster. In the next router, the message will be sent to the coordinator. If the communication fault get activated, therefore it sends back the message to the source, and this process will be repeated. Henceforth, the data will never get to the coordinator. Figure 6 describes the routing algorithm for the routers of each cluster (each room) which is developed by the authors. Figures 5 and 7 are the room and corridor models established in Simulink using TrueTime, respectively.

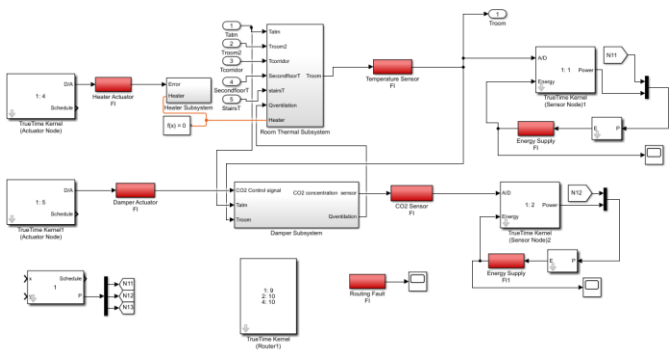

Fig. 5. Room Simulink Model.

Algorithm 1: Routing Algorithm (Router's Function)

```
Initialization the TureTime kernel block
Call the function of the router A
Receive the message from the sensor node or a router
  IF the routing fault is not activated
        IF the message is not empty THEN
            Forward the message to the coordinator
        ENDIF
  ELSEIF (when the route to the coordinator is disconnected)
        IF the message is not empty THEN
             Forward the message from the router A to the router B
             Call the function router B
        END IF
        IF (when the communication fault is activated)
             IF the message is not empty THEN
                  Send back the message to the source
             END IF
        END IF
  END IF
```

Fig. 6. Routing Algorithm.

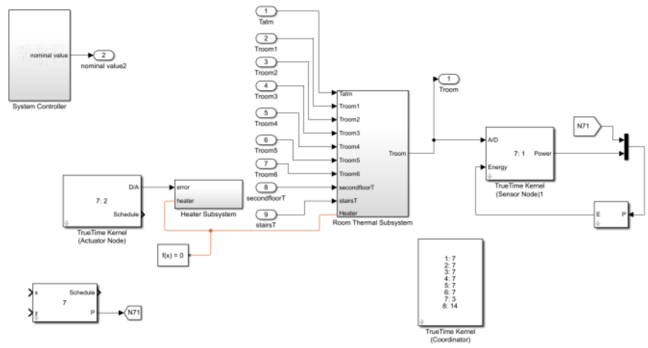

Fig. 7. Corridor Simulink Model.

IV. RESULTS

This section deals with some sample outputs of the control variables for the system which describes the capabilities of the established framework in MATLAB/Simulink to inject faults in the wireless sensor and actuator network based on DCV and heating system and system behavior monitoring. The samples are as the following. Temperature sensor is selected and the gain fault variable of A gets 3, for the offset fault B gets 5 and 10, for the stuck-at fault of damper actuator B gets 1 which means the damper is continuously open, and n is assumed a random number which describes the effect of noise due to interference or humans in network channel. The sensor battery fault and communication fault (routing loops fault) are described in the previous section. The double-Y axes figure 9 indicates some results for zone one (cluster one) as examples. The simulation describes the system behavior in one day which is 86,400 seconds (simulation time), but for the better view, the system behavior in first 50,000 seconds are reported. The figure 9(a) describes the behavior of the system in healthy (non-faulty) operation mode in dark blue color, and the rest (9(b) to 9(f)) are the faulty behaviors in different fault cases. The room temperature signal as the real room temperature in violet color and faulty sensor readings (in case of fault activation) in light green are also shown. In addition, the heater status and damper status in healthy (non-faulty) and faulty mode are available for comparison. The results indicate that the user can observe the severity of faults by inserting different values into the fault model (via designed GUI) which was described in equation 1 in this study. Also, this figure indicates that in some cases of fault injection, the behavior of the system will be changed however the signal value will be still in the desired operation range. For example, in figure 9(e) if the user put the value of 5 for the *B* parameter of equation 1, the temperature signal will differ from healthy mode, however the room temperature is mostly remained in the desired thresholds, and in figure 9(f) if the user put the value of 10, then the signal goes beyond the desired range. These difference will be used in the future study as a symptom for the failures prediction which shows a component e.g. a sensor can be failed in near future if the maintenance operator does not replace or repair it at the right time. Also, figure 8 indicates the designed heater subsystem and the duty cycle display of the heater that is proportional to the heating energy consumption. The calculation of duty cycle is for one day (simulation time). The result in this study shows that the heater duty cycle in room one is 43.87 % in healthy (non-faulty) mode. This value will change if any fault is activated and based on fault-error-failure propagation model in section III A., the arise of fault in one system can affect the other system. Two examples will be discussed for better understanding. If fault is activated in the battery of temperature sensor (see the figure 9(b) and compare it with the figure 9(a)), the results describe that the heater status will stuck-at ON, which is proportional to more duty cycle of 48.73% which describes more energy consumption (4.86% more). However, some faults cause discomfort of occupants instead of waste of energy such as activation of gain fault in temperature sensor (see the figure 9(d) and compare it to the figure 9(a)) which shows due to incorrect reporting of temperature which is very high, the heater will stuck-at OFF position as the control system assumes the room temperature is too high however in reality the temperature is not that high. As the result, the duty cycle block displays 19.57% which is very less in comparison of healthy mode (24.3% less) but the occupants experience a discomfort situation because the real room temperature signal in this figure is dropped lower than the comfort zone (desired range). Therefore, the scholars or companies can use this framework to find the optimum point of cost and quality trade-off for their products by monitoring and evaluating the fault effects on the system.

CONCLUSION

This study explicates two main achievements, the first achievement of this study is the simulation of the demand-controlled ventilation and heating system based on wireless sensor and actuator networks in MATLAB/Simulink. The second achievement in this study is an advanced fault injection framework via a designed graphical user interface to trace the behavior of the system in existence or absence of different kinds of faults and to understand the effect different types of faults in energy consumption and occupancy comfort based on a fault-error-failure propagation model. In this paper, the routing algorithm of the network is developed by the authors and the code is embedded in the Kernel block of TrueTime. In the end, the results show the successful implementation of the fault injection framework in MATLAB/Simulink.

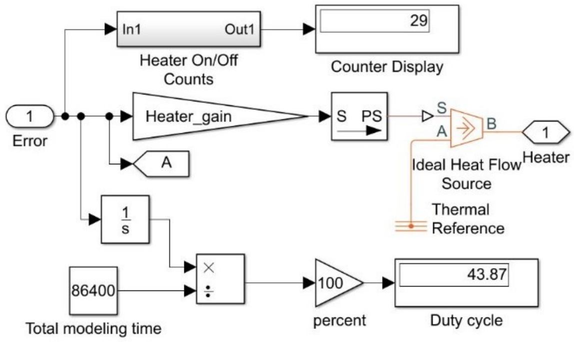

Fig. 8. Heater Subsystem.

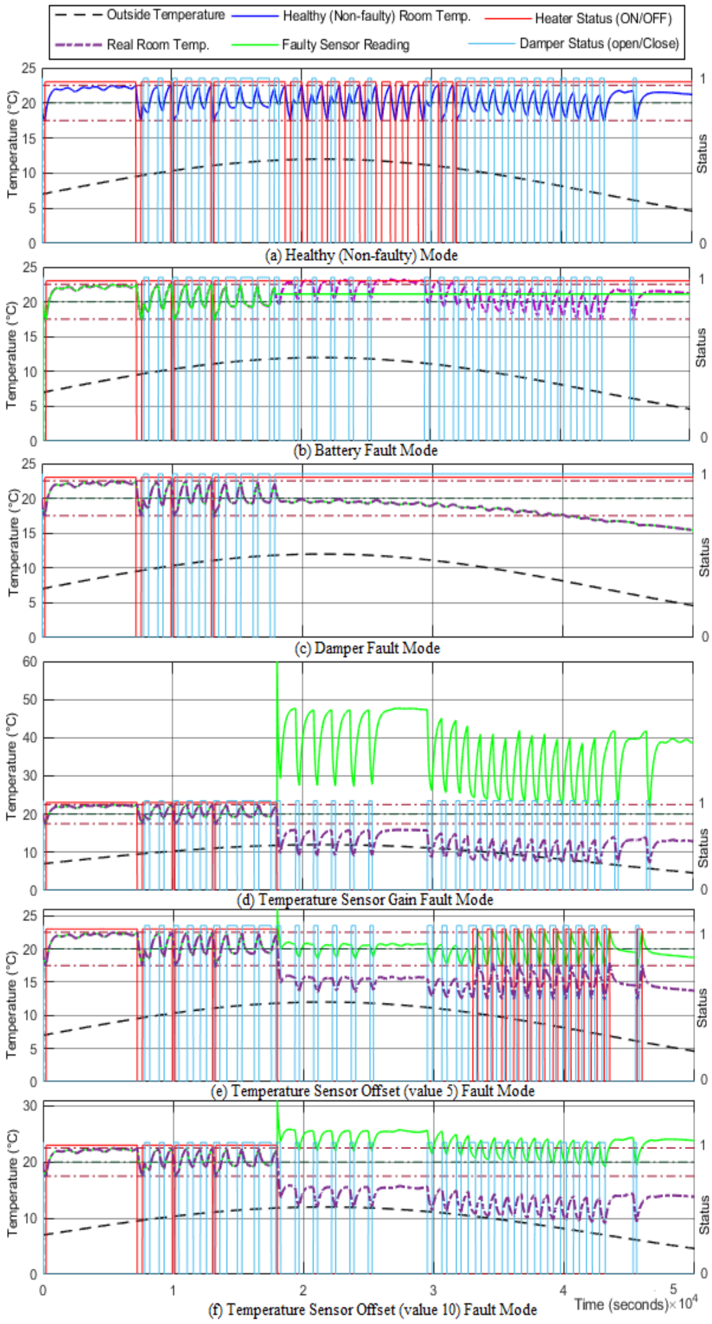

Fig. 9. Fault Injection Results.

ACKNOWLEDGMENT

This work was supported by the DFG research grant OB384/5-1.

REFERENCES

[1] EBPD, On the energy performance of buildings. Official Journal of the European Union, Directive 2010/31/EU of the European Parliament and of the council, 2010: pp. 13-34.

[2] N. Šajn, “Energy efficiency of buildings,” European parliamentary research service (EPRS), May 2016.

[3] A. Behravan, R. Obermaisser, A. Nasari, “Thermal dynamic modeling and simulation of a heating system for a multi-zone office building equipped with demand controlled ventilation using MATLAB/Simulink,” International conference on circuits, system and simulation, London, United Kingdom, July 2017, Conference proceeding pp. 103-108. IEEE.

[4] M. Ivanovich, D. Gustavson, “Future of intelligent buildings is now,” HPAC Heat Piping Air-Cond., May 1999, 71(5): pp. 73-79.

[5] Sullivan, F. Wireless sensors in building automation. 2005; Available from: http://www.frost.com.

[6] Wang, S., “Editorial Wireless Networks and Their Applications in Building Automation Systems,” HVAC&R Research, 2008, 14(4), p. 529-533.

[7] F. Xia, X. Kong, and Z. Xu, “Cyber-physical control over wireless sensor and actuator networks with packet loss,” in Wireless Networking Based Control, S. K. Mazumder, Ed. Springer New York, 2011, pp. 85–102.

[8] Y. Sadi, S. C. Ergen, and P. Park, “Minimum energy data transmission for wireless networked control systems,” IEEE Transactions on Wireless Communications, vol. 13, no. 4, pp. 2163-2175, 2014.

[9] A. W. Al-Dabbagh and T. Chen, “Design considerations for wireless networked control systems,” IEEE Transactions on Industrial Electronics, vol. 63, no. 9, pp. 5547-5557, 2016.

[10] S. P. Mohanty, “Nanoelectronic Mixed-Signal System Design,” New York: McGraw-Hill Education, 2015.

[11] M. L. Rajaram, “Comparative analysis and implementation of high data rate wireless sensor network simulation frameworks,” master’s thesis, Dept. Eng. Technol., Univ. North Texas, 2015.

[12] A. Pinto, et al. “Synthesis of embedded networks for building automation and control,” in American Control Conference, 2008. 2008. IEEE.

[13] A. Guinard, A. McGibney, and D. Pesch. “A wireless sensor network design tool to support building energy management,” in Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings. 2009. ACM.

[14] Z. Song, X. Zhou, “Research and Simulation of Wireless Sensor and Actuator Networked Control System,” 25th Chinese Control and Decision Conference (CCDC), 2013, pp. 3995-3998.

[15] G.H. Cibse, “Building Control Systems,” Butterworth-Heinemann, Oxford, 2000.

[16] A. Cervin, D. Henriksson, M. Ohlin, “TrueTime 2.0 beta 5 – Reference Manual,” Department of Automatic Control, Lund University, Sweden, June 2010.

[17] D.P. T. Mendes , R. Godina, E. M. G. Rodrigues, J. C. O. Matias,and J. P. S. Catalão, “Smart Home Communication Technologies and Applications: Wireless Protocol Assessment for Home Area Network Resources,” Energies 2015, 8, 7279-7311,ISSN 1996-1073.

[18] T. Obaid, H.Rashed, A.A.Elnour, M.Rehan, M.M. Saleh, and M.Tarique, “ZIGBEE TECHNOLOGY AND ITS APPLICATION IN WIRELESS HOME AUTOMATION SYSTEMS: A SURVEY,” International Journal of Computer Networks & Communications (IJCNC) Vol.6, No.4, July 2014.

[19] M. Kuzlu, M. Pipattanasomporn, and S. Rahman, “Review of Communication Technologies for Smart Homes/Building Applications,” 2015 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA), Bangkok, Thailand Nov, 2015.

[20] E. Irmak, A.Köse, G. Göçmen, “Simulation and ZigBee based wireless monitoring of the amount of consumed energy at smart homes,” 2016 IEEE International Conference on Renewable Energy Research and Applications (ICRERA), Nov 2016, Birmingham, UK.

[21] V. Moravcevic, M. Tucic, R. Pavlovic, A. Majdak, “An approach for uniform representation and control of ZigBee devices in home automation software,” In 2015 IEEE 5th International Conference on Consumer Electronics-Berlin (ICCEBerlin)(pp. 237–239), IEEE.

[22] T. Obaid, H. Rashed, A. Abou-Elnour, M. Rehan, M. Muhammad Saleh, M. Tarique, “ZIGBEE TECHNOLOGY AND ITS APPLICATION IN WIRELESS HOME AUTOMATION SYSTEMS: A SURVEY,” International Journal of Computer Networks and Communications (IJCNC), Vol. 6, No. 4, July 2014.

[23] K. Sohraby, D. Minoli, T. Znati, “Wireless Sensor Networks,” A John Wiley and Sons Inc., 2007, p. 179.

[24] W. C. Craig, “ZigBee: Wireless Control That Simply Works,” Program Manager Wireless Communication, ZMD America, Inc.

[25] A. Behravan, R. Obermaisser, D. Hanike Basavegowda, S. Meckel, “Automatic Model-Based Fault Detection and Diagnosis Using Diagnostic Directed Acyclic Graph for a Demand-Controlled Ventilation and Heating System in Simulink,” The 12th Annual IEEE International Systems Conference (SYSCON 2018), Vancouver, Canada, 2018.

[26] A. S. Tanenbaum, M. van Stehen, “Distributed Systems: Principles and Paradigms,” Prentice Hall, 2002.

[27] S. Zidi, T. Moulahi, B. Alaya, “Fault Detection in Wireless Sensor Networks Through SVM Classifier,” IEEE SENSORS JOURNAL, VOL. 18, NO. 1, JANUARY 2018.

[28] D. Raposo, A. Rodrigues, J. Sa´ Silva, F. Boavida, “A Taxonomy of Faults for Wireless Sensor Networks,” Journal Network System Management, Vol. 25, 2017, pp. 591–611.

[29] A. Avizienis, J.C. Laprie, B. Randell, C. Landwehr, “Basic Concepts and Taxonomy of Dependable and Secure Computing,” IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, Vol. 1, No. 1, 2004, pp. 11-33.

[30] U. Hengartner, S. Moon, R. Mortier, C. Diot, “Detection and Analysis of Routing Loops in Packet Traces,” Proc. of 2nd Internet Measurement Workshop (IMW 2002), Marseille, France, November 2002, pp. 107-112.

[31] H. Balakrishnan, “Network Routing – II Failures, Recovery, and Change,” Lecture 20.