Hierarchical Formal Modeling of Internet of Things System Oriented to User Behavior

Lei Yu[1,2,*], Yang Lu[3], Benhong Zhang[3], Lei Shi[3], Fangliang Huang[1], Ya Li[1], Yulian Shen[4]
(1 School of Medical Information Technology, Anhui University of Chinese Medicine, Hefei 230012, China
2 Institute of Computer Application in Traditional Chinese Medicine, Anhui Academy of Chinese Medicine, Hefei 230012, China
3 School of Computer and Information, Hefei University of Technology, Hefei 230601, China
4 First Affiliated Hospital, Anhui University of Chinese Medicine, Hefei 230038, China)
**Corresponding author: fishstonehfut1006@163.com*

***Abstract*—Ensuring the correctness and reliability of the Internet of Things system is the key to the advancement of the Internet of Things project. It is very necessary to fully inspect the Internet of Things system before it is actually deployed, so as to find the errors and defects in the system design as soon as possible and make improvements. Compared with conventional simulation and testing, the formal method has the advantages of low cost, short cycle and simple steps, which provides efficient support for the inspection and analysis of the Internet of Things system before deployment. Based on the stateful timed communication sequence process (STCSP), we consider the formal modeling framework for the Internet of things system from the perspective of external environment input and system architecture. We then propose a hierarchical formal modeling method for the Internet of things system oriented to user behavior. Taking the elderly home monitoring application scene as an example, as the input of the external environment, the user behavior and its implementation object are combined into a whole for modeling, so as to keep the two states in sync, restrict each other, and avoid unrealistic sequence of activities. From the perspectives of perception mode, communication mode, predefined rules and application services, we have completed the hierarchical modeling of the three-layer architecture of the Internet of Things system, that is, perception layer, middle layer and application layer. Finally, the model verification tool PAT analyzes and verifies the above model from the aspects of security, accessibility, and system consistency. This method provides scientific basis for the correctness inspection and reliability analysis of the Internet of Things system before deployment in the Internet of Things project.**

Keywords—Internet of things system; Formal modeling; User behavior; STCSP; PAT; Home monitoring for the elderly

I. Introduction

The Internet of Things system is a complex system, and ensuring its correctness and reliability is the key to the advancement of the Internet of Things project. Therefore, it is very necessary to fully inspect the system before it is actually deployed, so as to find the errors and defects in the system design as soon as possible and make improvements [1]. At present, the common methods for system inspection include software testing, simulation and formal methods [2]. Compared with the first two methods, formal methods have the advantages of low cost, short cycle and simple steps. As an efficient means of pre-deployment inspection of Internet of Things systems, formal methods have received widespread attention in recent years.

For the formal research of the Internet of Things system, most of the existing documents focus on the application service of the Internet of Things. The modeling of the Internet of Things service is completed from the aspects of service functions [3], service resources [4-5], service business processes [6] and service interaction [7] through hybrid system theory, process algebra, Petri net, UML, finite automata and other methods. The hierarchical formal modeling of the Internet of Things system is not carried out from the perspective of system design, starting from the three-layer general architecture of the Internet of Things, that is, perception layer, network layer and application layer. From the perspective of architecture, services belong to the application layer, so the existing literature focuses on the application layer, pays less attention to the perception layer and the network layer, and has not yet involved the perception mode and communication mode of the Internet of Things system, which is exactly the important description information of the Internet of Things system, and it is very necessary to integrate into the formal modeling of the Internet of Things system.

In addition, the modeling of the Internet of Things system in the existing literature is mostly based on the perception of the physical environment, such as real-time perception of indoor air temperature, air humidity, light brightness, object position [4-5,8-9], etc., and does not take more account of the dynamic behavior of users. In fact, most of the Internet of Things application systems are user-centered, and Internet of Things services often interact with users, for example, daily household behaviors of the elderly living alone such as going to the toilet and bathing, learning behaviors of minor children such as reading books and listening to online classes, and treatment behaviors of hospitalized patients such as bed rest and infusion. The user behaviors in these scenes need more attention, such as the long toilet time for the elderly living alone, the long study time for minor children, and the long bed time for hospitalized patients, which may have hidden dangers and require real-time monitoring and timely warning. Therefore, the modeling of user behaviors is very important in the formal research of Internet of Things system. However, in the existing literature, the modeling of the Internet of Things system oriented to the above user behaviors is rarely

978-1-7281-6514-1/20/$31.00 ©2020 IEEE
DOI 10.1109/SmartIoT49966.2020.00050

involved. It is more based on the acquisition of relevant data information by some wearable devices and then the extraction of some valuable characteristic values through digital signal processing methods, thus realizing the judgment of user behaviors such as falling, walking slowly, running, etc. [10-12]. However, this is only the perception of user behavior, and it is not integrated into the Internet of Things system, so it does not complete the modeling of user behavior from the perspective of the system. If the behavior is taken as the external input of the system, the behavior information such as the occurrence place, sequence, duration, etc. need to be described in the specific modeling of the behavior, instead of merely sensing the behavior.

For this reason, based on the stateful timed communication sequence process (STCSP), we consider the modeling method for the Internet of things system from the perspective of external environment input and system architecture. We then propose a hierarchical modeling framework for the Internet of things system oriented to user behavior, and take the elderly home monitoring application scenario as an example to give a specific modeling process. Then, the model is analyzed and verified by the model checking tool PAT, which provides scientific basis for the correctness check and reliability analysis of Internet of Things system before deployment.

The rest of this paper is organized as follows. Section II proposes a hierarchical modeling framework for Internet of Things system oriented to user behavior, from the aspects of environmental input and system architecture. In Section III, taking the elderly home monitoring application scenario as an example, the specific modeling method is given. Section IV analyzes and verifies the correctness and reliability of the above modeling method based on the PAT platform. Section V summarizes the full text and discusses the next research work.

II. Hierarchical Modeling Framework for Internet of Things System Oriented to User Behavior

A. *Specific Application Scene of Internet of Things System*

At present, China has entered an aging society, with an increasing number of empty-nest elderly. The health and safety of the elderly at home have attracted extensive attention from all walks of life. The Internet of Things provides technical support for home monitoring with its advantages of real-time sensing and wireless transmission. Taking the application scenario of the Internet of Things for home monitoring of the elderly as an example, the application requirements of the scenario are as follows: monitoring the indoor gas concentration, smoke concentration, formaldehyde concentration and other home environmental indicators through a gas sensor, once exceeding a set threshold, alarming in time to prevent accidents; Monitoring the body temperature, blood pressure, heart rate and other physiological indexes of the elderly family members through wearable devices, and informing the elderly family members, their guardians and community family doctors in time if they are not within the normal threshold range; Real-time monitoring of bathing behavior, toilet behavior and sleep behavior of elderly people at home, once the duration exceeds the set normal threshold, they are all unsafe factors and must be reminded in time to prevent accidents. From the above, it can be seen that the most important thing is to monitor the daily life of the elderly at home. This paper takes this as an example to focus on the formal modeling of Internet of Things system oriented to user behavior.

B. *Modeling Framework*

According to the above scenario, when formally modeling this kind of Internet of Things system, the user behavior should be modeled first, and the user behavior can be regarded as the external environment input of the system. At the same time, the application layer of this kind of system is mainly based on early warning and reminding, with high real-time requirements. Therefore, attention should be paid not only to the modeling of the application layer, but also to the perception of user behavior and the judgment of warning behavior, that is, to model the system from the perspective of the overall system architecture. In the following, taking the above application scenario as an example, the formal modeling framework of this kind of internet of things system is given from the aspects of environmental input and system architecture respectively, as shown in Fig. 1.

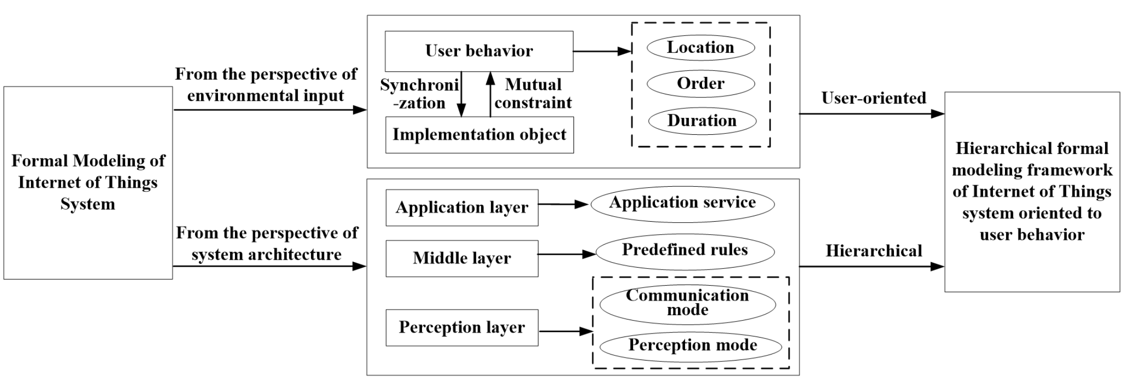

Fig. 1. User behavior oriented hierarchical formal modeling framework for internet of things system

1) Environmental input modeling

The modeling of environmental input mainly refers to the modeling of Internet of Things system from the perspective of user behavior input. The environment emphasized in reference [8-9] is mostly static physical environment, while the emphasis here is on input environment, that is, external user behavior. User behavior does not exist in isolation, and it has its implementation objects. Therefore, it is necessary to integrate the user behavior and its implementation objects into a whole for modeling, so that user behavior can keep synchronization with the state of its implementation objects. At the same time, modeling the implementation object can also restrict user behavior and avoid unrealistic sequence of actions in the model. The occurrence of user behavior is described in terms of location, sequence, duration, etc., all of which must be reflected in the modeling information.

2) System architecture modeling

The modeling of the system architecture is to model the Internet of Things system from the perspective of system design, starting from three levels of the Internet of Things system, that is perception layer, middle layer and application layer. The perception layer modeling is mainly to model the various sensors in the perception layer, including perception mode and communication mode. How the sensor senses information, whether it is periodic perception or event-driven perception, how the sensor transmits information, whether it is synchronous communication or asynchronous communication, these need to be reflected in the modeling information. The middle layer modeling is mainly to model the predefined rules of the system. For example, under what circumstances the user behavior will be given an early warning, this needs to be based on certain rules and meet certain conditions, then these rules and conditions must be modeled. Application layer modeling is to model various application services of the system, such as early warning service, reminder service, etc.

III. Hierarchical Modeling Method for Internet of Things System Oriented to User Behavior

CSP [13] (Communication Sequential Processes, CSP) is a communication sequential process method in process algebra, which is used to describe the specification and design of distributed concurrent software systems. The Internet of Things system is a distributed concurrent system, so CSP can be used to complete the formal modeling of the Internet of Things system. TCSP (Timed CSP, TCSP) is a formal language formed by adding time-related operations on the basis of CSP, while STCSP (Stateful Timed CSP, STCSP) is an extension of TCSP and supports formal modeling of hierarchical real-time systems [14]. Therefore, this paper completes hierarchical formal modeling of Internet of Things systems through STCSP. Due to limited space, the relevant syntax of STCSP will not be repeated here. According to the modeling framework in Section 2 above, a specific method for hierarchical modeling of Internet of Things system oriented to user behavior is given from the aspects of environmental input and system architecture.

A. *Environmental Input Modeling*

The modeling of environmental input is completed from two aspects of user behavior and its implementation objects.

1) Modeling user behavior

User behaviors are diverse and often unpredictable, so we only need to focus on user behaviors related to system requirements and model them, and the rest can be ignored. Taking the Internet of Things system for home monitoring of the elderly as an example, it is necessary to pay special attention to the activities of users entering and leaving the room, while the behaviors of users opening and closing the door can be ignored.

The specific process of modeling user behavior is as follows:

Step 1: Analyze the user behaviors related to the system requirements and abstract each user behavior into an event.

Step 2: Classify all behaviors according to the location where the behaviors occur, and model each class as a different process, where specific behaviors are modeled as events in the process, and associate different processes by adding location transfer in the model.

For user behaviors that may occur in any location, such as user fainting, this type of behavior can be added to the user behavior processes in each location respectively, or it can be individually packaged into a process, and appropriate operations can be selected to compound the process with the user behavior processes in different locations before.

Step 3: Combine these processes to get a model of user behavior.

Special attention is paid to the fact that all possible activity sequences of user behaviors can be generated by adopting model checking technology for the built model. When modeling a single process, it is necessary to consider that the sequence of user behaviors in the system is uncertain. When modeling, appropriate operations should be selected to appropriately combine the abstracted events so as to specify all possible event sequences.

Taking the Internet of Things system for home monitoring of the elderly as an example, there are three places where user behavior occurs, namely washroom, bedroom and outdoor, and each place establishes a process. After entering the washroom, you can go to the toilet, open the shower to take a bath, or perform other activities, and the STCSP model of the washroom scene in the system is shown in (1).

$$\begin{aligned} \text{User\_WashRoom()} = {} & \text{sitOnToilet} \rightarrow \text{User\_WashRoom()} \\ & [\,]\ \text{standUp} \rightarrow \text{User\_WashRoom()} \\ & [\,]\ \text{turnOnShowerTap} \rightarrow \text{User\_WashRoom()} \\ & [\,]\ \text{turnOffShowerTap} \rightarrow \text{User\_WashRoom()} \\ & [\,]\ \text{wandering} \rightarrow \text{User\_WashRoom()} \\ & [\,]\ \text{exitWashRoom} \rightarrow \text{User\_Bedroom()}; \end{aligned} \quad (1)$$

This model encapsulates all the user's events in the washroom into the process $User\_WashRoom$, which randomly selects one event to execute. For example, select to execute event $sitOnToilet$ first and then execute event $standUp$. In addition, location transfer is added to the model, that is, if event $exitWashRoom$ occurs, process $User\_Bedroom$ will control the occurrence of subsequent events, and then only the activities in the bedroom can be executed, so that the unrealistic sequence of activities like $exitShowerRoom \rightarrow turnOnTap$ will not occur.

Similarly, we can get the STCSP model $User\_Bedroom$ and $User\_Outside$ for bedroom scenes and outdoor scenes. Since user behaviors do not occur at the same time and one of them is selected to be executed at a certain moment, so the operator "[]" is used to compound to obtain the user behavior model, as shown in (2):

$$UserBehav() = User\_WashRoom()\,[]\,User\_BedRoom()\,[]\,User\_Outside(); \quad (2)$$

2) Modeling user behavior implementation objects

By modeling the implementation object of user behavior, user behavior can be constrained, so that user behavior can keep synchronization with the state of the implementation object, and unrealistic sequence of activities in the model can be avoided.

Taking the Internet of Things system for home monitoring of the elderly as an example, in the washroom scene of the system, if several events $sitOnToilet$ in the process $User\_WashRoom$ are continuous executed, it is not in line with the reality of life, so the toilet needs to be modeled in order to constrain the user behavior. The toilet model is shown in (3):

$$Toilet() = sitOnToilet \rightarrow standUp \rightarrow sitOnToilet(); \quad (3)$$

In the above process $Toilet$, if event $sitOnToilet$ occurs, then event $sitOnToilet$ cannot occur until event $standUp$ occurs.

Similarly, the shower faucet needs to be modeled. The shower faucet model is shown in (4):

$$ShowerTap() = turnOnShowerTap \rightarrow turnOffShowerTap \rightarrow ShowerTap(); \quad (4)$$

User behavior needs to be synchronized with the state of its implementation object. The essence of constraint modeling is to model concurrent behavior, which can be accomplished by the concurrent compounding of the user behavior process and its implementation object entity through the concurrent operator '||' in STCSP. For the elderly home monitoring Internet of Things system, the user behavior process $UserBehav$ and the implementation object entity $Toilet$ and $ShowerTap$ are compounded concurrently through the operator '||' to obtain the environmental input model $SystemEnvIp$, as shown in (5).

$$SystemEnvIp() = UserBehav() \,||\, Toilet() \,||\, ShowerTap(); \quad (5)$$

In this model, the same event $turnOnShowerTap$ of process $ShowerTap$ and process $UserBehav$ are executed concurrently, the same event $sitOnToilet$ of process $Toilet$ and process $UserBehav$ are executed concurrently, and other different events are executed separately, which avoids the continuous occurrence of events $turnOnShowerTap$ and $sitOnToilet$.

B. System Architecture Modeling

The system architecture of the Internet of Things is modeled from the perception layer, the middle layer and the application layer respectively.

1) Perception layer modeling

Modeling the sensing layer is mainly to model the sensor. The sensor sends the sensed information to the middle layer by monitoring the changes of the environment. Each sensor is modeled as a process, and then all sensor processes are combined to obtain a complete perception layer model. The modeling of a single sensor mainly starts from the sensor's perception mode and communication mode.

a) Modeling of perception mode

The perception modes of sensors include periodic sensing, event-driven sensing, mixed sensing and other modes, which can be selected according to the actual needs of the system. Periodic perception mode is to collect relevant information at regular intervals and describe it through the time characteristics of STCSP. Event-driven perception mode means that the sensor can sense relevant context information only after an event occurs, and the if-else process in STCSP is used for modeling. Mixed perception mode refers to the perception mode that integrates the first two modes. The modeling of mixed perception mode can be completed by modeling periodic perception mode and event-driven perception mode respectively and then combining the two models.

Taking the Internet of Things system for home monitoring of the elderly as an example, it is obtained by periodically sensing the water flow through the vibration sensor VSIR on the shower head to determine whether someone is taking a bath in the washroom. The VSIR model of the vibration sensor on the shower head is shown in (6).

$$\begin{aligned} VSIR\_WR() = (&turnOnShowerTap\text{-}>> port!VSIR\_WR.ON\text{-}>> Skip \\ &[]turnOffShowerTap\text{-}>> port!VSIR\_WR.OFF\text{-}>> Skip); \\ &Wait[15]; VSIR\_WR(); \end{aligned} \quad (6)$$

Wait[t] indicates that the system behavior is delayed by t time units before execution. VSIR_WR, ON and OFF are integer constants, where VSIR_WR is the ID of the vibration sensor, and ON and OFF are the possible states of the sensor. The above model indicates that after the vibration sensor sends the sensed information to the middle layer, it will continue to receive the sensed data after being idle for another 5 time units.

b) Modeling of communication mode

Sensors have different communication modes such as synchronous communication or asynchronous message transmission. In STCSP, Channel can be used to declare synchronous channel or asynchronous channel to model synchronous communication and asynchronous communication respectively.

Taking the Internet of Things system for monitoring the elderly home as an example, the presence of someone in the washroom to use toilet is sensed by the pressure sensor PSIR on the toilet. The model obtained by modeling its synchronous communication is as shown in (7):

$$\begin{aligned}&\text{Channel port 0;}\\&\text{PSIR\_WR()} = (\text{sitOnToilet-} >> \text{port!PSIR\_WR.SITTING -} >> \text{Skip}\\&\qquad [\,]\,\text{standUp-} >> \text{port!PSIR\_WR.EMPTY-} >> \text{Skip});\\&\qquad \text{PSIR\_WR();}\end{aligned} \quad (7)$$

Port is a synchronous channel, through which the pressure sensor and the middle layer communicate synchronously. The operator "-> >" indicates that the event on the left is an emergency and must occur as soon as possible, and the event cannot be cross-executed with other events. PSIR_WR, SITTING and EMPTY are integer constants, where PSIR_WR represents the ID of the pressure sensor, SITTING and EMPTY are the possible states of the sensor. In the above model, when someone enters the washroom to use toilet, the PSIR sensor sends a message SITTING to the middle layer; When no one is using toilet in the washroom, the PSIR sensor sends a message EMPTY to the middle layer.

c) Perception layer model

After modeling a single sensor in the system, a complete sensing layer model can be obtained by selecting appropriate logic to compound all sensor processes according to the working relationship among the sensors.

Taking the Internet of Things system for home monitoring of the elderly as an example, besides the vibration sensor installed on the shower faucet and the pressure sensor installed on the toilet in the washroom, there is also an RFID reader deployed at the door of the washroom to identify the user entering the washroom, and the passive pyroelectric infrared sensors installed in the washroom to detect whether there are people in the washroom. The sensor processes are VSIR_WR (), PSIR_WR (), RFID_WR () and PIR_WR (). Because each sensor works independently in this system, it is compounded with '|||', and the perception layer model is shown in (8):

$$\text{SLayer()} = \text{RFID\_WR()} \;|||\; \text{PIR\_WR()} \;|||\; \text{VSIR\_WR()} \;|||\; \text{PSIR\_WR()} \quad (8)$$

2) Middle layer modeling

After receiving the message from the perception layer, the middle layer analyzes and processes the context information according to the predefined rules, makes corresponding decisions based on the analysis, and sends the decisions to the application layer for execution. The conditions of the predefined rules are propositions about context variables, which can be modeled by daemons or if-else processes. The modeling of context variables can be achieved through shared variables in STCSP.

Taking the Internet of Things system for home monitoring of the elderly as an example, for example, STCSP is used to model the restriction rule that "users should not use toilet for more than half an hour", as shown in (9). It means that if the user spends more than 30 minutes in the toilet, the message Act.SitLong is sent to the application layer through the channel res.

$$\begin{aligned}&\text{Rule2()} = \text{if(sensors[PSIR\_WR]} == \text{SITTING}\\&\qquad \&\&\ \text{Duration[PSIR\_WR]} > 1800)\\&\qquad \text{res!Act.SitLong} \rightarrow \text{Skip;}\end{aligned} \quad (9)$$

In a specific system, there are often more than one predefined rule in the middle layer. Each predefined rule is modeled as a process, and then all the rules are compounded by appropriate compounding methods to obtain the overall rule model.

For example, there are 4 rules in the Internet of Things system for home monitoring of the elderly, namely Rule1, Rule2, Rule3 and Rule4, and the sequential compounding method is used to compound each rule process, then the overall rule modeling is shown in (10).

$$\text{AllRules()} = \text{Rule1(); Rule2(); Rule3(); Rule4();} \quad (10)$$

3) Application layer modeling

The application layer makes corresponding decisions according to the messages sent by the middle layer. When modeling the application layer, we should focus on the behaviors that affect users.

Taking the Internet of Things system for home monitoring of the elderly as an example, the specific decision made by the application layer is early warning and reminder service. The specific modeling is shown in (11).

$$\begin{aligned}&\text{ALayer()} = \text{res?status.rid} \rightarrow (\,[\text{status} == \text{Act}]\,\text{ActivateReminder(rid)}\\&\qquad [\,]\,[\text{status} == \text{Deact}]\,\text{DeactReminder(rid)}\,);\ \text{ALayer();}\end{aligned} \quad (11)$$

The application layer receives a two-tuple $(\text{status}, \text{rid})$ from the middle layer through the channel Res, where status represents a command and rid is the ID of the reminder. The application layer parses this and performs subsequent operations according to the value of status. If the received status is Act command, the system calls the ActivateReminder(rid) process to activate the reminder rid to the user. On the contrary, if the received status is Deact command, the system calls the DeactReminder(rid) process, and if the reminder rid is in the active state, the reminder rid cancels the reminder.

4) System architecture model

A complete system architecture model can be obtained by compounding the models of each layer. There are three kinds of cooperation relations between layers: sequential relation, mixed relation and concurrent relation. The sequential relation means that the components are executed sequentially in sequence. Mixed relation means that the execution of each component is independent of each other and is not affected by each other. Concurrent relation means that the same behaviors among components are executed synchronously, while other behaviors are executed independently. In STCSP, the operators ";", "|||" and "||" are used to model these three relationships respectively.

Taking the Internet of Things system for home monitoring of the elderly as an example, the cooperation relationship between the various layers is executed sequentially. Firstly, the sensor transmits the sensed data to the middle layer, then the middle layer analyzes and processes the data according to predefined rules to form a decision and transmits the decision to the application layer, and finally the application layer executes the decision. Its system architecture model is shown in (12).

$$\mathrm{SysArch}() = \mathrm{SLayer}();\mathrm{MLayer}();\mathrm{ALayer}(); \tag{12}$$

C. Complete System Model

By compounding the environmental input model and the system architecture model, a complete system model is obtained. There are the same events in the environmental input model and the system architecture model, and these same events are executed concurrently, so they are compounded concurrently using the operator '||' to obtain the system model System (), as shown in (13). In (13), SystemEnvIp is the environmental input model and SystemArch is the system architecture model.

$$\mathrm{System}() = \mathrm{SysEnvIp}() \,||\, \mathrm{SysArch}(); \tag{13}$$

IV. Analysis and Verification of Internet of Things System Based on PAT

PAT [15-17] is an effective tool for concurrent real-time system analysis and verification, which supports formal modeling and automated verification based on STCSP. The following takes the Internet of Things system of elderly home monitoring as an example and gives detailed verification process of the Internet of Things system based on the PAT platform.

For the Internet of Things system of elderly home monitoring, the verification is mainly carried out from three aspects of safety, accessibility and consistency, as follows:

a) Security (no deadlock)

The deadlock-free verification model of the system model System is shown in P1.

$$\#\mathrm{assert\ System()\ deadlockfree}; \tag{P1}$$

b) Accessibility

The key is to verify the accessibility of the reminder service. The system must ensure that there are two reminding services delivered to the user: One is to remind the user to end the bath as soon as possible when the user's bath time exceeds the set value; The other is to remind the user to end the toilet as soon as possible when the user's toilet time exceeds the set value. The models of these two attributes are shown in P2.1 and P2.2 respectively.

$$\#\mathrm{assert\ System()} \mid= [](\mathrm{ShowerLong} \rightarrow <> \mathrm{StopShowerReminder}); \tag{P2.1}$$

$$\#\mathrm{assert\ System()} \mid= [](\mathrm{SitToiletLong} \rightarrow <> \mathrm{StandToiletReminder}); \tag{P2.2}$$

c) System consistency

Check whether the following inconsistencies occur in the system: the user left the washroom, but the user location information in the system still shows that the user is in the washroom. The conflict state is defined to check whether this inconsistency will occur in the system, and the model is shown in P3.

$$\begin{aligned}&\#\mathrm{define\ Contradiction\ (userLocation == WashRoom}\\&\qquad\qquad \&\&\ \mathrm{Sensors[PIR\_WR] == EMPTY\ );}\\&\#\mathrm{assert\ System()\ reaches\ Contradiction;}\end{aligned} \tag{P3}$$

Through the PAT platform, the satisfiability verification of the above properties is carried out. The verification results are shown in Table I:

TABLE I. Verification Results of Home Monitoring Internet of Things System for the Elderly

Nature	Validation results	Whether it meets expectations
P1	True	Yes
P2.1	False	No
P2.2	False	No
P3	True	No

The specific analysis is as follows:

P1: The verification result is True, which meets the expectations, indicating that the system is safe without deadlock.

P2.1 and P2.2: The verification results are both False, neither of which meets the expectations. It indicates that the system cannot deliver these two reminding services accurately. The reason for this is that the system cannot accurately monitor the user's location.

P3: The verification result is True, which does not meet the expectations. The user's location information in the system is inconsistent with the actual location, which indicates that the system cannot accurately monitor the user's location.

From the above analysis, it can be seen that the main problem now is that the system cannot accurately monitor the user's position. For this reason, the deployment of sensors has been added to the system. Originally only one infrared sensor was installed at the washroom entrance, and now two sensors are installed to more accurately identify that the user has entered the washroom.

After improvement, the above properties are verified, and the verification results are shown in Table II. It can be seen from this that the improved system meets the user's expectation and meets the predetermined requirements, indicating that all the system defects discovered above have been solved.

TABLE II. VERIFICATION RESULTS OF IMPROVED HOME MONITORING INTERNET OF THINGS SYSTEM FOR THE ELDERLY

Nature	Validation results	Whether it meets expectations
P1	True	Yes
P2.1	True	Yes
P2.2	True	Yes
P3	False	Yes

V. CONCLUSION

Formal modeling and analysis of the Internet of Things system is a key issue in the advancement of the Internet of Things project. Based on the stateful timed communication sequence process STCSP, we consider the formal modeling method for the Internet of things system from the perspective of external environment input and system architecture. We then propose a hierarchical formal modeling framework for the Internet of Things system oriented to user behavior. Taking the elderly home monitoring Internet of Things application scenario as an example, the specific process of modeling is established, and then the model checking tool PAT is used to analyze and verify from the aspects of security, accessibility, system consistency, etc. respectively. This modeling method provides scientific basis for the correctness test and reliability analysis of Internet of Things system before deployment.

However, the modeling method proposed in this paper still has the following shortcomings. First, when modeling the sensors in the sensing layer, the individual sensors are modeled separately and then combined, and part of the work is repeated, especially when the system is large in scale. It is inefficient, so you can consider componentizing the sensor model to enhance the ease of use of the model and facilitate the formal modeling of the system more efficiently. The second is that the modeling method only considers the single-user situation, and in practical applications, most Internet of Things systems are multi-user situations. In addition, various sensors introduced for real-time sensing of user behaviors also bring the risks of information exposure. Therefore, the analysis and discussion of multi-user situation and privacy in the formal modeling of Internet of Things systems are also the focus of subsequent considerations.

ACKNOWLEDGMENT

This study was financially supported by the National Natural Science Foundation of China (Grant No.61701005), the Key Project of Outstanding Young Talents Support Program of Anhui Higher Education Institutions (Grant No.gxyqZD2016128), the Key Project of Natural Science Research in Anhui Higher Education Institutions (Grant No.KJ2015A054, No.KJ2019A0437), the Key Project of Humanities and Social Sciences Research in Anhui Higher Education Institutions (Grant No.SK2019A0242, No.SK2018A0216), the Domestic Visiting Research Project of Excellent Young Backbone Talents from Anhui Higher Education Institutions (Grant No.gxgnfx2019009), the Quality Project Foundation of Anhui Province (Grant No.2017mooc220, No.2018zhkt079, No.2015sxzx011, No.2019jyxm0238, No.2012sjjd025), the Natural Science Research Foundation of Anhui University of Chinese Medicine (Grant No.2019zrzd11, No.2018zryb06).

REFERENCES

[1] X. F. Deng, R. Z. Sun, J. Nie, W. D. Wang, and Y. X. Shi, “Modeling of Internet of Things System for Greenhouse Environment Monitoring Based on Time Automata,” Journal of Agricultural Machinery, vol. 47, no .7, pp. 301-308, Apr. 2016.
[2] J. Wang, N. J. Zhan, X. Y. Feng, and Z. M. Liu, “Overview of formal methods,”. Journal of Software, vol. 30, no .1, pp. 33-61, Nov. 2018.
[3] L. Yu, Y. Lu, X. J. Zhu, and S. W. Qiu, “A Web Services Description Language-Based Description Model of Internet of Things Services,” Sensor Letters, vol. 12, no .2, pp. 448-455, Feb. 2014.
[4] L. Ma, W. K. Li, C. Liang, and A. P. Li, “Formal Modeling and Verification of Resource-Oriented Internet of Things System,”. Miniature Microcomputer System, vol. 39, no .1, pp. 140-145, Jan. 2018.
[5] L. Yu, Y. Lu , B. H. Zhang, Y. Li, F. L. Huang, Y. L. Shen, and T. P. Shen, “Modeling and Verification of Resource-Oriented Internet of Things Services with Context Constraints,” 3rd International Conference on Cyberspace Data and Intelligence, Cyber DI 2019, and the International Conference on Cyber-Living, Cyber-Syndrome, and Cyber-Health, CyberLife 2019, Beijing, China, Dec. 2019, pp. 518-533.
[6] Z. Y. Zhu and Z. Q. Cheng, “Petri Net Modeling and Analysis of Intelligent Traceability Service System in Internet of Things,” Journal of Intelligent Systems, vol. 12, no .4, pp. 538-547, Nov. 2017.
[7] X. Y. Cong and H. Q. Yu, “Interactive Model of Internet of Things Based on Real-time UML Sequence Diagram,” Computer Science, vol. 41, no .11, pp. 79-87, Nov. 2014.
[8] L. X. Li, Z. Jin, and G. Li, Modeling and Verification of Internet of Things Service Based on Timed Automata,”. Chinese Journal of Computers, vol. 34, no. 8, pp. 1365-1377, Aug. 2011.
[9] G. Li, Q. Wei, L. X. Li, Z. Jin, Y. Xu, and L.W. Zhen, “Environment Based Modeling approach for services in the Internet of Things,” Chinese Science: Information Science, vol. 43, no. 10, pp. 1198-1218, Oct. 2013.
[10] L. S. Hu , S. Z. Wang, Y. Q. Chen, C. L. Gao, C. Y. Hu, X. L. Jiang, Z. Y. Chen, and X. Y. Gao, “Fall detection algorithms based on wearable device: a review,” Journal of Zhejiang University (Engineering Edition), vol. 52, no. 9, pp. 1717-1728, Sept. 2018.
[11] T. R. Wang, Z. Su, N. Liu, C. Li, and G. D. Fu, “Study on time-frequency feature extraction method for fall detection,” Journal of System Simulation, vol. 31, no. 12, pp. 2600-2605, Dec. 2019.
[12] X. L. Wang, D. H. Li, X. W. Zheng, T. S. Lou, G. Q. Ding, Y. Z. Jiao, and H. M. Zhao, “Research on Fall Detection Algorithm Based on RBF Neural Network,” Journal of Electronic Measurement and Instrument, vol. 33, no. 11, pp. 185-191, Nov. 2019.
[13] C.A.R. Hoare, “Communication Sequence Process,” Translated by C. C. Zhou, Beijing: Peking University Press, pp. 1-234, 2011.
[14] J. Sun, Y. Liu, J. S. Dong, Y. Liu, L. Shi, and É. André, “Modeling and Verifying Hierarchical Real-time Systems using Stateful Timed CSP,” ACM Transactions on Software Engineering and Methodology (TOSEM), vol. 22, no. 1, pp. 1-29, Mar. 2013.
[15] J. Sun, Y. Liu, J. S. Dong, and J. Pan, “PAT: Towards Flexible Verification under Fairness,” The 21th International Conference on

Computer Aided Verification (CAV 2009), Grenoble, France, June. 2009, pp. 709-714.
[16] Y. Liu, User Manual [EB/OL]. http://pat.sce.ntu.edu.sg/wp-source/resources/OnlineHelp/htm/index.htm. 2013-05-27.
[17] J. W. Ma, "Formal Analysis and Verification of Smart Home Platform Based on PAT Tool", Taiyuan: Taiyuan University of Technology, pp. 1-70, Apr. 2016.