A Behavioural Hierarchical Analysis Framework in a Smart Home: Integrating HMM and Probabilistic Model Checking

Wang, X., Liu, J., Moore, S., Nugent, CD., & Xu, Y. (2023). A Behavioural Hierarchical Analysis Framework in a Smart Home: Integrating HMM and Probabilistic Model Checking. *Information Fusion*, *95*, 275-292. https://doi.org/10.1016/j.inffus.2023.02.025

Published in:
Information Fusion

Publication Status:
Published (in print/issue): 01/07/2023

DOI:
10.1016/j.inffus.2023.02.025

Document Version
Author Accepted version

A Behavioural Hierarchical Analysis Framework in a Smart Home: Integrating HMM and Probabilistic Model Checking

Xia Wang[a,b], Jun Liu[b], Samuel J. Moore[b], Chris D. Nugent[b], Yang Xu*[c]

[a]*School of Computing and Artificial Intelligence, Southwest Jiaotong University, Chengdu 610031, China*
[b]*School of Computing, Ulster University, Northern Ireland BT15 1ED, UK*
[c]*School of Mathematics, Southwest Jiaotong University, Chengdu 610031, China*

Abstract

Smart homes offer great convenience for people living alone and assistance for physically impaired inhabitants. Robust behavioural analysis technology is one of the keys to maximizing the role of the Smart Home. Typically, when it comes to the behavioural analysis of its inhabitants, most researchers have acquired it through data collection from sensors, cameras, and portable Bluetooth sensors. However, a gap in research exists concerning activity recognition in the context of the users physical location in the environment. In this paper, we propose a hierarchical framework based on Hidden Markov Model (HMM) and suggest dividing the behavioural sequence analysis into two layers: spatial transfer and sensor transfer. In addition, we apply probabilistic model checking to verify the properties of each module's state transfer and obtain the probability of occurrence of the corresponding behavioural sequence. By integrating an implicit Markov model and probabilistic model checking, we effectively analyse the composition and probability of occurrence of three arbitrary sequences of complex behaviours. Finally, anomaly detection and behavioural guidance are discussed based on the proposed behavioural analysis methods.

Keywords: Smart home, Behavioural analysis, Hidden markov model, probabilistic model checking

1. Introduction

With the rapid development of the Internet of Things (IoT), Smart Home is favoured by most researchers as a product that benefits from the IoT. Smart homes are widely used in healthcare [1], health monitoring [2], and guidance service [3] by installing sensors, cameras, and infrared detection technology to record the daily behaviour of inhabitants. Sensor-based human behaviour analysis is one of the hot topics in these applications [4]. In daily behaviour analysis, human activities will have certain patterns due to the formation of behavioural habits. The sensor and human health state can be detected based on the daily human behaviour patterns [2]. For example, when the sensor data of a certain period is significantly different from the daily data and exceeds a certain threshold, it can be regarded as an abnormal scenario. The scenario will be analysed further to determine if it is the abnormal state of the resident or the abnormal sensor.

The challenges of human behaviour analysis include the complexity of human activities, the spatial and temporal variability of behaviour execution, the variability of human health states, and the uncertainty of sensor-based states [5]. The sensor-based smart home environment takes input from various underlying modules, including sensor readings, activation of sensor transfers, spatial recognition of human activity, and human behaviour detection, to record a sequence of behaviours that form human activity [6].

The complex activities' behavioural structure and semantics require higher-level representation and analysis methods. Compared to single-layer systems that represent and categorise low-level activities as distinct behavioural sequences, hierarchical approaches to design have the primary advantage of showing behavioural sequences with a more sophisticated structure. Hierarchical methods based on Hidden Markov Model (HMM) are particularly suitable for analysing human-environment interactions and complex behavioural sequences. By encapsulating behaviour sequences composed of multiple spaces, hierarchical approaches model data sets with smaller behaviour records and analyse the daily behaviour of an inhabitant more effectively. In addition, hierarchical modelling of behavioural sequences can more easily incorporate human knowledge. A clearer picture of the inhabitant's living state can be obtained by listing the sub-sequences in each space of the behaviour sequence and specifying the transfer between them.

One of the main challenges in this application domain is integrating the association between the human activity space and the given sensor data. Hierarchies can be regarded as structures representing the underlying semantic relationships of the considered behavioural sequences. For example, compound activities can be decomposed into simpler behavioural sequences and activities based on space and activation sensor sequences in human activity recognition. On the other hand, for modelling behavioural sequences, it is necessary to determine whether the model satisfies certain properties and the degree to which certain property is satisfied based on the model built. Thus, the behaviour state is judged. Traditional testing and simulation methods can only determine whether there are certain bugs with the model but not the correctness of the model. Therefore, the use of a more rigorous verification approach is essential for the analysis of human behaviour.

Formal methods are methods for modelling and verification of models based on strict mathematical semantics. In recent years, several researchers have made some progress in using it for research related to human behaviour recognition [7]. Formal verification [8] are employed to check whether the model is correct, whereas model checking [9] is one of the most popular techniques. In model checking, the system is specified as a model and fed into the model checker that thoroughly explores the model and confirms that it fulfils a logical formulation expressing the required properties. Due to complex and uncertain smart environment, a more effective approach which resolves this problem is integration of model-based method (e.g., the Markov Chain [10], such as Discrete-Time Markov Chains (DTMCs) [11]) and model checking based on logic reasoning. DTMCs are frequently used in performance analysis and implemented as a probabilistic model, e.g., specifies state transition and probabilistic state transition choice (e.g., for some conditions when more than one transition in a state is enabled) [12]. Probabilistic model checking is a formal verification method for probabilistic models that quantitatively verify the extent to which a probabilistic model satisfies certain properties [12]. Probabilistic model checking formulates the system as probabilistic transformation models such as Continuous-Time Markov Chains (CTMCs), or DTMCs. Quantitative logical properties are then applied to the models to check the results, returning ”True” if the requirements are satisfied, and ”False” and counterexamples if they are not. PRISM [13] is one of the most popular probabilistic model checker, which is a useful tool for formal modeling and study of systems that behave in a stochastic or probabilistic way. It has been used to examine systems in a variety of application domains [12].

This paper proposes a framework for analysing inhabitant behaviour in smart homes based on HMM and probabilistic model checking. Firstly, based on the HMM structure [14], the behaviour sequences are divided into hidden and observable layers. The hidden layer refers to the human activity space, and the observable layer is the sensor data. Based on Markov chains, the hidden layer and observable layer state transfer matrices are calculated. Formal models of the hidden and observed layers are developed based on DTMC and the input formats of PRISM. The behaviour sequences are divided according to the activities' space. The behaviour sequences under each space are represented as PCTL logic formulas. PRISM is used to verify the probability value that the formula holds and to verify that the model satisfies specific properties to judge the inhabitant's living status.

The main contributions of the paper include:

- A hierarchical analysis technique is suggested based on the structure of the HMM model. Analyse the space and sensor transfer in combination for the sequence of human behaviour in the smart home.
- The DTMC models for the hidden and observable layers are constructed using a probabilistic model checking framework, and the PCTL logic formulae are employed to create the behavioural sequences. PRISM is used to carry out the model's automatic verification.
- Integrate the results of HMM model and probabilistic model checking to represent the probability of the entire sequence of behaviours occurring.
- A discussion on anomaly detection and behavioural guiding in smart homes is developed within the context of the behavioural sequence analysis proposed in this study. The anomaly detection includes sensor anomalies and human health anomalies. Moreover, behavioural guidance refers to the ability to make the most feasible behavioural decision as soon as possible in the event of transient memory loss of the inhabitant.

The rest of the paper is organized as follows. Section 2 reviews the related works. Section 3 introduces the preliminaries. Section 4 presents an analysis framework for sensor network in a smart home environments based on HMM and probabilistic model checking. A case study based on an existing dataset is provided in Section 5 to demonstrate the effectiveness of our proposed methodology. Section 6 discusses anomaly detection and behaviour guidance based on human behaviour analysis. Section 7 concludes this paper.

2. Related works

Behaviour analysis covers a wide range of areas of investigation, from motion detection and context extraction to expert systems and advanced abstract behaviour models. Excellent behavioural analysis are necessary as the foundation for several in-depth research, such as anomaly identification and human health monitoring. Numerous approaches are already used to analyse and identify a smart home's human behaviour. Among them, Markov chains and HMMs are used by a wide range of researchers with good results. The work of Yamato *et al.* is one of the earliest attempts on HMM for behaviour analysis [15]. To better analyse the complex multi-scale structure that appears in many natural sequences, especially in language, handwriting and speech, Fine *et al.* proposed the Hierarchical Hidden Markov Models (HHMM)[16].

Their model's structure is fairly open-ended and allows for an infinite number of sub-model activations. Quintas *et al.* [17] discussed learning and recognising models of human behaviour from multi-modal observations in a smart home, and the method was successfully implemented to recognise concurrent Hidden Markov Models of occurrence. Liisberg *et al.* [18] developed methods for indirect observations and characterisation of inhabitant behaviour based on HMMs. They discovered that by including more states in the HMM, a novel interpretation of these states is generated, which may help explain how many people live in the apartment. Sánchez *et al.* [19] used human behaviour modelling to determine whether the current activity is normal or abnormal to simulate human behaviour in smart home environments, detect abnormal behaviour, and alert family members or caregivers when assistance is required. Wang *et al.* [20] constructed a hidden Markov Model of an individual's daily behaviour activity state based on the long-term activity data. The authors demonstrated that the daily behaviour HMM developed for individuals in the study could detect changes in human behaviour patterns and indicate specific behaviours that had changed. Even though existing smart home technologies have achieved substantial success, more research is still required to maintain the mode's reliability and safety because of the environment's complexity and unpredictability.

In recent years, there has been a great deal of research focusing on the factors influencing the analysis of human behaviour in smart homes [21, 22, 23, 24, 25]. For example, Rialle *et al.* [22] introduced a wide range of new information, communication, and data acquisition technologies used in Health Smart Home (HSH), presented the HSH concept in terms of technical, economic, and human requirements. To identify unusual behaviour among inhabitants of smart homes, Lühr *et al.* [5] developed a novel application of Intertransaction Association Rule (IAR) mining. Lundström *et al.* [26] applied clustering and the Random Forest (RF) algorithm to detect deviating human behaviour. Their study considered three different forms of deviation time, space, and the variation between clusters of comparable behavioural patterns. Li *et al.* [27] proposed Joint Domain and Semantic Transfer Learning (JD-STL) algorithms for radar-based Human Activity Recognition (HAR). The HAR model is trained by using a sparsely approach alleviates the need to label a large number of radar signals. Birnbach *et al.* [28] leverages information from several smart home sensors to verify physical events. These method guard against both sophisticated attackers and malfunctioning event sensors. Meanwhile, they introduced the P_{EEVES} system, which uses the smart home's physical properties to verify events automatically. Lina *et al.* [29] proposed information filter-based fusion data in the research of sensors in a smart home, where an inverted pendulum biomechanics model is introduced. Moore *et al.* [30] suggested to detect sensor anomalies based on Markov model in the research on a smart home. The results of the three different test vectors which are to be tested in each of the first, second, and third order Markov models are also compared. Yang proposed an approach to activity recognition by using an Extended Belief Rule-based System (EBRBS), which offers promising performance compared to popular benchmark activity recognition models and exhibits high robustness in the presence of sensor failures [31]. Li *et al.* [32] developed a method for recognizing a single user's daily behaviour that may adaptively control the sensor noise during human activities in multitenant smart home scenarios. However, most of the current studies lack the verification of the probabilistic model. Verifying probabilistic models of indeterminate behaviour allows for clearer identification of patterns in behaviour, such as the probability of behaviour transfer, the sequence in which behaviour occurs, or the probability of the next behaviour being performed in the current state.

In the last few years, formal methods have been favoured by a large number of researchers in ensuring the safety design of the software and hardware systems. Saives *et al.* employed formal methods to check the required qualities to get judgments on behavioural deviations [33]. The nondeterministic behaviour of inhabitants living in smart homes requires the introduction of quantitative properties when analysing behaviour and verifying the extent to which the behaviour satisfies a specified property, such as, what is the probability of completing an activity with a set sequence of behaviours. Therefore, this paper is of greater interest in probabilistic model checking techniques. Probabilistic model checking [12] is a well-established technology for quantitatively and qualitative analysing state-based models of operation, such as Markov decision processes. At present, many researchers in the industrial field have adopted the probabilistic model checking technology. Wang *et al*. [34] integrated the probabilistic model checking and reliability analysis method to solve reliability value in the research of the flight control system. To check the reliability and quantitative properties of Energy Routing (ER) systems during the design phase, Gao *et al*. [35] described the ER system architecture as a continuous-time Markov chain model to provide a formal verification solution for the ER-based systems. Baouya *et al*. [36] presented a deployment decision formulation based on the probabilistic model checker PRISM. I'Yvonnet *et al*. [37] used the PRISM framework and the model checker to express and examine interesting temporal logic properties of the dynamic evolution of human activities. It is intended that this modeling approach may provide new behavioural guidelines for interpreting medical patient performance. In order to assist Alzheimer's disease patients, Gao *et al*. [38] suggested a probabilistic model checking-based strategy to anticipate patient behaviours and identify unusual behaviours connected to moderate cognitive impairment.

We have proposed a method to analyse the reliability of sensors in smart homes based on the Markov model and probabilistic model checking.[39]. Building on existing researches, this paper integrates HMM and probabilistic model checking method to propose a hierarchical analysis framework in human behaviour analysis. It is possible to swiftly assess the probabilities that a stationary inhabitant will engage in a series of behaviours. We build DTMC models based on HMM and probabilistic model checking for the Markov models generated from the Van Kasteren dataset [40]. Convert the sequence of behaviours to be verified into property formulas and verify it by PRISM. It is feasible to notice the properties that can be satisfied between state transfers, and we can conclude the intermediate behaviours from the results of the property verification. The method can form the foundation for extended research related to the analysis of human behaviour, such as anomaly detection.

3. Probabilistic Model and Model Checking

3.1. Markov chain

When a stochastic process possesses the Markov property, the conditional probability distribution of its future state is entirely dependent on its present state. This is true for both the current state and all past states. The Markov process is explained with the following definition.

Definition 1. *(Markov process)[41]. Suppose* $\{X(t), t \in T\}$ *is a stochastic process,* $\mathbf{E}$ *is the state space, if* $\{t_1 < t_2 ... < t_n < t\}$*, any* $x_1, x_2, ..., x_n, x \in \mathbf{E}$*, the conditional distribution function of the random variable* $X(t)$ *under the known variable* $X(t_1) = x_1, ..., X(t_n) = x_n$ *is only related to* $X(t_n) = x_n$*, but not related to* $X(t_1) = x_1, ..., X(t_{n-1}) = x_{n-1}$*. That is, the conditional distribution function satisfies Equation (1).*

$$F(x, t|x_n, x_{n-1}, ..., x_2, x_1, t_n, t_{n-1}, ..., t_2, t_1) = F(x, t|x_n, t_n) \tag{1}$$

Discrete Time Markov Chain (DTMC) is a special form of Markov process, which has discrete values in the general definition by implementing the time parameters and the state space.

Definition 2. *(DTMC)[11]. A DTMC is a tuple* $\mathbb{D} = (S, \overline{s}, P, L, AP)$*, where,*

- S is a finite set of states.
- $\overline{s}$ denotes the initial state.
- P is a probabilistic transition function, such that for every state $s \in S$, and $\Sigma_{s' \in S} P(s, s') = 1$.
- $L : S \rightarrow 2^{AP}$ is a labelling function that assigns a set of atomic propositions to each state $s \in S$.
- AP is a finite set of atomic propositions.

3.2. Hidden Markov Model (HMM)

The HMM is the simplest dynamic Bayesian network and a particularly well-known directed graph structure, which is mainly used for modeling time series date. In general, HMM contain two levels of uncertainty. A Markov chain that describes the probabilistic relationship between states in terms of the likelihood that one state follows another, and a stochastic observation process associated with each hidden state.

Definition 3. *(HMM)[42]. A HMM is a tuple* $HMM = (O_t, I_t, \pi, A, B)$*, where,*

- O_t is a set of observable sequences.
- I_t denotes the hidden state sequences.
- π is the probability of the initial state.
- A is the state transition matrix for hidden layer.
- B is the transition matrix for observable states.

Let I_t indicates the hidden state at time t and O_t the corresponding observation. Assuming that there are n possible state, we have $I_t \in \{1, ..., n\}$. Let $o_1, o_2, ..., o_T$ denotes the observation sequence of random variable O_t. The three most important parameters in HMM are $\{\pi, A, B\}$. Two basic assumptions need to be satisfied: the state at time t only depends on the state at time t-1 and the observation at any moment only depends on the state at current moment.

3.3. Probabilistic model checking

The purpose of verifying a probabilistic model is to ensure that it satisfies fundamental qualitative and quantitative properties. Probabilistic model checking is a technique to achieve this purpose. It includes calculating the probability that specified properties are satisfied, as well as determining whether specific properties are satisfied. A probability model checker PRISM [12], developed initially at the University of Birmingham and now maintained and further developed at the University of Oxford, is a helpful tool for studying and modelling stochastic or probabilistic systems. Systems in various application fields, such as risk evaluation [43], communication protocols [44], and security threats [45], have adopted probabilistic model checking techniques for research. The model in PRISM [12] subsumes some well-known temporal logics by including PCTL [46, 47], Continuous Stochastic Logic (CSL) [48], and Probabilistic Logic (PL) [49]. PCTL is a well-known temporal logic for probabilistic verification of DTMC model.

Definition 4. *(PCTL syntax)[46]. Given a set of atomic propositions AP, the PCTL formulae are defined by the following BNF grammar.*
$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \psi \mid \mathbb{P}_{\bowtie\,\alpha}(\psi)$
$\psi ::= X\varphi \mid F\varphi \mid \varphi \cup \psi \mid \varphi \cup^{\leq k} \psi$

Where:

- $p \in AP$ denotes a finite set of atomic propositions.
- φ and ψ are the state formulae and path formulae interpreted over the states and paths of the model, respectively.
- $\neg$ and $\vee$ are the boolean connectives that defined in the usual way.
- $\mathbb{P}_{\bowtie\,\alpha}(\psi)$ is a probabilistic operator where $\bowtie \in \{<, \ \leq, \ \geq, \ >\}$ and α *in* $[0, 1]$ is a probability bound or threshod.
- $k \in \mathbb{N}^{+}$ is a positive integer number reflecting the maximum number of transitions needed to reach a certain system.
- X, F are defined as 'next' and 'finally', respectively. $X\varphi$ denotes the next state satisfies the φ, $F\varphi$ represents that the state will eventually meet φ.

A state s in a DTMC model meets an atomic proposition p if $p \in AP$. s satisfies a state formula $\mathbb{P}_{\bowtie\alpha}(\psi)$, denotes $s \models \mathbb{P}_{\bowtie\alpha}(\psi)$, if the probability of taking a path starting from s and satisfying ψ satisfies the bound $\bowtie \alpha$. The path formula $\varphi \cup^{\leq k} \psi$ is true on a path if ψ holds in the state at several time step $i \leq k$ and at all preceding states φ holds.

Definition 5. *(PCTL sematic)[46]. The following is a list of PCTL formulae's formal semantic. Assume V is the valuation function mapping each states s to a collection of propositions and M is either a DTMC or MDP. The satisfaction relation $\models$ is inductively defined on the structure of a given PCTL formula ρ for a particular state s. Where,*

- $M, s \models \rho$ iff $\rho \in V(s)$.
- $M, s \models \neg\rho$ iff $M, s \not\models \rho$.
- $M, s \models \rho \wedge \psi$ iff $M, s \models \rho$ and $M, s \models \psi$.
- $M, s \models P_{\sim r}[\varphi]$ iff $\pi_m(\sigma \in Paths(s)\ s.t., M, \sigma \models \varphi) \sim r$.
- $M, s \models X\rho$ iff $M, \sigma[1] \models \rho$.
- $M, s \models \rho U \psi$ iff $\exists i \geq 0 s.t., M, \sigma[i] \models \psi$ *and* $(\forall j < i) M, \sigma[j] \models \rho$.
- $M, s \models \rho U^{k} \psi$ iff $\exists 0 \leq i \leq k s.t., M, \sigma[i] \models \psi$ *and* $(\forall j < i) M, \sigma[j] \models \rho$.

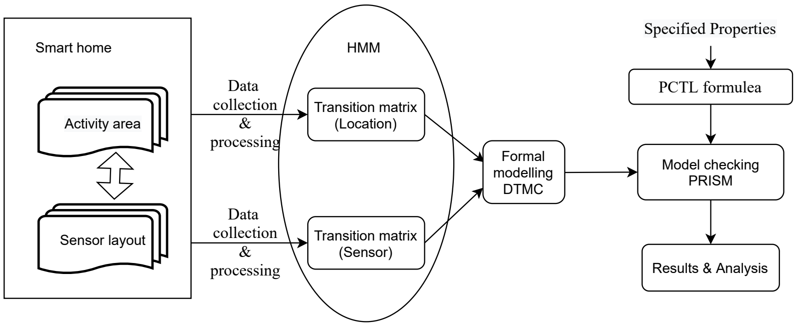


Figure 1: The framework for behaviour analysis.

4. Proposed Methodology and Framework

This section discusses the behaviour analysis framework, the complexity of HMM-based hierarchical analysis, and combination methods. In Fig. 1, the proposed methodology is illustrated, outlining the definition of the HMM-based hidden state and observable state layers. It is worth noting that the framework employs a hierarchical approach to take the transmission of sensor states in various spaces into distinct consideration. This analytical technique is appropriate for scenarios when one or more independently operating sensors are spatially distributed throughout the environment.

Initially, the given dataset is preprocessed to model the spatial and behavioural transfer of the inhabitant in the smart home for the hidden and observable layers, respectively. Two types of transfer matrix are obtained in the first step, they are location transfer matrix and sensor transfer matrix. The location transfer matrix is calculated based on the activity transfer. The sensor transfers reflect the inhabitant's behaviours in a location. Employing these Markov transfer matrices, DTMC models are constructed and entered into PRISM in a customized format. Then, convert the specified properties into PCTL formulae. Finally, the model and specific properties are verified with PRISM. The results of the verification shows the probability values of performing the corresponding behaviour sequences. More detail for each aspect of the framework is given below.

4.1. Hierarchical analysis based on HMM

The hierarchical analysis method suggested in this paper indicates that the spatial transfer and sensor transfer layers are separated in the behaviour analysis of an inhabitant in a smart home. Among other things, the space where the inhabitant is located is determined by activity, such as sleeping should be in the bedroom and bathing should be in the washroom. The space in which the inhabitant is located belongs to the hidden state layer. And sensor transfer refers to the transfer of activated sensors in a space, where the activated sensors are considered as observable states. Fig. 2 shows the process. Inside the rectangular box is a set of behaviour sequences. In a smart home it can be regarded as a set of successive activated sensors. L_i refers to the space where the i^{th} sensor activation occurs. We separate this behaviour sequence into two components based on the HMM: the transfer of space and the transfer of activated sensors under a specific space. First, each space's transfer probability is calculated. On the other hand, the corresponding sensor activation probability or the probability of the corresponding sensor activation sequence in

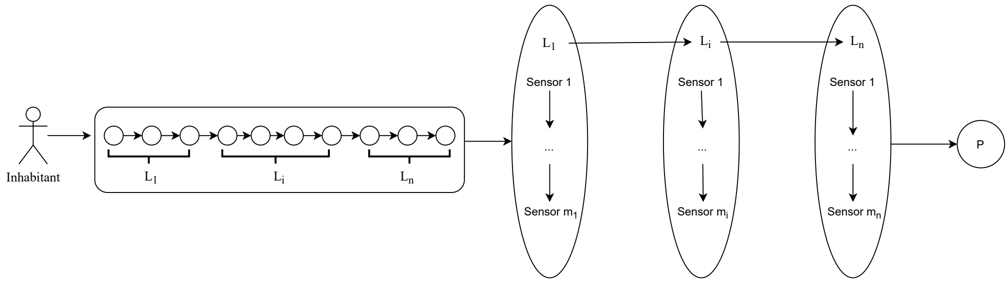


Figure 2: Hierarchical analysis between hidden layer and observable layer.

a certain space is calculated. Finally, it is possible to determine the probability that the sequence of behaviours will occur. The probability is calculated by the formula (2). s_i can be a transfer probability value between spaces, or a probability value for the occurrence of a behaviour, and or a set of behaviours under a space.

$$\prod s_i = s_1 s_2 s_3 \cdots s_n. \tag{2}$$

4.2. Calculate the state transfer matrix of the hidden and observable layers

Algorithm 1 shows the pseudo-code for solving the state transfer matrix based on Markov chains. The behaviour sequence dataset is used as the input to the algorithm and the state transition matrix as the output. Extracts the possible combinations in the sequence of behaviour, i.e., the combination of two consecutive behaviours. The first behaviour denotes the current state s_t and the second denotes the state s_{t+1} entering at the next time. Calculate the set of all possible states entering at the next moment when the current state is s_t, and the number of times each state occurs. For the calculation of space transition probability, we consider all the instances of the same state. The probability (p_{ij}) of transition from i^{th} state to j^{th} state is calculated by (3).

$$p_{ij} = \frac{n_{ij}}{n_i} \tag{3}$$

Where n_{ij} denotes the number of transition from state i to state j and n_i the total number of transition from i^{th} state. In principle, the probability sum of all states that may occur at the next moment is 1 when stay in the state s_t.

4.3. Formal modelling and probabilistic model checking for the behaviour analysis

Formal models are constructed for the state transfer of the hidden and observable layers based on DTMC and PRISM input formats, respectively. The state transfer statements are defined in the module. First, this module's variables and comments are all defined. The format of the comments is: $[action]guard- > rate_1 : updated_1 + ... + rate_n : updated_n$ [50]. The action is used to carry out the module synchronisation. One update will take place based on the value of rate if the action is synchronising and the guard is satisfied. Each update shows a transition took place. Formulas with a name and an expression are defined in the PRISM model to prevent code duplication. On the other hand, rewards can be incorporated to PRISM models [50], linking actual values to model states or transitions.

Algorithm 1 Transition matrix based on Markov property

REQUIRE Behaviour transfer sequence.
ENSURE Behaviour transfer matrix.

1 : Extract distinct binary sequences $s_t s_{t+1}$ in behavioural sequences;
2 : The first in the binary sequence s_t is the current state;
3 : The second in the binary sequence s_{t+1} is the next state that likely to enter;
4 : Calculate the number of entering the next state in the current state;
5 : Generate a state transition matrix that records the number of times each binary sequence occurs;
6 : Based on the probability formula , convert the result of the fifth step into a transition probability matrix.
6 : Output the transition probability matrix.

Here, PCTL is used to express the specified properties. P, which enables one to consider the probability that an event will occur, it is the most significate operator. As mentioned previously, $P_{\bowtie m}\varphi$ is true in a state s of a model if the probability that the event happen is met by the paths from state s satisfies the $\bowtie m$. For instance, the PCTL formula P=0.9[X (s=2)] holds in the initial state if the probability that s=2 is true in the next state equals 0.9. In addition, the probability of the existence of a path in the model can be obtained by probability model checking. Such as, P=?[X (s=3)] indicates the probability that s=3 is true in the next state.

In this paper, we model the state transfer of the hidden and observable layers separately and verify the spatial transfer and the sequence of behaviours occurring in each space based on the sequence of behaviours, respectively, to obtain the corresponding probability. Finally, the integrated probability value of a specific behaviour sequence is calculated.

5. Case Study

5.1. Illustrative example

A smart home with multiple sensors is developed in [40]. Here is an intelligent environment equipped with 14 binary sensors, the detailed information is shown in Fig. 3. In this smart home, there are four rooms where sensors are installed. Each room is a separate activity space. The first is installing the front door sensor (F1) at the doorway. The second room is the bedroom with bedroom door sensor (HS1). The third is the kitchen. Nine sensors are installed in the kitchen. They are a Microwave sensor (KS1), a Cups cupboard sensor (KS2), a Fridge sensor (KS3), a Plates Cupboard sensor (KS4), a Dishwasher sensor (KS5), a Freezer sensor (KS6), a Pans cupboard sensor(KS7), a Washing Machine sensor (KS8), and a Groceries sensor (KS9). The fourth is the washroom, with a Hall-Toilet door sensor (TS1), a Toilet Flush (TS2), and a Hall-Bathroom door sensor (BS1) installed.

In the Van Kasteren dataset [40], the sensor data of a 26 years old man living alone in a smart home for 28 days was recorded. During these 28 days, only the man was present in this home and nobody else was there, not even guests or pets. Table 1 displays the typical activities of the inhabitant in each space.

The daily behaviour of inhabitants is complex. Here we give 3 cases randomly. They will be analysed in detail later.

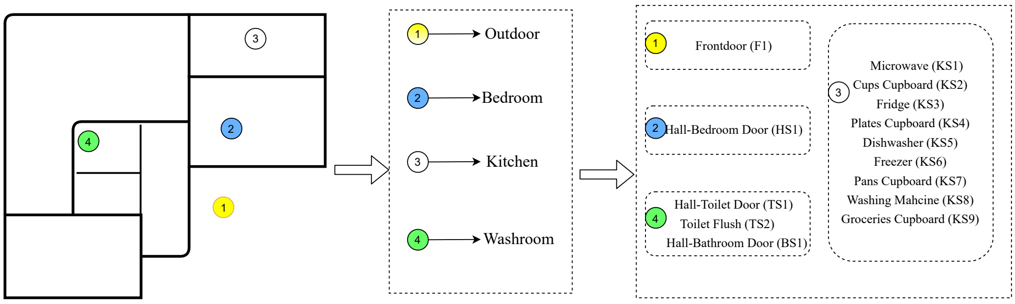


Figure 3: The sensors layout in the smart home.

Table 1: Sensors in the smart home

Room ID	Label	Activity
1	Outdoor	Leave house
2	Bedroom	Go to bed
3	Kitchen	Cook (prepare breakfast, prepare dinner, get drink)
4	Washroom	Use toilet, Take shower

Case 1 : In the morning, the inhabitant woke up, walked out of the bedroom, entered the washroom, used the toilet flush, walked out of the washroom, and then walked into the kitchen. After walking into the kitchen, he opened the fridge, took the bread, opened the plates cupboard, took out a plate, put the bread on the plate, put it in the microwave oven and heated it up.

Case 2 : When the inhabitant returned at noon, he first put the vegetable into the fridge, then took out the commonly used cup from the cupboard to drink water and put the cup back into the cupboard. Then walk into the washroom, flush the toilet, and walk out. Walk into the kitchen again and wash the dinner plates with the dishwasher.

Case 3 : In the evening, the inhabitant walks into the washroom to shower, uses the toilet, flushes the toilet, walks out of the washroom and enters the bedroom. The three cases are randomized, and each case corresponds to a sequence of behaviours. Based on the proposed method, we aim to calculate the probability of the inhabitant completing the corresponding sequence of behaviours in the smart home.

5.1.1. HMM parameters

Based on the sensor layout of the smart home, we can perform a more sophisticated analysis of the inhabitant's behaviour through the HMM. In a smart home, there are four spaces where sensors can detect inhabitant behaviour. In the behaviour dataset, seven activities are given, including leave house, use toilet, take shower, go to bed, prepare breakfast, prepare dinner, get drink. Normally, these seven activities can be done in four spaces respectively. When leaving the house, the front door sensor will be activated to enter the outdoor space. Use toilet and take shower have to be completed in the washroom. Go to bed is implemented in the bedroom. Prepare breakfast, prepare dinner, and get drink all belong to cooking, which is done in the kitchen. Therefore, we reduce the analysis of activities to the analysis of the spaces in which the inhabitants are located.

As shown in Fig. 4, inside the dotted box are the features we extracted from the environment, and the circles represent different spaces as hidden layers. And the box below the circle

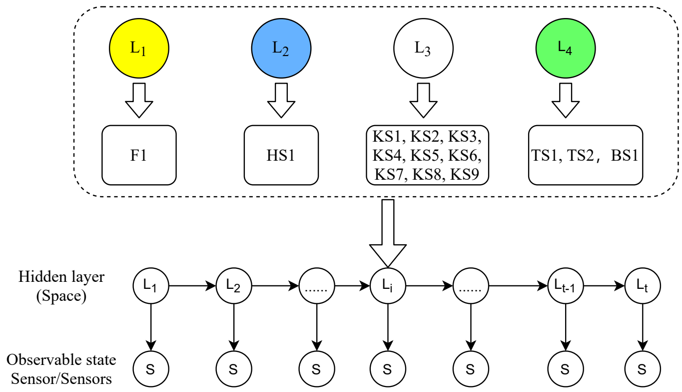


Figure 4: Activity HMM diagram for the inhabitant.

represents the sensors set up in the corresponding space. The bottom of Fig. 4 is the HMM model. The first layer represents the hidden state layer and L_i represents the space where the inhabitant stays at time i. The lower S represents the sensor that is activated at this moment, it can be a single sensor, or it represents a sequence of sensors that may be activated in this space. The values of all the space transitions are provided in Matrix (4). L_1, L_2, L_3, and L_4 denote outdoor, bedroom, kitchen, and washroom, respectively. Each row represents the current space, and the column represents the space to be entered at next moment. For example, the value 0.958 marked in red in the Matrix means that the current is in bedroom and the probability of entering washroom at the next moment is 0.958.

According to the records in the Van Kasteren dataset, the probability of the inhabitant entering each space at start can be calculated as outdoor (0.076), bedroom (0.091), kitchen (0.378), and washroom (0.456). Therefore, $\pi = [0.076, 0.091, 0.378, 0.456]$.

$$\begin{array}{c} \\ L_1 \\ L_2 \\ L_3 \\ L_4 \end{array} \begin{array}{c} \begin{array}{cccc} L_1 & L_2 & L_3 & L_4 \end{array} \\ \begin{bmatrix} 0.030 & 0.030 & 0.182 & 0.757 \\ 0.000 & 0.000 & 0.042 & 0.958 \\ 0.020 & 0.000 & 0.220 & 0.760 \\ 0.228 & 0.162 & 0.235 & 0.375 \end{bmatrix} \end{array} \tag{4}$$

As for the sensor trigger events of each space, there is only one sensor in the front door and bedroom. Therefore, once entering those spaces, the probability of the corresponding sensor occurrence is 1.

In the washroom, a Hall-Toilet door sensor, a Toilet Flush sensor and a Hall-Bathroom door sensor are installed. The first activated sensor to enter the washroom is the Hall-Bathroom door sensor. Matrix (5) is the state transition matrix between the Hall-Toilet door sensor, the Toilet Flush sensor, and the Hall-Bathroom door sensor.

$$\begin{array}{c} \\ TS1 \\ TS2 \\ BS2 \end{array} \begin{array}{c} \begin{array}{ccc} TS1 & TS2 & BS1 \end{array} \\ \begin{bmatrix} 0.737 & 0.083 & 0.180 \\ 0.151 & 0.198 & 0.651 \\ 0.193 & 0.475 & 0.332 \end{bmatrix} \end{array} \tag{5}$$

Nine sensors were installed in the kitchen. By calculating the frequency of events in the dataset, it can be seen that the probability of activating each sensor in turn at the initial moment after entering the kitchen is KS1 (0.064), KS2 (0.100), KS3 (0.289), KS4 (0.131), KS5 (0.042), KS6 (0.082), KS7 (0.102), KS8 (0.034), KS9 (0.155). Based on recording the number of sensor activations and the transfer between states for this inhabitant during 28 days, we can calculate the probability Matrix 6 of sensor activation transition in the kitchen.

$$\begin{array}{c} \\ KS1 \\ KS2 \\ KS3 \\ KS4 \\ KS5 \\ KS6 \\ KS7 \\ KS8 \\ KS9 \end{array}
\begin{array}{c}
\begin{array}{ccccccccc} KS1 & KS2 & KS3 & KS4 & KS5 & KS6 & KS7 & KS8 & KS9 \end{array} \\
\begin{bmatrix}
0.374 & 0.042 & 0.250 & 0.042 & 0.042 & 0.083 & 0.042 & 0.000 & 0.125 \\
0.043 & 0.361 & 0.277 & 0.106 & 0.021 & 0.043 & 0.064 & 0.000 & 0.085 \\
0.064 & 0.100 & 0.517 & 0.100 & 0.018 & 0.055 & 0.055 & 0.027 & 0.064 \\
0.098 & 0.115 & 0.180 & 0.115 & 0.000 & 0.131 & 0.066 & 0.000 & 0.295 \\
0.000 & 0.000 & 0.056 & 0.277 & 0.388 & 0.000 & 0.167 & 0.056 & 0.056 \\
0.079 & 0.105 & 0.211 & 0.184 & 0.000 & 0.237 & 0.026 & 0.026 & 0.132 \\
0.000 & 0.061 & 0.184 & 0.122 & 0.082 & 0.000 & 0.469 & 0.000 & 0.082 \\
0.000 & 0.000 & 0.100 & 0.000 & 0.100 & 0.000 & 0.000 & 0.700 & 0.100 \\
0.042 & 0.028 & 0.310 & 0.085 & 0.028 & 0.099 & 0.028 & 0.000 & 0.380
\end{bmatrix}
\end{array} \tag{6}$$

5.1.2. PRISM implementation

The PRISM code describing the DTMC model of the space transfer is shown in **Listing 1**. The model has an initial state and 4 states, 1 through 4, which denote the differ spaces, i.e., outdoor, bedroom, kitchen, and washroom, respectively. It is worth noting that the initial state can be any space, and the corresponding probabilities is $\pi = [0.076, 0.091, 0.378, 0.456]$. The left side of the "→" represents the current state, and the right side represents the state and probability that may be entered in the next step. For example, $s = 1 \rightarrow 0.030 : (s' = 1) + 0.030 : (s' = 2) + 0.182 : (s' = 3) + 0.758 : (s' = 4)$ means that the agent is now in space S_1, the probability of entering S_1 is 0.030, entering S_2 is 0.030, entering S_3 is 0.183, and entering S_4 is 0.758.

Listing 1: The model of the space transition in smart home

```
dtmc
module space transition
s: [1..4] init 1;
[]s=1->0.030:(s'=1)+0.030:(s'=2)+0.182:(s'=3)+0.758:(s'=4);
[]s=2->0:(s'=1)+0:(s'=2)+0.042:(s'=3)+0.958:(s'=4);
[]s=3->0.020:(s'=1)+0:(s'=2)+0.220:(s'=3)+0.760:(s'=4);
[]s=4->0.228:(s'=1)+0.162:(s'=2)+0.235:(s'=3)+0.375:(s'=4);
end module
```

Listing 2 is the PRISM code for the activation sensor's transitions in washroom. There are three states here, 1 denotes activating Hall-Toilet door sensor, 2 indicates activating Toilet Flush sensor, and 3 represents activating Hall-Bathroom door sensor. di represents the i^{th} $i \in \{1, 2, 3\}$ sensor is activated. In the smart home, there are only two possible values for each sensor due to the binary sensors that were used. The sensor is either active or inactive when it is given a value of 1 or 0. The initial value of s is 3, which means that entering the space will first activate Hall-Bathroom door sensor. Therefore, the initial value of d1 and d2 is 0, and the initial value of d3 is 1. When the Hall-Bathroom door sensor is activated, there is 0.193 probability that Hall-Toilet door sensor will be activated at the next moment, 0.475 probability that Toilet Flush sensor will be activated, and 0.332 likelihood that Hall-Bathroom door sensor will be activated again. Note that when the activated sensor changes at the next time step, the d-value of the activated sensor at the current time step becomes 0. More generally, it implies that only the activated sensor has a d-value of 1, while the d-values of all other sensors remain at 0 or change to 0.

Listing 2: The DTMC of the activation sensor's transitions in washroom

```
dtmc
module state transition in washroom
s: [1..3] init 3;
d1: [0..1] init 0;
d2: [0..1] init 0;
d3: [0..1] init 1;
[] s=1->0.737:(s'=1)&(d1'=1)&(d2'=0)&(d3'=0)+0.083:(s'=2)&(d2'=1)&(d1'=0)
&(d3'=0)+0.180:(s'=3)&(d3'=1)&(d2'=0)&(d1'=0);
[] s=2->0.151:(s'=1)&(d1'=1)&(d2'=0)&(d3'=0)+0.198:(s'=2)&(d2'=1)&(d1'=0)
&(d3'=0)+0.651:(s'=3)&(d3'=1)&(d2'=0)&(d1'=0);
[] s=3->0.193:(s'=1)&(d1'=1)&(d2'=0)&(d3'=0)+0.475:(s'=2)&(d2'=1)&(d1'=0)
&(d3'=0)+0.332:(s'=3)&(d3'=1)&(d2'=0)&(d1'=0);
end module
rewards "Halltoiletdoor"
d1=1:1;
endrewards
rewards "Toiletflush"
d2=1:1;
endrewards
rewards "Hallbathroomdoor"
d3=1:1;
endrewards
```

Three rewards comments are given at the end of Listing 2. For example,
rewards"*Hallbathroomdoor*";
$d3 = 1 : 1$;
endrewards

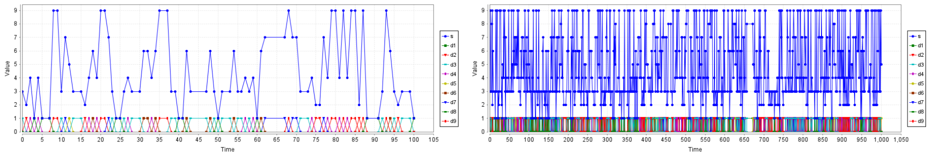
This reward comment denotes each time d3=1(the sensor is activated), the reward is incremented by 1.

Listing 3 in Appendix is the PRISM code for the active sensor's transitions in kitchen. The initialization of the model is defined in the module section including state transitions. There are 9 states. s is the overall state represents the current state of the j^{th} ($j \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$) sensor. The interpretation of dj and reward is the same as in Listing 2. Nine examples of reward are given at the end of Listing 3.

5.2. Simulations in PRISM

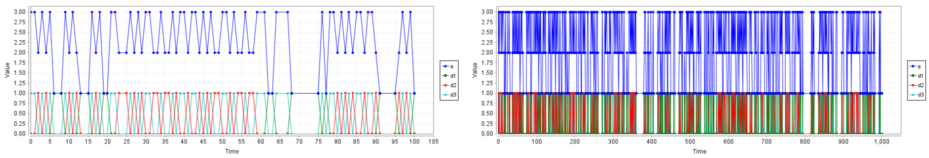
After the model is completed, we can simulate and debug the model in the simulator in PRISM. This paper presents sensors' state transfer simulation results in the kitchen and washroom.

Fig. 5 shows the simulation figure of the activated sensor transfer in the kitchen. The value of dj is 1 when the activation sensor is the j^{th} sensor. This result is as it should be and satisfies the nature of the binary sensor. For the experiments, we set two kinds of time steps, 100 and 1000. From Fig. 5 (a), we can find that the kitchen's sensors activated by the inhabitant are rather scattered, but there is still a regular pattern. For example, the inhabitant activated KS4 and



(a) (b)

Figure 5: Simulation path for the sensors in kitchen.(a)Simulation step=100;(b)Simulation step=1000;



(a) (b)

Figure 6: Simulation path for the sensors in washroom.(a)Simulation step=100;(b)Simulation step=1000;

KS9 alternately more frequently. And this phenomenon is more pronounced in the experiment results with time step 1000.

Fig. 6 shows the simulation figure of the activated sensor transfer in the washroom. Similarly, from the figure, we can notice that the inhabitant alternately activated the Hall-bathroom door sensor and the Toilet flush sensor more frequently. The observers analyse the inhabitant's behaviour from the simulation results of the model and thereby summarize the behaviour habits.

5.3. Temporal logic properties for the inhabitant behaviour

For the model described in the previous section, we encoded and verified several properties in PCTL. Two kinds of properties may be defined, those to verify the space transition, and sensor transitions in each space.

5.3.1. Space transitions in the smart home

Property 1. What is the probability to reach the kitchen from bedroom within 5 time steps. The verification shows the probability is 0.042.

$$P = ?[s = 2\ U^{\leq 5} s = 3]$$

Alternatively, we can simultaneously consider what is the probability of reaching the outdoor and the washroom, respectively, from the bedroom within 5 steps. Here it is only necessary to change "s=3" to "s=1" or "s=4".

5.3.2. Active sensor transitions in kitchen

Property 2. What is the probability of activating each sensor in turn, in the order of *Cups Cupboard* → *Fridge sensor* → *Plates Cupboard sensor* → *Groceries sensor* → *Firdge sensor*. The initial state is 2, which means the active sensor is the Cups Cupboard sensor. The result is 0.00253.

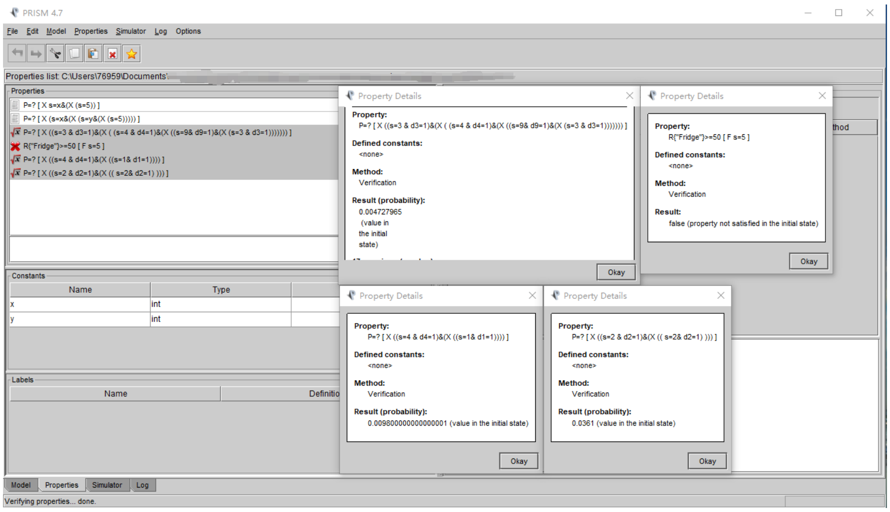


Figure 7: Verification for behaviour properties in kitchen.

$$P =?[X((s = 3\&d3 = 1)\&(X((s = 4\&d4 = 1)\&(X((s = 9\&d9 = 1)\&(X(s = 3\&d3 = 1)))))))]$$

Property 3. This property is used to verify that the fridge sensor will be activated more than or equal to 50 times from the initial state to the eventual transfer to state 5. Verification shows that this property is false. This property represents the frequency of events and thus determines if an exception exists for events that should occur rarely but occur more frequently at a particular time.

$$R\{\text{"}Fridge\text{"}\} >= 50[F\ s = 5]$$

Property 4. What is the probability of activating each sensor in turn, in the order of $Fridge\ sensor \rightarrow Plates\ Cupboard\ sensor \rightarrow Microwave\ sensor$. The initial state is 3, which means the active sensor is Fridge sensor.

$$P =?[X((s = 4\&d4 = 1)\&(X((s = 1\&d1 = 1))))]$$

Property 5. What is the probability of activating each sensor in turn, in the order of $Fridge\ sensor \rightarrow Cups\ Cupboard\ sensor \rightarrow Cups\ Cupboard\ sensor$. The initial state is 1, which means the active sensor is Microwave sensor.

$$P =?[X((s = 2\&d2 = 1)\&(X((s = 2\&d2 = 1))))]$$

Fig. 7 shows the interface in PRISM verification for the behaviour in kitchen. This interface shows the four relevant properties listed above and their verification results. For example, the verification result for property 2 is approximately 0.0047, which means that the probability of the inhabitant completing the activity in the smart home according to the behavioural sequence is only 0.0047.

5.3.3. Active sensor transitions in washroom

Property 6. What is the probability of activating each sensor in turn, in the order of $HallBathroom\ sensor \rightarrow Toilet\ Flush\ sensor \rightarrow HallBathroom\ sensor$. The initial state is 3, which means the active sensor is Hall-Bathroom door sensor.

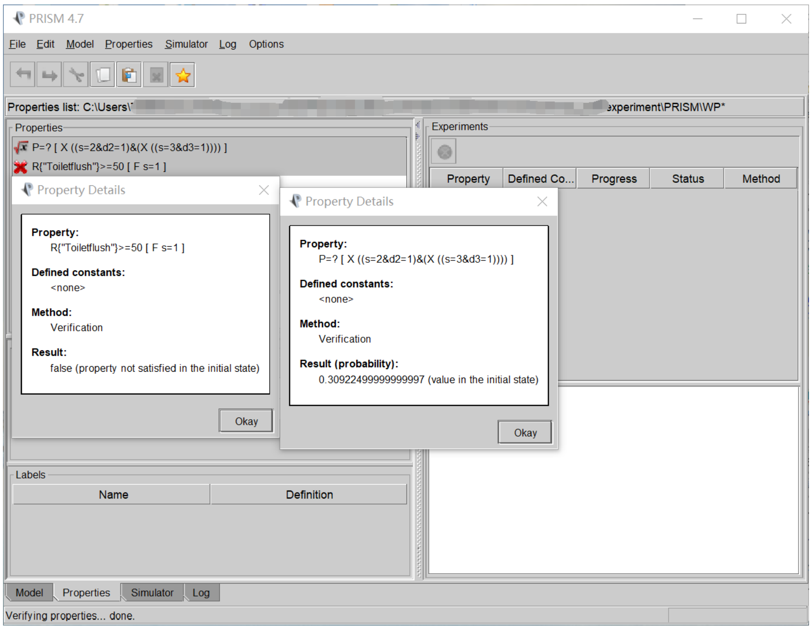


Figure 8: Verification for behaviour properties in washroom.

$$P = ?[X((s = 2\&d2 = 1)\&(X((s = 3\&d3 = 1))))]$$

Property 7. Is the number of the Toilet Flush sensor activated greater than or equal to 50? Verification shows that this property is true.

$$R\{\text{"}Toiletflush\text{"}\} >= 50[F \ s = 1]$$

Fig. 8 shows the interface in PRISM verification for the behaviour in washroom. The verification results for the two washroom-related properties are also displayed in the verification interface.

5.4. HMM-based behaviour analyses

Three arbitrary cases were provided at the beginning of this section. Each case has a set of behavioural sequences that complement it. Here, we will analyse each set of behavioural sequences.

Behavioural Sequence 1 (**BeS1**). In the morning, the inhabitant wake up, walked out of the bedroom (activated Hall-Bedroom Door sensor), entered the washroom (activated Hall-Bathroom Door sensor), used the toilet flush (activated Toilet Flush sensor), walked out of the washroom (activated Hall-Bathroom Door sensor), and then walked into the kitchen. After walking into the kitchen, he opened the fridge (activated the Fridge sensor), took the bread, opened the plates cupboard (activated Plates Cupboard sensor), took out a plate, put the bread on the plate, put it in the microwave oven and heated it up (activated Microwave sensor). In this process, the order of the activated sensors is,

$$BeS1 = [HS1 \rightarrow BS1 \rightarrow TS2 \rightarrow BS1 \rightarrow KS3 \rightarrow KS4 \rightarrow KS1]$$

Fig. 9 shows the structure of the state sequence 1. We can observe that the inhabitant needs to reach three spaces to complete the sequence of actions. First enter L_2, activate the HS1 sensor in the L_2. Then move to the L_4 and activate the BS1, TS2, and BS1 sensors in sequence in the L_4. Finally, enter the L_3 and activate the KS3, KS4, and KS1 sensors in sequence. There are six transition probabilities to be considered for moves in this sequence, as shown in the Table 2.

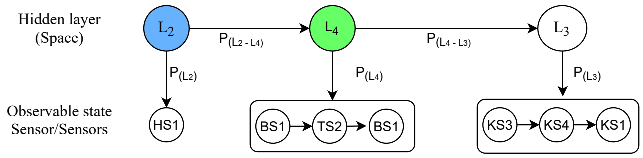


Figure 9: HMM framework for State sequence 1.

Table 2: Probability statistics for BeS1

	s_0	$s_1(P_{L2})$	$s_2(P_{L2-L4})$	$s_3(P_{L4})$	$s_4(P_{L4-L3})$	$s_5(P_{L3})$	P_{BeS1}
Value	0.091	1	0.958	0.309	0.274	1/9 * 0.010	$\prod si$

The probability values can be calculated according to the previously presented transition matrix and the property verification of behavioural transitions in space. The result of completing the sequence of behaviours at the end is the cumulative product of all probabilities.

Behavioural Sequence 2 (BeS2). When the inhabitant returned from grocery shopping at noon (activated the Frontdoor sensor), he first put the vegetable into the fridge (activated the Fridge sensor), then took out the commonly used cup from the cupboard to drink water (activated the Cup Cupboard sensor)and put the cup back into the cupboard (activated the Cup Cupboard sensor). Then walk into the washroom (activated the Hall-Bathroom Door sensor), flush the toilet (activated the Toilet Flush sensor), and walk out of the washroom (activated Hall-Bathroom Door sensor). Walk into the kitchen again and use the dishwasher to wash the dinner plates (activated the Dishwasher sensor). In this process, the order of the activated sensors is,

$$BeS2 = [FS1 \rightarrow KS3 \rightarrow KS2 \rightarrow KS2 \rightarrow BS1 \rightarrow TS2 \rightarrow BS1 \rightarrow KS5]$$

Fig. 10 shows the structure of the $BeS2$. We can observe that the inhabitant needs to reach four spaces to complete the sequence of actions. First enter L_1, activate the F1 sensor in the L_1. Next enter L_3, trigger the KS3, KS2, and KS2 sensors in sequence. Then move to the L_4 and activate the BS1, TS2, and BS1 sensors in sequence in the L_4. Finally, enter the L_3 and activate the KS5 sensor.

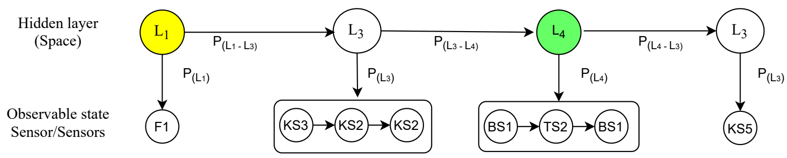


Figure 10: HMM framework for State sequence 2.

There are eight transition probabilities to be considered for moves in this sequence, as shown in the Table 3. The result of completing the sequence of actions at the end is the cumulative product of all probabilities.

Table 3: Probability statistics for BeS2

	s_0	$s_1(P_{L1})$	$s_2(P_{L1-L3})$	$s_3(P_{L3})$	$s_4(P_{L3-L4})$	$s_5(P_{L4})$	$s_6(P_{L4-L3})$	$s_7(P_{L3})$	P_{BeS2}
Value	0.076	1	1/9*0.182	1/9 * 0.0361	0.540	0.309	0.274	1/9	$\prod si$

Behavioural Sequence 3 (**BeS3**). In the evening, the inhabitant walks into the bathroom to take a shower (activated Hall-Bathroom Door sensor), uses the toilet (activated Hall-Toilet sensor), flushes the toilet (activated Toilet Flush sensor), closes the toilet (activated Hall-Toilet sensor), walks out of the washroom (activated Hall-Bathroom Door sensor) and enters the bedroom (activated Hall-Bedroom Door sensor).

$$BeS3 = [BS1 \rightarrow TS1 \rightarrow TS2 \rightarrow TS1 \rightarrow HS1]$$

Fig. 11 shows the structure of the state sequence 3. We can observe that the inhabitant requires to reach two spaces to complete the sequence of actions. First, he enters L_4, activates the BS1 sensor and takes a shower. Then, he activates the TS1, TS2, and TS1 sensors in sequence in the L_4. Finally, the users enters the L_2 and activates the HS1 sensor. There are four transition probabilities to be considered for moves in this sequence, as shown in the Table 4. The result of completing the behaviours is same as mentioned previously.

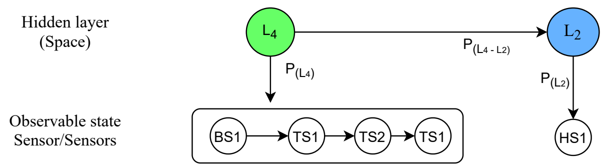


Figure 11: HMM framework for State sequence 3.

Table 4: Probability statistics for BeS3

	s_0	P_{L4}	P_{L4-L2}	P_{L2}	P_{BeS3}
Value	0.456	0.002	0.195	1	$\prod si$

Our experiments are conducted for datasets that record a single user in a smart home. When analysing behaviour in the presence of multiple users, the behavioural analysis framework proposed in this paper can be used as a basis for improvement. The main difference is data collection and organization. There are two conceptual solutions. The first one is the overall collection, which considers when there are multiple users, assuming that they are in a long-term shared relationship. The data from all the sensors in that smart home are collected over some time for behavioural analysis according to the analysis framework proposed in this paper. This approach, which is based on human inertia, is proposed. In other words, there is a pattern in their behaviours over time, whether a single user or multiple users. Even for multiple users, the overall behavioural habits stay mostly the same in the same environment. The second method involves collecting and analysing each user's data separately. This method records each user's behavioural data independently over time and analyses each user's behaviour using the methodology suggested in this paper. It uses more intelligent sensors that recognize user biometric information (such as fingerprints).

6. Discussions with anomaly detection and behaviour guidance

Our comprehensive analysis of human behaviour in the smart home is to provide a reference for the diagnosis of anomalies in the smart home. The anomalies that exist in a smart home include sensor anomalies and human physical aberrations. Typically, a person living in a fixed environment for a long time will have a certain pattern in their behavioural sequence. Therefore, if the probability of a behavioural sequence occurring in a certain time deviates significantly from the regular data and exceeds a certain threshold, it can be judged as an anomaly. This method is similar to supervised learning. It is worth noting that this anomaly can be caused by a malfunctioning sensor or a change in a person's health. It is beyond the scope of our research to trace who it is. We propose behavioural analysis methods that can provide a reference for anomaly detection. It is also possible to provide behavioural guidance for patients with memory impairment based on routine behavioural analysis.

6.1. Anomaly Detection

The identification of anomalies in this section encompasses both sensor and human anomalies. Here, a few intriguing behavioural sequence analysis-based instances are given.

6.1.1. Sensor Fault detection

The most prevalent failures of binary sensors used in smart homes fall into two categories; one is when the battery is depleted and no data is output. The second category is the daily failure, where the sensor appears to be reading normally, but the actual data output from the sensor is abnormal [51]. Here we will only discuss the first scenario. This paper assumes Cups Cupboard sensor's battery is exhausted in search of a better explanation. Then, even if the resident triggers it, it will only display 0 for a while. In the model for sensors' transfer in the kitchen, we need to change all of the values of d2 to 0. At this point the $KS3 \rightarrow KS2 \rightarrow KS2$ part of the $SS2$ is verified based on the following PCTL formula,

$$P =?[X((s = 2\&d2 = 1)\&(X((s = 2\&d2 = 1))))]$$

The result obtained was 0. If this happens in practice, it can be determined that the sensor's battery is depleted. An alarm in the smart home can send an alert so that the inhabitant can replenish the battery in time.

6.1.2. Health judgement

According to the typical pattern, human behaviour will form certain habits throughout life. For example, some people will go to the bathroom before the bedroom, while others may drink a bottle of milk before going to the bedroom. Habits formed over a long time will mostly stay the same for a short time. Therefore, we can determine whether there are abnormalities in the health of the inhabitants through the analysis of behavioural drift.

Behavioural drift [52] can be understood as a significant difference between the inhabitant's behaviour and daily behaviour over a while. Under the assumption that the sensor network is fault-free, such behavioural drift may be due to the mechanical abnormality or emotion of the inhabitant. Especially for inhabitants with physical disabilities, it will help family members to give help timely if the abnormal behaviour can be identified.

When determining behavioural drift, a threshold value can be given according to the conventional practice [52]. Once the probability of state transfer or the frequency of behaviour occurrence is greater than a certain threshold value, it can be judged as an abnormal event. The alert system will automatically send an alert prompt to the guardian at this time. Experts usually select the threshold value based on experience and knowledge. We will thoroughly analyse the obtained behavioural data and make a judgment based on the threshold value.

We consider that the inhabitant repeatedly performs a certain behaviour over a while, activating a certain sensor. For example, when the inhabitant suffers a temporary memory loss, he opens the fridge and then repeats the action of opening and closing it because he forgets the motivation.

In this case, the currently activated sensor is the Fridge sensor, and the probability that the next activated sensor will be the Fridge sensor in $t \in \{t_i...t_j\}$ is 1. The result obtained after verification of Property 3 at this point is true, which means that he has continuously activated the Fridge sensor more than 50 times and finally activating the Dishwasher sensor. The sensor network in a smart home can be set up to alert both the guardian and the inhabitant when a behaviour drifts and is determined to be abnormal.

6.2. Behavioural guidance

Amnesia is a very common mental illness, with Alzheimer's disease often occurring among middle-aged and older adults. The most obvious symptom of the disease is transient amnesia, i.e., forgetting the behaviour one is supposed to do at a certain moment [53]. IoT-based smart homes provide greater convenience for people living alone, and collecting data on the inhabitants' behaviour in the smart home can provide behavioural guidance in the event of transient amnesia [54]. In this case, probabilistic model checking provides a great help for what we want to do. Add the variable i to the logical equation, indicating the missing part of the memory. Experimentally, calculate the probability of occurrence of the sequence of behaviours under i taking all available values. Finally, comparing the results, the value with the highest probability of occurrence can be selected as the value of i taken to perform the corresponding behaviour. For better illustration, here we list 3 properties with parameters.

Property 8. What is the probability of entering the room x from the initial state (in outdoor), entering the room y at the next time, and then entering the washroom. The PCTL that represents the Property 8 is,

$$P =?[X(s = x\&(X(s = y\&(X(s = 4)))))]$$

Table 5 shows the experimental results. Columns 2-5 in the table represent the verification results when x, y take the corresponding values. It can be seen that the probability is highest when x=4 and y=3. when the inhabitant is currently in the outdoor that he wants to enter the washroom after passing through two rooms. Then the most likely order of passage is, outdoor to washroom to kitchen to washroom. In addition, the following columns 6-9 are the corresponding time for model checking. It can be found that the verification is very fast. The behaviour analyst only needs to input the specified behaviour sequence into the model checker after expressing it with the PCTL formula, the verification is completed automatically, and the whole process is extremely efficient.

Table 5: Verification results for property 8 in the PRISM

No	Verification Results				Time for model checking (seconds)			
	y=1	y=2	y=3	y=4	y=1	y=2	y=3	y=4
x=1	6.822E-4	8.622E-4	0.00415	0.00853	0.003	0.009	0.004	0.002
x=2	0	0	9.576E-4	0.01078	0.002	0.001	0.009	0.007
x=3	0.00276	0	0.03043	0.05187	0.002	0.008	0.002	0.002
x=4	0.13100	0.11764	**0.13538**	0.10659	0.002	0.001	0.002	0.002

Property 9. What is the probability of triggering the sensor a ($a \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$) from the initial state, triggering the Dishwasher sensor at next time. The initial state is 3, which means the active sensor is Fridge sensor.

$$P = ?[X(s = a\&(X(s = 5))))]$$

From the verification results that shown in Table 6, it can be observed that the probability is the largest when a=3. The model checking times for the relevant properties are also listed in the table. Simulation experiments were completed in PRISM and Fig. 12 shows the verification results for this property. Through this, it can be judged that if the actor performs another action between the Fridge sensor and the Dishwasher sensor, the most likely is that the Fridge sensor is triggered again. This seems reasonable since we normally close the fridge just in time when we open it.

Table 6: Verification results for property 9 in the PRISM

No	Verification Results	Time for model checking (Seconds)
a=1	0.00269	0.008
a=2	0.00210	0.011
a=3	**0.00931**	0.006
a=4	0	0.003
a=5	0.00698	0.003
a=6	0	0.004
a=7	0.00451	0.006
a=8	0.00270	0.008
a=9	0.00179	0.021

Property 10. What is the probability of triggering the sensor b ($b \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$) from the initial state, triggering the sensor c ($c \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$) at next time, and then triggering the Dishwasher sensor. The initial state is 3, which means the active sensor is the Fridge sensor.

$$P = ?[X(s = b\&(X(s = c\&(X(s = 5)))))]$$

Table 7 shows the probabilistic model checking for Property 10. Note that this table identifies the verification time below the corresponding one for each verification result. The results show that the maximum value is obtained when b=3 and c=3. Fig. 13 shows the simulation that developed in PRISM. We can assume that this behaviour sequence is most likely to occur. Therefore, if the inhabitant completes activating the Fridge sensor and activates the Dishwasher sensor after both behaviours have been completed, consider that the two intermediate behaviours are both activating the Fridge sensor.

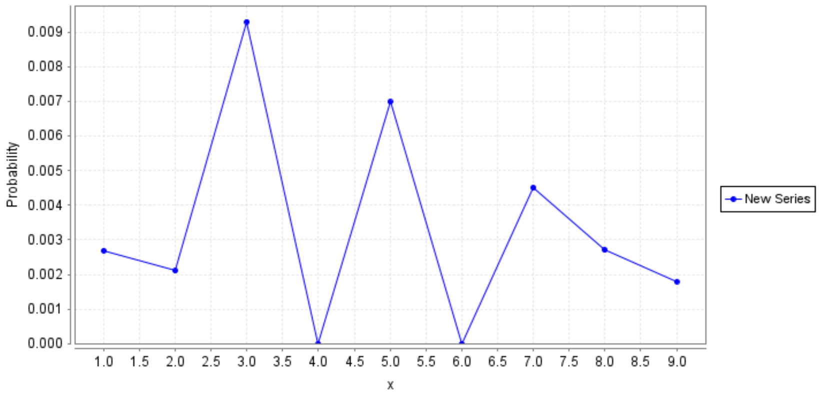


Figure 12: Experiment for Property 9 in PRISM.

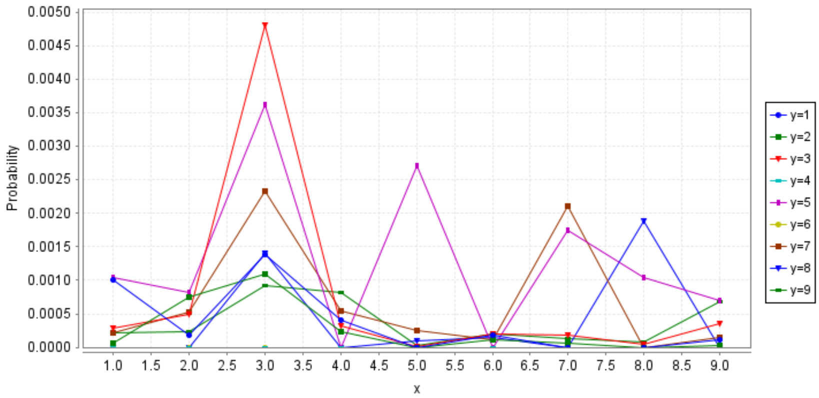


Figure 13: Experiment for Property 10 in PRISM.

7. Conclusions

Based on HMM and probabilistic model checking, this paper proposed a hierarchical framework for analysing human behaviour. Firstly, the behavioural sequences to be analysed were stratified according to space and sensors. Where the space location is the state of the hidden layer, and the sensors' state is the state included in the observable layer. DTMC models were developed for the hidden and observable layers and were introduced into PRISM in a specific format. The constructed models were verified to check whether the models satisfy the required properties. Finally, the HMM layering and probabilistic model checking results were integrated to calculate the probability of occurrence of the corresponding behavioural sequence.

Practically, the most difficult part of this study was the hierarchical partitioning of the behavioural sequence analysis and the combined analysis of HMM and probabilistic model checking. The proposed framework allows for a clear and effective analysis of the structure of the behavioural sequences inhabiting a smart home and the occurrence likelihood.

In addition, we also discussed the implications of sound behavioural sequence analysis for

Table 7: Verification results for the Property 10 in the PRISM

No	Verification Results (Time for model checking (Seconds))								
	c=1	*c*=2	*c*=3	*c*=4	*c*=5	*c*=6	*c*=7	*c*=8	*c*=9
b=1	0.00101 (0.004)	5.645E-5 (0.005)	2.880E-4 (0.004)	0 (0.003)	**0.00104** (0.004)	0 (0.003)	2.204E-4 (0.008)	0 (0.003)	2.24E-4 (0.004)
b=2	1.806E-4 (0.004)	7.581E-4 (0.005)	4.986E-4 (0.004)	0 (0.003)	**8.148E − 4** (0.01)	0 (0.003)	5.248E-4 (0.005)	0 (0.002)	2.380E-4 (0.005)
b=3	0.00139 (0.01)	0.00109 (0.005)	0.00481 (0.004)	0 (0.003)	0.00361 (0.004)	0 (0.004)	0.00233 (0.005)	0.00140 (0.006)	9.265E-4 (0.005)
b=4	4.114E-4 (0.005)	2.415E-4 (0.006)	3.240E-4 (0.005)	0 (0.002)	0 (0.003)	0 (0.003)	5.412E-5 (0.005)	0 (0.002)	**8.260E − 4** (0.007)
b=5	0 (0.002)	0 (0.002)	1.814E-5 (0.005)	0 (0.003)	**0.00271** (0.004)	0 (0.002)	2.465E-4 (0.003)	1.008E-4 (0.004)	2.822E-5 (0.004)
b=6	1.825E-4 (0.006)	1.213E-4 (0.008)	**2.089E − 4** (0.008)	0 (0.003)	0 (0.003)	0 (0.002)	1.173E-4 (0.005)	1.430E-4 (0.005)	2.033E-4 (0.005)
b=7	0 (0.003)	7.046E-5 (0.007)	1.822E-4 (0.007)	0 (0.003)	0.00175 (0.004)	0 (0.002)	**0.00212** (0.003)	0 (0.002)	1.263E-4 (0.008)
b=8	0 (0.003)	0 (0.013)	4.860E-5 (0.006)	0 (0.002)	0.00105 (0.003)	0 (0.002)	0 (0.002)	**0.00189** (0.003)	7.560E-5 (0.006)
b=9	1.129E-4 (0.007)	3.763E-5 (0.006)	3.571E-4 (0.007)	0 (0.003)	**6.953E − 4** (0.01)	0 (0.004)	1.469E-4 (0.004)	0 (0.003)	6.810E-4 (0.003)

anomaly detection and behavioural guidance at the end of the paper. The objective of the smart home is to improve human living, particularly for lone inhabitants with inadequate living skills. Behavioural analysis can explore the behavioural habits of the inhabitants so that abnormalities can be detected in time. These anomalies could result from malfunctioning sensors or indicate changes in the person's health. Abnormalities can be quickly identified by employing excellent behavioural analysis, maximizing the efficiency of the smart home. On the other hand, we represented behavioural sequences based on a parametric PCTL formula with parameters instead of uncertain values in the formula. In the verification, the parameter with the highest probability value was taken by an ergodic check, which can thus be helpful for inhabitants who want behavioural guidance.

This paper focused on analysing behaviour sequences, and the extended application discussed was relatively simple. For example, when considering anomaly detection, the situation of sensor failure or abnormal human health was considered separately. However, the situation where the two situations occurred at the same time was not considered. We assumed that the activity was completed in a specific space to use the transfer of the activity to infer the corresponding space transfer. These assumptions were idealized scenarios, while the actual sequence of actions is more complex. Further in-depth analysis is required in future research. The proposed method's influence on accurately of anomaly detection and behavioural guidance will be demonstrated in future research. Furthermore, we will also consider how action recognition of multiple inhabitants can be accomplished based on behavioural sequence analysis.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (No. 61976130, 62206227), the Chengdu International Science Cooperation Project under Grant 2020-GH02-00064-HZ, and China Scholarship Council.

Appendix

Listing 3: The DTMC of the sensor transition in kitchen

```
dtmc
module state transition in kitchen
s:[1..9] init 3;
d1: [0..1] init 0;
d2: [0..1] init 0;
d3: [0..1] init 1;
d4: [0..1] init 0;
d5: [0..1] init 0;
d6: [0..1] init 0;
d7: [0..1] init 0;
d8: [0..1] init 0;
d9: [0..1] init 0;
[] s=1->0.374:(d1'=1)&(s'=1)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)
&(d7'=0)&(d8'=0)&(d9'=0)+0.042:(d2'=1)&(s'=2)&(d1'=0)&(d3'=0)&(d4'=0)
&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.250:(d3'=1)&(s'=3)&(d2'=0)
&(d1'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.042:(d4'=1)
&(s'=4)&(d2'=0)&(d3'=0)&(d1'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)
+0.042:(d5'=1)&(s'=5)&(d2'=0)&(d3'=0)&(d4'=0)&(d1'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d9'=0)+0.083:(d6'=1)&(s'=6)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)
&(d1'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.042:(d7'=1)&(s'=7)&(d2'=0)&(d3'=0)
&(d4'=0)&(d5'=0)&(d6'=0)&(d1'=0)&(d8'=0)&(d9'=0)+0:(d8'=1)&(s'=8)
&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d1'=0)&(d9'=0)+
0.125:(d9'=1)&(s'=9)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d1'=0);
[] s=2->0.043:(d1'=1)&(s'=1)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)
&(d7'=0)&(d8'=0)&(d9'=0)+0.361:(d2'=1)&(s'=2)&(d1'=0)&(d3'=0)&(d4'=0)
&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.277:(d3'=1)&(s'=3)&(d2'=0)
&(d1'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.106:(d4'=1)
&(s'=4)&(d2'=0)&(d3'=0)&(d1'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)
+0.021:(d5'=1)&(s'=5)&(d2'=0)&(d3'=0)&(d4'=0)&(d1'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d9'=0)+0.043:(d6'=1)&(s'=6)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)
&(d1'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.064:(d7'=1)&(s'=7)&(d2'=0)&(d3'=0)
&(d4'=0)&(d5'=0)&(d6'=0)&(d1'=0)&(d8'=0)&(d9'=0)+0:(d8'=1)&(s'=8)&
(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d1'=0)&(d9'=0)+
0.085:(d9'=1)&(s'=9)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d1'=0);
[] s=3->0.064:(d1'=1)&(s'=1)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)
&(d7'=0)&(d8'=0)&(d9'=0)+0.100:(d2'=1)&(s'=2)&(d1'=0)&(d3'=0)&(d4'=0)
&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.517:(d3'=1)&(s'=3)&(d2'=0)
&(d1'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.100:(d4'=1)
&(s'=4)&(d2'=0)&(d3'=0)&(d1'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)
+0.018:(d5'=1)&(s'=5)&(d2'=0)&(d3'=0)&(d4'=0)&(d1'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d9'=0)+0.055:(d6'=1)&(s'=6)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)
&(d1'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.055:(d7'=1)&(s'=7)&(d2'=0)&(d3'=0)
&(d4'=0)&(d5'=0)&(d6'=0)&(d1'=0)&(d8'=0)&(d9'=0)+0.027:(d8'=1)&(s'=8)
&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d1'=0)&(d9'=0)+
0.064:(d9'=1)&(s'=9)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
```

```
&(d8'=0)&(d1'=0);
[]s=4->0.098:(d1'=1)&(s'=1)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)
&(d7'=0)&(d8'=0)&(d9'=0)+0.115:(d2'=1)&(s'=2)&(d1'=0)&(d3'=0)&(d4'=0)
&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.180:(d3'=1)&(s'=3)&(d2'=0)
&(d1'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.115:(d4'=1)
&(s'=4)&(d2'=0)&(d3'=0)&(d1'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)
+0:(d5'=1)&(s'=5)&(d2'=0)&(d3'=0)&(d4'=0)&(d1'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d9'=0)+0.131:(d6'=1)&(s'=6)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)
&(d1'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.066:(d7'=1)&(s'=7)&(d2'=0)&(d3'=0)
&(d4'=0)&(d5'=0)&(d6'=0)&(d1'=0)&(d8'=0)&(d9'=0)+0:(d8'=1)&(s'=8)
&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d1'=0)&(d9'=0)+
0.295:(d9'=1)&(s'=9)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d1'=0);
[]s=5->0:(d1'=1)&(s'=1)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d9'=0)+0:(d2'=1)&(s'=2)&(d1'=0)&(d3'=0)&(d4'=0)&(d5'=0)
&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.056:(d3'=1)&(s'=3)&(d2'=0)&(d1'=0)
&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.277:(d4'=1)&(s'=4)
&(d2'=0)&(d3'=0)&(d1'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)
+0.388:(d5'=1)&(s'=5)&(d2'=0)&(d3'=0)&(d4'=0)&(d1'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d9'=0)+0:(d6'=1)&(s'=6)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)
&(d1'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.167:(d7'=1)&(s'=7)&(d2'=0)&(d3'=0)
&(d4'=0)&(d5'=0)&(d6'=0)&(d1'=0)&(d8'=0)&(d9'=0)+0.056:(d8'=1)&(s'=8)
&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d1'=0)&(d9'=0)
+0.056:(d9'=1)&(s'=9)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)& (d1'=0);
[]s=6->0.079:(d1'=1)&(s'=1)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)
&(d7'=0)&(d8'=0)&(d9'=0)+0.105:(d2'=1)&(s'=2)&(d1'=0)&(d3'=0)&(d4'=0)
&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.211:(d3'=1)&(s'=3)&(d2'=0)
&(d1'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.184:(d4'=1)
&(s'=4)&(d2'=0)&(d3'=0)&(d1'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)
&(d9'=0)+0:(d5'=1)&(s'=5)&(d2'=0)&(d3'=0)&(d4'=0)&(d1'=0)&(d6'=0)
&(d7'=0)&(d8'=0)&(d9'=0)+0.237:(d6'=1)&(s'=6)&(d2'=0)&(d3'=0)&(d4'=0)
&(d5'=0)&(d1'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.026:(d7'=1)&(s'=7)&(d2'=0)
&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d1'=0)&(d8'=0)&(d9'=0)+0.026:(d8'=1)
&(s'=8)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d1'=0)&(d9'=0)
+0.132:(d9'=1)&(s'=9)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d1'=0);
[]s=7->0:(d1'=1)&(s'=1)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d9'=0)+0.061:(d2'=1)&(s'=2)&(d1'=0)&(d3'=0)&(d4'=0)&(d5'=0)
&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.184:(d3'=1)&(s'=3)&(d2'=0)&(d1'=0)
&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.122:(d4'=1)&(s'=4)
&(d2'=0)&(d3'=0)&(d1'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)
+0.082:(d5'=1)&(s'=5)&(d2'=0)&(d3'=0)&(d4'=0)&(d1'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d9'=0)+0:(d6'=1)&(s'=6)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)
&(d1'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.469:(d7'=1)&(s'=7)&(d2'=0)&(d3'=0)
&(d4'=0)&(d5'=0)&(d6'=0)&(d1'=0)&(d8'=0)&(d9'=0)+0:(d8'=1)&(s'=8)
&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d1'=0)&(d9'=0)
+0.082:(d9'=1)&(s'=9)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d1'=0);
[]s=8->0:(d1'=1)&(s'=1)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d9'=0)+0:(d2'=1)&(s'=2)&(d1'=0)&(d3'=0)&(d4'=0)&(d5'=0)
```

```
&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.100:(d3'=1)&(s'=3)&(d2'=0)&(d1'=0)
&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0:(d4'=1)&(s'=4)
&(d2'=0)&(d3'=0)&(d1'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)
+0.100:(d5'=1)&(s'=5)&(d2'=0)&(d3'=0)&(d4'=0)&(d1'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d9'=0)+0:(d6'=1)&(s'=6)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)
&(d1'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0:(d7'=1)&(s'=7)&(d2'=0)&(d3'=0)
&(d4'=0)&(d5'=0)&(d6'=0)&(d1'=0)&(d8'=0)&(d9'=0)+0.700:(d8'=1)&(s'=8)
&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d1'=0)&(d9'=0)
+0.100:(d9'=1)&(s'=9)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d1'=0);
[] s=9->0.042:(d1'=1)&(s'=1)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)
&(d7'=0)&(d8'=0)&(d9'=0)+0.028:(d2'=1)&(s'=2)&(d1'=0)&(d3'=0)&(d4'=0)
&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.310:(d3'=1)&(s'=3)&(d2'=0)
&(d1'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.085:(d4'=1)
&(s'=4)&(d2'=0)&(d3'=0)&(d1'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d8'=0)
&(d9'=0)+0.028:(d5'=1)&(s'=5)&(d2'=0)&(d3'=0)&(d4'=0)&(d1'=0)&(d6'=0)
&(d7'=0)&(d8'=0)&(d9'=0)+0.099:(d6'=1)&(s'=6)&(d2'=0)&(d3'=0)&(d4'=0)
&(d5'=0)&(d1'=0)&(d7'=0)&(d8'=0)&(d9'=0)+0.028:(d7'=1)&(s'=7)&(d2'=0)
&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d1'=0)&(d8'=0)&(d9'=0)+0:(d8'=1)
&(s'=8)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)&(d1'=0)&(d9'=0)
+0.380:(d9'=1)&(s'=9)&(d2'=0)&(d3'=0)&(d4'=0)&(d5'=0)&(d6'=0)&(d7'=0)
&(d8'=0)&(d1'=0);
endmodule
rewards "Microwave"
d1=1:1;
endrewards
rewards "CupsCupboard"
d2=1:1;
endrewards
rewards "Fridge"
d3=1:1;
endrewards
rewards "PlatesCupboard"
d4=1:1;
endrewards
rewards "Dishwasher"
d5=1:1;
endrewards
rewards "Freezer"
d6=1:1;
endrewards
rewards "PansCupboard"
d7=1:1;
endrewards
rewards "WashingMachine"
d8=1:1;
endrewards
rewards "GroceriesCupboard"
d9=1:1;
endrewards
```

References

[1] M. Pham, Y. Mengistu, H. Do, W. Sheng, Delivering home healthcare through a cloud-based smart home environment (coshe), Future Generation Computer Systems 81 (2018) 129–140.

[2] M. Talal, A. Zaidan, B. Zaidan, A. S. Albahri, A. H. Alamoodi, O. S. Albahri, M. Alsalem, C. K. Lim, K. L. Tan, W. Shir, et al., Smart home-based iot for real-time and secure remote health monitoring of triage and priority system using body sensors: Multi-driven systematic review, Journal of medical systems 43 (3) (2019) 1–34.

[3] Y. Zhang, G. Tian, S. Zhang, C. Li, A knowledge-based approach for multiagent collaboration in smart home: From activity recognition to guidance service, IEEE Transactions on Instrumentation and Measurement 69 (2) (2019) 317–329.

[4] A. A. Chaaraoui, P. Climent-Pérez, F. Flórez-Revuelta, A review on vision techniques applied to human behaviour analysis for ambient-assisted living, Expert Systems with Applications 39 (12) (2012) 10873–10888.

[5] S. Lühr, G. West, S. Venkatesh, Recognition of emergent human behaviour in a smart home: A data mining approach, Pervasive and Mobile Computing 3 (2) (2007) 95–116.

[6] Y. Liu, L. Nie, L. Liu, D. S. Rosenblum, From action to activity: sensor-based activity recognition, Neurocomputing 181 (2016) 108–115.

[7] H. Jia, S. Chen, Integrated data and knowledge driven methodology for human activity recognition, Information Sciences 536 (2020) 409–430.

[8] P. Bjesse, What is formal verification?, ACM SIGDA Newsletter 35 (24) (2005) 1–es.

[9] E. M. Clarke, Model checking, in: International Conference on Foundations of Software Technology and Theoretical Computer Science, Springer, 1997, pp. 54–56.

[10] J. A. Whittaker, M. G. Thomason, A markov chain model for statistical software testing, IEEE Transactions on Software engineering 20 (10) (1994) 812–824.

[11] K. Sultan, J. Bentahar, M. El-Menshawy, Model checking probabilistic social commitments for intelligent agent communication, Applied Soft Computing 22 (2014) 397–409.

[12] M. Kwiatkowska, G. Norman, D. Parker, Probabilistic model checking: Advances and applications, Formal System Verification (2018) 73–121.

[13] M. Kwiatkowska, G. Norman, D. Parker, Prism 4.0: Verification of probabilistic real-time systems, in: International conference on computer aided verification, Springer, 2011, pp. 585–591.

[14] S. R. Eddy, Hidden markov models, Current opinion in structural biology 6 (3) (1996) 361–365.

[15] J. Yamato, J. Ohya, K. Ishii, Recognizing human action in time-sequential images using hidden markov model., in: CVPR, Vol. 92, 1992, pp. 379–385.

[16] S. Fine, Y. Singer, N. Tishby, The hierarchical hidden markov model: Analysis and applications, Machine learning 32 (1) (1998) 41–62.

[17] J. Quintas, K. Khoshhal, H. Aliakbarpour, M. Hofmann, J. Dias, Using concurrent hidden markov models to analyze human behaviours in a smart home environment, in: Proc. 12th Intern. Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS) Delft, The Netherlands, 2011.

[18] J. Liisberg, J. K. Møller, H. Bloem, J. Cipriano, G. Mor, H. Madsen, Hidden markov models for indirect classification of occupant behaviour, Sustainable Cities and Society 27 (2016) 83–98.

[19] V. G. Sánchez, O. M. Lysaker, N.-O. Skeie, Human behaviour modelling for welfare technology using hidden markov models, Pattern Recognition Letters 137 (2020) 71–79.

[20] L. Wang, Y. Zhou, R. Li, L. Ding, A fusion of a deep neural network and a hidden markov model to recognize the multiclass abnormal behavior of elderly people, Knowledge-Based Systems 252 (2022) 109351.

[21] S. K. Das, D. J. Cook, A. Battacharya, E. O. Heierman, T.-Y. Lin, The role of prediction algorithms in the mavhome smart home architecture, IEEE Wireless Communications 9 (6) (2002) 77–84.

[22] V. Rialle, F. Duchene, N. Noury, L. Bajolle, J. Demongeot, Health" smart" home: information technology for patients at home, Telemedicine Journal and E-Health 8 (4) (2002) 395–409.

[23] A. Hong, C. Nam, S. Kim, What will be the possible barriers to consumers' adoption of smart home services?, Telecommunications Policy 44 (2) (2020) 101867.

[24] S. N. Makhadmeh, M. A. Al-Betar, Z. A. A. Alyasseri, A. K. Abasi, A. T. Khader, R. Damaševičius, M. A. Mohammed, K. H. Abdulkareem, Smart home battery for the multi-objective power scheduling problem in a smart home using grey wolf optimizer, Electronics 10 (4) (2021) 447.

[25] E. Ramanujam, T. Perumal, S. Padmavathi, Human activity recognition with smartphone and wearable sensors using deep learning techniques: A review, IEEE Sensors Journal 21 (12) (2021) 13029–13040.

[26] J. Lundström, E. Järpe, A. Verikas, Detecting and exploring deviating behaviour of smart home residents, Expert Systems with Applications 55 (2016) 429–440.

[27] X. Li, Y. He, F. Fioranelli, X. Jing, Semisupervised human activity recognition with radar micro-doppler signatures, IEEE Transactions on Geoscience and Remote Sensing 60 (2021) 1–12.

[28] S. Birnbach, S. Eberz, I. Martinovic, Haunted house: Physical smart home event verification in the presence of compromised sensors, ACM Transactions on Internet of Things 3 (3) (2022) 1–28.
[29] L. Achaji, M. Daher, M. E. B. El Najjar, F. Charpillet, Multi-sensor data fusion for smart home reliable pedestrian localization, in: 2021 IEEE 3rd International Multidisciplinary Conference on Engineering Technology (IMCET), IEEE, 2021, pp. 144–149.
[30] S. J. Moore, C. D. Nugent, S. Zhang, I. Cleland, S. Sani, A. Healing, A markov model to detect sensor failure in iot environments, in: 2020 IEEE World Congress on Services (SERVICES), IEEE, 2020, pp. 13–16.
[31] L.-H. Yang, J. Liu, Y.-M. Wang, C. Nugent, L. Martínez, Online updating extended belief rule-based system for sensor-based activity recognition, Expert Systems with Applications 186 (2021) 115737.
[32] Y. Li, G. Yang, Z. Su, S. Li, Y. Wang, Human activity recognition based on multienvironment sensor data, Information Fusion.
[33] J. Saives, C. Pianon, G. Faraut, Activity discovery and detection of behavioral deviations of an inhabitant from binary sensors, IEEE Transactions on Automation Science & Engineering 12 (4) (2015) 1211–1224.
[34] L. Wang, F. Cai, Reliability analysis for flight control systems using probabilistic model checking, in: 2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS).
[35] M. Gao, K. Wang, L. He, Probabilistic model checking and scheduling implementation of an energy router system in energy internet for green cities, IEEE Transactions on Industrial Informatics 14 (4) (2018) 1501–1510.
[36] A. Baouya, O. A. Mohamed, S. Ouchani, D. Bennouar, Reliability-driven automotive software deployment based on a parametrizable probabilistic model checking, Expert Systems with Applications 174 (2021) 114572.
[37] T. l'Yvonnet, E. De Maria, S. Moisan, J.-P. Rigault, Probabilistic model checking for human activity recognition in medical serious games, Science of Computer Programming 206 (2021) 102629.
[38] H. Gao, L. Zhou, J. Y. Kim, Y. Li, W. Huang, The behavior guidance and abnormality detection for a-mci patients under wireless sensor network, ACM Transactions on Sensor Networks (2021).
[39] X. Wang, J. Liu, C. Nugent, S. J. Moore, Y. Xu, Reliability analysis of smart home sensor systems based on probabilistic model checking, in: 2022 IEEE 15th International Conference on Automation Science and Engineering (CASE)Proceedings of the International Conference on Ubiquitous Computing & Ambient Intelligence, UCAmI 2022, Córdoba, Spain, 29 November - 2 December 2022, 2022.
[40] V. K. Tim, N. Athanasios, E. Gwenn, K. Ben, Accurate activity recognition in a home setting, in: Proceedings of the 10th international conference on Ubiquitous computing, 2008, pp. 1–9.
[41] S. N. Ethier, T. G. Kurtz, Markov processes: characterization and convergence, John Wiley & Sons, 2009.
[42] M. A. Shahab, M. U. Iqbal, B. Srinivasan, R. Srinivasan, Hmm-based models of control room operator's cognition during process abnormalities. 1. formalism and model identification, Journal of Loss Prevention in the Process Industries 76 (2022) 104748.
[43] H. Gao, S. Mao, W. Huang, X. Yang, Applying probabilistic model checking to financial production risk evaluation and control: A case study of alibabas yue bao, IEEE Transactions on Computational Social Systems 5 (3) (2018) 785–795.
[44] M. Duflot, M. Kwiatkowska, G. Norman, D. Parker, S. Peyronnet, C. Picaronny, J. Sproston, Practical applications of probabilistic model checking to communication protocols (2012).
[45] S. Basagiannis, P. Katsaros, A. Pombortsis, N. Alexiou, Probabilistic model checking for the quantification of dos security threats, Computers & Security 28 (6) (2009) 450–465.
[46] H. Hansson, B. Jonsson, A logic for reasoning about time and reliability, Formal aspects of computing 6 (5) (1994) 512–535.
[47] F. Ciesinski, M. Größer, On probabilistic computation tree logic, in: Validation of Stochastic Systems, Springer, 2004, pp. 147–188.
[48] J. Desharnais, P. Panangaden, Continuous stochastic logic characterizes bisimulation of continuous-time markov processes, The Journal of Logic and Algebraic Programming 56 (1-2) (2003) 99–115.
[49] N. J. Nilsson, Probabilistic logic, Artificial intelligence 28 (1) (1986) 71–87.
[50] M. Kwiatkowska, G. Norman, D. Parker, Stochastic model checking, in: International School on Formal Methods for the Design of Computer, Communication and Software Systems, Springer, 2007, pp. 220–270.
[51] S. J. Moore, C. D. Nugent, S. Zhang, I. Cleland, Iot reliability: a review leading to 5 key research directions, CCF Transactions on Pervasive Computing and Interaction 2 (3) (2020) 147–163.
[52] C. Azefack, R. Phan, V. Augusto, G. Gardin, C. Coquard, R. Bouvire, X. Xie, An approach for behavioral drift detection in a smart home, in: 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE), 2019.
[53] W. R. Russell, P. Nathan, Traumatic amnesia, Brain 69 (1946) 280–300.
[54] P. H. Vardhini, M. S. Harsha, P. N. Sai, P. Srikanth, Iot based smart medicine assistive system for memory impairment patient, in: 2020 12th international conference on computational intelligence and communication networks (CICN), IEEE, 2020, pp. 182–186.