Liu L, Hasan O, Tahar S. Formal reasoning about finite-state discrete-time Markov chains in HOL. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 28(2): 217–231 Mar. 2013. DOI 10.1007/s11390-013-1324-6

Formal Reasoning About Finite-State Discrete-Time Markov Chains in HOL

Liya Liu, *Student Member*, *IEEE*, Osman Hasan, and Sofiène Tahar, *Senior Member*, *ACM*, *IEEE*

Department of Electrical and Computer Engineering, Concordia University, Montreal, Canada

E-mail: {liy_liu, o_hasan, tahar}@ece.concordia.ca

Received January 6, 2012; revised September 11, 2012.

Abstract Markov chains are extensively used in modeling different aspects of engineering and scientific systems, such as performance of algorithms and reliability of systems. Different techniques have been developed for analyzing Markovian models, for example, Markov Chain Monte Carlo based simulation, Markov Analyzer, and more recently probabilistic model-checking. However, these techniques either do not guarantee accurate analysis or are not scalable. Higher-order-logic theorem proving is a formal method that has the ability to overcome the above mentioned limitations. However, it is not mature enough to handle all sorts of Markovian models. In this paper, we propose a formalization of Discrete-Time Markov Chain (DTMC) that facilitates formal reasoning about time-homogeneous finite-state discrete-time Markov chain. In particular, we provide a formal verification on some of its important properties, such as joint probabilities, Chapman-Kolmogorov equation, reversibility property, using higher-order logic. To demonstrate the usefulness of our work, we analyze two applications: a simplified binary communication channel and the Automatic Mail Quality Measurement protocol.

Keywords discrete-time Markov chain, higher-order logic, probability theory, theorem prover

1 Introduction

In our daily life, most of the natural phenomena are random or unpredictable. To quantify the possibility of the appearance of random events, probability theory has been built up as an important branch of mathematics for probabilistic analysis of the random phenomena. As we know, the majority of the randomness has some sort of time-dependency. For example, noise signals vary with time, the duration of a telephone call is somehow related to the time it is made, population growth is time dependant and so is the case with chemical reactions. Such random processes usually exhibit the memoryless property[1], which means that the future state depends only on the current state and is independent of any past state. The random processes possessing such a memoryless property, also called Markov property, are Markov processes. The study of Markov process[1], which is a sub-branch of probability theory, is extensively investigated and applied for analyzing systems in many different fields of science and engineering. Some of their important applications include functional correctness and performance analysis of telecommunication and security protocols, reliability analysis of hardware circuits, software testing, Internet page ranking and statistical mechanics.

Traditionally, simulation is the most commonly used computer-based analysis technique for Markovian models. A typical example using this technique is applying Markov Chain Monte Carlo (MCMC) methods[2], which involve sampling from the desired probability distributions by constructing a Markov chain with the desired distribution. Although some sophisticated MCMC-based algorithms are capable of producing exact samples in order to improve the accuracy of results, in general the analysis can never be termed as 100% precise due to the inaccurate nature of simulation. Inaccurate results, however, pose a serious threat in highly sensitive and safety critical applications, such as, nuclear reactor control and aerospace software engineering. On the other hand, the additional computation and unbounded running time introduced by these complex algorithms are generally not acceptable due to the increasingly shorter time-to-market and high productivity increase requirements.

Other state-based approaches to analyzing Markovian models include software packages, such as Markov analyzers and reliability or performance evaluation tools, which are all based on numerical methods[3]. Although these software packages can be successfully applied to analyze large-scale Markovian models, the results cannot be guaranteed to be accurate because the underlying iterative calculation are not 100% precise. Another technique, Stochastic Petri Nets (SPN)[4], has

Regular Paper

been found as a powerful method for modeling and analyzing Markovian systems because it allows local state modeling instead of global modeling. The key limiting factor of the application of SPN models using this approach is the complexity of their analysis.

Formal methods provide effective solutions to solve the inaccuracy problem mentioned above. Due to the extensive usage of Markov chains in analyzing safety-critical systems, probabilistic model checking[5] has been recently proposed for analyzing systems that can be abstracted as Markovian models. Probabilistic model checking tools are able to be used to conduct precise system analysis by modeling the system behaviors, including the random components in a precise logic and reasoning about the probabilistic properties of the system. This technique offers exact solutions but is limited by the state-space explosion problem[6] and the time of analyzing some of the safety properties of a system is largely dependent on the convergence speed of the underlying algorithms. Similarly, we cannot verify generic mathematical expressions for probabilistic analysis using probabilistic model checking due to the inherent state-based nature of the approach. Thus, the probabilistic model checking approach, even though is capable of providing exact solutions automatically, is quite limited in terms of supporting complicated systems and handling the accurate results of a wide variety of systems and properties.

Another formal technique, higher-order logic interactive theorem proving[7], provides a conceptually simple formalism with a precise semantics, allowing secure extensions for many mathematical theories, including some parts of the Markov chain theory[8]. Due to the highly expressive nature of higher-order logic and the inherent soundness of theorem proving, this technique is capable of providing precise analysis of all sorts of Markovian models. However, the existing higher-order-logic formalization of Markov chain theory[8] is not rich enough to handle formal reasoning about many interesting characteristics of Markovian models, such as the reversibility of a Markov chain and stationary properties. This paper presents a formalization of discrete-time Markov chain to raise the scope of formal reasoning about Markovian models in a higher-order-logic theorem prover. Particularly, we focus on formalizing time-homogeneous Discrete-Time Markov Chain (DTMC) with finite state space in higher-order logic. We also formally verify some of the fundamental properties of a DTMC, such as, Joint Probability Distribution, Chapman-Kolmogorov Equation, Reversibility of a Markov Chain, and Steady-State Probabilities[1]. These properties play a vital role in reasoning about many interesting characteristics while analyzing the Markovian models of real-world systems as well as pave the path to the verification of more advanced properties related to DTMC. Also, this foundation can be extended to formalize Markov chains with infinite state space, Continuous-Time Makrov Chains (CTMC) and Hidden Markov Chain Models (HMMs). In order to illustrate the effectiveness of our work and demonstrate its utilization, we present the formal analysis of a simplified binary communication channel and the performance of some algorithms in the Automatic Mail Quality Measurement (AMQM) system.

The rest of this paper is organized as follows. In Section 2, we present a brief review of the related work. In Section 3, we provide some preliminaries that are required to understand the formalization described in the rest of the paper. In Section 4, we will describe the proposed higher-order-logic definition of DTMC with finite state space. In Section 5, some important properties of DTMC are formally verified based on the proposed definition of DTMC. Then, in Section 6, we present two applications for illustration purposes. Finally, we conclude the paper in Section 7.

2 Related Work

As a conventional technique, simulation is very effective for industrial engineering. A large number of software tools have been developed for the analysis of Markovian systems. Due to the inherent nature of simulation, the majority of the algorithms employed in software tools provide approximate results. Markov Analyzers, such as MARCA① and DNAmaca[9], which contain numerous matrix manipulation and numerical solution procedures, are powerful autonomous tools for analyzing large-scale Markovian models. Unfortunately, most of their algorithms are based on iterative methods that begin from some initial approximation and end at some convergent point, which is the main source of inaccuracy in such methods.

Many reliability evaluation software tools integrate simulation and numerical analyzers for modeling and analyzing the reliability, maintainability or safety of systems using Markov methods. These tools offer simplistic modeling approaches and are more flexible compared to traditional approaches, such as Fault Tree[10]. Some prevalent tool examples are Möbius② and PTC Windchill Markov③. Some other software

①MARCA. www4.ncsu.edu/∼billy/MARCA/marca.html, Jan. 2012.

②Möbius. www.mobius.illinois.edu, Jan. 2012.

③PTC Windchill Markov. www.ptc.com/products/windchill/markov, Jan. 2012.

tools for evaluating performance, e.g., MACOM[11] and HYDRA[12], take the advantages of a popular Markovian algebra, i.e., PEPA④ to model systems and efficiently compute passage time densities and quantities in large-scale Markov chains. However, the algorithms used to solve the models are based on approximations, which lead to inaccuracies.

Stochastic Petri nets provide a versatile modeling technique for stochastic systems. The most popular softwares are SPNP[13] and GreatSPN⑤. These tools can model, validate, and evaluate distributed systems and analyze the dynamic events of models using distributions other than the exponential. Although they can easily manage larger system models, most of the solutions for computing the stationary probabilities of a large-scale Markov chain are based on the iterative methods or an initial approximation in order to reach the convergent point. Obviously, iterative methods introduce the approximation at different levels while calculating transient probabilities of a model and this results in inaccurate analysis.

Numerous model checking tools have been proposed in the open literature to formally analyze Markovian systems, e.g., VESTA[14] is a statistical model checker, MRMC⑥ is a tool for verifying Markov reward models, Ymer⑦ is used to verify probabilistic transient properties of Continuous-Time Markov Chains (CTMCs) and Generalized Semi-Markov Processes (GSMPs), etc. Probabilistic model checking[5,15] is the state-of-the-art formal Markov chain analysis technique. PRISM⑧ is the most popular model checking tool, which supports the analysis of probabilistic properties of DTMC, CTMC, and Markov Decision Processes (MDPs) and has been used to analyze many practical systems including communication and multimedia protocols. But model checkers suffer from state-space explosion as well as do not support the verification of generic mathematical expressions. Also, because of numerical methods implemented in these tools, the final results cannot be termed 100% accurate. Whereas, the proposed HOL theorem proving based approach is capable of specifying larger systems besides providing accurate results.

Theorem proving is an alternative formal method used for conducting formal probabilistic analysis. Using this method, the system to be analyzed is mathematically modeled in an appropriate logic and the properties of interest are mathematically verified in a computer-based formal tool. For instance, Nedzusiak[16] and Bialas[17] were among the first ones who proposed to formalize some probability theory in higher-order-logic. Hurd[18] formalized some measure theory in higher-order logic and proposed techniques to formalize discrete random variables in HOL. Then, Hasan[19] extended Hurd's work by providing the support to formalize continuous random variables and verify statistical properties, such as, expectation and variance, for both discrete and continuous random variables[20]. Recently, Mhamdi[21] proposed a significant formalization of entropy measures in HOL and presented a formalization of measure theory based on extended reals using the HOL theorem prover. Hölzl[22] has also formalized three chapters of measure theory in Isabelle/HOL. However, the work of Mhamdi and Hölzl do not include the formalization of a particular probability space and thus do not include the formal verification of distribution properties of commonly used random variables like the case of Hurd and Hasan. Random variables play a vital role in constructing Markovian models of real-world systems. Due to this reason, we built upon the work of Hurd[18] and Hasan[19] to formalize DTMC in higher-order logic and formally verify some of its properties[8]. This formalization facilitates the reasoning about some aspects of DTMC. The current paper extends this formalization by providing some additional verified stationary properties and the formalization of the reversible DTMC to reason about Markovian models. It also presents a couple of interesting case studies in order to demonstrate the usefulness of the verified DTMC properties in verifying the properties of practical systems using theorem proving.

3 Preliminaries

In this section, we provide a brief overview of the HOL theorem prover and Hurd's formalization[18] of probability theory and random variables. These fundamental concepts will be used in the rest of this paper.

3.1 HOL Theorem Prover

HOL denotes a family of interactive theorem proving systems for conducting proofs in higher-order logic by using the strongly-typed functional Meta-Language (ML)[23] or its successors. Based on the first version developed by Mike Gordon[24], HOL88, HOL90, HOL98, and HOL4 have been continuously developed. All these

④PEPA. www.dcs.ed.ac.uk/pepa, Jan. 2012.

⑤GreatSPN. www.di.unito.it/∼greatspn/index.html, Jan. 2012.

⑥MRMC. www.mrmc-tool.org/trac, Jan. 2012.

⑦Ymer. www.tempastic.org/ymer, Jan. 2011.

⑧PRISM. www.prismmodelchecker.org, Jan. 2012.

tools are using Robin Milner's Logic for Computable Functions (LCF) approach[25]. As a system of deduction with a precise semantics, HOL4 is capable of verifying a wide variety of hardware and software as well as pure mathematics due to the high expressiveness higher-order logic. One of the key principles of the HOL4 system is that its logical core consists of only five axioms and eight inference rules and all the subsequent theorems are verified based on these foundations or any other previously verified theorems. It supports both forward and backward proofs by applying tactics, which are ML functions that simplify goals into subgoals. Over the past few decades, the formalization of many foundational mathematical theories has led to tremendous progress in HOL4. For example, Harrison[26] formalized real numbers, topology, limits, sequences and series, differentiation and integration and his work is part of the current distribution of HOL. Hurd[18] developed a probability theory and Hasan[19] formalized statistical theorems for continuous random variables and their Cumulative Distribution Function (CDF) in the HOL4 system. Due to the undecidable nature of higher-order logic, the users have to verify theorems in an interactive way but in order to facilitate this process, the HOL theorem prover provides many proof assistants and automatic proof methods.

3.2 Probability Theory and Random Variables in HOL

A *measure space* is defined as a triple (Ω, Σ, μ), where Ω is a set, called the *sample space*, Σ represents a σ-algebra of subsets of Ω and the subsets are usually referred to as *measurable sets*, and μ is a *measure* with domain Σ. A *probability space* is a measure space $(\Omega, \Sigma, \mathcal{P}r)$ such that the measure, referred to as the probability and denoted by $\mathcal{P}r$, of the sample space is 1.

The measure theory developed by Hurd[18] defines a measure space as a pair (Σ, μ). Whereas the sample space, on which this pair is defined, is implicitly implied from the higher-order-logic definitions to be equal to the universal set of the appropriate data-type. Building upon this formalization, the probability space was also defined in HOL as a pair $(\mathcal{E}, \mathbb{P})$, where the domain of $\mathbb{P}$ is the set $\mathcal{E}$, which is a set of subsets of infinite Boolean sequences $\mathbb{B}^\infty$. Both $\mathbb{P}$ and $\mathcal{E}$ are defined using the Carathéodory's Extension Theorem, which ensures that $\mathcal{E}$ is a σ-algebra: closed under complements and countable unions.

Now, a random variable, which is one of the core concepts in probabilistic analysis, is a fundamental probabilistic function and thus can be modeled in higher-order logic as a deterministic function, which accepts the infinite Boolean sequence as an argument. These deterministic functions make random choices based on the result of popping the top most bit in the infinite Boolean sequence and may pop as many random bits as they need for their computation. When the functions terminate, they return the result along with the remaining portion of the infinite Boolean sequence to be used by other programs. Thus, a random variable which takes a parameter of type α and ranges over values of type β can be represented in HOL by the following function.

$$\mathcal{F} : \alpha \to B^\infty \to \beta \times B^\infty.$$

As an example, consider a Bernoulli $(\frac{1}{2})$ random variable that returns 1 or 0 with equal probability $\frac{1}{2}$. It has been formalized in higher-order logic as follows

```
∀ s. bit s =
     (if shd s then 1 else 0, stl s),
```

where the functions `shd` and `stl` are the sequence equivalents of the list operations *"head"* and *"tail"*, respectively. The function `bit` accepts the infinite Boolean sequence `s` and returns a pair. The first element of the returned pair is a random number that is either 0 or 1, depending on the Boolean value of the top most element of `s`. Whereas, the second element of the pair is the unused portion of the infinite Boolean sequence, which in this case is the tail of the sequence.

Once random variables are formalized, as mentioned above, we can utilize the formalized probability theory to reason about their probabilistic properties. For example, the following Probability Mass Function (PMF) property can be verified for the function `bit` using the HOL theorem prover:

$$\vdash \mathbb{P}\ \{\texttt{s|FST (bit s) = 1}\} = \frac{1}{2},$$

where the function `FST` selects the first component of a pair and $\{x|C(x)\}$ represents a set of all x that satisfy the condition C.

The above approach has been successfully used to formally verify most basic probability theorems[18], such as the law of additivity, and conditional probability related properties[27]. For instance, the conditional probability has been formalized as:

Definition 1 (Conditional Probability).

```
⊢ ∀A B.
     cond_prob A B = ℙ(A ⋂ B) / ℙ(B).
```

It plays a vital role in our work. Another frequently used formally verified theorem, needed for our work, is

the Total Probability Theorem[27], which is described, for a finite, mutually exclusive, and exhaustive sequence B_i of events and an event A, as follows:

$$\mathcal{P}r(A) = \sum_{i=0}^{n-1} \mathcal{P}r(B_i)\mathcal{P}r(A|B_i). \quad (1)$$

4 Formalization of DTMC in HOL

Given a probability space, a stochastic process $\{X_t, t \in T\}$ represents a sequence of random variables X, where t represents the time that can be discrete (represented by non-negative integers) or continuous (represented by real numbers)[1]. The set of values taken by each X_t, commonly called *states*, is referred to as the *state space* Ω. Now, based on these definitions, a *Markov process* can be defined as a stochastic process with the Markov property. If a Markov process has finite or countably infinite state space, then it is called a Markov chain and satisfies the following Markov property.

For all t, if state x_i $(\forall i \in [0, t+1])$ is in the state space, then

$$\begin{aligned}&\mathcal{P}r\{X_{t+1} = x_{t+1}|X_t = x_t, \ldots, X_0 = x_0\}\\ &= \mathcal{P}r\{X_{t+1} = x_{t+1}|X_t = x_t\}. \end{aligned} \quad (2)$$

Additionally, if t ranges over nonnegative integers or, in other words, the time is a discrete quantity, and the states are in a finite discrete space, then such a Markov chain is called a finite-state discrete-time Markov chain. A Markov chain[1] is referred to as the time-homogeneous Markov chain, if the conditional probability $\mathcal{P}r(X_{n+1} = a|X_n = b)$ is independent of n. Time-homogeneousity is an important concept in analyzing Markovian models and therefore, in our development, we focus on formalizing Time-Homogeneous Discrete-Time Markov Chain with finite state space, which we refer to in this paper as DTMC. A DTMC is usually expressed by specifying[28]:

• an initial distribution defined by $\forall s \in \Omega$, $\pi_0(s) = \mathcal{P}r(X_0 = s)$, $\pi_0(s) \geqslant 0$, and $\sum_{s\in\Omega} \pi_0(s) = 1$.

• transition probabilities p_{ij} defined as $\forall i, j \in \Omega$, $p_{ij} = \mathcal{P}r\{X_{t+1} = j|X_t = i\}$, $p_{ij} \geqslant 0$ and $\sum_{j\in\Omega} p_{ij} = 1$.

Based on the above mentioned definition, the notion of a DTMC in HOL can be formalized as the following predicate:

Definition 2 (DTMC).

```
⊢ ∀ X N x Linit Ltrans.
  Time_homo_mc X N x Linit Ltrans
  = (∀ i.  i < N ⇒
     (ℙ{s | FST (X 0 s) = x_i} = EL i Linit) ∧
     (∑_{k=0}^{N-1} EL k Linit = 1)) ∧
  (∀ t i j.  i < N ∧ j < N ⇒
     (ℙ{s | FST (X (t + 1) s) = x_j}|
       {s | FST (X t s) = x_i}
       = EL (i * N + j) Ltrans) ∧
     (∑_{k=0}^{N-1} EL (i * N + k) Ltrans = 1 ) ) ∧
  (∀ t k.  k < N ⇒
    measurable {s|FST (X t s) = x_k}) ∧
  (∀ t.  ⋃_{k=0}^{N-1} {s|FST (X t s)=x_k}=UNIV) ∧
  (∀t u v.  u < N ∧ v < N ∧ u ≠ v ⇒
     disjoint {s|FST (X t s) = x_u}
              {s|FST (X t s) = x_v}) ∧
  (∀ i j m r t w L Lt.
    ((∀ k.  k ⩽ r ⇒ EL k L < N) ∧
     i < N ∧ j < N ∧ Lt ⊆ [m, r] ∧
     m ⩽ r ∧ (Lt ≠ ∅ ⇒ w + r < t) ∧
    (ℙ(⋂_{kϵLt} {s | FST (X (w + k) s)
                  = x_(EL k L)}) ≠ 0) ⇒
    (ℙ({s | FST (X (t + 1) s) = x_j}|
       {{s | FST (X t s) = x_i} ⋂
       (⋂_{kϵLt} {s | FST (X (w + k) s)
                  = x_(EL k L)})})
     = ℙ({s | FST (X (t + 1) s) = x_j}|
        {s | FST (X t s)= x_i}))).
```

The function `Time_homo_mc` accepts a sequence of random variables `X`, the cardinality of the set of their possible states `N`, a function `x` that accepts the index and returns the value of the state corresponding to the given DTMC, and two real number lists: the initial states probability distribution `Linit` and the transition probabilities `Ltrans`.

The predicate `Time_homo_mc` contains the following conditions:

• The DTMC must follow the given initial distribution `Linit`, in which the summation of all the elements is 1. The transition probabilities `Ltrans`, in which the summation of each N elements is 1, is an intrinsic characteristic of a stochastic matrix. In the condition `(∀ t i j.  i < N ∧ j < N ⇒ (ℙ{s | FST (X (t + 1) s) = x_j}|{s | FST (X t s) = x_i} = EL (i * N + j) Ltrans)` it is explicit that transition probabilities are independent of time `t`, which implies the time homogeneous property.

• All events involving the Markov chain random variables are measurable `(∀ t k.  (k < N) ⇒ measurable {s | FST (X t s) = x_k})`.

• The union of all states forms the state space as a universal set `UNIV` `(∀ t.  ⋃_{k=0}^{N-1} {s | FST (X t s) = x_k} = UNIV)`.

• The fifth condition ensures that the states in the state space of a given Markov chain are mutually exclusive `(∀ t u v.  (u < N) ∧ (v < N) ∧ (u ≠ v)`

$\Rightarrow$ `disjoint ({s | FST (X t s) =` x_u`} {s | FST (X t s) =` x_v`}))`.

• The sixth condition corresponds to the memoryless property in (2). We model the history of states in our formalization by a list L, which contains the state elements ranging from 0 to $N-1$. Thus, the list L, with $r-m+1$ elements or less, represents the indices of passed states and its elements have to be less than `N` (`∀ k. (k ≤ r) ⇒ (EL k L < N)`). In ($\bigcap_{k\in Lt}$ {`s | FST (X (w + k) s) =` $x_{(EL\ k\ L)}$}), where the function (`EL k L`) returns the k-th element of the list L, it gives a general time index of every event and a flexible length of the event sequence. (`k ∈ Lt`) makes sure that the passed states can be freely chosen from a set `Lt`, which includes natural numbers and is a subset of the interval [m, r] (`Lt ⊆ [m, r]`). The condition (`Lt ≠ ∅ ⇒ w + r < t`) ensures that the states in this intersection set are past states if the considered list `Lt` is not empty. The reason why the passed states path is expressed in such a complex way is that the underlying information in the mathematic expression (2) including many cases, such as,

$$\begin{aligned}&\forall\, k \in [0,t), x_k \in \Omega\\ &\mathcal{P}r\{X_{t+1}=x_{t+1}|X_t=x_t, X_k=x_k\}\\ &=\mathcal{P}r\{X_{t+1}=x_{t+1}|X_t=x_t\}.\end{aligned}\tag{3}$$

The last condition (`ℙ(`$\bigcap_{k\epsilon Lt}$ {`s | FST (X (w + k) s) =` $x_{(EL\ k\ L)}$`}) ≠ 0`) is used to exclude the path of passed states, which do not appear in the chain.

It is important to note that the type of X is *num* $\rightarrow$ (*num* $\rightarrow$ *bool*) $\rightarrow$ *'a* $\#$ (*num* $\rightarrow$ *bool*), so the value of the state can be any type (in HOL, the arbitrary type is represented as *'a* automatically), which ranges over a sequence with type (*num* $\rightarrow$ *bool*). This makes our definition general enough to work with discrete-time random variables of any data type.

5 Verification of Discrete-Time Markov Chain Properties

In this section, we present the formal verification of some of the most important properties of time-homogeneous DTMC with finite-state space. The formal verification of these properties not only ensures the correctness of our formalization of DTMC, given in Definition 2, but also paves the path to reason about DTMC models of practical systems, as will be depicted in Section 6.

5.1 Joint Probability

The joint probability of a Markov chain defines the probability of events involving two or more random variables associated with a chain. Joint probability is very useful in analyzing multi-stage experiments, when an event chain happens. Also, this concept is the basis for joint probability generating function, which is used in many different fields. Mathematically, the joint probability of $n+1$ discrete random variables $X_0, X_1, \ldots, X_n$ in a Markov chain can be expressed as[1]:

$$\begin{aligned}&\mathcal{P}r\{X_t=x_0,\ldots,X_{t+n}=x_n\}\\ &=\Big(\prod_{k=0}^{n-1}\mathcal{P}r\{X_{t+k+1}=x_{k+1}|X_{t+k}=x_k\}\Big)\mathcal{P}r\{X_t=x_0\}.\end{aligned}\tag{4}$$

We formalize this property in HOL as the following theorem:

Theorem 1 (Joint Probability).

```
⊢ ∀ X N x t n L Linit Ltrans.
    Time_homo_mc X N x Linit Ltrans ∧
    EVERY (λa. a < N) L ∧
    n + 1 ⩽ LENGTH L ⇒
    ℙ(⋂_{k=0}^{n} {s | FST (X (t + k) s)
                     = x_(EL k L)})
    = (∏_{k=0}^{n-1} ℙ({s | FST (X (t + k + 1) s)
                          = x_(EL (k+1) L)} |
          {s | FST (X (t + k) s)
               = x_(EL k L)}))
    ℙ{s | FST (X t s) = x_(EL 0 L)}.
```

The variables above are used in the same context as Definition 2. The first assumption ensures that `X` is a Markov chain. All elements of the indices sequence `L` are less than `N` and the length of `L` is larger than or equal to the length of the segment considered in the joint events. The conclusion of the theorem represented (4) in higher-order logic based on the probability theory formalization, presented in Subsection 3.2. The proof of Theorem 1 is based on induction on the variable `n`, (1) and some arithmetic reasoning.

5.2 Chapman-Kolmogorov Equation

The Chapman-Kolmogorov equation[1] is a widely used property of time homogeneous Markov chains as it facilitates the use of a matrix theory for analyzing large Markov chains. It basically gives the probability of going from state i to j in $m+n$ steps. Assuming the first m steps take the system from state i to some intermediate state k, which is in the state space Ω and the remaining n steps then take the system from state k to j, we can obtain the desired probability by adding the probabilities associated with all the intermediate steps.

$$p_{ij}^{(m+n)}=\sum_{k\in\Omega}p_{kj}^{(n)}p_{ik}^{(m)}.\tag{5}$$

The notation $p_{ij}(n)$ denotes the n-step transition probabilities from state i to j.

$$p_{ij}^{(n)} = \mathcal{P}r\{X_{t+n} = x_j | X_t = x_i\}. \tag{6}$$

When $n = 1$, $p_{ij}^{(1)}$ is usually written as p_{ij} and (5) becomes

$$p_{ij}^{(m+1)} = \sum_{k \in \Omega} p_{kj} p_{ik}^{(m)}. \tag{7}$$

Based on (5) and Definition 2, the Chapman-Kolmogorov equation is formalized as follows.

Theorem 2 (Chapman-Kolmogorov Equation).

```
⊢ ∀ X i j x N m n Linit Ltrans.
   Time_homo_mc X N x Linit Ltrans ∧
   i < N ∧ j < N ∧
   (∀ a b.  a < N ∧ b < N ⇒
     ℙ({s | FST (X 0 s) = x_b}|
       {s | FST (X 0 s) = x_a})
     = if (a = b) then 1 else 0) ⇒
   ℙ({s | FST (X (m + n) s) = x_j}|
     {s | FST (X 0 s) = x_i})
   = ∑_{k=0}^{N-1} (ℙ({s | FST (X n s) = x_j}|
                      {s | FST (X 0 s) = x_k})
                    ℙ({s | FST (X m s) = x_k}|
                      {s | FST (X 0 s) = x_i})).
```

The variables `m` and `n` denote the steps between two states and both of them represent time. The first assumption ensures that the random process `X` is a time homogeneous DTMC, using Definition 2. The following two assumptions, `i < N` and `j < N`, define the allowable bounds for the index variables. The last assumption defines the zero-step transition probabilities to be a δ function, i.e.,

$$\delta_{ab} = \begin{cases} 1, & \text{if } a = b, \\ 0, & \text{if } a \neq b. \end{cases}$$

The conclusion of the theorem formally represents (5).

The proof of Theorem 2 again involves induction on the variable `n` and both of the base and step cases are discharged using the following lemma corresponding to (7).

Lemma 1 (Multistep Transition Probability).

```
⊢ ∀ X i j x m N Linit Ltrans.
   Time_homo_mc X N x Linit Ltrans ∧
   i < N ∧ j < N ⇒
   ℙ({s | FST (X (m + 1) s) = x_j}|
     {s | FST (X 0 s) = x_i})
   = ∑_{k=0}^{N-1} ℙ({s | FST (X 1 s) = x_j}|
                     {s | FST (X 0 s) = x_k})
                   ℙ({s | FST (X m s) = x_k}|
                     {s | FST (X 0 s) = x_i}).
```

The proof of Lemma 1 is primarily based on Definition 2 and the additivity property of conditional probabilities.

5.3 Absolute Probabilities

The unconditional probabilities associated with a Markov chain are referred to as the absolute probabilities[1]. If the initial probability distribution of the system being in a state, which has index k, is given by $\mathcal{P}r\{X_0 = x_k\}$ then the absolute probability of the system being in state j is given by

$$\begin{aligned} p_j^{(n)} &= \mathcal{P}r\{X_n = x_j\} \\ &= \sum\nolimits_{k=0}^{N-1} \mathcal{P}r\{X_0 = x_k\} \mathcal{P}r\{X_n = x_j | X_0 = x_k\}. \end{aligned} \tag{8}$$

This shows that, given an initial probability distribution and the n-step transition probabilities, the absolute probabilities in the state j after n steps from the start time 0 can be obtained by using this equation. Based on our formal Markov chain definition, this property has been formalized as the following theorem:

Theorem 3 (Absolute Probability).

```
⊢ ∀ X j x N n Linit Ltrans.
   Time_homo_mc X N x Linit Ltrans ∧
   j < N ⇒
   ℙ{s | FST (X n s) = x_j}
   = ∑_{k=0}^{N-1} ℙ{s | FST (X 0 s) = x_k}
                   ℙ({s | FST (X n s) = x_j}|
                     {s | FST (X 0 s) = x_k}).
```

The proof of Theorem 3 is based on the Total Probability Theorem along with some basic arithmetic and probability theoretic reasoning.

5.4 Steady State Probabilities

In many applications, analyzing the stability of Markovian models is of prime importance. For example, we are interested in the probability of states as time tends to infinity under certain conditions, like irreducibility and aperiodicity.

Let $\{X_n, n \geqslant 0\}$ be a Markov chain having state space Ω and one-step transition probability p_{xy} for going from a state with value x to a state with value y. If $\pi(x)$, $x \in \Omega$, are nonnegative numbers summing to one, and if $y \in \Omega$,

$$\pi(y) = \sum_{x \in \Omega} \pi(x) p_{xy}, \tag{9}$$

then π is called a stationary distribution. The corresponding HOL definition is as follows.

Definition 3 (Stationary Distribution).

$$\vdash \forall\ \texttt{p X x N n}.\ \texttt{stationary\_dist p X x N n} = \forall\ \texttt{i}.\ \ 0 \leqslant \texttt{p}\ x_i \wedge \sum_{k=0}^{N-1} (\texttt{p}\ x_k) = 1 \wedge (\texttt{p}\ x_i = \sum_{k=0}^{N-1} \texttt{p}\ x_k \mathbb{P}(\{\texttt{s|FST (X (n + 1) s)=}x_i\} | \{\texttt{s|FST (X n s) =}\ x_k\})).$$

In this definition, x_k and x_i represent the variables x and y of (9), respectively.

As a Markov chain with finite state space, the *steady state probabilities* are defined to be a vector $\boldsymbol{V}_j = \lim_{n\to\infty} \mathbb{P}(n)$. For a DTMC with one-step transition probability p_{ij}, if $\boldsymbol{V}_j$ exists for all $j \in \Omega$, then $\boldsymbol{V}_j$ is known as the stationary probability vector of that Markov chain. In other words, $\boldsymbol{V}_j$ is a stationary distribution of a Markov chain if, $\forall j \in \Omega$,

- $0 \leqslant \lim\limits_{n\to\infty} p_j^{(n)}$,
- $\sum\limits_{i=0}^{N-1} \lim\limits_{n\to\infty} p_i^{(n)} = 1$,
- $\lim\limits_{n\to\infty} p_j^{(n)} = \sum\limits_{i=0}^{N-1} \lim\limits_{n\to\infty} p_i^{(n)} p_{ij}$.

The steady state probability is formalized in HOL as follows.

Theorem 4 (Steady State Probability).

$$\vdash \forall\ \texttt{X n x N Linit Ltrans}.\ \texttt{Time\_homo\_mc X N x Linit Ltrans} \wedge (\forall\ \texttt{x j}.\ \exists \texttt{u}.\ \mathbb{P}\{\texttt{s | FST (X n s) =}\ x_j\} \rightarrow \texttt{u}) \Rightarrow (\texttt{stationary\_dist}\ (\lambda \texttt{x k}.\ \lim_{n\to\infty} \mathbb{P}\{\texttt{s | FST (X n s) =}\ x_k\})\ \texttt{X x N n}).$$

The proof of Theorem 4 starts from rewriting the goal using Definition 3 and then splitting it into three subgoals. Utilizing the Probability Bounds Theorem[27], we can prove the first subgoal $0 \leqslant \lim\limits_{n\to\infty} p_j^{(n)}$. The proof of the second subgoal is primarily based on the following lemma, which can be proved using the Total Probability Theorem, given in (1).

Lemma 2.

$$\vdash \forall\ \texttt{X x N i n Linit Ltrans}.\ \texttt{Time\_homo\_mc X N x Linit Ltrans} \wedge \texttt{i < N} \wedge (\texttt{0 <}\ \mathbb{P}\{\texttt{s|FST (X 0 s)=}x_i\}) \Rightarrow \sum_{j=0}^{N-1} \mathbb{P}(\{\texttt{s | FST (X n s) =}\ x_j\} | \{\texttt{s | FST (X 0 s) =}\ x_i\} = 1.$$

Then, the last subgoal can be proved by applying the linearity of limit of a sequence and the linearity of real summation.

5.5 Generalized Stationary Distribution

If a discrete-time Markov chain with state space Ω and one-step transition probability p_{xy} has a probability distribution π that satisfies the detailed balance equations, given below,

$$\forall x, y \in \Omega, \pi(x) p_{xy} = \pi(y) p_{xy}, \tag{10}$$

then this distribution π is stationary for p_{xy}. This theorem is called a *generalized stationary theorem* and can be mathematically described as Theorem 5.

The detailed balance equations can be formalized in higher-order logic as the following definition, where x_i and x_j represent variables x and y of (10), respectively.

Definition 4 (Detailed Balance Equation).

$$\vdash \forall\ \texttt{p X N}.\ \texttt{db\_equations p X N} = \forall\ \texttt{x i j n}.\ \ \texttt{i < N} \wedge \texttt{j < N} \wedge (\texttt{p}\ x_i)\mathbb{P}(\{\texttt{s|FST (X (n + 1) s) =}\ x_j\} | \{\texttt{s|FST (X n s) =}\ x_i\}) = (\texttt{p}\ x_j)\mathbb{P}(\{\texttt{s|FST (X (n + 1) s) =}\ x_i\} | \{\texttt{s|FST (X n s) =}\ x_j\}).$$

The first input variable `p` in the above predicate is a function that accepts the state as the parameter and returns the probability given in (10). Based on this definition, the stationary theorem can be defined as follows:

Theorem 5 (Generalized Stationary Distribution).

$$\vdash \forall\ \texttt{X x N n Linit Ltrans}.\ \texttt{Time\_homo\_mc X N x Linit Ltrans} \wedge \texttt{db\_equations}\ (\lambda \texttt{x i}.\mathbb{P}\{\texttt{s|FST (X n s)=}x_i\})\ \texttt{X N} \Rightarrow \texttt{stationary\_dist}\ (\lambda \texttt{x k}.\mathbb{P}\{\texttt{s|FST (X n s)=}x_k\})\ \texttt{X x N n}.$$

Here, $\pi(x)$ is specified as a function $(\lambda \texttt{x i}.\ \ \mathbb{P}\{\texttt{s| FST (X n s) =}\ x_i\})$. Similar to the proof of Theorem 4, the proof of Theorem 5 is based on the Probability Bounds Theorem, Lemma 2, and Definitions 3, 4.

5.6 Stationary Process

Stationary processes are frequently used stochastic processes in analyzing time series, which is characterized by having weak white noise. Mathematically, a stochastic process $\{X_t, t \in T\}$ is said to be stationary in the strict sense if $\forall n \geqslant 1$, $t_1, t_2, \ldots, t_n$, $\tau \in T$, the random variables $X_{t_1}, X_{t_2}, \ldots, X_{t_n}$ have the same joint distributions as $X_{t_1+\tau}, X_{t_2+\tau}, \ldots, X_{t_n+\tau}$. In a discrete-time stochastic process, τ is a natural number. From its mathematical definition, we know that a stationary process is different from the process with stationary distribution. In HOL, we formalize a stationary process as follows.

Definition 5 (Stationary Process).

$$\begin{aligned}
&\vdash \forall\ \texttt{X N x}.\\
&\quad \texttt{stationary_proc X N x}\\
&\quad = \forall\ \texttt{L w t n}.\\
&\qquad (\forall\ \texttt{t k.}\ \ \texttt{measurable}\ \{\texttt{s|FST (X t s)=}x_k\}) \wedge\\
&\qquad (\forall\ \texttt{t.}\ \bigcup_{k=0}^{N}\{\texttt{s|FST (X t s)=}\ x_k\}\texttt{=UNIV}) \wedge\\
&\qquad \texttt{EVERY}\ (\lambda\texttt{a.}\ \ \texttt{a} < \texttt{N})\ \texttt{L} \wedge \texttt{n} < \texttt{LENGTH L} \Rightarrow\\
&\qquad (\mathbb{P}(\bigcap_{k=0}^{n}\{\texttt{s|FST (X (w + k) s) =}\ x_{(EL\ k\ L)}\})\\
&\quad = \mathbb{P}(\bigcap_{k=0}^{n}\{\texttt{s|FST (X (t + k) s) =}\ x_{(EL\ k\ L)}\})).
\end{aligned}$$

In this definition, `X` represents the stochastic process. `N` is the cardinality of the states in the states space. `x` refers to a function, which provides the state value for the given index augment. The list `L` contains all the possible state indices. Variables `w` and `t` represent the start time of two successive event sequences. `n` is the number of the states considered in such a joint probability.

Basically, this definition defines a stochastic process for which the joint probability does not depend on the start time for all the possible sequences. The first condition in Definition 5 ensures that all the events possibly involved in this process are measurable. The second condition identifies the state space. Since the elements of `L` represent state indices, they have to be less than the cardinality of the state space and the length of `L` should be longer than the number of events in such a stochastic process.

Using this definition, we can prove that the PMF of a stationary process is independent of the time.

Theorem 6 (PMF of a Stationary Process).

$$\begin{aligned}
&\vdash \forall\ \texttt{X x i n t N}.\\
&\quad \texttt{stationary_proc X N x} \wedge \texttt{i} < \texttt{N} \Rightarrow\\
&\quad (\mathbb{P}\{\texttt{s}\ |\ \texttt{FST (X n s)} = x_i\}\\
&\quad = \mathbb{P}\{\texttt{s}\ |\ \texttt{FST (X t s)} = x_i\}).
\end{aligned}$$

The proof of this theorem is based on Definition 5 and some arithmetic reasoning.

As mentioned in Section 5, a time-homogenous Markov chain has stationary transition probabilities, but the Markov chain itself does not need to be a stationary process in general[29]. In fact, a time-homogeneous Markov chain is stationary if and only if its initial distribution is stationary. We formally verified these results from two different perspectives: a stationary time-homogenous Markov chain has stationary initial distribution (as Theorem 7); and a time-homogenous Markov chain with stationary initial distribution is always a stationary process (as Theorem 8).

Theorem 7 (Stationary DTMC has Stationary Distribution).

$$\begin{aligned}
&\vdash \forall\ \texttt{X x n N Linit Ltrans}.\\
&\quad \texttt{Time_homo_mc X N x Linit Ltrans} \wedge\\
&\quad \texttt{stationary_proc X N x} \Rightarrow\\
&\quad \texttt{stationary_dist}\\
&\qquad (\lambda\texttt{x i.}\mathbb{P}\{\texttt{s|FST (X n s)=}x_i\})\ \texttt{X x N n}.
\end{aligned}$$

The proof of Theorem 7 is based on the stationary distribution definition along with Theorems 3 and 6. If the variable n in Theorem 7 is assigned a value 0 then the stationary DTMC is said to have a stationary initial distribution. In the next theorem, we verify that if the initial distribution of a DTMC is stationary then the corresponding Markov chain is stationary as well.

Theorem 8 (A DTMC with Stationary Initial Distribution is a Stationary Process).

$$\begin{aligned}
&\vdash \forall\ \texttt{X x N Linit Ltrans}.\\
&\quad \texttt{Time_homo_mc X N x Linit Ltrans} \wedge\\
&\quad \texttt{stationary_dist}\\
&\qquad (\lambda\texttt{i.}\mathbb{P}\{\texttt{s|FST (X 0 s)=}x_i\})\ \texttt{X x N 0} \Rightarrow\\
&\quad \texttt{stationary_proc X N x}.
\end{aligned}$$

We proceed with the verification of this theorem by first rewriting the goal using Definitions 2 and 5 and then performing induction on the variable `n` of the stationary process definition, given in Definition 5. The base case is true obviously and the step case is proved using Theorem 1.

Another interesting consequence of Theorems 6 and 8 is that if the initial distribution of a Markov chain is a stationary distribution then its absolute distributions are independent of `n`. That is, if the initial distribution satisfies (9), then the absolute distribution of this Markov chain should be independent of n:

$$\forall x\ t\ n\ j.\ j \in \Omega \Rightarrow Pr(X_t = x_j) = Pr(X_n = x_j).$$

This theorem is formalized in HOL as

Theorem 9 (Stationary PMF).

$$\begin{aligned}
&\vdash \forall\ \texttt{X x j t n N Linit Ltrans}.\\
&\quad \texttt{Time_homo_mc X N x Linit Ltrans} \wedge\\
&\quad \texttt{j} < \texttt{N} \wedge\\
&\quad \texttt{stationary_dist}\\
&\qquad (\lambda\texttt{x i.}\mathbb{P}\{\texttt{s|FST (X 0 s)=}x_i\})\ \texttt{X x N 0} \Rightarrow\\
&\quad (\mathbb{P}\{\texttt{s}\ |\ \texttt{FST (X t s)} = x_j\}\\
&\quad = \mathbb{P}\{\texttt{s}\ |\ \texttt{FST (X n s)} = x_j\}).
\end{aligned}$$

5.7 Reversibility of Markov Chain

The concept of reversible processes is mainly applied in the area of thermodynamics, while reversible Markov chains are commonly used in MCMC-based approaches. The main idea here is to construct a Markov chain

based on a steady state distribution π, as given in (10). Mathematically, a process is said to be reversible if the joint probability of $(X_0, X_1, \ldots, X_n)$ is the same as the joint probability of $(X_n, X_{n-1}, \ldots, X_0)$. The following theorem is used to verify that a time-homogeneous Markov chain satisfying (10) is reversible.

Theorem 10 (Reversible Markov Chain).

$$\begin{aligned}
&\vdash \forall\ \texttt{X t x n N Linit Ltrans L.}\\
&\quad \texttt{Time_homo_mc X N x Linit Ltrans} \wedge\\
&\quad \texttt{db_equations}\\
&\qquad (\lambda \texttt{x i.}\{\texttt{s|FST (X t s)} = x_i\})\ \texttt{X N}) \wedge\\
&\quad \texttt{(EVERY } (\lambda \texttt{a.}\ \ \texttt{a} < \texttt{N)\ L)} \wedge\\
&\quad \texttt{(LENGTH L = n + 1)} \Rightarrow\\
&\quad (\mathbb{P}(\textstyle\bigcap_{k=0}^{n}\{\texttt{s|FST (X (t + k) s)} = x_{(EL\ k\ L)}\})\\
&\quad = \mathbb{P}(\textstyle\bigcap_{k=0}^{n}\ \{\texttt{s}\ |\ \texttt{FST (X (t + k) s)}\\
&\qquad\qquad = x_{(EL\ k\ (REVERSE\ L))}\})).
\end{aligned}$$

The first seven variables in the above theorem have the same context as the ones used in Definition 2 and the last variable L represents a sequence of state indices, in the state space. The first two conditions are the same as the ones used in Theorem 5. While the last two constraint that all elements in L should be less than the cardinality of the states in the state space because in this theorem, $\mathtt{n}+1$ events are considered and thus the length of the index sequence is $\mathtt{n}+1$. The above theorem can be verified using induction on the variable n. The base case is proved based on Theorem 5, in which the absolute distribution of a time-homogeneous Markov chain, which satisfies detail balance equations has stationary distributions. Hence, its initial distribution is also stationary. In the step case proof, we reach the following subgoal after rewriting with the joint probability relationship, given in Theorem 1.

Lemma 3.

$$\begin{aligned}
&\vdash \forall\ \texttt{X t x n N Linit Ltrans L.}\\
&\quad \texttt{Time_homo_mc X N x Linit Ltrans} \wedge\\
&\quad \texttt{db_equations}\\
&\qquad (\lambda \texttt{x i.}\{\texttt{s|FST (X t s)} = x_i\})\ \texttt{X N} \wedge\\
&\quad \texttt{EVERY } (\lambda \texttt{a.}\ \ \texttt{a} < \texttt{N)\ L} \wedge\\
&\quad \texttt{LENGTH L = n + 1} \Rightarrow\\
&\quad (\prod_{k=0}^{n-1}\mathbb{P}(\{\texttt{s|FST (X (t + k + 1) s)}\\
&\qquad\qquad = x_{(EL\ (k+1)\ L)}\}|\\
&\qquad \{\texttt{s|FST (X (t + k) s)} = x_{(EL\ k\ L)}\})\\
&\qquad \mathbb{P}\{\texttt{s|FST (X t s)} = x_{(EL\ 0\ L)}\}\\
&= \prod_{k=0}^{n-1}\mathbb{P}(\{\texttt{s|FST (X (t + k + 1) s)} = x_{(EL\ k\ L)}\}|\\
&\qquad \{\texttt{s|FST (X (t + k) s)} = x_{(EL\ (k+1)\ L)}\})\\
&\qquad \mathbb{P}\{\texttt{s|FST (X t s)} = x_{(EL\ n\ L)}\}),
\end{aligned}$$

which can be verified based on Theorems 1 and 9 along with arithmetic reasoning.

Mathematically, if a Markov chain is reversible, then it has to have the memoryless property as well.

$$\begin{aligned}
&\mathcal{P}r\{X_t = x_0 | X_{t-1} = x_1, \ldots, X_0 = x_n\}\\
&= \mathcal{P}r\{X_t = x_0 | X_{t-1} = x_1\}. \qquad (11)
\end{aligned}$$

We formally verified this property as the following theorem based on probabilistic and arithmetic reasoning in HOL.

Theorem 11 (Joint Probability of Reversible DTMC).

$$\begin{aligned}
&\vdash \forall\ \texttt{X t x N n Linit Ltrans.}\\
&\quad \texttt{Time_homo_mc X N x Linit Ltrans} \wedge\\
&\quad \texttt{EVERY } (\lambda \texttt{a.}\ \ \texttt{a} < \texttt{N)\ L} \wedge \texttt{n + 2 <= LENGTH L} \wedge\\
&\quad \texttt{db_equations}\\
&\qquad (\lambda \texttt{x i.}\{\texttt{s|FST (X t s)} = x_i\})\ \texttt{X N} \wedge\\
&\quad (\forall \texttt{n t.}\ \ \mathbb{P}(\textstyle\bigcap_{k=0}^{n}\{\texttt{s|FST (X (t + k + 1) s)}\\
&\qquad\qquad = x_{(EL\ (k+1)\ L)}\}) \neq \texttt{0}) \Rightarrow\\
&\qquad \mathbb{P}(\{\texttt{s|FST (X t s)} = x_{(EL\ 0\ L)}\}|\\
&\qquad \textstyle\bigcap_{k=0}^{n}\{\texttt{s|FST (X (t + k) s)} = x_{(EL\ k\ L)}\})\\
&= \mathbb{P}(\{\texttt{s|FST (X t s)} = x_{(EL\ 0\ L)}\}|\\
&\qquad \{\texttt{s|FST (X (t + 1) s)} = x_{(EL\ 1\ L)}\}).
\end{aligned}$$

These formally verified theorems not only ensure the correctness of our formal DTMC definitions, presented in Section 4, but also facilitate reasoning about Markovian models in a theorem prover. For illustration purposes, we utilize this formalization to reason about two applications in the next section. Besides that, these properties can also be used to formalize and reason about more advanced Markov chain theory concepts, such as, classified Markov chains, Markov decision process and semi Markov chains. The proof script is about 4 200 lines for the formal verification of the above mentioned properties.

6 Applications

In this section, we present two applications: a simplified binary communication channel[30] and the AMQM protocol⑨.

6.1 Binary Communication Channel Analysis

A binary communication channel[30] is a channel with binary inputs and outputs. The transmission channel is assumed to be noisy or imperfect, i.e., it is likely that the receiver gets the wrong digit. This

⑨Nokovic B, Sekerinski E. http://bnnsolution.com/TagsPaper.pdf, 2010.
Lyngsoe Company. http://www.lyngsoesystems.com/postal/quality_monitoring.asp, Jan. 2012.

channel can be modeled as a two-state DTMC with the following state transition probabilities.

$$\mathcal{P}\mathtt{r}\ \{X_{n+1}\ \mathtt{=}\ 0\ \mid\ X_n\ \mathtt{=}\ 0\}\ \mathtt{=}\ 1-a;$$
$$\mathcal{P}\mathtt{r}\ \{X_{n+1}\ \mathtt{=}\ 1\ \mid X_n\ \mathtt{=}\ 0\}\ \mathtt{=}\ a;$$
$$\mathcal{P}\mathtt{r}\ \{X_{n+1}\ \mathtt{=}\ 0\ \mid X_n\ \mathtt{=}\ 1\}\ \mathtt{=}\ b;$$
$$\mathcal{P}\mathtt{r}\ \{X_{n+1}\ \mathtt{=}\ 1\ \mid X_n\ \mathtt{=}\ 1\}\ \mathtt{=}\ 1-b.$$

The corresponding state and channel diagrams are given in Fig.1 and Fig.2, respectively.

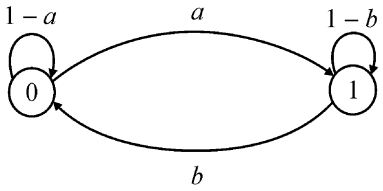


Fig.1. State diagram.

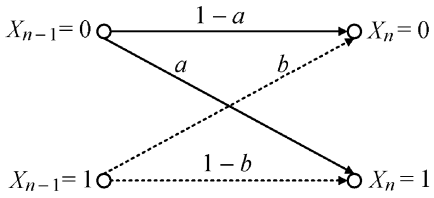


Fig.2. Channel diagram.

The binary communication channel is widely used in telecommunication theory as more complicated channels are modeled by cascading several of them. Here, variables X_{n-1} and X_n denote the digits leaving the systems $(n-1)$-th stage and entering the n-th one, respectively. a and b are the crossover bit error probabilities. Because X_0 is also a random variable, the initial state cannot be determined and thus $\mathcal{P}r(X_0 = 0)$ and $\mathcal{P}r(X_0 = 1)$ cannot be 0 or 1. Although the initial distribution is unknown, the n-step transition probabilities can be verified as the elements of the matrix in (12). Also, the steady-state probabilities can be concluded as that in (13).

$$\boldsymbol{P}^n = \begin{pmatrix} \dfrac{b+a(1-a-b)^n}{a+b} & \dfrac{a-a(1-a-b)^n}{a+b} \\ \dfrac{b-b(1-a-b)^n}{a+b} & \dfrac{a+b(1-a-b)^n}{a+b} \end{pmatrix}, \quad (12)$$

$$\lim_{n\to\infty} \boldsymbol{P}^n = \begin{pmatrix} \dfrac{b}{a+b} & \dfrac{a}{a+b} \\ \dfrac{b}{a+b} & \dfrac{a}{a+b} \end{pmatrix}. \quad (13)$$

Based on the description of the binary communication channel, it has been formalized in HOL as a generic model, using Definition 6.

Definition 6 (Binary Communication Channel Model).

```
⊢ ∀ X x a b p q.
  BCCM X x a b p q
  = (Time_homo_mc X 2 x [p; q] [1-a; a; b; 1-b]) ∧
  (|1 - a - b| < 1) ∧ (0 ⩽ a ⩽ 1) ∧
  (0 ⩽ b ⩽ 1) ∧ (p + q = 1) ∧
  (0 < p < 1) ∧ (0 < q < 1).
```

In this formal model, variable `X` represents the Markov chain. The variable `X` represents a function that provides the state at a given index. The function `x` takes the indices 0 and 1 and returns the value of the state, so that $x_0 = 0$, $x_1 = 1$. Variables `a`, `b`, `p` and `q` are parameters of the functions of initial distribution and transition probabilities.

The first condition ensures that `X` is a time-homogeneous DTMC, with two states in the state space. List `[p; q]` corresponds to `Linit` in Definition 2 and another list `[1 - a; a; b; 1 - b]` gives the one-step transition probability matrix by combining all the rows into a list and corresponds to `Ltrans` in Definition 2. The next three conditions define the allowable intervals for parameters `a` and `b` to restrict the probability terms in `[0, 1]`. It is important to note that, `|1 - a - b| < 1` ensures that both `a` and `b` cannot be equal to 0 and 1 at the same time and thus avoids the zero transition probabilities. The remaining conditions correspond to the one-step transition probabilities.

Next, we use our formal model to reason about the following properties, which correspond to (12) and (13).

Theorem 12 (n-th Step Transition Probabilities).

```
⊢ ∀ X x a b n p q.
  (BCCM X x a b p q) ⇒
  (ℙ({s|FST (X n s)=x₀}|{s|FST (X 0 s)=x₀})
  = (b + a(1 − a − b)^n)/(a + b)) ∧
  (ℙ({s|FST (X n s)=x₁}|{s|FST (X 0 s)=x₀})
  = (a − a(1 − a − b)^n)/(a + b)) ∧
  (ℙ({s|FST (X n s)=x₀}|{s|FST (X 0 s)=x₁})
  = (b − b(1 − a − b)^n)/(a + b)) ∧
  (ℙ({s|FST (X n s)=x₁}|{s|FST (X 0 s)=x₁})
  = (a + b(1 − a − b)^n)/(a + b)).
```

Theorem 13 (Limiting State Probabilities).

$$
\begin{aligned}
&\vdash \forall\ \texttt{X x a b p q}.\\
&\texttt{(BCCM X x a b p q)} \Rightarrow\\
&(\lim_{n\to\infty} \mathbb{P}(\{\texttt{s|FST (X n s)=}x_0\}\,|\\
&\qquad \{\texttt{s|FST (X 0 s)=}x_0\})=\tfrac{b}{a+b}) \wedge\\
&(\lim_{n\to\infty} \mathbb{P}(\{\texttt{s|FST (X n s)=}x_1\}\,|\\
&\qquad \{\texttt{s|FST (X 0 s)=}x_0\})=\tfrac{a}{a+b}) \wedge\\
&(\lim_{n\to\infty} \mathbb{P}(\{\texttt{s|FST (X n s)=}x_0\}\,|\\
&\qquad \{\texttt{s|FST (X 0 s)=}x_1\})=\tfrac{b}{a+b}) \wedge\\
&(\lim_{n\to\infty} \mathbb{P}(\{\texttt{s|FST (X n s)=}x_1\}\,|\\
&\qquad \{\texttt{s|FST (X 0 s)=}x_1\})=\tfrac{a}{a+b}).
\end{aligned}
$$

Theorem 12 has been verified by performing induction on `n` and then applying Lemma 1 and Lemma 2 along with some arithmetic reasoning. Theorem 12 is then used to verify Theorem 13 along with the limit of real sequence principles.

This small two-state DTMC case study clearly illustrates the main strength of the proposed theorem proving based technique against the probabilistic model checking approach by allowing us to verify the desired probabilistic characteristics as generic theorems that are universally quantified for all allowable values of variables `p`, `q`, `a`, `b` and `n`. These variables can also be specialized to specific values to obtain corresponding precise conditional probabilistic values.

6.2 Analysis of Probability of Reaching a State

In this subsection, we will study the probability of reaching a targeted state in an Automatic Mail Quality Measurement (AMQM) system based on the ISO/IEC 18000-7 Standard[31] by building upon our formalized DTMC described in Section 4.

An AMQM system is used to measure the quality of postal service transport and delivery by IPC (International Post Corporation). It measures how fast mail travels from one point to another by using an inplanting process monitoring of the tag serial number and recording the time when a message from the tag is received. This kind of quality measurement of solutions is based on Radio-Frequency Identification (RFID)[31], which is a technology that identifies and tracks objects, such as a product, an animal or a person by using radio waves to transfer data from an electronic tag, called RFID tag. In the last decade, a large volume of research was conducted on complying RFID systems with the international standard ISO/IEC 18000-7. The AMQM system exhibits some features of the ISO/IEC 18000-7 standard and hence its formal analysis is quite important.

In an AMQM system, tags are intended for identifying the objects that are to be managed. The interrogator communicates with the tag in its RF (radio frequency) communication range and controls the protocol, reads information from the tag, directs the tag to store data in some cases, and makes sure that messages are delivered and are also valid. An interrogator controls the messages that are transmitted during their allotted time periods called slots and an acknowledgement received for each message. Based on the AMQM communication protocol, the timing diagram of a tag collection process is depicted in Fig.3.

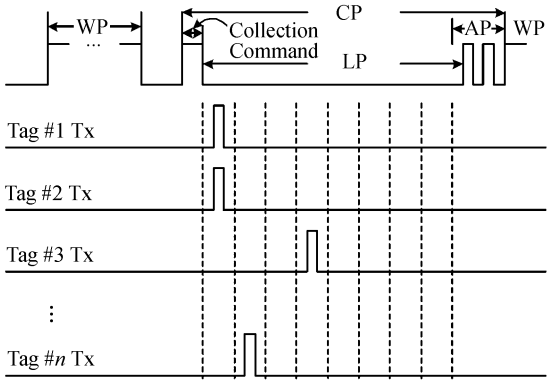


Fig.3. Tag collection process.

The communication sequence starts with a Wakeup Period (WP), within which wake up signals are sent to bring all tags in the ready state. The WP is followed by a collection round named Command Period (CP), which in turn consists of a collection command period, a Listen Period (LP) and an Acknowledge Period (AP). The interrogator then waits for the responses from the tags that are sent randomly. The tag collection is done based on a predetermined algorithm that complies with the ISO/IEC 18000-7 standard. Thus, this system has two properties:

1) The probability that a message can be delivered successfully within i slots is $1-\left(\frac{n-1}{n}\right)^i$.

2) If the collection process is long enough, eventually any message can be delivered successfully.

This communication protocol can be modeled as a DTMC with four states: s_0 (start), s_1 (try), s_2 (lost) and s_3 (delivered)[31], as shown in Fig.4.

In the start state, the message is generated. The next state is always the state try and thus the probability from the start state to try state is 1. The probability of losing a message is α. Thus in the case of losing a message, the system will move to the lost state with probability α. Whereas, it moves to the delivered state with probability $\beta = 1-\alpha$ in case of a successful transmission. Hence, the probability that a message can be

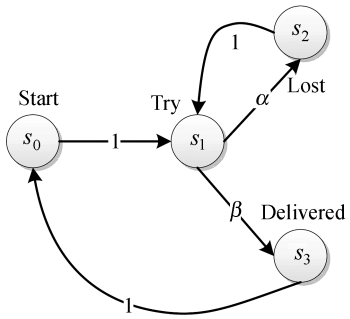


Fig.4. DTMC model of the AMQM protocol.

delivered successfully is β, which equals to $1-\alpha$. Once a message is delivered successfully, the system moves to the start state for getting ready to identify the other tags in next time slot. When the collection process ends, the system falls to sleep mode in order to minimize power consumption. The state transition probability matrix, corresponding to the Markov chain given in Fig.4, is as follows:

$$\boldsymbol{P}=\begin{pmatrix}0 & 1 & 0 & 0\\ 0 & 0 & 1-1/n & 1/n\\ 0 & 1 & 0 & 0\\ 1 & 0 & 0 & 0\end{pmatrix},\quad \boldsymbol{I}=\begin{pmatrix}1\\0\\0\\0\end{pmatrix}. \tag{14}$$

Generally, the possible path of delivering a message successfully can be expressed as:

$$\pi=(\text{start, try, (lost, try)}^k\text{, delivered}).$$

Here, k represents the number of iterations required for a successful message transmission. We use $\mathcal{P}_r(\diamond delivered_i)$ to represent the probability of delivering a message within i trials. Then the probability of reaching state s_3 is given by the following equation where n represents the number of tags.

$$\mathcal{P}_r(\diamond delivered_i)=\sum_{k=0}^{i-1}\alpha^k\beta=1-\Big(\frac{n-1}{n}\Big)^i. \tag{15}$$

As we know, if the collection process is long enough, that is i tends to $+\infty$, then finally the message always can be delivered successfully. So the probability of delivering a message successfully in the future is

$$\begin{aligned}\mathcal{P}_r(\diamond delivered)=&\sum_{k=0}^{\infty}\alpha^k\beta=\frac{\beta}{1-\alpha}\\ =&\frac{\frac{1}{n}}{1-\frac{n-1}{n}}=1.\end{aligned} \tag{16}$$

As mentioned before, the probability of reaching the delivered state depends on the tag collection algorithms, for example, in [31], an improved algorithm is presented for fast tag collection. Thus, (15) and (16) play a vital role in assessing the performance of a tag collection algorithm. In this paper, we formally verify these equations and our results can in turn be used to formally reason about the effectiveness of a tag collection algorithm.

Based on the initial distribution and transition probability matrix, this Markov chain corresponding to the AMQM protocol model can be formalized as:

Definition 7 (AMQM Protocol Model).

```
⊢ ∀ X x n.
    AMQM_MODEL X x n
  = Time_homo_mc X 4 x [1; 0; 0; 0]
      [0; 1; 0; 0;
       0; 0; 1 - 1 / n; 1 / n;
       0; 1; 0; 0;
       1; 0; 0; 0]).
```

Here, `X` represents a stochastic process, and variable `x` represents a function providing the state with a given index and `n` represents the number of tags that are sent randomly. The sole condition in this model constrains `X` to be a time-homogeneous Markov chain with four states. The initial distribution is expressed as a list `[1; 0; 0; 0]` and the transition probability matrix is also shown as a list with row-major order, corresponding to (14).

Now, the two properties presented in (15) and (16) can be verified as:

Theorem 14 (Probability of Reaching Delivered State in AMQM Protocol Model).

```
⊢ ∀ X x n i.
   (AMQM_MODEL X x n) ∧ (n ≠ 0) ⇒
   ∑_{k=0}^{i-1} ℙ({s|FST (X (2 + k * 2) s) = x_3} ∩
          (⋂_{m=0}^{k-1} ({s|FST (X (3 + m * 2) s)=x_1}
           ∩ {s|FST (X (2 + m * 2) s)=x_2}) ∩
             {s|FST (X 1 s) = x_1} ∩
             {s|FST (X 0 s) = x_0})
   = 1 - (((n-1)/n))^i.
```

Theorem 15 (Reachability Probability of AMQM Protocol).

```
⊢ ∀ X x n.
  (AMQM_MODEL X x n) ∧ (n ≠ 0) ⇒
  lim_{i→∞} (∑_{k=0}^{i-1} ℙ({s|FST (X (2+k*2) s)=x_3} ∩
                ⋂_{m=0}^{k-1} ({s|FST (X (3+m*2) s)=x_1} ∩
                    {s|FST (X (2+m*2) s)=x_2}) ∩
                    {s|FST (X 1 s) = x_1} ∩
                    {s|FST (X 0 s) = x_0}) = 1.
```

Theorem 14 corresponds to (15), in which i refers to the number of trials required for successfully delivering `n` tags. The condition $\texttt{n} \neq \texttt{0}$ means that the system will not be waken up if no tag is detected. The performance of a tag collection algorithm can be evaluated by this probability.

Theorem 15 verifies that the probability of reaching the delivered state in infinite trials is 1. That is to say, if the tag collection process is long enough, at last all the tags generated at start state will be received by the reader successfully.

In [32], the PRISM model checker has been used to analyze the AMQM protocol described above. To verify its correctness, the property expressed in Theorem 15 was verified from the point of view of reaching a good state in [32]. The verification of this property is based on solving a group of linear equations instead of verifying a Probabilistic Computation Tree Logic (PCTL) expression mainly because this property involves an infinite summation, which is impossible to express in PCTL. Similarly, the collision probabilities, corresponding to (15), have been verified for some special cases using iterative algorithms. Due to the inherent nature of numerical methods based analysis, these analyses cannot be termed accurate despite consuming enormous computing resources. Moreover, these results are not generic like the ones reported in Theorem 14 of our paper, which means that the complete analysis has to be redone in case the information about number of tags or time slots changes. On the other hand, the proposed theorem proving based approach allows us to formally reason about the generic expressions of two of the most important characteristics of the AMQM protocol, namely, probability of reaching delivered state in AMQM protocol model and reachability probability of AMQM protocol, and the results exactly match the results obtained via paper-and-pencil proof methods.

7 Conclusions

Markov chains, which are stochastic processes with memoryless property, are widely applied to model and analyze a large number of engineering and scientific problems. This paper presents a formalization of time-homogeneous Markov chains with finite state space in a higher-order-logic theorem prover. In particular, we presented a formal definition of DTMC and formally verify some of its classical properties, such as joint probabilities, absolute probabilities and stationary probabilities, using the HOL theorem prover. This work facilitates the formal analysis of Markov chains and provides the foundations for formalizing more advanced concepts of Markov chain theory, like classified Markov chains. Due to the inherent soundness of the proposed approach, it is guaranteed to provide exact answers, which is a very useful feature while analyzing the Markovian models associated with safety or mission-critical systems. In order to illustrate the usefulness of the proposed approach, we analyzed the n-step transition probabilities of a binary communication channel and the probability of reaching some special state in the AMQM protocol. Our results exactly match the corresponding paper-and-pencil based analysis, which ascertains the precise nature of the proposed approach.

The presented work opens the door to a new and very promising research direction, i.e., integrating HOL theorem proving in the domain of analyzing Markov chain based system models. We are currently working on extending the set of formally verified properties regarding DTMCs and extending our work to time-inhomogeneous discrete-time Markov chains, which will enable us to target a wider set of systems. We also plan to build upon the formalization of continuous random variables[19] and statistical properties[19–20] to formalize continuous-time Markov chains to be able to formally reason about statistical characteristics of a wider range of Markovian models.

Acknowledgment We would like to thank Professor Sheng-Zhen Jin from the Chinese Academy of Sciences for the useful discussions on Markov chain theory and his feedback on the reported formalization.

References

[1] Bhattacharya R N, Waymire E C. Stochastic Processes with Applications (1st edition). Wiley-Interscience, 1990.

[2] MacKay D J C. Introduction to Monte Carlo methods. In *Proc. the NATO Advanced Study Institute on Learning in Graphical Models*, Jordan M I (ed.), Kluwer Academic Publishers, 1998, pp.175-204.

[3] Steward W J. Introduction to the Numerical Solution of Markov Chain. Princeton University Press, 1994.

[4] Haas P J. Stochastic Petri Nets: Modelling, Stability, Simulation. Springer, 2002.

[5] Rutten J, Kwaiatkowska M, Normal G, Parker D. Mathematical techniques for analyzing concurrent and probabilisitc systems. In *CRM Monograph Series*, Vol.23, American Mathematical Society, 2004.

[6] Baier C, Katoen J. Principles of Model Checking (Representation and Mind Series). MIT Press, 2008.

[7] Gordon M J C. Mechanizing programming logics in higher-order logic. In *Current Trends in Hardware Verification and Automated Theorem Proving*, Springer, 1989, pp.387-439.

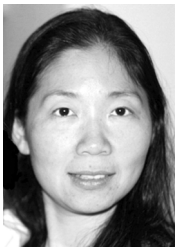
[8] Liu L, Hasan O, Tahar S. Formalization of finite-state discrete-time Markov chains in HOL. In *Proc. the 9th Int. Conf. Automated Technology for Verification and Analysis*, Oct. 2011, pp.90-104.

[9] Knottenbelt W J. Generalised Markovian analysis of timed transition systems [Master's Thesis]. Department of Computer Science, University of Cape Town, South Africa, 1996.

[10] Jonassen H, Tessmer M D, Hannum W H. Task Analysis Methods for Instructional Design. Lawrence Erlbaum, 1999.

[11] Sczittnick M. MACOM — A tool for evaluating communication systems. In *Proc. the 7th Int. Conf. Modelling Tech.*

and Tools for Comput. Performance Evaluation, May 1994, pp.7-10.
[12] Dingle N J, Harrison P G, Knottenbelt W J. HYDRA — Hypergraph-based distributed response-time analyser. In *Proc. Int. Conf. Parallel and Distributed Processing Technique and Applications*, June 2003, pp.215-219.
[13] Ciardo G, Muppala J K, Trivedi K S. SPNP: Stochastic Petri net package. In *Proc. the 3rd Workshop on Petri Nets and Performance Models*, Dec. 1989, pp.142-151.
[14] Sen K, Viswanathan M, Agha G. VESTA: A statistical model-checker and analyzer for probabilistic systems. In *Proc. the 2nd International Conference on the Quantitative Evaluation of Systems*, Sept. 2005, pp.251-252.
[15] Baier C, Haverkort B, Hermanns H *et al.* Model-checking algorithms for continuous-time Markov chains. *IEEE Transactions on Software Engineering*, 2003, 29(4): 524-541.
[16] Nedzusiak A. σ-fields and probability. *Journal of Formalized Mathematics*, 1989, 1, pp.1-6.
[17] Bialas J. The σ-additive measure theory. *Journal of Formalized Mathematics*, 1990, 2, pp.1-7.
[18] Hurd J. Formal verification of probabilistic algorithms [Ph.D. Thesis]. University of Cambridge, UK, 2002.
[19] Hasan O. Formal probabilistic analysis using theorem proving [Ph.D. Thesis]. Concordia University, Canada, 2008.
[20] Hasan O, Abbasi N, Akbarpour B, Tahar S, Akbarpour R. Formal reasoning about expectation properties for continuous random variables. In *Proc. Formal Methods 2009*, Nov. 2009, pp.435-450.
[21] Mhamdi T, Hasan O, Tahar S. Formalization of entropy measures in HOL. In *Proc. the 2nd Interactive Theorem Proving*, Aug. 2011, pp.233-248.
[22] Hölzl J, Heller A. Three chapters of measure theory in Isabelle/HOL. In *Proc. the 2nd Interactive Theorem Proving*, Aug. 2011, pp.135-151.
[23] Paulson L C. Isabelle: A Generic Theorem Prover, Springer, 1994.
[24] Gordon M J C, Melham T F. Introduction to HOL: A Theorem Proving Environment for Higher-Order Logic. Cambridge University Press, 1993.
[25] Milner R. A theory of type polymorphism in programming. *J. Computer and System Sciences*, 1977, 17(3): 348-375.
[26] Harrison J. Theorem Proving with the Real Numbers. Springer, 1998.
[27] Hasan O, Tahar S. Reasoning about conditional probabilities in a higher-order-logic theorem prover. *Journal of Applied Logic*, 2011, 9(1): 23-40.
[28] Norris J R. Markov Cains. Cambridge University Press, 1999.
[29] Prabhu N U. Stochastic Processes: Basic Theory and Its Applications. World Scientific Publisher, 2007.
[30] Trivedi K S. Probability and Statistics with Reliability, Queuing, and Computer Science Applications (2nd edition). John Wiley & Sons, 2001.
[31] ISO/IEC 18000-7 — Information Technology — Radio frequency identification for item management — Part 7: Parameters for active air interface communications at 433 MHz, 2008.
[32] Nokovic B, Sekerinski E. Analysis of interrogator-tag communication protocols. Technical Report, McMaster University, 2010.



Liya Liu received her M.A.Sc degree from Wuhan Technical University of Surveying and Mapping, China, in 1999 and the M.Eng degree from Concordia University, Canada, in 2009. From 1999 to 2005, she has worked as an FPGA design engineer at the China Aerospace Science and Technology Corporation. She has joined the Hardware Verification Group of Concordia University, in 2009, where she is a Ph.D. candidate. Her current research interests include Markov theory, formal methods, higher-order-logic theorem proving and probabilistic analysis of systems. She is a student member of IEEE since 2010.

Osman Hasan received the B.Eng. (Hons.) degree from the N-W.F.P University of Engineering and Technology, Pakistan, in 1997, and the M.Eng. and Ph.D. degrees from Concordia University, Canada, in 2001 and 2008, respectively. He served as an ASIC design engineer from 2001 to 2003 in the industry prior to joining Concordia University in 2004 for his Ph.D. degree. Currently, he is a research associate at the Hardware Verification Group, Concordia University. His current research interests include formal methods, higher-order-logic theorem proving and probabilistic analysis.

Sofiène Tahar received the Diploma degree in computer engineering from the University of Darmstadt, Germany, in 1990, and the Ph.D. degree with distinction in computer science from the University of Karlsruhe, Germany, in 1994. Currently, he is a professor and the research chair in formal verification of system-on-chip at the Department of Electrical and Computer Engineering, Concordia University. His research interests are in the areas of formal hardware verification, system-on-chip verification, analog and mixed signal circuits verification, and probabilistic, statistical and reliability analysis of systems. Dr. Tahar, a professional engineer in the Province of Quebec, is the founder and director of the Hardware Verification Group at Concordia University. He is a senior member of ACM and a senior member of IEEE.