Statistical Verification of Dynamical Systems Using Set Oriented Methods

Yu Wang
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign, USA
yuwang8@illinois.edu

Nima Roohi
Department of Computer Science
University of Illinois at Urbana-Champaign, USA
roohi2@illinois.edu

Matthew West
Department of Mechanical Science and Engineering
University of Illinois at Urbana-Champaign, USA
mwest@illinois.edu

Mahesh Viswanathan
Department of Computer Science
University of Illinois at Urbana-Champaign, USA
vmahesh@illinois.edu

Geir E. Dullerud
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign, USA
dullerud@illinois.edu

ABSTRACT

Modeling, analyzing and verifying real physical systems has long been a challenging task since the state space of the systems is usually infinite and the dynamics of the systems is generally nonlinear and stochastic. In this work, we employ an extension of linear temporal logic (LTL) to describe the behavior of discrete-time nonlinear stochastic systems; this extension is so-called linear inequality LTL (iLTL) which allows for atomic propositions that are linear inequalities on state spaces. To statistically verify iLTL formulas on the systems, we first reformulate discrete-time nonlinear stochastic dynamical systems into Markov processes on their continuous state spaces and then reduce them to discrete-time Markov chains (DTMC) using set-oriented methods. Furthermore, a statistical verification algorithm is proposed to verify iLTL formulas on the reduced systems. The correctness of this statistical verification algorithm is checked both by theoretical analysis and the simulation of a fluid problem. We will show in the successive work that the framework extends to hybrid systems, which is a significant motivation for the approach taken.

1. INTRODUCTION

Temporal logic is an effective tool to describe the behavior of systems whose states change over time. In computer science, various kinds of temporal logic, including linear temporal logic (LTL), have found important applications in formal verification for decades [5]. The time-dependent requirements on hardware or software systems, such as safety and liveness, can be properly expressed by temporal logic formulas [19]. On finite-state systems, such as finite-state automata and discrete time Markov chains (DTMC), these temporal formulas can usually be checked automatically by model checkers.

During the last decades, LTL has been introduced to the study of control systems to specify design objectives. The main challenge there is that analyzing and model checking temporal logic formulas on these infinite-state systems directly is beyond the computing capacity of the model checkers. A possible way to circumvent this problem is to reduce the infinite-state systems to finite-state systems. This is partly implemented by using abstraction-based methods in [13,22, 24], where the authors first find finite-state systems whose trajectories simulate the trajectories of the infinite-state systems, and then perform synthesis and verification on the finite-state systems instead. However, finding such an abstraction is currently only possible for relatively small classes of control systems, such as linear time-invariant systems and piecewise affine systems.

Another method to implement the idea of reducing infinite-state systems to finite-state systems is based on sampling and simulation. In [17,18], the authors construct a dynamic Bayesian network to approximate a continuous dynamical system by first partitioning the state space into finite intervals and then deciding the transition probability between the intervals by drawing samples. Though this method works for a large class of dynamical systems, there is no deterministic guarantee on the error introduced in this procedure.

In practice, discrete-time nonlinear stochastic dynamical systems naturally arise as the time discretization of many physical processes. In this work, we proposed a framework for defining and verifying a kind of temporal logic on these dynamical systems. Specifically, we reformulate the systems as Markov processes on their state spaces and use linear inequality linear inequality (iLTL) to specify their behavior over time [14]. The iLTL extends linear temporal logic by using linear inequalities on the state spaces, which are convenient tools in modeling physical processes, as atomic propositions.

Rather than abstraction-based methods, we use the set ori-

HSCC '15, April 14 - 16, 2015, Seattle, WA, USA
Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM 978-1-4503-3433-4/15/04.
http://dx.doi.org/10.1145/2728606.2728627.

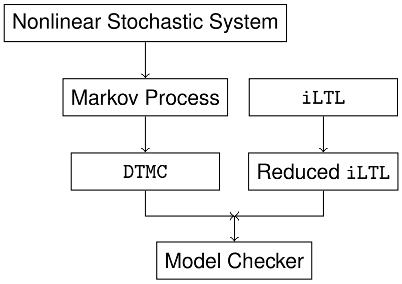

Figure 1: A road map of this work

ented methods [8] stemming from the study of model reduction of dynamical systems to reduce the infinite-state nonlinear stochastic dynamical system into DTMC. The set oriented methods are mathematically Galerkin projections, which are widely used in various areas under different names [15, 16, 20]. But unlike most other model reduction methods based on Galerkin projections, in the set oriented methods, the projections are made over the distributions of the state spaces. In addition, the set oriented methods distinguish from a similar model reduction method, the Mori-Zwanzig method [2,3], in the way that they do not involve the invariant distributions of the dynamical systems in the model reduction procedure.

When it comes to DTMC, there are generally two approaches to model check temporal logic formulas on them: symbolic and statistical. While symbolic methods are generally faster and more precise to handle Markov chain of a small number of states, only statistical methods, based on sampling and simulation, are possible to handle Markov chains of a large number of states [6,12,26–28]. Noting that it usually needs a large number of discrete states to approximate a continuous domain, we adopt a statistical approach to handle the DTMC derived from model reduction and propose a concrete algorithm to model check iLTL formulas on the DTMC.

A road map of this work is shown in Figure 1 and the rest of the article is organized as follows. In Section 2, prerequisites on automata, temporal logic, DTMCand general discrete-time Markov processes are presented. In Section 3, we formulate the problem with a discrete-time nonlinear stochastic system and show that the system can be viewed as a Markov process on the continuous state space. In Section 4, we use the set oriented methods to reduce the Markov process to a discrete-time Markov process on a finite-dimensional state space, which is equivalent to a DTMC and reduce the iLTL formulas on the original Markov processes to iLTL formulas on the DTMC at the same time. In addition, the error bounds of model reduction are given under different conditions. In Section 5, we propose a statistical verification algorithm for the iLTL formulas on the DTMC and show that the result given by the algorithm are of high confidence. In Section 6, a simulation on an advection-diffusion model is given as an application of the theory. Finally, we conclude the main contributions in this work in Section 7.

2. PRELIMINARIES

2.1 Logic and Automata

Linear temporal logic (and some of its variants) are used to specify properties about the system in the paper. We recall the syntax and semantics of LTL, as well as its connection to automata on infinite sequences in this section.

In what follows, we will denote the set of infinite sequences over a set Σ by Σ^ω. For such a sequence $w \in \Sigma^\omega$, w_i will denote the ith element in the sequence w; the first element of the sequence will be w_0. The suffix starting at position i will be denoted by $w_{[i,\infty)}$.

Definition 1. *A Büchi automaton is a tuple* $B = (\mathtt{S}, \Sigma, \Gamma, \mathtt{S}^{\mathsf{init}}, \mathtt{F})$ *where:*

- $\mathtt{S}$ *is a finite non-empty set of* states,
- Σ *is a finite non-empty set of* alphabet,
- $\Gamma \subseteq \mathtt{S} \times \Sigma \times \mathtt{S}$ *is the* transition relation,
- $\mathtt{S}^{\mathsf{init}} \in \mathtt{S}$ *is the* initial *state, and*
- $\mathtt{F} \subseteq \mathtt{S}$ *is the set of* final *states.*

A Büchi automaton is a machine that takes as input an infinite sequence over Σ. Informally, given such an input w, the automata starts in the initial state, reads a symbol in each step from w, and changes its state according to the transition relation Γ. Thus, an *execution* or *run* of B on input w is an infinite sequence of states $\rho \in \mathtt{S}^\omega$ such that $\rho_0 = \mathtt{S}^{\mathsf{init}}$, and for each i, $(\rho_i, w_i, \rho_{i+1}) \in \Gamma$. Such a run is said to be accepting if there is some $s \in \mathtt{F}$ such that for infinitely many i, $\rho_i = s$. The automaton B is said to *accept* input w, if some run ρ of B on w is accepting. The *language* of B, denoted by $\mathtt{Lang}(B)$, is the set of all sequences w that are accepted.

Linear Temporal Logic is a logic built from propositions, using logical connectives ($\neg, \wedge, \vee$) and modal operators ($\mathtt{X}$, $\mathtt{U}$, $\mathtt{R}$). The syntax and semantics of this logic are given below.

Definition 2 (LTL Syntax)**.**

$$\begin{array}{rcccccc} \psi ::= & \top & \mid & \bot & \mid & P & \mid \\ & \neg P & \mid & \psi \wedge \phi & \mid & \psi \vee \phi & \mid \\ & \mathtt{X}\psi & \mid & \psi \,\mathtt{U}\, \phi & \mid & \psi \,\mathtt{R}\, \phi & \end{array}$$

where $P \in \mathtt{AP}$ *is an atomic proposition.*

An LTL formula is evaluated on an infinite sequence of truth values of the atomic propositions in AP. This sequence can be viewed as an infinite sequence over $2^{\mathtt{AP}}$.

Definition 3 (LTL Semantics)**.** *Assume w is an infinite sequence over $2^{\mathtt{AP}}$. Satisfaction relation $\models$ between w and LTL formulas is defined using the following inference rules ($[i] = \{1, 2, \ldots, i\}$):*

- $w \models \top$
- $w \not\models \bot$
- $w \models P$ *iff* $P \in w_0$ *where* $P \in \mathtt{AP}$
- $w \models \neg P$ *iff* $P \notin w_0$
- $w \models \psi \wedge \phi$ *iff* $w \models \psi$ *and* $w \models \phi$
- $w \models \psi \vee \phi$ *iff* $w \models \psi$ *or* $w \models \phi$
- $w \models \mathtt{X}\psi$ *iff* $w_{[1,\infty)} \models \psi$
- $w \models \psi \,\mathtt{U}\, \phi$ *iff* $\exists i \in \mathbb{N} \,\bullet\, w_{[i,\infty)} \models \phi \wedge \forall j \in [i] \,\bullet\, w_{[j,\infty)} \models \psi$
- $w \models \psi \,\mathtt{R}\, \phi$ *iff* $\exists i \in \mathbb{N} \,\bullet\, w_{[i,\infty)} \models \psi \wedge \forall j \in [i+1] \,\bullet\, w_{[j,\infty)} \models \phi$ *or* $\forall i \in \mathbb{N} \,\bullet\, w_{[i,\infty)} \models \phi$

Observe that even though negation ($\neg$) was restricted to only apply to atomic propositions in Definition 2, the logic is closed under negation — $\neg(\psi \vee / \wedge \phi) \equiv (\neg\psi \wedge / \vee \neg\phi)$, $\neg\mathtt{X}\psi \equiv \mathtt{X}\neg\psi$, and $\neg(\psi \,\mathtt{U}\, \phi) \equiv \neg\psi \,\mathtt{R}\, \neg\phi$. As a consequence we will write "$\neg\psi$" to mean the formula in negation normal form obtained by pushing the negation all the way inside ψ using these rules.

LTL and Büchi automata have a close relationship that is often exploited by verification algorithms.

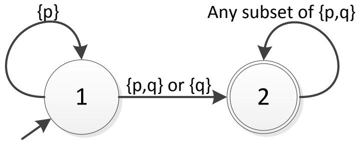

Figure 2: Büchi automaton for $p \, \mathtt{U} \, q$

Theorem 1 (LTL to Büchi automaton [9–11])**.** *For any* LTL *formula ψ one can construct a Büchi automaton B_ψ such that the set of infinite sequences that satisfy ψ is exactly $\mathtt{Lang}(B_\psi)$.*

Example 1. *Consider the Büchi automaton shown in Figure 2.1. It has $\{1,2\}$ as the set of locations, 1 as the initial location, $\{2\}$ as the set of final locations, and $2^{\{p,q\}}$ as the alphabet. The transition from 1 to 1 labeled $\{p\}$, transitions from 1 to 2 labeled $\{p,q\}$ and $\{q\}$, and four transitions from 2 to 2 labeled by subsets of $\{p,q\}$.*

The automaton accepts exactly those sequences that are models of the formula $p \, \mathtt{U} \, q$. This can be understood as follows. Initially we start at state 1, if q is true, we move to location 2, otherwise if p is true, we stay at 1. Once we get to state 2 we stay there forever. Final set contains only state 2. Therefore, in order for a path to satisfy $p \, \mathtt{U} \, q$, q must eventually becomes true and p must always be true before that.

In this paper, the system will be modeled by Markov processes. Thus, specifications will express constraints on the sequence of probability distributions (or distributions for short) that the Markov chain defines. One logic that has been proposed such properties is linear inequality linear temporal logic (iLTL) that was proposed in [14]. Formulas in iLTL are the same as those in LTL except that the propositions are given by linear constraints on the distribution. In other words, iLTL = LTL[AP], where the atomic propositions in AP are constraints of the form $\int_X f \mathrm{d}\mu > b$, where X is compact subspace of $\mathbb{R}^n$, f is an integrable function on X, μ is a measure on X and $b \in \mathbb{R}$. Formally the syntax can be given as

$$\begin{array}{rllll} \psi ::= & \top & \mid \ \bot & \mid \ ineq & \mid \\ & \neg ineq & \mid \ \psi \wedge \phi & \mid \ \psi \vee \phi & \mid \\ & \mathtt{X}\psi & \mid \ \psi \, \mathtt{U} \, \phi & \mid \ \psi \, \mathtt{R} \, \phi & \\ ineq ::= & \int_X f \mathrm{d}\mu > b & & & \end{array}$$

The semantics of iLTL is defined on a sequence of distributions. The semantics of the logical and temporal operators are the same those for LTL. The only novelty is in the definition of when a distribution satisfies an inequality "$\int_X f \mathrm{d}\mu > b$", we defined as expected.

$$\mu \models \text{``}\int_X f \mathrm{d}\mu > b\text{''} \quad iff \quad \int_X f \mathrm{d}\mu > b$$

2.2 Markov Processes and Markov Chains

Let $S = \{s_1, s_2, \ldots, s_n\}$ be a finite state space and $\mathcal{M}(S)$ be the set of probability distributions (or probability mass functions) on S.

Definition 4. *A discrete time Markov chain (*DTMC*) on a finite state space S is a tuple (T_r, p_0) where $p_0 \in \mathcal{M}(S)$ is an initial distribution, and $T_r \in (S \times S) \to [0,1]$ is a* transition matrix *that governs the evolution of probability distributions of the discrete time Markov chain.*

Let $X \subseteq \mathbb{R}^n$ be a compact subset of $\mathbb{R}^n$, $\mathcal{B}(X)$ be the Borel σ-algebra on X, and $\mathcal{M}(X)$ be the set of probability measures on X. Similarly, we define a Markov process on X by the combination of a initial distribution $\mu_0 \in \mathcal{M}(X)$ and a *Markov kernel* T, which determines the evolution of distributions. The Markov kernels play a similar role as the *transition matrices* do in a discrete time Markov chain (DTMC).

Definition 5. *A discrete-time Markov process (or Markov process for short) on a compact set X is a tuple (T, μ_0) where $\mu_0 \in \mathcal{M}(X)$ is an initial distribution and $T : X \times \mathcal{B}(X) \to [0,1]$ is a* Markov kernel *satisfying that*

1. *the map $x \mapsto T(x, A)$ is $\mathcal{B}(X)$-measurable for every $A \in \mathcal{B}(X)$;*
2. *the map $A \mapsto T(x, A)$ is a probability measure on $(X, \mathcal{B}(X))$ for every $x \in X$.*

In the rest of this section, we present the properties of Markov processes and DTMC in parallel. Given a Markov process (T, μ_0) or a DTMC (T_r, p_0), we can derive the distribution at time t inductively by

$$\mu_{t+1}(A) = \int_X T(x, A) \mathrm{d}\mu_t(x), \tag{1}$$

where A is an arbitrary measurable subset of X, or

$$p_{t+1}(j) = \sum_{i=1}^{n} p_t(i) T_r(i, j). \tag{2}$$

From (1) and (2), we know that the Markov kernel T can be viewed as a map $\mathcal{M}(X) \to \mathcal{M}(X)$ and transition matrix T_r can be viewed as a map $\mathcal{M}(S) \to \mathcal{M}(S)$. Therefore, we may also write $\mu_{t+1} = T\mu_t$ and $p_{t+1} = T_r p_t$ in the rest of the article.

For $\mu, \nu \in \mathcal{M}(X)$ or $p, q \in \mathcal{M}(S)$, we call

$$\|\mu - \nu\|_{\mathrm{TV}} = \sup_{A \in \mathcal{B}(X)} |\mu(A) - \nu(A)| \tag{3}$$

or

$$\|p - q\|_{\mathrm{TV}} = \sup_{B \in 2^S} |p(B) - q(B)| \tag{4}$$

the total variation distance, or the distance between μ and ν or p and q. The set $\mathcal{M}(X)$ or $\mathcal{M}(S)$ equipped with distance $\|\cdot\|_{\mathrm{TV}}$ forms a metric space.

A useful property of T is that for any $\mu, \nu \in \mathcal{M}(X)$,

$$\|T\mu - T\nu\|_{\mathrm{TV}} \leq \|\mu - \nu\|_{\mathrm{TV}}. \tag{5}$$

Similarly, for any $p, q \in \mathcal{M}(S)$,

$$\|T_r p - T_r q\|_{\mathrm{TV}} \leq \|p - q\|_{\mathrm{TV}}. \tag{6}$$

Readers may refer to [7] for details.

The concept of *invariant distributions* plays an important role in studying the long-time behavior of a Markov process.

Definition 6. *A distribution $\mu_{inv} \in \mathcal{M}(X)$ or $p_{inv} \in \mathcal{M}(S)$ is called* invariant *if*

$$\mu_{inv}(A) = \int_X T(x, A) \mathrm{d}\mu_{inv}(x) \tag{7}$$

for any $A \in \mathcal{M}(X)$, or

$$p_{inv}(j) = \sum_{i=1}^{n} p_{inv}(i) T_r(i, j) \tag{8}$$

for any $j = 1, 2, \ldots, n$.

An invariant distribution μ_{inv} of a Markov kernel T or an invariant distribution p_{inv} of a transition matrix T_r is mathematically a fixed point. An invariant distribution μ_{inv} or p_{inv} exists and is unique if the Markov kernel T or the transition matrix T_r is *strictly contractive*.

Definition 7. *A Markov kernel T is called strictly contractive by factor $\alpha \in (0,1)$ if for any $\mu, \nu \in \mathcal{M}(X)$,*

$$\|T\mu - T\nu\|_{TV} \leq \alpha \|\mu - \nu\|_{TV}, \tag{9}$$

and a transition matrix T_r is called strictly contractive by factor $\alpha \in (0,1)$ if for any $p, q \in \mathcal{M}(X)$,

$$\|T_r p - T_r q\|_{TV} \leq \alpha \|p - q\|_{TV}, \tag{10}$$

For example, generally, diffusive processes on compact state spaces are strictly contractive.

3. PROBLEM FORMULATION

In practice, the time discretization of most physical processes can be represented by a discrete-time nonlinear stochastic dynamical system

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t), \quad t = 0, 1, 2, \ldots \tag{11}$$

where the *system state* $\mathbf{x}_t$ is a random variable on the *state space* $U \subseteq \mathbb{R}^n$ and f is a nonlinear stochastic function between random variables on U. In particular, for any real number $x_0 \in U$, $f(x_0)$ is a random variable on U. Though the system (11) is time-invariant, a general time-varying system can be reformulated into this form by incorporating time t into the system state $\mathbf{x}_t$.

In this work, we focus on dynamical systems on *compact* state spaces. If the system (11) is Lyapunov stable (in the stochastic sense), then there exists a compact *invariant set* $X \subseteq U$ such that for any random variable $\mathbf{x}$ on X, $f(\mathbf{x}) \in X$ almost surely. In this case, the restriction of the system (11) on X by $f|_X : X \to X$ is a dynamical system on the *compact* state space X.

Now we show that a discrete-time nonlinear stochastic system can be reformulated into a Markov process.

Lemma 2. *Let $\{\mathbf{x}_t \mid t = 0, 1, 2, \ldots\}$ be a sequence of random variables generated by* (11)*, then the distributions $\{\mu_t \mid t = 0, 1, 2, \ldots\}$ of $\{\mathbf{x}_t \mid t = 0, 1, 2, \ldots\}$ is generated by the Markov process (T, μ_0) where*

$$T(x, A) = \mathbb{E}\left[\delta_{f(x)}(A)\right] \tag{12}$$

In particular, if the system (11) is a deterministic nonlinear system, then $T(x, A) = \delta_{f(x)}(A)$. In this case, we have

$$\mu_{t+1}(A) = \mu_t(f^{-1}(A)), \tag{13}$$

for every $A \in \mathcal{B}(X)$. Thus (7) reduces to

$$\mu_{\text{inv}}(A) = \mu_{\text{inv}}(f^{-1}(A)). \tag{14}$$

Finally, we recall from Section 2 that `iLTL` can be used to describe the behavior of the system over time.

4. MODEL REDUCTION

For a Markov process (T, μ_0) and a `iLTL` formula ϕ generated by Definition 2, it is generally impossible to check $(T, \mu_0) \models \phi$ directly because the Markov process contains infinite states. In this section, we first show that the infinite-state Markov process can be reduced to a finite-state Markov chain by set oriented methods and then study the connection between these two models.

$$\begin{tikzcd}
\mathcal{M}(X) \arrow[r, "T"'] & \mathcal{M}(X) \arrow[d, "P"] \\
\mathcal{N}(X) \arrow[u, "Q"] \arrow[r, "T_r"'] & \mathcal{N}(X)
\end{tikzcd}$$

Figure 3: Diagram for single-step Galerkin projection

Definition 8. *$S = \{s_1, s_2, \ldots, s_n\}$ is called a measurable partition of a state space X if*

1. *s_i is Borel measurable for $i = 1, 2, \ldots$,*
2. *$\bigcup_{i=1}^{n} s_i = X$,*
3. *$s_i \cap s_j = \phi$ for any $i \neq j$.*

$s_1, \ldots, s_n$ are not necessarily simply connected, but they are often chosen to be simply connected in practice. Let μ_{Borel} be the Borel measure on $\mathbb{R}^n$. For $A \in \mathcal{B}(X)$, the *characteristic distribution* of A is defined by

$$\nu_A(B) = \frac{\mu_{\text{Borel}}(A \cap B)}{\mu_{\text{Borel}}(A)} \tag{15}$$

where $B \in \mathcal{B}(X)$. Clearly, the characteristic distribution ν_A is a probability distribution on X. Let

$$\mathcal{N}(X) = \{\mu \in \mathcal{M}(X) \mid \mu = \sum_{i=1}^{n} p(i)\nu_{s_i}, p(i) > 0, i = 1, 2, \ldots\} \tag{16}$$

be the set of probability distributions generated by characteristic measures $\nu_{s_1}, \ldots, \nu_{s_n}$. Clearly, there is a bijection from $\mathcal{N}(X)$ to $\mathcal{M}(S)$.

4.1 Galerkin Projection

Since $\mathcal{N}(X) \subseteq \mathcal{M}(X)$, we can define a projection $P : \mathcal{M}(X) \to \mathcal{N}(X)$ by

$$P\mu = \sum_{i=1}^{n} p(i)\nu_{s_i} = \sum_{i=1}^{n} \left(\int_{s_i} \mathrm{d}\mu(x) \right) \nu_{s_i}. \tag{17}$$

and an injection $Q : \mathcal{N}(X) \to \mathcal{M}(X)$ by $Qp = p$. The projection P and the injection Q are well-defined, because $\{s_1, s_2, \ldots, s_n\}$ is a measurable partition of X. In addition, we have $PQ = I$ and $QP \neq I$, where I is the identity map.

As shown in Figure 3, the projection P and injection Q induce a projection from the Markov kernel $T : \mathcal{M}(X) \to \mathcal{M}(X)$ to a Markov kernel $T_r : \mathcal{N}(X) \to \mathcal{N}(X)$ by

$$T_r = PTQ = PT|_{\mathcal{N}(X)}. \tag{18}$$

The above projection procedure, commonly referred to as *Galerkin projection*, is summarized by the following theorem.

Theorem 3. *For a Markov process (T, μ_0) on X, let $S = \{s_1, \ldots, s_n\}$ be a measurable partition of X and P be the projection operator associated with S, then the projection P reduces the Markov process (T, μ_0) to a Markov process (T_r, p_0) by*

$$p_0 = \sum_{i=1}^{n} p_0(i)\nu_{s_i} \tag{19}$$

$$p_{t+1} = \sum_{i=1}^{n} \sum_{j=1}^{n} p_t(i) T_r(i, j) \nu_{s_j} \tag{20}$$

$$\begin{tikzcd}
\mathcal{M}(X) \arrow[r, "T"'] \arrow[d, "P"'] & \mathcal{M}(X) \arrow[r, dotted] & \mathcal{M}(X) \arrow[r, "T"'] & \mathcal{M}(X) \arrow[d, "P"] \\
\mathcal{N}(X) \arrow[r, dashed, "T_r"'] & \mathcal{N}(X) \arrow[r, dotted] & \mathcal{N}(X) \arrow[r, dashed, "T_r"'] & \mathcal{N}(X)
\end{tikzcd}$$

Figure 4: Diagram for multiple-step Galerkin projection

where

$$p_0(i) = \int_{s_i} \mathrm{d}\mu_0(x) \tag{21}$$

$$T_r(i,j) = \int_{s_j} \mathrm{d}\left(\int_X T(x,\cdot)\mathrm{d}\nu_{s_i}\right) \tag{22}$$

Proof. The projection of initial state (21) derives from (17) directly. To prove (22), let $p_t = \sum_{i=1}^n p_t(i)\nu_{s_i} \in \mathcal{N}(X)$ and $p_{t+1} = \sum_{i=1}^n p_{t+1}(i)\nu_{s_i} \in \mathcal{N}(X)$. Combining (1), (17) and (18), we have

$$\begin{aligned} p_{t+1}(j) &= \int_{s_j} \mathrm{d}\mu_{t+1} \\ &= \int_{s_j} \mathrm{d}\left(\int_X T(x,A)\mathrm{d}\left(\sum_{i=1}^n p_t(i)\nu_{s_i}\right)\right) \\ &= \sum_{i=1}^n p_t(i)\left(\int_{s_j} \mathrm{d}\left(\int_X T(x,A)\mathrm{d}\nu_{s_i}\right)\right) \end{aligned} \tag{23}$$

Therefore, $T_r(i,j) = \int_{s_j} \mathrm{d}\left(\int_X T(x,A)\mathrm{d}\nu_{s_i}\right)$. □

The new Markov process (T_r, p_0) derived by model reduction of (T, μ_0) is equivalent with a DTMC. Taking $\{s_1, s_2, \ldots, s_n\}$ as n states, the probability distribution $p = \sum_{i=1}^n p(i)\nu_{s_i}$ on X is identified with a probability distribution $p = (p(1), p(2), \ldots, p(n))$ on S and the evolution operator T_r is identified with a transition matrix $(T_r(i,j))$. Therefore, we may refer to (T_r, p_0) as a DTMC and T_r as a transition matrix later.

For multiple steps, the diagram for Galerkin projection is shown by the non-commutative diagram in 4. Given an initial distribution μ_0, it may either evolve by the Markov kernel T via $\mu_t = T^{(t)}\mu_0$ first and then project to $\mathcal{N}(X)$, or project to $\mathcal{N}(X)$ first and then evolve by the transition matrix T_r via $p_t = T_r^{(t)}p_0$. The different results derived from these two paths determine the error of model reduction (see Section 4.3).

4.2 Reduced iLTL

The projection P also induces an reduction of the iLTL formulas associated with the Markov process (T, μ_0). For $\mu \in \mathcal{M}(X)$, by (17), we have

$$\int_X f\mathrm{d}(P\mu) = \sum_{i=1}^n \left(\int_X f\mathrm{d}\nu_{s_i}\right)\left(\int_{s_i} \mathrm{d}\mu(x)\right) = \sum_{i=1}^n r_i p(i). \tag{24}$$

where

$$r_i = \int_X f\mathrm{d}\nu_{s_i}. \tag{25}$$

Therefore, we can reduce iLTL formulas associated with Markov process (T, μ_0) to iLTL formulas associated with Markov process (T_r, p_0) by replacing the integration $\int_X f\mathrm{d}(P\mu)$ with the summation $\sum_{i=1}^n r_i p(i)$. We call the new temporal logic *reduced* iLTL. This is exactly the form of iLTL proposed in [14].

The relationship between a iLTL formula and its reduction are presented by the following theorem.

Theorem 4. *For $\mu \in \mathcal{M}(X)$ and $p = P\mu = \sum_{i=1}^n p(i)\nu_{s_i} \in \mathcal{N}(X)$, we have*

$$\int_X f\mathrm{d}\mu > b + \delta_P(\mu)esssup_{x\in X}|f(x)| \Longrightarrow \sum_{i=1}^n r_i p(i) > b,$$

$$\sum_{i=1}^n r_i p(i) > b + \delta_P(\mu)esssup_{x\in X}|f(x)| \Longrightarrow \int_X f\mathrm{d}\mu > b,$$

$$\int_X f\mathrm{d}\mu < b - \delta_P(\mu)esssup_{x\in X}|f(x)| \Longrightarrow \sum_{i=1}^n r_i p(i) < b,$$

$$\sum_{i=1}^n r_i p(i) < b - \delta_P(\mu)esssup_{x\in X}|f(x)| \Longrightarrow \int_X f\mathrm{d}\mu < b,$$

where "esssup" stands for essential supremum and

$$\delta_P(\mu) = \|\mu - P\mu\|_{TV} \tag{26}$$

is the error of projection operator P of distribution μ.

The above theorem can be proved by the fact that for any integrable function $f(x)$ on X,

$$|\int_X f\mathrm{d}\mu - \int_X f\mathrm{d}(P\mu)| \le \delta_P(\mu)\mathrm{esssup}_{x\in X}|f(x)|. \tag{27}$$

A iLTL formula ψ is called *compatible* with the partition $S = \{s_1, s_2, \ldots, s_n\}$ if

$$\int_X f\mathrm{d}\mu - \int_X f\mathrm{d}(P\mu) = 0. \tag{28}$$

for any atomic proposition in ψ. It is easy to check that (28) holds if and only if

$$f(x) = \sum_{i=1}^n k_i \mathbf{1}_{s_i} \tag{29}$$

where $k_i \in \mathbb{R}$ and $\mathbf{1}_{s_i}$ is the characteristic function of s_i.

4.3 Error Estimation

Now we study the connection between the original Markov process (T, μ_0) and the reduced Markov chain (T_r, p_0). First, we note that the projection operator P is contractive.

Lemma 5. *Let $S = \{s_1, \ldots, s_n\}$ be a measurable partition of X and P be the projection operator associated with S. For any $\mu, \nu \in \mathcal{M}(X)$,*

$$\|P\mu - P\nu\|_{TV} \le \|\mu - \nu\|_{TV}. \tag{30}$$

Proof. For $i = 1, \ldots, n$, let

$$B_i = \begin{cases} s_i, & \int_{s_i} \mathrm{d}(\mu - \nu) \ge 0 \\ \phi, & \text{otherwise} \end{cases} \tag{31}$$

Then, it is easy to check that for any $A \in \mathcal{B}(X)$,

$$\nu_{s_i}(A)\int_{s_i} \mathrm{d}(\mu - \nu) \le \int_{B_i} \mathrm{d}(\mu - \nu). \tag{32}$$

Thus,

$$\begin{aligned}(P\mu - P\nu)(A) =& \sum_{i=1}^{n} \left(\int_{s_i} \mathrm{d}(\mu - \nu) \right) \nu_{s_i}(A) \\ \leq& \sum_{i=1}^{n} \left(\int_{B_i} \mathrm{d}(\mu - \nu) \right) \\ =& \int_{\bigcup_{i=1}^{n} B_i} \mathrm{d}(\mu - \nu) \\ \leq& \|\mu - \nu\|_{\mathrm{TV}}\end{aligned} \tag{33}$$

Similarly, we can show that $(P\mu - P\nu)(A) \geq -\|\mu - \nu\|_{\mathrm{TV}}$. Therefore, the lemma holds. □

As shown in the non-commutative diagram in Figure 4, given a initial distribution μ_0, there are two paths from the left-up corner to the right-bottom corner: one corresponds to the original Markov process $T \to \ldots \to T \to P$; the other corresponds to the reduced Markov process $P \to T_r \to \ldots \to T_r$. The difference between the two results derived by evolving the distribution along the two paths are defined to be the error the t-step Galerkin projection

$$\Delta_t = \|PT^{(t)}\mu_0 - T_r^{(t)}P\mu_0\|_{\mathrm{TV}} \tag{34}$$

By (18), we have

$$\Delta_t = \|PT^{(t)}\mu_0 - P(TQP)^{(t)}\mu_0\|_{\mathrm{TV}} \tag{35}$$

The error bound of t-step Galerkin projection is given by the following theorem.

Theorem 6. *Given a Markov process (T, μ_0) and a projection operator P, the t-step ($t \geq 1$) error of Galerkin projection*

$$\Delta_t \leq \sum_{i=0}^{t-1} \delta_P((TQP)^{(i)}\mu_0), \tag{36}$$

where δ_P is given in (26).

Proof. For $t = 1$, by Lemma 5 and (5),

$$\begin{aligned}\Delta_1 =& \|PT\mu_0 - P(TQP)\mu_0\|_{\mathrm{TV}} \leq \|T\mu_0 - TQP\mu_0\|_{\mathrm{TV}} \\ \leq& \|\mu_0 - QP\mu_0\|_{\mathrm{TV}} = \delta_P(\mu_0).\end{aligned} \tag{37}$$

For $t > 1$, noting that for any $\mu, \nu \in \mathcal{M}(X)$,

$$\|(TQP)\mu - (TQP)\nu\|_{\mathrm{TV}} \leq \|QP\mu - QP\nu\|_{\mathrm{TV}} \leq \|\mu - \nu\|_{\mathrm{TV}}, \tag{38}$$

we have

$$\begin{aligned}\Delta_t =& \|PT^{(t)}\mu_0 - P(TQP)^{(t)}\mu_0\|_{\mathrm{TV}} \\ \leq& \|T^{(t)}\mu_0 - (TQP)^{(t)}\mu_0\|_{\mathrm{TV}} \\ \leq& \|T^{(t)}\mu_0 - T^{(t-1)}(TQP)\mu_0\|_{\mathrm{TV}} \\ &+ \|T^{(t-1)}(TQP)\mu_0 - T^{(t-2)}(TQP)^{(2)}\mu_0\|_{\mathrm{TV}} \\ &+ \ldots + \|T(TQP)^{(t-1)}\mu_0 - (TQP)^{(t)}\mu_0\|_{\mathrm{TV}} \\ \leq& \sum_{i=0}^{t-1} \delta_P((TQP)^{(i)}\mu_0)\end{aligned} \tag{39}$$

In sum, $\Delta_t \leq \sum_{i=0}^{t-1} \delta_P((TQP)^{(i)}\mu_0)$ for $t = 1, 2, \ldots$. □

Recalling Definition 7, when T is strictly contractive, we can derive a tighter error bound.

Theorem 7. *Given a Markov process (T, μ_0), a projection operator P and the corresponding injection Q, if the Markov kernel T is strictly contractive by factor $\alpha \in (0, 1)$, then the t-step ($t \geq 1$) error of Galerkin projection*

$$\Delta_t \leq \frac{\delta_P}{1 - \alpha}, \tag{40}$$

where

$$\delta_P = \sup_{i \in \mathbb{N}} \delta_P((TQP)^{(i)}\mu_0). \tag{41}$$

Proof. For $t = 1$, clearly $\Delta_t = \delta_P$.

For $t \geq 2$, by (39), we have

$$\begin{aligned}\Delta_t \leq& \|T^{(t)}\mu_0 - T^{(t-1)}(TQP)\mu_0\|_{\mathrm{TV}} \\ &+ \|T^{(t-1)}(TQP)\mu_0 - T^{(t-2)}(TQP)^{(2)}\mu_0\|_{\mathrm{TV}} \\ &+ \ldots + \|T(TQP)^{(t-1)}\mu_0 - (TQP)^{(t)}\mu_0\|_{\mathrm{TV}} \\ \leq& (1 + \alpha + \ldots + \alpha^t)\delta_P \\ \leq& \frac{\delta_P}{1 - \alpha}.\end{aligned} \tag{42}$$

In sum, the theorem holds. □

By combining Theorem 4 and Theorem 7, we can derive the following theorem on the relationship between linear inequalities on the original Markov process and linear inequalities on the reduced Markov process.

Theorem 8. *Given a measurable partition $S = \{s_1, \ldots, s_n\}$ and the corresponding projection operator P, a Markov process (T, μ_0) and its reduction (T_r, p_0) satisfies the following equations:*

$$\begin{aligned}&\int_X f \mathrm{d}\mu_t > b + \frac{\delta_P esssup_{x \in X}|f(x)|}{1 - \alpha} \implies \sum_{i=1}^{n} r_i p_t(i) > b, \\ &\sum_{i=1}^{n} r_i p_t(i) > b + \frac{\delta_P esssup_{x \in X}|f(x)|}{1 - \alpha} \implies \int_X f \mathrm{d}\mu_t > b, \\ &\int_X f \mathrm{d}\mu_t < b - \frac{\delta_P esssup_{x \in X}|f(x)|}{1 - \alpha} \implies \sum_{i=1}^{n} r_i p_t(i) < b, \\ &\sum_{i=1}^{n} r_i p_t(i) < b - \frac{\delta_P esssup_{x \in X}|f(x)|}{1 - \alpha} \implies \int_X f \mathrm{d}\mu_t < b,\end{aligned}$$

where $p_t(i)$, r_i and δ_p are given by (24), (25) *and* (41) *respectively.*

The following corollary of theorem 8 serves as the foundation of Section 5.

Corollary 9. *For an* iLTL *formula ψ on a Markov process (T, μ_0), let (T_r, p_0) be the reduced Markov process by projection operator p, then*

1. *$(T, \mu_0) \models \psi \Longleftarrow (T_r, p_0) \models \psi'$, where ψ' is derived by replacing atomic propositions $\int_X f \mathrm{d}\mu_t > b$ with $\sum_{i=1}^{n} r_i p_t(i) > b + \frac{\delta_P esssup_{x \in X}|f(x)|}{1-\alpha}$ and $\int_X f \mathrm{d}\mu_t < b$ with $\sum_{i=1}^{n} r_i p_t(i) < b - \frac{\delta_P esssup_{x \in X}|f(x)|}{1-\alpha}$.*
2. *$(T, \mu_0) \not\models \psi \Longleftarrow (T_r, p_0) \not\models \psi'$, where ψ' is derived by replacing atomic propositions $\int_X f \mathrm{d}\mu_t > b$ with $\sum_{i=1}^{n} r_i p_t(i) > b - \frac{\delta_P esssup_{x \in X}|f(x)|}{1-\alpha}$ and $\int_X f \mathrm{d}\mu_t < b$ with $\sum_{i=1}^{n} r_i p_t(i) < b + \frac{\delta_P esssup_{x \in X}|f(x)|}{1-\alpha}$.*

where $p_t(i)$, r_i and δ_p are given by (24), (25) *and* (41) *respectively.*

5. STATISTICAL MODEL CHECKING OF iLTL

As pointed out in Section 2, a formula ψ in iLTL can be thought of as a formula ϕ in LTL over a finite set of propositions $\{\mathsf{P}_i\}_{i=1}^{k}$, where each proposition is a constraint of the form $\int_X f_i \mathrm{d}\mu > b_i$. When reasoning over Markov chain after model reduction, the integral can be replaced by a sum using (24) and (25). For simplicity, we denote

$$\sum_{i=1}^{n} r_i p_t(i) = r \cdot p_t. \qquad (43)$$

In the rest of this paper, each r_i denotes a vector with index i instead of the component of a vector.

In the description below we assume that we are reasoning about a finite-state Markov chain. Given a sequence of distributions $w = p_0 p_1 p_2 \cdots$, $w \models \psi$ iff $u \models \phi$, where $u_t = \{\mathsf{P}_i \mid r_i \cdot p_t > b_i\}$. This suggests the following algorithm check if a Markov chain (T_r, p_0) satisfies an iLTL formula ψ. Let w be the (unique) sequence of distributions generated by (T_r, p_0).

1. Construct the sequence u over $2^{\{\mathsf{P}_i \mid 1 \le i \le k\}}$ of labels corresponding to w
2. (T_r, p_0) satisfies ψ iff w is accepted by the Büchi automaton B_ϕ.

In what follows we outline how the above two steps can be accomplished.

Constructing the labels for distributions.

To construct the set of labels u_t corresponding to the distribution $w_t = p_t$, the simplest algorithm would be compute $p_t = T_r^{(t)} p_0$ and then check the constraints corresponding to P_i on p_t. However, this would be expensive for Markov chains with a large number of states. Instead, we compute these labels statistically. First observe that we can draw samples according to distribution p_t by simulating the Markov chain for t steps. Next, recall that P_i is the constraint $r_i \cdot p > b_i$, where r_i is a vector assigning values to each state s of the Markov chain (T_r, p_0). Let us for simplicity assume that for each state s, $r_i(s) \in \{0, 1\}$. In this case, p_t satisfies P_i if the probability of drawing a state s (according to p_t) such that $r_i(s) = 1$ is strictly greater than b_i. This can be statistically checked by drawing samples from p_t and using either Chernoff bounds, or Sequential Probability Ratio Test [23] (see [25] and [21]). Such a statistical test usually takes as parameters an indifference parameter δ_1, error bounds α_1, γ_1. The output of this test, called $\mathcal{A}_1$, is yes, no, or unknownand $\mathcal{A}_1$ ensures

1. $\mathbb{P}[res = \mathtt{no} \mid p_t \vDash \mathsf{P}_i] \le \alpha_1$
2. $\mathbb{P}[res = \mathtt{yes} \mid p_t \nvDash \mathsf{P}_i] \le \alpha_1$
3. $\mathbb{P}[res = \mathtt{unknown} \mid |r_i \cdot p_t(i) - b_i| > \delta_1] \le \gamma_1$

The parameters $\delta_1, \alpha_1, \gamma_1$ can be made arbitrarily small, though that will increase the number of samples needed. In the general case when $r_i(s)$, for a state s, can be any real number, requires one to estimate the mean of a random variable that is not necessarily Bernoulli. In such a situations, the Sequential Probability Ratio Test cannot be used, but we can use a technique due to Chow and Robbins [4].

Running B_ϕ on the labels.

The sequence of labels u can be constructed statistically, symbol by symbol. However, it is an infinite sequence, and in order to run B_ϕ on u, u needs to *ultimately periodic*, i.e., there are finite sequences u_1 and u_2 such that $u = u_1 u_2^\omega$. We assume that the mapping defined by the reduced model is contracting and hence the sequence w converges to the invariant distribution p_{inv}. Now if we assume that there is a δ_2 such that for every i, $|r_i \cdot p_{\text{inv}} - b_i| > \delta_2$, then eventually, for large enough t, p_t and p_{inv} will satisfy exactly the same propositions and so u is ultimately periodic. We assume that p_{inv} is known (or known with some bounded uncertainty). This assumption is readily verified for large classes of important physical models, such as those with energy balance laws that are dissipative in aggregate. For such classes of model general arguments can be used to derive the prior condition, even in the presence of strong nonlinearities, discontinuous dynamics, or other complexities. Since p_{inv} is known we can check if such a δ_2 exists.

```
$t \leftarrow 0$
while $\mathcal{A}_1\left(p_t, p_{\text{inv}}, \frac{1}{2}\min\{\alpha_1, \gamma_1\}, \frac{\delta_1}{3}\right) = \text{failed}$ do
    $t \leftarrow t + 1$
return $t$
```

Figure 5: Part 1 - Finding Number of Sampling Steps

```
for all $t \in [m]$, $\mathsf{P} \in \mathtt{AP}$ do
    $asg(t, \mathsf{P}) = \mathcal{A}_2^{\delta_2}(p_t, \mathsf{P}, \frac{\alpha_2}{2m|\mathtt{AP}|}, \frac{\gamma_2}{2m|\mathtt{AP}|})$
return $asg$
```

Figure 6: Part 2 - Labeling

Now, in order to execute B_ϕ on input u, we need to know when the labels remain invariant. If $\|p_t - p_{\text{inv}}\|_{\text{TV}} < \delta_2/2$ then we know for all $t' > t$, since $\|p_{t'} - p_{\text{inv}}\|_{\text{TV}} \le \|p_t - p_{\text{inv}}\|_{\text{TV}}$, the $p_{t'}$ has the same label as p_{inv}. Thus, we need a statistical test that checks if $\|p_t - p_{\text{inv}}\|_{\text{TV}} < \delta_2/2$ given that we can draw samples from p_t and p_{inv}. A naive algorithm would be to check if $p_t(s)$ is close to $p_{\text{inv}}(s)$ for each state s using algorithm $\mathcal{A}_1$ outlined above. Another possible algorithm to check closeness is the one outlined in [1]. This algorithm is sublinear and provides the following guarantee.

Theorem 10 (Batu *et al.* [1]). *Given parameters ζ and ϵ, and distributions p and p' over a set of n elements, there is a test which runs in time $O\left(n^{2/3}\epsilon^{-8/3}\log\left(\frac{n}{\zeta}\right)\right)$ such that if $\|p - p'\|_{TV} \le \max\left(\frac{\epsilon^{4/3}}{64\sqrt[3]{n}}, \frac{\epsilon}{8\sqrt{n}}\right)$ then the test accepts with probability at least $1 - \zeta$, and if $\|p - p'\|_{TV} > \frac{\epsilon}{2}$ then the test rejects with probability at least $1 - \zeta$.*

The naive algorithm is not as efficient as the sublinear algorithm on large Markov chains. On the other hand, the naive algorithm is much more efficient when p_{inv} has a small support. In what follows, we will refer to the algorithm to check closeness of distributions, whether it be the naive one or the sublinear one, as $\mathcal{A}_2$.

Using $\mathcal{A}_1$ and $\mathcal{A}_2$ we can outline the overall procedure as follows. Parameters α, γ and δ are parameters to the algorithm.

1. Using $\mathcal{A}_1$, with parameters $\delta/3$ and $\alpha/2$ find m such that p_m is within distance $\delta/2$ of p_{inv} (see Figure 5)
2. For each $t < m$ and each proposition P_i, use $\mathcal{A}_2$ with parameters $\delta_1 = \delta$ and $\alpha_1 = \frac{\alpha}{2m|\mathtt{AP}|}$ and $\gamma_1 = \frac{\gamma}{2m|\mathtt{AP}|}$ (see Figure 6)
3. Run B_ϕ on the sequence constructed in the first two

```
for all $\mathsf{P} \in \mathtt{AP}$ do
    $asg(m, \mathsf{P}) = r_{\mathsf{P}} \cdot p_{\text{inv}}$
        ▷ $asg$ defines a (possibly empty) set of infinite path.
if $\mathtt{Lang}(B_\psi) \cap asg \neq \emptyset \wedge \overline{\mathtt{Lang}}(B_\psi) \cap asg \neq \emptyset$ then
    return unknown
else if $\mathtt{Lang}(B_\psi) \cap asg \neq \emptyset$ then
    return yes
else
    return no
end if
```

Figure 7: Part 3 - Model Checking

steps. If the truth value of some proposition is unknown then we consider both truth values for it. We accept if B_ϕ accepts on all such paths; reject if B_ϕ rejects all such paths; and return unknown otherwise (see Figure 7)

Our algorithm $\mathcal{A}$ outlined above provides the following guarantees

$$\mathbb{P}[\mathcal{A}((T_r, p_0), \psi, \alpha, \gamma) = \mathtt{no} \mid (T_r, p_0) \vDash \psi] \leq \alpha \tag{44a}$$

$$\mathbb{P}[\mathcal{A}((T_r, p_0), \psi, \alpha, \gamma) = \mathtt{yes} \mid (T_r, p_0) \nvDash \psi] \leq \alpha \tag{44b}$$

$$\mathbb{P}\begin{bmatrix} \mathcal{A}((T_r, p_0), \psi, \alpha, \gamma) = \mathtt{unknown} \mid \forall t \in \mathbb{N} \bullet \\ \|p_t - p_{\text{inv}}\|_{\text{TV}} \geq \frac{\delta}{4} \Rightarrow \\ \nexists \mathsf{P}_i \bullet |r_i \cdot p_t - b_i| \leq \delta \end{bmatrix} \leq \gamma \tag{44c}$$

The first two inequalities state that probability of having false positive or negative is at most α. The last inequality states that if in all steps that have distributions far enough from the invariant distribution, the actual probability of no atomic proposition P in ψ is too close to b_{P} then the probability of returning unknown is at most γ.

The error analysis of our algorithm can be carried out as follows. When the algorithm returns no (yes) while the correct answer is yes (no), it means that the algorithm made at least one mistake. The probability of finding wrong m is at most $\frac{\alpha}{2}$. Assuming m is computed correctly, the probability of having a step t and an atomic formula P such that truth value of P at step t is computed incorrectly is at most $\frac{\alpha}{2m|AP|}$ (here unknown is considered a correct answer, because it did not effect the output of the algorithm). Hence, the probability of incorrectly determining at least one truth value is at most $\frac{\alpha}{2}$.

Similarly, if the algorithm returns unknown while for any step that is far enough from the invariant distribution, we know the actual probability of no atomic proposition is too close to the threshold of that proposition, it means either the algorithm found m incorrectly, or it found unknown for at least one step and one atomic proposition incorrectly. But we know that the probability of making each of these mistakes is at most $\frac{\gamma}{2}$. Thus the probability of incorrectly returning unknown is at most γ.

$\mathcal{A}$ takes δ as one of its parameters. The problem with δ is that one may not know in advance the correct value for δ. Large values causes the algorithm to return unknown, and small values make the algorithm slow. In order to solve this problem one can start with a large value for δ and decrease it when the algorithm returns unknown for that δ.

6. SIMULATION

Before conclusion, we apply the above theoretical results to a $2D$ advection-diffusion problem. Though the problem starts as a continuous-time problem, we discretize it to a discrete-time problem at some later point and the error in discretization is neglected.

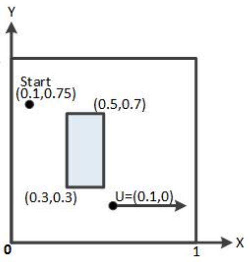

Figure 8: An advection-diffusion problem

As shown in Figure 8, in the 2D plane, the fluid flows uniformly to the right with velocity $u = 0.1$. At time $t = 0$, a particle drops randomly into a 0.03×0.03 region E centered at $(0.1, 0.75)$. Set the intensity of the particle's Brownian motion to be 1 and the time step to be 0.4, then the particle moves by

$$x(n) = x(0) + \frac{2n}{5} + B_1(\frac{2n}{5}) \tag{45}$$

$$y(n) = y(n) + B_2(\frac{2n}{5}) \tag{46}$$

where $B_1(x), B_2(x)$ are mean-zero Gaussians with variance x. The boundary is absorbing, namely, the particle will stop after hitting the boundary.

The initial distribution function $f_0(x, y)$ is the uniform probability distribution function on E and the Markov kernel of the process is given by

$$T((x_1, y_1), (x_2, y_2)) = \frac{5}{4\pi} \exp\left(\frac{5(x_1 - x_2 + \frac{1}{20})^2 + 5(y_1 - y_2)^2}{4}\right). \tag{47}$$

The domain of computation is taken to be $X = [0, 1] \times [0, 1]$. By Definition 7, it is easy to verify that the process contractive by a factor $\alpha = 0.77$.

Let $f_n(x, y)$ be the probability distribution function of the particle at step n. The property we want to check here is

$$\psi = \top \,\mathtt{U} \left(\iint_C f_n(x, y) \mathrm{d}x\mathrm{d}y > 0.1\right), \tag{48}$$

where $C = [30, 50] \times [30, 70]$. The formula means that there exists a time n such that the probability of finding the particle in C is strictly greater than 0.1.

To reduce the system into a DTMC, we partition the system into $n \times n$ boxes and compute the transition probabilities using the integral kernel T. Since T decreases exponentially with position, we assume that there is only transitions between two adjacent boxes. ψ is compatible to the partitions, therefore, ψ is reduced to the iLTL formula

$$\phi = \top \,\mathtt{U} \left(\sum_{s_i \subseteq C} p(i) > 0.1\right) \tag{49}$$

on the DTMC with no additional error.

Table 6 shows our experimental results. Columns States and Transitions are number of states and transition in each example. Column Length is the result of the first part of the algorithm. Columns Time 1 and Time 2 are respectively, the amount of time spent to find the length (first part of the algorithm), and the amount of time spent to on the next two part of the algorithm. Finally, column Total is the total time spent on running each example (it is simply sum of the other times). All times are in seconds. Also times and lengths are the average over 10 runs. In all of our tests, $\alpha = \gamma = \delta = 0.05$ and the results are true.

In the fourth experiment, the size of boxes is 0.0025×0.0025. Using $\alpha = 0.77$ and the initial distribution $f_0(x, y)$, by Theorem 6, we can derive that the error Δ_n is less than 0.03 for all n. Therefore, we know that with confidence $\alpha = \gamma = \delta = 0.05$,

$$\psi = \top \, \mathtt{U} \left(\iint_C f_n(x, y) \mathrm{d}x \mathrm{d}y > 0.07 \right) \tag{50}$$

In order to improve performance, we do not find the smallest possible length. Any length that satisfies the error bound can be used. Therefore, we use some simple heuristic to check when a found length is close enough to the best possible length. Also, in our examples, supports of the invariant distributions always have only one state in it. So we use the naive algorithm to find whether or not a distribution is close enough to the invariant distribution. All tests are run on a laptop with Intel i5 2.50GHz CPU and 6GB of RAM.

Ex.	States	Transitions	Length	Time 1	Time 2	Total
1	10,001	89,181	1,654	0.1	0.01	0.1
2	40,001	358,381	14,248	0.8	0.1	0.9
3	90,001	807,581	48,819	2.6	0.6	3.2
4	160,001	1,436,781	110,191	6.1	3.3	9.4

Table 1: Experimental Results for Verifying ϕ

7. CONCLUSION

In this work, we used `iLTL` to describe the behavior of discrete-time nonlinear stochastic dynamical systems over time and proposed a framework for defining and statistically verifying temporal formulas on the systems using the set oriented methods. Specifically, the systems were first reformulated into Markov processes on a compact state space and then were reduced to `DTMC` using set oriented method. Meanwhile, the `iLTL` formulas on the original Markov processes were also reduced to `iLTL` formulas on `DTMC`. Finally, the reduced `iLTL` formulas were checked by the statistical verification algorithm we proposed. The correctness of this framework is guaranteed by comprehensive analysis on the errors of both model reduction and model checking. We will show in the successive work that the framework extends to hybrid systems.

8. ACKNOWLEDGMENT

The authors acknowledge support for this work from NSF CPS grant 1329991.

9. REFERENCES

[1] T. Batu, L. Fortnow, R. Rubinfeld, W. D. Smith, and P. White. Testing closeness of discrete distributions. *J. ACM*, 60(1):4:1–4:25, Feb. 2013.

[2] C. Beck, S. Lall, T. Liang, and M. West. Model reduction, optimal prediction, and the mori-zwanzig representation of markov chains. In *Proceedings of the 48th IEEE Conference on Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009*, pages 3282–3287, Dec. 2009.

[3] A. J. Chorin, O. H. Hald, and R. Kupferman. Optimal prediction and the mori-zwanzig representation of irreversible processes. *Proceedings of the National Academy of Sciences*, 97(7):2968–2973, Mar. 2000.

[4] Y. S. Chow and H. Robbins. On the asymptotic theory of fixed-width sequential confidence intervals for the mean. *The Annals of Mathematical Statistics*, 36(2):457–462, 04 1965.

[5] E. M. Clarke, E. A. Emerson, and A. P. Sistla. Automatic verification of finite-state concurrent systems using temporal logic specifications. *ACM Trans. Program. Lang. Syst.*, 8(2):244–263, Apr. 1986.

[6] E. M. Clarke and P. Zuliani. Statistical model checking for cyber-physical systems. In T. Bultan and P.-A. Hsiung, editors, *Automated Technology for Verification and Analysis*, number 6996 in Lecture Notes in Computer Science, pages 1–12. Springer Berlin Heidelberg, Jan. 2011.

[7] P. Del Moral, M. Ledoux, and L. Miclo. On contraction properties of markov kernels. *Probability Theory and Related Fields*, 126(3):395–420, 2003.

[8] M. Dellnitz and O. Junge. On the approximation of complicated dynamical behavior. *SIAM Journal on Numerical Analysis*, 36(2):491–515, Jan. 1999.

[9] A. Duret-Lutz. Ltl translation improvements in spot. In *Proceedings of the Fifth International Conference on Verification and Evaluation of Computer and Communication Systems*, VECoS'11, pages 72–83, Swinton, UK, UK, 2011. British Computer Society.

[10] A. Duret-Lutz and D. Poitrenaud. Spot: an extensible model checking library using transition-based generalized büchi automata. In *IN PROC. OF MASCOTS'04*, pages 76–83. IEEE Computer Society, 2004.

[11] P. Gastin and D. Oddoux. Fast ltl to büchi automata translation. In *Proceedings of the 13th International Conference on Computer Aided Verification*, CAV '01, pages 53–65, London, UK, UK, 2001. Springer-Verlag.

[12] D. Henriques, J. Martins, P. Zuliani, A. Platzer, and E. Clarke. Statistical model checking for markov decision processes. In *2012 Ninth International Conference on Quantitative Evaluation of Systems (QEST)*, pages 84–93, Sept. 2012.

[13] M. Kloetzer and C. Belta. A fully automated framework for control of linear systems from temporal logic specifications. *IEEE Transactions on Automatic Control*, 53(1):287–297, Feb. 2008.

[14] Y. Kwon and G. Agha. Linear inequality ltl (iltl): A model checker for discrete time markov chains. In J. Davies, W. Schulte, and M. Barnett, editors, *Formal Methods and Software Engineering*, volume 3308 of *Lecture Notes in Computer Science*, pages 194–208. Springer Berlin Heidelberg, 2004.

[15] S. Lall and C. Beck. Error-bounds for balanced model-reduction of linear time-varying systems. *IEEE Transactions on Automatic Control*, 48(6):946–956, June 2003.

[16] S. Lall, J. E. Marsden, and S. Glavaški. A subspace approach to balanced truncation for model reduction of

nonlinear control systems. *International Journal of Robust and Nonlinear Control*, 12(6):519–535, May 2002.

[17] B. Liu, A. Hagiescu, S. K. Palaniappan, B. Chattopadhyay, Z. Cui, W.-F. Wong, and P. S. Thiagarajan. Approximate probabilistic analysis of biopathway dynamics. *Bioinformatics*, 28(11):1508–1516, June 2012.

[18] B. Liu, D. Hsu, and P. S. Thiagarajan. Probabilistic approximations of ODEs based bio-pathway dynamics. *Theoretical Computer Science*, 412(21):2188–2206, May 2011.

[19] Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems*. Springer-Verlag New York, Inc., New York, NY, USA, 1992.

[20] B. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control*, 26(1):17–32, Feb. 1981.

[21] K. Sen, M. Viswanathan, and G. Agha. On statistical model checking of stochastic systems. In K. Etessami and S. K. Rajamani, editors, *Computer Aided Verification*, number 3576 in Lecture Notes in Computer Science, pages 266–280. Springer Berlin Heidelberg, Jan. 2005.

[22] P. Tabuada and G. Pappas. Linear time logic control of discrete-time linear systems. *IEEE Transactions on Automatic Control*, 51(12):1862–1877, Dec. 2006.

[23] A. Wald. Sequential tests of statistical hypotheses. *The Annals of Mathematical Statistics*, 16(2):pp. 117–186, 1945.

[24] T. Wongpiromsarn, U. Topcu, and R. M. Murray. Receding horizon control for temporal logic specifications. In *Proceedings of the 13th ACM International Conference on Hybrid Systems: Computation and Control*, HSCC '10, pages 101–110, New York, NY, USA, 2010. ACM.

[25] H. L. S. Younes. Error control for probabilistic model checking. In *Verification, Model Checking, and Abstract Interpretation, 7th International Conference, VMCAI 2006, Charleston, SC, USA, January 8-10, 2006, Proceedings*, pages 142–156, 2006.

[26] H. L. S. Younes, E. M. Clarke, and P. Zuliani. Statistical verification of probabilistic properties with unbounded until. In J. Davies, L. Silva, and A. Simao, editors, *Formal Methods: Foundations and Applications*, number 6527 in Lecture Notes in Computer Science, pages 144–160. Springer Berlin Heidelberg, Jan. 2011.

[27] H. L. S. Younes and R. G. Simmons. Statistical probabilistic model checking with a focus on time-bounded properties. *Information and Computation*, 204(9):1368–1409, Sept. 2006.

[28] P. Zuliani, A. Platzer, and E. M. Clarke. Bayesian statistical model checking with application to stateflow/simulink verification. *Formal Methods in System Design*, 43(2):338–367, Oct. 2013.