Reasoning About MDPs as Transformers of Probability Distributions

Vijay Anand Korthikanti, Mahesh Viswanathan, Gul Agha
University of Illinois at Urbana Champaign
{vkortho2, vmahesh, agha}@illinois.edu

YoungMin Kwon
Microsoft Corporation
{ykwon}@cs.uiuc.edu

***Abstract*—We consider Markov Decision Processes (MDPs) as transformers on probability distributions, where with respect to a scheduler that resolves nondeterminism, the MDP can be seen as exhibiting a behavior that is a sequence of probability distributions. Defining propositions using linear inequalities, one can reason about executions over the space of probability distributions. In this framework, one can analyze properties that cannot be expressed in logics like PCTL*, such as expressing bounds on transient rewards and expected values of random variables, and comparing the probability of being in one set of states at a given time with that of being in another set of states. We show that model checking MDPs with this semantics against ω-regular properties is in general undecidable. We then identify special classes of propositions and schedulers with respect to which the model checking problem becomes decidable. We demonstrate the potential usefulness of our results through an example.**

I. Introduction

Markov Decision Processes (MDPs) are the standard model for describing systems with probabilistic and nondeterministic or controlled behavior [1]–[5]. At every state of an MDP, one or more actions are available, and each action is associated with a probability distribution over the successor states. In order to define the semantics of such an MDP, the nondeterministic choices are assumed to be resolved by a scheduler or adversary. Informally, at each step, the scheduler picks an action (based on the execution thus far), and then a next state is picked according to probability distribution associated with the action. Thus, fixing an initial state and a scheduler, the MDP can be viewed as defining a probability measure on the space of trajectories (runs) of the system. By considering logics that reason about the measure of trajectories satisfying modal properties [2], [6], a variety of quantitative properties can be established about such systems [5].

Another way to view the semantics of an MDP is as a transformer of probability measures or probability distributions. In this view, the scheduler's strategy and initial probability measure determine a single infinite sequence of probability distributions, where the ith probability distribution in the sequence represents the probability distribution over the state space at step i. Such an interpretation yields natural models of many real-world systems, where distributions over a set of possible states is the "configuration" of the system. For example, in a sensor network, individual sensor nodes may be asleep, computing, transmitting, sensing or dead. The sensor network may be represented by the probability that each node is in a particular state. In Queuing models, at a given time for a queue, a probability may be assigned for each possible state of the queue. As a third example, consider the compartment model for drug *Absorption, Distribution, Metabolism, and Elimination* (ADME) processes: ADME captures the diffusion of drugs through different organs of the body [7]. Based the compartment model, we can assign a probability that a drug is in a particular organ at a given point in time. In all the above examples, we can represent the system at any point by a *probability distribution* over the *possible states* that the system may be in. It follows that the evolution of these systems can be represented by the evolution of their probability distributions.

There are a number of different properties for such systems that we would we would like to express. One such property is how the probability of being in one set of states compares with the probability of being a different set of states. Another kind of property is based on values associated with each state; in the above examples, these properties are quantities, respectively, the energy consumed by a sensor node, the length of a queue, and the concentration of a drug in an organ. Such quantities are termed *rewards*. In particular, we are interested in expected rewards at a given instant of time, for example, the expected queue length. Note that this reward would be a simple function of the probability at the given instant.

Other properties of interest are temporal in nature. Again, consider the examples we mentioned earlier. In queuing systems, we may interested in guaranteeing that eventually the expected queue length is always less than some constant. In the drug dissemination case, we may be interested in ensuring that the expected drug concentration never exceeds a given maximum and is always above some minimum threshold. We more also be interested in the comparison of temporal properties. For example, we may want to ensure that the expected concentration of the drug in one organ always exceeds the expected concentration in another organ by some threshold.

Such properties about sequences of probability distributions can be expressed by defining propositions using linear inequalities over probability distributions, and using modal operators to reason about temporal behavior. These properties cannot be expressed by typical specification logics as such logics reason about the probability space of trajectories (see Section II on Related Work for details).

The approach we use is taken in [8] for Discrete Time Markov Chains (DTMCs), i.e., for models of probabilistic systems without nondeterminism. In the case of DTMCs, fixing the initial probability distribution determines a single sequence of probability distributions. In this paper, we extend these ideas to reason about MDPs with respect to different sets of schedulers.

We focus on a special class of schedulers called *Markovian* (or *step-dependent*) schedulers [1]. Markovian schedulers are those where the action chosen a step depends on the state and the number of steps taken thus far. Observe that *memoryless* (or *simple*) schedulers (i.e., those where the choice of action only depends on the state) are a special kind of Markovian schedulers. MDPs with such schedulers arise in a number of contexts, including that of Non Homogeneous Discrete Time Markov Chains (NHDTMC) [9].

A Markovian scheduler can be seen as a mapping from a each discrete time instant to a stochastic matrix, where the stochastic matrix captures the nondeterministic choice from each state. Thus, a given Markovian scheduler is nothing but an infinite sequence of stochastic matrices; assuming that each state has finitely many nondeterministic choices, the scheduler is an infinite word over a finite alphabet. Therefore, a set of Markovian schedulers is a language of infinite words, and we consider sets of schedulers that are described using ω-regular languages. The basic computational question we investigate is the following: Given an MDP $\mathcal{M}$, a regular set of schedulers $\mathcal{V}$, propositions over probability distributions defined using linear inequalities, and a regular set $\mathcal{B}$ of infinite words over these propositions that defines "bad" behaviors, is there a scheduler $s \in \mathcal{V}$ such that $\mathcal{M}$'s run with respect to s is a bad execution (i.e., the execution is in $\mathcal{B}$).

We prove a number of results about the decidability of this basic model checking question. We first show that the model checking problem is undecidable, even when the set of schedulers are all Markovian schedulers. This result is proved by reducing the emptiness problem of Probabilistic Finite Automata (PFA) [10]. Next, we show decidability in two special cases. First we show that if we restrict our attention to propositions that only check if the probability measure of some state (or set of states) is non-zero, then the problem is decidable in PSPACE. This is shown by demonstrating that with respect to such propositions, any MDP (with finite states) is bisimilar to a finite state transition system. We also show that this upper bound is tight, by proving that the problem is PSPACE-complete. Next, we show that the model checking problem for general propositions is decidable if we restrict the set of schedulers to be *almost acyclic*; a set of schedulers is almost acyclic if the Büchi automaton recognizing the set is acyclic except for states having a single self loop. We prove this by showing the set of sequences of propositions that an MDP can exhibit with respect to such an almost acyclic set of schedulers is an ω-regular language, and an automaton recognizing such a language can be effectively constructed by using the decidability of the first order theory of reals.

Our theoretical results generalize previous results [8], [11] in a number of directions: we consider the model checking problem with respect to more general properties (ω-regular as opposed to iLTL), more general schedulers (Markovian as opposed to stationary), and more general sets of schedulers (multiple as opposed to single).

Finally, we conclude the paper by presenting an example demonstrating the usefulness of the model we consider and results.

II. Related Work

There has been significant amount of work done on model checking MDP's, where they consider probability space of infinite runs (trajectories) of the MDP with respect to a scheduler [5], [12]. In this line of work, properties are specified in logic like PCTL and PCTL* that reason about the probability measure (on runs) induced by the MDP under some scheduler. Such properties are incomparable to the kinds of properties we consider here. Intuitively, logics like PCTL and PCTL* do not allow one to reason about the probability of being at different states at the *same time*, and on the other hand, the properties we consider do not take into account the branching structure of the system. Precise mathematical proofs of the expressive powers of these logics is given in the section II-A.

Apart from the line of work on logics like PCTL, there have a couple of proposals on exogenous logics that reason about probability distributions [13], [14]. Beauquier et. al. [13] consider a predicate logic of probability that does allow comparing the probability of being in different states at the same time. However, this logic of probability is also incomparable to the properties we consider here. While the logic of probability can reason about the measure of certain execution paths (which we don't consider), the logic can only reason about simple comparison of the probability of different states at a given time instant; detailed comparison can again be found in II-A. Finally, so far only the model checking problem of DTMCs against this predicate logic of probabilities has been considered; the focus of this paper is MDPs. Another logic reasoning about distributions is proposed in [14]. The class of properties considered here is likely to be incomparable to the exogenous logic in [14] primarily based on the difference in model checking complexity. However, precise comparison of the two approachs is left for future investigations.

This paper is a continuation of the line of work initiated by Kwon and Agha, where they propose a logic iLTL to express temporal behaviors of probability distribution executions of Discrete Time Markov Chain(DTMC) [8], [11]. We extend this line of work by considering MDPs, and more general class of properties (ω-regular as opposed to LTL).

A. Comparison with PCTL* and logic of probability of Beauquier et al.

The class of properties we consider in this paper, is incomparable to both PCTL* [5] and the logic of probability, introduced by Beauquier et. al. [13]. We recall that the logic of probability and PCTL* were shown to have different expressiveness in [13]. We begin by considering PCTL*.

Consider the systems D_1 and D_2 shown in 1. Both these systems are Discrete Time Markov Chains (DTMCs). So

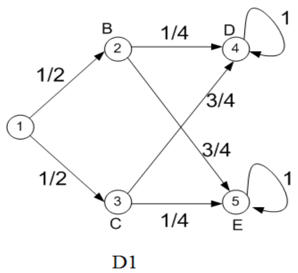

Fig. 1. Discrete Time Markov Chains distinguishable by PCTL* formula.

they do not have any non-determinism and are defined by a single stochastic matrix. Thus, in our semantics, D_1 and D_2 each have a single execution sequence. Moreover, D_1 and D_2 exhibit the same infinite execution sequence of distributions. However, consider the PCTL* property $\varphi_1 = \mathrm{Prob}_{>1/8}(X(BUE))$. Observe that D_1 satisfies φ_1 but D_2 does not. Thus, φ_1 is not expressible in the framework we consider. Conversely, it was shown in [13], that the property $\psi_1 = \exists t.\, \mathrm{Prob}_{\geq 1}(Q(t))$ in logic of probability is not expressible in PCTL*. Property ψ_1 says that there is time step t when the total measure of states labeled Q is at least 1. The property ψ_1 can be expressed in our setting as follows. Consider a labeling function that labels the distributions in the set $\{\mu \mid \sum_{q\in Q} \mu(q) \geq 1\}$ as a. Thus, checking for ψ_1 is the same as checking if eventually we get to a distribution labeled a, i.e., Fa.

The logic of probability is also incomparable with the properties we consider. Once again consider the systems D_1 and D_2 in 1. The property $\varphi_2 = \mathrm{Prob}_{>1/8}(\exists t, t'.\ (t < t') \wedge B(t) \wedge E(t'))$ is satisfied by D_1 but not D_2. Once again since D_1 and D_2 are equivalent in our semantics, we cannot express φ_2. If the we restrict our attention to linear labeling functions where all the coefficients in the linear expressions are constrained to be 1 then all the properties we consider can be expressed in the logic of probability. However, using general linear labeling functions, we can express properties that cannot be expressed using the restricted labeling functions. For example, consider distributions of the set $S = \{q_1, q_2\}$. The labeling function that assigns label b to the set $\{\mu \mid \mu(q_1) + 2\mu(q_2) < 2/3\}$ cannot be expressed by any finite Boolean combination of restricted labels.

III. Motivation

We begin by motivating the use of Markov Decision Processes as transformers of probability distributions. We consider a generic class of models in drug administration, called *compartment models*, and show how these fit the framework we consider in this paper. This will serve as a running example throughout this paper.

In pharmacokinetics, drug disposition changes in our body are often modeled as compartment models [15]. A compartment is a group of tissues that have similar blood flow and drug affinity. Compartment models have the *memoryless property* since drug transition rates leaving a compartment are proportional to the drug concentration levels in the compartment. Hence they can be modeled as Markov chains.

However, one of the complications in the analysis of the pharmacokinetic models is that the drug kinetics often has a multimodal behavior. For example, the drug elimination process shows a saturation behavior when there are more drugs than enzymes; in other words, depending on the drug concentration level, a body be in a normal mode or a saturation mode. The saturated mode has often been modeled using a nonlinear kinetics called *Michaelie-Menten* kinetics. However, this model can be simplified to a linear model when the drug concentration level is large compared to the Michaelie constant [15]. Thus, this multimodal behavior can be effectively captured as an MDP.

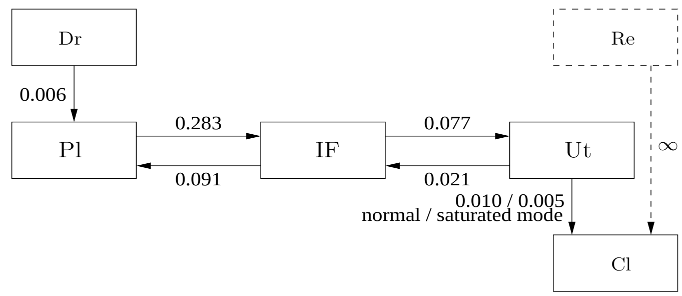

Fig. 2. A three compartment model. The unit of the rate constants is $1/min$.

We now describe the compartment model of a specific drug called Insulin$-^{131}$I [16], [17]. Figure 2 shows a labeled state transition diagram of a three compartment model for Insulin$-^{131}$I. In this diagram, the boxes are the states and the labels at the directed edges are the transition rate constants. The states *Pl*, *IF*, and *Ut* are the three compartments representing, plasma, interstitial fluid, and the site of utilization and degradation. *Dr* and *Cl* are the states for the unabsorbed and the cleared drug respectively. We introduce the *Re* state to use physical units in the specification: if α (mg) of drug is initially taken, then we can put α in the *Dr* state and $1 - \alpha$ in the *Re* state so that the elements of the initial probability distribution adds up to one. Also, by choosing a large unit, one can always make $0 \leq \alpha \leq 1$. Because the rate constant from *Re* to *Cl* is infinite, the remaining drug in this state is instantly moved to the cleared state without interacting with the rest of the system. This infinite rate can be easily handled in the corresponding Markov matrices by setting the transition probability from *Re* state to *Cl* state to 1.

We now build an MDP model for both the normal mode and the saturated mode of drug kinetics for Insulin. Let R_n and R_s be the infinitesimal generator matrices, respectively, for the normal mode and the saturation mode (See Figure 2). We let the sampling time be $T = 10$ (in *min*). Then $N = e^{T\cdot R_n}$ and $S = e^{T\cdot R_s}$, as shown in Figure 3, are the stochastic matrices for the corresponding modes. Observe that as the drug is absorbed in the body, the mode may switch from the normal mode to the saturated mode; conversely, as the drug is

$$N = \begin{bmatrix} 0.94 & 0.02434 & 0.02567 & 0.00798 & 0.00024 & 0 \\ 0.00 & 0.20724 & 0.48298 & 0.29624 & 0.01353 & 0 \\ 0.00 & 0.15531 & 0.42549 & 0.39520 & 0.02400 & 0 \\ 0.00 & 0.02598 & 0.10778 & 0.77854 & 0.08770 & 0 \\ 0.00 & 0.00000 & 0.00000 & 0.00000 & 1.00000 & 0 \\ 0.00 & 0.00000 & 0.00000 & 0.00000 & 1.00000 & 0 \end{bmatrix}$$

$$S = \begin{bmatrix} 0.94 & 0.02435 & 0.02568 & 0.00809 & 0.00012 & 0 \\ 0.00 & 0.20728 & 0.48329 & 0.30257 & 0.00686 & 0 \\ 0.00 & 0.15540 & 0.42612 & 0.40626 & 0.01221 & 0 \\ 0.00 & 0.02653 & 0.11080 & 0.81776 & 0.04491 & 0 \\ 0.00 & 0.00000 & 0.00000 & 0.00000 & 1.00000 & 0 \\ 0.00 & 0.00000 & 0.00000 & 0.00000 & 1.00000 & 0 \end{bmatrix}$$

Fig. 3. Stochastic matrices for the normal mode (N) and the saturated mode (S), when the states are ordered as *Dr*, *Pl*, *IF*, *Ut*, *Cl*, and *Re*.

cleared from the body, the mode will eventually be switched back to the normal mode. If the drug concentration level never reached the saturation level, then the model will remain in the normal mode. If we regard the stochastic matrices as the alphabet $\{N, S\}$, then these mode transitions can be concisely expressed in a regular language as:

$$\mathcal{V} \quad = \quad N^* S^* N^\omega,$$

Later in this paper, we show that matrices and the schedulers defined above satisfies our conditions for decidable model checking.

The compartment model should satisfy the following sort of specification: regardless of the mode changes, a dose should adhere to certain conditions relating to the *Minimum Effective Concentration* (MEC) and the *Minimum Toxic Concentration* (MTC). Specifically, we require that (1) the drug concentration level should never exceed MTC (2.1 $\mu g/ml$), (2) the drug concentration level should remain above MEC (1.4 $\mu g/ml$) for a certain duration for the drug to be effective, and (3) the first drug effect should occur within a certain time limit. If we assume that the body weight to be 60 (Kg) and the volume of *Ut* compartment is 15.8% of the body weight, then the amount of drug for MTC and MEC are $mtc = 0.020$ and $mec = 0.013$ (g) respectively.

Let $L(\mathcal{V})$ be the language of $\mathcal{V}$. A $v \in L(\mathcal{V})$ is one of the possible sequences of the drug kinetic mode changes. Given a v and a probability distribution of an initial drug distribution, the rest of the drug disposition changes are determined. Let μ_t be the probability distribution representing the the amount of drug in the compartments at time *t* and let $\mu_t(c)$ be the amount of the drug in c compartment. Now, suppose that initially α (*mg*) of the drug is taken and $v \in L(\mathcal{V})$ is a valid drug kinetic mode changes. Then, $\mu_{t+1} = \mu_t \cdot v[t]$ for $t \geq 0$, where $v[t]$ is the t^{th} character of v, and $\mu_0(Dr) = \alpha$ and $\mu_0(Re) = 1 - \alpha$.

Borrowing the syntax of iLTL, which is a subset of more general Büchi automata specification of this paper, we specify the above conditions. Before we express the property in our logic, we define propositions *effective* and *nontoxic* over the space of probability distributions. A probability distribution μ is labeled *effective* if $\mu(Ut) > mec$ or labeled *nontoxic* if $\mu(Ut) < mtc$. Using $\Box$ operator (*always*) the first condition can be simply written as $\Box$*nontoxic*. That is, for all $v \in L(\mathcal{V})$, $\mu_t(Ut) < mtc$ for $t \geq 0$.

Regarding the second condition, let the required active duration be at least two sampling periods which can be expressed as (*effective* $\wedge$ X *effective* $\wedge$ XX *effective*). Because this condition may not need to occur immediately, but should eventually occur, using the $\Diamond$ operator (*eventually*), we can write the second condition as $\Diamond$ (*effective* $\wedge$ X *effective* $\wedge$ XX *effective*). In other words, for all $v \in L(\mathcal{V})$, there is a $t \geq 0$ such that $\mu_t(Ut) > mec$, $\mu_{t+1}(Ut) > mec$, and $\mu_{t+2}(Ut) > mec$. Finally, suppose that we want to have the first drug effect within two sampling periods. This condition can be simply written as (*effective* $\vee$ X *effective* $\vee$ XX *effective*).

Combining these three conditions together, the entire specification can be written as:

$$\begin{aligned} \psi \quad = \quad & (\textit{effective} \vee \texttt{X}\,\textit{effective} \vee \texttt{XX}\,\textit{effective}) \wedge \Box\,(\textit{nontoxic}) \\ & \wedge \Diamond\,(\textit{effective} \wedge \texttt{X}\,\textit{effective} \wedge \texttt{XX}\,\textit{effective}). \end{aligned}$$

As ψ is an LTL formula, there is a Büchi automaton $\mathcal{B}_{\neg\psi}$ for the negated specification $\neg\psi$ that comprises the bad behavior in this paper. In this paper, we show that the model checking whether any of the drug concentration level changes of the drug kinetic model is in the bad behavior of $\mathcal{B}_{\neg\psi}$ is decidable.

IV. Preliminaries

A. Notations

Relations A relation L over the sets $X_1, \ldots, X_k$ is a subset of their Cartesian product, written $L \subseteq X_1 \times \cdots \times X_k$

Sequences Let S be a finite set. We let $|S|$ denote the cardinality of S. Let $\eta = s_1, s_2, \ldots$ be a possibly infinite sequence over S. The length of η, denoted as $|\eta|$, is defined to be the number of elements in η, if η is finite, and ω otherwise. S^* denotes the set of finite sequences, S^+ the set of finite sequences of length ≥ 1 and S^ω denotes the set of infinite sequences. If α is a finite sequence, and η is either a finite or an infinite sequence then $\alpha\eta$ denotes the concatenation of the two sequences in that order. For integers i and j such that $1 \leq i \leq j < |\eta|$, $\eta[i, j]$ denotes the (finite) sequence $s_i, \ldots, s_j$ and the element $\eta[i]$ denotes the element s_i. A finite prefix of η is any $\eta[1, j]$ for $j < |\eta|$.

Büchi Automata A Büchi automaton $\mathcal{A}$ on infinite strings over a finite alphabet Σ is a 4-tuple (Q, Δ, q_0, F) where Q is a finite set of states; $\Delta \subseteq Q \times \Sigma \times Q$ is a transition relation; $q_0 \in Q$ is an initial state; $F \subseteq Q$ is a set of accepting/final automaton states. If for every $q \in Q$ and $a \in \Sigma$, there is exactly one q' such that $(q, a, q') \in \Delta$ then $\mathcal{A}$ is called a deterministic automaton. Let $\alpha = a_1, a_2, \ldots$ be an infinite sequence over Σ. A run r of $\mathcal{A}$ on α is an infinite sequence $r_0, r_1, \ldots$ over Q such that $r_0 = q_0$ and for every $i > 0, (r_{i-1}, a_i, r_i) \in \Delta$. The run r is accepting if some state in F appears infinitely often in r. The automaton $\mathcal{A}$ accepts the string α if it has an accepting run over α. The language accepted (recognized) by

$\mathcal{A}$, denoted by $L(\mathcal{A})$, is the set of strings that $\mathcal{A}$ accepts. A language L' is called ω-regular if it is accepted by some finite state Büchi automaton.

B. Stochastic Matrices

A stochastic matrix over a set of states S is a matrix $P : S \times S \to [0,1]$ such that $\forall s \in S. \sum_{s' \in S} P(s,s') = 1$. Let $SM(S)$ denote the set of all stochastic matrices over the set S.

Communicating Classes: Let us fix a stochastic matrix P over a set of states S. We say a state q *leads to* a state q' (denoted by $q \to q'$) if $P^n(q,q') > 0$ for some $n \geq 0$, where P^n denotes the n-fold product of matrix P. State q is said to *communicate with* q' ($q \leftrightarrow q'$) if both $q \to q'$ and $q' \to q$. Observe that $\leftrightarrow$ is an equivalence relation, and its equivalence classes are called *communicating classes*. A communicating class C is said to *closed*(recurrent) if $q \in C$, $q \to q'$ imply $q' \in C$. Thus a closed class is one from which there is no escape. A state i is absorbing if $\{i\}$ is a closed class. A transition matrix P, where S is a single class is called irreducible. Observe that every stochastic matrix on a finite state-space has at least one closed communicating class.

Aperiodicity: In a stochastic matrix P, state q is *aperiodic* if $P^n(q,q) > 0$ for all sufficiently large n. Equivalently, state q is aperiodic if and only if the set $\{n | n \geq 0 \wedge P^n(q,q) > 0\}$ has no common divisor other than 1. A communicating class is aperiodic if all the states in the class are aperiodic. Suppose $S' \subseteq S$ is closed and has an aperiodic state q then S' is an aperiodic communicating class.

Distributions A *probability distribution* over a set S is $\mu : S \to [0,1]$ such that $\sum_{q \in S} \mu(q) = 1$. We will denote $\mathrm{Dist}(S)$ to be the set of all distributions over S. A distribution μ is said to be a *stationary distribution* of matrix P if $\mu \cdot P = \mu$ and is denoted by $\mathcal{S}(P)$; if P has exactly one stationary distribution then it is said to have a unique stationary distribution. A distribution π is said to be a *limiting distribution* of matrix P if there exists a distribution μ such that $\mu \cdot P^n \to \pi$ as $n \to \infty$; if P has exactly one limiting distribution then it is said to have a unique limiting distribution. If matrix P has a unique limiting distribution π then it is also a unique stationary distribution but not vice versa [18]. Fix a set $S = \{1, 2, \ldots n\}$ and a collection of distributions $X \subseteq \mathrm{Dist}(S)$. X is said to be *linear* if there are $a_1, a_2, \ldots a_n, b \in \mathbb{Q}$ such that $\mu \in X$ iff $\sum_{i \in S} a_i \mu(i) \bowtie b$, where $\bowtie \in \{<, \leq, \geq, >\}$. Finally, X is said to be an *indicator* set if there are $a_1, a_2, \ldots a_n \in \mathbb{Q}^+$ such that $\mu \in X$ iff $\sum_{i \in S} a_i \mu(i) > 0$.

Regular Stochastic Matrices A stochastic matrix P is *regular* if it has only one closed class $I \subseteq S$ and I is aperiodic. One result on regular stochastic matrices that we will use is the Ergodic Theorem [19].

Theorem IV.1 (Ergodic Theorem). *A stochastic matrix has an unique limiting distribution iff it is regular.*

C. Metric Spaces

Recall that a metric space (M,d) is a set M equipped with a distance metric $d : M \times M \to \mathbb{R}_{\geq 0}$ with the following properties: for every $x,y,z \in M$, $d(x,y) = 0$ iff $x = y$; $d(x,y) = d(y,x)$; and $d(x,z) \leq d(x,y) + d(y,z)$. An *open ball* of radius $r (> 0)$ about a element x in M is defined as the set $B(x;r) = \{y \in M \mid d(x,y) < r\}$. Similarly, an open ball of radius $r > 0$ about a set $X \subseteq M$ is defined as the set $B(X;r) = \{y \in M \,|\, \exists x \in X.\ d(x,y) < r\}$. A subset U of M is called *open* if for every x in U there exists an $r > 0$ such that $B(x;r)$ is contained in U. The complement of an open set is called *closed*. The *interior* of a set S, $int(S)$, is the set of all x in S such that $B(x;\epsilon)$ is contained in S for some $\epsilon > 0$. A metric space (M,d) is said to be *complete* if every Cauchy sequence has a limit in M. A mapping $f : M \to M$ is a said to *contracting* if there is a k such that $0 \leq k < 1$, and for all $x,y \in M$, $d(T(x),T(y)) < kd(x,y)$.

In this paper, we will consider the metric space on $M = \mathrm{Dist}(S)$ with distance metric defined as

$$d(\mu,\mu') = max_{A \subseteq S} |\mu(A) - \mu'(A)| = \frac{1}{2} \sum_{\omega \in S} |\mu(\omega) - \mu'(\omega)|.$$

We will use the following observation about regular matrices.

Lemma IV.2. *If a stochastic matrix P is regular then there exists k such that the mapping from the set of all probability distributions given by $\mu \mapsto \mu \cdot P^k$ is contracting with respect to the metric d. In other words, there is a t such that $0 \leq t < 1$ and*

$$d(\mu \cdot P^k, \mu' \cdot P^k) < t \cdot d(\mu,\mu').$$

Proof: The proof is algebraic and is given in the appendix. ■

V. MDPs as transformers of probability distributions

Markov decision processes (MDPs) are a natural way for modelling systems with both probabilistic and nondeterministic behavior.

Definition A Markov decision process is a tuple $\mathcal{M} = (S, \mu_0, \mathcal{P})$, where S is a set of states, $\mu_0 \in \mathrm{Dist}(S)$ is an initial probability distribution, $\mathcal{P} \subseteq SM(S)$ is a finite set of stochastic matrices.

A probabilistic transition $s \xrightarrow{P} s'$ is made from a state s by first nondeterministically selecting a matrix $P \in \mathcal{P}$ and then making a probabilistic choice of target state s' according to the distribution $P(s)$, where $P(s)$ denotes the probability distribution defined at state s in the stochastic matrix P.

Remark Our definition of MDP differs from the standard definition of a MDP used else where. A MDP $\mathcal{M}$ is defined as a tuple $(S, \mu_0, Steps)$, where S and μ_0 have the same interpretation as ours, and $Steps : S \to 2^{\mathrm{Dist}(S))}$ is the probability transition function. In this definition, a probabilistic transition $s \xrightarrow{\mu} s'$ is made from a state s by first nondeterministically selecting a distribution $\mu \in Steps(s)$ and then making a probabilistic choice of target state s' according to the distribution μ. It is easy to see that these two presentations of MDPs are equivalent.

Definition An MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$ is termed contracting if every stochastic matrix P in $\mathcal{P}$ is regular.

A path of an MDP represents a particular resolution of both nondeterminism and probability. Formally, a path of an MDP is a non-empty finite or infinite sequence of probabilistic transitions:

$$\pi = s_0 \xrightarrow{P_0} s_1 \xrightarrow{P_1} s_2 \xrightarrow{P_2} \cdots$$

such that $P_i(s_i, s_{i+1}) > 0$ for all $i \geq 0$. We denote by $\pi(i)$ the state s_i.

In contrast to a path, a scheduler (sometimes also known as a adversary or policy) represents a particular resolution of nondeterminism only. For deciding which of the next nondeterministic steps to take, a scheduler may have access to the current state only or to the path from the initial to the current state. More precisely, a deterministic [1] scheduler is a function mapping every finite path π to a stochastic matrix $P \in \mathcal{P}$. There are special classes of schedulers that have been considered. For a MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$, a *Markovian scheduler* (also called *step-dependent* scheduler) $D : \mathbb{N} \to SM(S)$ represents a map from $\mathbb{N}$ to $\mathcal{P}$. Note that a stationary Markovian scheduler (also called *simple* scheduler) which corresponds to a single stochastic matrix in $\mathcal{P}$ is a special kind of Markovian scheduler. In this paper we consider only Markovian schedulers.

A Markovian scheduler can be thought of as an infinite string over the alphabet $\mathcal{P}$. A set of Markovian schedulers (possibly infinite) $\mathcal{V}$ is termed as regular if there is a finite state Büchi automata $\mathcal{A} = (Q_c, \rightarrow, q_0, F)$ over alphabet $\mathcal{P}$ such that $L(\mathcal{A}) = \mathcal{V}$. We will abuse notation and represent both the set of regular Markovian schedulers and the Büchi automaton representing it, by the same letter.

Transition System: Let $\mathcal{T}(\mathcal{M}) = \langle \mathcal{Q}, \rightarrow \rangle$ denote the transition system of MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$ where $\mathcal{Q} = \text{Dist}(S)$ and $\rightarrow \subseteq \text{Dist}(S) \times \mathcal{P} \times \text{Dist}(S)$ is such that $\forall \mu, \mu' \in \mathcal{Q}.\ (\mu, P, \mu') \in \rightarrow$ iff $\mu \cdot P = \mu'$. From now on we use the notation $\mu \xrightarrow{P} \mu'$ instead of $(\mu, P, \mu') \in \rightarrow$.

Given a MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$, a Markovian scheduler D defines an infinite execution sequence of $\mathcal{T}(\mathcal{M})$: $\mu_0\mu_1\mu_2\ldots$ such that $\mu_i \xrightarrow{D(i)} \mu_{i+1}$. Let $\mathcal{R}^{\mathcal{V}}(\mathcal{M}, \mu)$ denote the set of all execution sequences of transition system $\mathcal{T}(\mathcal{M})$ starting at state (probability distribution) μ with respect to the set of Markovian schedulers $\mathcal{V}$. Similarly, $\mathcal{R}(\mathcal{M}, \mu)$ denote the set of all execution sequences of transition system $\mathcal{T}(\mathcal{M})$ starting at state μ. We define $\mathcal{R}^{\mathcal{V}}(\mathcal{M})$ to be $\mathcal{R}^{\mathcal{V}}(\mathcal{M}, \mu_0)$ and $\mathcal{R}(\mathcal{M})$ to be $\mathcal{R}(\mathcal{M}, \mu_0)$.

Labeling: Let $\lambda : \text{Dist}(S) \rightarrow 2^{\Sigma}$ denote a 2^{Σ}-labeling function over the state space of the transition system $\mathcal{T}(\mathcal{M})$, where Σ is a finite set of labels. A 2^{Σ}-labeling function λ is linear iff $\forall a \in \Sigma.\ U_a = \{\mu \mid a \in \lambda(\mu)\}$ is a linear set. Similarly a 2^{Σ}-labeling function λ is termed indicator iff $\forall a \in \Sigma.\ U_a = \{\mu \mid a \in \lambda(\mu)\}$ is an indicator set. We extend the definition of λ to execution sequences of $\mathcal{T}(\mathcal{M})$ such that for every execution sequence $\gamma = \mu_0\mu_1\mu_2\ldots$ of $\mathcal{T}(\mathcal{M})$, $\lambda(\gamma) = \lambda(\mu_0)\lambda(\mu_1)\lambda(\mu_2)\ldots$. Language $L^{\lambda}_{\mathcal{V}}(\mathcal{M}, \mu)$ defined by transition system of MDP $\mathcal{M}$ with respect to a labeling function λ and the set of Markovian schedulers $\mathcal{V}$ is as follows

$$L^{\lambda}_{\mathcal{V}}(\mathcal{M}, \mu) = \{\alpha \in (2^{\Sigma})^{\omega} \mid \exists \beta \in \mathcal{R}^{\mathcal{V}}(\mathcal{M}, \mu).\ \lambda(\beta) = \alpha\}$$

we define $L^{\lambda}_{\mathcal{V}}(\mathcal{M})$ to be $L^{\lambda}_{\mathcal{V}}(\mathcal{M}, \mu_0)$.

Model Checking Problem: Given a MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$, a 2^{Σ}-labeling function λ, a *regular* set of Markovian schedulers $\mathcal{A} = (Q_c, \rightarrow, q_0, F)$ and a Büchi automata $\mathcal{B}$ over the alphabet 2^{Σ}, determine if $L^{\lambda}_{\mathcal{A}}(\mathcal{M}) \cap L(\mathcal{B}) = \emptyset$.

VI. Model Checking MDP under indicator labelling functions

In this section we show that under indicator labelling functions model checking MDPs is decidable. We, in fact, show that the problem is PSPACE-complete.

Proposition VI.1. *Given a MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$ and a regular set of Markovian schedulers $\mathcal{A} = (Q_c, \rightarrow, q_0, F)$, $L^{\lambda}_{\mathcal{A}}(\mathcal{M})$ is ω-regular with respect to a indicator labelling function λ.*

Proof: We define a relation $\cong$ between states of the transition system $\mathcal{T}(\mathcal{M})$. Two states μ_1 and μ_2 are related ($\mu_1 \cong \mu_2$) if $\forall s \in S.\ \mu_1(s) > 0$ iff $\mu_2(s) > 0$. By definition of a indicator labelling function if $\mu_1 \cong \mu_2$ then $\lambda(\mu_1) = \lambda(\mu_2)$. Suppose $\mu_1 \cong \mu_2$ and $\mu_1 \xrightarrow{P} \mu_1'$ for some $P \in \mathcal{P}$ i.e., $\mu_1 \cdot P = \mu_1'$. Now, consider $\mu_2' \in Dist(S)$ such that $\mu_2' = \mu_2 \cdot P$. Since, by definition, $\forall s \in S.\ \mu_1'(s) = \sum_{s' \in S} \mu_1(s') \cdot P(s', s)$ and $\forall s \in S.\ \mu_2'(s) = \sum_{s' \in S} \mu_2(s') \cdot P(s', s)$, we can infer that $\forall s \in S.\ \mu_1'(s) > 0$ iff $\mu_2'(s) > 0$ and thus $\mu_1' \cong \mu_2'$. Similarly, if we consider a transition from μ_2, we can show there is a matching transition from μ_1. Hence the relation $\cong$ is a bisimulation relation over the state space of $\mathcal{T}(\mathcal{M})$ with respect to the indicator labelling function λ. By definition, bisimulation relation $\cong$ has finite number ($2^{|S|}$) of equivalence classes over the state space of $\mathcal{T}(\mathcal{M})$.

Let $(G, \rightarrow_g)$ be the quotient of $\mathcal{T}(\mathcal{M})$ with respect to $\cong$ ($\mathcal{T}(\mathcal{M})/\cong$), i.e., each state $g \in G$ is an equivalence class of $\cong$ and $g_1 \xrightarrow{P}_g g_2$ iff $\exists \mu_1 \in g_1$ and $\mu_2 \in g_2$ such that $\mu_1 \xrightarrow{P} \mu_2$. The language $L^{\lambda}_{\mathcal{A}}(\mathcal{M})$ is recognized by the Büchi automaton $\mathcal{C}$ constructed by taking the synchronous product of $(G, \rightarrow_g)$ with $\mathcal{A}$. More formally, $\mathcal{C}$ represents the Büchi automata $(G \times Q_c, \rightarrow_c, (g_0, q_0), G \times F)$, where g_0 correspond to the equivalence class containing initial distribution μ_0 and $\rightarrow_c$ is defined as follows: $\forall (g_1, q_1), (g_2, q_2) \in G \times Q_c, P \in \mathcal{P}.\ (g_1, q_1) \xrightarrow{P}_c (g_2, q_2)$ iff $g_1 \xrightarrow{P}_g g_2 \wedge q_1 \xrightarrow{P} q_2$. Thus, the language $L^{\lambda}_{\mathcal{A}}(\mathcal{M})$ is ω-regular, with respect to the indicator labelling function λ. ■

[1] There are other types of schedulers such as randomized schedulers where next action is chosen probabilistically.

Proposition VI.2. *The problem of model checking an MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$ for a Büchi automata $\mathcal{B}$ with respect to an indicator labelling function λ and a* regular *set of Markovian schedulers $\mathcal{A} = (Q_c, \rightarrow, q_0, F)$ is in PSPACE.*

Proof: From the proof of Proposition 1, the model checking problem is equivalent to checking if $L(\mathcal{C}) \cap L(\mathcal{B}) = \emptyset$, where $\mathcal{C}$ is the Büchi automaton constructed in the proof of Proposition 1. The complexity bound is a consequence of the following observations. First, the Büchi automaton $\mathcal{Z}$ that is the cross-product of $\mathcal{C}$ and $\mathcal{B}$ which recognizes $L(\mathcal{C}) \cap L(\mathcal{B})$ has $2^{O(|S|)}$ states; thus a single state requires space $O(|S|)$. Next, the transition relation of $\mathcal{Z}$ can be computed in $O(\log |S|)$ space, since it involves computing a matrix product. Finally, the emptiness problem of Büchi automata is in NL. ∎

Proposition VI.3. *The problem of model checking an MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$ for a Büchi automata $\mathcal{B}$ with respect to an indicator labelling function λ and a* regular *set of Markovian schedulers $\mathcal{A} = (Q_c, \rightarrow, q_0, F)$ is in PSPACE-hard.*

Proof: The hardness result is proved by reducing the acceptance problem of *linear bounded automata (LBA)* [20] to the model checking problem for MDPs under a regular set of Markovian schedulers. Recall that an LBA is single tape Turing machine that on input x is constrained to only read/write on the tape cells that contain the input x. Formally, an LBA is $\mathcal{W} = (Q, \Delta, \Gamma, \delta, q_0, q_f)$, where Q is a finite set of control states, Δ and Γ are the input and tape alphabets, respectively, with $\Delta \subsetneq \Gamma$ and the blank symbol $\sqcup \in \Gamma \setminus \Delta$, q_0 is the initial state, q_f is the sole accepting state with the assumption that there are no transition out of q_f, and $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{\text{left}, \text{right}\}$ is the transition function that given a current state and current symbol being read, describes what the next state, symbol written and direction of moving the tape head is. The input x for $\mathcal{W}$ is given between a left end marker $\rhd$ and a right end marker $\lhd$, with the assumption that $\mathcal{W}$ moves right as soon as it reads $\rhd$, and moves left when it reads $\lhd$. The acceptance problem of LBAs asks, given an input x and an LBA $\mathcal{W}$, does $\mathcal{W}$ accept the input x (i.e., reach the sole accepting state q_f on x).

For the rest of this proof let us fix an input x of length n and an LBA $\mathcal{W} = (Q, \Delta, \Gamma, \delta, q_0, q_f)$. Let $\Phi = \Gamma \times Q$ be the set of composite symbols. Recall that a configuration of $\mathcal{W}$ during a computation on x can be encoded by a string of length n in $\Gamma^* \Phi \Gamma^*$ that describes the contents of all the tape cells with the composite symbol (in Φ) indicating the head position as well as the control state. The initial configuration on $x = x_1 x_2 \cdots x_n$ is $(q_0, x_1) x_2 x_3 \cdots x_n$. A computation is a sequence of configurations, starting from the initial configuration, such that each succeeding one is obtained by a step of $\mathcal{W}$, and the computation is accepting if the last configuration in the sequence is in control state q_f.

We will construct a MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$, a regular set of Markovian schedulers $\mathcal{A}$, a indicator labelling function λ over Σ such that there is an execution of $\mathcal{T}(\mathcal{M})$ following a Markovian scheduler from $\mathcal{A}$ that eventually reaches a state labeled say $g_1 \wedge g_2 \wedge \cdots \wedge g_k$, where $g_i \in \Sigma$ iff $\mathcal{W}$ accepts input x. Before giving the formal construction of $\mathcal{M}$ and $\mathcal{A}$, we give the intuition behind it. The MDP states S will be used to store the current configuration in the computation. A configuration c will be encoded by a distribution μ on S as follows: if an MDP state (i, a) has non-zero probability in μ, where $a \in \Gamma \cup \Phi$, then it denotes that the ith cell of the configuration c contains symbol a. The initial distribution μ_0 will be the encoding of the initial configuration; thus, $\mu_0((i, a)) = 1/n$ iff $a = x_i$ (when $i > 1$), or $a = (q_0, x_1)$ (when $i = 1$). The transitions of $\mathcal{T}(\mathcal{M})$ will guess the position of the head, control state of $\mathcal{W}$ and the symbol being scanned in the current configuration, and the matrix labeling the transition will ensure that the probability distribution changes to reflect the new configuration; if the guess is incorrect then there will be a transition to an error MDP state to reflect the error. Thus, if there is an accepting computation x then the $\mathcal{T}(\mathcal{M})$ will have an execution where one of the cells is an accepting state and no error MDP states are reached; we will use the labelling function to check this property.

Having outlined the intuition, we now present the formal construction. The states of MDP are $S = \{1, \ldots n\} \times (\Gamma \cup \Phi \cup \{q_{\text{err}}\})$. The initial distribution μ_0 being such that $\mu_0(i, a) = 1/n$ if $i = 1$ and $a = (q_0, x_1)$ or $i > 1$ and $a = x_i$, and $\mu_0(i, a) = 0$ in all other cases. The set of stochastic matrices $\mathcal{P}$ contains matrices P_τ, and $P_{(i,a,q)}$, for each $(i, a, q) \in \{1, \ldots n\} \times \Gamma \times Q$. P_τ is the identity matrix and $P_{(i,a,q)}$ is a stochastic matrix that "simulates" the step of $\mathcal{W}$ when the state is q, the head is in position i, and a is being read. We now define $P_{(i,a,q)}$. Let $\delta(q, a) = (q', a', d)$, where $d \in \{\text{left}, \text{right}\}$. Then, for a distribution μ on S, $\mu \cdot P_{(i,a,q)}((j, b))$ is given in Fig 4

$\mathcal{A}$ is the Büchi automata $(q, \{(q, M, q) | M \in \mathcal{P}\}, q, \{q\})$, which accepts all possible Markovian schedulers over alphabet $\mathcal{P}$. Let us call a distribution μ to be good if for every i, $\mu(i, c) > 0$ for exactly one c. Observe that if μ is good, then so is $\mu \cdot P_{(i,a,q)}$. The reduction relies on the observation that good distributions encode a configuration of $\mathcal{W}$; contents of cell i is c iff $\mu((i, c)) > 0$ and if $\mu((i, q_{\text{err}})) > 0$ then μ encodes an error configuration. The definition of $P_{(i,a,q)}$ is given based on making sure that it simulates one step of the machine on the basis that good distributions encode configurations.

For a configuration c of $\mathcal{W}$ let $\text{strip}(c)$ be (i, a, q) if $c_i = (a, q)$ is the unique composite symbol in c. Now observe that $c_1, c_2, \ldots c_k$ is an accepting computation of $\mathcal{W}$ on x iff $P_{\text{strip}(c_1)} P_{\text{strip}(c_2)} \cdots P_{\text{strip}(c_k)}$ defines a accepting scheduler in $\mathcal{A}$ that leads to μ, where μ is a good distribution that encodes c_k. This is the crux of the correctness proof of the reduction. Checking the existence of such a path can be reduced to model checking with respect to an appropriate property. Let $\Sigma = \{g_1, \ldots g_n\}$ and a labeling function λ as follows: $g_i \in \lambda(\mu)$ iff $\sum_{c \in \Gamma \cup (\Gamma \times \{q_f\})} \mu(i, c) > 0$. In other words, μ is labeled g_i if the ith cell is either a symbol in Γ or a composite symbol with state being q_f. Thus, $\mathcal{W}$ accepts x iff $\mathcal{M}$ under $\mathcal{A}$ satisfies the property that there is an execution that eventually reaches a configuration with label $\wedge_i g_i$. ∎

$$\mu \cdot P_{(i,a,q)}((j,b)) = \begin{cases} \mu((i,(q,a))) & \text{if } j=i \text{ and } b=a' \\ \sum_{c\neq(a,q)} \mu((i,c)) & \text{if } j=i \text{ and } b=q_{\text{err}} \\ 0 & \text{if } j=i \text{ and } b\notin\{a',q_{\text{err}}\} \\ \mu((i+1,b')) & \text{if } j=i+1, \\ & \quad d=\text{right, and } b=(b',q') \\ \sum_{c\in\Phi\cup\{q_{\text{err}}\}} \mu((i+1,c)) & \text{if } j=i+1, \\ & \quad d=\text{right, and } b=q_{\text{err}} \\ 0 & \text{if } j=i+1, \\ & \quad d=\text{right, and } b\in\Gamma \\ \mu((i-1,b')) & \text{if } j=i-1, \\ & \quad d=\text{left, and } b=(b',q') \\ \sum_{c\in\Phi\cup\{q_{\text{err}}\}} \mu((i-1,c)) & \text{if } j=i-1, \\ & \quad d=\text{left, and } b=q_{\text{err}} \\ 0 & \text{if } j=i-1, \\ & \quad d=\text{left, and } b\in\Gamma \\ \mu(j,b) & \text{otherwise} \end{cases}$$

Fig. 4. Definition of $\mu \cdot P_{(i,a,q)}((j,b))$

Theorem VI.4. *The problem of model checking an MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$ for a Büchi automata $\mathcal{B}$ with respect to an indicator labelling function λ and a* regular *set of Markovian schedulers $\mathcal{A} = (Q_c, \rightarrow, q_0, F)$ is PSPACE-complete.*

Proof: Follows from propositions VI.2 and VI.3. ■

VII. Model Checking MDPs under Linear Labeling Functions

In the previous section, we showed that the model checking problem for MDPs with respect to indicator labelling functions is PSPACE-complete. A natural question to ask is what is the complexity of the model checking problem when we consider more general kinds of labeling functions. The main results of this section are 1) model checking problem is undecidable with respect to linear labeling functions, and 2) model checking problem is decidable for specific type of MDPs under specific Markovian schedulers with respect to linear labeling functions. We first present the undecidability result before concluding the section with the decidability result.

A. Undecidability Proof for Linear Labeling Functions

Our undecidability proof relies on reducing the emptiness problem of *probabilistic finite automata (PFA)*, which is known to be undecidable [10]. Therefore, before presenting the result we introduce probabilistic finite automata.

Formally, a probabilistic finite automata (PFA) over alphabet Δ is $\mathcal{F} = (Q, q_0, F, \delta)$, where Q is a finite set of states, $q_0 \in Q$ is the initial state, $F \subseteq Q$ is the set of final states, and $\delta : Q \times \Delta \rightarrow \text{Dist}(Q)$ is the transition function. It is convenient to view the transition function δ on input a as a stochastic matrix δ_a over Q where the q'th row of δ_a is $\delta(q,a)$. Given a word $x = x_1 x_2 \cdots x_n$, δ_x is defined to be the matrix product $\delta_{x_1}\delta_{x_2}\cdots\delta_{x_n}$; $\delta_x(q,q')$ gives the probability of going from state q to q' on input x. Finally, given a word x and $A \subseteq Q$, $\delta_x(q,A) = \sum_{p\in A}\delta_x(q,p)$. Given $t \in \mathbb{R}$, the language of $\mathcal{F}$ with threshold t, denoted by $\mathcal{L}_{>t}(\mathcal{F})$, is defined as $\{x \in \Sigma^* | \delta_x(q_0, F) > t\}$. In other words, the language is the set of all words that, starting from the initial state q_0, reach an accepting state with probability $> t$. A celebrated result about PFAs due to Condon-Lipton [10] is that the emptiness problem for PFAs (with respect to any non-zero threshold) is undecidable.

Theorem VII.1 (Condon-Lipton [10])**.** *Given a PFA $\mathcal{F}$ over alphabet Δ the problem of determining if $\mathcal{L}_{>\frac{1}{2}}(\mathcal{F}) = \emptyset$ is undecidable.*

Using this result we can show,

Theorem VII.2. *The problem of model checking a MDP $\mathcal{M}$ with respect to a linear labelling function λ is undecidable.*

Proof: We will prove the theorem by reducing the emptiness problem of PFAs to our model checking problem. Let $\mathcal{F} = (Q, q_0, F, \delta)$ be a PFA over alphabet Δ. We will construct an MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$ and a regular set of Markovian schedulers $\mathcal{A}$, such that the restricted transitions of the transition system $\mathcal{T}(\mathcal{M})$ will guess the string in the language of $\mathcal{F}$, and at the same time faithfully simulate $\mathcal{F}$ on the input. The formal definition of $\mathcal{M}$ and $\mathcal{P}$ is as follows. S will be equal to Q. The initial distribution μ_0 is such that $\mu_0(q_0) = 1$, and $\mu_0(q) = 0$ for all $q \neq q_0$. The set of stochastic matrices $\mathcal{P}$ contains matrices δ_a where $a \in \Delta$. $\mathcal{A}$ is the Büchi automata $(q, \{(q, P, q) | P \in \mathcal{P}\}, q, \{q\})$, which accepts all possible Markovian schedulers over alphabet $\mathcal{P}$. Observe that if there is a execution sequence $\mu_0, \mu_1, \ldots \mu_{n+1}$ and $x = x_1 x_2 \cdots x_n$ such that $\mu_i = \mu_{i-1} \cdot \delta_{x_i}$ then $\mu_{n+1}(F) = \delta_x(q_0, F)$; the converse of this statement also holds.

Consider the set $\Sigma = \{g\}$ and labeling function λ such that $g \in \lambda(\mu)$ iff $\sum_{q\in F}\mu(q) > \frac{1}{2}$. Clearly based on the observations in the previous paragraph, $\mathcal{M}$ under $\mathcal{A}$ has an execution that eventually reaches a configuration labeled g iff $\mathcal{L}_{>\frac{1}{2}}(\mathcal{F}) \neq \emptyset$. Thus, the undecidability of the model checking problem for MDPs follows from Theorem VII.1. ■

B. Decidability Results under Linear Labeling Functions

In this section, we identify restrictions on schedulers and MDPs that ensure that the model checking problem is decidable. Specifically, we consider contracting MDPs and schedulers that can be recognized by "almost acyclic" Büchi automata. Before presenting the main decidability result of this section, we formally define the restrictions on schedulers and MDPs that ensure decidability.

Almost acyclic set of schedulers. A regular set of Markovian schedulers $\mathcal{A} = (Q_c, \rightarrow, q_0, F)$ is called almost acyclic iff there is a total order $>$ on Q_c such that the following conditions hold: (a) If for $q_1 \neq q_2$, there is $P \in \mathcal{P}$ such that $q_1 \xrightarrow{P} q_2$ then $q_1 > q_2$, and (b) For any $q \in Q_c$, if P_1 and P_2 are such that $q \xrightarrow{P_1} q$ and $q \xrightarrow{P_2} q$ then $P_1 = P_2$. Thus, informally, $\mathcal{A}$ is almost acyclic, if the transition graph of $\mathcal{A}$ is acyclic with optional self loops on states. The relation $>$ on Q_c in the above definition will be referred to as the order of the almost acyclic set of schedulers.

Stability with respect to labeling. We will focus on contracting MDPs, which are MDPs all of whose transition matrices are regular. We will require that the behaviors of the MDPs under a set of schedulers be "stable" with respect to the labeling function on distributions. Intuitively, stability can be understood as follows. Recall that since the schedulers we consider are acyclic, any scheduler in the set is of the form $s = wP^\omega$, where w is a finite sequence of matrices. Further, since the MDP is contracting, this means that the execution under such a scheduler s converges to a unique distribution. Stability requires that if one looks at the labels of any such execution it also eventually stabilizes (just like the sequence of distributions). Thus it requires that the limiting distribution of such a sequence be in the interior (rather than the boundary) of sets of distributions that are relevant from the perspective of language equivalence. We will now formalize this intuition. In order to do this we first identify partitions on distributions that play a role in defining the labeled language, and later use these partitions to define stability.

Let us fix a contracting MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$, an acyclic set of Markovian schedulers $\mathcal{A} = (Q_c, \rightarrow, q_0, F)$, and a linear 2^Σ-labeling function λ. Let $\mu_1, \mu_2 \in \mathrm{Dist}(S)$. We say $\mu_1 \approx_\lambda \mu_2$ iff $\lambda(\mu_1) = \lambda(\mu_2)$. For every, state $q \in Q_c$, define the following two equivalences on $\mathrm{Dist}(S)$. $\mu_1 \approx_q \mu_2$ iff $L^\lambda_{\mathcal{A}'}(\mathcal{M}, \mu_1) = L^\lambda_{\mathcal{A}'}(\mathcal{M}, \mu_2)$, where $\mathcal{A}'$ is the Büchi automaton $(Q_c, \rightarrow, q, F)$. Next, we say $\mu_1 \approx'_q \mu_2$ iff $L^\lambda_{\mathcal{A}''}(\mathcal{M}, \mu_1) = L^\lambda_{\mathcal{A}''}(\mathcal{M}, \mu_2)$ where $\mathcal{A}''$ is the Büchi automaton $(Q_c, \rightarrow_q, q, F)$ where

$$\rightarrow_q = \begin{cases} \rightarrow \setminus \{(q, P, q) \mid P \in \mathcal{P}\} & \text{if } \exists q' \in Q_c, P \in \mathcal{P}.\, q \xrightarrow{P} q' \\ \rightarrow & \text{otherwise} \end{cases}$$

Notice, that if q does not have any outgoing transitions or self loops then $\approx_q = \approx'_q$.

We are now ready to define the notion of stability.

Definition An MDP $\mathcal{M} = (S, \mu_0, \mathcal{P})$ under an almost acyclic set of schedulers $\mathcal{A} = (Q_c, \rightarrow, q_0, F)$ is stable with respect to a linear 2^Σ-labeling function λ if

$$\forall q \in Q_c.\, (\exists P \in \mathcal{P}.\, q \xrightarrow{P} q) \Rightarrow$$

$$(\mathcal{S}(P) \in \mathcal{I}(\approx_\lambda) \wedge (\exists P' \in \mathcal{P}, q' \in Q_c. q \xrightarrow{P'} q' \Rightarrow \mathcal{S}(P) \in \mathcal{I}(\approx'_q)))$$

where $\mathcal{S}(P)$ is the unique limiting distribution of P, and $\mathcal{I}(\approx)$ for an equivalence relation, is the union of the interiors [2] of all of the equivalence classes of $\approx$.

Theorem VII.3. *Let $\mathcal{M} = (S, \mu_0, \mathcal{P})$ be a contracting MDP, $\mathcal{A} = (Q_c, \rightarrow, q_0, F)$ be an acyclic set of Markovian schedulers, and λ be a linear 2^Σ-labeling function. Suppose $\mathcal{M}$ under $\mathcal{A}$ is stable with respect to λ. Then $L^\lambda_{\mathcal{A}}(\mathcal{M})$ is recognized by a Büchi automaton that can be effectively constructed. Thus, model checking $\mathcal{M}$ against any Büchi specification $\mathcal{B}$ is decidable.*

Proof: Observe that since $\mathcal{M}$ is contracting, and is stable under $\mathcal{A}$ with respect to λ, $L^\lambda_{\mathcal{A}}(\mathcal{M})$ is of the form $\cup_i L_i p_i^\omega$ where $p_i \in 2^\Sigma$, and $L_i \subseteq (2^\Sigma)^*$. Thus, $L^\lambda_{\mathcal{A}}(\mathcal{M})$ is a $\mathcal{F}_\sigma \cap \mathcal{G}_\delta$ in the Cantor topology [3]. An important property [21], [22] of $\mathcal{F}_\sigma \cap \mathcal{G}_\delta$ sets is as follows. Consider the following equivalence on strings of finite length: $u \equiv_L v$ iff for all $\alpha \in \Sigma^\omega$, $u\alpha \in L$ iff $v\alpha \in L$. Now, $L \in \mathcal{F}_\sigma \cap \mathcal{G}_\delta$ is regular iff $\equiv_L$ has finitely many equivalence classes. Moreover, there is a unique deterministic Büchi automaton of minimum number of states, whose states correspond to the equivalence classes of $\equiv_L$. For $L_0 = L^\lambda_{\mathcal{A}}(\mathcal{M})$, we will show that $\equiv_{L_0}$ has finitely many equivalence classes that can be effectively constructed. We prove this by showing that $\approx_q$ has finitely many equivalence classes (for every $q \in Q_c$) that can be effectively constructed and by the fact that $|\equiv_{L_0}| \leq \sum_q |\approx_q|$, where $|\equiv|$ denotes the number of classes in $\equiv$.

We will show that $\approx_q$ has finite index and can be effectively constructed by showing that both $\approx_q$ and $\approx'_q$ (for every q) are of finite index and can be effectively expressed in the first order theory of reals. We begin by observing that $\approx_\lambda$ has finite index (bounded by $2^{|\Sigma|}$) and can be expressed in the first order theory of reals. The proof that $\approx_q$ and $\approx'_q$ have the necessary properties, proceeds by induction on $>$, the order of the almost acyclic Markovian Schedulers $\mathcal{V}$.

Base Case. If q is the least state (w.r.t. $>$) then there can be only one transition out of q, namely of the form $q \xrightarrow{P} q$, where P is a regular matrix. Thus, $\approx_q$ is the same as $\approx'_q$. Since $\mathcal{M}$ under $\mathcal{A}$ is stable with respect to λ, by definition, $\mathcal{S}(P) \in \mathcal{I}(\approx_\lambda)$. In other words, there exists an ϵ, such that $B(\mathcal{S}(P); \epsilon)$ is fully contained in a single equivalence class of $\approx_\lambda$. Furthermore, since P is contracting and has a unique limiting distribution (as it is regular), there exists k such that $\forall l \geq k.\, \forall \mu \in \mathrm{Dist}(S).\, d(\mu \cdot P^l, \mathcal{S}(P)) < \epsilon$. Therefore, all distributions after k matrix multiplications will be end up in a single equivalence class of $\approx_\lambda$. We define an equivalence relation $\sim_q$ over $\mathrm{Dist}(S)$ such that $\forall \mu_1, \mu_2 \in \mathrm{Dist}(S).\, \mu_1 \sim_q \mu_2$ iff $(\lambda(\mu_1) = \lambda(\mu_2)) \wedge (\forall i < k. (\lambda(\mu_1 \cdot P^i) = \lambda(\mu_2 \cdot P^i)))$. Thus, from the observation above, we can conclude that if $\mu_1 \sim_q \mu_2$ then $L^\lambda_{\mathcal{A}'}(\mathcal{P}, \mu_1) = L^\lambda_{\mathcal{A}'}(\mathcal{P}, \mu_2))$ where $\mathcal{A}'$ correspond to the Büchi automata $(Q_c, \rightarrow, q, F)$. Therefore, the equivalence $\sim_q$ is a refinement of the equivalence $\approx_q$. Moreover, since the number of equivalence classes of $\sim_q$ is bounded by $((2^{|\Sigma|})^k)$, $\approx_q$ has finite index. Furthermore, since each equivalence class of $\sim_q$ corresponds to a sequence $\ell_1 \ell_2 \cdots \ell_k E$, where $\ell_i \subseteq \Sigma$ and E is an equivalence class of $\approx_\lambda$, they can

[2] Recall that interior of a set X is the largest open set contained within X. Here we are using the topology defined by the metric d on distribution defined in Lemma IV.2.

[3] Recall that the open sets (denoted by $\mathcal{G}$) in the Cantor topology are all the sets of the form $L\Sigma^\omega$, where $L \subseteq \Sigma^*$; the closed sets $\mathcal{F}$, are subsets of Σ^ω whose complements are in $\mathcal{G}$. Finally, $\mathcal{G}_\delta$ are the collection of all sets that can be written as a countable intersection of open sets, and $\mathcal{F}_\sigma$ are all the sets that can be written as a countable union of closed sets.

be effectively expressed in the first order theory since the labelling function is linear and $\approx_\lambda$ is expressible. Finally, since $\sim_q$ is a refinement of $\approx_q$ (and $\approx'_q$) the base case is proved.

Case 2. (States with no self loops) Let q be a state with no self loops. For such a state we know that if $q \xrightarrow{P} q'$ then $q' < q$. Moreover, once again, $\approx_q$ is the same as $\approx'_q$. Consider an equivalence relation $\sim_q$ over $\mathrm{Dist}(S)$ such that $\mu_1 \sim_q \mu_2$ iff $(\lambda(\mu_1) = \lambda(\mu_2)) \wedge (\forall q' \in Q. \exists P \in \mathcal{P}. q \xrightarrow{P} q' \Rightarrow (\mu_1 \cdot P \approx_{q'} \mu_2 \cdot P))$. We can see that $\forall \mu_1, \mu_2 \in \mathrm{Dist}(S).\ \mu_1 \sim_q \mu_2 \Rightarrow L^\lambda_{\mathcal{A}'}(\mathcal{P}, \mu_1) = L^\lambda_{\mathcal{A}'}(\mathcal{P}, \mu_2))$ where $\mathcal{A}'$ correspond to the Büchi automata $(Q_c, \rightarrow, q, F)$. Therefore, the equivalence relation $\sim_q$ is a refinement of the equivalence relation $\approx_q$. Finiteness of the number of equivalence classes and its effective expression in the first theory of reals for the equivalence relation $\sim_q$ (and therefore, $\approx_q$ and $\approx'_q$) follows from the induction hypothesis about $\approx_{q'}$.

Case 3. (State with both self loops and successors) Let q be a state that has self loops and transitions to other states. Observe that by a reasoning similar to Case 2 above, we can conclude that $\approx'_q$ has finite index and can be effectively expressed in the first order theory of reals. Let P be *the* regular matrix such that $q \xrightarrow{P} q$ is a transition. Once again, the stability condition ensures that $\mathcal{S}(P) \in \mathcal{I}(\approx_\lambda)$ and $\mathcal{S}(P) \in \mathcal{I}(\approx'_q)$. In other words, there exists an ϵ, such that $B(\mathcal{S}(P); \epsilon)$ is contained in a single equivalence class of $\approx_\lambda$ and also contained in a single equivalence class of $\approx'_q$. Moreover, since P is a regular stochastic matrix, by definition, there exists k such that $\forall l \geq k. \forall \mu \in \mathrm{Dist}(S). d(\mu \cdot P^l, \mathcal{S}(P)) < \epsilon$. Therefore, all distributions after k matrix multiplications will be end up in a single equivalence class of $\approx_\lambda$ and also in a single equivalence class of $\approx'_q$. From the above observations, using an argument very similar to the one used in the Base Case, we can show that $\approx_q$ is of finite index and can be effectively expressed in the first order theory of reals.

Thus we can construct a deterministic Büchi automaton that recognizes $L^\lambda_{\mathcal{A}}(\mathcal{M})$. The model checking algorithm will construct this machine, intersect with $\mathcal{B}$, and check for emptiness. ∎

VIII. Analyzing the Compartment Model of Insulin-^{131}I

In this section, we show that the MDP $\mathcal{M} = (\{Dr, Pl, IF, Ut, Cl, Re\}, \mu_0, \{N, S\})$ corresponding to compartment model described in section III is contracting and is stable under the regular set of schedulers $\mathcal{V}$. Since both the matrices S and N have only single closed class $\{Cl\}$ which is obviously aperiodic, they are regular matrices. Thus by definition, MDP $\mathcal{M}$ is also contracting. Note that the limiting distribution of both these matrices is an unit distribution μ_l at state *Cl*. In other words, in the long run, all the drug will end up in the compartment *Cl*. Given the labels *effective* and *nontoxic*, as defined earlier, consider the set of distributions $\mathcal{D}$ such that $\mathcal{D} = \{\mu \mid \mu(Cl) > 1 - mec\}$. Note that all distributions in $\mathcal{D}$ do not satisfy *effective*. Furthermore, since $mtc > mec$, all distributions in the set $\mathcal{D}$ trivially satisfies proposition *nontoxic*. Observe that, all the distributions appearing along any execution starting from a distribution in $\mathcal{D}$ are once again in the set $\mathcal{D}$. Specifically, once the proportion of drug in compartment *Cl* is more than $1 - mec$, the proportion of drug in compartment *Ut* remains less than mec. Therefore, the set of distributions $\mathcal{D}$ is a single equivalence class of $\simeq_{q_0}$, where q_0 is the initial state of the automaton recognizing the regular set of schedulers $\mathcal{V}$. Furthermore, since the limiting distribution μ_l of both S and N is in the interior of the set $\mathcal{D}$, the MDP $\mathcal{M}$ under $\mathcal{V}$ is stable. Thus by Theorem VII.3 model checking $\mathcal{M}$ against any Büchi specification $\mathcal{B}$ is decidable.

References

[1] M. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc. New York, NY, USA, 1994.
[2] M. Y. Vardi, "Automatic verification of probabilistic concurrent finite-state programs," in *FOCS*, 1985, pp. 327–338.
[3] C. Courcoubetis and M. Yannakakis, "Markov decision processes and regular events," in *ICALP*, 1990, pp. 336–349.
[4] D. White, "A Survey of Applications of Markov Decision Processes," *Journal of the Operational Research Society*, pp. 1073–1096, 1993.
[5] J. M. Rutten, M. Kwiatkowska, G. Norman, and D. Parker, *Mathematical Techniques for Analyzing Concurrent and Probabilistic Systems*. AMS, 2004.
[6] H. Hansson and B. Jonsson, "A logic for reasoning about time and reliability," *Formal Aspects of Computing*, vol. 6, no. 5, pp. 512–535, 1994.
[7] D. M. Roberts and N. A. Buckley, "Pharmacokinetic considerations in clinical toxicology," in *Clinical Pharmacokinetics*, vol. 46, no. 11, 2007, pp. 897–939.
[8] Y. Kwon and G. Agha, "Linear inequality ltl (iltl): A model checker for discrete time markov chains," in *ICFEM*, 2004, pp. 194–208.
[9] A. Platis, N. Limnios, and M. Le Du, "Performability of Electric-Power Systems Modeled by Non-Homogeneous Markov Chains," *IEEE Transactions on Reliability*, vol. 45, no. 4, pp. 605–610, 1996.
[10] A. Condon and R. J. Lipton, "On the complexity of space bounded interactive proofs (extended abstract)," in *FOCS*, 1989, pp. 462–467.
[11] Y. Kwon and G. Agha, "A markov reward model for software reliability," in *IPDPS*, 2007, pp. 1–6.
[12] A. Hinton, M. Z. Kwiatkowska, G. Norman, and D. Parker, "Prism: A tool for automatic verification of probabilistic systems," in *TACAS*, 2006, pp. 441–444.
[13] D. Beauquier, A. M. Rabinovich, and A. Slissenko, "A logic of probability with decidable model-checking," in *CSL*, 2002, pp. 306–321.
[14] P. Baltazar, P. Mateus, R. Nagarajan, and N. Papanikolaou, "Exogenous Probabilistic Computation Tree Logic," *QAPL*, pp. 95–110, 2007.
[15] L. Shargel and A. B.C., "Applied biopharmaceutics and pharmacokinetics." Appleton-Century-Crofts, 1985.
[16] A. Silvers, R. S. Swenson, J. W. Farquhar, and G. M. Reaven, "Derivation of a three compartment model describing disappearance of plasma insulin-131i in man," vol. 48. The Journal of Clinical Investigation, 1969.
[17] Y. Kwon and E. Kim, "Specification and verification of pharmacokinetic models," in *Computational Biology and Bioinformatics (to appear)*. Springer, 2010.
[18] J. Norris, *Markov Chains*. Cambridge University press, 1997.
[19] F. R. Gantmakher, *Applications of the theory of matrices*. Interscience Publishers Newyork, 1959.
[20] J. Ullman and J. Hopcroft, "Introduction to Automata Theory, Languages, and Computation," 1979.
[21] O. Maler and L. Staiger, "On syntactic congruences for omega-languages," *Theoretical Computer Science*, vol. 183, no. 1, pp. 93–112, 1997.
[22] O. Maler and A. Pnueli, "On the learnability of infinitary regular sets," *Information and Computation*, vol. 118, no. 2, pp. 316–326, 1995.

APPENDIX

A. Proof of Lemma IV.2

Before we prove this lemma, we prove the following useful proposition regarding regular stochastic matrices.

Proposition A.1. *Given a stochastic matrix P, following statements are equivalent.*

(i) *Matrix P is regular.*

(ii) $\{q_i \mid \exists k \geq 0. \forall q_j \in S. P^k(q_j, q_i) > 0\}$ *is non empty.*

(iii) $\exists I, b$ *such that* $I \neq \emptyset \wedge I \subseteq S \wedge \forall \epsilon < b. \exists j. \forall k \geq j.$
- $P^k(s_1, s_2) > b$ *if* $s_2 \in I$ *where* $s_1, s_2 \in S$
- $P^k(s_1, s_2) \leq \epsilon$ *if* $s_2 \notin I$ *where* $s_1, s_2 \in S$

Proof: $((i) \Rightarrow (ii))$ Suppose there is only one non empty set $I \subseteq S$ which is both closed and aperiodic. Now, consider an arbitrary state $q_i \in I$, it is aperiodic i.e, $\exists l. \forall k \geq l. P^k(q_i, q_i) > 0$. Suppose l_i be one such l. Moreover, since state q_i belongs to the single closed set, it can be reached from all states by definition i.e, $\forall q_j \in S. \exists k \geq 0. P^k(q_j, q_i) > 0$. It is enough to prove that state q_i satisfies the condition $\exists k \geq 0. \forall q_j \in S. P^k(q_j, q_i) > 0$. Consider some arbitrary state $q_j \in S$, by above observation, there exists $k_j \geq 0$ such that $P^{k_j}(q_j, q_i) > 0$. Additionally, since q_i is aperiodic, we can infer that $\forall k \geq (k_j + l_i). P^k(q_j, q_i) > 0$. By taking $k_i = max_{q_j \in S} k_j + l_i$, it is evident that $\forall q_j \in S. P^{k_i}(q_j, q_i) > 0$.

$((ii) \Rightarrow (iii))$ Suppose $T = \{q_i \mid \exists k \geq 0. \forall q_j \in S. P^k(q_j, q_i) > 0\}$ is non empty. Here we prove that the first condition in (iii) is valid for all states in T and the second condition is valid for all remaining states. In other words we prove that the required non-empty set I is T itself. Consider some state $q_i \in T$, it satisfies the condition that $\exists k \geq 0. \forall q_j \in S. P^k(q_j, q_i) > 0$. In other words, q_i can be reached from any state in k steps. Observe that in a stochastic matrix P, $\forall q_j \in S. \exists q_s \in S. P(q_j, q_s) > 0$. In other words, every state in a stochastic matrix has at least one successor. We now prove a stronger statement that there exists some k such state q_i can be reached from any state in any number of steps greater than or equal to k ($\exists k \geq 0. \forall l \geq k. \forall q_j \in S. P^l(q_j, q_i) > 0$ holds for state q_i) by induction on l. The statement is true for base case $l = k$. Suppose the statement is true for some $r > k$ i.e, $\forall q_j \in S. P^r(q_j, q_i) > 0$. For any arbitrary state $q_j \in S$, $P^{r+1}(q_j, q_i) = \Sigma_{q_z \in S} P(q_j, q_z) \cdot P^r(q_z, q_i)$. By inductive hypothesis and by the existence of successor for each state, it is clear that $P^{r+1}(q_j, q_i) > 0$ for all $q_j \in S$. Hence by induction, the condition that there exists some k such state q_i can be reached from any state in any number of steps greater than k is true. Similar condition holds for all other states in T. Putting all the conditions together, we can infer a stronger condition that $\exists k \geq 0. \forall l \geq k. \forall q_i \in T. \forall q_j \in S. P^l(q_j, q_i) > 0$.

Let k_{min} be the minimum k such that the above condition holds and p be the minimum positive probability in the stochastic matrix. Then, $\forall q_i \in T. \forall q_j \in S. P^{k_{min}}(q_j, q_i) > p^{k_{min}+1}$. In other words, probability of reaching any of states q_i in T from any state in k_{min} steps is greater than $p^{k_{min}+1}$. We now prove a stronger statement that probability of reaching any of the states q_i in T form any state in more than k_{min} steps is greater than $p^{k_{min}+1}$ ($\forall l \geq k_{min}. \forall q_j \in S. P^l(q_j, q_i) > p^{k_{min}+1}$) by induction on l. The statement is already proven for base case. Suppose that statement is true for some $r > k_{min}$ i.e., $\forall q_j \in S. P^r(q_j, q_i) > p^{k_{min}+1}$. For any arbitrary state $q_j \in S$,

$$
\begin{aligned}
P^{r+1}(q_j, q_i) &= \Sigma_{q_z \in S} P(q_j, q_z) \cdot P^r(q_z, q_i) \\
&> p^{k_{min}+1} \cdot \Sigma_{q_z \in S} P(q_j, q_z) \text{ (by Ind. Hyp)} \\
&= p^{k_{min}+1} \text{ (by definition) .}
\end{aligned}
$$

Hence by induction, probability of reaching any of the states q_i in T form any state in more than k_{min} steps is greater than $p^{k_{min}+1}$. Thus the first condition in (iii) is true with $I = T$ and $b = p^{k_{min}+1}$. It is enough to prove that the second condition holds for all states not in I but in S. Note that, starting from any node, the probability that we end up in I is greater than $p^{k_{min}+1}$. Let v denote $k_{min} + 1$. Now consider some state $q_i \in (S \setminus I)$. Let probability of reaching q_i from some state q_j in $(N \cdot v)$ steps be $P^{N \cdot v}(q_j, q_i)$ for some N. Then the following inequality holds

$1 - P^{Nv}(q_j, q_i)$ (probability of not reaching q_i from q_j in $N \cdot v$ steps)

$$
\begin{aligned}
&\geq p^v + (1 - p^v) \cdot p^v + (1 - p^v)^2 \cdot p^v \cdots (1 - p^v)^{N-1} \cdot p^v \\
&= 1 - (1 - p^v)^N
\end{aligned}
$$

Therefore $p^{Nv}(q_j, q_i) \leq (1 - p^v)^N$. Given an $\epsilon < b$ we can choose N such that $\forall k > 0. P^{Nv+k}(q_j, q_i) \leq \epsilon$. Thus all states in $(S \setminus I)$ satisfy the second condition. Hence the statement.

$((iii) \Rightarrow (i))$ We prove the contrapositive statement. Suppose there is no single closed class $I \subseteq S$ which is also aperiodic then either, there is no closed set or there are more than one closed set or there is only one closed set and is not aperiodic. For finite state Markov chains, there is at least one closed set in the transition matrix, thus first case is not possible. In the second case, suppose there is more than one closed set, by definition there will be no path from any state in one closed set to any another state in a different closed set. The first condition in (iii) does not hold for all states in these closed sets and trivially not satisfied for states which do not belong to any of these closed sets. Thus the set of states which satisfy the first condition is empty i.e, $I = \emptyset$. Hence a contradiction. In the final case, suppose there is only one closed set S_1 which is not aperiodic. From previous argument, it is true that first condition in (iii) does not hold for states not in the closed set. Consider some arbitrary state $q_i \in S_1$. Since q_i is aperiodic i.e., $\exists j. \forall k > j. P^k(q_i, q_i) > 0$, it does not satisfy the first condition in (iii). Hence, no states in the closed set satisfy the first condition. From both the arguments above, we can conclude that the set of states which satisfy the first condition is empty i.e, $I = \emptyset$. Hence a contradiction. ■

1) Proof of Lemma IV.2 : *Proof:* Take arbitrary $A \subseteq S$ and let $P(q_i, A) = \sum_{q_j \in A} P(q_i, q_j)$. Further let μ and μ' be probability distributions on S with $\mu \neq \mu'$, and denote $B = \{q_i \in S \mid \mu(q_i) \geq \mu'(q_i)\} \subset S$. Suppose there exist b (by proposition A.1) such that for some large enough k, $\forall q_i \in S. \forall q_j \in I. P^k(q_i, q_j) > b$ holds. Now we consider two cases. Case 1: $(A \cap I \neq \emptyset)$

$$\begin{aligned}
&|\mu \cdot P^k(A) - \mu' \cdot P^k(A)| \\
=& \left| \sum_{q_i \in S} \mu(q_i) \cdot P^k(q_i, A) - \sum_{q_i \in S} \mu'(q_i) P^k(q_i, A) \right| \\
=& \left| \sum_{q_i \in S} P^k(q_i, A)(\mu(q_i) - \mu'(q_i)) \right| \\
=& \Big| \sum_{q_i \in B} P^k(q_i, A) \, |\mu(q_i) - \mu'(q_i)| \\
& - \sum_{q_i \notin B} P^k(q_i, A) \, |\mu(q_i) - \mu'(q_i)| \Big| \\
=& \max\Big\{ \sum_{q_i \in B} P^k(q_i, A)|\mu(q_i) - \mu'(q_i)|, \\
& \sum_{q_i \notin B} P^k(q_i, A)|\mu(q_i) - \mu'(q_i)| \Big\} \\
& - \min\Big\{ \sum_{q_i \in B} P^k(q_i, A)|\mu(q_i) \\
& - \mu'(q_i)|, \sum_{q_i \notin B} P^k(q_i, A)|\mu(q_i) - \mu'(q_i)| \Big\} \\
=& \max\Big\{ \\
& \sum_{q_i \in B} (P^k(q_i, A \cap I) + P^k(q_i, A \cap \bar{I}))|\mu(q_i) - \mu'(q_i)|, \\
& \sum_{q_i \notin B} (P^k(q_i, A \cap I) + P^k(q_i, A \cap \bar{I}))|\mu(q_i) - \mu'(q_i)| \Big\} \\
& - \min\Big\{ \\
& \sum_{q_i \in B} (P^k(q_i, A \cap I) + P^k(q_i, A \cap \bar{I}))|\mu(q_i) - \mu'(q_i)|, \\
& \sum_{q_i \notin B} (P^k(q_i, A \cap I) + P^k(q_i, A \cap \bar{I}))|\mu(q_i) - \mu'(q_i)| \Big\} \\
\leq& \max\Big\{ \sum_{q_i \in B} (1 + \epsilon \cdot |A \cap \bar{I}|)|\mu(q_i) - \mu'(q_i)|, \\
& \sum_{q_i \notin B} (1 + \epsilon \cdot |A \cap \bar{I}|)|\mu(q_i) - \mu'(q_i)| \Big\} \\
& - \min\Big\{ \sum_{q_i \in B} b|\mu(q_i) - \mu'(q_i)|, \sum_{q_i \notin B} b|\mu(q_i) - \mu'(q_i)| \Big\} \\
& \text{(By proposition A.1 such } \epsilon > 0 \text{ exist.)} \\
=& (1 + \epsilon \cdot |A \cap \bar{I}|) \cdot \max\Big\{ \sum_{q_i \in B} |\mu(q_i) - \mu'(q_i)|, \\
& \sum_{q_i \notin B} |\mu(q_i) - \mu'(q_i)| \Big\} \\
& - b \cdot \min\Big\{ \sum_{q_i \in B} |\mu(q_i) - \mu'(q_i)|, \\
& \sum_{q_i \notin B} |\mu(q_i) - \mu'(q_i)| \Big\} \\
=& (1 + \epsilon \cdot |A \cap \bar{I}|) d(\mu, \mu') - b \cdot d(\mu, \mu') \\
& \text{Since } d(\mu, \mu') = \sum_{q_i \in B} |\mu(q_i) - \mu'(q_i)| \\
& = \sum_{q_i \notin B} |\mu(q_i) - \mu'(q_i)| \\
=& (1 - b + \epsilon \cdot |A \cap \bar{I}|) d(\mu, \mu')
\end{aligned}$$

Since the matrix P is regular, we can choose k large enough such that $\epsilon \cdot |A \cap \bar{I}| < b$ and thus there exist k such that the mapping is a contraction.

Case 2: $(A \cap I = \emptyset)$

$$\begin{aligned}
&|\mu \cdot P^k(A) - \mu' \cdot P^k(A)| \\
=& \left| \sum_{q_i \in S} \mu(q_i) \cdot P^k(q_i, A) - \sum_{q_i \in S} \mu'(q_i) P^k(q_i, A) \right| \\
=& \left| \sum_{q_i \in S} P^k(q_i, A)(\mu(q_i) - \mu'(q_i)) \right| \\
\leq& \left| \sum_{q_i \in S} \epsilon |\mu(q_i) - \mu'(q_i)| \right| \\
& \text{(By proposition A.1 such } \epsilon > 0 \text{ exist) .} \\
=& 2\epsilon \cdot d(\mu, \mu')
\end{aligned}$$

Since the matrix P is regular, we can choose k large enough such that $2\epsilon < 1$ and thus there exist k such that the mapping is a contraction. ∎