Using Stochastic Comparison for Efficient Model Checking of Uncertain Markov Chains

Serge Haddad
LSV, ENS de Cachan
61 Av. du Président Wilson 94235 Cachan, France
Email: Serge.Haddad@lsv.ens-cachan.fr

Nihal Pekergin
LACL, Université Paris Est
61 Av. du Général de Gaulle 94010 Créteil, France
Email: nihal.pekergin@univ-paris12.fr

***Abstract*—We consider model checking of Discrete Time Markov Chains (DTMC) with transition probabilities which are not exactly known but lie in a given interval. Model checking a Probabilistic Computation Tree Logic (PCTL) formula for interval-valued DTMCs (IMC) has been shown to be NP hard and co-NP hard. Since the state space of a realistic DTMC is generally huge, these lower bounds prevent the application of exact algorithms for such models. Therefore we propose to apply the stochastic comparison method to check an extended version of PCTL for IMCs. More precisely, we first design linear time algorithms to quantitatively analyze IMCs. Then we develop an efficient, semi-decidable PCTL model checking procedure for IMCs. Furthermore, our procedure returns more refined answers than traditional ones: *YES, NO, DON'T KNOW*. Thus we may provide useful partial information for modelers in the 'DON'T KNOW' case.**

I. Introduction

Specification. Markovian models have been largely used in performance, dependability and reliability analysis of computer and communication systems. Modeling a quantitative stochastic system by a DTMC requires the specification of an initial distribution and a transition probability matrix. However, sometimes it may be impossible or unrealistic to determine precisely these probabilities. First of all, transition rates are estimated through statistical experiences which provide intervals of values (bounds) but not exact values. Moreover, these models are the abstraction of complex interactions or dependence of system parameters thus interval values for transition probabilities would be more appropriate than precise ones. Furthermore substituting a reduced Markov chain for the original one naturally leads to interval values. Interval-valued Markov chains for which transition probabilities are supposed to lie within a range of values have been introduced to capture such uncertainties [18].

Verification. In quantitative model checking, one first specifies a complex performability or safety guarantee by a temporal logic formula and then check its satisfiability based on the transition graph. Different languages have been proposed depending on the considered stochastic model and properties. In DTMCs, PCTL [15] has been first proposed and then extended with more complex operators (e.g. [7]). In CTMCs, a similar logic, Continuous Stochastic Logic [3] and its extensions [2], [10] also adapt CTL logic (defined for discrete-event systems). PCTL is also adequate for Markovian Decision Processes [5].

Model checking of uncertain models. In order to specify expected behavior, [16] introduces IMCs and check whether an IMC conforms to the model of the system w.r.t. different relations. In [18], the authors show how to obtain parameters of an IMC. Model checking of IMCs has been investigated first in [23] where it is shown that PCTL model checking of IMCs is in PSPACE and it is NP hard and co-NP-hard. The resultats have been generalized to ω-PCTL logic in [7]. Both [23], [7] also study interval-valued Markov decision processes.

Our contribution. In this paper, we provide an efficient semi-decision model checking of PCTL formulas over interval-valued DTMCs (IMCs) based on stochastic comparisons. We first design (or improve) linear time algorithms to quantitatively analyze IMCs.

- For any fixed state s and every subset of states $\mathcal{S}'$, we compute the minimal and maximal cumulative probability to go from s to $\mathcal{S}'$ in one step.
- [13] establish the existence of the greatest $\leq_{st}$ monotone and lower bounding transition probability matrix for a set of substochastic Markov chains specified by an IMC. Furthermore they design a linear time algorithm to build such a chain and state a structural characterization w.r.t. the IMC of finiteness of mean sojourn time in this chain. We refine this last result by building the subset of (initial) states for which this sojourn time starting from them is finite.

The model checking procedure is performed as usual by a bottom-up evaluation of subformulas and corresponding labelling of states. However due to the fact that we rely on a semi-decision procedure, the labels of states are 6-valued: *satisfied for all* ($\forall^+$), *satisfied for none* ($\forall^-$), *exists satisfied and exists not satisfied* ($\exists^{+-}$), *exists satisfied* ($\exists^+$), *exists not satisfied* ($\exists^-$), *don't know* (?). The three first cases are exact answers while the last three ones are partial ones.

Contrary to [23], [7] our variant of PCTL includes the mean reachability time operator ($\mathcal{D}$). This operator is useful for performability studies e.g. the mean time to failure can be expressed with this operator [24]. Moreover the stochastic comparison approach is the key point to handle both this operator and the (time bounded) until operator.

Organization. The remaining of the paper is organized as follows. Section II is devoted to IMC (definitions and algo-

rithms). We present the syntax and semantic of the considered PCTL in section III. Model checking of PCTL is developed in section IV. Related work is discussed in section V. Finally we conclude and give some perspectives for this approach in section V.

II. Interval Markov Chains

A. Definitions

Definition 1: A labelled time-homogeneous DTMC $\mathcal{M}$ is a 3-tuple $(\mathcal{S}, \mathbf{P}, L)$ where $\mathcal{S}$ is a finite set of *states*, $\mathbf{P}$ is the transition probability matrix, and $L : \mathcal{S} \to 2^{AP}$ is the *labelling* function which assigns to each state $s \in \mathcal{S}$, the set $L(s)$ of atomic propositions $a \in AP$ that are valid in s, AP denotes the finite set of atomic propositions.

Following (with slight changes) the definition in [23], we define the interval-valued, labelled DTMCs as follows :

Definition 2: A labelled interval-valued time homogeneous DTMCs $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ is defined by a 4-tuple $(\mathcal{S}, \mathbf{P}^-, \mathbf{P}^+, L)$, where $\mathcal{S}$ and L are defined as in a labelled DTMC. The interval-valued DTMC is defined by $\mathbf{P}^-$ (resp. $\mathbf{P}^+$) which is a substochastic matrix, ie. the row sums may be less or equal to 1 (resp. superstochastic matrix, ie. the row sums may be greater or equal to 1).

For all $s, t \in \mathcal{S}$, the following inequalities are satisfied between $\mathbf{P}^-$ and $\mathbf{P}^+$:

$$0 \leq \mathbf{P}^-[s,t] \leq \mathbf{P}^+[s,t] \wedge \sum_{t' \in \mathcal{S}} \mathbf{P}^+[s,t'] \geq 1 \geq \sum_{t' \in \mathcal{S}} \mathbf{P}^-[s,t'] \quad (1)$$

$$\mathbf{P}^-[s,t] \geq 1 - \sum_{t' \neq t} \mathbf{P}^+[s,t'] \quad (2)$$

$$\mathbf{P}^+[s,t] \leq 1 - \sum_{t' \neq t} \mathbf{P}^-[s,t'] \quad (3)$$

A DTMC with transition probability matrix $\mathbf{P}$ is said to belong to $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ (denoted $\mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$), if

$$\forall s, t, \quad \mathbf{P}^-[s,t] \leq \mathbf{P}[s,t] \leq \mathbf{P}^+[s,t] \quad (4)$$

Remark. Observe that the set of inequalities (1) is a necessary and sufficient condition for $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ to be non empty. Furthermore one can always change $\mathbf{P}^-$ and $\mathbf{P}^+$ in order to satisfy inequalities (2) and (3) without modifying $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$. Transformations consist in $\mathbf{P}^-[s,t] := \max(\mathbf{P}^-[s,t], 1 - \sum_{t' \neq t} \mathbf{P}^+[s,t'])$ and $\mathbf{P}^+[s,t] := \min(\mathbf{P}^+[s,t], 1 - \sum_{t' \neq t} \mathbf{P}^-[s,t'])$.

In the sequel, we denote by $\mathcal{M}$ (resp. $\mathbf{P}$) a DTMC (resp. a transition probability matrix) belonging to the set of Markov chains $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$.

Verifying PCTL formulas requires to transform the Markov chain and in particular to build sub-stochastic chains. Thus we simultaneously deal with stochastic and sub-stochastic matrices. A sub-stochastic $n \times n$ matrix $\mathbf{P}$ can also be considered as a $(n+1) \times (n+1)$ stochastic matrix by adding an additional absorbing state s such that $\mathbf{P}[s,s] = 1$ and $\forall s' \neq s, \mathbf{P}[s',s] = 1 - \sum_{s'' \neq s} \mathbf{P}[s',s'']$. In the sequel we interchangeably use these two representations.

So we introduce a set of sub-stochastic matrices in order to mainly study the strict sub-stochasticity, i.e. with probability 1, the additional absorbing state is reached. This explains the asymmetry of the next definition regarding to definition 2.

Definition 3: An interval valued $n \times n$ sub-stochastic matrix $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+, \mathbf{out})$ is defined by a $n \times n$ sub-stochastic matrix $\mathbf{P}^-$, a $n \times n$ positive matrix $\mathbf{P}^+$, and a positive vector of size n $\mathbf{out}$ that fulfill:

$$0 \leq \mathbf{P}^-[s,t] \leq \mathbf{P}^+[s,t] \wedge \sum_{t' \in \mathcal{S}} \mathbf{P}^-[s,t'] + \mathbf{out}[s] \leq 1$$

$$\mathbf{P}^+[s,t] \leq 1 - \sum_{t' \neq t} \mathbf{P}^-[s,t'] - \mathbf{out}[s]$$

A sub-stochastic $n \times n$ matrix $\mathbf{P}$ is said to belong to $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+, \mathbf{out})$ (denoted $\mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+, \mathbf{out})$), if $\forall s, t$:

$$\mathbf{P}^-[s,t] \leq \mathbf{P}[s,t] \leq \mathbf{P}^+[s,t] \wedge \sum_{t \in \mathcal{S}} \mathbf{P}[s,t] \leq 1 - \mathbf{out}[s] \quad (5)$$

B. Structural Properties of IMC

Given an IMC, a state s and a subset of states $\mathcal{S}'$, we first characterize the maximal and minimal value to reach from s in one step $\mathcal{S}'$.

Lemma 1: Let $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ be a labelled interval-valued time homogeneous DTMC, $\mathcal{S}' \subseteq \mathcal{S}$ be a subset of states and $s \in \mathcal{S}$ be a state. Then:

- $\min(\sum_{t \in \mathcal{S}'} \mathbf{P}[s,t] \mid \mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)\}$
  $= \max(\sum_{t \in \mathcal{S}'} \mathbf{P}^-[s,t], 1 - \sum_{t \notin \mathcal{S}'} \mathbf{P}^+[s,t])$
- $\max\{\sum_{t \in \mathcal{S}'} \mathbf{P}[s,t] \mid \mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+))$
  $= \min(\sum_{t \in \mathcal{S}'} \mathbf{P}^+[s,t], 1 - \sum_{t \notin \mathcal{S}'} \mathbf{P}^-[s,t])$

Proof. We only prove the first assertion since the proof of the second one is similar. Let us note $m_s \equiv \min(\sum_{t \in \mathcal{S}'} \mathbf{P}[s,t] \mid \mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+))$. It follows from Equation (4) that any $\mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ fulfills for any subset $\mathcal{S}' \in \mathcal{S}$ $\sum_{t \in \mathcal{S}'} \mathbf{P}[s,t] \geq \sum_{t \in \mathcal{S}'} \mathbf{P}^-[s,t]$. Moreover, since $\mathbf{P}$ is stochastic $\sum_{t \in \mathcal{S}'} \mathbf{P}[s,t] \geq 1 - \sum_{t \notin \mathcal{S}'} \mathbf{P}^+[s,t]$. Thus $m_s \geq \max(\sum_{t \in \mathcal{S}'} \mathbf{P}^-[s,t], 1 - \sum_{t \notin \mathcal{S}'} \mathbf{P}^+[s,t])$. In order to prove equality, we exhibit some $\mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ that reaches this value. We observe that we only have to specify $\mathbf{P}[s, *]$. We order the states of $\mathcal{S}$ such that any state of $\mathcal{S}'$ occurs before the states out of $\mathcal{S}'$. We fill row s to minimize the sum of probabilities for a given set (the first case) with Algorithm 1.

Algorithm 1: Filling algorithm to minimize

Input : $\mathbf{P}^-, \mathbf{P}^+, \mathcal{S} = \{s_1, s_2, \cdots s_n\}$;
Output : row s of $\mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$;
$sum = 0$;
for $i = 1$ **to** n **do**
  1.
  $\mathbf{P}[s, s_i] = \max(\mathbf{P}^-[s, s_i], 1 - sum - \sum_{j>i} \mathbf{P}^+[s, s_j])$;
  2. $sum = sum + \mathbf{P}[s, s_i]$;
end

We prove by induction that at the beginning of each iteration i (including the case $i = n + 1$ meaning that the algorithm exits the loop), the following equations are satisfied:

$$sum = \sum_{j<i} \mathbf{P}[s, s_j]$$

$$\forall j < i, \;\; \mathbf{P}^-[s, s_j] \leq \mathbf{P}[s, s_j] \leq \mathbf{P}^+[s, s_j]$$

$$\sum_{j<i} \mathbf{P}[s, s_j] = \max(\sum_{j<i} \mathbf{P}^-[s, s_j], 1 - \sum_{j \geq i} \mathbf{P}^+[s, s_j])$$

The basis case $i = 1$ is straightforward except for the last assertion which follows from $\sum_{j \geq 1} \mathbf{P}^+[s, s_j] \geq 1$. Assume that the above inequalities are satisfied for $i - 1$. Instruction 2 and the inductive hypothesis ensure that $sum = \sum_{j<i} \mathbf{P}[s, s_j] + \mathbf{P}[s, s_i] = \sum_{j<i+1} \mathbf{P}[s, s_j]$. Instruction 1 ensures that $\mathbf{P}[s, s_i] \geq \mathbf{P}^-[s, s_i]$. Furthermore before instruction 1,
$1 - sum - \sum_{j>i} \mathbf{P}^+[s, s_j]$
$= 1 - \sum_{j<i} \mathbf{P}[s, s_j] - \sum_{j>i} \mathbf{P}^+[s, s_j]$
$\leq 1 - (1 - \sum_{i \geq j} \mathbf{P}^+[s, s_j]) - \sum_{j>i} \mathbf{P}^+[s, s_j]$
$= \mathbf{P}^+[s, s_i]$.
Thus after instruction 1, $\mathbf{P}[s, s_i] \leq \mathbf{P}^+[s, s_i]$ (we also use the inequality $\mathbf{P}^-[s, s_i] \leq \mathbf{P}^+[s, s_i]$).

In order to establish the last inequality, we perform a case study.
Case 1: $\sum_{j \leq i} \mathbf{P}^-[s, s_j] \geq 1 - \sum_{j \geq i} \mathbf{P}^+[s, s_j]$
$\sum_{j<i} \mathbf{P}^-[s, s_j] + \mathbf{P}^-[s, s_i] \geq 1 - \sum_{j \geq i} \mathbf{P}^+[s, s_j] + \mathbf{P}^+[s, s_i]$
Hence before instruction 1,
$sum + \mathbf{P}^-[s, s_i] \geq 1 - \sum_{j>i} \mathbf{P}^+[s, s_j]$
$\mathbf{P}^-[s, s_i] \geq 1 - sum - \sum_{j>i} \mathbf{P}^+[s, s_j]$
then after instruction 1, $\mathbf{P}[s, s_i] = \mathbf{P}^-[s, s_i]$
and after instruction 2, $sum = \sum_{j \leq i} \mathbf{P}^-[s, s_j]$.
Case 2: $\sum_{j<i} \mathbf{P}^-[s, s_j] < 1 - \sum_{j \geq i} \mathbf{P}^+[s, s_j]$
Using the first and the last inductive assertions, we deduce that $sum = 1 - \sum_{j \geq i} \mathbf{P}^+[s, s_i]$.
Hence before instruction 1,
$1 - sum - \sum_{j>i} \mathbf{P}^+[s, s_j] = \mathbf{P}^+[s, s_i] \geq \mathbf{P}^-[s, s_i]$
Then after instruction 1,
$\mathbf{P}[s, s_i] = \mathbf{P}^+[s, s_i]$
and after instruction 2, $sum = 1 - \sum_{j>i} \mathbf{P}^+[s, s_j]$
Moreover $\sum_{j \leq i} \mathbf{P}^-[s, s_j] \leq \sum_{j<i} \mathbf{P}^-[s, s_j] + \mathbf{P}^+[s, s_i]$
$< (1 - \sum_{j \geq i} \mathbf{P}^+[s, s_j]) + \mathbf{P}^+[s, s_i]$
$= 1 - \sum_{j>i} \mathbf{P}^+[s, s_j]$.
Case 3: $\sum_{j<i} \mathbf{P}^-[s, s_j] \geq 1 - \sum_{j \geq i} \mathbf{P}^+[s, s_j]$
$\wedge \sum_{j \leq i} \mathbf{P}^-[s, s_j] < 1 - \sum_{j>i} \mathbf{P}^+[s, s_j]$
Thus before instruction 1, $sum = \sum_{j<i} \mathbf{P}^-[s, s_j]$.
$1 - sum - \sum_{j>i} \mathbf{P}^+[s, s_j] =$
$1 - \sum_{j<i} \mathbf{P}^-[s, s_j] - \sum_{j>i} \mathbf{P}^+[s, s_j]$
$> -\sum_{j<i} \mathbf{P}^-[s, s_j] + \sum_{j \leq i} \mathbf{P}^-[s, s_j] = \mathbf{P}^-[s, s_i]$.
Here we have used the second hypothesis of Case 3.
Therefore after instruction 1,
$\mathbf{P}[s, s_i] = 1 - sum - \sum_{j>i} \mathbf{P}^+[s, s_j]$
$\leq 1 - (1 - \sum_{j \geq i} \mathbf{P}^+[s, s_j]) - \sum_{j>i} \mathbf{P}^+[s, s_j] = \mathbf{P}^+[s, s_i]$
The inequality follows from the first hypothesis of Case 3.
After instruction 2,
$sum =$
$\sum_{j<i} \mathbf{P}^-[s, s_j] + 1 - \sum_{j<i} \mathbf{P}^-[s, s_j] - \sum_{j>i} \mathbf{P}^-[s, s_j]$
$= 1 - \sum_{j>i} \mathbf{P}^+[s, s_j]$.
Using the third inductive assertion with $i = n + 1$ and inequation 1, we obtain $\sum_{j<n+1} \mathbf{P}[s, s_i] = 1$. Using again the third inductive assertion with i the index of the first state not in $\mathcal{S}'$, we obtain $\sum_{t \in \mathcal{S}'} \mathbf{P}[s, t] = \max(\sum_{t \in \mathcal{S}'} \mathbf{P}^-[s, t], 1 - \sum_{t \notin \mathcal{S}'} \mathbf{P}^+[s, t])$. □

In the second case to maximize the sum of probabilities, we fill row s by Algorithm 2. Let us remark here that if one is interested in minimizing or maximizing of a partial sum of probabilities over a subset of states $\mathcal{S}' \in \mathcal{S}$, it would be sufficient to perform the loop of these algorithms only for this subset since they occur first in the enumeration of states.

Algorithm 2: Filling algorithm to maximize

Input : $\mathbf{P}^-, \mathbf{P}^+, \mathcal{S} = \{s_1, s_2, \cdots s_n\}$;
Output : row s of $\mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$;
$sum = 0$;
for $i = 1$ **to** n **do**
  $\mathbf{P}[s, s_i] = \min(\mathbf{P}^+[s, s_i], 1 - sum - \sum_{j>i} \mathbf{P}^-[s, s_j])$;
  $sum = sum + \mathbf{P}[s, s_i]$;
end

C. Algorithms for Stochastic Bounds

In this subsection, we present algorithms to construct bounding matrices in the sense of $\leq_{st}$ ordering for a given IMC. We first give the basic definitions and theorems for stochastic comparison and we refer to [20] for further informations.

1) Stochastic Comparison: The following is the generic definition for the $\leq_{st}$ ordering which is known also as strong ordering or sample-path ordering.

Definition 4: Let X and Y be two random variables taking values on a totally ordered space $\mathcal{S}$,

$$X \leq_{st} Y \iff \mathtt{E}f(X) \leq \mathtt{E}f(Y)$$

for all increasing functions $f : \mathcal{S} \to \mathcal{R}$ whenever expectations exist.
In the case of finite state space that we will consider in the sequel, the comparison of random variables are defined through following probability inequalities.

Property 1: Let X and Y be two random variables taking values on $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$, and $p = [p_1 \ldots p_i \ldots p_n]$, $q = [q_1 \ldots q_i \ldots q_n]$ be probability vectors respectively denoting distributions of X and Y ($p_i = Prob(X = s_i)$, and $q_i = Prob(Y = s_i)$).

$$1 \leq i \leq n, \qquad X \leq_{st} Y \Leftrightarrow \sum_{k=1}^{i} p_k \geq \sum_{k=1}^{i} q_k \qquad (6)$$

The above inequalities are often given by beginning from the last state. However Eq. (6) is straightforwardly generalizable for sub-stochastic vectors and thus is more appropriate for our goals.

Let us notice here that in the sequel we interchangeably use $X \leq_{st} Y$ and $p \leq_{st} q$. We apply the following definition to compare Markov chains.

Definition 5: Let $\{X(t_i)\}_{i\geq 0}$ (resp. $\{Y(t_i)\}_{i\geq 0}$) be a DTMC. We say $\{X(t_i)\}_{i\geq 0} \leq_{st} \{Y(t_i)\}_{i\geq 0}$, if $\forall i,\ X(t_i) \leq_{st} Y(t_i)$.

Intuitively, this means that the probability to reach states having a higher number than a fixed one is greater or equal in Y at every instant n. The following folk theorem provides sufficient conditions to establish the comparison of DTMCs that will be used in the sequel.

Theorem 1: Let $\mathbf{P}$ (resp. $\mathbf{P}'$) be the probability transition matrix of the time-homogeneous Markov chain $\{X(t_i)\}_{i\geq 0}$ (resp. $\{Y(t_i), i \geq 0\}$). The comparison of Markov chains is established ($\{X(t_i)\}_{i\geq 0} \leq_{st} \{Y(t_i)\}_{i\geq 0}$), if the following conditions are satisfied :

- $X(t_0) \leq_{st} Y(t_0)$,
- at least one of the probability transition matrices is monotone, that is, either $\mathbf{P}$ or $\mathbf{P}'$ (say $\mathbf{P}$) is $\leq_{st}$ monotone, if for all probability vectors p and q,

$$p \leq_{st} q \implies p\mathbf{P} \leq_{st} q\mathbf{P}$$

which is equivalent to

$$1 \leq i \leq n-1, \qquad \mathbf{P}[s_i, *] \leq_{st} \mathbf{P}[s_{i+1}, *]$$

where $\mathbf{P}[s_i, *]$ denotes the row of matrix $\mathbf{P}$ for state s_i.
- the transition matrices are comparable in the sense of the $\leq_{st}$ order :

$$\mathbf{P} \leq_{st} \mathbf{P}' \iff 1 \leq i \leq n,\ \ \mathbf{P}[s_i, *] \leq_{st} \mathbf{P}'[s_i, *]$$

Algorithm 3: Construction of the greatest lower bounding matrix $\mathbf{P}^\bullet$

Input : $\mathbf{P}^-, \mathbf{P}^+ : n \times n$ matrices; $\mathbf{out}$: a vector of size n representing the minimal transition probabilities to reach the absorbing state;
Output : $\mathbf{P}^\bullet \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$;
$\mathbf{P}^\bullet \leq_{st} \forall \mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$;
for $i = 1$ **to** n **do**
  for $j = 1$ **to** n **do**
    $\mathbf{P}^a[s_i, s_j] = \min(\sum_{k=1}^{j} \mathbf{P}^+[s_i, s_k], 1 - \sum_{k=j+1}^{n} \mathbf{P}^-[s_i, s_k]) - \mathbf{out}[i])$;
    (Ih) **if** *($i \leq j$) and* ($\mathbf{P}^a[s_i, s_j] == 1$) **then** halt;
  end
  $\mathbf{P}^\bullet[s_i, 1] = \mathbf{P}^a[s_i, 1]$;
  for $j = 2$ **to** n **do**
    $\mathbf{P}^\bullet[s_i, s_j] = \mathbf{P}^a[s_i, s_j] - \mathbf{P}^a[s_i, s_{j-1}]$;
  end
end

2) Bounding Algorithms: We now present algorithms to construct bounding algorithms for IMC with one absorbing state proposed in [13], [14].

Algorithm 3 builds the greatest lower bounding matrix in the sense of $\leq_{st}$ ordering for matrices in the given interval with an additional information specified (minimal transition probabilities to reach the absorbing state). Thus it follows from Eq. 6 that $\forall \mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$:

$$1 \leq i \leq n,\ \ 1 \leq j \leq n, \quad \sum_{k=1}^{j} \mathbf{P}^\bullet[s_i, s_k] \geq \sum_{k=1}^{j} \mathbf{P}[s_i, s_k] \tag{7}$$

Given an input matrix, Algorithm 4 produces the greatest *monotone* lower bounding matrix in the sense of $\leq_{st}$ ordering. Moreover $\forall t, (\mathbf{P}^\star)^t$ is monotone and provides a lower bounding matrix for all transition probability matrices in the interval. Thus $\forall \mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$:

$$1 \leq i \leq n,\ \ 1 \leq j \leq n,\ \ \sum_{k=1}^{j} (\mathbf{P}^\star)^t[s_i, s_k] \geq \sum_{k=1}^{j} (\mathbf{P})^t[s_i, s_k] \tag{8}$$

These inequalities yields indeed the upper bounds to reach states $s_1 \cdots s_j$ in t steps beginning from state s_i, in the case $\mathbf{P}^\star$ is strictly substochastic.

Algorithm 4: Construction of monotone lower bounding matrix $\mathbf{P}^\star$

Input : $\mathbf{P}^\bullet$, see Algorithm 3;
Output : $\mathbf{P}^\star \leq_{st} \forall \mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$; $\mathbf{P}^\star$ is monotone;
$\mathbf{P}^\star[s_n, .] = \mathbf{P}^\bullet[s_n, .]$;
for $i = n-1$ **downto** 1 **do**
  $x = 0$;
  for $j = 1$ **to** n **do**
    $\mathbf{P}^\star[s_i, s_j] = \max(\sum_{k=1}^{j} \mathbf{P}^\bullet[s_i, s_k], \sum_{k=1}^{j} \mathbf{P}^\star[s_{i+1}, s_k]) - x$;
    $x = x + \mathbf{P}^\star[s_i, s_j]$;
  end
end

A fundamental issue related to a sub-stochastic matrix, $\mathbf{P}$, is the following one: which components of the vector given below are finite?

$$\sum_{t\geq 0} (\mathbf{P}^t)\mathbf{1}_n$$

where $\mathbf{1}_n$ is the unit vector (all entries are 1) of size n. This question can be solved in the general case by the construction of the strongly connected components of the underlying graph related to $\mathbf{P}$ and then by local summations related to this decomposition. In the particular case when the matrix is monotone a quick criterion whether the set of infinite values is empty has been established in [13].

Property 2: The following statements are equivalent:

- $\mathbf{P}^\star$ is strictly substochastic which is equivalent to the convergence of the series $\sum_{t\geq 1} (\mathbf{P}^\star)^t$,
- $\forall i\ \sum_{j\leq i} \mathbf{P}^\bullet[s_i, s_j] < 1$,

- $\forall i\ \sum_{j\leq i}\mathbf{P}^{+}[s_i,s_j]<1$ or $\sum_{j>i}\mathbf{P}^{-}[s_i,s_j]>0$,
- Condition (Ih) of Algorithm 3 is never satisfied.

Here we refine this criterion for monotone matrices by determining the subset of states with finite values by an efficient algorithm that simply parses once every entry.

Algorithm 5: Determination of states that reach the absorbing state with probability 1 in a monotone sub-stochastic matrix

Input : $\mathbf{P}^{\star}$, see Algorithm 4;
Output : set of states;
$reach = true$; $iprec = 0$;
for $i = 1$ **to** n **do**
  if *(reach) and* $(\sum_{j=1}^{i}\mathbf{P}^{\star}[s_i,s_j]==1)$ **then**
    $reach = false$; $iprec = i$;
  else if $(\sum_{j=1}^{i}\mathbf{P}^{\star}[s_i,s_j]==1)$ *or* $(\sum_{j=1}^{i-1}\mathbf{P}^{\star}[s_i,s_j]>0)$ **then**
    $iprec = i$;
  else $reach = true$;
end
return $\{iprec+1,\ldots,n\}$

Property 3: Given an input sub-stochastic $(n\times n)$ matrix $\mathbf{P}^{\star}$, Algorithm 5 returns the set of states that reach with probability 1 the additional absorbing state (indexed by $n+1$)

Proof. Assume that a state i fulfills $\sum_{j=1}^{i}\mathbf{P}^{\star}[s_i,s_j]==1$, then for all $i'\leq i$, $\sum_{j=1}^{i}\mathbf{P}^{\star}[s'_i,s_j]==1$ by monotonicity. Consequently the subchain reduced to states 1 to i is a Markov chain and there is a null probability to reach the absorbing state. There are two cases: there is no such i . Then $\forall i<n\ \sum_{j>i}\mathbf{P}[s_i,s_j]>0$ and $\sum_{j\geq 1}\mathbf{P}[s_n,s_j]<1$. This means that in the graph deduced this chain, there is a path from any state to the absorbing state. Thus with probability 1, each state reaches the absorbing state.

Otherwise, let us call $imax$ the greatest state that fulfills $\sum_{j=1}^{imax}\mathbf{P}^{\star}[s_{imax},s_j]==1$. After iteration $imax$, $reach$ becomes false and $iprec = imax$. We distinguish two cases:

- $reach$ is false at the end of the algorithm. So $iprec = n$ and $\forall i>imax\sum_{j=1}^{i-1}\mathbf{P}^{\star}[s_i,s_j]>0)$. The inequality means that in the graph there is an edge from i to a smaller state. And by induction, there is a path to the set $\{1,\ldots,imax\}$. There is a non null probability to never reach the absorbing state.
- $reach$ becomes true at iteration $jmin$ (and remains true until the end of the algorithm). By the same reasoning, the set of states $\{imax+1,\ldots,jmin-1\}$, there is a non null probability to never reach the absorbing state. By monotonocity , $\forall j'\geq jmin,\sum_{k=1}^{jmin-1}\mathbf{P}[s_j,s_k]=0$. This means that the states $\{jmin,n\}$ may be considered in isolation. By definition of $imax$, $\sum_{j=1}^{imax}\mathbf{P}^{\star}[s_{imax},s_j]==1$ is never satisfied. Thus with probability 1, these states reach the absorbing state. □

Remarks

- Algorithm 3 can be also applied for the cases without any absorbing state. In such a case **out** vector must be taken as 0.
- Algorithms 3 and 4 are given separately for the sake of the readability, however it is possible to build the greatest lower bounding, monotone matrix by parsing once every entry starting from the greatest row. Thus the worst-case complexity for a $n\times n$ matrices is $O(n^2)$.

III. PCTL

A. PCTL for MCs

We give here the syntax of PCTL close to [15] but extended by a duration operator (see [19]). Let α,β be integers, $p\in[0,1]$ be a probability, $r\in\mathbb{R}_{>0}$ be a positive real, a be an atomic proposition, and $\triangleleft$ be a comparison operator $\in\{\leq,\geq\}$. The syntax of PCTL is defined by:

$$\phi ::= true \mid a \mid \phi\wedge\phi \mid \neg\phi \mid \mathcal{P}_{\triangleleft p}(\mathcal{X}\phi) \mid \mathcal{P}_{\triangleleft p}(\phi_1\ \mathcal{U}^{[\alpha,\beta]}\phi_2) \mid \mathcal{D}_{\triangleleft r}(\phi)$$

The path formula $\mathcal{X}\phi$ asserts that the second state of the path satisfies the state formula ϕ. The path formula $\phi_1\ \mathcal{U}^{[\alpha,\beta]}\phi_2$ asserts that there exists an $i\in[\alpha,\beta]$ s.t. the ith state satisfies the state formula ϕ_2 while all preceeding states satisfy ϕ_1. $\mathcal{P}_{\triangleleft p}(\varphi)$ asserts that the probability measure π of random paths satisfying the path formula φ fulfills $\pi\triangleleft p$. $\mathcal{D}_{\triangleleft r}(\phi)$ asserts that the expected time ρ to reach a state satisfying ϕ fulfills $\rho\triangleleft r$. In the sequel we call it as *the mean reachability time operator*.

Let us present the formal semantics of these formulas. We denote $s\models\phi$, the satisfaction of a state formula ϕ by s and $S_\phi\equiv\{s\mid s\models\phi\}$ is the subset of states that satisfy ϕ. A path $\sigma\equiv s_0s_1\ldots$ is an infinite sequence of states of the Markov chain. We denote $\sigma\models\varphi$, the satisfaction of a path φ formula by σ.

$$\begin{array}{ll}\sigma\models\mathcal{X}\phi & \text{iff } s_1\models\phi\\ \sigma\models\phi_1\mathcal{U}^{[\alpha,\beta]}\phi_2 & \text{iff } \exists i\ \alpha\leq i\leq\beta\wedge s_i\models\phi_2\\ & \qquad\wedge\forall j<i\ s_j\models\phi_1\end{array}$$

Let ϕ be a state formula and σ be a sequence then $FTime(\sigma,\phi)\equiv\min\{i\mid s_i\models\phi\}$. Observe that if ϕ is never satisfied then $FTime(\sigma,\phi)=\infty$.

Let $\mathcal{M}$ be a Markov chain and φ be a path formula. Then $Prob^{\mathcal{M}}(s,\varphi)$ is the probability that a random path in $\mathcal{M}$ starting from s satisfies φ. $\mathtt{E}$ denotes the expectation operator. $\sigma^{\mathcal{M}}(s)$ is a random path in $\mathcal{M}$ starting from s (i.e. a random variable).

$$\begin{array}{ll}s\models true & \text{for all } s\in\mathcal{S}\\ s\models a & \text{iff } a\in L(s)\\ s\models\neg\phi & \text{iff } s\not\models\phi\\ s\models\phi_1\wedge\phi_2 & \text{iff } s\models\phi_1\wedge s\models\phi_2\\ s\models\mathcal{P}_{\triangleleft p}(\mathcal{X}\phi) & \text{iff } Prob^{\mathcal{M}}(s,\mathcal{X}\phi)\triangleleft p\\ s\models\mathcal{P}_{\triangleleft p}(\phi_1\ \mathcal{U}^{[\alpha,\beta]}\phi_2) & \text{iff } Prob^{\mathcal{M}}(s,\phi_1\mathcal{U}^{[\alpha,\beta]}\phi_2)\triangleleft p\\ s\models\mathcal{D}_{\triangleleft r}(\phi) & \text{iff } \mathtt{E}(FTime(\sigma^{\mathcal{M}}(s),\phi))\triangleleft r\end{array}$$

B. PCTL for IMCs

As an IMC is a set of Markov chains, different semantics are possible. In [23], the authors propose a "boolean" universal

semantics, i.e.:

$\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+), s \models \phi$ iff $\forall \mathcal{M} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)\ \mathcal{M}, s \models \phi$

Combining the universal satisfiability of ϕ and the one of $\neg\phi$, one obtains three cases:

1) $\forall \mathcal{M} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+),\ \mathcal{M}, s \models \phi$
2) $\forall \mathcal{M} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+),\ \mathcal{M}, s \models \neg\phi$
3) $\exists \mathcal{M}, \mathcal{M}' \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+),\ \mathcal{M}, s \models \phi \wedge \mathcal{M}', s \models \neg\phi$

However if we apply a semi-decision procedure for the model checking, the number of cases increases. Elaborating this idea, this leads to six semi decision cases described below. In order to concisely represent them we denote the result of checking ϕ on a state s, $s.\phi \in \{\forall^+, \forall^-, \exists^{+-}, \exists^+, \exists^-, ?\}$.

- $s.\phi = \forall^+$ ensures that (s, ϕ) belongs to case 1.
- $s.\phi = \forall^-$ ensures that (s, ϕ) belongs to case 2.
- $s.\phi = \exists^{+-}$ ensures that (s, ϕ) belongs to case 3.
- $s.\phi = \exists^+$ ensures that (s, ϕ) belongs to cases 1 or 3.
- $s.\phi = \exists^-$ ensures that (s, ϕ) belongs to cases 2 or 3.
- $s.\phi =?$ all cases are possible.

The three first answers fully characterize the situation while the next two ones partially characterize it and the last one provides no conclusion.

IV. Model Checking PCTL

Given an interval valued DTMC $\mathcal{M}$ and a PCTL formula ϕ, the verification algorithm proceeds by a bottom-up evaluation of sub-formulae of ϕ in the syntactic tree of the formula ϕ. From leaves to the root, each state is labelled with an assignement of a value to the sub-formula. Hence, every step of the algorithm evaluates a formula viewing the operands of the most external operator as values assigned by the previous evaluations. Let in the sequel, ψ, ψ_1, ψ_2 denote an already evaluated state formula. This leads us to study each operator. In the sequel, the assignment of the state by label $\exists^{+-}$ is implicit and corresponds to cases where the state is successively labelled with both $\exists^+$ and $\exists^-$.

$\boxed{\phi = \neg\psi}$ The algorithm labels a state s with $s.\phi = \forall^+$ (resp. $s.\phi = \forall^-$, $s.\phi = \exists^{+-}$, $s.\phi = \exists^+$, $s.\phi = \exists^-$, $s.\phi =?$) if it is labelled with $s.\psi = \forall^-$ (resp. $s.\psi = \forall^+$, $s.\psi = \exists^{+-}$,$s.\psi = \exists^-$, $s.\psi = \exists^+$, $s.\psi =?$).

$\boxed{\phi = \psi_1 \wedge \psi_2}$ The algorithm labels a state s depending on values $s.\psi_1, s.\psi_2$ as presented in the table below. For instance, when $s.\psi_1 = \exists^+$ and $s.\psi_2 = \exists^-$, we know that there is a model such that s does not fulfill ψ_2. So this model does not fulfill ϕ and this is the only information that can be deduced thus leading to $s.\phi = \exists^-$.

$s.\psi_1 \setminus s.\psi_2$	$\forall^+$	$\forall^-$	$\exists^{+-}$	$\exists^+$	$\exists^-$	$?$
$\forall^+$	$\forall^+$	$\forall^-$	$\exists^{+-}$	$\exists^+$	$\exists^-$	$?$
$\forall^-$	$\forall^-$	$\forall^-$	$\forall^-$	$\forall^-$	$\forall^-$	$\forall^-$
$\exists^{+-}$	$\exists^{+-}$	$\forall^-$	$\exists^-$	$\exists^-$	$\exists^-$	$\exists^-$
$\exists^+$	$\exists^+$	$\forall^-$	$\exists^-$	$?$	$\exists^-$	$?$
$\exists^-$	$\exists^-$	$\forall^-$	$\exists^-$	$\exists^-$	$\exists^-$	$\exists^-$
$?$	$?$	$\forall^-$	$\exists^-$	$?$	$\exists^-$	$?$

$\boxed{\phi = \mathcal{P}_{\lhd p}(\mathcal{X}\psi)}$ We handle the case $\lhd = \leq$. The other case is omitted as it is similar.

We can label $s.\phi = \forall^+$, by considering the upper bounding case: if the one-step transition probability from state s remains less than p with maximal transition probabilities to states where ϕ is possibly satisfied, then condition 1 of the satisfability is ensured. We label a state s with $s.\phi = \forall^+$ if

$$\min(\sum_{s'.\phi \neq \forall^-} \mathbf{P}^+[s, s'], 1 - \sum_{s'.\phi = \forall^-} \mathbf{P}^-[s, s']) \leq p$$

We can label $s.\phi = \forall^-$, by considering the lower bounding case: if the one-step transition probability from state s exceeds p with minimal transition probabilities to states where ϕ is surely satisfied, then condition 2 of the satisfability is ensured. We label a state s with $s.\phi = \forall^-$ if

$$\max(\sum_{s'.\phi = \forall^+} \mathbf{P}^-[s, s'], 1 - \sum_{s'.\phi \neq \forall^+} \mathbf{P}^+[s, s']) > p$$

For all states not yet labelled, we compute two reals m_s and M_s by means of the filling algorithms given in the previous section. To compute m_s, we first apply Algorithm 1 with input parameter $\mathcal{S}' = \{s' \mid s'.\phi \neq \forall^-\}$, to determine the output parameter $\mathbf{P}[s, *]$ with $s \in \mathcal{S}$ and then compute $m_s = \sum_{s' \in \mathcal{S}'} \mathbf{P}[s, s']$. Similarly, M_s is computed from Algorithm 2 with input parameter $\mathcal{S}' = \{s' \mid s'.\phi = \forall^+\}$ and M_s is computed from the obtained vector for the set $\mathcal{S}' = \{s' \mid s'.\phi = \forall^+\}$: $M_s = \sum_{s' \in \mathcal{S}'} \mathbf{P}[s, s'])$. Then the not yet labelled states are labelled as follows: If $m_s \leq p \wedge M_s > p$ then $s.\phi = \exists^{+-}$ else if $m_s \leq p$ then $s.\phi = \exists^+$ else if $M_s > p$ then $s.\phi = \exists^-$ else $s.\phi =?$.

$\boxed{\phi = \mathcal{P}_{\lhd p}(\psi_1\ \mathcal{U}^{[\alpha,\beta]}\psi_2)}$

Principle. Once ψ_1 and ψ_2 have been evaluated, the standard method consists to eliminate states fulfilling $\neg(\psi_1 \vee \psi_2)$ to merge states fulfilling ψ_2 in an absorbing state and to study the behaviour of the transformed substochastic chain *without* the absorbing state during the interval $[0, \alpha - 1]$ and the behaviour of the transformed substochastic chain *with* the absorbing state during the interval $[\alpha, \beta]$. In the framework of IMC, the probability to stay in a subset of states and then reach some absorbing state can be lower bounded using $\mathbf{P}^-$ (see below case 4). However using $\mathbf{P}^+$ for the upper bound does not provide accurate results (since for instance the pointwise upper bounding often transforms a substochastic matrix in a superstochastic one!) and this is where stochastic comparison takes place (see case 3). Other cases are simpler.

As in the case of the former operator, we consider here the case $\lhd = \leq$ and there are 6 possible answers that we can assign to a state.

1. In the case $\alpha = 0$, some immediate conclusions are possible from the label of s.
   - if $s.\psi_2 = \forall^+$, formula ϕ is satisfied with probability 1, thus these states are labelled with $s.\phi = \forall^-$.
   - if $s.\psi_1 = \forall^-$, somme immediate conclusions depending on the label for $s.\psi_2$ are possible:

* $s.\psi_2 = \forall^- \implies s.\phi = \forall^+$. Formula ϕ is not satisfied with probability 1, thus these states are labelled with $s.\phi = \forall^+$.
* $s.\psi_2 = \exists^- \implies s.\phi = \exists^+$. Formula ϕ is not satisfied with probability 1 for some chains of the interval, thus these states are labelled with $s.\phi = \exists^+$.
* $s.\psi_2 = \exists^+ \implies s.\phi = \exists^-$. Formula ϕ is satisfied for some chains of the interval with probability 1 thus these states are labelled with $s.\phi = \exists^-$.
* $s.\psi_2 = \exists^{+-} \implies s.\phi = \exists^{+-}$. Formula ϕ is satisfied for some chains and it is not satisfied for other chains of the interval.

2. In the case $s.\psi_1 = \forall^-$, and $\alpha > 0$, formula ϕ is satisfied with probability 0. Thus these states are labelled with $s.\phi = \forall^+$.
3. We now see if we can label with $\forall^+$. We define two sets: $\mathcal{S}_1 = \{s \mid s.\psi_2 = \forall^- \wedge s.\psi_1 \neq \forall^-\}$, $\mathcal{S}_2 = \{s \mid s.\psi_2 \neq \forall^-\}$. The states $\mathcal{S} - \mathcal{S}_1$ are made absorbing. We consider the upper bounding case to reach absorbing states $\mathcal{S}_2$ from $\mathcal{S}_1$ states. First we reorder states of $\mathcal{S}_1$with respect to the maximal transition probabilities to $\mathcal{S}_2$ states (the matrices are reordered by row and column permutations). Then we build lower bounding matrix restricted to states $\mathcal{S}_1$, and make it monotone by means of algorithms given in section II-C.

3.1. Construct a column vector, $\mathbf{r}^+$ of size n for maximal transition probabilities to the absorbing $\mathcal{S}_2$ states from $\mathcal{S}_1$ states. The maximal transition probability from a state $s_i \in \mathcal{S}_1$ to $\mathcal{S}_2$ for all possible Markov chains in the interval $\in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ is defined by $\mathbf{r}^+[s_i]$:

$$\mathbf{r}^+[s_i] = \min(\sum_{s_k \in \mathcal{S}_2} \mathbf{P}^+[s_i, s_k], 1 - \sum_{s_k \notin \mathcal{S}_2} \mathbf{P}^-[s_i, s_k]) \tag{9}$$

We reorder this vector in the decreasing order $(\mathbf{r}^+[s_1] \geq \mathbf{r}^+[s_2] \cdots \geq \mathbf{r}^+[s_n])$.

3.2. Construct $\mathbf{P}^\bullet$ through Algorithm 3 by considering the set of states $\mathcal{S}_1$. In the sequel, we denote this set by $\{s_1, s_2, \cdots s_n\}$ The input parameters of Algorithm 3 are $\mathbf{P}^-$ and $\mathbf{P}^+$ matrices of size n; vector $\mathbf{out}$ is defined by summing the probabilities over $\mathcal{S} - \mathcal{S}_1$ states: $\forall s_i \in \mathcal{S}_1$, $\mathbf{out}[s_i] = \max(\sum_{s_k \notin \mathcal{S}_1} \mathbf{P}^-[s_i, s_k], 1 - \sum_{s_k \in \mathcal{S}_1} \mathbf{P}^+[s_i, s_k])$. The output matrix $\mathbf{P}^\bullet$ is the lower bounding matrix for all Markov chains in the interval $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ with probability transition matrix $\mathbf{P}$. Thus from Eq. 7, $\forall s_i \in S_1$:

$$1 \leq j \leq n, \ \sum_{k=1}^{j} \mathbf{P}^\bullet[s_i, s_k] \geq \sum_{k=1}^{j} \mathbf{P}[s_i, s_k]$$

3.3. The monotone lower bounding matrix for all the Markov chains in the interval $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ is computed through Algorithm 4 and denoted by $\mathbf{P}^\star$. The input parameter of the algorithm is matrix $\mathbf{P}^\bullet$ obtained in the previous step.

We have the following inequalities for each state $s_i \in \mathcal{S}_1$, for all power matrices ($t \geq 1$) of any Markov chain in the interval $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ with probability transition matrix $\mathbf{P}$:

$$1 \leq j \leq n, \ \sum_{k=1}^{j} (\mathbf{P}^\star)^t[s_i, s_k] \geq \sum_{k=1}^{j} (\mathbf{P})^t[s_i, s_k] \tag{10}$$

These inequalities still hold, if we multiply both part of the inequality for a state s_j by $(\mathbf{r}^+[s_j] - \mathbf{r}^+[s_{j+1}])$ (the $n+1th$ entry for vectors $\mathbf{r}$ and $\mathbf{r}^+$ is assumed to be 0). Then by summing all inequalities over $j = \{1, \cdots n\}$, we can deduce that

$$\sum_{k=1}^{n} (\mathbf{P}^\star)^t[s_i, s_k]\mathbf{r}^+[s_k] \geq \sum_{k=1}^{n} (\mathbf{P})^t[s_i, s_k]\mathbf{r}^+[s_k] \tag{11}$$

This inequality can be rewritten as

$$(\mathbf{P}^\star)^t \cdot \mathbf{r}^+ \geq_{el} (\mathbf{P})^t \cdot \mathbf{r}^+$$

where $\leq_{el}$ denotes the component (element)-wise ordering.

Let $\mathbf{r}$ be the column vector computed for a given $\mathbf{P} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ in the interval. $\forall s_i \in \mathcal{S}_1$:

$$\mathbf{r}[s_i] = \sum_{s_k \in \mathcal{S}_2} \mathbf{P}[s_i, s_k] = \mathbf{P}[s_i, \mathcal{S}_2]$$

Obviously, $\mathbf{r}^+$ provides the maximal vector for all vectors $\mathbf{r}$ computed from any matrix $\mathbf{P}$ in the interval $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$:

$$\mathbf{r}^+ \geq_{el} \mathbf{r}$$

Combining this inequality with 11, we have

$$(\mathbf{P}^\star)^t \cdot \mathbf{r}^+ \geq_{el} (\mathbf{P})^t \cdot \mathbf{r}^+ \geq_{el} (\mathbf{P})^t \cdot \mathbf{r} \tag{12}$$

where power 0 for a matrix is the identity matrix. Let us remark that $(\mathbf{P})^t\mathbf{r}$ represents the probabilities to reach $\mathcal{S}_2$ states within $t+1$ steps.

3.4. We check if the upper bound to reach $\mathcal{S}_2$ states in time interval $[\alpha, \beta]$ remains less or equal to p. Thus we consider maximal probabilities to reach $\mathcal{S}_2$ states at time t (see Eq. 12), that means to reach a state within $t-1$ steps by the power $t-1$ of matrix $\mathbf{P}^\star$ and then within 1 step from this state to a state $\mathcal{S}_2$ by $\mathbf{r}$. Therefore we sum over in time interval $t = (\alpha-1)^+$ to $\beta-1$ [1] where $(\alpha-1)^+ = \max(0, \alpha-1)$. For each state $s \in \mathcal{S}_1$, if the following inequality is satisfied [2] then $s.\phi = \forall^+$

[1] The case where both $\alpha = \beta = 0$ will not be considered, since $\psi_1 \, \mathcal{U}^{[0,0]} \psi_2 \equiv \psi_2$.

[2] When $\beta = \infty$, the sum is infinite. We discuss this case at the end of the section.

$$\left((\sum_{t=(\alpha-1)^+}^{\beta-1} (\mathbf{P}^\star)^t) \cdot \mathbf{r}^+ \right)[s] \leq p \qquad (13)$$

4. We now see if we can label with $\forall^-$. We define two sets: $\mathcal{S}_2^{'} = \{s \mid s.\psi_2 = \forall^+\}$. $\mathcal{S}_1^{'} = \{s \mid s.\psi_2 \neq \forall^+ \wedge s.\psi_1 = \forall^+\}$. Let $\mathcal{S}_1^{'} = \{s_{1,2}, \cdots, s_n'\}$. States out of $S_1^{'}$ will be absorbing. First we construct a column vector $\mathbf{r}^-$ of size n' to compute minimal transition probabilities from states $\mathcal{S}_1^{'}$ to the set of absorbing $\mathcal{S}_2^{'}$ states. $\forall s_i \in \mathcal{S}_1^{'}$:

$$\mathbf{r}^-[s_i] = \max(\sum_{s_k \in \mathcal{S}_2^{'}} \mathbf{P}^-[s_i, s_k], 1 - \sum_{s_k \notin \mathcal{S}_2^{'}} \mathbf{P}^+[s_i, s_k]) \qquad (14)$$

We consider the lower bounding case, thus we consider $\mathbf{P}^-$ instead of $\mathbf{P}^\star$ of the former case. Obviously we have the following inequalities for all power matrices ($t \geq 1$) of any chain restricted to $\mathcal{S}_1^{'}$ in the interval $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ with probability transition matrix $\mathbf{P}$:

$$(\mathbf{P}^-)^t \cdot \mathbf{r}^- \leq_{el} (\mathbf{P})^t \cdot \mathbf{r}^- \leq_{el} (\mathbf{P})^t \cdot \mathbf{r}$$

We check if the upper bound to reach $\mathcal{S}_2^{'}$ states in time interval $[\alpha, \beta]$ exceeds p. Thus for each state $s \in \mathcal{S}_1^{'}$, if the following inequality is satisfied, then $s.\phi = \forall^-$

$$\left((\sum_{t=(\alpha-1)^+}^{\beta-1} (\mathbf{P}^-)^t) \cdot \mathbf{r}^- \right)[s] > p \qquad (15)$$

5. We now see if we can label $\exists^+$. The sets $\mathcal{S}_1$ and $\mathcal{S}_2$ are defined as in case 3.
   5.1. Define a column matrix $\mathbf{r}_m$ of size n for the lower bounding reaching probability to $\mathcal{S}_2$ states from $\mathcal{S}_1$ states :
   $$\mathbf{r}^m[s] = \max(\sum_{s_k \in \mathcal{S}_2} \mathbf{P}^-[s, s_k], (1 - \sum_{s_k \notin \mathcal{S}_2} \mathbf{P}^+[s, s_k]))$$
   5.2. Reorder this vector in the decreasing order. State space $\mathcal{S}_1$ will be also ordered in this order. Let us remark here that this reordering is not required contrary to the previous case but it is heuristic.
   5.3. We guess a matrix $\mathbf{P}_m$ restricted to the set $\mathcal{S}_1 = \{s_1, \cdots, s_n\}$ belonging to $\mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ by applying Algorithm 1 to construct each row.
   5.4. We check if formula ϕ is checked by considering matrix $\mathbf{P}_m$ and the vector $\mathbf{r}_m$. For each state $s \in \mathcal{S}$, If the following inequality is satisfied then $s.\phi = \exists^+$
   $$(\sum_{t=(\alpha-1)^+}^{\beta-1} (\mathbf{P}_m)^t) \cdot \mathbf{r}_m[s] \leq p$$
6. We now see if we can label $\exists^-$. The sets $\mathcal{S}_1^{'}$ and $\mathcal{S}_2^{'}$ are defined as in case 4.
   6.1. Define a column matrix $\mathbf{r}_M$ of size n' for the upper bounding reaching probability to $\mathcal{S}_2^{'}$ states from $\mathcal{S}_1^{'}$ states :
   $$\mathbf{r}_M[s] = \min(\sum_{s_k \in \mathcal{S}_2^{'}} \mathbf{P}^+[s, s_k], 1 - \sum_{s_k \notin \mathcal{S}_2^{'}} \mathbf{P}^-[s, s_k])$$
   6.2. Reorder this vector in the increasing order. The set $\mathcal{S}_1^{'}$ will be also ordered with respect to this order.
   6.3. Similiar to the former case, we construct a matrix $\mathbf{P}_M$ restricted to the set of states $\mathcal{S}_1^{'} = \{s_1, \cdots, s_{n'}\}$ by applying Algorithm 2 to construct each row.
   6.4. We now check if formula ϕ is checked by considering matrix $\mathbf{P}_M$ and the vector $\mathbf{r}_M$. For each state $s \in \mathcal{S}^{'}$, if the following inequality is satisfied then if $s.\phi = \exists^+$ then $s.\phi = \exists^{+-}$ else $s.\phi = \exists^-$
   $$(\sum_{t=(\alpha-1)^+}^{\beta-1} (\mathbf{P}_M)^t) \cdot \mathbf{r}_M[s] > p$$
7. In the case $\alpha = 0$, for all states yet already labelled
   - $s.\psi_2 = \exists^{+-} \Longrightarrow s.\phi = \exists^-$
   - $s.\psi_2 = \exists^+ \Longrightarrow s.\phi = \exists^-$
8. We assign $s.\phi =?$ to all the states which have not already labelled.

$\boxed{\phi = \mathcal{D}_{\lhd r}(\psi)}$

Principle. Once ψ has been evaluated, the standard method consists to merge states fulfilling ψ in an absorbing state and to study the behaviour of the transformed chain during the interval $[0, \infty[$. More precisely, let us recall that given a subset of states $\mathcal{S}^{'}$, the vector indexed by $\mathcal{S} - \mathcal{S}^{'}$ corresponding to the mean time to reach $\mathcal{S}^{'}$ is given by the formula:

$$(\sum_{t \geq 0} (\mathbf{P})^t) \mathbf{1}_m \qquad (16)$$

where $\mathbf{P}$ is the transition probability matrix restricted to $\mathcal{S} - \mathcal{S}^{'}$ states of cardinality m and $\mathbf{1}_m$ is a column vector of size m with all entries equal to 1. This is the starting point of our method which substitutes in the equation (16) a matrix for $\mathbf{P}$. As for the case of "until" operator the choice of the matrix depends on the information one looks for: case 1 involves the stochastic comparison, case 2 uses $\mathbf{P}^-$ for a pointwise lower bounding and cases 3 and 4 construct an *ad hoc* matrix.

As in the former cases, we consider here the case $\lhd = \leq$.

1. We first see if we can label $s.\phi = \forall^+$. We define $\mathcal{S}_1 = \{s \mid s.\psi \neq \forall^+\}$. Our goal is to provide an upper bound on the mean reaching time to states for which ψ is surely satisfied ($\mathcal{S} - \mathcal{S}_1$). Thus the states out of $\mathcal{S}_1$ is made absorbing. We construct first the lower bounding matrix $\mathbf{P}^\bullet$ restricted to $\mathcal{S}_1 = \{s_1, \cdots, s_n\}$ states from Algorithm 3. The input parameter $\mathbf{out}$ (the minimal transition probabilities to the absorbing state) is computed as follows: $\forall s_i \in \mathcal{S}_1$, $\mathbf{out}[s_i] = max(\sum_{s_k \notin \mathcal{S}_1} \mathbf{P}^-[s_i, s_k], 1 - \sum_{s_k \in \mathcal{S}_1} \mathbf{P}^+[s_i, s_k])$. The monotone version is built by Algorithm 4. Thus for any chain $\mathcal{M} \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ with probability transition matrix $\mathbf{P}$:
$$(\sum_{t \geq 0} (\mathbf{P}^\star)^t) \mathbf{1}_n \geq_{el} (\sum_{t \geq 0} (\mathbf{P})^t) \mathbf{1}_n$$

For each state in $\mathcal{S}_1$, if the following inequality is satisfied then $s.\phi = \forall^+$

$$\left((\sum_{t\geq 0} (\mathbf{P}^\star)^t) \mathbf{1}_n \right) [s] \leq r$$

Observe that some components of this computed vector could be infinite. However we apply beforehand algorithm 5 that determines the subset of $\mathcal{S}_1$ that corresponds to states with finite value. Then we only compute the above infinite sum for this subset of states.

2. We now see if we can label $s.\phi = \forall^-$. We consider lower bounding case to reach states satisfying ψ. Thus we define $\mathcal{S}_1^{'} = \{s \mid s.\psi = \forall^-\}$, which is the set of states for which ψ is surely not satisfied. If the infinite sum (see Eq. 16) is greater than r for the lower bounding case, one can conclude that the mean reaching time is always greater than r. Hence for each state $s \in \mathcal{S}_1^{'}$, if the following inequality is satisfied then $s.\phi = \forall^-$

$$\left(\sum_{t\geq 0} (\mathbf{P}^-)^t) \mathbf{1}_{n'} \right) [s] > r$$

3. We now see the case for label $s.\phi = \exists^+$. We consider the set of states $\mathcal{S}_1$ as in case 1 and guess a matrix $\mathbf{P}_m \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ restricted to $\mathcal{S}_1$. Each row is constructed by Algorithm 1 in order to minimize the transition probabilities. Thus for each state in $\mathcal{S}_1$, if the following inequality is satisfied then $s.\phi = \exists^+$

$$\left(\sum_{t\geq 0} (\mathbf{P}_m)^t) \mathbf{1}_n \right) [s] \leq r$$

4. Similar to the previous case, we guess a matrix $\mathbf{P}_M \in \mathcal{M}(\mathbf{P}^-, \mathbf{P}^+)$ restricted to the set of states $\mathcal{S}_1^{'}$ (defined in case 2.) The rows are constructed by Algorithm 2 in order to maximize the transition probabilities. For each state $s \in \mathcal{S}_1^{'}$, if the following inequality is satisfied then if $s.\phi = \exists^+$ then $s.\phi = \exists^{+-}$ else $s.\phi = \exists^-$

$$\left((\sum_{t\geq 0} (\mathbf{P}_M)^t) \mathbf{1}_{n'} \right) [s] > r$$

5. For all states which are not already labelled we assign ?.

Remark. Observe that the convergence of infinite sums involved in the algorithms related to the $\mathcal{D}$ and the $\mathcal{U}$ operators can be checked before starting the computation. This is performed either by standard graph analysis when the substochastic matrix is arbitrary (based on the decomposition in strongly connected components) or by algorithm 5 when the matrix is monotone. In both cases, the algorithms determine the subset of initial states for which the computation is necessary and transform the matrix (depending on the considered operator) in such a way that the convergence is ensured. As usual, the convergence is exponentially quick (since the sum is geometric). Therefore for a reasonable precision, a finite approximating sum is efficiently computed.

V. Related Work

A. Interval-valued Markov chains

In [16] IMCs are introduced to specify the expected behavior of a model under uncertainties. The obtention of parameters of an IMC is considered in [18]. Following another approach, in [25], algorithms are proposed to build extremal monotone chains for an IMC. These results have been applied in the framework of the bounding aggregation for Near Complete Decomposable Markov chains [25], [22]. In [13], IMC subchains are considered and polynomial time algorithms are designed to compute the maximal monotone lower bound both in continuous and discrete time settings. These results have been applied to study different reliability and performance problems [14].

While the bounds developed in [25], [13] can be applied to analyze the transient and the steady-state behaviour, in [6], P. Buchholz only focused on the steady-state analysis and built optimal bounds for steady-state distributions based on the polyhedra theory initially proposed by [9].

The model checking of interval valued Markov chains has been investigated in [23]. The authors showed that the probability to satisfy a PCTL formula are specified by polynomial inequalities (rather than linear ones in the case of DTMC) which leads to a PSPACE algorithm. They also established that PCTL model checking is NP-hard and co-NP-hard. In [7], these results have been generalized to ω-PCTL logic.

B. Semi-Decision procedures for Model Checking

Abstraction is an useful technique in order to analyze systems with huge state spaces. It consists in grouping states and producing an abstract system which can be an "under" or "over" approximation of the original system. It has been first applied in the framework in the discrete event systems and has been recently generalized for probabilistic systems. In [11] the abstraction of a DTMC naturally yields a continuous time interval valued Markov chain. Then using three-valued semantic (YES, NO, DON'T KNOW) the authors apply a method based on resolution of an associated Markov Decision Process (MDP). [17] handles the case of CTMCs by uniformising the CTMC then applying the abstraction procedure as in [11]. A different view of abstraction is proposed in [8] whose goal is to obtain a "purely" stochastic system excluding the non determinism induced by the intervals.

All previous approaches are based on the bounds for state probabilities. In this context it must be observed a general theory exists: stochastic comparison [20]. Bounding methods are suitable to apply in model checking, since we need to check if some constraints are satisfied or not without considering exact values. The stochastic comparison approach provides an interesting alternative for model checking since this approach lets us to provide the bounds on transient distributions as well as the stationary distribution of the underlying Markovian model. Indeed, the stochastic comparison of distributions

provides the inequalities on the partial sum of probabilities. In model checking, given a formula $\mathcal{F}$, the verification is resumed to compute the sum of probabilities of states satisfying $\mathcal{F}$ in a transient or the stationary distribution. We call this set of states $S_{success}$. Thus we must first reorder the state space in order to put $S_{success}$ states at the end or in the beginning of the state space. This is necessary in order to extract the inequalities on the sum of the probabilities of these states from the bounding distributions.

The second step is the verification step. Let B_{inf} and B_{sup} be the bounds on the probabilities for $S_{success}$ states. The verification depends on the comparison operator, in the case $\triangleleft$ is $\leq$:

- if $B^{sup} \leq p$ then we can decide that formula $\mathcal{F}$ is checked (YES).
- if $B^{inf} > p$ then we can decide that formula $\mathcal{F}$ is not checked (NO).
- otherwise it is not possible to decide with these bounding values (DON'T KNOW).

This approach has been applied in order to reduce the complexity of the underlying Markov chains. In [21], the state space is reduced by applying bounded aggregations to study PCTL state formulas. In [4], bounding models which have closed-form solutions to compute transient and the steady-state distributions to check CSL formulas have been considered. Since the underlying formulas are checked by means of closed-form solutions for underlying distributions, the complexity is largely reduced.

VI. Conclusion

Stochastic comparison has demonstrated its usefulness to overcome the complexity of state based performance evaluation methods. Here we have proposed to apply it for model checking PCTL formulas over IMCs. To this aim, we have designed a semi-decision procedure. This procedure has three advantages: its efficiency (the complexity of known exact algorithms is in PSPACE), its scope (previous algorithms do not deal with the mean reachability time operator) and the kinds of answers (it includes the partial answers $\exists^+, \exists^-$).

The main practical problem for methods based on stochastic comparison is the appropriate choice of the order over states. In our case the order may be different during every operator evaluation step and thus is the critical factor for accuracy of the bounds. So we plane to develop a prototype for high level models of IMCs (SANs, Stochastic Petri nets, etc.) and experiment heuristics that rely on the structure of this model. The specification of intervals associated with an IMC are usually derived from the uncertainty about transitions of the high level model. So the number of different intervals in the IMC is very small w.r.t. the size of the chain. We want to take into account this feature in order to improve the efficiency and/or the accuracy of our method.

Acknowledgement

This work is partially supported by French ANR project ANR06-SETIN-002, CheckBound.

References

[1] A. Aziz, K. Sanwal, V. Singhal, and R. Brayton. Model Checking Continuous Time Markov Chains. *ACM Trans. on Comp. Logic*, 1(1), pages 162-170, 2000.

[2] C. Baier, L. Cloth, B. R. Haverkort, M. Kuntz, M. Siegle. Model Checking Action- and State-Labelled Markov Chains. In *DSN 2004:*, pages 701-710, 2004.

[3] C. Baier, B. Haverkort, H. Hermanns, and J.P. Katoen. Model-Checking Algorithms for Continuous-Time Markov Chains. In *IEEE Trans. Software Eng.* 29(6), pages 524-541, 2003.

[4] M. Ben Mamoun, N. Pekergin and S. Younès. Model checking of continous-time markov chains by closed-form bounding distributions. In *QEST2006*, pages 199-211, 2006.

[5] A. Bianco, L. de Alfaro. Model Checking of Probabilistic and Nondeterministic Systems. In *FST TSC95, LNCS 1026*, pages 499-513, Springer 1995.

[6] P. Buchholz. An improved method for bounding stationary measures of finite Markov Processes. *Performance Evaluation*, 62(1-4) pages 349-365, 2005

[7] K. Chatterjee, K. Sen and T. A. Henzinger. Model-Checking omega-Regular Properties of Interval Markov Chains. In *Proc. FoSSaCS08, LNCS 4962*, pages 302-317, Springer 2008.

[8] R. Chadha, M. Viswanthan and R. Viswanthan. Least upper bounds for probability measures and their applications to abstractions. In *CONCUR 2008 - 19th International Conference on Concurrency Theory, LNCS5201*, pages 264-278. Springer, 2008.

[9] P.J. Courtois and P. Semal. Computable bounds on conditional steady-state probabilities in large Markov chains and queueing models. *IEEE Journal on Selected Areas in Communications*, 4(6), pages 926-937, 1986.

[10] S. Donatelli, S. Haddad, J. Sproston. CSLTA: an Expressive Logic for Continuous-Time Markov Chains. In *QEST 2007*, pages 31-40, 2007.

[11] H. Fecher, M. Leucker, and V. Wolf. Don't know in probabilistic systems. In *Proceedings of 13th International SPIN Workshop on Model Checking of Software (SPIN'06) LNCS 3925*, pages 71 - 88, Springer 2006.

[12] J.M. Fourneau and N. Pekergin. An algorithmic approach to stochastic bounds. In *LNCS 2459, Performance evaluation of complex systems: Techniques and Tools*, pages 64-88, 2002.

[13] S. Haddad and P. Moreaux. Sub-stochastic matrix analysis for bounds computationTheoretical results. *European Journal of Operational Research*, 176(0), pages 999-1015, 2007.

[14] S. Haddad and P. Moreaux. Sub-stochastic matrix analysis and performance bounds. *Research Reprt RAP-CReSTIC-1, CReSTIC*, Universit de Reims Champagne-Ardenne, France, 2004.

[15] H. Hansson and B. Jonsson. A logic for reasoning about time and reliability. *Formal Aspects of Computing* 6, pages 512-535, 1994.

[16] B. Jonnson and K.G. Larsen. Specification and refinement of probabilistic processes. In *Proceedings of the IEEE Symp. on Logic in Computer Science*, pages 266-277, 1991.

[17] J. P. Katoen, D. Klink, M. Leucker, and V. Wolf. Three-valued abstraction for continuous-time Markov chains. In *Proc. CAV07, LNCS 4590)*. pages 311-324, Springer 2007.

[18] I.O. Kozine and L.V. Utkin. Interval-valued finite Markov chains. *Reliable Computing*, 8(2) pages 97-113, 2002.

[19] F. Laroussinie, J. Sproston. Model Checking Durational Probabilistic Systems. In *FOSSACS'05, LNCS 3441*, pages 140-154, Springer 2005.

[20] A. Muller and D. Stoyan, *Comparison Methods for Stochastic Models and Risks*, Wiley, New York, 2002.

[21] N. Pekergin, S. Younès. Stochastic Model Checking with Stochastic Comparison. In *Proc. EPEW 2005, LNCS 3670*, pages 109-123, Springer 2005.

[22] N. Pekergin, T. Dayar and D. Alparslan. Componentwise bounds for nearly completely decomposable Markov chains using stochastic comparison and reordering. *European Journal of Operational Research*, 165, pages 810-825, 2005.

[23] K. Sen, M. Viswanathan, G. Agha. Model-Checking Markov chains in the presence of uncertainties. In *Proc. Tacas06, LNCS 3920*, pages 394-410, Springer 2006.

[24] K. Trivedi. *Probability and Statistics with Reliability, Queuing and Computer Science Applications* Wiley, New York 2002.

[25] L. Truffet. Near Complete Decomposability: Bounding the error by Stochastic Comparison Method. *Advances in Applied Probability*, pages 830-855, 1997.