

Quantified reduced views of state graphs using Markovian and timed observational equivalence

K. Drira¹ and Y. Atamna² and G. Juanole¹

¹*LAAS-CNRS 7, av. du Colonel Roche, 31077 Toulouse Cedex, France.*

e-mail: {drira,juanole}@laas.fr, Tel: (+33)61336200, Fax: (+33)61336411

²*West Virginia Univ., Dep. of Elec. and Comp. Eng., P.O. Box 6104, Morgantown, WV, 26506-6104, USA.*

e-mail: youcef@ece.wvu.edu, Tel: (+1)3042936371, Fax: (+1)3042938602

Abstract

This paper defines a new equivalence relation over the states of graphs labeled with probabilities and time feature. The Markovian aggregation technique and the observational transition relation compatibility are used. A computation algorithm is provided using the technique of sequence equivalences of Milner. The reduced quantified view of a state graph is defined as the quotient graph associated with the so-called Markovian-observational equivalence.

Keywords

Observational equivalence, Markovian aggregation, Probabilistic equivalence relation, quotient graph

1 INTRODUCTION

The labeled state graphs constitute a formalism which is extensively used for representing the dynamic behavior of distributed communicating systems. A labeled state graph consists of a set of global states (a global state is composed of all the states of communicating entities (local states)), and a set of arcs linking the global states and a set of labels associated with the set of arcs. Most often, the labels are of the “qualitative” or

logical (by opposition to quantitative) type appearing in the form of Predicate/Action (a predicate can be a local condition or message reception; an action can be only local or covers too the emission of a message).

A labeled state graph representing the dynamic behavior of a system is most often obtained by composing models representing each communicating entity (for example if each communicating entity is represented by a Petri Net (local Petri net), the global states are the global Petri net markings (composition of local Petri nets)). A labeled global state graph allows analysis of general properties (that do not depend on the system mission) and specific properties (that depend on the system mission and that are therefore based on graph interpretations). Generally, the labeled global state graphs can be of a large size and it is then highly difficult to validate the specific properties (given the difficulty in interpreting a large size graph). The abstract views of reduced graphs which are equivalent to the initial graph according to an equivalence criterion (which depends of course on the intended observation), offer a user-friendly means of validating specific properties. Thus, in the field of global state graphs with qualitative type labels (the well-used model of Plotkin, the Labeled Transition Systems), one often makes use of abstract views derived from projection techniques based on equivalence relations: the abstract view is given by a reduced graph which represents the modeled system relative to a set of events. The observational equivalence (known also as bisimulation equivalence of Park [Park, 1981]), proposed by Milner [Milner, 1980], is the basis of many of these equivalences. The most known are compared in [Van Glabbeek, 1990]. Projection (or reduction) is now a largely used technique to verify communicating systems [Simon, 1984, Drira, 1993].

However, in the current technological context featuring the advent of real time distributed systems (concept of time constraints), one cannot only represent the dynamic behavior of these systems with labeled global state graphs only featuring qualitative attributes. Thus quantitative type attributes must equally be considered such as probabilities and times. In these conditions, validating the specific conditions requires obtaining quantified abstract views. To our knowledge, the notion of quantified abstract views is seldom or never utilized (except those, based on the exploitation, of the Markovian aggregation technique [Kemeny, 1960], or the first passage time concept [Atamna, 1995]).

In this paper, we propose to enhance probabilistic (and more generally stochastic) concept of aggregation with observational criteria. On the one hand, this allows us to compare states of a system with respect to both logical criteria (possible traces, associated deadlocks and all logical properties captured by observational equivalence and expressed by the Hennessey-Milner Logic [Hennessey, 1985]) and performance criteria (all properties captured by the Markovian aggregation technique). On the other hand, we will provide an algorithm to reduce the description of stochastic processes while preserving both criteria. Two cases are considered : the probabilized and the timed probabilized equivalences.

This paper is composed by the following three sections: section 2 defines the randomized state graph, the structure obtained from Stochastic Timed Petri Nets and consisting in showing transitions of the modeled system with a probability label and a timing label. The observational-Markovian equivalence is then defined. A computation algorithm based on k-equivalence is proposed. Section 3 illustrates the utilization of the result of the reduction (i.e. the quotient graph) on a small example of two concurrently executing tasks. Section 4 gives some concluding remarks and future direction.

2 RANDOMIZED STATE GRAPH REDUCTION

Definition 1 ((finite) Randomized State Graph: RSG) A randomized state graph (or shortly a randomized graph) is a triple $\mathcal{G} = \langle \Omega, L, \Delta \rangle$, where:

- Ω is a finite set of states. This set is ranged by $p, q, p', q' \dots$.
- L is a finite set of transition identifiers called alphabet. We shall call **actions** element of this set. This set is ranged by $t, a, b, c \dots$.
- $\Delta \subseteq \Omega \times L \times [0, 1] \times [0, \infty) \times \Omega$. An element $(p, t, \nu, \theta, q) \in \Delta$ is denoted $p \xrightarrow{t, \nu, \theta} q$. The value ν denotes the probability to reach state q by executing transition t from state p after a sojourn of θ time units at this state.

The following condition is assumed on RSG:

$$\forall p \in \Omega : \left(\sum_{t \in L, \nu \in [0, 1], \theta \in [0, \infty), q \in \Omega, p \xrightarrow{t, \nu, \theta} q} \nu \right) \leq 1$$

When the graph is deterministic (i.e. at the most one successor q is possible from a state p by executing action t) the value ν of the transition $p \xrightarrow{t, \nu} q$ denotes the probability to execute action t .

The randomized state graph represents the dynamic behavior of a Stochastic Timed Petri Net [Juanole, 1991, Atamna, 1994]. This graph is a stochastic object and is isomorphic to an embedded chain [Howard, 1971] which makes usual performance evaluation techniques possible. The transitions are labeled by the Petri Net transition inducing the state changing, t , the branching probability, ν , and the mean firing time, θ , of the transition (called also the conditional sojourn time).

When the time feature is not used, we simply write $p \xrightarrow{t, \nu} q$ to say that state q is reached by transition t from state p with the probability ν . Theoretically, we may accept arcs labeled by the same transition identifier t and having the same origin and destination states: $p \xrightarrow{t, \nu, \theta} q$ and $p \xrightarrow{t, \nu', \theta'} q$. In this case, when the time feature is ignored, the probability value is the sum of the probability values associated with all time features. That is in the expression $p \xrightarrow{t, \nu} q$, the value ν denotes $\sum_{\nu' \in [0, 1], \theta \in [0, \infty), p \xrightarrow{t, \nu', \theta} q} \nu'$.

When both probability and mean time values are not used, we simply write $p \xrightarrow{t} q$ to say that state q is reached by transition t from state p (by non-zero probability).

In order to simplify this presentation, we will consider the time feature only at the end of this paper to make a generalization of the untimed equivalences which will be defined in the next section.

The Randomized State Graph is different from the probabilistic transition systems (PTS) of Larsen and Skou [Larsen, 1991]. In PTS, probability is used to annotate non-deterministic transitions (i.e. transitions that are in conflict and which have the same transition identifier). In a transition, $p \xrightarrow{t, \nu} q$, of this model, the value ν denotes the probability that the system reaches q after the action t is executed at state p (that is the conditional probability associated with the fact that the system reaches q knowing that

action a is executed). This means that the probability to “leave”¹ a state by executing a given action (represented by transition identifier t) belonging to the output of the state is 1. Formally in the PTS model it is supposed that (when from state p a t transition is possible):

$$\forall p \in \Omega, \forall t \in L : \left(\sum_{\nu \in [0,1], \theta \in [0,\infty), q \in \Omega, p \xrightarrow{t,\nu} q} \nu \right) = 1$$

The randomized state graph does not suppose this condition and substitutes it by the weakest assumption given in definition 1 and which allows probabilities associated with different transition identifiers to be mixed. On the one hand, this has the advantage to provide weak equivalence allowing RSG to be more reduced than with Larsen’s like equivalence. On the other hand, the reduced RSG remains a stochastic object supporting performance evaluation techniques.

The PTS model is derived from the Labeled Transition System model of Plotkin and Milner’s semantics of communicating systems based on observational criteria [Milner, 1980]. Within this model, systems are seen as a black box with buttons (representing the alphabet of the system). The system interacts with its environment: the weather as called by Milner, or the tester or observer in the testing theory of Larsen. PTS model is then an extension of LTS with probabilistic criteria. In [Hansson, 1990] an alternating model (which uses two kinds of transitions: probabilistic transition and logical transitions) is used to represent quantified behavior of communicating systems. By introducing the idea of alternating bisimulation, an equivalence relation is defined within this model and used for axiomatic comparison of algebraic terms of a timed and probabilized calculus.

The approach we propose here is different. We want to provide “observational” (or logical) criteria to distinguish stochastic processes represented by their randomized state graph. This graph is in practice automatically generated by analyzing a higher model: Stochastic Timed Petri Nets. The labels of transitions (i.e. actions) are not representing interaction (or communication) between the described system and its environment. They represent actions the system may freely (or internally) execute without being controlled by the environment. Such systems can be seen as a black box with several pilot-lamps (representing the actions set) instead of buttons. This makes unrealistic the PTS assumption which admits that the probability to leave state p by executing action a is 1, for any possible actions at this state. RSG model supposes that the action the system will execute at a given state does not depend on the choice of any environment.

A probabilistic extension for transition systems was proposed in [Christoff, 1991] and is similar to the randomized state graph we are using in this paper. This work considers comparison of systems w.r.t. testing equivalences (probabilistic traces and failures). Contrarily to the approach we propose in this paper, the work of Christoff does not consider reduction nor observational equivalence.

¹This unformal formulation is ambiguous. So we note here that saying “the probability to leave a given state” does not exclude this state from the set of reached states.

2.1 Observational Markovian aggregation

This section presents first separately properties that an equivalence relation must satisfy to be considered as a Markovian aggregation or to be considered compatible with the transition relation.

2.1.1 Notations

This section is dedicated for mathematical representation of binary relation in general and of equivalence relation in particular. Equivalence relations, \asymp , over Ω are transitive reflexive and symmetric binary relations that can be represented as subsets of the cross product: $\asymp \subseteq \Omega \times \Omega$. A couple of states that are equivalent with respect to the relation \asymp are usually denoted indifferently “ $p \asymp q$ ” or “ $(p, q) \in \asymp$ ”. In this paper we will use the first notation.

partition associated with an equivalence relation Every equivalence relation induces a partition of the set Ω with disjoint subsets of Ω . The notation Ω/\asymp denotes the partition of Ω associated with the equivalence relation \asymp , and is called the quotient set of Ω by \asymp . That is $\Omega/\asymp = \{C_1, \dots, C_n\}$ is a set of subsets of Ω such that every element of Ω is in one and only one equivalence class C_i :

- $C_i \subseteq \Omega, \forall i \in \{1, \dots, n\}$
and
- The partition is a coverage of the whole set: $C_1 \cup C_2 \cup \dots \cup C_n = \Omega$
and
- different classes have no common elements: if $i \neq j$ then $C_i \cap C_j = \emptyset$.

Elements of the same class are equivalent states with respect to the considered equivalence relation: $\forall p, q \in \Omega, \forall C \in \Omega/\asymp: (p \in C \text{ and } q \in C) \text{ iff } p \asymp q$.

Example Figure 2 gives two examples of partitions associated with observational aggregation and Markovian aggregation which are two equivalence relations over the state space of the RSG. For the first equivalence, the partition is composed of 3 classes: $\Omega/\approx = \{\underbrace{\{1, 2\}}_{C_1}, \underbrace{\{3, 4\}}_{C_2}, \underbrace{\{5\}}_{C_3}\}$. For the second equivalence, the partition is composed of the 4 following classes: $\Omega/\approx = \{\underbrace{\{1, 2\}}_{C'_1}, \underbrace{\{3\}}_{C'_2}, \underbrace{\{4\}}_{C'_3}, \underbrace{\{5\}}_{C'_4}\}$. Without loss of rigor, we sometimes confuse equivalence relations (\asymp) with their associated partitions (Ω/\asymp) instead of their representation as a set of couple of states.

When using the cross product notations, we can write each equivalence relation as the set of all couples of equivalent states. For the example of Figure 2, we have: $\approx = \{(1, 1), (2, 2), (1, 2), (2, 1), (3, 3), (4, 4), (3, 4), (4, 3), (5, 5)\}$, and $\equiv = \{(1, 1), (2, 2), (1, 2), (2, 1), (3, 3), (4, 4), (5, 5)\}$.

Comparison of equivalence relations Comparison of equivalence relation may be defined and used using the cross product representation and the usual set inclusion relation \subseteq which is defined by: $A \subseteq B$ iff ($a \in A$ implies $a \in B$). For the example of equivalence relations considered in Figure 2, we have: $\equiv \subseteq \approx$ since every couple of states of the set \equiv is also an element of the set \approx . When a relation \asymp_2 is included in \asymp_1 we say also that \asymp_2 is stronger than \asymp_1 because “ $\asymp_2 \subseteq \asymp_1$ ” is equivalent to “ $p \asymp_2 q$ implies $p \asymp_1 q$ ” which means that states that are equivalent for \asymp_2 are necessarily equivalent for \asymp_1 . Comparison of equivalence relations can also be equivalently defined using associated partitions. Formally it is easy to prove that “ $\asymp_2 \subseteq \asymp_1$ ” is equivalent to $\forall C \in \Omega / \asymp_2$ there exists $C' \in \Omega / \asymp_1$ such that $C \subseteq C'$. Moreover, it is easy to prove that this statement is also equivalent to the following one: every equivalence class of \asymp_1 may be partitioned by an associated set of classes belonging to \asymp_2 . Formally this means that if $\{C_i\}_{i \in I}$ is the partition associated with \asymp_1 and $\{C'_j\}_{j \in J}$ is the partition associated with \asymp_2 then there a partition of J , denoted $J/\pi = \{J_1, \dots, J_k\}$, such that $\forall K \in J/\pi$ the set $\{C'_k\}_{j \in K}$ is a partition of one (and only one) class C_i . This decomposition will be helpful for the proof of some properties over decreasing family of equivalence relations. The following figure illustrates this decomposition.

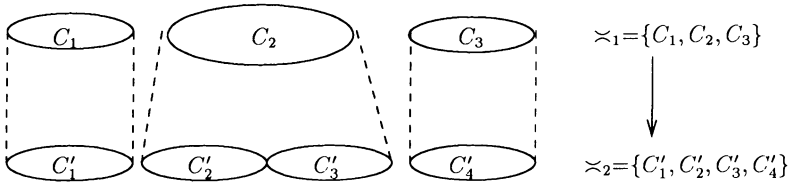


Figure 1: Splitting equivalence classes

In the sequel we will consider, over the state space Ω , equivalence relations that verify a given condition (as the Markovian condition), and we will look for the “largest” equivalence which verifies this condition. That is an equivalence relation $\asymp_L \subseteq \Omega \times \Omega$ such that condition P holds for \asymp_L and such that every equivalence relation $\asymp \subseteq \Omega \times \Omega$ which verifies P is necessarily included within \asymp_L . In other words we will look for the partition of ω which groups as many states as possible in the same class (and then which has the less classes as possible) without breaking condition P . For the purpose of simplicity we will consider the cross product representation in the description of computation algorithm associated with the equivalence relations we will define here.

Conjunction of relations is defined as intersection of associated sets. The statement “ $p \asymp_1 q$ and $p \asymp_2 q$ ” is equivalent to “ $p (\asymp_1 \cap \asymp_2) q$ ”.

2.1.2 Markovian aggregations

The Markovian aggregation is an analysis technique that consists in lumping the states of a stochastic process into classes, in a way that guarantees the following strong condition:

the probability for passing from one state class to another one does not depend on the state, and is the same for all the states of the class it belongs to.

In [Kemeny, 1960], a strong condition of aggregation is given. By applying this condition to the graph structure, we are using here, the following definition is derived: a partition \equiv is said to be compatible with the Probabilistic specification, and called *Markovian equivalence*, when it verifies the following property, called Markovian property:

$$\forall p, q \in \Omega : p \equiv q \text{ implies } \forall C \in \Omega/\equiv : Prob(p \rightsquigarrow C) = Prob(q \rightsquigarrow C) \quad (1)$$

where

$$Prob(p \rightsquigarrow C) = \sum_{q \in C} Prob(p \rightsquigarrow q)$$

denotes the branching probability from state p to class C , and where

$$Prob(p \rightsquigarrow q) = \sum_{\nu \text{ such that } \exists t: p \xrightarrow{t, \nu} q} \nu$$

denotes the branching probability from state p to state q . That is the probability to reach p from q by executing only one transition.

2.1.3 Transition relation compatibility: observational property

An equivalence relation \approx is called observational aggregation (or equivalently \approx is said to be compatible with the transition relation) when it verifies the following property called the observational property:

$$p \approx q \text{ implies } \begin{cases} (i) \quad \forall t, \text{ whenever } p \xrightarrow{t} p' \text{ then for some } q', q \xrightarrow{t} q' \text{ and } p' \approx q' \\ (ii) \quad \forall t, \text{ whenever } q \xrightarrow{t} q' \text{ then for some } p', p \xrightarrow{t} p' \text{ and } p' \approx q' \end{cases} \quad (2)$$

The transition-compatibility property is equivalent to the following condition:

$$p \approx q \text{ implies } \forall C \in \Omega/\approx, p \xrightarrow{t} C \text{ iff } q \xrightarrow{t} C \quad (3)$$

Where $p \xrightarrow{t} C$ means $\exists p' \in C : p \xrightarrow{t} p'$.

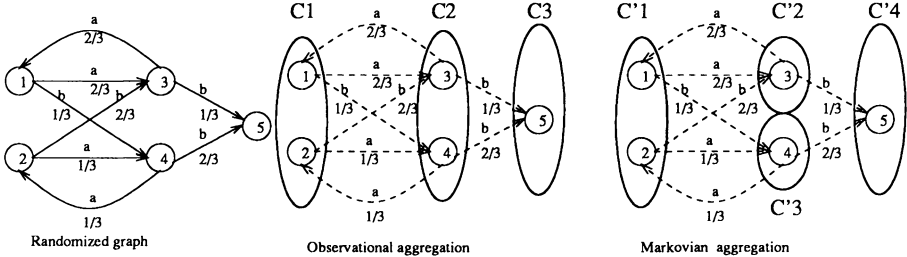


Figure 2: example of observational and Markovian aggregations

Observational aggregation is not a Markovian aggregation: $Prob(3 \rightsquigarrow C1) \neq Prob(4 \rightsquigarrow C1)$.

2.1.4 Observational-Markovian aggregation

In this section we aim to define the largest Markovian aggregation which is compatible with an observational criteria defined as the transition relation compatibility relation. This aggregation is characterized as the largest equivalence relation which is compatible with both properties: the observational property and the Markovian property.

A new probabilized equivalence is defined over the states space of a Randomized graph as the largest equivalence relation \cong verifying the following property:

$$p \cong q \text{ implies } \begin{cases} (i) & \forall t, \text{ whenever } p \xrightarrow{t} p' \text{ then for some } q', q \xrightarrow{t} q' \text{ and } p' \cong q' \\ (ii) & \forall t, \text{ whenever } q \xrightarrow{t} q' \text{ then for some } p', p \xrightarrow{t} p' \text{ and } p' \cong q' \\ (iii) & \forall C \in \Omega/\cong, Prob(p \rightsquigarrow C) = Prob(q \rightsquigarrow C) \end{cases} \quad (4)$$

Clauses (i) and (ii) express that equivalent states have same “progress” possibilities whatever the executed transition is. Clause (iii) expresses that equivalent states have the same probability to lead to a given class C .

Similarly to the original definition of the equivalence introduced by Milner in CCS, this equivalence may be characterized as the intersection of k -equivalences.

Definition 2 (probabilized k -equivalences)

- $p \cong_0 q$ is always true
- $p \cong_{k+1} q$ iff $\begin{cases} (i) & \forall t, \text{ whenever } p \xrightarrow{t} p' \text{ then for some } q', q \xrightarrow{t} q' \text{ and } p' \cong_k q' \\ (ii) & \forall t, \text{ whenever } q \xrightarrow{t} q' \text{ then for some } p', p \xrightarrow{t} p' \text{ and } p' \cong_k q' \\ (iii) & \forall C \in \Omega/\cong_k, Prob(p \rightsquigarrow C) = Prob(q \rightsquigarrow C) \end{cases}$

The following property states the k -equivalence family is decreasing and will be used for the proof of the next proposition.

Property 2.1 ($\{\cong_k\}_{k \geq 0}$ is decreasing) *The sequence $\{\cong_k\}_{k \geq 0}$ verifies: $\cong_{k+1} \subseteq \cong_k$.*

Proof We show by induction that: for all k , $p \cong_{k+1} q$ implies $p \cong_k q$.

First, we have obviously $p \cong_1 q$ implies $p \cong_0 q$. Assume that $p \cong_{n+1} q$ implies $p \cong_n q$ for all $n \leq k-1$, let show that $p \cong_{k+1} q$ implies $p \cong_k q$.

$$p \cong_{k+1} q \quad \text{implies} \quad \forall t, \text{ whenever } p \xrightarrow{t} p' \text{ then for some } q', q \xrightarrow{t} q' \text{ and } p' \cong_k q' \\ \text{(by induction hypo.) implies} \quad \forall t, \text{ whenever } p \xrightarrow{t} p' \text{ then for some } q', q \xrightarrow{t} q' \text{ and } p' \cong_{k-1} q'$$

This means that \cong_{k+1} satisfies the condition (i) of the definition of \cong_k . The proof for (ii) is similar. It remains to show that \cong_{k+1} satisfies the condition (iii) of the definition of \cong_k . This follows from the fact that if A and B are two disjoint subsets of Ω (i.e. $A \cap B = \emptyset$) then $Prob(p \rightsquigarrow A \cup B) = Prob(p \rightsquigarrow A) + Prob(p \rightsquigarrow B)$; and then if $Prob(p \rightsquigarrow A) = Prob(q \rightsquigarrow A)$ and $Prob(p \rightsquigarrow B) = Prob(q \rightsquigarrow B)$ then $Prob(p \rightsquigarrow A \cup B) = Prob(q \rightsquigarrow A \cup B)$. Indeed, $Prob(p \rightsquigarrow C) = Prob(q \rightsquigarrow C)$ for all $C \in \Omega/\cong_k$ and $\cong_k \subseteq \cong_{k-1}$ (induction hypothesis) implies (since this hypothesis means that \cong_{k-1} is a partition of \cong_k which means that each class of \cong_{k-1} is a union of disjoint classes of \cong_k) that $Prob(p \rightsquigarrow C) = Prob(q \rightsquigarrow C)$ for all $C \in \Omega/\cong_{k-1}$ which means that (iii) of the definition of \cong_k is verified. \square

Proposition 1 (characterization of \cong) For finite graphs, the following equality holds:

$$\cong = \bigcap_{k \geq 0} \cong_k$$

Let \cong' denotes the equivalence relation $\bigcap_{k \geq 0} \cong_k$. We will show that $\cong = \cong'$ by showing that

1. \cong' verifies the property equation 4, and that
2. $\cong \subseteq \cong'$ (i.e. $p \cong q$ implies $p \cong' q$).

We will deduce then that $\cong = \cong'$, since \cong is defined to be the largest equivalence relation verifying equation 4.

Proof of proposition 1

- (1) Since $p \cong_{k+1} q$ for all k , we have if $p \xrightarrow{t} p'$ then, for each k , $\exists q'_k$ such that $q \xrightarrow{t} q'_k$ and $p' \cong_k q'_k$. Since, we assumed, that the derivative set $\{q'; q \xrightarrow{t} q'\}$ is finite, we deduce that for some q' , $q \xrightarrow{t} q'$ and $p' \cong_k q'$ for infinitely many k . This implies $p' \cong_k q'$ for all k , since the relations \cong_k are decreasing in k . Thus (i) of equation 4 is proved, and (ii) is similar. It remains to show that $\forall C \in \Omega/\cong'$, $Prob(p \rightsquigarrow C) = Prob(q \rightsquigarrow C)$. That is if for all k , for all $C \in \Omega/\cong_k$, we have $p \cong_k q$ implies $Prob(p \rightsquigarrow C) = Prob(q \rightsquigarrow C)$ then $p \cong_k q$, for all k , implies $\forall C \in \Omega/(\bigcap_k \cong_k)$: $Prob(p \rightsquigarrow C) = Prob(q \rightsquigarrow C)$. This holds because the sequence $(\cong_k)_{k \geq 0}$ is decreasing (see property 2.1).
- (2) we must show that: if $p \cong q$ then $\forall k \geq 0, p \cong_k q$. This holds for $k = 0$. By induction on k , if we suppose that $(p \cong q) \Rightarrow (p \cong_k q)$, then we can deduce from the definition of \cong_{k+1} that $p \cong_{k+1} q$. \square

2.2 A computation algorithm

The equivalence \cong_{k+1} can be defined as $F(\cong_k)$ using the function F defined on equivalence relations \equiv over Ω as follows:

$$p F(\equiv) q \text{ if } \begin{cases} (i) \quad \forall t, \text{ whenever } p \xrightarrow{t} p' \text{ then for some } q', q \xrightarrow{t} q' \text{ and } p' \equiv q' \\ (ii) \quad \forall t, \text{ whenever } q \xrightarrow{t} q' \text{ then for some } p', p \xrightarrow{t} p' \text{ and } p' \equiv q' \\ (iii) \quad \forall C \in \Omega/\equiv, Prob(p \rightsquigarrow C) = Prob(q \rightsquigarrow C) \end{cases}$$

The algorithm of Table 1 can be used with this function F to compute the largest equivalence relation verifying the Observational Markovian property given by equation 4.

$$\left. \begin{array}{l} \cong := \Omega \times \Omega \\ \text{Repeat} \\ \quad \cong_{\text{previous}} := \cong \\ \quad \cong := \mathbf{F}(\cong) \\ \text{Until } \cong = \cong_{\text{previous}} \end{array} \right\} \cong = \Phi_{\cong}(\Omega \times \Omega)$$

Table 1: Computing the Observational-Markovian Aggregation

The algorithm of table 1 denoted by Φ , describes computation of the limit of k-equivalences. Initially it is assumed that all states are equivalent to each others. This assumption is represented by the initialization of \cong with $\Omega \times \Omega$. In general the algorithm can be started with another initial equivalence which is then represented as a parameter of Φ . Moreover this algorithm is parameterized by the function F . The same algorithm may be used to compute the observational aggregation (respectively the Markovian aggregation) by taking only clauses (i) and (ii) (respectively only clause (iii)) in the definition of function F . To distinguish all these algorithms, we use the associated equivalence in subscript of Φ and the initial equivalence as a parameter.

Definition 3 (quotient graph associated with \cong) *The quotient graph $\langle \Omega/\cong, L_{\cong}, \Delta_{\cong} \rangle$ is defined by:*

- $\Omega/\cong = \{C_i, i = 1..n\}$ is the state space quotient. The new state partition covers the initial state space: $\bigcup_{i=1..n} C_i = \Omega$ and the new states are pairwise disjoint: $C_i \cap C_j = \emptyset$ for distinct i, j .
- $L_{\cong} \subseteq \mathcal{P}(L)$, the powerset of L , is the set of alphabet. An element of L_{\cong} is a set of transitions identifiers representing all transitions that relate a class to another.
- $\Delta_{\cong} \subseteq \Omega/\cong \times L_{\cong} \times [0, 1] \times \Omega/\cong$ is the transition relation family. An element (C, A, Υ, C') of Δ_{\cong} is again denoted $C \xrightarrow{A, \Upsilon} C'$. Where $A = \{a \in L : \forall p \in C, \exists p' \in C' : p \xrightarrow{a} p'\}$ and $\Upsilon = \text{Prob}(C \rightsquigarrow C') = \sum_{p' \in C'} \text{Prob}(p \rightsquigarrow p') = \sum_{p' \in C'} \sum_{p \in C} \nu_{p \rightarrow p'} \nu$ for any $p \in C$.

The definition of set A and value Υ are valid (i.e. A really exists and Υ does not depend on the state p which is used in its definition) since the following statements hold for the computed partition:

- members of a given class must have the same possible transitions leading to the same classes;
- members of a given class must have equal branching probabilities to other classes.

Remark 2.1 *It is not correct to serialize observational aggregation computation (Φ_{\cong}) with Markovian aggregation (Φ_{\equiv}) to compute the Observational-Markovian aggregation (Φ_{\approx}). In Other words: $\Phi_{\cong}(\Omega \times \Omega) \neq \Phi_{\approx}(\Phi_{\equiv}(\Omega \times \Omega))$ and $\Phi_{\cong}(\Omega \times \Omega) \neq \Phi_{\equiv}(\Phi_{\approx}(\Omega \times \Omega))$. For instance, the figure 2 depicts counter-example. $\Omega = \{1, 2, 3, 4, 5\}$, $\Phi_{\cong}(\Omega \times \Omega) = \{\{1, 2\}, \{3, 4\}, \{5\}\}$, $\Phi_{\equiv}(\Phi_{\approx}(\Omega \times \Omega)) = \{\{1, 2\}, \{3\}, \{4\}, \{5\}\}$ which is different from $\Phi_{\cong}(\Omega \times \Omega) = \{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}\}$.*

2.3 Example

In practice, we can be interested in the behavior of the described system only with respect to some subset of transitions: $\Omega_{\text{small}} = \{t_1, \dots, t_n\} \subseteq \Omega$. A reduced system can be obtained by renaming all transitions that do not belong to Ω_{small} into a unique identifier different from those the user wants to distinguish: $t \notin \Omega_{\text{small}}$.

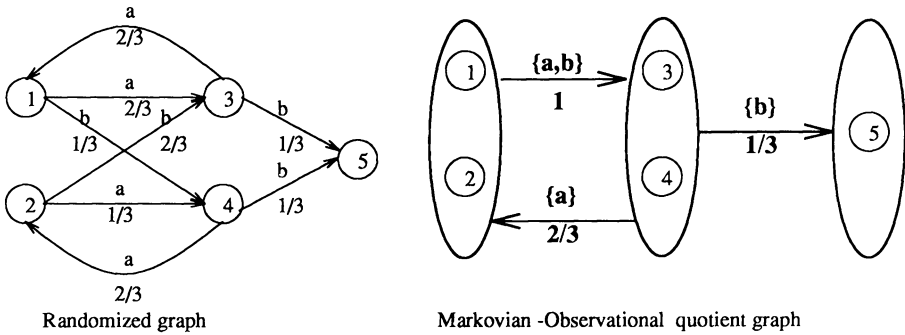


Figure 3: Example of quotient graph

For example the Graph of Fig. 4 is minimal and then cannot be reduced (that is it is equal to its quotient). But if we are interested only in transition *a*, we can rename *b* and *c* into a unique identifier, let's say *d* for example. The graph can then be reduced as depicted by the figure. The transition $C1 \xrightarrow{\{a,d\}} C2$ means that:

- the system can go from state *C1* to state *C2* by firing transition *a* **AND**
- the system can go from state *C1* to state *C2* by firing transition one of *b* **OR** *c*. We don't know if both are possible because the user decided to confuse them.

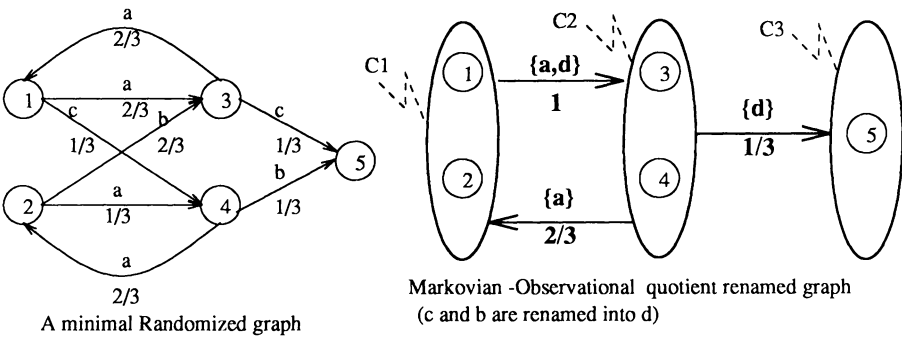


Figure 4: Example of quotient renamed graph

2.4 Extension to timing features

The previous equivalences can be refined using the mean time parameter. The mean time from a state, *p*, to a set of states, *C*, can be defined as the weighted sum, Θ , of the mean

times, θ , associated with any transition from state p to set C . Formally:

$$\Theta = \frac{\sum_{p' \in C, p \xrightarrow{a, \nu, \theta} p'} \nu * \theta}{\sum_{p' \in C, p \xrightarrow{a, \nu, \theta} p'} \nu}$$

The refined equivalence is then defined by adding a fourth field in the definition of equation 4. The timed observational-Markovian equivalence is then defined as the largest equivalence relation over the state space of a randomized graph as the largest equivalence relation \simeq verifying the following property:

$$p \simeq q \text{ implies } \left\{ \begin{array}{l} (i) \quad \forall t, \text{ whenever } p \xrightarrow{t} p' \text{ then for some } q', q \xrightarrow{t} q' \text{ and } p' \simeq q' \\ (ii) \quad \forall t, \text{ whenever } q \xrightarrow{t} q' \text{ then for some } p', p \xrightarrow{t} p' \text{ and } p' \simeq q' \\ (iii) \quad \forall C \in \Omega / \simeq, \text{ Prob}(p \rightsquigarrow C) = \text{Prob}(q \rightsquigarrow C) \\ (iv) \quad \forall C \in \Omega / \simeq, \frac{\sum_{p' \in C, p \xrightarrow{a, \nu, \theta} p'} \nu * \theta}{\sum_{p' \in C, p \xrightarrow{a, \nu, \theta} p'} \nu} = \frac{\sum_{p' \in C, q \xrightarrow{a, \nu, \theta} p'} \nu * \theta}{\sum_{p' \in C, q \xrightarrow{a, \nu, \theta} p'} \nu} \end{array} \right. \quad (5)$$

This equivalence may be characterized as the intersection of k -equivalences similarly to definition 2 and proposition 1.

Definition 4 (timed probabilized k -equivalences)

- $p \simeq_0 q$ is always true

$$\bullet p \simeq_{k+1} q \equiv_{def} \left\{ \begin{array}{l} (i) \quad \forall t, \text{ whenever } p \xrightarrow{t} p' \text{ then for some } q', q \xrightarrow{t} q' \text{ and } p' \simeq_k q' \\ (ii) \quad \forall t, \text{ whenever } q \xrightarrow{t} q' \text{ then for some } p', p \xrightarrow{t} p' \text{ and } p' \simeq_k q' \\ (iii) \quad \forall C \in \Omega / \simeq_k, \text{ Prob}(p \rightsquigarrow C) = \text{Prob}(q \rightsquigarrow C) \\ (iv) \quad \forall C \in \Omega / \simeq_k, \frac{\sum_{p' \in C, p \xrightarrow{a, \nu, \theta} p'} \nu * \theta}{\sum_{p' \in C, p \xrightarrow{a, \nu, \theta} p'} \nu} = \frac{\sum_{p' \in C, q \xrightarrow{a, \nu, \theta} p'} \nu * \theta}{\sum_{p' \in C, q \xrightarrow{a, \nu, \theta} p'} \nu} \end{array} \right.$$

Proposition 2 (characterization of \simeq) For image-finite graphs, the following equality holds:

$$\simeq = \bigcap_{k \geq 0} \simeq_k$$

The quotient graph associated with this equivalence is slightly different from the one associated with the untimed observational-Markovian equivalence. In fact, the only new feature is the time parameter which must be added to the transition label. Transitions of this quotient graph are then of the form: $C \xrightarrow{A, \Upsilon, \Theta} C'$. Where A and Υ have the same meaning as in the untimed definition and Θ is the weighted mean time as defined at the beginning of this section. The computation algorithm is obtained by extension of the untimed algorithm by adding the clause (iv) to the definition of the function F used by this algorithm. For implementation purpose, this clause can be simplified by removing the lower parts (i.e. denominators) of the compared fractions without changing the computed equivalence since these values denote in fact the probability features that must be equal according to clause (iii) of the same definition.

For the special case when the randomized state graph is isomorphic to a Markov chain (this case can occur if all the time distributions associated to transitions of the Stochastic

Timed Petri Net are of only one of the following types: exponential, geometric and discrete with a zero occurrence time), the timed and the untimed observational-Markovian equivalences coincide.

3 EXAMPLE: PARTIALLY CONCURRENT TASKS

In this section we consider the example of two tasks (A and B) which execute asynchronously each one two actions (first a1 then a2 for A; first b1 then b2 for B) and are then synchronized by either of transitions s1 or s2.

This behavior is modeled by a stochastic Petri Net (figure 3) using exponential temporal distributions defined on $[0, \infty)$ by $f(x) = \lambda e^{-\lambda x}$. With the following values: for a1 (or b1) $\lambda = 1$;for a2 (or b2) $\lambda = 2$; for s1 (or s2) $\lambda = \frac{1}{2}$.

An associated quantified reachability graph is generated according to the technique of [Atamna, 1994]. This graph is depicted by figure 3. Transition $s_i \xrightarrow{a,p,t} s_j$ of this graph means: from state s_i action "a" can be executed with probability "p" at the mean time "t" and leads to state s_j .

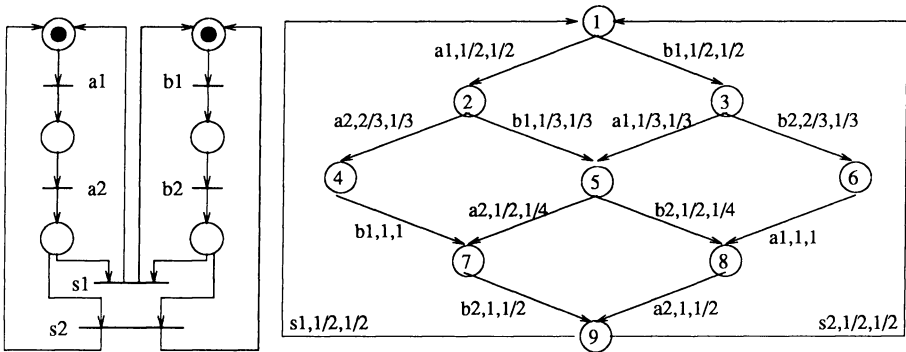


Figure 5: Stochastic Petri Net and its associated quantified graph

Whenever we consider properties where A and B have not to be distinguished, both actions "a" and "b" can be renamed into the same new action, say "c". Applying the (probabilized and timed) projection to this graph provides smaller graph than the initial one. Using the reduced graph can then be more convenient to quantify or to verify these properties.

For example we can be interested in computing the probability that the modeled system executes without concurrency (i.e. every time execution of a given task is started, then this task must be achieved before the starting of the other task: only sequences of the form a1.a2.b1.b2 or b1.b2.a1.a2 are accepted). This probability can be easily computed on the reduced graph by considering the unique path corresponding to the sequence c1.c2.c1.c2. With c1 (resp. c2), being again the first (resp. second) action of either task A or B. Computing this value on the initial graph would proceed as if the behaviors were to be distinguished and would explore paths for both A and B. The user has to identify in total

10 transitions among 14 and to make 16 multiplications (every then) and 3 additions (every or):

$$\begin{aligned}
 &1 \xrightarrow{a_{1,1/2}} 2 \text{ then } 2 \xrightarrow{a_{2,2/3}} 4 \text{ then } 4 \xrightarrow{b_{1,1}} 7 \text{ then } 7 \xrightarrow{b_{2,1}} 9 \text{ then } 9 \xrightarrow{s_{1,1/2}} 1 \text{ or} \\
 &1 \xrightarrow{a_{1,1/2}} 2 \text{ then } 2 \xrightarrow{a_{2,2/3}} 4 \text{ then } 4 \xrightarrow{b_{1,1}} 7 \text{ then } 7 \xrightarrow{b_{2,1}} 9 \text{ then } 9 \xrightarrow{s_{2,1/2}} 1 \text{ or} \\
 &1 \xrightarrow{b_{1,1/2}} 3 \text{ then } 3 \xrightarrow{b_{2,2/3}} 6, \text{ then } 6 \xrightarrow{a_{1,1}} 8 \text{ then } 8 \xrightarrow{a_{2,1}} 9 \text{ then } 9 \xrightarrow{s_{1,1/2}} 1 \text{ or} \\
 &1 \xrightarrow{b_{1,1/2}} 3 \text{ then } 3 \xrightarrow{b_{2,2/3}} 6, \text{ then } 6 \xrightarrow{a_{1,1}} 8 \text{ then } 8 \xrightarrow{a_{2,1}} 9 \text{ then } 9 \xrightarrow{s_{2,1/2}} 1.
 \end{aligned}$$

On the reduced graph (figure 3) this probability is computed by considering only 5 transitions (among 7) and by making 4 multiplications:

$$\begin{aligned}
 &C1 \xrightarrow{\{c1\},1} C2 \text{ then } C2 \xrightarrow{\{c2\},2/3} C4 \text{ then } C4 \xrightarrow{\{c1\},1} C5 \text{ then } C5 \xrightarrow{\{c2\},1} C6 \text{ then} \\
 &C6 \xrightarrow{\{s1,s2\},1} C1 \text{ and is: } 1 * 2/3 * 1 * 1 * 1 = 2/3 \text{ which is the same as} \\
 &\underbrace{1/2 * 2/3 * 1 * 1 * 1/2 + 1/2 * 2/3 * 1 * 1 * 1/2}_{\text{probability that the property holds for task A}} + \underbrace{1/2 * 2/3 * 1 * 1 * 1/2 + 1/2 * 2/3 * 1 * 1 * 1/2}_{\text{probability that the property holds for task B}}
 \end{aligned}$$

We can also compute the probability of the maximal concurrency (that is two action of the same task are always separated by an action of the other task.) by considering $C1 \xrightarrow{\{c1\},1} C2 \text{ then } C2 \xrightarrow{\{c1\},1/3} C3 \text{ then } C3 \xrightarrow{\{c2\},1} C5 \text{ then } C5 \xrightarrow{\{c2\},1} C6 \text{ then } C6 \xrightarrow{\{s1,s2\},1} C1$ and get: $1 * 1/3 * 1 * 1 * 1 = 1/3$ which equals the expected value, $1 - 2/3$, considering the complementarity of this propriety with the previously computed one.

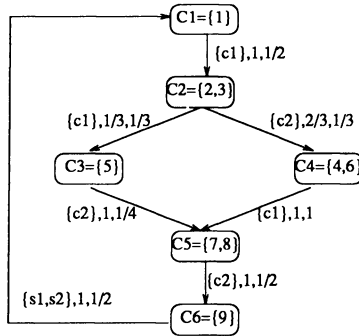


Figure 6: Reduced graph obtained by projection after renaming a and b into c

In practice, much more results can be computed using only the reduced graph. Indeed, we can obtain both the transient and the steady state solution of the aggregated stochastic process by only using the own informations of the quotient graph (stochastic and topological information), and then we can directly obtain most of the performance criteria relatives to the original graph.

Furthermore, the previous equivalences can be refined by adding another clause to guarantee that all the equivalent states of a class have also the same input probabilities. In this case, all the equivalent states of a class will then have the same probabilities in both transient and steady state solution. And then we can say that this aggregation have

been made without any loss of information (we can directly obtain all the performance criteria of the original graph by only using the own information of the quotient graph).

4 CONCLUSION

In this paper Markovian aggregation and Milner's k-equivalences were grouped and used to define the Markovian-Observational equivalence. This allowed a simple algorithm to be proposed to reduce randomized state graphs w.r.t. both observational and probabilistic (and mean time) criteria. We have considered that all actions are observable, and showed how renaming actions can be used to contribute to more reduce the input graph. Using the example of the synchronization of two tasks, we have shown the utilization of the reduced graph to compute or to verify behavioral requirements.

This work constitutes a first attempt to propose a framework to reduce transition graphs labeled with probabilities and times and to provide a methodology for an automatic implementation of the strong condition of the Markovian aggregation. Indeed, the Markovian aggregation technique provides a method to verify that a given partition of states into classes lead to Markovian process.

This work will be continued and refined with more probabilistic and timed features (for example by dealing with the quasi-lumping techniques [Courtois, 1986]). We also aim to consider the problem of reduction with unobservable actions.

Acknowledgment: This work has been developed within the project CNET-CNRS CESAME and partially supported under grant 921B178 of CNET-FRANCE-TELECOM.

References

- [Atamna, 1994] Atamna, Y. (1994). *Réseaux de Petri Temporisés Stochastiques Classiques et Bien Formés: Définition, Analyse et Application aux Systèmes Distribués Temps Réel*. PhD thesis, Université Paul Sabatier, Toulouse III.
- [Atamna, 1995] Atamna, Y. and Juanole, G. (1995). Methodology for obtaining abstract views of state graphs labeled with probabilities and times. an example of application to a communication protocol. In *MASCOTS, the 3rd International Workshop on Modelling Analysis and Simulation of Computer and Telecommunication Systems*, IEEE-CS Press, Durham, North Carolina, USA.
- [Christoff, 1991] Christoff, L. and Christoff, I. (1991). Efficient algorithms for verification of equivalences for probabilistic processes. In Larsen, K. and Skou, A., editors, *Proc. 3rd International Workshop CAV'91*, number 575 in Lecture Notes in Computer Science. Springer-Verlag.
- [Courtois, 1986] Courtois, P. J. and Semal, P. (1986). Computable Bounds on conditional steady-state probabilities in large Markov chain and Queuing models. *IEEE Journal Selected Areas in Communication, SAC*, 4(6):926-937.

- [Drira, 1993] Drira, K. and Azéma, P. (1993). Verifying communication protocols via testing-projection. In M. Nivat, C. Rattray, T. R. and Scollo, G., editors, *Proceedings of the Third International Conference on Algebraic Methodology and Software Technology, University of Twente, The Netherlands, 21-25 June 1993*, Workshops in Computing, pages 255–264, London. Springer-Verlag.
- [Hansson, 1990] Hansson, H. and Jonsson, B. (1990). A Calculus for Communicating Systems with Time and Probabilities. In *Proc. Real-time systems symposium*, pages 278–287, Lake Buena Vista, Florida. IEEE Computer Society Press.
- [Hennessy, 1985] Hennessy, M. and Milner, R. (1985). Algebraic Laws for Nondeterminism and Concurrency. *Journal of the Association for Computing Machinery*, 32(1):137–161.
- [Howard, 1971] Howard, R. A. (1971). *Dynamic Probabilistic Systems, Volume 2: Semi-Markov and Decision Processes*. Wiley, J and Sons, INC.
- [Juanole, 1991] Juanole, G. and Atamna, Y. (1991). Dealing with Arbitrary time Distributions with the Stochastic Timed Petri Net model - Applications to Queueing systems. In *Fourth International Workshop on Petri Nets and Performance Models*, IEEE-CS Press, Melbourne, Australia.
- [Kemeny, 1960] Kemeny, J. G. and Snell, J. L. (1960). *Finite Markov Chains*. Nostrand, V and Princeton, N. J.
- [Larsen, 1991] Larsen, K. and Skou, A. (1991). Bisimulation through Probabilistic Testing. *Information and Computation*, 94:1–28.
- [Milner, 1980] Milner, R. (1980). A Calculus of Communicating Systems. In *Lecture Notes in Computer Science*, volume 92. Springer-Verlag, Berlin Heidelberg.
- [Park, 1981] Park, D. (1981). Concurrency and automata on infinite sequences. In *Lecture Notes in Computer Science*, volume 104. Springer-Verlag, Berlin Heidelberg.
- [Simon, 1984] Simon, S. and Shankar, A. (1984). Protocol verification via projection. *IEEE Transactions on Software Engineering*, 10(4):325–342.
- [Van Glabbeek, 1990] van Glabbeek, R. (1990). The linear Time-Branching Time Spectrum . In *CONCUR '90, Theories of Concurrency: Unification and Extension*, volume 458, Amsterdam, The Netherlands. Lecture Notes in Computer Science, Springer-Verlag.