Quantitative Relations and Approximate Process Equivalences

Alessandra Di Pierro[1]*, Chris Hankin[2]**, and Herbert Wiklicky[2]

[1] Dipartimento di Informatica, Universitá di Pisa, Italy
[2] Department of Computing, Imperial College London, UK

Abstract. We introduce a characterisation of probabilistic transition systems (PTS) in terms of linear operators on some suitably defined vector space representing the set of states. Various notions of process equivalences can then be re-formulated as abstract linear operators related to the concrete PTS semantics via a probabilistic abstract interpretation. These process equivalences can be turned into corresponding approximate notions by identifying processes whose abstract operators "differ" by a given quantity, which can be calculated as the norm of the difference operator. We argue that this number can be given a statistical interpretation in terms of the tests needed to distinguish two behaviours.

1 Introduction

We study the notion of *relation* on a set X in terms of linear operators on a space representing the elements in X. In this setting classical relations corresponds to 0/1 matrices. By considering matrices with generic (numerical) entries, we generalise the classical notion by introducing *quantitative relations*. We will concentrate on a special type of quantitative relations, namely probabilistic transition relations. These represent a central notion in probabilistic process algebra [19], where process semantics and thus the various process equivalences are defined in terms of probabilistic transition systems (PTS).

We introduce a technique for defining approximated versions of various process equivalences, which exploits the operator algebraic view of quantitative relations. The fact that these quantities correspond in a PTS to probabilities allows for a statistical interpretation of the approximation according to the "button-pushing experiments" view of process semantics [22, 31].

The technique is based on the definition of a PTS as a continuous linear operator on a Hilbert space built out of the states and actions. Process equivalences are special linear operators which correspond to some probabilistic abstractions of the PTS semantics. By using some appropriate operator norm we are then able to quantify equivalences, and use the resulting measure ε to define corresponding notions of approximate equivalences. These ε-relations are no longer equivalence relations but instead they approximate equivalence relations.

* Supported by Progetto MEFISTO (Metodi Formali per la Sicurezza e il Tempo).
** Partly funded by the EU FET open project SecSafe.

We will illustrate our approach on two process semantics, namely graph isomorphism and a generic notion of probabilistic bisimulation, which we will characterise by using the Probabilistic Abstract Interpretation framework introduced in [13, 14]. The possibility of reasoning in terms of a non-exact semantics is important for program analysis, where it is often more realistic to consider a margin of tolerance in the identification of two processes. For example, in the area of security, approximate versions of process equivalences can be used to define security properties which reflect more closely the various security problems which occur in practice. For the approximate version of bisimulation, which we call ε-bisimulation, we will mention possible applications in this area. This approach has been adopted in [12, 11], where an approximate notion of observational equivalence is considered to address the problem of confidentiality.

2 Quantitative Relations

Standard models in semantics are usually based on a *qualitative* concept of a relation $R \subseteq X \times X$, which states whether two elements are related or not. We are concerned here with *quantitative* (more precisely probabilistic) relations. Such relations not only specify which elements in X are related, but also how "strong" this relation is. As an example, probabilistic transition relations are quantitative relations which specify how likely it is that one state is reachable from another. We begin with an investigation of the general notion of quantitative relation, which we characterise as a linear operator; we then apply these general results to the special case of probabilistic transition relations, which are at the base of the process equivalences we will study in the following.

Definition 1. (i) *A* quantitative *or* weighted relation *R over a space X with weights in $\mathbb{W}$ is a subset $R \subseteq X \times \mathbb{W} \times X$.*

(ii) *A* labelled quantitative relation *L is a subset $L \subseteq X \times A \times \mathbb{W} \times X$, where A is a set of labels.*

(iii) *A* probabilistic relation *P is a quantitative relation with $\mathbb{W} = [0, 1]$, i.e. $P \subseteq X \times [0, 1] \times X$, where for each $x \in X$ the function $\mu_x : X \mapsto [0, 1]$ defined by $\mu_x(y) = p$ for $(x, p, y) \in P$ is a distribution, i.e. for a fixed $x \in X$: $\sum_{y \in X} \mu_x(y) = \sum_{(x,p,y) \in P} p = 1$.*

We will consider here only quantitative relations over *countable* sets X and *finite* sets of labels A. Furthermore we will assume complex weights, i.e. $\mathbb{W} = \mathbb{C}$, as we can embed the other common weight sets, e.g. $\mathbb{Z}, \ldots, \mathbb{R}$, easily in $\mathbb{C}$.

Note that for numerical weights — i.e. for $\mathbb{W}$ a ring, field, etc. — we can interpret $R \subseteq X \times \mathbb{W} \times X$ as a function $R : X \times X \to \mathbb{W}$ by adding all the weights associated to the same pair $(x, y) \in X \times X$, i.e. $R(x, y) = \sum_{(x,w,y) \in R} w$.

2.1 Linear Representations

Qualitative as well as quantitative relations have a simple representation as linear operators. In order to define the matrix associated to a relation on a set X, we first have to lift X to a vector space.

Definition 2. *The vector space $\mathcal{V}(X)$ over a set X is the space of formal linear combinations of elements in X with coefficients in some field $\mathbb{W}$ (e.g. $\mathbb{W} = \mathbb{C}$) which are represented by sequences of elements in $\mathbb{W}$ indexed by elements in X:*

$$\mathcal{V}(X) = \{(c_x)_{x \in X} \mid c_x \in \mathbb{W}\} .$$

We associate to each relation $R \subseteq X \times X$ a 0/1-matrix, i.e. a linear operator $\mathbf{M}(R)$ on $\mathcal{V}(X)$ defined by:

$$(\mathbf{M}(R))_{xy} = \begin{cases} 1 \text{ iff } (x, y) \in R \\ 0 \text{ otherwise} \end{cases}$$

where $x, y \in X$, and $(\mathbf{M}(R))_{xy}$ denotes the entry in column x and row y in the matrix representing $\mathbf{M}(R)$. Analogously, the matrix representing a quantitative relation $R \subseteq X \times \mathbb{W} \times X$ is defined by:

$$(\mathbf{M}(R))_{xy} = \begin{cases} w \text{ iff } (x, w, y) \in R \\ 0 \ \text{ otherwise} \end{cases}$$

Note that these definitions rely on the interpretation of (numerical) quantitative relations as functions mentioned above. For probabilistic relations, where $\mathbb{W} = [0, 1]$, we obtain a *stochastic matrix*, that is a positive matrix where the entries in each row sum up to one.

For finite sets X the representation of (quantitative) relations as linear operators on $\mathcal{V}(X) \simeq \mathbb{C}^n$ is rather straightforward: since all finite dimensional vector spaces are isomorphic to the n-dimensional complex vector space $\mathbb{C}^n$ for some $n < \infty$, their topological structure is unique [18, 1.22] and every linear operator is automatically continuous. For infinite (countable) sets, however, the algebra of infinite matrices which we obtain this way is topologically "unstable". The algebra of infinite matrices has no universal topological structure and the notions of linearity and continuity do not coincide. It is therefore difficult, for example, to define the limit of a sequence of infinite matrices in a general way. In [15] Di Pierro and Wiklicky address this problem by concentrating on relations which can be represented as elements of a C*-algebra, or concretely as elements in $\mathcal{B}(\ell^2)$, i.e. the algebra of bounded, and therefore continuous linear operators on the standard Hilbert space $\ell^2(X) \subseteq \mathcal{V}(X)$. This is the space of infinite vectors:

$$\ell^2 = \ell^2(X) = \{(x_i)_{i \in X} \mid x_i \in \mathbb{C} : \sum_{i \in X} |x_i|^2 < \infty\}.$$

The algebraic structure of a C*-algebra allows for exactly one norm topology and thus offers the same advantages as the linear algebra of finite dimensional matrices. A formal justification for this framework is given in [15]. We just mention here that the representation of (probabilistic transition) relations as operators on $\ell^2(X)$ — and not for example on $\ell^1(S)$ (which a priori might seem to be a more appropriate structure, e.g. [20]) — allows us to treat "computational states" and "observables" as elements of the same space (as Hilbert spaces are *self-dual*). Furthermore, this approach is consistent with the well established study of *(countable) infinite graphs* via their adjacency operator as an element in $\mathcal{B}(\ell^2)$, e.g. [23].

2.2 Probabilistic Transition Relations

A labelled transition system specifies a class of sequential processes $\mathcal{P}$ on which binary predicates $\xrightarrow{a}$ are defined for each action a a process is capable to perform. Probabilistic Transition Systems (PTS) are labelled transition systems with a probabilistic branching: a process p can be in a relation $\xrightarrow{a}$ with any p' in a set S of possible successors with a given probability $\mu(p')$ such that μ forms a distribution over the set S [19, 21].

Given a countable set S, we call a function $\pi : S \mapsto [0,1]$ a *distribution* on S iff $\sum_{s \in S} \pi(s) = 1$. We denote by $Dist(S)$ the set of all distributions on S. Every distribution corresponds to a vector in the vector space $\mathcal{V}(S)$. Furthermore as $x^2 \leq x$ for $x \in [0,1]$ we have $\sum_{s \in S} \pi(s)^2 \leq \sum_{s \in S} \pi(s) = 1$, i.e. every distribution corresponds to a vector in $\ell^2(S) \subseteq \mathcal{V}(S)$.

Given an equivalence relation $\sim$ on S and a distribution π on S, the *lifting* of π to the set of equivalence classes of $\sim$ in S, $S/_\sim$, is defined for each equivalence class $[s] \in S/_\sim$ by $\pi([s]) = \sum_{s' \in [s]} \pi(s')$. It is straightforward to show that this is indeed a distribution on $S/_\sim$ (e.g. [19, Def 1 & Thm 1]). We write $\pi \sim \varrho$ if the lifting of π and ϱ coincide.

Definition 3. *A* probabilistic transition system *is a tuple* $(S, A, \longrightarrow, \pi_0)$*, where:*

- S *is a non-empty, countable set of* states*,*
- A *is a non-empty, finite set of* actions*,*
- $\longrightarrow \subseteq S \times A \times Dist(S)$ *is a* transition relation*, and*
- $\pi_0 \in Dist(S)$ *is an* initial distribution *on* S*.*

For $s \in S$, $\alpha \in A$ and $\pi \in Dist(S)$ we write $s \xrightarrow{\alpha} \pi$ for $(s, \alpha, \pi) \in \longrightarrow$. By $s \xrightarrow{\alpha}_{\pi(t)} t$ we denote the transition to individual states t with probability $\pi(t)$.

The above definition of a PTS allows for fully probabilistic as well as non-deterministic transitions as there might be more than one distribution associated to a state s and an action α. In this paper we will concentrate on fully probabilistic models where a non-deterministic choice never occurs.

Definition 4. *Given a probabilistic transition system* $X = (S, A, \longrightarrow, \pi_0)$*, we define its* matrix *or* operator representation $\mathbf{X} = (\mathbf{M}(X), \mathbf{M}(\pi_0))$ *as the direct sum of the operator representations of the transition relations for each* $\alpha \in A$*:*

$$\mathbf{M}(X) = \bigoplus_{\alpha \in A} \mathbf{M}(\xrightarrow{\alpha}),$$

and $|A|$ *copies of the vector representing* π_0*:* $\mathbf{M}(\pi_0) = \bigoplus_{\alpha \in A} \boldsymbol{\pi}_0$.

In the following we will denote $\mathbf{M}(\xrightarrow{\alpha})$ by $\mathbf{M}_\alpha$.

Given a set $\{\mathbf{M}_i\}_{i=1}^k$ of $n_i \times m_i$ matrices, then the direct sum of these matrices is given by the $(\sum_{i=1}^k n_i) \times (\sum_{i=1}^k m_i)$ matrix:

$$\mathbf{M} = \bigoplus_i \mathbf{M}_i = \begin{pmatrix} \mathbf{M}_1 & 0 & 0 \dots & 0 \\ 0 & \mathbf{M}_2 & 0 \dots & 0 \\ \vdots & \vdots & \vdots \ddots & \vdots \\ 0 & 0 & 0 \dots & \mathbf{M}_k \end{pmatrix}$$

Distributions are represented by vectors in the vector space $\ell^2(S) \oplus \ldots \oplus \ell^2(S) = (\ell^2)^{|A|} \subseteq \mathcal{V}(S)^{|A|}$. The matrix $\mathbf{M}(X)$ represents a linear operator on this space.

It is easy to see that starting with $\mathbf{M}(\pi_0)$ and applying $\mathbf{M}(X)$ repeatedly for n steps we get the distributions corresponding to the n-step closure of $\longrightarrow$ (by summing up the factors in the direct sum). More precisely:

- Take an initial $\pi_0 \in Dist(S)$ and represent it as a vector $\mathbf{M}(\pi_0) \in \mathcal{V}(S)$
- Combine $|A|$ copies of $\mathbf{M}(\pi_0)$ to obtain $\mathbf{M}(\pi_0)^{|\mathcal{A}|} = \bigoplus_{\alpha \in \mathcal{A}} \mathbf{M}(\pi_0)$.
- Apply $\mathbf{M}(X) = \bigoplus_{\alpha \in \mathcal{A}} \mathbf{M}_\alpha$ to this vector.
- Obtain $(\bigoplus_{\alpha \in \mathcal{A}} \mathbf{M}_\alpha)(\mathbf{M}(\pi_0)^{|\mathcal{A}|}) = \bigoplus_{\alpha \in \mathcal{A}} \mathbf{M}_\alpha(\mathbf{M}(\pi_0))$.
- Denote the factors by $\mathbf{M}(\pi'_\alpha) = \mathbf{M}_\alpha(\mathbf{M}(\pi_0))$.
- Construct the compactification $\mathbf{M}(\pi_1) = \sum_{\alpha \in \mathcal{A}} \mathbf{M}(\pi'_\alpha)$.
- Restart the iteration process with π_1.

For the sake of simplicity we will denote by $\mathbf{PX}$ the multiplication of a direct sum $\bigoplus_\alpha \mathbf{P}$ of the same matrix $\mathbf{P}$ with the matrix $\mathbf{X} = \bigoplus_\alpha \mathbf{X}_\alpha$. By the properties of the direct sum this is the same as $\bigoplus_\alpha (\mathbf{PX}_\alpha)$.

Given a PTS $X = (S, A, \longrightarrow, \pi_0)$ and a state $p \in S$, we denote by $S_p \subseteq S$ the set of all states reachable from p, by $T(p)$ the transition system induced on the restricted state space S_p, and by $\mathbf{M}(p)$ the matrix representation of $T(p)$.

3 Probabilistic Abstract Interpretation

Probabilistic Abstract Interpretation was introduced in [13, 14] as a probabilistic version of the classical *abstract interpretation* framework by Cousot & Cousot [5, 6]. This framework provides general techniques for the analysis of programs which are based on the construction of *safe* approximations of concrete semantics of programs via the notion of *Galois connection* [7, 25]. Probabilistic abstract interpretation re-casts these techniques in a probabilistic setting, where linear spaces replace the classical order-theoretic based domains, and the notion of *Moore-Penrose pseudo-inverse* of a linear operator replaces the classical notion of a Galois connections. It is thus essentially different from approaches applying classical abstract interpretation to probabilistic domains [24].

By a *probabilistic domain* we mean a space which represents the distributions $Dist(S)$ on the state space S of a PTS, i.e. in our setting the Hilbert space $\ell^2(S)$. For finite state spaces we can identify $\mathcal{V}(S) \simeq \ell^2(S)$.

Definition 5. *Let $\mathcal{C}$ and $\mathcal{D}$ be two probabilistic domains. A* probabilistic abstract interpretation *is a pair of bounded linear operators $\mathbf{A} : \mathcal{C} \to \mathcal{D}$ and $\mathbf{G} : \mathcal{D} \to \mathcal{C}$, between (the concrete domain) $\mathcal{C}$ and (the abstract domain) $\mathcal{D}$, such that $\mathbf{G}$ is the Moore-Penrose pseudo-inverse of $\mathbf{A}$, and vice versa.*

The Moore-Penrose pseudo-inverse is usually considered in the context of so-called *least-square approximations* as it allows the definition of an optimal generalised solution of linear equations. The Moore-Penrose pseudo-inverse of a linear map between two Hilbert spaces is defined as follows (for further details see e.g. [4], or [3]):

Definition 6. *Let $\mathcal{C}$ and $\mathcal{D}$ be two Hilbert spaces and $\mathbf{A} : \mathcal{C} \mapsto \mathcal{D}$ a linear map between them. A linear map $\mathbf{A}^\dagger = \mathbf{G} : \mathcal{D} \mapsto \mathcal{C}$ is the* Moore-Penrose pseudo-inverse *of $\mathbf{A}$ iff*

$$\mathbf{A} \circ \mathbf{G} = \mathbf{P}_A \quad \textit{and} \quad \mathbf{G} \circ \mathbf{A} = \mathbf{P}_G$$

where $\mathbf{P}_A$ and $\mathbf{P}_G$ denote orthogonal projections onto the ranges of $\mathbf{A}$ and $\mathbf{G}$.

A simple method for constructing a probabilistic abstract interpretation which we will use in this paper is as follows: given a linear operator Φ on some Hilbert space $\mathcal{V}$ expressing the probabilistic semantics of a concrete system, and a linear abstraction function $\mathbf{A} : \mathcal{V} \mapsto \mathcal{W}$ from the concrete domain into an abstract domain $\mathcal{W}$, we compute the Moore-Penrose pseudo-inverse $\mathbf{G} = \mathbf{A}^\dagger$ of $\mathbf{A}$. The abstract semantics can then be defined as the linear operator on the abstract domain $\mathcal{W}$:

$$\Psi = \mathbf{A} \circ \Phi \circ \mathbf{G}.$$

Moore-Penrose inverses always exist for operators on finite dimensional vector spaces [3]. For operator algebras, i.e. operators over infinite dimensional Hilbert spaces, the following theorem provides conditions under which the existence of Moore-Penrose inverses is guaranteed [3, Thm 4.24]:

Theorem 1. *An operator $\mathbf{A} : \mathcal{C} \rightarrow \mathcal{D}$ between two Hilbert spaces is Moore-Penrose invertible if and only if it is* normally solvable*, i.e. if its range $\{\mathbf{A}x \mid x \in \mathcal{C}\}$ is closed.*

For the special case of operators $\mathbf{A}$ which are defined via an approximating sequence $(\mathbf{A}_n)_n$ of finite-dimensional operators, we are not only guaranteed that the Moore-Penrose pseudo-inverse exists, but we can also construct it via an approximation sequence provided that the sequence $(\mathbf{A}_n)_n$ and the sequence $(\mathbf{A}_n^*)_n$ of their adjoints converges *strongly* to $\mathbf{A}$ and $\mathbf{A}^*$ [3, Cor 4.34]. In the strong operator topology a sequence of operators $(\mathbf{A}_n)_n$ converges *strongly* if there exists an $\mathbf{A} \in \mathcal{B}(\ell^2)$ such that for all $\boldsymbol{x} \in \ell^2$: $\lim_{n\rightarrow\infty} \|\mathbf{A}_n\boldsymbol{x} - \mathbf{A}\boldsymbol{x}\| = 0$.

Proposition 1. *Let $\mathbf{A} : \mathcal{C} \rightarrow \mathcal{D}$ be an operator between two separable Hilbert spaces. If there is a sequence $\mathbf{A}_n$ of finite dimensional operators with $\sup_n \|\mathbf{A}_n\| < \infty$ and such that $\mathbf{A}_n \rightarrow \mathbf{A}$ and $\mathbf{A}_n^* \rightarrow \mathbf{A}^*$ strongly, then $\mathbf{A}$ is normally solvable and $\mathbf{A}_n^\dagger \rightarrow \mathbf{A}^\dagger$ strongly.*

This construction is sufficient for most cases as it can be shown that the operational or collecting semantics of finitely branching processes can always be approximated in this way [15].

4 Approximate Process Equivalences

In the classical approaches process equivalences are qualitative relations. Alternatively, process equivalences can be seen as a kind of quantitative relations, namely probabilistic relations. One advantage of having a quantity (the probability) attached to a relation is that we can calculate the behavioural difference between two processes and use the resulting quantity to define *approximate* notions of equivalences. The latter weaken strict equivalences by identifying processes whose behaviour is "the same up to ε", ε being the approximation error.

The ε versions of process equivalences are closely related to approaches which aim to distinguish probabilistic processes by statistical testing. A general setting for a statistical interpretation of ε is provided by the concept of *hypothesis testing*, see e.g. [28]. The problem can be stated as follows: given two processes A and B let us assume that one of these is executed as a black-box process X, i.e. we know that either $X = A$ or $X = B$. The idea is to formulate two (exclusive) hypotheses $H_0 : X$ is A and $H_1 : X$ is B. The aim is to determine the probability that either H_0 or H_1 holds based on a number of statistical tests performed on X. The number ε gives us a direct measure for how many tests we have to perform in order to accept H_0 or H_1 with a certain confidence. In essence: the smaller the ε, the more tests we have to perform in order to obtain the same level of confidence.

The details of the exact relation between the number of required tests n to distinguish H_0 and H_1 with a certain confidence α are not easy to be worked out in general, but can in principle be achieved using methods from mathematical statistics. More details for a concrete case — applied to the problem of probabilistic confinement related to the simple notion of process equivalence based on input/output observables — can be found in [12, 11].

Approximate equivalences turn out to be very useful in program analysis where they can be used to define approximate and yet more realistic analyses of programs properties, such as confinement [12, 11, 10], which are directly defined in terms of some process equivalences.

In order to define approximate process equivalences we first look at relations as linear operators; then using an appropriate operator norm we measure the "distance" between relations. In this way we are able to define a relation which is ε-close to the strict (original) equivalence. For the characterisation of equivalence relations as linear operators we use the framework of probabilistic abstract interpretation. In particular, we will show that each equivalence on a given system corresponds to a pair of Moore-Penrose pseudo-inverses which define a probabilistic abstract interpretation of the system.

4.1 Graph Equivalence

To illustrate our basic strategy for approximating process equivalences let us first look at the strongest — in some sense too strong [31, Fig 1] — notion of process equivalence, that is tree equivalence. Following [31, Def 1.3] the graph associated to a process p of a labelled transition system with actions A is a directed graph rooted in p whose edges are labelled by elements in A. Two processes are *tree equivalent* if their associated graphs are isomorphic. Graph isomorphism is defined as follows ([31, Def 1.3,Def 1.4], [17, p2]):

Definition 7. *An* isomorphism *between directed graphs* (V_1, E_1) *and* (V_2, E_2) *is a bijection* $\varphi : V_1 \mapsto V_2$ *such that* $\langle v, w\rangle \in E_1 \Leftrightarrow \langle \varphi(v), \varphi(w)\rangle \in E_2$.

In the usual way, we define the *adjacency operator* $\mathbf{A}(X)$ of a directed graph $X = (V, E)$ as an operator on $\ell^2(V)$ representing the edge-relation E [23]. Then the notion of isomorphism between (finite) graphs can be re-stated in terms of permutation matrices.

An $n \times n$-matrix $\mathbf{P}$ is called a *permutation matrix* if there exists a permutation $\pi : \{1, \ldots, n\} \rightarrow \{1, \ldots, n\}$ such that $\mathbf{P}_{ij} = 1$ iff $j = \pi(i)$ and otherwise $\mathbf{P}_{ij} = 0$. This notion can easily be extended to *permutation operators* for infinite structures.

We denote by $\mathcal{P}(n)$ the set of all $n \times n$ permutation matrices and by $\mathcal{P}(\mathcal{H})$ the set of permutation operators on $\mathcal{H}$; obviously we have $\mathcal{P}(n) = \mathcal{P}(\mathbb{C}^n)$.

Proposition 2. *For any permutation operator $\mathbf{P} \in \mathcal{P}(\mathcal{H})$ the following holds: $\mathbf{P}^{-1} = \mathbf{P}^* = \mathbf{P}^T = \mathbf{P}^\dagger$, i.e.* inverse, adjoint, transpose, *and* pseudo-inverse *of permutation operators coincide.*

We then have the following result [17, Lemma 8.8.1]:

Proposition 3. *Let $X = (V, E_1)$ and $Y = (V, E_2)$ be two directed graphs on the same set of nodes V. Then X and Y are isomorphic if and only if there is a permutation operator $\mathbf{P}$ such that the following holds: $\mathbf{P}^T\mathbf{A}(X)\mathbf{P} = \mathbf{A}(Y)$.*

By using these notions and the operator representation of (probabilistic) transition systems (cf. Definition 4) we can reformulate tree-equivalence of processes as follows.

Proposition 4. *Given the operator representations $\mathbf{X}$ and $\mathbf{Y}$ of two probabilistic transition systems $X = (S, A, \longrightarrow, s_0)$ and $Y = (S', A, \longrightarrow', s'_0)$ with $|S| = |S'|$, then X and Y are tree-equivalent iff there exists $\mathbf{P} \in \mathcal{P}(\ell^2(S)) = \mathcal{P}(\ell^2(S'))$, such that:*

$$\mathbf{P}^T\mathbf{X}\mathbf{P} = \mathbf{Y},$$

i.e. for all $\alpha \in A$ we have $\mathbf{P}^T\mathbf{M}(\xrightarrow{\alpha})\mathbf{P} = \mathbf{M}(\xrightarrow{\alpha}{}')$ and $\mathbf{P}^T\pi_0\mathbf{P} = \pi'_0$.

Therefore, tree equivalence of two systems X and Y corresponds to the existence of an abstraction operator (the operator $\mathbf{P}$) which induces a probabilistic abstract interpretation Y of X.

Approximate Graph Equivalence. In the case where there is no $\mathbf{P}$ which satisfies the property in Proposition 4, i.e. X and Y are definitely not isomorphic, we could still ask how close X and Y are to being isomorphic. The most direct way to define a kind of "isomorphism defect" would be to look at the difference $\mathbf{X} - \mathbf{Y}$ between the operators representing X and Y and then measure in some way, e.g. using a norm, this difference.

Obviously, this is not the idea we are looking for: it is easy to see that the same graph — after enumerating its vertices in a different ways — has different adjacency operators; it would thus have a non-zero "isomorphism defect" with itself. To remedy this we have to allow first for a reordering of vertices before we measure the difference between the operators representing two probabilistic transition systems. This is the underlying idea behind the following definition.

Definition 8. *Let $X = (S, A, \longrightarrow, \pi_0)$ and $Y = (S', A, \longrightarrow', \pi'_0)$ be probabilistic transition systems over the same set of actions A, and let $\mathbf{X}$ and $\mathbf{Y}$ be their operator representations. We say that X and Y are* ε-graph equivalent*, denoted by $X \sim_i^\varepsilon Y$, iff*

$$\inf_{\mathbf{P}\in\mathcal{P}} \|\mathbf{P}^T\mathbf{X}\mathbf{P} - \mathbf{Y}\| = \varepsilon$$

where $\|.\|$ denotes an appropriate norm.

Note that, in the case of finite probabilistic transition systems, for $\varepsilon = 0$ we recover the original notion of (strict) graph equivalence, i.e. $\sim_i = \sim_i^0$.

Proposition 5. *An ε-isomorphism for $\varepsilon = 0$, i.e. $\sim_i^0$, of finite transition systems is an isomorphism.*

We believe that a similar proposition can be stated for infinite PTS's too. However, this would require the development of a more elaborate operator algebraic framework for modelling PTS's than the one presented in this paper, and we refer to [15] for a more detailed treatment of this case.

4.2 Probabilistic Bisimulation Equivalence

The finest process equivalence is bisimulation equivalence [31]. Bisimulation is a relation on processes, i.e. states of a labelled transition system. Alternatively, it can be seen as a relation between the *transition graphs* associated to the processes. The classical notion of bisimulation equivalence for labelled transition systems is as follows, e.g. [31, Def 12]:

Definition 9. *A* bisimulation *is a binary relation $\sim_b$ on states of a labelled transition system satisfying for all $\alpha \in A$:*

$$\begin{array}{l} p \sim_b q \text{ and } p \xrightarrow{\alpha} p' \Rightarrow \exists q' : q \xrightarrow{\alpha} q' \text{ and } p' \sim_b q', \\ p \sim_b q \text{ and } q \xrightarrow{\alpha} q' \Rightarrow \exists p' : p \xrightarrow{\alpha} p' \text{ and } q' \sim_b p'. \end{array}$$

Given two processes p and q, we say that they are *bisimilar* if there exists a bisimulation relation $\sim_b$ such that $p \sim_b q$. Bisimulations are equivalence relations [31, Prop 8.1].

The standard generalisation of this notion to probabilistic transition systems, i.e. *probabilistic bisimulation*, is due to [21]. We will concentrate here on fully probabilistic systems or reactive systems in the terminology of [19]. In this model all states $s \in S$ are deterministic in the sense that for each action $\alpha \in A$, there is only one distribution π such that $s \xrightarrow{\alpha} \pi$.

Definition 10. *[19, Def 4][9, Def 3.2] A* probabilistic bisimulation *is an equivalence relation $\sim_b$ on states of a probabilistic transition system satisfying for all $\alpha \in A$:*

$$p \sim_b q \text{ and } p \xrightarrow{\alpha} \pi \Rightarrow q \xrightarrow{\alpha} \varrho \text{ and } \pi \sim_b \varrho.$$

We now introduce the notion of a classification operator which we will use to define a probabilistic bisimulation equivalence via a probabilistic abstract interpretation. A *classification matrix* or *classification operator* is given by an (infinite) matrix containing only a single non zero entry equal to one in each row, and no column with only zero entries. Classification operators simply represent a classical relation, i.e. is a $0/1$ matrix, which happens to be a (surjective) function from one state space into another.

Classification matrices and operators are thus particular kinds of stochastic matrices and operators. We denote by $\mathcal{C}(n, m)$ the set of all $n \times m$-classification matrices, and by $\mathcal{C}(\mathcal{H}_1, \mathcal{H}_2)$ the set of classification operators; again we have $\mathcal{C}(n, m) = \mathcal{C}(\mathbb{C}^n, \mathbb{C}^m)$.

Obviously, every permutation matrix is also a classification matrix: $\mathcal{P}(n) \subseteq \mathcal{C}(n, n)$, and similarly $\mathcal{P}(\mathcal{H}) \subseteq \mathcal{C}(\mathcal{H}, \mathcal{H})$. Furthermore, the multiplication of two classification operators gives again a classification operator. These properties follow easily from the following correspondence between classification operators and equivalence relations:

Proposition 6. *Let X be a countable set. Then for each equivalence relation $\approx$ on X there exists a classification operator $\mathbf{K} \in \mathcal{C}(\ell^2(X), \ell^2(X/_\approx))$ and vice versa.*

For finite sets with $|X| = n$ and $|X/_\approx| = m$ we get a classification matrix in $\mathcal{C}(n, m)$.

Proposition 7. *The pseudo-inverse of a classification operator $\mathbf{K}$ corresponds to its normalised transpose or adjoint (these operations coincide for real $\mathbf{K}$).*

The *normalisation* operation $\mathcal{N}$ is defined for a matrix $\mathbf{A}$ by $\mathcal{N}(\mathbf{A})_{ij} = \frac{\mathbf{A}_{ij}}{a_j}$ if $a_j = \sum_i \mathbf{A}_{ij} \neq 0$ and $\mathcal{N}(\mathbf{A})_{ij} = 0$ otherwise. Although the classification operator $\mathbf{K}$ represents a classical function, i.e. corresponds to an (infinite) $0/1$-matrix, the pseudo-inverse will in general not be an (infinite) $0/1$-matrix.

It is easy to see that a probabilistic bisimulation equivalence $\sim$ on a PTS $T = (S, A, \rightarrow, \pi_0)$ defines a probabilistic abstract interpretation of T. In fact, by Proposition 6, there is a classification matrix $\mathbf{K} \in \mathcal{C}(\ell^2(S), \ell^2(S')$, for some S' which represents $\sim$. If $\mathbf{M}(T)$ is the operator representation of T then $\mathbf{K}^\dagger\mathbf{M}(T)\mathbf{K}$ is the abstract operator induced by $\mathbf{K}$. Intuitively, this is an operator which abstracts the original system T by encoding only the transitions between equivalence classes instead of the ones between single states.

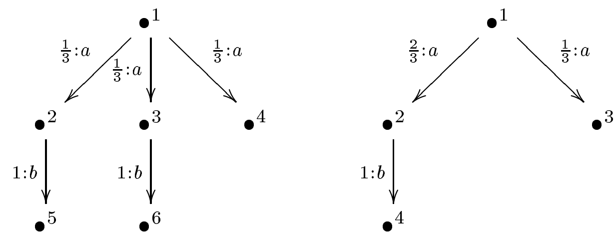
Consider now two processes $p, q \in S$ and their operator representations $\mathbf{M}(p)$ and $\mathbf{M}(q)$. The restrictions of $\mathbf{K}$ to these two sets of nodes, which we call $\mathbf{K}_p$ and $\mathbf{K}_q$, are the abstraction operators for the two processes p and q and allow us to express exactly the condition for the probabilistic bisimilarity of p and q:

Proposition 8. *Given the operator representation $\mathbf{M}(p)$ and $\mathbf{M}(q)$ of two probabilistic processes p and q, then p and q are bisimilar iff there exists a $\mathbf{K}_p \in \mathcal{C}(\ell^2(S_p), \ell^2(S))$ and $\mathbf{K}_q \in \mathcal{C}(\ell^2(S_q), \ell^2(S))$ for some set S such that*

$$\mathbf{K}_p^\dagger\mathbf{M}(p)\mathbf{K}_p = \mathbf{K}_q^\dagger\mathbf{M}(q)\mathbf{K}_q.$$

Corollary 1. *Given the matrix representation* $\mathbf{M}(p)$ *and* $\mathbf{M}(q)$ *of two processes* p *and* q*. Then* p *and* q *are bisimilar, i.e.* $p \sim_b q$*, iff there exists a PTS* x *which is the probabilistic abstract interpretation of both* p *and* q*.*

Example 1. Consider the following two processes A and B from [21, Fig.4]:



The corresponding matrices are:

$$\mathbf{M}(A) = \mathbf{M}_a(A) \oplus \mathbf{M}_b(A) = \begin{pmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \oplus \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

and

$$\mathbf{M}(B) = \mathbf{M}_a(B) \oplus \mathbf{M}_b(B) = \begin{pmatrix} 0 & \frac{2}{3} & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \oplus \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

The classification operators and their pseudo-inverses are given by:

$$\mathbf{K}_A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{K}_A^\dagger = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

and $\mathbf{K}_B$ and $\mathbf{K}_B^\dagger$ are simply 4×4 identity matrices. We then get:

$$\mathbf{K}_A^\dagger \cdot \mathbf{M}_a(A) \cdot \mathbf{K}_A = \mathbf{M}_a(B)$$
$$\mathbf{K}_A^\dagger \cdot \mathbf{M}_b(A) \cdot \mathbf{K}_A = \mathbf{M}_b(B)$$

which shows that A and B are probabilistically bisimilar.

The matrix formulation of (probabilistic) bisimulation makes it also easy to see how graph and bisimulation equivalence are related, as $\mathcal{P}(n) \subset \mathcal{C}(n,n)$ we have:

Proposition 9. *If $p \sim_i q$ then $p \sim_b q$.*

Note that probabilistic bisimulation is only related to a particular kind of probabilistic abstract interpretation: we consider only abstractions which are induced by classification matrices and not by more general ones. The relation between abstract interpretation and (bi)simulation has been recognised before in the classical Galois connection based setting ([8], [27]), but this appears to be the first investigation of such a relation in a probabilistic setting.

Approximate Bisimulation Equivalences. When it is not possible to find a bisimulation equivalence for two processes p and q of a PTS T, we can still identify them although only approximately. In order to do so, we introduce an ε-version of probabilistic bisimilarity. The intuitive idea is to find a classification operator $\mathbf{K}$ which is the closest one to a bisimulation relation in which p and q are equivalent. The difference between the abstract operators induced by $\mathbf{K}$ for the two processes will give us an estimate of the non-bisimilarity degree of p and q.

Definition 11. *Let $T = (S, A, \longrightarrow, \pi_0)$ be a probabilistic transition systems and let p and q be two states in S with operator representations $\mathbf{X}$ and $\mathbf{Y}$. We say that p and q are* ε-bisimilar*, denoted by $p \sim_b^{\varepsilon} q$, iff*

$$\min_{\mathbf{K}_p, \mathbf{K}_q \in \mathcal{C}} \|\mathbf{K}_p^{\dagger}\mathbf{X}\mathbf{K}_p - \mathbf{K}_q^{\dagger}\mathbf{Y}\mathbf{K}_q\| = \varepsilon$$

where $\|.\|$ denotes an appropriate norm.

In determining the "degree of similarity" ε of two processes $\mathbf{X}$ and $\mathbf{Y}$ our aim is to identify two "abstract processes" $\mathbf{K}_p^{\dagger}\mathbf{X}\mathbf{K}_p$ and $\mathbf{K}_q^{\dagger}\mathbf{X}\mathbf{K}_q$ such that their behaviour is most similar. The concrete numerical value of ε depends on the norm we choose and the type of classification operators we consider. In particular, we can strengthen the above definition by restricting the number of "abstract states", i.e. the dimension of $\mathbf{K}_p$ and $\mathbf{K}_q$, in order to obtain an estimation ε relative to only those equivalences with a fixed number of classes.

Note that it is possible to use this definition also to introduce an approximate version of the classical notion of bisimulation. Furthermore, for $\varepsilon = 0$ we recover partially the original notion of strict (probabilistic)bisimulation:

Proposition 10. *An ε-bisimulation for $\varepsilon = 0$, i.e. $\sim_b^0$, is a (probabilistic) bisimulation for finite transition systems.*

For infinite PTS, the same remarks as for Proposition 5 apply.

Example 2. In this example we will use a more "probabilistic" form of PTS which are called *generative* in [26]. In this model the probability distribution on the branching takes into account the internal decision of the process to react to a given action. Thus the transition relation is a subset of $S \times \mathit{Dist}(A \times S)$.

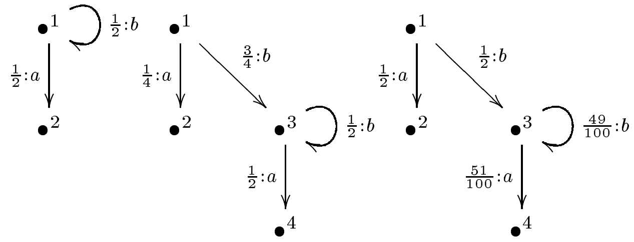
Let us compare the following, obviously somehow "similar", processes:

$$A \equiv \mathit{fix}\ A.b : A +_{\frac{1}{2}} a : \mathbf{0}$$

$$B \equiv a : \mathbf{0} +_{\frac{3}{4}} (\mathit{fix}\ X.b : X +_{\frac{1}{2}} a : \mathbf{0})$$

$$C \equiv a : \mathbf{0} +_{\frac{1}{2}} (\mathit{fix}\ X.b : X +_{\frac{51}{100}} a : \mathbf{0})$$

Their transition graphs are given by:



These processes are not probabilistically bisimilar. However one can try to determine how similar they are. The matrix representations are as follows:

$$\mathbf{A} = \mathbf{M}(A) = \mathbf{M}_a(A) \oplus \mathbf{M}_b(A) = \begin{pmatrix} 0 & \frac{1}{2} \\ 0 & 0 \end{pmatrix} \oplus \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & 0 \end{pmatrix}$$

$$\mathbf{B} = \mathbf{M}(B) = \mathbf{M}_a(B) \oplus \mathbf{M}_b(B) = \begin{pmatrix} 0 & \frac{1}{4} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 \end{pmatrix} \oplus \begin{pmatrix} 0 & 0 & \frac{3}{4} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\mathbf{C} = \mathbf{M}(C) = \mathbf{M}_a(C) \oplus \mathbf{M}_b(C) = \begin{pmatrix} 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{51}{100} \\ 0 & 0 & 0 & 0 \end{pmatrix} \oplus \begin{pmatrix} 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{49}{100} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

The problem is to find a $\mathbf{K}_A, \mathbf{K}_B$, and $\mathbf{K}_C \in \mathcal{C}$ such that the norm of the difference between $\mathbf{K}_A^\dagger \mathbf{A} \mathbf{K}_A$ and $\mathbf{K}_B^\dagger \mathbf{B} \mathbf{K}_B$ or $\mathbf{K}_C^\dagger \mathbf{C} \mathbf{K}_C$ is minimal. There is only a finite (though exponentially growing) number of possible classification operators $\mathbf{K} \in \mathcal{C}$. A brute force approach looking at all possible $\mathbf{K}$ allows us to determine the ε-bisimilarity of A and B, and of A and C. Interestingly the optimal $\mathbf{K} = \mathbf{K}_B = \mathbf{K}_C$ is coincidentally the same for both B and C:

$$\mathbf{K} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \mathbf{K}^\dagger = \begin{pmatrix} \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix},$$

while for $\mathbf{K}_A$ we can take the identity.

Measuring the difference based on the operator norm leads to the following:

$$\inf_{\mathbf{K} \in \mathcal{C}} \|\mathbf{A} - \mathbf{K}^\dagger \mathbf{B} \mathbf{K}\| = \frac{1}{8}, \qquad \inf_{\mathbf{K} \in \mathcal{C}} \|\mathbf{A} - \mathbf{K}^\dagger \mathbf{C} \mathbf{K}\| = \frac{1}{200}.$$

5 Conclusions

In this paper we have investigated *quantitative relations*, in particular probabilistic transition relations. We were able to extend the classical framework of Abstract Interpretation to a quantitative domain by taking the Moore-Penrose pseudo-inverse as an appropriate replacement for the order-theoretic concept of Galois connections. Based on this methodology of *Probabilistic Abstract Interpretation*, previously introduced only in a finite dimensional setting [13], we recast (probabilistic) process equivalences in terms of linear operators. This formulation has a very strong resemblance to notions of similarity in mathematical control theory, e.g. [29, Def 4.1.1]. Finally we were able to weaken strict process equivalences to approximate ones. This provides a novel approach towards the notion of approximative or ε-bisimilarity and adds new aspects to existing approaches, like those based on metrics [16] or pseudo-metrics [9, 30]. In particular, our approach allows for a statistical interpretation of the approximation ε which relates this quantity to the number of tests we need to perform in order to accept a given hypothesis with a certain confidence in a "hypothesis testing" approach to statistical testing. This is particularly important in a security context; we are confident that these notions of approximate similarity can be fruitfully employed in security related applications, such as *approximate confinement*, which provided the original motivation for this work [12]. Aldini et al. adopted a similar approach to study probabilistic non-interference in a CSP-like language [1].

References

1. A. Aldini, M. Bravetti, and R. Gorrieri. A process algebraic approach for the analysis of probabilistic non-interference. *Journal of Computer Security*, 2003. To appear.
2. J.A. Bergstra, A. Ponse, and S.A. Smolka, editors. *Handbook of Process Algebra.* Elsevier Science, Amsterdam, 2001.
3. A. Böttcher and B. Silbermann. *Introduction to Large Truncated Toeplitz Matrices.* Springer Verlag, New York, 1999.
4. S.L. Campbell and D. Meyer. *Generalized Inverse of Linear Transformations.* Constable and Company, London, 1979.
5. P. Cousot and R. Cousot. Abstract Interpretation: A Unified Lattice Model for Static Analysis of Programs by Construction or Approximation of Fixpoints. In *Proceedings of POPL'77*, pages 238–252, Los Angeles, 1977.
6. P. Cousot and R. Cousot. Systematic Design of Program Analysis Frameworks. In *Proceedings of POPL'79*, pages 269–282, San Antonio, Texas, 1979.
7. P. Cousot and R. Cousot. Abstract Interpretation and Applications to Logic Programs. *Journal of Logic Programming*, 13(2-3):103–180, July 1992.
8. D. Dams, R. Gerth, and O. Grumberg. Abstract interpretation of reactive systems. *ACM Transactions on Programming Languages and Systems*, 19(2):253–291, 1997.
9. J. Desharnais, R. Jagadeesan, V. Gupta, and P.Panangaden. The metric analogue of weak bisimulation for probabilistic processes. In *Proceedings of LICS'02*, pages 413–422, Copenhagen, Denmark, 22-25 July 2002. IEEE.

10. A. Di Pierro, C. Hankin, and H. Wiklicky. Approximate confinement under uniform attacks. In *Proceedings of SAS'02*, volume 2477 of *Lecture Notes in Computer Science*. Springer Verlag, 2002.
11. A. Di Pierro, C. Hankin, and H. Wiklicky. Approximate non-interference. In *Proceedings of CSFW'02*, pages 3–17, Cape Breton, 24–26 June 2002. IEEE.
12. A. Di Pierro, C. Hankin, and H. Wiklicky. Approximate non-interference. *Journal of Computer Security (WITS '02 Issue)*, 2003. To appear.
13. A. Di Pierro and H. Wiklicky. Concurrent Constraint Programming: Towards Probabilistic Abstract Interpretation. In *Proceedings of PPDP'00*, pages 127–138, Montréal, Canada, 2000. ACM.
14. A. Di Pierro and H. Wiklicky. Measuring the precision of abstract interpretations. In *Proceedings of LOPSTR'00*, volume 2042 of *Lecture Notes in Computer Science*, pages 147–164. Springer Verlag, 2001.
15. A. Di Pierro and H. Wiklicky. A C*-algebraic approach to the operational semantics of programming languages. In preparation, 2003.
16. A. Giacalone, C.-C. Jou, and S.A. Smolka. Algebraic reasoning for probabilistic concurrent systems. In *Proceedings of the IFIP WG 2.2/2.3 Working Conference on Programming Concepts and Methods*, pages 443–458. North-Holland, 1990.
17. C. Godsil and G. Royle. *Algebraic Graph Theory*, volume 207 of *Graduate Texts in Mathematics*. Springer Verlag, New York – Heidelberg – Berlin, 2001.
18. W.H. Greub. *Linear Algebra*, volume 97 of *Grundlehren der mathematischen Wissenschaften*. Springer Verlag, New York, third edition, 1967.
19. B. Jonsson, W. Yi, and K.G. Larsen. *Probabilistic Extensions of Process Algebras*, chapter 11, pages 685–710. Elsevier Science, Amsterdam, 2001. see [2].
20. D. Kozen. Semantics for probabilistic programs. *Journal of Computer and System Sciences*, 22:328–350, 1981.
21. K.G. Larsen and A. Skou. Bisimulation through probabilistic testing. *Information and Computation*, 94:1–28, 1991.
22. R. Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin – New York, 1980.
23. B. Mohar and W. Woess. A survey on spectra of infinite graphs. *Bulletin of the London Mathematical Society*, 21:209–234, 1988.
24. D. Monniaux. Abstract interpretation of probabilistic semantics. In *Proceedings of SAS'00*, volume 1824 of *Lecture Notes in Computer Science*. Springer Verlag, 2000.
25. F. Nielson, H. Riis Nielson, and C. Hankin. *Principles of Program Analysis*. Springer Verlag, Berlin – Heidelberg, 1999.
26. S.A. Smolka R.J. van Glabbeek and B. Steffen. Reactive, Generative and Stratified Models of Probabilistic Processes. *Information and Computation*, 121:59–80, 1995.
27. D.A. Schmidt. Binary relations for abstraction and refinement. In *Workshop on Refinement and Abstraction*, Amagasaki, Japan, November 1999.
28. J. Shao. *Mathematical Statistics*. Springer Texts in Statistics. Springer Verlag, New York – Berlin – Heidelberg, 1999.
29. E.D. Sontag. *Mathematical Control Theory: Deterministic Finite Dimensional Systems*, volume 6 of *Texts in Applied Mathematics*. Springer Verlag, 1990.
30. F. van Breugel and J. Worrell. Towards quantitative verification of probabilistic transition systems. In *Proceedings of ICALP'01*, volume 2076 of *Lecture Notes in Computer Science*, pages 421–432. Springer Verlag, 2001.
31. R.J. van Glabbeek. *The Linear Time - Branching Time Spectrum I. The Semantics of Concrete, Sequential Processes*, chapter 1, pages 3–99. Elsevier Science, Amsterdam, 2001. see [2].