Approximating Labelled Markov Processes

Josée Desharnais*
Département d'Informatique
Université Laval
Québec, Canada, G1K 7P4
josee.desharnais@ift.ulaval.ca

Vineet Gupta
Stratify Inc.
501 Ellis St.
Mountain View CA 94043, USA
vineet@stratify.com

Radha Jagadeesan†
School of CTI, DePaul University,
243 S. Wabash Avenue
Chicago, Illinois 60604-2287, USA
rjagadeesan@cs.depaul.edu

Prakash Panangaden*
School of Computer Science
McGill University
Montréal, Canada, H3A 2A7
prakash@cs.mcgill.ca

October 23, 2002

Abstract

Labelled Markov processes are probabilistic versions of labelled transition systems. In general, the state space of a labelled Markov process may be a continuum. In this paper, we study approximation techniques for continuous-state labelled Markov processes.

We show that the collection of labelled Markov processes carries a Polish-space structure with a countable basis given by finite-state Markov chains with rational probabilities; thus permitting the approximation of quantitative observations (e.g. an integral of a continuous function) of a continuous-state labelled Markov process by the observations on finite-state Markov chains. The primary technical tools that we develop to reach these results are

- A variant of a finite-model theorem for the modal logic used to characterize bisimulation, and
- an isomorphism between the poset of Markov processes (ordered by simulation) with the ω-continuous dcpo *Proc* (defined as the solution of the recursive domain equation $\mathit{Proc} = \prod_{\mathsf{L}} \mathcal{P}_{\mathsf{Pr}}(\mathit{Proc})$).

The isomorphism between labelled Markov processes and *Proc* can be independently viewed as a full-abstraction result relating an operational (labelled Markov process) and a denotational (*Proc*) model and yields a logic complete for reasoning about simulation for continuous-state processes.

*Research supported in part by NSERC and MITACS.
†Research supported by NSF.

Contents

1 Introduction

Markov processes with continuous state spaces or continuous time evolution (or both) arise naturally in several fields of physics, biology, economics and computer science. Examples of such systems are brownian motion, gas diffusion, population growth models and changes in stock prices[1]. In order to come to *computational* grips with such systems one needs a notion of discrete approximation. The purpose of this paper is to study such general *interacting* Markov processes from the point of view of approximation. In all our work the Markov property - sometimes called the "memoryless property" - is essential. This says that the transition probabilities only depend on the current state and not on the past history.

Continuous state Markov processes arise naturally in performance modelling of computer systems.

Example 1.1 A system resource scheduler may be modelled as a *queue* [KS60] — a Markov process in which jobs come in, and are processed by a server. The rate at which jobs arrive (or their inter-arrival times) and the rate at which they are serviced are given by various distributions.

The next example shows that continuous state systems arise in even superficially discrete paradigms.

Example 1.2 Consider a probabilistic process algebra with recursion. The natural model of probabilistic process algebra process with recursion may have uncountably many states. For instance consider the process $\mathcal{P} = a.\mathcal{P} +_{\frac{1}{2}} b.\mathcal{P}$, where the probabilistic choice operator takes either branch with equal probability. This process produces a uniform distribution over all infinite strings of a's and b's, so a continuous state space is inevitable.

The next example shows the importance of concurrency and verification.

Example 1.3 Consider the flight management system of an aircraft. It is responsible for monitoring the state of the aircraft – the altitude, windspeed, direction, roll, yaw etc. – periodically (usually several times a second), it also monitors navigational data from satellites and makes corrections, as needed, by issuing commands to the engines and the wing flaps. The physical system is a complex continuous real-time stochastic system; stochastic because the response of the physical system to commands cannot be completely deterministic and also because of unexpected situations like turbulence. From the point of view of the flight management system, however, the system is discrete-time and has continuous space. The time unit is the sampling rate. The entire system consists of many interacting concurrent components and programming it correctly – letting alone verifying that the system works – is very challenging. A formal model of this type of software brings us into the realm of process algebra, because of the concurrent interacting components, stochastic processes and real-time systems, the last because the responses have hard deadlines. Such a system is being studied with an industrial partner from the point of view of formal modelling and the feasibility of eventual formal verification.

The fundamental example of a continuous stochastic system is the so called brownian motion observed by Brown in 1827 and studied systematically by Einstein [Ein05] and Smoluchowski [Smo06] and others.

Example 1.4 The basic phenomenon that Brown observed is that grains of pollen moved in an apparently random fashion in the air. Later this was understood as arising from the impact of air molecules and this observation became one of the fundamental pieces of direct evidence for the molecular theory of matter. Mathematically, one can think of a brownian motion as a "random variable extended in time". More precisely, consider the Weiner process W, a function from reals (representing time) to random variables that satisfies:

[1] It might come as a surprise that discrete quantities, such as money or population, are best analyzed as continuous quantities. If one wants to use analytical tools, the continuous is used to approximate the discrete.

- Gaussian property: $W(t_1), \ldots, W(t_n)$ is a multivariate normal distribution;
- $W(t+s) - W(s)$ has a normal distribution with mean 0 and variance proportional to t;
- $W(t+s) - W(s)$ is independent of $W(t+s') - W(s')$ whenever $s' > t+s$;
- the sample paths of W can be chosen to be continuous with probability 1.

This mathematical model is the starting point of many stochastic systems; see the book "Stochastic Differential Equations" by Øksendal [Øks95] for a very readable survey.

In a series of previous papers [BDEP97, DEP98, DEP02] we studied Markov processes with continuous state spaces, called *labelled Markov processes* or **LMP**s. We gave a definition of bisimulation between **LMP**s, and gave a logical characterization of this bisimulation. We have also explored the expressiveness and semantics of probabilistic programming languages with continuous state spaces [GJP99]. In this paper our aim is to establish the foundations of computing with these processes, with the aim of computing approximations to the observations that one makes on continuous stochastic systems.

Observations What are the observations of interest in such systems? At the outset we wish to emphasize the importance of quantitative properties and the need for techniques to compute these quantities.

- In the Wiener process example, we wish to know properties like the distribution of times when $W(t) = n$, the asymptotic distribution, whether a steady-state asymptotic distribution exists, and the distribution after a certain time.
- In the performance modelling example, we want to know the expected length of time a job will need to wait, the average number of jobs in the queue, the probability that the number of jobs in the queue exceeds a certain number, the expected fraction of time that the server is busy — these numbers guide decisions on the amount of resource needed, resource bottlenecks etc.
- In the process algebra example, we are interested in computing various conditional probabilities relating occurrences of events, for example see [GJP99]. Recursion forces us to use measure-theoretic apparatus such as the Radon-Nikodym theorem for this purpose.
- In the flight management system example we would want to know average response times of components, drift probabilities and large deviation bounds.

An important reason for the focus on "quantitative" rather than "logical" properties for observations is the incoherence between the notions of approximation and logical reasoning.

Example 1.5 Consider the real line. Let us say that we are approximating real numbers by sequences of rational numbers. For example, the sequence $\{1, -1, \frac{1}{2}, -\frac{1}{2}, \ldots \frac{1}{n}, -\frac{1}{n}\}$ converges to 0. Consider the logical property $\phi(x) = x > 0$. ϕ is satisfied infinitely often (and hence arbitrarily late) in the approximating sequence. Unfortunately, the property is also invalidated infinitely often (and hence arbitrarily late) in the approximating sequence.

What this example shows is that with an arbitrary logic and notion of approximation we have no guarantee that the approximation will behave well with respect to the logical formulas. With our modal logic for characterizing bisimulation this will not happen, but this modal logic is useless as a logic for the *specification* of and reasoning about systems; it is far too weak to capture properties of interest[2].

Later in the paper, in subsection 3, we present an example of a continuous state system and a similarly unstable property written in the logic **pCTL***. The problem is that with a two-valued logic one has to make a sharp decision whereas, in general, approximation schemes approach the correct answer in an incremental fashion. If one uses approximation techniques to determine the result of a boolean answer one will not be able to tell when the approximation has converged. This general kind of example of the subtle interaction between approximation and logic leads us to revisit Kozen's seminal ideas on logics in the context of probability [Koz81, Koz85].

He argued that - in the probabilistic context - one should take measurable functions as a natural generalization of logical formulas. Truth values now take values in the interval $[0, 1]$ rather than in the set $\{0, 1\}$. A state s is replaced by a distribution μ over states. Traditionally, the satisfaction relation is a pairing between formulas and states yielding a truth value; $s \models \phi$; the analogue is given by the integral $\int f \mathrm{d}\mu$ yielding a real number between 0 and 1.

In a related paper - on metrics for probabilistic processes [DGJP99] - we use these ideas very literally. Here we use the spirit of these same ideas. Essentially we take the point of view that such integrals define the quantities of interest. We provide a technique for systematically approximating such integrals by showing that there are "enough" finite approximants of the labelled Markov process; or, in more mathematical language, we show that the collection of labelled Markov processes have a Polish space structure.

In this viewpoint, the notion of approximation acquires a precise quantitative character — we are demanding the result of an observation (e.g. an integral) within some error bound ϵ. In this world-view, the wild instability of logical reasoning (as reflected in the above example) in the context of approximations is replaced by the more stable decrease in the permitted errors ϵ.

Our Results Our main result is a systematic approximation scheme for labelled Markov processes. We have essentially given the collection of labelled Markov processes the kind of structure enjoyed by the real numbers, namely that there is a countable collection of elements, the rationals, which approximate all the elements and determine the behavior of all continuous functions. Our primary technical result is:

> The set of **LMP**s is a Polish space[3] with a countable dense subset given by finite rational **LMP**s.

This shows that finite rational **LMP**s can be used to approximate any **LMP**— we provide a construction of these finite approximants for any **LMP**. Furthermore, our approximation results allow us to approximate integrals of continuous functions by computing them on finite rational **LMP**s, using standard techniques such as:

- Parthasarathy [Par67] shows that the space of all measures on a Polish space X is itself a Polish space. Furthermore, the set of measures, whose supports are finite subsets of the countable set dense in X, is itself dense in the space of measures. Finally, if $\mu_n \longrightarrow \mu$, then $\int f d\mu_n \longrightarrow \int f d\mu$ for all bounded continuous functions f.

- Edalat [Eda94, Eda95a, Eda95c, Eda95b] has exploited domain theoretic methods to define R-integration and show that R-integrability w.r.t. a bounded Borel measure extends Riemann integrability to compact metric spaces. The R-integral of f with respect to measure μ is approximated by the sums of f w.r.t. simple valuations less than μ.

[2] Of course, the weakness of the logic is a virtue when it comes to characterizing bisimulation since it shows that bisimulation can be described very simply.

[3] A Polish space is a topological space underlying a complete separable metric space.

These two ways of approximating integrals can both be used by us since we have shown that we have a (compact) Polish space of **LMP**s. However, the results of the present paper do not *use* the results of Edalat or Parthasarathy.

The route to our results is based first on the observation that the category of **LMP**s is a natural "timed" extension of the poset $([0,1], \leq)$. Since the usual Polish space structure of $[0,1]$ is recovered from the poset $[0,1]$ by the Lawson topology, we study the domain theoretic structure of **LMP**s. Indeed, we establish the following equivalence of categories:

$$\mathbf{LMP} \simeq \mathit{Proc}$$

where **LMP** is the category with objects **LMP**s and with morphisms simulations; and *Proc* is the solution to the recursive domain equation

$$\mathit{Proc} \simeq \prod_{\mathsf{L}} \mathcal{P}_{\mathsf{Pr}}(\mathit{Proc}).$$

Since *Proc* is an ω-continuous dcpo, a standard construction (its Lawson topology) yields a Polish space inducing a Polish space structure on **LMP**. The equivalence $\mathbf{LMP} \simeq \mathit{Proc}$ can also be viewed as a full abstraction result relating an operational model (**LMP**) and a denotational model (*Proc*).

The proof of this result breaks up naturally into two parts:

1. We develop the notion of "finite" approximants to a **LMP** and analyze the logical properties of this set of approximants. For any **LMP**, we explicitly provide a (countable) sequence of approximants such that:

   - For every logical property satisfied by a process, there is an element of the chain that also satisfies the property.
   - The sequence is a chain in the simulation ordering; the sup of this chain gives the original process back.

   The key technical tools in this development are the logical characterization of bisimulation [DEP98] and a construction (in Section 4) of a class of approximants.

2. We analyse the logic of *Proc* and show that there is a perfect match between simulation/bisimulation on the one hand and the partial order/equality in D on the other hand.

It is quite interesting that there seem to be two alternate paths to these results. The presentation in section 4 is purely based on measure theory and combinatorics and (almost) proves all the results. The presentation of section 5 is based on domain theory and also proves all the results, in particular we get logical characterization of simulation. The reader without background in domain theory can read section 4 and skip the following section and vice versa for the reader who is more comfortable with domain theory. In fact from the viewpoint of domain theory the explicit description of the approximants is not really necessary, the whole argument proceeds more abstractly.

Some new and interesting consequences of the equivalence are:

- The equivalence endows **LMP** with least upper bounds of ω-chains (w.r.t. the simulation ordering). This shows that **LMP** can be used as the target of interpretation of a syntax that includes recursion.
- The internal logic of *Proc* is a logic complete for reasoning about simulation of **LMP**s.
- Baier and Kwiatkowska [BK96]show that bisimulation and simulation equivalence agree for discrete space systems. Our results show that this coincidence of bisimulation and simulation equivalence extends to continuous space systems.

1.1 Summary of paper

In Section 2 we recall background material on labelled Markov processes. Section 4 develops the class of approximants, and studies its properties. In section 5, we analyze the solution of the recursive domain equation $D \simeq \Pi_{\mathsf{L}}\mathcal{P}_{\mathsf{Pr}}(D)$. Finally, in section 6 we prove the equivalence $\mathbf{LMP} \simeq$ *Proc*. In section 8, we place our work in the context of extant research.

2 Background: Labelled Markov Processes

In this section we recapitulate the definitions of labelled Markov processes from [DEP02][4]. However, the presentation of bisimulation has been reformulated in a more relational style very close to the original definition of Larsen and Skou [LS91]. This entails a modification and reorganization of the proofs. We assume that the reader is familiar with process algebra and labelled transition systems [Mil89] and the concept of bisimulation in the non-probabilistic setting.

Labelled Markov processes are probabilistic versions of labelled transition systems. Corresponding to each label a Markov process is defined. The transition probability is given by a *stochastic kernel* (Feller's terminology [Fel71]) also commonly called a *Markov kernel.* Thus the indeterminacy has two sources: the "choice" made by the environment - no probabilities are attributed to this at all - and the probabilistic transitions made by the process. This is the so called "reactive" model and is due to Larsen and Skou [LS91] who used it in a discrete state-space setting.

In brief, a labelled Markov process can be described as follows. There is a set of states and a set of labels. The system is in a state at each point in time. The environment selects an action, and the system reacts by moving to another state. The transition to another state is governed by a probabilistic law. For each label there is a transition probability distribution which gives the probability distribution of the possible final states given the initial state. For discrete state spaces, this is essentially the model developed by Larsen and Skou [LS91].

We extended this to continuous-state systems, thus forcing our formalism to be couched in measure-theoretic terms. For instance, we cannot ask for the transition probability to any set of states — we need to restrict ourselves to measurable sets. In fact, we need to assume metric space structure on the state space. The classical theory of Markov processes is typically carried out in the setting of Polish spaces rather than on abstract measure spaces. We work with *analytic spaces* which generalize Polish spaces. However, in this paper we focus on more combinatorial issues and suppress some of the finer points of measure theory, see [DEP02] for details.

A key ingredient in the theory is the stochastic kernel or Markov kernel. We will call it a *transition probability function.*

Definition 2.1 *A* ***transition (sub-)probability function*** *on a measurable space* (S, Σ) *is a function* $\tau : S \times \Sigma \longrightarrow [0, 1]$ *such that for each fixed* $s \in S$*, the set function* $\tau(s, \cdot)$ *is a (sub-)probability measure, and for each fixed* $X \in \Sigma$ *the function* $\tau(\cdot, X)$ *is a measurable function.*

One interprets $\tau(s, X)$ as the probability of the process starting in state s making a transition into one of the states in X. The transition probability is really a *conditional probability*; it gives the probability of the process being in one of the states of the set X after the transition, *given* that it was in the state s before the transition. In general the transition probabilities could depend on time, in the sense that the transition probability could be different at every step, but still independent of past history; we always consider the time-independent case.

[4] The work was first announced in [BDEP97]

We will work with *sub-probability* functions; i.e. with functions where $\tau(s, S) \leq 1$ rather than $\tau(s, S) = 1$. The mathematical results go through in this extended case. We view processes where the transition functions are only sub-probabilities as being *partially defined*, opening the way for a notion of approximation.

The stochastic systems studied in the literature are usually only the very special version where $\tau(s, S)$ is either 1 or 0. We call such processes *total* and the general processes are called *partial*. We capture the idea that an action is rejected by setting $\tau(s, S)$ to be 0.

Definition 2.2 *A* ***partial labelled Markov process (LMP)*** $\mathcal{S}$ *with label set* $\mathcal{A}$ *is a structure* $(S, i, \Sigma, \{\tau_a \mid a \in \mathcal{A}\})$*, where* S *is the set of states,* i *is the initial state, and* Σ *is the Borel* σ*-field on* S*, and*

$$\forall a \in \mathcal{A}, \tau_a : S \times \Sigma \longrightarrow [0, 1]$$

is a transition sub-probability function.

We will fix the label set to be $\mathcal{A}$ once and for all. The resulting theory is not seriously restricted by this. We will write (S, i, Σ, τ) for partial labelled Markov processes, instead of the more precise $(S, i, \Sigma, \{\tau_a \mid a \in \mathcal{A}\})$. Most of the time we will just say labelled Markov process rather than *partial* labelled Markov process.

We give a couple of simple examples to illustrate the ideas.

Example 2.3 (From [DEP02]) Consider a process with two labels $\{a, b\}$. The state space is upper right quadrant of the real plane, $\mathbf{R}^2$ together with a single extra point. In order to describe this system conveniently we will pretend, at first, that the state space is the entire real plane. When the process makes an a-move from state (x_0, y_0), it jumps to (x, y_0), where the probability distribution for x is given by the density $K_\alpha \exp(-\alpha(x - x_0)^2)$, where $K_\alpha = \sqrt{\alpha/\pi}$ is the normalizing factor. When it makes a b-move it jumps from state (x_0, y_0) to (x_0, y), where the distribution of y is given by the density function $K_\beta \exp(-\beta(y - y_0)^2)$. The meaning of these densities is as follows. The probability of jumping from (x_0, y_0) to a state with x-coordinate in the interval $[s, t]$ under an a-move is $\int_s^t K_\alpha \exp(-\alpha(x - x_0)^2)dx$. All points with $x < 0$ or $y < 0$ are identified as a single absorbing state. Once it is in this state no more transitions are possible. Note that the probability of jumping to any given point is, of course, 0. In this process the interaction with the environment controls whether the jump is along the x-axis or along the y-axis but the actual extent of the jump is governed by a probability distribution. If there were just a single label we would have an ordinary (time-independent) Markov process; in fact it would be a brownian motion with absorbing walls.

Our second example comes from queuing theory.

Example 2.4 Consider how to represent a process that is a single queue with infinitely many buffers or cells. (Later in the paper, we show how to model observations on such a queue in the context of our concrete representation.) The process is modelled as a **LMP**. The state set is $\mathbf{N} \times \mathbf{R}$. Intuitively, the first number will represent the number of buffers filled and the second will represent the length of time to be spent in this state.

Each state (k, t) has a transition labelled *arr* to states $(k + 1, t')$, and if $k > 0$, a transition labelled *dep* to $(k - 1, t')$. The probabilities on these are given by the arrival and departure probabilities for the queue — for example for $M/M/1$ queues these are given by exponential distributions.

Example 2.5 We now represent a Wiener process with drift. This process is the solution to the stochastic differential equation

$$dx = Cdt + dW \quad x(0) = 0$$

where C is a constant and W is the Wiener process.

The set of states is $\mathbf{R}$, with the usual σ-algebra. The start state is 0. For every rational $q > 0$ we have a label a_q, and for every state s we have an a_q labelled transition whose probability distribution is normal, with mean $s + C \times q$ and variance q. Each state is also labelled with its real number, as described before. Each label a_q intuitively represents the passage of q time units.

2.1 Bisimulation and Logic

The fundamental process equivalence that we consider is *strong probabilistic bisimulation* or just "bisimulation" for the present paper. The definition that we use is an adaptation of a definition due to Larsen and Skou [LS91], with extra conditions to deal with measure-theoretic issues. In an earlier paper [BDEP97] we had introduced a version of this definition based on categorical ideas but in the present paper we present a version much closer in form to that of Larsen and Skou[5]. In this section we also recapitulate[6] the result of our paper [DEP98] on a logical characterization of bisimulation.

Probabilistic bisimulation means matching the moves and probabilities *exactly*— thus each system must be able to make the same transitions with the same probabilities as the other. Larsen and Skou define a bisimulation relation R as an equivalence relation on the states satisfying the condition that, for each label a, equivalent states have equal probability of making an a-transition to any R-equivalence class of states. In the continuous case, we demand that equivalent states have equal probability of making an a-transition to any union of equivalence classes of states *provided that the union is measurable.*

Instead of talking about sets of equivalence classes we will instead use the notion of R-*closed* sets. Let R be a binary relation on a set S. We say a set $X \subseteq S$ is R-*closed* if $R(X) := \{t | \exists s \in X, sRt\}$ is a subset of X. If R is reflexive, this becomes $R(X) = X$. If R is an equivalence relation, a set is R-closed if and only if it is a union of equivalence classes. It will also be convenient to explicitly define the notion of *direct sum* of two labelled Markov processes.

Definition 2.6 *Let $\mathcal{S} = (S, i, \Sigma, \tau)$ and $\mathcal{S}' = (S', i', \Sigma', \tau')$ be two labelled Markov processes. The* ***direct sum*** *$\mathcal{S} + \mathcal{S}'$ of these processes is a process $\mathcal{U} = (U, u_0, \Omega, \rho)$ where $U = S \uplus S' \cup \{u_0\}$, Ω is the σ-field generated by $\Sigma \cup \Sigma'$, and the transitions are as follows: for all $s \in S$, $s' \in S'$, $\rho_a(s, X \uplus X') = \tau_a(s, X)$ and $\rho_a(s', X \uplus X') = \tau'_a(s', X')$. The initial distribution is given by $\rho_a(u_0, i) = \rho_a(u_0, i') = 1/2$.*

This construction is purely formal and is only used in order to define a relation on the common state space.

Definition 2.7 *Let $\mathcal{S} = (S, i, \Sigma, \tau)$ be a labelled Markov process. An equivalence relation R on S is a* ***bisimulation*** *if whenever sRs', with $s, s' \in S$, we have that for all $a \in \mathcal{A}$ and every R-closed measurable set $X \in \Sigma$, $\tau_a(s, X) = \tau_a(s', X)$. Two states are bisimilar if they are related by a bisimulation relation.*

Let $\mathcal{S} = (S, i, \Sigma, \tau)$ and $\mathcal{S}' = (S', i', \Sigma', \tau')$ be a pair of labelled Markov process. $\mathcal{S}$ and $\mathcal{S}'$ are bisimilar if there is a bisimulation relation on some process $\mathcal{U}$, of which $\mathcal{S}$ and $\mathcal{S}'$ are direct summands, relating i and i' in $\mathcal{U}$.

Alternately, bisimulation on the states of a labelled Markov process can be viewed as the maximum fixed point of the following (monotone) functional F on the lattice of equivalence relations on $(S \times S, \subseteq)$:

$$s\, F(R)\, t \text{ if for all } a \in \mathcal{A}, \text{and all } R\text{-closed } C \in \Sigma, \tau_a(s, C) \leq \tau_a(t, C).$$

The intuition of this definition is that the relation R relates those states that can be "lumped" together. Bisimulation is the largest such relation. In fact the notion of bisimulation was known in the queuing theory

[5] For those who are interested, our version amounts to working with cospans rather than with spans.

[6] Actually the proof has to be slightly redone because we have slightly changed the logic.

community [KS60] under the name of "lumpability". With regard to bisimulation on processes, note that we do not require $\mathcal{U}$ to be exactly $\mathcal{S}+\mathcal{S}'$ but rather a sum of a number of processes, of which are $\mathcal{S}$ and $\mathcal{S}'$. The reason for this is that transitivity of bisimulation would not follow in any obvious way with $\mathcal{U}$ being exactly the direct sum. However, the logical characterization of bisimulation below will allow us to restrict ourselves to only considering $\mathcal{U} = \mathcal{S} + \mathcal{S}'$ in the above definition.

Proposition 2.8 *Bisimulation is an equivalence relation.*

Proof . Bisimulation is obviously reflexive and symmetric. For transitivity, consider two bisimulations R_1 and R_2 on a single process $\mathcal{S} = (S, i, \Sigma, \tau)$. Let R be the transitive closure of $R_1 \cup R_2$. Then every measurable R-closed set is also R_i-closed, $i = 1, 2$, then it follows easily that R is a bisimulation on $\mathcal{S}$.

In the case of bisimulation between processes, let R_1 be a bisimulation between $\mathcal{S}$ and $\mathcal{S}'$ through process $\mathcal{U}_1 = \mathcal{S} + \mathcal{S}' + \mathcal{T}$ (for some $\mathcal{T}$) and R_2 a bisimulation between $\mathcal{S}'$ and $\mathcal{S}''$ through process $\mathcal{U}_2 = \mathcal{S}' + \mathcal{S}'' + \mathcal{T}'$(for some $\mathcal{T}'$). Then construct the direct sum $\mathcal{U} = \mathcal{U}_1 + \mathcal{U}_2$. Consider the following relations in $\mathcal{U} \times \mathcal{U}$:

- Let $\mathtt{I}$ the symmetric and reflexive closure of the relation in $\mathcal{U} \times \mathcal{U}$ that relates copies of states from $\mathcal{S}'$ in $\mathcal{U}_1$ to the copies of the same states in $\mathcal{U}_2$.
- Let $E_1 = R_1 + \mathtt{id}_{\mathcal{U}_2}$, $E_2 = \mathtt{id}_{\mathcal{U}_1} + R_2$

It is easy to show that I, E_1, E_2 are bisimulations on $\mathcal{U}$. Considering the transitive closure of $I \cup E_1 \cup E_2$ as in the above proof yields the required result. ∎

2.2 A Logical Characterization of Bisimulation

One can define a simple modal logic and prove that two states are bisimilar if and only if they satisfy exactly the same formulas. Indeed for finite-state processes one can decide whether two states are bisimilar and effectively construct a distinguishing formula in case they are not [DEP02].

As before we assume that there is a fixed set of "actions" $\mathcal{A}$. The logic is called $\mathcal{L}$ and has the following syntax:

$$\mathsf{T} \mid \phi_1 \wedge \phi_2 \mid \langle a \rangle_q \phi$$

where a is an action and q is a rational number. This is the basic logic with which we establish the logical characterization. In later sections we will work with this logic augmented with disjunction, $\mathcal{L}_\vee$:

$$\mathcal{L} \mid \phi_1 \vee \phi_2.$$

Definition 2.9 *Define the depth of formulas inductively as follows:*

$$\begin{aligned} depth(\mathsf{T}) &= 0 \\ depth(\phi \wedge \psi) &= max(depth(\phi), depth(\psi)) \\ depth(\phi \vee \psi) &= max(depth(\phi), depth(\psi)) \\ depth(\langle a \rangle_r \phi) &= depth(\phi) + 1 \end{aligned}$$

Given a labelled Markov process $\mathcal{S} = (S, i, \Sigma, \tau)$ we write $s \models \phi$ to mean that the state s satisfies the formula ϕ. The definition of the relation $\models$ is given by induction on formulas. The definition is obvious for the propositional constant T, conjunction and disjunction. We say $s \models \langle a \rangle_q \phi$ if and only if $\exists X \in \Sigma.(\forall s' \in X.s' \models \phi) \wedge (\tau_a(s, X) > q)$. In other words, the process in state s can make an a-move to a state, that satisfies ϕ, with probability strictly greater than q[7]. We write $[\![\phi]\!]_{\mathcal{S}}$ for the set $\{s \in S \mid s \models \phi\}$, and $\mathcal{S} \models \phi$ if $i \models \phi$. We often omit the subscript when no confusion can arise.

The main theorem relating $\mathcal{L}$ and bisimulation is the following[8]. This was proved in [DEP98, DEP02]. The present proof is adapted to our relational presentation of bisimulation, certain technical details are suppressed, see [DEP02] for the complete proof.

Theorem 2.10 *Let (S, i, Σ, τ) be a labelled Markov process. Two states $s, s' \in S$ are bisimilar if and only if they satisfy the same formulas of $\mathcal{L}$.*

Proof . $\Rightarrow$: Let R be a bisimulation on $\mathcal{S}$. We prove by induction on the structure of formulas that if sRs' then s and s' satisfy the same formulas. The cases of $\top$ and conjunction are trivial. Now assume the implication is true for ϕ, i.e., for every pair of bisimilar states either both satisfy ϕ or neither of them does. This means that the set $[\![\phi]\!]$ is R-closed. It is easy to prove (see [DEP02] for details) that $[\![\phi]\!]$ is always measurable. Since R is a bisimulation, $\tau_a(s, [\![\phi]\!]) = \tau_a(s', [\![\phi]\!])$ for all $a \in \mathcal{A}$. So s and s' satisfy the same modal formulas of the form $\langle a \rangle_q \phi$.

$\Leftarrow$: In our previous paper [DEP98, DEP02] we constructed a quotient process obtained by defining an equivalence relation on the states of a process $\mathcal{S} = (S, i, \Sigma, \tau)$ as follows. Briefly, two states were defined to be equivalent if they satisfy the same formulas of the logic; we write $s \approx s'$ when this is the case. We then form the quotient $\mathcal{Q} = (Q, q_0, \Omega, \rho)$ with the canonical projection q where Ω is the largest σ-field such that q is measurable. We defined a transition probability ρ on the quotient process and proved the following properties:

1. $B \in \Omega$ if and only if $q^{-1}(B) \in \Sigma$,
2. $\forall s \in S, B \in \Omega . \rho(q(s), B) = \tau(s, q^{-1}(B))$.

Now we will use these facts to show that the relation $\approx$ defined on the states of $\mathcal{S}$ is in fact a bisimulation relation. Let $X \in \Sigma$ be $\approx$-closed. Then we have $X = q^{-1}q(X)$ and hence $q(X) \in \Omega$. Now if $s \approx s'$, then $q(s) = q(s')$, and $\tau_a(s, X) = \rho_a(f(s), f(X)) = \tau_a(s', X)$, and hence $\approx$ is bisimulation. It is straightforward to adapt this to the case where we are talking about bisimulation between two processes. ∎

As a corollary, we deduce that the closure ordinal of the functional F defining bisimulation is ω.

Corollary 2.11 *Define a family of relations $R_i \subseteq$ **LMP** $\times$ **LMP** as follows:*

$$\begin{aligned} R_0 &= \mathbf{LMP} \times \mathbf{LMP} \\ R_{i+1} &= F(R_i) \textit{ for } F \textit{ as defined in discussion of Definition2.7} \\ R &= \cap_i R_i \end{aligned}$$

pRq if and only if p is bisimilar to q.

Proof . The bisimulation relation is contained in R_0; hence by induction on i and by using $R_{i+1} = F(R_i)$, and the fact that bisimulation is a fixed point of the monotone functional F, we get that the bisimulation relation is contained in R_i for all i, and hence is contained in $R = \cap_i R_i$.

[7] In our earlier work we had used $\geq$ instead of $>$. We have moved to a strict inequality in order that the result about approximations be correct. This is reflected in the proof of Theorem 4.4.

[8] The logic that Larsen and Skou used in [LS91] has more constructs including disjunction and some negative constructs. They show that for systems satisfying a "minimum deviation condition" — a uniform bound on the degree of branching everywhere— two *states* of the same process are bisimilar if and only if they satisfy the same formulas of their logic.

Now we need to prove that pRq implies p is bisimilar to q. We prove by induction on i that $[\![\phi]\!]$ is the union of R_i equivalence classes for $i \geq d(\phi)$. Formally, sR_it implies for all formulas ϕ of depth i or less, $s \models \phi$ if and only if $t \models \phi$. The base case is trivial.

Inductive step: Let $sR_{i+1}t$. Consider a formula $\langle a \rangle_r \phi$ such that $s \models \langle a \rangle_r \phi$. Thus $\tau_a(s, [\![\phi]\!]) > r$. But since $[\![\phi]\!]_{\mathcal{S}}$ is an R_i equivalence class and is measurable, $\tau_a(t, [\![\phi]\!]_{\mathcal{S}}) = \tau_a(s, [\![\phi]\!]_{\mathcal{S}}) > r$. ∎

2.3 Simulation

The notion of simulation is the natural one-directional version of the definition of bisimulation. Normally, the fact that the definition of bisimulation is coinductive means that, in general, two-way simulation is not bisimulation. However, in the case of our reactive systems, two-way simulation *is* bisimulation; this is in contrast with the usual situation with indeterminate processes. Furthermore, when we make contact with domain-theoretic ideas the notion of simulation will correspond to the domain ordering. Thus when we say $\mathcal{S}$ *approximates* $\mathcal{S}'$ we mean that $\mathcal{S}'$ simulates $\mathcal{S}$. We also introduce a concept called *strict simulation* which will correspond to the "way-below" relation.

Our definition of simulation follows [Des99].

Definition 2.12 *Let $\mathcal{S} = (S, i, \Sigma, \tau)$ be a labelled Markov process. A reflexive and transitive relation (a preorder) R on S is a **simulation** if whenever sRs', with $s, s' \in S$, we have that for all $a \in \mathcal{A}$ and every R-closed measurable set $X \in \Sigma$, $\tau_a(s, X) \leq \tau_a(s', X)$. We say s is simulated by s' if sRs' for some simulation relation R.*

*A simulation R is a **strict simulation** if there is an $\epsilon > 0$ such that for all R-closed $X \in \Sigma$, we have $\tau_a(s, X) < \tau_a(s', X) - \epsilon$ whenever $\tau_a(s', X) > \epsilon$. If we wish to emphasize the role of ϵ we will say that R is an ϵ-**strict** simulation, and we will write R_ϵ. We say s is simulated (strictly simulated) by s' if sRs' for some simulation (resp. strict simulation) relation R.*

Let $\mathcal{S} = (S, i, \Sigma, \tau)$ and $\mathcal{S}' = (S', i', \Sigma', \tau')$ be a pair of labelled Markov process. $\mathcal{S}$ is simulated (strictly simulated) by $\mathcal{S}$ if there is a simulation (resp. strict simulation) relation on some process $\mathcal{U}$ of which $\mathcal{S}$ and $\mathcal{S}'$ are direct summands, relating i and i' in $\mathcal{U}$.

Alternately, simulation on the states of a labelled Markov process can be viewed as the maximum fixed point of the following (monotone) functional G on the lattice of preorders on $(S \times S, \subseteq)$, defined as follows:

$$s\, G(R)\, t \text{ if for all } a \in \mathcal{A}, \text{ for all } R\text{-closed } C \in \Sigma, \tau_a(s, C) = \tau_a(t, C)$$

As before, we do not require $\mathcal{U}$ to be exactly $\mathcal{S} + \mathcal{S}'$ but rather a sum of a number of processes, of which are $\mathcal{S}$ and $\mathcal{S}'$. The reason for this is that transitivity of simulation would not follow in any obvious way with $\mathcal{U}$ being exactly the direct sum. However, in the particular case where the simulated process $\mathcal{S}$ is discrete, if a simulation exists between $\mathcal{S}$ and $\mathcal{S}'$, then there is a simulation on $\mathcal{S} + \mathcal{S}'$. The proof of transitivity of simulation and strict simulation follows the transitivity proof for bisimulation.

The notion of simulation meshes properly with the logic in the sense of the following proposition. Later in this paper, we prove the converse of the following proposition[9]

Proposition 2.13 *If s is simulated (or strictly simulated) by s', then for all formulas $\phi \in \mathcal{L}_\vee$, $s \models \phi$ implies $s' \models \phi$.*

[9] And use it to show that the closure ordinal of G is ω.

Proof . Let R be a simulation on a single process $\mathcal{S} = (S, i, \Sigma, \tau)$. We prove by induction on the structure of formulas that for every formula ϕ, $[\![\phi]\!]$ is R-closed, which implies the result. It is obvious for $\top$, conjunction and disjunction. Now assume it is true for ϕ, and let sRs'. Then, since R is a simulation and $[\![\phi]\!]$ is measurable and R-closed, we have $\tau_a(s, [\![\phi]\!]) \leq \tau_a(s', [\![\phi]\!])$, and hence $[\![\langle a \rangle_q \phi]\!]$ is R-closed for every rational q.

Now if s and s' come from two different processes, observe that if $\mathcal{S}$ is a direct summand of $\mathcal{U}$, a state of $\mathcal{S}$ satisfies exactly the same formulas in $\mathcal{S}$ as in $\mathcal{U}$. Hence the result. ∎

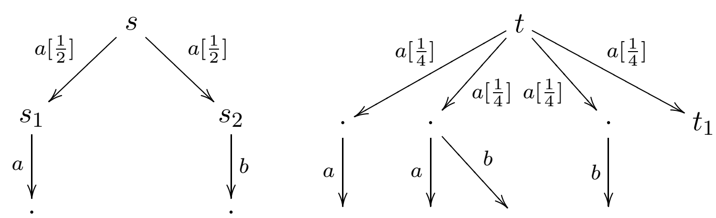
The converse of this result for discrete processes was proven in [Des99] and gives a logical characterization of simulation between a discrete process and an arbitrary one.

Theorem 2.14 *A state s in a discrete process is simulated by a state t in an arbitrary process if and only for all formulas $\phi \in \mathcal{L}_\vee$, $s \models \phi$ implies $t \models \phi$.*

Later in this paper, we extend this result to all processes.

Although $\mathcal{L}$ is enough to characterize bisimulation, characterization of simulation needs disjunction. We now give an example of two simple finite processes, one satisfying all the formulas of $\mathcal{L}$ that the other satisfies but which does not simulate it.

Example 2.15 In the following picture, t satisfies all formulas of $\mathcal{L}$ that s satisfies but t does not simulate s.



Of course there is a formula of $\mathcal{L}$ that distinguishes s and t, namely the formula $\langle a \rangle_0(\langle a \rangle_0 \top \wedge \langle b \rangle_0 \top)$. This formula is satisfied by t but not by s. To see that t satisfies all formulas of $\mathcal{L}$ that s satisfies, note that the only relevant formulas of $\mathcal{L}$ that are satisfied by s are: $\langle a \rangle_r \top$, for $0 < r < 1$, $\langle a \rangle_r \langle a \rangle_0 \top$ and $\langle a \rangle_r \langle b \rangle_0 \top$, for $0 < r < 1/2$. All these formulas are also satisfied by t. To see that t does not simulate s, we could prove that there is no simulation relation between them, but it is simpler to only exhibit a formula of $\mathcal{L}_\vee$, namely $\langle a \rangle_{3/4}(\langle a \rangle_0 \top \vee \langle b \rangle_0 \top)$ that is satisfied by s but not by t. This shows that disjunction is indeed necessary for characterizing simulation.

The next couple of results will allow us to use a simpler definition of simulation when we work with discrete processes.

Corollary 2.16 *If a process simulates a discrete process, then it simulates it through their direct sum.*

Corollary 2.17 *If a process strictly simulates a finite process, then it strictly simulates it through their direct sum.*

Proof . Let $\mathcal{P} = (P, p_0, \mathcal{P}(P), \rho)$ be a finite process strictly simulated by a process $\mathcal{S} = (S, i, \Sigma, \tau)$. Then consider $\mathcal{P}'$ a finite process having the same state-space as $\mathcal{P}$ and whose transition are obtained by increasing all the probabilities of the non-zero transitions of $\mathcal{P}$ in such a way that $\mathcal{P}'$ is still simulated by $\mathcal{S}$. Then it is easy to see that $\mathcal{P}'$ strictly simulates $\mathcal{P}$, through the relation R that relates a state in $\mathcal{P}$ to the same state in $\mathcal{P}'$. This relation lives on the direct sum of $\mathcal{P}$ and $\mathcal{P}'$. Now we know that $\mathcal{P}'$ is simulated by $\mathcal{S}$ through the direct sum of

Figure 1: Three figures, showing how further refinement can toggle the status of a property.

$\mathcal{P}'$ and $\mathcal{S}$ (the above result is for simulation relations). Let R' be the witnessing relation. We say tWs if $t'R's$, where t' is the state in $\mathcal{P}'$ corresponding to t. Now let Y be a W-closed set of $\mathcal{P}+\mathcal{S}$. Then $R(Y \cap P')$ is obviously measurable in $\mathcal{P}'$ and R-closed. Moreover $R(Y \cap P') \cup (Y \cap S)$ is also R'-closed. Hence we have

$$\rho_a(t, Y \cap P) < \rho'_a(t, R(Y \cap P)) \leq \tau_a(s, Y \cap S).$$

∎

The preceding results allow us to use a simpler definition of simulation and strict simulation when a finite process is involved that will not involve direct sums. It is easy to check that – in the following special cases – the following definition of simulation and strict simulation is equivalent to the one we have defined previously.

Definition 2.18 *A simulation between a finite process $\mathcal{P} = (P, p_0, \mathcal{P}(P), \rho)$ and another process $\mathcal{S}$ is a reflexive and transitive relation on $P \cup S$ such that the restrictions of R to $\mathcal{P}$ and $\mathcal{S}$ are simulations and if pRs implies that for every R-closed set $X \subseteq P \cup S$ such that $X \cap S \in \Sigma$, we have $\rho_a(p, X \cap P) \leq \tau_a(s, X \cap S)$.*

A simulation R is a strict simulation if there is an ϵ such that for all R-closed sets X we have $\rho_a(p, X \cap P) < \tau_a(s, X \cap S) - \epsilon$ whenever $\tau_a(s, X \cap S) > \epsilon$.

3 Pathology of logic+approximation: An example

In this section we discuss an example showing that in any reasonably strong logic - roughly speaking, having negation is enough - it is possible for approximations to be very misleading or even totally uninformative about whether the limit process satisfies a formula or not. Of course this is not very surprising when we have negation. However, even without negation one can have problems.

In the next section we will show how to construct approximations to continuous systems such that there exists a chain in the simulation ordering "converging" to the original process in such a way that it respects our logic $\mathcal{L}$. That is, every formula which is satisfied by the continuous process is satisfied infinitely often from some point in the chain and hence is unsatisfied only finitely often. However, in a logic that contains a form of negation, this is hardly possible.

We now give an example of a **LMP** and a formula that it satisfies. We exhibit a family of possible approximations such that infinitely many of the approximants satisfy the property but also infinitely many approximants *fail to* satisfy the property. This means that the truth value alternates and gives no clue as to whether the final process satisfies the formula.

We consider the **pCTL*** property

$$\phi = Pr_{\geq 1}X((Pr_{>1/3}Xa) \vee (Pr_{<1/3}Xb))$$

State $s \models a$ if s is labelled a. A sequence of states π satisfies $X\psi$ if the second state in π satisfies ψ. A state $s \models Pr_{>c}\psi$ if the measure of all sequences which start from the state and satisfy ψ is greater than c, where a measure is imposed on the space of all sequences by letting the measure of the set of all sequences starting with a given finite prefix be the probability of that prefix, and extending this to other sets of sequences. For more details on **pCTL***, we refer the reader to [ASB$^+$95].

Suppose we want to check this property on the following **LMP** $\mathcal{P}$: its state set is given by $\{s_0\} \cup [0,1] \cup [0,1]^2$. There is a transition from s_0 to $[0,1]$, with uniform probability. For every $x \in [0,1]$, there is a transition from x to $\{(x,y)|y \in [0,1]\}$ with uniform probability. Furthermore, a state $(x,y) \in [0,1]^2$ is labelled a if $y > \frac{2}{3} - \epsilon x$, for some positive constant $\epsilon < \frac{1}{6}$. Also, a state $(x,y), y < \frac{1}{3} + \delta x$, for some positive constant $\delta < \frac{1}{6}$ is labelled b. It is easy to see that this is a **LMP**, and that the property is true for it — for all states $x \in (0,1]$, we have that $x \models Pr_{>1/3}Xa$.

Now consider the finite approximations to P given intuitively by partitioning $[0,1]^2$ with a finite grid (see figure 1). Each rectangle in the grid is a state, and is labelled a if all states in it are labelled a in P, and similarly for b. The grid also partitions the x-axis, and each interval on it corresponds to a state also. Each such interval has transitions to the rectangles directly above it, with probability given by the height of the rectangle. The start state has a transition to each interval state, with probability given by the size of the interval.

It is easy to check that each grid induces a finite approximation to P. Consider the first grid — here the property is true because no state is labelled b — hence both the interval states satisfy $Pr_{<1/3}Xb$. But now if we refine the grid as in the middle picture, we have a state labelled b, and thus the right interval does not satisfy $Pr_{<1/3}Xb$, so the property ϕ is not satisfied. However refining the grid further makes $Pr_{>1/3}Xa$ be true of this state, so the property is again satisfied. Now if we add some vertical segments on the left half of the grid, we can repeat this, showing that the property is alternately satisfied and not satisfied for this approximation sequence.

As we have commented before, one expects this in the presence of negation. However, this example does not have an explicit negation. One can argue that since we allow both upper and lower bounds on the probabilities appearing in the formula some sort of negation has been smuggled in. However, even if one restricted the logic so that there really was no negation at all (as in our $\mathcal{L}_\vee$) one could still have problems. In such logics the alternation that we have exhibited will not occur but it will be possible to exhibit sequences of approximants such that every approximant satisfies a property and the limit process does not. This is just as bad as alternation. The real message of the example is that we should use the approximation technique to approximate quantitative properties.

4 Approximations and Finite Models

In this section we develop the basic approximation result — for any labelled Markov process $\mathcal{S}$, we build a chain (in the strict simulation ordering) of *finite acyclic labelled Markov processes* (FAMPs for short; finite-state processes in which the transition graph is acyclic) $\{\mathcal{S}_i\}$ such that:

1. For any formula ϕ satisfied by $\mathcal{S}$, there is an i such that $\mathcal{S}_i$ satisfies ϕ — this is a finite model theorem since it gives finite models for any formula

2. $\mathcal{S}_i$ is simulated by $\mathcal{S}$, for all i.

In the final subsection, we also present the example that illustrates the subtleties of logical reasoning in the presence of approximations.

4.1 Finite-state approximation

The key tool in our analysis is the construction of some approximants via an "unfolding" construction. As the approximation is refined there are more and more transitions possible. There are two parameters to the approximation, one is a natural number n, and the other is a positive rational ϵ. The number n gives the number of successive transitions possible from the start state. The number ϵ measures the accuracy with which the probabilities approximate the transition probabilities of the original process.

Given a labelled Markov process $\mathcal{S} = (S, i, \Sigma, \tau)$, an integer n and a rational number $\epsilon > 0$, we define $\mathcal{S}(n, \epsilon)$ to be an n-step unfolding approximation of $\mathcal{S}$. Its state-space is divided into $n + 1$ levels which are numbered $0, 1, \ldots, n$. Recall, from Corollary 2.11, that bisimulation is a fixed point and that one has - for each n - a level n approximation to bisimulation. At each level, say n, the states of the approximant is a partition of S; these partitions correspond to the equivalence classes corresponding to the level n approximation to bisimulation. The initial state of $\mathcal{S}(n, \epsilon)$ is at level n and transitions only occur between a state of one level to a state of one lower level. Thus, in particular, states of level 0 have no outgoing transitions. In the following we omit the curly brackets around singletons.

Definition 4.1 *Let (S, i, Σ, τ) be a labelled Markov process, $n \in \mathbf{N}$ and ϵ a positive rational. We denote the finite-state approximation by $\mathcal{S}(n, \epsilon) = (P, p_0, \rho)$ where P is a subset of $\Sigma \times \{0, \ldots, n\}$. It is defined as follows, for $n \in \mathbf{N}$ and $\epsilon > 0$. $\mathcal{S}(n, \epsilon)$ has $n+1$ levels. States are defined by induction on their level. Level 0 has one state $(S, 0)$. Now, given the sets from level l, we define states of level $l + 1$ as follows. Suppose that there are m states at level l, we partition the interval $[0, 1]$ into intervals of size ϵ/m. Let $(B_j)_{j\in I}$ stand for this partition; i.e. for $\{\{0\}, (0, \epsilon/m], (\epsilon/m, 2\epsilon/m], \ldots\}$. States of level $l + 1$ are obtained by the partition of S that is generated by the sets $\tau_a(\cdot, C)^{-1}(B_j)$, for every set C corresponding to state at level l and every label $a \in \{a_1, \ldots, a_n\}$, $i \in I$. Thus, if a set X is in this partition of S, $(X, l + 1)$ is a state of level $l + 1$. Transitions can happen from a state of level $l + 1$ to a state of level l, and the transition probability function is given by*

$$\rho_a((X, k), (B, l)) = \begin{cases} \inf_{t\in X} \tau_a(t, B)) & \text{if } k = l + 1, \\ 0 & \text{otherwise.} \end{cases}$$

The initial state p_0 of $\mathcal{S}(n, \epsilon)$ is the unique state (X, n) such that X contains i, the initial state of $\mathcal{S}$.

If $B = \cup B_j$, is a (finite and disjoint) union of sets at level l, we will write (B, l) for the set $\{(B_1, l), (B_2, l), \ldots\}$ of all of the corresponding states, and by extension, we will write $\rho_a((X, l + 1), (B, l))$ to mean $\sum_{j\in I} \rho_a((X, l + 1), (B_j, l))$. If $s \in S$, we denote by (X_s, l) the unique state at level l such that $s \in X_s$.

The following lemma is a trivial but useful result. It is a consequence of the construction of finite approximants and uses crucially the fact that the partition of $[0, 1]$ takes into account the number of states m at the preceding level.

Lemma 4.2 *Let $\mathcal{S}$ be a labelled Markov process, and $s \in S$. In $\mathcal{S}(n, \epsilon)$, if B is a finite union of sets appearing at level l, then $0 < \tau_a(s, B) - \rho_a((X_s, l + 1), (B, l)) \leq \epsilon$.*

Proof . Let $(X, l + 1)$, (B_j, l), $j = 1, \ldots, k$ be states of $\mathcal{S}(n, \epsilon)$. Let m be the number of states at level l. Then for all $s, t \in X$ we have

$$|\tau_a(s, B_j) - \tau_a(t, B_j)| < \epsilon/m,$$

because of the way S is partitioned on level $l+1$. Thus we have

$$\begin{aligned} |\tau_a(s,B) - \rho_a((X,l+1),(B,l))| &= |\tau_a(s,B) - \sum_{j=1}^{k} \inf_{t\in X} \tau_a(t,B_j))| \\ &\le \sum_{j=1}^{k} |\tau_a(s,B_j) - \inf_{t\in X} \tau_a(t,B_j))| \\ &\le \sum_{j=1}^{k} \epsilon/m \\ &\le \epsilon. \end{aligned}$$

∎

In a previous version of the approximation result, for example in [DGJP00], the proof of the preceding lemma was not the same because the partition of the state-space at each level was different in two ways: firstly, the partition of $[0,1]$ was done in intervals of size ϵ and secondly, the partition of the state-space of one level was done by considering not only each set of the preceding level but also every union of these sets (hence the lemma was trivial). It was used for the proof of Theorem 4.5, which says that we can reconstruct the original system from all its approximants. Replacing ϵ with ϵ/m allows us to avoid considering union of sets of the construction, and Theorem 4.5 is now proven using a weaker property of the partitions.

It turns out that every state (X,l) in $\mathcal{S}(n,\epsilon)$ is simulated in $\mathcal{S}$ by every state $s \in X$.

Proposition 4.3 *Every labelled Markov process $\mathcal{S}$ simulates all its approximations of the form $\mathcal{S}(n,\epsilon)$. More precisely, every state (X,l) of $\mathcal{S}(n,\epsilon)$ ($l \le n$) is simulated in $\mathcal{S}$ by every $s \in X$.*

Proof . Let $\mathcal{S}(n,\epsilon) = (P, p_0, \rho)$ and $\mathcal{U} = (U, u_0, \Omega, \nu)$ be the direct sum of $\mathcal{S}(n,\epsilon)$ and $\mathcal{S}$. Now let R be the reflexive relation on U relating a state (X,l) from $\mathcal{S}(n,\epsilon)$ to every state $s \in X$ from $\mathcal{S}$. We prove that R is a simulation. Consider two related states, (X,l) and $s \in X$ and let $Y \in \Omega$ be R-closed, that is, $Y \cap S \in \Sigma$ and $R(Y \cap P) \subseteq Y$. The only positive transitions from (X,l) are to states of the form $(B, l-1)$ so we can assume that $Y \cap P$ is a union B of states of level $l-1$. Now observe that $R((B,l-1)) \cap S = B$ and by the preceding lemma we have:

$$\begin{aligned} \nu_a((X,l),(B,l-1)\cup B) &= \rho_a((X,l),(B,l-1) \\ &\le \tau_a(s,B) \\ &= \nu_a(s,(B,l-1)\cup B), \end{aligned}$$

and hence the result. ∎

The next theorem is the main result of this section.

Theorem 4.4 *If a state $s \in S$ satisfies a formula $\phi \in \mathcal{L}_\vee$, then there is some approximation $\mathcal{S}(n,\epsilon)$ such that $(X_s, n) \models \phi$.*

Proof . The proof is by induction on the structure of formulas. We prove the following stronger induction hypothesis. We prove that for all formulas ϕ there is an increasing sequence $(X_n)_{n\ge depth(\phi)}$ of sets in Σ which satisfy:

(i) $\cup_{n \geq depth(\phi)} X_n = [\![\phi]\!]_{\mathcal{S}}$;

(ii) $X_n = \cup_{s \in X_n} C_s$, where $(C_s, l) \in \mathcal{S}(n, 1/2^n)$ and $l \geq depth(\phi)$;

(iii) the states (C_s, l) satisfy ϕ in $\mathcal{S}(n, 1/2^n)$.

It is obvious for T with $X_n = S$ for all n.

Consider $\phi = \phi_1 \wedge \phi_2$. Assume the claim is true for ϕ_j, $j = 1, 2$. Let $(X_n^j)_{n \geq depth(\phi_j)}$ be the sequence for ϕ_j. Now define for $n \geq depth(\phi)$, the sequence

$$X_n = X_n^1 \cap X_n^2.$$

Note that this is an increasing sequence of sets in Σ. We first prove (i): for all $s \models \phi$, there is some n such that $s \in X_n$. Choose $n = \max(n_1, n_2)$ where n_j is such that $s \in X_{n_j}^j$. Now for (ii) and (iii), let $s \in X_n$, for a fixed $n \geq depth(\phi)$. Then because all states (C_s, l) satisfy ϕ_j and $C_s \subseteq X_n^j$, we have $(C_s, l) \models \phi_1 \wedge \phi_2$ and $X_n = \cup_{s \in X_n} C_s$. The proof for the case $\phi_1 \vee \phi_2$ is similar.

Consider $\phi' = \langle a \rangle_q \phi$, and assume the claim is true for ϕ. Let $d = depth(\langle a \rangle_q \phi)$, $\epsilon_n = 1/2^n$ and let $(X_n)_{n \geq d-1}$ be the sequence for ϕ. Now define for $n \geq d$, the sequence

$$Y_n = \cup\{C : (C, d) \in \mathcal{S}(n, \epsilon_n), \text{ and } \forall s \in C, \tau_a(s, X_n) > q + \epsilon_n\}.$$

This is an increasing sequence of sets in Σ because if $(C, d) \in \mathcal{S}(n, \epsilon_n)$ and $C \subseteq Y_n$, then for all $s \in C$ we have $\tau_a(s, X_{n+1}) \geq \tau_a(s, X_n) \geq q + \epsilon_n$. Moreover, if (C', d) is a state of $\mathcal{S}(n, \epsilon_{n+1})$ and $s, t \in C'$, then $\tau_a(t, X_{n+1}) > \tau_a(s, X_{n+1}) - \epsilon_{n+1} \geq q + \epsilon_n - \epsilon_{n+1} = q + \epsilon_{n+1}$.

We now prove (i), that is, for all $s \models \phi'$, there is some n such that $s \in Y_n$. So assume $\tau_a(s, [\![\phi]\!]) > q$. Then there is some n such that $\tau_a(s, X_n) - q > 2\epsilon_n$ because $\tau_a(s, \cdot)$ is a measure and X_n is an increasing sequence which converges to $[\![\phi]\!]$ and ϵ_n $(= 1/2^n)$ is decreasing to 0. Now since X_n is a union of states of level $l - 1 \geq d - 1$, then for every $t \in C_s$, with (C_s, l) a state of $\mathcal{S}(n, \epsilon_n)$ we have

$$|\tau_a(s, X_n) - \tau_a(t, X_n)| < \epsilon_n$$

and hence $\tau_a(t, X_n) - q > \epsilon_n$. Thus $C_s \subseteq Y_n$ and (i) and (ii) are proved. Note that the inequality sign in the meaning of the modal formula was crucial to this part of the proof.

We now prove (iii). Let $s \in Y_n$, for a fixed $n \geq d$. Then because all states $(X, l-1)$, where $X \subseteq X_n$ and $l - 1 \geq d - 1$, satisfy ϕ and by Lemma 4.2, we have

$$\begin{aligned} \rho_a((C_s, l), ([\![\phi]\!]_{\mathcal{S}(n,\epsilon_n)}, l-1)) &\geq \rho_a((C_s, l), (X_n, l-1)) \\ &\geq \tau_a(s, X_n) - \epsilon_n \\ &> q + \epsilon_n - \epsilon_n = q, \end{aligned}$$

and hence, $(C_s, l) \models \phi'$ for all $l \geq d$ as wanted in (iii). ∎

We now show that one can reconstruct the original process –more precisely a bisimulation equivalent of the original process– from the approximants $\mathcal{S}(n, \epsilon)$. We do not reconstruct the original state space but we reconstruct all the transition probability information, i.e., the dynamical aspects of the process.

Theorem 4.5 *Let (S, i, Σ, τ) be a labelled Markov process that is maximally collapsed, that is, $S = S/_{\approx}$. If we are given all finite-state approximations $\mathcal{S}(n, \epsilon)$, we can recover (S, i, Σ, τ).*

Proof . We can recover the state space trivially by taking the union of states at any level of any approximation. We know, from the fact that $\mathcal{S}$ is maximally collapsed, that Σ is generated by the sets of the form $[\![\phi]\!]$, by the quotient theorem of [DEP98]. Moreover, in the proof of Theorem 4.4 we prove that for every $\mathcal{S}$ and every formula ϕ, there exist states in some approximation $\mathcal{S}(n, 1/2^n)$ such that the union of the sets X_n representing these states is exactly the set of states in $\mathcal{S}$ that satisfy ϕ; i.e. $\cup_{n\geq l} X_n = [\![\phi]\!]_{\mathcal{S}}$. Those two facts imply that

$$\mathcal{B} := \{B : (B, l) \in \mathcal{S}(l, 1/2^n) \text{ for } l, n \in \mathbf{N}\}$$

generates Σ (obviously, $\mathcal{B} \subseteq \Sigma$).

The main difficulty is that we have to recover the transition probability function. To do so, let $\mathcal{F}(\mathcal{B})$ be the set containing finite unions of sets in $\mathcal{B}$. We first argue that $\mathcal{F}(\mathcal{B})$ forms a field, then we define $\nu_a(s, \cdot)$ on it and we show that $\nu_a(s, \cdot)$ and $\tau_a(s, \cdot)$ agree on it for all $s \in S$. This will imply that $\nu_a(s, \cdot)$ is finitely additive on $\mathcal{F}(\mathcal{B})$ and hence that it can be extended uniquely to a measure on Σ, and hence that ν_a and τ_a agree on $S \times \Sigma$, as desired.

We now show that $\mathcal{F}(\mathcal{B})$ forms a field. It is obviously closed under finite unions. To see that it is also closed under intersection and complementation, note that if $(C, n) \in \mathcal{S}(n, \epsilon)$, then for all $m > n$ and all δ such that ϵ is an integer multiple of δ, C is a union of a family of sets C_i such that $(C_i, m) \in \mathcal{S}(m, \delta)$. This is clear from the fact that the construction proceeds by splitting existing blocks. From this observation it is clear that we have closure under intersections because given two sets we can move to a stage of the approximation process that refines both of them. At this stage the intersection of the two sets that we started out with is clearly a union of the sets at the present stage.

Now let $C \in \mathcal{F}(\mathcal{B})$, $s \in S$, $a \in \mathcal{A}$ and let

$$\nu_a(s, C) := \bigsqcup_{n,\epsilon} \sum_{\substack{B \subseteq C \\ (B, n-1) \in \mathcal{S}(n,\epsilon)}} \rho_a((X_s, n), (B, n-1)).$$

We prove that $\nu_a(s, \cdot)$ and $\tau_a(s, \cdot)$ agree on $\mathcal{F}(\mathcal{B})$ for all $s \in S$. Obviously, $\nu_a(s, C) \leq \tau_a(s, C)$ for $C \in \mathcal{F}(\mathcal{B})$. The reverse inequality follows from Lemma 4.2:

$$\begin{aligned}
\bigsqcup_{n,\epsilon} \sum_{\substack{B \subseteq C \\ (B, n-1) \in \mathcal{S}(n,\epsilon)}} \rho_a((X_s, n), (B, n-1)) &= \bigsqcup_{n,\epsilon} \rho_a((X_s, n), (\cup B, n-1)) \\
&\geq \bigsqcup_{n,\epsilon} (\tau_a(s, \cup B) - \epsilon) \\
&\geq \bigsqcup_{(n,\epsilon) \in I} (\tau_a(s, C) - \epsilon) \\
&= \tau_a(s, C),
\end{aligned}$$

where I is the set of pairs (n, ϵ) such that in $\mathcal{S}(n, \epsilon)$, level n contains a partition of C (note that there are arbitrary small ϵ's that are involved in I). This concludes the proof that ν and τ agree and we are done. ∎

4.2 Example

We compute a few approximations of a simple continuous process. States are from the set $\{s, t\} \cup [0, 3]$, the initial state is 1 and transitions are as follows:

- if $x \in [0,1]$, $p_a(x,[0,y)) = \frac{x+y}{4}$, where $0 \le y \le 1$.
  $p_a(x,\{1\}) = \frac{1-x}{4}$,
  $p_a(x,(1,1+y]) = \frac{y}{4}$,
  $p_a(x,(2,2+y]) = \frac{xy}{4}$,
- if $x \in (1,2]$, $p_a(x,s) = 1$.
- if $x \in (2,3]$, $p_b(x,t) = 1$.

We can see that this process has indeed a continuous state-space because every state x in $[0,1]$ has a different probability, $x/4$, of making an a-transition to the equivalence class $[2,3]$.

We draw this process in an informal way in Figure 2, where we label the transitions with expressions that should be interpreted as above.

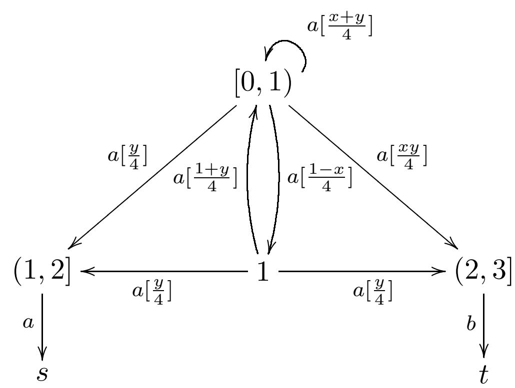
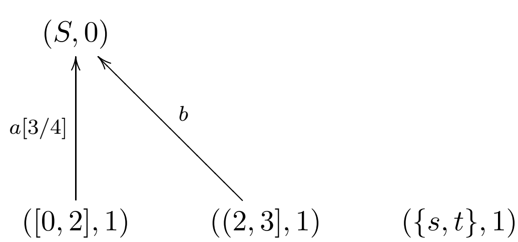


Figure 2: A simple continuous process

Let us compute the approximation $\mathcal{S}(2,1/2)$. At level 0, we have state $(S,0)$. At level 1, we partition S according to the partition of $[0,1]$ into intervals of size $1/2$: $\{\{0\},(0,1/2],\,(1/2,1]\}$. Note that if $x \in [0,1]$, then $p_a(x,S) = \frac{x+1}{4} + \frac{1-x}{4} + \frac{1}{4} + \frac{x}{4} = \frac{3+x}{4}$. Hence

$$p_a(x,S)\begin{cases} = 0 & \text{if } x \notin [0,2] \\ \in (1/2,1] & \text{if } x \in [0,2] \end{cases} \qquad p_b(x,S) = \begin{cases} 0 & \text{if } x \notin (2,3] \\ 1 & \text{if } x \in (2,3]. \end{cases}$$

This yields, for level 1, the sets $[0,2]$, $(2,3]$ and $\{s,t\}$. The transitions are as in the following picture which represents the approximation $\mathcal{S}(1,1/2)$.



The initial state is $([0,2],1)$ and from the picture, one can see that this state satisfies the formula $\langle a\rangle_{3/4-\epsilon}\mathsf{T}$, for all $\epsilon > 0$.

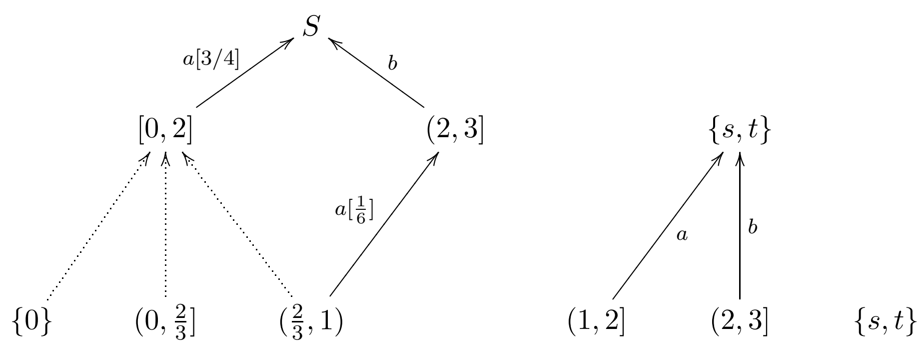
Now for level two of $\mathcal{S}(2, 1/2)$, the partition of S will be obtained using all transitions to any set that appears at level 1. We must consider the partition of $[0, 1]$ into intervals of size $1/2 \times 1/3$ since there are 3 states at level 1.

$$p_a(x, \{s,t\}) = \begin{cases} 0 & \text{if } x \notin (1,2] \\ 1 & \text{if } x \in (1,2] \end{cases} \qquad p_b(x, \{s,t\}) = \begin{cases} 0 & \text{if } x \notin (2,3] \\ 1 & \text{if } x \in (2,3] \end{cases}$$

Now for $x \in [0, 1]$, we have

$$p_a(x, [0,2]) = \frac{x+1}{4} + \frac{1-x}{4} + \frac{1}{4} = 3/4$$
$$p_a(x, (2,3]) = x/4 \in \begin{cases} \{0\} & \text{if } x = 0 \\ (0, 1/6] & \text{if } x \in (0, 2/3] \\ (1/6, 2/6] & \text{if } x \in (2/3, 1], \end{cases}$$

Thus we get the following sets constituting the partition of S at level 2: $\{0\}$, $(0, 2/3]$, $(2/3, 1]$, $(1, 2]$, $(2, 3]$, $\{s, t\}$. Transitions are illustrated in the following picture, where we omit the levels. In order not to clutter up the picture, we have not labelled the dotted lines. Dotted lines represent a-transitions with probability $3/4$, so should be labelled $a[\frac{3}{4}]$.



This picture represents the approximation $\mathcal{S}(2, 1/2)$. The initial state is $((2/3, 1], 2)$ and from the picture, it is easy to see that this state satisfies the following formulas where $\epsilon > 0$.

$$\langle a \rangle_{\frac{11}{12}-\epsilon} \mathsf{T}, \qquad \langle a \rangle_{3/4-\epsilon} \langle a \rangle_0 \mathsf{T}, \qquad \langle a \rangle_{1/6-\epsilon} \langle b \rangle_0 \mathsf{T}.$$

This implies that in the original process, the initial state also satisfies these formulas.

We can obtain more complex formulas by studying approximation $\mathcal{S}(3, 1/2)$. The partition of S that is generated isolates state 1 of the original process. We draw part of this approximation in Figure 3, keeping only the initial state at level 3. Here again, dotted lines represent a-transitions with probability $3/4$. Transitions from the initial state $(\{1\}, 3)$ are all a-transitions.

In this picture we see that the initial state satisfies in particular the following formulas, where $\epsilon > 0$:

$$\langle a \rangle_{1-\epsilon} \mathsf{T}, \qquad \langle a \rangle_{1/2-\epsilon} \langle a \rangle_0 \langle a \rangle_0 \mathsf{T}, \qquad \langle a \rangle_{1/12-\epsilon} \langle a \rangle_0 \langle b \rangle_0 \mathsf{T}.$$

These formulas are also satisfied by the initial state of the original process.

Note that the set $(1, 2]$ will never be split in any approximations, for it contains only bisimilar states. By changing the probability of state $x \in (1, 2]$ of jumping with label a to state s to the value $x - 1$, one can get a more complex process.

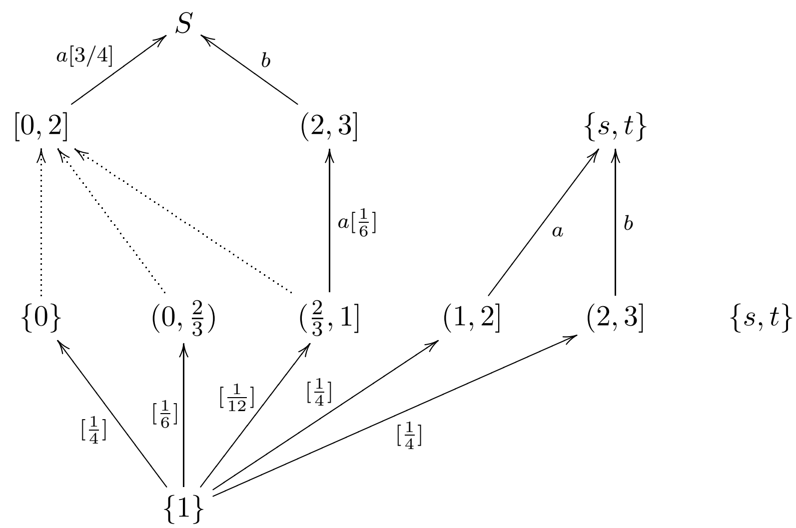


Figure 3: Approximation $\mathcal{S}(3, 1/2)$

4.3 A countable basis for labelled Markov processes

The space of all labelled Markov processes appears too large to be used in a computational way. In fact this space has a countable subset – the rational trees defined below – which serves to approximate all labelled Markov processes. In the domain theory section following, we will show that the collection of labelled Markov processes forms an ω-continuous dcpo and that with the Lawson topology one gets a Polish space. Here we will derive similar results and will justify the approximation in terms of the logic.

In the construction of approximations of Section 4.1, the finite processes we construct have the following special structure. The transition graph is a DAG (directed acyclic graph) and the states are partitioned into levels. The transitions always go from one level to the next and never go to greater depths.

In fact one can just use processes where the transition graph is a tree and with the states partitioned into levels as above. Such finite processes with rational transition probabilities play a special role. We examine their properties below. For brevity we will just say "rational tree" when we mean a finite-state process with a tree-like transition graph and rational transition probabilities.

Lemma 4.6 *Let $\mathcal{T}$ be a rational tree that is strictly simulated by a labelled Markov process $\mathcal{S}$. Then there is a finite approximation $\mathcal{S}(n,\epsilon)$ strictly simulating $\mathcal{T}$.*

Proof . Let R_ϵ be the strict simulation between $\mathcal{T} = (T, t_0, \theta)$ and $\mathcal{S}$. Here we use the definition of simulation from 2.18. Consider $\mathcal{S}(n, \epsilon/4)$, where n is the height of $\mathcal{T}$. We assume that $\mathcal{T}$ only involves labels $a_1, \dots, a_n$, but the proof can be adapted easily if it is not the case, because for sure $\mathcal{T}$ only involves a finite number of labels. We first extend R_ϵ to R' in the following way. Let $t \in T$ be at level l, and let $s \in S$. Let R' be the transitive closure of the relation that relates t and s if there is some $s' \in B$ such that (B, l) is a state of $\mathcal{S}(n, \epsilon/4)$ and $tR_\epsilon s'$. We prove that R' is a strict simulation between $\mathcal{T}$ and $\mathcal{S}$. Observe that R' coincides with R_ϵ on both $\mathcal{T}$ and $\mathcal{S}$. Let $sR't$ and let $Y \subseteq T \cup S$ be an R'-closed set such that $Y \cap S \in \Sigma$. Then it is also R_ϵ-closed and $Y \cap S$ is a union of sets at level $l-1$ of $\mathcal{S}(n,\epsilon/4)$. Hence we have the following:

$$\begin{array}{rcl}\theta_a(t, Y\cap T) & < & \tau_a(s', Y\cap S)-\epsilon \\ & \leq & \tau_a(s, Y\cap S)+\epsilon/4-\epsilon \\ & = & \tau_a(s, Y\cap S)-3\epsilon/4\end{array}$$

because s and s' belong to the same set of level l. We have proved that R' is a $3\epsilon/4$-strict simulation between $\mathcal{T}$ and $\mathcal{S}$.

We now define the relation W between $\mathcal{T}$ and $\mathcal{S}(n,\epsilon/4)$. Let W be the transitive closure of the reflexive relation which contains the restriction of R' to $\mathcal{T}$ and relates t and (X,l) if t is at level l in $\mathcal{T}$ and $tR's$ for some $s\in X$. Observe that W coincides with R' on $\mathcal{T}$ and is the identity relation on $\mathcal{S}(n,\epsilon/4)$. Now take two W-related states $t\in T$ and $(X,l)\in\mathcal{S}(n,\epsilon/4)$. Let $Y\subseteq T\cup P$ be W-closed. Let $(B,l-1)$ be the "set" formed by taking the union of sets of the states of $Y\cap\mathcal{S}(n,\epsilon/4)$ restricted to level $l-1$ in $\mathcal{S}(n,\epsilon/4)$. Then B is obviously measurable in $\mathcal{S}$ and $(Y\cap T)\cup B$ is R'-closed. Then if s is the state in X such that $tR's$,

$$\begin{array}{rcl}\theta_a(t, Y\cap T) & < & \tau_a(s, B)-3\epsilon/4 \\ & \leq & p_a((X,l),(B,l-1))+\epsilon/4-3\epsilon/4 \\ & < & p_a((X,l), Y\cap\mathcal{S}(n,\epsilon/4))-\epsilon/2\end{array}$$

by Lemma 4.2. Thus we are done. ∎

The following theorem shows that rational trees that are strictly simulated by a process form a directed set.

Theorem 4.7 *Let $\mathcal{T}$ and $\mathcal{T}'$ be two rational trees that are strictly simulated by a labelled Markov process $\mathcal{S}$. Then there is a rational tree which is strictly simulated by $\mathcal{S}$ and also strictly simulates both $\mathcal{T}$ and $\mathcal{T}'$.*

Proof . Let $\mathcal{T}=(T,t_0,\theta)$ and $\mathcal{T}'=(T',t_0',\theta')$ be strictly simulated by $\mathcal{S}$ by relations R and R'. Then by Lemma 4.6, there are $n,n'\in\mathbf{N}$ and $\delta,\delta',\epsilon,\epsilon'>0$ such that $\mathcal{T}$ is δ-strictly simulated by $\mathcal{S}(n,\epsilon)=(P,p_0,\mathcal{P}(P),\rho)$, and similarly for $\mathcal{T}'$. We choose $\epsilon^*<\delta,\delta'$ such that both ϵ and ϵ' are integral multiples of ϵ^* and let $n^*=\max(n,n')$.

We show that $\mathcal{T}$ and $\mathcal{T}'$ are strictly simulated by $\mathcal{S}(n^*,\epsilon^*)$. Since ϵ is an integral multiple of ϵ^*, the partition at any level l of $\mathcal{S}(n^*,\epsilon^*)$ is a refinement of the partition at level l of $\mathcal{S}(n,\epsilon)$. Let W be the reflexive relation on $T\cup P^*$ which coincides with R on $\mathcal{T}$ and relates t and (X,l) if $tR(B,l)$ for some $(B,l)\in P$ such that $X\subseteq B$. Let $Y\subseteq T\cup P^*$ be W-closed. Then $(Y\cap T)\cup R(Y\cap T)$ is R-closed and $R(Y\cap T)$ is measurable in $\mathcal{S}$ when considered as a set in $\mathcal{S}$. We now prove that $R(Y\cap T)$ is included in $Y\cap P^*$ when they are considered as sets in $\mathcal{S}$. If $(B,l)\in R(Y\cap T)$, then there is some $t\in Y\cap T$ such that $tR(B,l)$ which implies that for all $(X,l)\in\mathcal{S}(n^*,\epsilon^*)$ such that $X\subseteq B$, $(X,l)\in Y\cap P^*$. Since sets of level l in $\mathcal{S}(n^*,\epsilon^*)$ form a refinement of sets of level l in $\mathcal{S}(n,\epsilon)$, we have $B\subseteq Y\cap P^*$ ($Y\cap P^*$ considered as a set in $\mathcal{S}$).

Now let $s\in X\subseteq B$.

$$\begin{array}{rcl}\theta_a(t, Y\cap T) & < & \rho_a((B,l), R(Y\cap T))-\delta \\ & \leq & \tau_a(s, R(Y\cap T))-\delta \\ & \leq & \tau_a(s, Y\cap P^*)-\delta \\ & \leq & \rho_a^*((X,l), Y\cap P^*)+\epsilon^*-\delta \\ & < & \rho_a^*((X,l), Y\cap P^*),\end{array}$$

as wanted. A similar proof shows that $\mathcal{T}'$ is strictly simulated by $\mathcal{S}(n^*,\epsilon^*)$.

We now construct a rational tree, call it $\mathcal{T}^*$, from $\mathcal{S}(n^*, \epsilon^*)$. The initial state t_0^* is p_0, which is a state at level n in $\mathcal{S}(n^*, \epsilon^*)$. In order to obtain a tree we have to ensure that every state of $\mathcal{T}^*$ has only one incoming transition. For every level strictly below n, in $\mathcal{S}(n^*, \epsilon^*)$, we duplicate the states in such a way that no two transitions arrive at the same state and we take the resulting set T^* to be the set of states of $\mathcal{T}^*$. We define θ^*, the transition function of $\mathcal{T}^*$, by decreasing every probability associated with a transition of $\mathcal{S}(n^*, \epsilon^*)$ to a rational number below it in such a way that $\mathcal{T}^*$ will be strictly simulated by $\mathcal{S}(n^*, \epsilon^*)$ and will still strictly simulate $\mathcal{T}$ and $\mathcal{T}'$. ∎

This theorem is one of the properties that follows from the fact that labelled Markov processes form a continuous domain. In the domain-theoretic treatment we show that this result follows from the fact that labelled Markov processes can be described by the solution of a recursive domain equation in a suitable category of ω-continuous posets. The proof just above does not use any domain-theoretic machinery.

Lemma 4.8 *Given any process of the form $\mathcal{S}(n, \epsilon)$ we can construct a sequence of rational trees $\mathcal{T}_i$ such that $\mathcal{T}_i$ is strictly simulated by $\mathcal{T}_{i+1}$ and all of them are strictly simulated by $\mathcal{S}(n, \epsilon)$. Moreover, the sequence converges to $\mathcal{S}(n, \epsilon)$ in the sense that the sets of formulas satisfies by $\mathcal{T}_i$ converges to the set of formulas satisfied by $\mathcal{S}(n, \epsilon)$.*

Proof . By duplicating states of $\mathcal{S}(n, \epsilon)$ as in the previous proof, and keeping only the initial state from level n, we get a tree $\mathcal{T}$ which is bisimilar to $\mathcal{S}(n, \epsilon)$. Now consider a family of trees with the same shape as $\mathcal{T}$ and with the probabilities chosen to be rational and to converge to the probabilities occurring in $\mathcal{T}$. We can always choose these numbers to be strictly increasing. Thus we get immediately that the strict simulation relation holds.

Since the trees in question have the same shape as $\mathcal{T}$, it is easy to see that the family of rational trees converges in the logic to $\mathcal{S}(n, \epsilon)$. ∎

Theorem 4.9 *Given any labelled Markov process $\mathcal{S}$ there is a directed set of rational trees $\mathcal{T}_i$ with each $\mathcal{T}_i$ being strictly simulated by $\mathcal{S}$ and such that any logical formula satisfied by $\mathcal{S}$ is satisfied by some $\mathcal{T}_i$.*

Proof . The system $\mathcal{S}$ is approximated by a family $\mathcal{S}(n, \epsilon)$ in such a way that any formula ϕ satisfied by $\mathcal{S}$ is satisfied by one of the $\mathcal{S}(n, \epsilon)$, call it $\mathcal{P}$. Using lemma 4.8 we see that we can find sequences of rational trees converging in the logical sense to each of the $\mathcal{S}(n, \epsilon)$. Now if we look at the sequence of rational trees approximating $\mathcal{P}$ we can find one that satisfies ϕ. In view of theorem 4.7 we can see that the collection of all these rational trees forms a directed set. ∎

5 A domain of processes

So far we have defined processes in terms of standard concepts of probability theory. In this section we describe how to view the collection of processes as a dcpo. The domain of processes will be constructed by solving a recursive domain equation giving a dcpo very close, in spirit, to the domain of synchronization trees constructed by Abramsky [Abr91]. We show that the order in the domain corresponds to simulation, the way-below relation corresponds to strict simulation and equality corresponds to bisimulation. Finally, we show that simulation in the domain is characterized by $\mathcal{L}_\vee$.

5.1 Background

5.1.1 Basic Definitions and Results

Definition 5.1 *A partial order is a **dcpo**, if it is closed under limits of directed sets.*

We will only consider dcpos with a bottom element.

Definition 5.2 *$b \ll x$ ("b is **way-below** x") if for all directed sets X such that $x \sqsubseteq \bigsqcup X$, $(\exists x_i \in X)\ b \sqsubseteq x_i$.*

Definition 5.3 *A dcpo D is **continuous**, if for all $d \in D$, the set $\{b \mid b \ll d\}$ is directed and has lub d. It is ω-**continuous**, if there is a countable subset B such that for all $d \in D$, the set $\{b \in B \mid b \ll d\}$ is directed and has lub d. Such a set B is called a **basis**.*

Definition 5.4 *$b\Uparrow \stackrel{d}{=} \{x \mid b \ll x\}$. $(b)\uparrow \stackrel{d}{=} \{x \mid b \sqsubseteq x\}$.*

Our analysis will rest on the topological properties of continuous dcpos.

Definition 5.5 *The **Scott topology** on a dcpo D, written σ_D, consists of all sets U satisfying $(U)\uparrow = U$ and for all directed sets $X \subseteq D$, $\bigsqcup X \in U$ implies $X \cap U \neq \emptyset$. In any ω-continuous dcpo, $\{b\Uparrow \mid\ b \in$ basis for $D\}$ is a base for the Scott topology.*

5.1.2 Valuations and the probabilistic powerdomain

The Scott topology allows us to define valuations as continuous functions, on the lattice σ_D, that satisfy modularity.

Definition 5.6 *A valuation ν on a dcpo D is a monotone and continuous function from $(\sigma_D, \subseteq)$ to $[0,1]$ that satisfies:*

$$(\forall U, V \in \sigma_D\ [\nu(U \cup V) = \nu(U) + \nu(V) - \nu(U \cap V)]$$

The following special valuation play an important role.

Definition 5.7 *For any $x \in D$, the **point valuation** η_x is 1 for all Scott-opens that contain x, and 0 for all others.*

Definition 5.8 *The **probabilistic powerdomain** of D, written $\mathcal{P}_{\mathsf{Pr}}(D)$, is the set of all valuations on D ordered by $\nu \sqsubseteq \mu \Leftrightarrow\ (\forall U \in \sigma_D)\ [\nu(U) \leq \mu(U)]$.*

If D is ω-continuous, so is $\mathcal{P}_{\mathsf{Pr}}(D)$ with a countable basis given by valuations of the form:

$$r_1 \times \eta_{x1} + \ldots + r_n \times \eta_{xn}$$

where r_i are rationals.

There is a unique extension of valuations to measures on the Borel sets associated with the Scott topology. For ω-continuous dcpos, see [SD80, Jon90, AMESD98, Law82] for the extension theorems.

5.1.3 The Lawson topology

The Scott topology meshes closely with the order structure. However, it has weak separation properties and does not capture *by itself* all properties of interest. The closely related Lawson topology is often needed; roughly speaking the Lawson topology gives "negative" information.

Definition 5.9 *The **Lawson topology** on a dcpo D, written $\lambda(D)$, is generated by the base $U \setminus (F)\uparrow$, where U is Scott open, and F is a finite subset of D. In any ω-continuous dcpo, a countable base for the Lawson topology is given by $\{b\Uparrow \setminus (\{b_1, \ldots, b_n\})\uparrow \mid\ b, b_i \in$ a basis for $D\}$.*

As an example of the Lawson topology consider the domain of streams over a finite alphabet with the prefix ordering. This is an ω-algebraic dcpo. The Lawson topology in this case coincides with the topology induced by the following metric. Given two streams s and t we look for the first position, say the nth, where s and t differ. We then define $d(s,t) := 2^{-n}$. Now the sequence of streams $0^n 1^\infty$ will converge in this metric - hence, in the Lawson topology, to 0^∞. In the Scott topology the sequence of streams $0^n 1^\infty$ is not convergent.

For ω-continuous dcpos D, the sets $(b)\uparrow$ are G_δ in σ_D. Thus, in this case, the Borel algebra generated by $\lambda(D)$ is the same as that generated by σ_D. The Lawson topology on ω-continuous dcpos is separable and metrizable; the Scott topology is T_0 and usually not even T_1. In the case that the Lawson topology is also compact, we get a Polish space for ω-continuous dcpos[10].

We now explore some consequences of Lawson compactness. First, the following result (and proof) from [Jun88] relates Scott compactness and Lawson closedness.

Lemma 5.10 *In a ω-continuous dcpo, every Scott-compact upper set A is Lawson closed and is expressible as the countable intersection of Scott open sets.*

Proof . Consider any Scott-open neighborhood $O \supseteq A$. Each element $x \in A$ is contained in some basic Scott open of the form $b\Uparrow, b \in O$. These sets form a cover for A. By Scott compactness, there is a finite subcover of such sets. Thus, for each $O \supseteq A$, there is a finite collection of b_i such that $A \subseteq \cup_{i=1}^{n} b_i\Uparrow \subseteq O$. However, $b_i \in O$, so $(b_i)\uparrow \subseteq O$, so we also have $A \subseteq \cup_{i=1}^{n}(b_i)\uparrow \subseteq O$. However, for any upper set A, $A = \cap O, O \supseteq A$ with O Scott open. Hence $A = \cap_O \cup_{i=1}^{n_O} b_i\Uparrow$, and $A = \cap_O \cup_{i=1}^{n_O} (b_i)\uparrow$.

Now $(b_i)\uparrow$ is a (basic) Lawson-closed set, hence A is also Lawson closed. From ω-continuity, there are only countably many collections of the form $\cup_{i=1}^{n_O} b_i\Uparrow$. ∎

If D is Lawson compact, the ordering relation on valuations can be characterized in terms of upper sets. More precisely, the ordering relation between the induced measures on upper sets can be characterized in terms of the ordering on valuations. In the following lemma we will write ν for a valuation and $\hat{\nu}$ for the induced measure on the Borel sets[11]. Thereafter we will revert to using the same symbol for the valuation and its induced measure.

Lemma 5.11 *Let D be Lawson compact. Let $\nu_1, \nu_2 \in \mathcal{P}_{\mathsf{Pr}}(D)$.*

1. $\nu_1 \sqsubseteq \nu_2 \Rightarrow$ *($\forall$ upper Borel A).*$[\hat{\nu_1}(A) \leq \hat{\nu_2}(A)]$
2. *($\forall$ Lawson closed upper A)*$[\hat{\nu_1}(A) \leq \hat{\nu_2}(A)] \Rightarrow \nu_1 \sqsubseteq \nu_2$

Proof .

1. If A is a Scott compact upper set, then by Lemma 5.10 it is the countable intersection of Scott open sets O_i. Let $U_k = \cap_{i=1}^{k} O_i$. Then $(U_k)_{k \in \mathbf{N}}$ is a nested sequence of Scott-open sets decreasing to A. Thus $\hat{\nu_j}(A) = \inf_i \hat{\nu_j}(U_i)$, for $j = 1,2$ because $\hat{\nu_j}, j = 1,2$ are measures. But, since the U_i are Scott opens, we have

$$\hat{\nu_1}(U_i) = \nu_1(U_i) \leq \nu_2(U_i) = \hat{\nu_2}(U_i),$$

and we get - after taking infs - that $\hat{\nu_1}(A) \leq \hat{\nu_2}(A)$, for Scott-compact upper sets A.

For general upper sets A we proceed as follows. Since the domain is a metrizable space, we know [Par67] that $\hat{\nu_j}(A) = \bigsqcup\{\hat{\nu_j}(C) \mid C \text{ Lawson-closed}, C \subseteq A\}$ for $j = 1,2$. Now if C is closed it is Lawson compact,

[10] Recently Keye Martin has shown that one can drop the Lawson compactness condition.

[11] Recall that the Borel sets are the same for the Scott topology and the Lawson topology.

since it is a closed subset of a Lawson-compact space. Thus, it is also Scott-compact (the Scott topology has fewer sets). Thus its upper set is also Scott compact. Now we claim that

$$\bigsqcup\{\hat{\nu_j}(C) \mid \ C \text{ Lawson-closed}, C \subseteq A\} =$$
$$\bigsqcup\{\hat{\nu_j}(C) \mid C \text{ is a Scott-compact upper set}, C \subseteq A\}.$$

To see this, observe that since every Scott-compact set is Lawson closed (Lemma 5.10), the right hand side $\leq$ the left hand side. Conversely, for every element in the left hand side, there is a bigger element in the right hand side (namely its upclosure). Hence the two must be equal.

But since we know the result for all Scott-compact upper sets, we have our result in the general case.

2. Let ($\forall$ Lawson closed upper C)$[\hat{\nu_1}(C) \leq \hat{\nu_2}(C)]$. Since $\lambda(D)$ is metrizable, we have for all Scott opens $O : \nu_j(O)\hat{\nu_j}(O) = \bigsqcup\{\hat{\nu}(C) \mid C \subseteq O,\ C \text{ Lawson closed}\}, j = 1, 2$. Since D is Lawson compact, every closed C is also Lawson compact, and hence Scott compact. Thus, its upper set is also Scott compact, and by lemma 5.10 Lawson closed. Thus, we have: $\nu_j(O) = \bigsqcup\{\hat{\nu}(C) \mid C \subseteq O,\ C \text{ upper and Lawson closed}\}, j = 1, 2$. The result now follows from the assumption on upper Lawson-closed sets.

∎

Lawson compactness is stable under inverse limits. The proof of the following lemma was outlined by A. Jung[12].

Lemma 5.12 (A. Jung) *Lawson compact, ω-continuous dcpo's are closed under inverse limits.*

Proof . Let $\{(D_i, f_i, g_i)\}$ be an inverse-limit system, i.e.

$$\begin{aligned} f_i &: D_{i+1} \longrightarrow D_i \\ g_{i+1} &: D_i \longrightarrow D_{i+1} \\ f_i \circ g_{i+1} &= 1 \\ g_{i+1} \circ f_i &\sqsubseteq 1 \end{aligned}$$

Here all the f_i, g_i are Scott continuous, monotone functions. However, one can immediately show that f_i must be Lawson continuous – one only needs to check that $f^{-1}((x)\uparrow)$ is Lawson closed, since f^{-1} respects all set operations. However, this is immediate since $f^{-1}((x)\uparrow) = (g(x))\uparrow$, which is Lawson-closed.

D, the inverse limit, is the subspace of $\Pi_i D_i$, given by sequences of the form $\langle d_i \rangle$, where $\forall i. d_i \in D_i, f_i(d_{i+1}) = d_i$. D_i embeds in D via $e_i(x) = \langle d_j \rangle$ where $d_i = x$, $d_j = f_j \circ \ldots \circ f_i(x)$, if $j < i$ and $d_j = g_j \circ \ldots \circ g_{i+1}(x)$ if $j > i$. D has a countable basis given by $\cup_i\{e_i(x_i) \mid x_i \text{ is a basis element of } D_i\}$.

The Lawson topology on the inverse limit is the subspace topology inherited from the product topology of all the domains in the diagram. If all D_i's are compact Hausdorff then so is the product (Tychonoff). So, we are done if we prove that the sequences which constitute the elements of the bilimit form a closed subset, which would imply that the subspace they form is also compact. Equivalently, we are done if we show that D^c (the complement of D in $\Pi_i D_i$) is open. However, $D^c = \cup_i X_i$, where $X_i = (\Pi_{j<i} D_j) \times \{\langle x, y \rangle \mid f_i(y) \neq x\} \times (\Pi_{j>i+1} D_j)$ So, we are done if we prove that $\{\langle x, y \rangle \mid f_i(y) \neq x\}$ is open in $D_i \times D_{i+1}$. Since the space D_i is Hausdorff in the Lawson topology, for every $x \neq f_i(y)$ we have disjoint open sets $A_{xy}, B_{xy} \subseteq D_i$ such that $x \in A_{xy}$, $f_i(y) \in B_{xy}$. Thus, the given set is $\bigcup_{xy} A_{xy} \times f_i^{-1}(B_{xy})$, and thus is open, as f_i is Lawson continuous.

∎

[12]The authors accept responsibility for any errors and typos in this rendition.

5.2 The domain *Proc*

We fix a (countable) set L of labels and use the Jones-Plotkin probabilistic powerdomain [JP89, Jon90]. For notational convenience, we write $\mathsf{L} \longrightarrow D$ for the product $\prod_{\mathsf{L}} D$ indexed by the set of labels. Processes are given by the recursive domain equation:

$$Proc = \mathsf{L} \longrightarrow \mathcal{P}_{\mathsf{Pr}}(Proc).$$

We will write $\sqsubseteq$ for the partial order in the domain.

Proposition 5.13 *The domain equation*

$$Proc = \mathsf{L} \longrightarrow \mathcal{P}_{\mathsf{Pr}}(Proc)$$

can be solved in the category of ω-continuous, Lawson-compact dcpos.

Proof . By [JT98] the probabilistic powerdomain is closed on Lawson-compact, ω-continuous dcpos. We have already showed closure of ω-continuous, Lawson-compact dcpos under inverse limits. Thus the domain equation can be solved in this category using standard techniques [SP82]. ∎

This domain can be viewed as a single "universal" labelled Markov process. The next few results show how to define transition probabilities in order to do this. To start with, we have the transition probabilities to Scott-open sets given by the definition of the Jones-Plotkin powerdomain. In section 6 we show how to extend these results to obtain the transition probabilities to arbitrary measurable sets (i.e. the Borel sets generated by the Lawson topology.)

Definition 5.14 $\tau_a(p, U) \stackrel{d}{=} p(a)(U)$ *for* $p \in Proc$, *and* U *a Scott-open set in Proc.*

Lemma 5.15 $\tau_a(., U)$ *is upper semicontinuous for each Scott-open* U.

Proof . We need to show that $[\tau_a(., U)]^{-1}(r, 1]$ is a Scott-open set. In order to show that it is upper closed we proceed as follows. Let $p \sqsubseteq q$, and $\tau_a(p, U) > r$. Now we have $\tau_a(q, U) = q(a)(U) \geq p(a)(U) > r$.

In order to show the remaining property of Scott opens we let $p = \sqcup p_i$, and suppose $\tau_a(p, U) > r$. Now $p(a)(U) = \tau_a(p, U) > r$. The fact that p is the sup of the p_i implies that $\exists i. p_i(a)(U) > r$. ∎

The domain *Proc* is a ω-continuous domain with a basis given by the following notion of "finite process".

Definition 5.16 *A **finite process** is generated by the following grammar:*

$$\begin{array}{rcl} q & ::= & 0 \\ & & \mid \{\{a_i \longrightarrow \{(q_{i,1}, r_{i,1}), \ldots, (q_{i,n_i}, r_{i,n_i})\}\} \mid i = 1, \ldots, k, \sum_{j=1,\ldots,n_i} r_{i,j} \leq 1\} \end{array}$$

*where a_i are labels and $r_{i,j}$ are real numbers in $[0,1]$. A finite process is **rational** if all probabilities are rational.*

A finite process is interpreted as an element of *Proc* via the following inductive definition.

Definition 5.17

$$\begin{array}{rcl} [\![0]\!] & = & \bot, \\ [\![\{\{a_i \longrightarrow \{(q_{i1}, r_{i1}), \ldots, (q_{in_i}, r_{in_i})\}\} \mid i = 1, \ldots, k\}]\!](a) & = & \sum_{k=1\ldots n_i} r_{i,n_k} \eta([\![q_{i,n_k}]\!]),\ a = a_i \\ & = & \bot,\ \textit{otherwise} \end{array}$$

Lemma 5.18 *The set of denotations of finite rational processes is a countable basis for Proc.*

Proof . *Proc* is the limit of the inverse limit system $\{(D_i, f_i, g_i)\}$ where $D_{i+1} = \mathsf{L} \longrightarrow \mathcal{P}_{\mathsf{Pr}}(D_i)$. *Proc* has a countable basis induced by the basis of the D_i's. The result now follows from Chapter 5 of [Jon90], which characterizes basis elements of the probabilistic powerdomain. ∎

Definition 5.19 *An equivalence relation R on Proc, is an **internal bisimulation** if: sRt implies that for all labels a, and all Scott-open R-closed C, $s(a)(C) = t(a)(C)$. We say that s is **internally bisimilar** to t if there exists a internal bisimulation R such that sRt.*

Internal bisimulation is an equivalence relation and is the maximum fixed point of the following monotone function F on the lattice of equivalence relations on *Proc* $\times$ *Proc*, where the ordering is inclusion :

$$s\ F(R)\ t \text{ if for all labels } a, \text{ and all Scott-open } R\text{-closed } C, s(a)(C) = t(a)(C)$$

Definition 5.20 *A preorder R on Proc is an **internal simulation** if sRt implies that for all labels a, and all Scott-open R-closed sets C, $s(a)(C) \leq t(a)(C)$.*

Internal simulation is a preorder and is the maximum fixed point of the following monotone function G on the lattice of preorders on *Proc* $\times$ *Proc*, where the ordering is inclusion :

$$s\ G(R)\ t \text{ if for all labels } a, \text{ and all Scott-open } R\text{-closed } C, s(a)(C) \leq t(a)(C)$$

The following proposition is immediate from the above definition.

Proposition 5.21 $\sqsubseteq$ *of Proc is an internal simulation.*

5.3 $\mathcal{L}_\vee$ as the logic of *Proc*

In this subsection, we show that $\mathcal{L}_\vee$ is complete for internal simulation. In the context of this subsection, the logic will be interpreted over the domain *Proc*.

Definition 5.22 (Logical satisfaction)

$$\begin{array}{ll}
p \models \top & \\
p \models \phi_1 \wedge \phi_2 & \textit{if } p \models \phi_1 \textit{ and } p \models \phi_2 \\
p \models \phi_1 \vee \phi_2 & \textit{if } p \models \phi_1 \textit{ or } p \models \phi_2 \\
p \models \langle a \rangle_q \phi & \textit{if } \exists U :\ \textit{Scott open with } U \subseteq [\![\phi]\!] \textit{ such that } p(a)(U) > r
\end{array}$$

where $[\![\phi]\!] = \{p \mid p \models \phi\}$.

An important property of the logically definable sets is that they are all Scott open.

Lemma 5.23 $[\![\phi]\!]$ *is a Scott-open subset of Proc.*

Proof . Recall that a set A is Scott open if A is up-closed and if whenever a directed sup, say $\sqcup X$, is in A then some member of X is in A. We proceed by induction on the structure of ϕ. Since we have, $[\![\top]\!] = \mathit{Proc}$, $[\![\phi_1 \wedge \phi_2]\!] = [\![\phi_1]\!] \cap [\![\phi_2]\!]$ and $[\![\phi_1 \vee \phi_2]\!] = [\![\phi_1]\!] \cup [\![\phi_2]\!]$; it follows immediately that the boolean connectives preserve Scott openness.

For the case of the modal operator we proceed as follows. We assume that $[\![\phi]\!]$ is Scott open and show that $[\![\langle a \rangle_q \phi]\!]$ is Scott open. Let $p \sqsubseteq p'$. Then: $p(a) \leq p'(a)$, and hence $p(a)([\![\phi]\!]) > q$ implies $p'(a)([\![\phi]\!]) > q$. So, $[\![\langle a \rangle_q \phi]\!]$ is up-closed. Let $p = \sqcup p_i$, and $p \models \langle a \rangle_q \phi$. Then, $\exists p_i$ such that $p_i(a)[\![\phi]\!] > q$. ∎

Lemma 5.24 *Let p and q be elements of Proc. Then, the following are equivalent:*

(1) $p \sqsubseteq q$,

(2) p *is simulated by* q,

(3) $p \models \phi$ *implies* $q \models \phi$.

Proof . (1) $\Rightarrow$ (2): From Proposition 5.21 $\sqsubseteq$ is an internal simulation.
(2) $\Rightarrow$ (3): By structural induction on formulas.
(3) $\Rightarrow$ (1): First assume that p is a finite process. We will proceed by induction on the height of p. Let $p(a) = (p_1, r_1), \ldots, (p_n, r_n)$ We use lemma 4.8 of [Jon90]. We need to show that for all upclosed (under $\sqsubseteq$) subsets K of $\{p_1, \ldots, p_n\}$:

$$\sum_{p_i \in K} r_i \leq q(a)(\cup_{p_i \in K}\{x \mid p_i \sqsubseteq x\})$$

First we show that: $\{x \mid p_i \sqsubseteq x\} = \cap\{[\![\psi]\!] \mid p_i \models \psi\}$.

- LHS $\subseteq$ RHS: $p_i \sqsubseteq x \Rightarrow (p_i \models \psi \Rightarrow x \models \psi)$.
- RHS $\subseteq$ LHS: $x \in \cap\{[\![\psi]\!] \mid p_i \models \psi\}$ implies $(p_i \models \psi \Rightarrow x \models \psi)$ implies $p_i \sqsubseteq x$ (Induction hypothesis on p_i)

Next, by distributivity, we see that:

$$\bigcup_{p_i \in K} \bigcap \{[\![\psi]\!] | p_i \models \psi\} = \bigcap_{\langle \psi_1, ..\psi_n \rangle} \{[\![\psi_1 \vee \psi_2 \vee ..\psi_n]\!] \mid p_i \in K, p_i \models \psi_i\}$$

Thus, it suffices to show that for all upclosed (under $\sqsubseteq$) subsets K of $\{p_1, \ldots, p_n\}$:

$$\sum_{p_i \in K} r_i \leq q(a)(\{[\![\psi_1 \vee \psi_2 \vee ..\psi_n]\!] \mid p_i \in K, p_i \models \psi_i\})$$

But, for any formula ϕ, by considering the formulas of the form: $\langle a \rangle_r \phi$, we get $p(a)([\![\phi]\!]) \leq q(a)([\![\phi]\!])$.

Now for general processes we proceed as follows. Let p' be a finite process way-below p, i.e. $p' \ll p$. We will show that $p' \sqsubseteq q$ and hence it will follow (since the finite processes form a basis) that $p \sqsubseteq q$. Every formula satisfied by p' is satisfied by p (since (1) implies (3)) and hence by q (by assumption) and hence by the proof in the preceding paragraph it follows that $p' \sqsubseteq q$. This concludes the proof that (3) implies (1) in the general case. ∎

Corollary 5.25 *p is internally bisimilar to q if and only if $p = q$ if and only if* $(\forall \phi \in \mathcal{L}_\vee)\ p \models \phi \Leftrightarrow q \models \phi$.

6 Relating *Proc* and LMP

In the preceding sections we have developed the theory of labelled Markov processes from two points of view. First we have the probability-theory view of labelled Markov processes and have developed the notion of finite approximation; second we have the domain theory view based on the Jones-Plotkin powerdomain. Here we show how one can go back and forth between these views: every element of *Proc* is a labelled Makov process and conversely, there is an embedding from labelled Markov processes to *Proc*. This correspondence yields the desired Polish space structure on labelled Markov processes and shows that the simple modal logic we have considered characterizes simulation for all labelled Markov processes.

6.1 From *Proc* to Labelled Markov Processes

The main result of this section is that the dcpo *Proc* can be made into a "universal" labelled Markov process, in an appropriate sense.

Let p be an element in the probabilistic powerdomain *Proc*. Consider the putative labelled Markov process $\mathcal{U}_0 = (|Proc|, p, \tau_a)$ where

- we are considering the elements of *Proc* under the Lawson topology (yielding a Polish space) [Law97],
- τ_a is given by the unique extension of the valuation $p(a)(\cdot)$ to measures on the Borel sets associated with the Lawson topology.

With the above notation we are able to state the main theorem of this section; it says that *Proc* defines a labelled Markov process.

Theorem 6.1 *The structure* $\mathcal{U}_0 = (|Proc|, p, \tau_a)$ *is a labelled Markov process.*

Proof . We only have to prove that $\tau_a(., E)$ is a measurable function for measurable $E \subseteq |Proc|$. We already know this for Scott open E, by Lemma 5.15. We now show that this is true for the σ-algebra generated by the Scott topology — since this is the same as the σ-algebra generated by the Lawson topology, we are done.

$\tau_a(., E^c) = \tau_a(., D) - \tau_a(., E)$, hence if $\tau_a(., E)$ is measurable, so is $\tau_a(., E^c)$.

Let E_i be a countable pairwise disjoint collection of sets. $\tau_a(., \cup_i E_i) = \sum_i \tau_a(., E_i)$, hence if each $\tau_a(., E_i)$, so is $\tau_a(., \cup_i E_i)$. This shows the result for all measurable sets E. ∎

Now we show that internal simulation - which we know coincides with the domain order from lemma 5.21 - coincides with simulation on labelled Markov processes. Since we already know that $\mathcal{L}_\vee$ characterizes internal simulation we get that $\mathcal{L}_\vee$ gives a logical characterization of simulation.

Theorem 6.2 *The* $\sqsubseteq$ *order on Proc is a simulation between the corresponding labelled Markov processes.*

Proof . Now suppose $s \sqsubseteq t$ in *Proc*. Thus for every label a and every Scott-open set U, $s(a)(U) \leq t(a)(U)$. Let R be the relation on $\mathcal{U}_0$ defined by sRt if and only if $s \sqsubseteq t$ in *Proc*. We want to show that R is a simulation. Let X be an R-closed measurable set of $\mathcal{U}_0$. We need to show $\tau_a(s, X) \leq \tau_a(t, X)$. Since X is R-closed in $\mathcal{U}_0$, it is up-closed in *Proc*. The result now follows from lemma 5.11. ∎

Note how Lawson compactness - through the use of lemma 5.11 - played a crucial role here.

6.2 Embedding labelled Markov processes into *Proc*

In this subsection we show how one can embed the poset of labelled Markov processes ordered by simulation (henceforth LMP) into the domain *Proc*. This completes the passage between the two views. In order to do this we will embed the finite acyclic labelled Markov processes (FAMPs for short) into *Proc*. Since FAMPs are acyclic they have a well-defined height, we define the embedding function $\psi(.) : FAMP \longrightarrow Proc$ by induction on the height of the DAG:

–$\psi(NIL) = \perp$, for a FAMP of height 0

–Let $(\mathcal{P}, p_0, \rho)$ be a FAMP. Then $\psi(\mathcal{P}) = \psi(p_0)$ is defined by $\psi(p_0)(a_i) = \sum_{p \in P} \rho_{a_i}(p_0, p) * \eta_{\psi(p)}$ where η_p is the point valuation at p.

The next lemma relates satisfaction of logical formulas by FAMPs and by their corresponding elements in the domain. In order to distinguish the two notions of satisfaction we will write $\models_D$ for the domain-theoretic notion and $\models_M$ for the Markov process concept.

Lemma 6.3 *Let $\mathcal{P}$ be a FAMP and ϕ a formula then*

$$\mathcal{P} \models_M \phi \iff \psi(\mathcal{P}) \models_D \phi.$$

Proof . Let $\mathcal{P} = (P, p_0, \rho)$ be a FAMP. We prove by induction on the structure of formulas that for every formula ϕ we have $[\![\phi]\!]_{\mathcal{P}} = P \cap \psi^{-1}([\![\phi]\!]_D)$. The base case and the induction step for conjunction and disjunction are obvious. Assume the claim is true for ϕ, we want to prove it for $\langle a \rangle_q \phi$. By definition of ψ, we have that for every $p \in P$ $\psi(p)(a) = \sum_{p' \in P} \rho_a(p, p')\mu_{\psi(p')}$ (even if p is NIL). Hence by induction hypothesis we have

$$\begin{aligned}
\rho_a(p, [\![\phi]\!]_{\mathcal{P}}) &= \rho_a(p, P \cap \psi^{-1}([\![\phi]\!]_D)) \\
&= \sum_{p' \in P} \rho_a(p, p')\mu_{\psi(p')}([\![\phi]\!]_D) \\
&= \psi(p)(a)([\![\phi]\!]_D).
\end{aligned}$$

Then we get $[\![\langle a \rangle_q \phi]\!]_{\mathcal{P}} = P \cap \psi^{-1}([\![\langle a \rangle_q \phi]\!]_D)$ which completes the proof that $[\![\phi]\!]_{\mathcal{P}} = P \cap \psi^{-1}([\![\phi]\!]_D)$. Hence we proved that $\mathcal{P} \models \phi$ if and only if $\psi(\mathcal{P}) \models \phi$ for every formula ϕ. ∎

It follows from the above lemma that the embedding function is monotone.

Lemma 6.4 *If a FAMP $\mathcal{P}_2$ simulates another FAMP, say $\mathcal{P}_1$, then $\psi(\mathcal{P}_1) \sqsubseteq \psi(\mathcal{P}_2)$.*

Proof . Suppose that $\mathcal{P}_2$ simulates $\mathcal{P}_1$ in the notation of the lemma, and that $\psi(\mathcal{P}_1) \models_D \phi$. Then by lemma 6.3 $\mathcal{P}_1 \models_M \phi$. Since $\mathcal{P}_2$ simulates $\mathcal{P}_1$ we have that $\mathcal{P}_2 \models_M \phi$. Then we have $\psi(\mathcal{P}_2) \models_M \phi$. Thus $\psi(\mathcal{P}_1) \sqsubseteq \psi(\mathcal{P}_2)$. ∎

Now we turn to the task of embedding all labelled Markov processes into *Proc*. Let $\mathcal{S}$ be a labelled Markov process and suppose that $\{\mathcal{P}_i \mid i \in I\}$ is a sequence of FAMPs such that:

1. $i \leq j \Rightarrow \mathcal{P}_i$ is simulated by $\mathcal{P}_j$,
2. every formula satisfied by $\mathcal{P}$ is satisfied by some $\mathcal{P}_i$.

In section 4 we have shown how to construct such a family of approximations. Now by lemma 6.4, condition 1 means that the family $\{\psi(\mathcal{P}_i) \mid i \in I\}$ is a chain in the domain *Proc*. We define

$$\psi(\mathcal{S}) = \sqcup_{i \in I} \psi(\mathcal{P}_i).$$

From the fact that the sets $[\![\phi]\!]_D$ are open in the Scott topology of *Proc* we know that every formula satisfied by $\psi(\mathcal{S})$ is satisfied by some $\psi(\mathcal{P}_i)$. Together with condition 2 we immediately get that

Proposition 6.5

$$\psi(\mathcal{S}) \models_D \phi \iff \mathcal{S} \models_M \phi.$$

It follows immediately - from this and from the logical characterization of bisimulation - that

Corollary 6.6 *If $\psi(\mathcal{S}_1) = \psi(\mathcal{S}_2)$ then $\mathcal{S}_1$ is bisimilar to $\mathcal{S}_2$.*

7 Example Applications

We recall our basic philosophy. Logical formulas are too "sharp" to be used in conjunction with quantitative approximations, as we have shown via the example of section 3. Instead, we advocate the idea that one should try to express the properties of interest as real-valued measurable functions or their integrals. Then the approximation techniques can be used to compute approximations to these integrals.

The primary way in which we use approximations is to compute values of continuous functions from $Proc$ to the real line. Recall that $Proc$ is the collection of all labelled Markov processes. A continuous function from $Proc$ might well be defined as the integral of some quantity over the state space of each process. Given $g \in \mathcal{C}(Proc, \mathbf{R})$, our theory tells us that we can approximate $g(\mathcal{S})$ for any process $\mathcal{S}$ by computing $g(\mathcal{P}_n)$, where $\mathcal{P}_n$ is a finite approximation to $\mathcal{S}$. Since $\mathcal{P}_n$ is finite, computing $g(\mathcal{P}_n)$ is straightforward.

Labelling states with values. In our model the states do not carry any labels that could be used for interesting calculations. For example, we might want to put into the state description some quantitative information that allows one to compute averages of interesting quantities. What we can do is encode this into the transitions and then use the formalism of labelled Markov processes to calculate quantities of interest.

- If we want to label some state s with an integer n from some bounded range, add a self loop labelled $\mathbf{n}$ to s by defining a transition $\tau_{\mathbf{n}}(s, A) = 1$ if $s \in A$, 0 otherwise.

- If we want to label s with a real number r, we use the probabilities to encode this number. Of course we can only encode numbers in the range $[0, 1]$ but a suitable scaling function can be used to encode anything else. We introduce a new dead state, call it Δ. We introduce a new label Z. We now introduce a transition $\tau_Z(s, \{\Delta\}) = r$. Now the functional expression $decode = \langle Z \rangle \cdot 1$, where 1 is the constant function, has the property $decode(s) = r$ if and only if s is labelled by r. Thus if we want to pick out the states labelled by r we can do so with our functional expressions. This way of encoding allows us to encode real-valued labels and still have only finitely many labels, as needed for the approximation theory.

The aim of the first example is to illustrate the nature of the approximations in a more familiar context.

Example 7.1 We consider $\int f d\mu$, the integration of continuous real functions $f : [0, 1] \to [0, 1]$ with respect to a measure μ, and sketch the approximation schemes yielded by our results. We first construct a **LMP** $\mathcal{S}$ and $g^f : \mathcal{S} \to [0, 1]$, such that $g^f(\mathcal{S}) = \int f d\mu$. $\mathcal{S}$ has a single start state s_0, along with states corresponding to the reals. Define a transition labelled a via $\tau_a(s_0, A) = \mu(A)$, for all measurable A. We label each state with its real number, as described above.

Now we consider the function $g^f = \langle a \rangle.(f \circ decode)$. Thus $g^f(s) = \int f \circ decode(t) \tau_a(s, dt) = \int f(t)\mu(dt)$. The continuity of g^f is immediate. So, our results yield:

$$g^f(\mathcal{S}) = \lim_{\mathcal{P}_i \longrightarrow \mathcal{S}} g^f(\mathcal{P}_i)$$

Let $\mathcal{P}$ be a finite state approximation to $\mathcal{S}$. Such a $\mathcal{P}$ has the following structure. $\mathcal{P}$ has a start state p_0 with an a transition. Let the set of states which form the target of the a-transition from p_0 be $s_1, \ldots, s_n$ with respective probabilities $p_1, \ldots, p_n$. Q satisfies

$$(\forall 1 \leq i \leq n) \ [(\sum_{r_i \leq r_j} p_j) \leq \mu([r_i, 1])]$$

We can explicitly calculate:

$$g^f(Q) = \sum_i f(r_i) \times p_i$$

where r_i is the maximum q such that s_i has a transition labelled d_{r_i}. Note however that the net of approximate answers $\{g^f(Q)\}$ is not monotone. In particular, we can think of Q implicitly inducing a partition, say $A_1, \ldots, A_m$ of $[0,1]$, with endpoints in $r_1, \ldots r_n$. Then, it is not necessarily the case that: $g_f(Q) \leq \sum_i f(\inf(A_i))\mu(A_i)$.

In fact we can consider various variants, say g^v, of g^f which replace *decode* with any continuous function $v : Proc \longrightarrow \mathbf{R}$ such that $decode(s) \leq v(s)$. In each case:

$$\int f d\mu = g^v(P) = \lim_{Q \longrightarrow P} g^v(Q)$$

The next example is a simple illustration of the estimation of drift.

Example 7.2 Consider a system - perhaps a cruise missile - being directed towards its target. It has a controller that checks its position, say 20 times every second. We idealize the system to a point in 3-dimensional space. This is a typical discrete-time continuous-space system. Every time step the x-coordinate advances by say 30 meters while the y and z coordinates are supposed to stay fixed. However, these coordinates "drift" - perhaps according to a gaussian, perhaps some other distribution is appropriate for windy conditions. The transition probabilities are given explicitly by $\tau((x,y,z),[a,a'] \times [b,b'] \times [c,c'])$ in terms of some distribution function $f(x,y,z,a,a',\ldots)$. From these τ can be calculated for all Borel sets. Thus we have defined a **LMP**.

Now we wish to know the *root-mean-square* (rms) drift in the y direction; this is the square root of the average value[13] of $y' - y$. Exactly similar remarks would apply for the z direction. Now after one step this is given by

$$\int\int\int (y - y_0)^2 f(x_0, y_0, z_0, \ldots) dx dy dz.$$

We might be interested in the rms drift after k steps. This would be obtained by integrating the function $(y-y_0)^2$, which is clearly continuous, against the measure obtained by the k-fold convolution of τ. If the fs are "smooth enough" these will be continuous and the resulting integral can be regarded as the evaluation of the integral of a continuous function using τ as the measure. The approximations can now be used to reduce these integrals to finite sums.

8 Related Work

The present paper exhibits a remarkable parallel between measure theory and domain theory to yield a theory of approximation for probabilistic processes. In our presentation we used our earlier results on the logical characterization of bisimulation to set up the correspondence between labelled Markov processes and the domain *Proc* but it could have been done - as we saw in hindsight purely domain theoretically. The approximation results were first discovered measure theoretically as was the logical characterization. On the other hand, the domain theory is crucial to get the completeness of the logic for simulation and establish the required Polish space structure on labelled Markov processes. It is possible that there is a purely measure theoretic path to this - perhaps using the notion of tightness of measures - but we have not discovered it yet.

[13] Calculating the mean value of the drift is not of much use in practice. For example for gaussian drifts the mean value of the drift is zero, giving no indication of how variable the system is.

We deal with probabilistic processes; all the internal indeterminacy is quantified via probability distributions, all the indeterminacy on the part of the environment is treated as external and modelled as nondeterminism. In a probabilistic analysis, quantitative information is recorded and used in the reasoning. In contrast, a purely qualitative nondeterministic analysis does not require and does not yield quantitative information. In particular when one has no quantitative information at all, one has to work with indeterminacy — using a uniform probability distribution is not the same as expressing complete ignorance about the possible outcomes. It is this which guides our choice of model. The environment is not being modelled so we do not attempt to ascribe probabilities to its transitions, but the system is being modelled so we have complete probabilistic data for internal transitions. Of course we are not suggesting that other models are not reasonable, but the model that we have chosen is in fact closer to what one sees in the probability theory literature than models that mix probability and nondeterminism. Furthermore in situations like performance analysis the purely probabilistic model makes the most sense. The mixture of nondeterminism and probability is interesting when one is dealing with specifications or in general in situations where one does not know the probability distributions, for example in economics.

The foundational work on the use of probability in semantics is due to Kozen [Koz81, Koz85] and Saheb-Djahromi [SD78, SD80]. These are concerned with domain theory and programming languages rather than with process equivalences, but they both introduced nontrivial measure-theoretic ideas. Kozen's paper [Koz85] introduces a probabilistic dynamic logic and observes a very interesting Stone-type duality in this context. However, our logic derives from the modal logic of Larsen and Skou [LS91].

The first paper with an abstract categorical approach to stochastic processes is by Giry [Gir81]. She studies categorical constructions rather than process equivalences. Her work - inspired originally by Lawvere - does not directly influence the present paper but it had a substantial impact on precursors of this paper. In particular she shows that the stochastic kernels (conditional probability distributions) that we use to define transition probabilities arise as the Kleisli category of a monad, which is a natural generalization of the powerset monad to the probabilistic case. If we recall that the category **Rel** of sets and relations is the Kleisli category of the powerset monad, we see that the stochastic kernels can reasonably be viewed as the probabilistic analogues of relations [Pan98]. This makes the analogy between labelled Markov processes and ordinary labelled transition systems quite striking.

The study of the interaction of probability and nondeterminism, largely in the context of exact equivalence of probabilistic processes, has been explored extensively using different models of concurrency. Probabilistic process algebras add randomness to the process algebra models—see for example [Han94, HJ94, JL91, JY95, LS91, HS86, BBS95, vGSS95, CSZ92, Cle94]. Probabilistic Petri nets [Mar89, VN92] add Markov chains to the underlying Petri net model. Probabilistic extensions of IO Automata [Seg95, WSS97] have also been developed. The verification community has been active in developing model checking tools for probabilistic systems, for example [BLL$^+$96, BdA95, BCHG$^+$97, CY95, HK97].

By and large, the above work focuses on discrete state systems. The other investigation that we are aware of – apart from our earlier papers [BDEP97, DEP98] – into continuous state spaces was by deVink and Rutten [dVR97]. They define a probabilistic transition system as a coalgebra of a suitable kind using the Giry monad mentioned above. Unfortunately their work applies to *ultrametric* spaces and not to metric spaces like the reals. In fact, their published proof [dVR97] that bisimulation is an equivalence relation only works in the discrete case. In some sense, ultrametric spaces are more like discrete spaces than like continuous spaces. Their coalgebraic approach is definitely attractive and it should be interesting to explore whether there is any way of extending their results to ordinary metric spaces such as the reals. More recently - since this paper was first announced - van Breugel and Worrell [vBW01] have studied metrics on labelled Markov processes from the viewpoint of metric space theory; i.e. they take their state spaces to be complete separable metric spaces not just Polish spaces.

Work has appeared on continuous-time systems. Typically these have been discrete-space continuous time systems of the sort one sees in queueing theory. The pioneering work is contained in the thesis of Jane Hillston [Hil94] on developing a process algebra for performance evaluation. She works with temporal delay in discrete-space Markov chains. The main point of her work is a compositional approach to performance evaluation. In her framework, she address continuous time in the following way. The systems being modelled are described by a probabilistic process algebra called PEPA. The semantics of PEPA are given in terms of labelled transition systems. Associated with the transitions is a continuous-time random process with an exponentially distributed delay. Associated with each type of action is a different rate. Indeterminacy is resolved by races between events executing at different rates. In her work a crucial role is played by a congruence called strong equivalence. Strong equivalence is in fact closely analogous to Larsen-Skou bisimulation. It is defined in a way very similar to the way that Larsen and Skou proceed, with the difference being that instead of using the probabilities associated with the actions she uses the *rates* associated with the actions.

The other area where bisimulation has appeared in a continuous context is the theory of timed automata [AD94] and hybrid automata [HHWT97]; of course these are not probabilistic systems. Here the basic framework is ordinary automata theory augmented with clocks. The main technical result is the region construction. This is a quotienting of the state space of the timed automaton - which is a continuum because of the clocks - by an equivalence relation, which, like bisimulation, has a coinductive definition. By imposing certain conditions on the way clocks can be read they guarantee that the region construction leads to a finite-state system. We work with a different philosophy. Our goal is to approximate very general systems by finite-state systems rather than to collapse more specialized systems down to finite-state systems.

A very interesting recent paper by Baier, Katoen and Hermans [BKH99] has appeared. Like us they take seriously the idea of approximate reasoning for continuous systems – they consider continuous-time Markov chains – but they work with the logic rather than with measurable functions. They look at the question of verifying PCTL* formulas. The work makes very clever use of ideas from symbolic model checking and from numerical analysis. However the reasoning suffers from the usual defect that we have already pointed out with the use of traditional logic in an approximate setting. While they point out the problems in using approximation for logical reasoning, they do not provide an explicit example. This criticism notwithstanding, this is certainly a very interesting development. The work is not really comparable to ours since we are not attempting to use our formalism for verification in the traditional sense but it might well influence us in the future.

In terms of the domain theory, the work of Baier and Kwiatkowska [BK96] is closest to this paper. They study the solution of a similar recursive domain equation in a variety of categories. The focus of their paper is the interaction of nondeterminism and probability in the discrete state-space setting — they do not explicitly consider the issues of continuous state spaces and approximations. Thus, even though some of the categories that they consider permit continuous state spaces, they do not study issues of continuous state spaces and discrete approximations to continuous systems. The use of domain logic, as in our paper, is not present in their paper. Thus they do not attempt to relate their domain to labelled Markov processes.

9 Conclusions

Our work equips Markov processes with the structure required to apply extant theories of approximations of integrals.

- In a series of influential papers [Eda94, Eda95a, Eda95c, Eda95b] Edalat exploits domain theoretic methods to approximate the integrals of a large class of functions. The compact Polish space structure of **LMP** enables these theories to be applied to computing over **LMP**'s.

- For probability measures over metric spaces, there is a large well developed suite of techniques that permit the approximation of integrals [Par67]. Intuitively, in our context, given that **LMP**'s form a Polish space, these theories permit the approximation of integrals by finite sums evaluated at finite trees.

In previous work we had developed a theory of metrics between **LMP**s [DGJP99] for robust reasoning on probabilistic processes. Intuitively, the metrics of [DGJP99] measure the closeness of the transition probabilities of processes. The metric distance in this paper is unrelated to the closeness of probability numbers and have very different convergence properties, e.g. some of the metrics of [DGJP99] do not yield Polish spaces.

One of the principal activities that we are engaged in is showing how the present theory can be applied to examples. The basic strategy is to show that *quantitative* properties of interest can be expressed as integrals of continuous functions on labelled Markov processes. Then one can approximate these integrals by using the approximation techniques to evaluate the integrals as finite sums. In order to proceed one needs to show that there is a rich enough collection of such continuous functions. We have several preliminary results along these lines. A more ambitious plan is to apply the notion of approximation to Markov decision processes with rewards. We are actively investigating this possibility.

Acknowledgements

We have benefitted from discussions with Samson Abramsky, Franck van Breugel, Martin Escardo, Reinhold Heckmann, Achim Jung, Gordon Plotkin and Erik de Vink. We thank Falk Bartels for pointing out an error in an earlier draft. Josée Desharnais has been supported by a grant from NSERC and by funding from MITACS during the course of this work. Radha Jagadeesan's research was supported by NSF grants CCR 99010171 and CCR 02030716. Prakash Panangaden's research was supported by funding from NSERC and MITACS.

References

[Abr91] S. Abramsky. A domain equation for bisimulation. *Information and Computation*, 92(2):161–218, 1991.

[AD94] R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183:235, 1994.

[AMESD98] M. Alvarez-Manilla, A. Edalat, and N. Saheb-Djahromi. An extension result for continuous valuations. *Electronic Notes in Theoretical Computer Science*, 13, 1998.

[ASB+95] A. Aziz, V. Singhal, F. Balarin, R. K. Brayton, and A. L. Sangiovanni-Vincentelli. It usually works:the temporal logic of stochastic systems. In *Proceedings of the Conference on Computer-Aided Verification*, 1995.

[BBS95] J.C.M. Baeten, J.A. Bergstra, and S.A. Smolka. Axiomatizing probabilistic processes: ACP with generative probabilities. *Information and Computation*, 121(2):234–255, 1995.

[BCHG+97] Christel Baier, Ed Clark, Vasiliki Hartonas-Garmhausen, Marta Kwiatkowska, and Mark Ryan. Symbolic model checking for probabilistic processes. In *Proceedings of the 24th International Colloquium On Automata Languages And Programming*, number 1256 in Lecture Notes In Computer Science, pages 430–440, 1997.

[BdA95] A. Bianco and L. de Alfaro. Model checking of probabilistic and nondeterministic systems. In P. S. Thiagarajan, editor, *Proceedings of the 15th Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, number 1026 in Lecture Notes In Computer Science, pages 499–513, 1995.

[BDEP97] R. Blute, J. Desharnais, A. Edalat, and P. Panangaden. Bisimulation for labelled Markov processes. In *Proceedings of the Twelfth IEEE Symposium On Logic In Computer Science, Warsaw, Poland.*, 1997.

[BK96] C. Baier and M. Kwiatkowska. Domain equations for probabilistic processes. Available from URL http://www.cs.bham.ac.uk/ mzk/, March 1996.

[BKH99] C. Baier, J-P. Katonen, and H. Hermans. Approximative symbolic model checking of continuous-time markov chains. In J. C. M. Baeten and S. Mauw, editors, *Proceedings of the Tenth International Conference on Concurrency Theory, CONCUR99*, number 1664 in Lecture Notes In Computer Science, pages 146–161. Springer-Verlag, 1999.

[BLL+96] J. Bengtsson, K. G. Larsen, F. Larsson, P. Pettersson, and W. Yi. UppAal: A tool suite for automatic verification of real-time systems. In R. Alur, T. Henzinger, and E. Sontag, editors, *Hybrid Systems III*, number 1066 in Lecture Notes In Computer Science, pages 232–243, 1996.

[Cle94] R. Cleaveland. Fully abstract characterizations of testing preorders for probabilistic processes. In *CONCUR 94*, number 836 in Lecture Notes in Computer Science, pages 497–512. Springer-Verlag, August 1994.

[CSZ92] R. Cleaveland, S. Smolka, and A. Zwarico. Testing preorders for probabilistic processes. In W. Kuich, editor, *Automata, Languages and Programming (ICALP 92)*, number 623 in Lecture Notes in Computer Science, pages 708–719. Springer-Verlag, 1992.

[CY95] C. Courcoubetis and M. Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.

[DEP98] J. Desharnais, A. Edalat, and P. Panangaden. A logical characterization of bisimulation for labeled Markov processes. In *proceedings of the 13th IEEE Symposium On Logic In Computer Science, Indianapolis*, pages 478–489. IEEE Press, June 1998.

[DEP02] J. Desharnais, A. Edalat, and P. Panangaden. Bisimulation for labeled Markov processes. *Information and Computation*, 2002.

[Des99] J. Desharnais. Logical characterization of simulation for markov chains. In M. Kwiatkowska, editor, *Probmiv99, Proceedings of the Second International Workshop on Probabilistic Methods in Verification*, pages 33–48. The University of Birmingham, 1999. Available as TR: CSR-99-8.

[DGJP99] J. Desharnais, V. Gupta, R. Jagadeesan, and P. Panangaden. Metrics for labeled Markov systems. In *Proceedings of CONCUR99*, Lecture Notes in Computer Science. Springer-Verlag, 1999.

[DGJP00] J. Desharnais, V. Gupta, R. Jagadeesan, and P. Panangaden. Approximation of labeled markov processes. In *Proceedings of the Fifteenth Annual IEEE Symposium On Logic In Computer Science*, pages 95–106. IEEE Computer Society Press, June 2000.

[dVR97] E. de Vink and J. J. M. M. Rutten. Bisimulation for probabilistic transition systems: A coalgebraic approach. In *Proceedings of the 24th International Colloquium On Automata Languages And Programming*, 1997.

[Eda94] Abbas Edalat. Domain of computation of a random field in statistical physics. In C. Hankin, I. mackie, and R. Nagarajan, editors, *Theory and Formal Methods 1994: Proceedings of the second Imperial College Department of Computing Workshop on Theory and Formal Methods*, pages 11–14. IC Press, 1994.

[Eda95a] Abbas Edalat. Domain theory and integration. *Theoretical Computer Science*, 151:163–193, 1995.

[Eda95b] Abbas Edalat. Domain theory in stochastic processes. In *Proceedings of the Tenth Annual IEEE Symposium On Logic In Computer Science*. IEEE Computer Society Press, 1995.

[Eda95c] Abbas Edalat. Dynamical systems, measures and fractals via domain theory. *Information and Computation*, 120(1):32–48, 1995.

[Ein05] A. Einstein. The theory of the brownian movement. *Ann. der Physik*, 17:549, 1905.

[Fel71] W. Feller. *An Introduction to Probability Theory and its Applications II.* John Wiley and Sons, 2nd edition, 1971.

[Gir81] M. Giry. A categorical approach to probability theory. In B. Banaschewski, editor, *Categorical Aspects of Topology and Analysis*, number 915 in Lecture Notes In Mathematics, pages 68–85. Springer-Verlag, 1981.

[GJP99] V. Gupta, R. Jagadeesan, and P. Panangaden. Stochastic processes as concurrent constraint programs. In *Proceedings of the 26th Proceedings Of The Annual ACM Symposium On Principles Of Programming Languages*, 1999.

[Han94] Hans A. Hansson. *Time and Probability in Formal Design of Distributed Systems*, volume 1 of *Real-time Safety-critical Systems*. Elseiver, 1994.

[HHWT97] T. Henzinger, P.-H. Ho, and H. Wong-Toi. Hytech: a model checker for hybrid systems. *Software Tools for Technology Transfer*, 1(1):110–122, 1997.

[Hil94] J. Hillston. *A Compositional Approach to Performance Modelling*. PhD thesis, University of Edinburgh, 1994. Published as a Distinguished Dissertation by Cambridge University Press in 1996.

[HJ94] H. Hansson and B. Jonsson. A logic for reasoning about time and reliability. *Formal Aspects of Computing*, 6(5):512–535, 1994.

[HK97] M. Huth and M. Kwiatkowska. Quantitative analysis and model checking. In *proceedings of the 12 IEEE Symposium On Logic In Computer Science*, pages 111–122. IEEE Press, 1997.

[HS86] S. Hart and M. Sharir. Probabilistic propositional temporal logics. *Information and Control*, 70:97–155, 1986.

[JL91] B. Jonsson and K. Larsen. Specification and refinement of probabilistic processes. In *Proceedings of the 6th Annual IEEE Symposium On Logic In Computer Science*, 1991.

[Jon90] C. Jones. *Probabilistic Non-determinism*. PhD thesis, University of Edinburgh, 1990. CST-63-90.

[JP89] C. Jones and G. D. Plotkin. A probabilistic powerdomain of evaluations. In *Proceedings of the Fourth Annual IEEE Symposium On Logic In Computer Science*, pages 186–195, 1989.

[JT98] A. Jung and R. Tix. The troublesome probabilistic powerdomain. *Electronic Notes in Theoretical Computer Science*, 13, 1998.

[Jun88] A. Jung. *Cartesian Closed Categories of Domains*. PhD thesis, Technischen Hogschule Darmstadt, 1988.

[JY95] B. Jonsson and W. Yi. Compositional testing preorders for probabilistic processes. In *Proceedings of the 10th Annual IEEE Symposium On Logic In Computer Science*, pages 431–441, 1995.

[Koz81] D. Kozen. Semantics of probabilistic programs. *Journal of Computer and Systems Sciences*, 22:328–350, 1981.

[Koz85] D. Kozen. A probabilistic PDL. *Journal of Computer and Systems Sciences*, 30(2):162–178, 1985.

[KS60] J. G. Kemeny and J. L. Snell. *Finite Markov Chains*. Van Nostrand, 1960.

[Law82] J. D. Lawson. Valuations on continuous lattices. In R.-E. Hoffman, editor, *Continuous lattices and related topics*, volume 27 of *Mathematik Arbeitspapiere*, pages 204–225. Universität Bremen, 1982.

[Law97] J. Lawson. Spaces of maximal points. *Mathematical Structures in Computer Science*, 7(5):543–555, October 1997.

[LS91] K. G. Larsen and A. Skou. Bisimulation through probablistic testing. *Information and Computation*, 94:1–28, 1991.

[Mar89] M. Ajmone Marsan. Stochastic petri nets: an elementary introduction. In *Advances in Petri Nets 1989*, pages 1–29. Springer-Verlag, 1989.

[Mil89] R. Milner. *Communication and Concurrency*. Prentice-Hall, 1989.

[Øks95] B. Øksendal. *Stochastic Differential Equations*. Springer-Verlag, 1995.

[Pan98] P. Panangaden. Probabilistic relations. In C. Baier, M. Huth, M. Kwiatkowska, and M. Ryan, editors, *PROBMIV98*, pages 59–74, 1998.

[Par67] K. R. Parthasarathy. *Probability Measures on Metric Spaces*. Academic Press, 1967.

[SD78] N. Saheb-Djahromi. Probabilistic LCF. In *Mathematical Foundations Of Computer Science*, number 64 in Lecture Notes In Computer Science. Springer-Verlag, 1978.

[SD80] N. Saheb-Djahromi. Cpos of measures for nondeterminism. *Theoretical Computer Science*, 12(1):19–37, 1980.

[Seg95] R. Segala. *Modeling and Verification of Randomized Distributed Real-Time Systems*. PhD thesis, MIT, Dept. of Electrical Engineering and Computer Science, 1995. Also appears as technical report MIT/LCS/TR-676.

[Smo06] M. Smoluchowski. Brownian motion. *Ann. der Physik*, 21:756, 1906.

[SP82] M. B. Smyth and G. D. Plotkin. The category theoretic solution of recursive domain equations. *Siam Journal of Computing*, 11(4), 1982.

[vBW01] Franck van Breugel and James Worrell. Towards quantitative verification of probabilistic systems. In *Proceedings of the Twenty-eighth International Colloquium on Automata, Languages and Programming*. Springer-Verlag, July 2001.

[vGSS95] R. van Glabbeek, S.A. Smolka, and B.U. Steffen. Reactive, generative, and stratified models of probabilistic processes. *Information and Computation*, 121(1):59–80, 1995.

[VN92] N. Viswanadham and Y. Narahari. *Performance Modeling of Automated Manufacturing Systems*. Prentice-Hall Inc, 1992.

[WSS97] S.-H. Wu, S.A. Smolka, and E. W. Stark. Composition and behaviors for probabilistic I/O automata. *Theoretical Computer Science*, 176(1–2):1–36, April 1997.