Hierarchical Reasoning in Probabilistic CSP

Karen Seidel and Carroll Morgan *

10 May 1996

Abstract

Probabilistic CSP extends the language of CSP with an operator for probabilistic choice. However reasoning about an intricate combination of nondeterminism, communication and probabilistic behaviour can be complicated. In standard CSP, and in formal methods generally, such complication is overcome (when possible) by use of hierarchical reasoning. In this paper we provide a foundation for lifting such reasoning to the probabilistic setting.

First we formalise the common observation that the standard models of CSP (traces, refusals and refusals/divergences) form a hierarchy, by showing that they are linked by embedding-projection pairs. Such structure underlies hierarchical reasoning in which complex process behaviour is reasoned about in terms of its simpler projections. Then we show how that hierarchy can be extended to a corresponding hierarchy between the probabilistic models, by using each of those three models of standard CSP as a basis for a probabilistic extension. Finally we show that there is a projection from the probabilistic models onto the standard models, which can be used to reason about non-probabilistic properties of probabilistic processes.

1 Introduction

An important criterion for the effective use of formal methods in practice is that the cost of formality should not outweigh the benefit gained from the

*Morgan is a member of the Programming Research Group at the University of Oxford: both authors may be contacted at {*karen,carroll*}*@comlab.ox.ac.uk*. Seidel was supported by the EPSRC.

extra rigour. That encourages in general a hierarchical structure of reasoning systems, from which for any specific practical purpose one chooses the 'cheapest' reasoning system — the least complex — that is sufficient for the problem at hand. In this article we study such hierarchies in the context of *probabilistic concurrency*, showing in particular how probabilistic hierarchies can be constructed from existing non-probabilistic ones in simple and uniform way — and in doing so we hope to take some steps towards the long-term goal of making it easier to reason about probabilistic concurrent systems in practice.

The denotational models of CSP [3] form a hierarchy ordered by the information content of their semantics. At the bottom lies the *traces model*, formed by observing the order in which events occur; in it a process is denoted by a set of finite sequences of events (the traces). Next lies the *failures model*, formed by observing the traces together with those sets of events which might not be performed (the failures); in it a process is denoted by a relation between traces and sets of events. And at the top (for present purposes) lies the *divergences* (or failures-divergences) *model*, formed by observing failures plus the traces after which divergence (or livelock) occurs. Each model forms an algebraic inductive partial order (or *ipo*) under its refinement ordering, a structure appropriate for a semantic model of a language which includes recursion.

Since each model contains fewer observables than the one directly above it, it can be embedded in the one above. An element of the traces model is usually embedded in the failures model by mapping it to the deterministic process having the given traces; an element of the failures model is usually embedded in the divergences model by mapping it to the divergences-free process having the given failures. Furthermore, mapping down the hierarchy are projection functions. The left-hand side of Figure 1 shows the hierarchy. Although we have no use for it in this paper, extension of the hierarchy is possible, for instance to include a timed model.

In such hierarchies of semantic models an embedding-projection pair (to be defined in the body of the paper) effectively identifies the simpler model with a subspace of the more complicated one, in which some of the properties of the simpler model are preserved. They are important for a number of reasons.

Firstly they enable reasoning about processes to be simplified: certain aspects of a system may be analysed in the simpler model, even though

the specification of the whole system requires the more complicated model. Safety properties described in the divergences model may be able to be analysed in the simpler traces model; deadlock-freedom may be able to be analysed in the simpler failures model; and analysis of divergences-freedom must take place in the divergences model itself. As another example, Davies and Schneider [1] have shown how the projection from a timed model onto an untimed one can be used to reason about untimed properties of a timed system.

Secondly it may be important, perhaps for the purposes of formal development, to know that a restricted model lies inside a more expressive one; typically the more expressive model is the scene for a development which must end involving just constructs from the restricted model (the 'code').

More generally the embedded subspace may exhibit some intrinsically interesting properties (for code, that is executability). For example the deterministic embedding from traces to failures preserves idempotence of parallel composition.

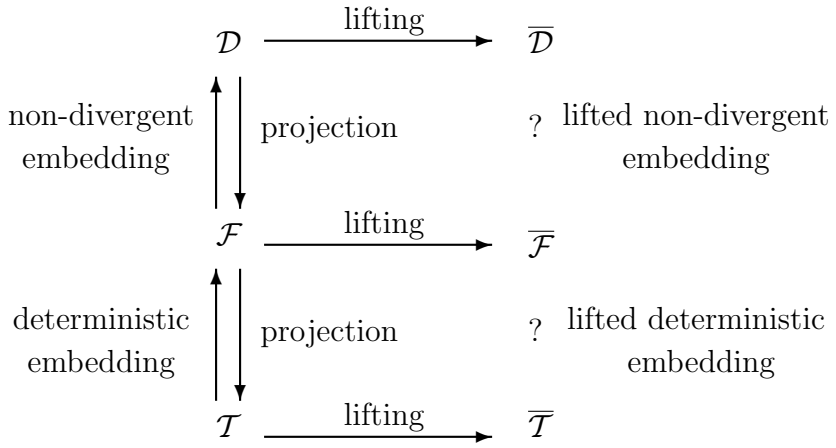

Figure 1: Required probabilistic lifting of the standard hierarchy

In [5] the authors have constructed a probabilistic process algebra based on the divergences model of CSP. They employ a very general method, due to Jones [4], for constructing a probabilistic extension of an *ipo*. The result in [5] is a model of CSP which contains also probabilistic choice: for any

processes P and Q, process $P \,{}_p\!\oplus\, Q$ behaves like P with probability p and like Q with probability $1-p$. Jones's method (which we call *lifting*) provides a functor on *ipo*'s and continuous functions between them. As a result the standard CSP operators, being continuous functions on the standard models, lift to operators on the lifted, probabilistic, models. It also means that any continuous function between standard models (for example, in a strong embedding-projection pair, as we shall define it) lifts to a continuous function between the corresponding lifted, probabilistic, models.

The resulting model of probabilistic CSP enables phenomena not previously formalised (involving probabilities) to be specified and analysed. However such specifications involve traces, failures, divergences *and* probabilistic behaviour and so are in general even more complicated than specifications in standard CSP. Some form of hierarchical reasoning is necessary to support reasoning about any but the simplest of examples.

In this paper we provide such support. We show that although the standard embeddings in Figure 1 are not continuous they can be replaced by embeddings which are, and which together with their projections form embedding-projection pairs in which the projections are actually continuous; we call such pairs *strong* embedding-projection pairs. We then use Jones's probabilistic functor to lift the resulting hierarchy to gain strong embedding-projection pairs between the lifted, probabilistic, models (see Figure 5).

The simplest, traces, model is used most frequently because of its ability to capture saftey properties. We thus concentrate on it, linking its embedding into the failures model to Hennessey's idea of *must*-tests [2]. That not only provides a relation between the probabilistic models but also transfers the idea of *must*-tests to the probabilistic domain. We also show that an earlier restricted probabilistic submodel of deterministic CSP due to Seidel ([8]) can be embedded, via an embedding-projection pair, into probabilistic deterministic CSP. We do so by showing that her model can be viewed as the lifting of a model which is even simpler than the traces model.

Finally it is occasionally useful to reason about a probabilistic process by ignoring probabilities entirely, replacing each probabilistic choice $P \,{}_p\!\oplus\, Q$ with the less informative nondeterministic choice $P \sqcap Q$. To support that method we show that such a transformation forms a weak inverse to the probabilistic lifting, so that together they form an embedding-projection pair; its projection is however not continuous.

The paper is organised as follows. In Section 2 we summarise the little

theory we require of *ipo*'s, the three standard models of CSP, and the simple properties we require of embedding-projection pairs. In Section 3 we construct strong embedding-projection pairs to form the hierarchy of standard models of CSP. In Section 4 we summarise the probabilistic lifting of an *ipo* and apply it to CSP. In Section 5 we produce the probabilistic hierarchy required on the right-hand side of Figure 1. In Section 6 we embed Seidel's simple model in probabilistic deterministic CSP; and in Section 7 we find a weak inverse for the probabilistic lifting.

2 Background

In this section we summarise the order structure underlying the results of this paper, and the models of CSP in which we are interested and which exhibit that structure.

2.1 Order

Definition 2.1 A *partially ordered space* $(X, \sqsubseteq)$ consists of a non-empty set X equipped with a reflexive, antisymmetric and transitive relation $\sqsubseteq$ on X. If Z is a subset of X, y is an *upper bound* of Z iff it dominates each element of Z: for each $z \in Z$, $z \sqsubseteq y$. A subset Z of X is *directed* iff each finite subset of Z has an upper bound in Z.

An *inductive partial order*, or *ipo*, is a partially ordered space $(X, \sqsubseteq)$ that contains a least upper bound $\sqcup Z$ for each directed subset Z of X:

1. $\sqcup Z$ is an upper bound
2. $\sqcup Z$ is a *least upper bound*: if y satisfies $z \sqsubseteq y$ for each $z \in Z$, then $\sqcup Z \sqsubseteq y$.

□

The morphisms between *ipo*'s in which we are interested are continuous functions.

Definition 2.2 If $(X, \sqsubseteq_X)$ and $(Y, \sqsubseteq_Y)$ are *ipo*'s, a function f from X to Y is *continuous* if it is monotonic and preserves least upper bounds:

1. f is *monotonic*: if $x \sqsubseteq_X y$ then $fx \sqsubseteq_Y fy$ (in which case it is routine to show that the image of a directed set is directed)

2. f is *continuous*: f is monotonic and for any directed set Z in X, $f(\sqcup Z) = \sqcup f(Z)$. Observe that for simplicity we shall employ the same symbol for the least upper bound in *ipo*'s having different orders.

□

2.2 Models of CSP

We wish to specialise to *ipo*'s which are models of CSP or their probabilistic liftings. Their morphisms will be either CSP operators (functions from a model to itself) or embeddings (functions from one model to another). Here is a summary of the models we use; details appear in [3]. Throughout we assume the universe of events to be finite and consider unalphabetised processes (so that parallel composition is parametrised).

Traces. The *traces model* of CSP is important firstly for its use in reasoning about safety properties of processes and secondly because it is fully abstract [10] for deterministic processes. In it, each process is denoted by a set of finite sequences of elements of A, called the set of traces of the process, consisting of the sequences of events in which the process is able to engage. In the traces model one process refines another if it can perform all the traces of that other. We let $A^\star$ denote the set of finite sequences from A; catenation of sequences is denoted by their juxtaposition. A subset of $A^\star$ is said to be *prefix closed* iff it contains each prefix (that is initial segment) of each of its elements. Thus for set A, the universe (or alphabet) of events, the *traces model* over A is defined to be the space $(\mathcal{T}, \sqsubseteq_\mathcal{T})$ whose set is defined

$$\mathcal{T} \quad := \quad \{T \mid T \text{ is a nonempty and prefix-closed subset of } A^\star\}.$$

(We write $:=$ for 'is defined to be'.) The order of $(\mathcal{T}, \sqsubseteq_\mathcal{T})$ is set inclusion, given by

$$T \sqsubseteq_\mathcal{T} U \quad := \quad T \subseteq U.$$

The least element of $(\mathcal{T}, \sqsubseteq_\mathcal{T})$ is the process $STOP$ which performs no events. The definition, recursively over the syntax, of the traces semantics of a general process can be found in [3]. For example the recursive process

$\mu X \cdot F(X)$ has traces equal to the least upper bound (that is, union) of the directed set (under inclusion), as $n \in \mathbf{N}$, of the traces of $F^n(STOP)$. $\Box$

Failures. The *failures model* of CSP is important for its use in reasoning about liveness properties of processes, and because it is fully abstract for nondivergent processes. In it, each process is denoted by a relation F between the set of traces of the process and the set of all subsets of A, that relates each trace to the sets of events which may be refused (that is, may lead to deadlock if its environment offers them) by the process immediately after it has engaged in that trace. Such failures information is sufficient to express nondeterminism (see [3]), with the result that a process which is less deterministic than another contains the failures of that other. In the failures model one process refines another if its behaviour is at least as deterministic.

Thus for universe A of events, the *failures model* over A is defined to be the space $(\mathcal{F}, \sqsubseteq_{\mathcal{F}})$ whose set is defined

$$\begin{array}{l} \mathcal{F} \quad := \\ \quad \{F \mid F \text{ is a relation from } A^\star \text{ to subsets of } A \text{ satisfying:} \\ \qquad dom\ F \in \mathcal{T}; \\ \qquad \text{if } (t, E) \in F \text{ and } D \text{ is a subset of } E \text{ then } (t, D) \in F; \\ \qquad \text{if } (t, E) \in F \text{ and } x \in A \text{ then either } (t, E \cup \{x\}) \in F \\ \qquad \text{or } (t\langle x \rangle, \{\}) \in F\}. \end{array}$$

The order of $(\mathcal{F}, \sqsubseteq_{\mathcal{F}})$ is set containment

$$F \sqsubseteq_{\mathcal{F}} G \quad := \quad F \supseteq G.$$

The least element of $(\mathcal{F}, \sqsubseteq_{\mathcal{F}})$ is the process *CHAOS* which may perform or refuse any element in A at any interaction. Observe that the traces of a process F are obtained in the failures model as $\{t \mid (t, \{\}) \in F\}$. The definition, again by recursion over the syntax, of the failures of a process can be found in [3]. For example the failures of the recursive process $\mu X \cdot F(X)$ equals the least upper bound (that is, intersection) of the directed set (under containment), as $n \in \mathbf{N}$, of the failures of $F^n(CHAOS)$.

A process is nondeterministic iff at some interaction it can both perform and refuse an event. Writing

$$det(P, t, a) \quad := \quad t\langle a \rangle \in dom\ P \Rightarrow (t, \{a\}) \notin P,$$

we define P to be *deterministic* iff the following holds:

$$det(P) \quad := \quad \forall t : dom\ P \cdot \forall a : A \cdot det(P, t, a).$$

Otherwise P is *nondeterministic*. For later use we also define

$$det(P,t) \quad := \quad t \in dom\ P \ \wedge\ \forall a : A \bullet det(P,t,a).$$

□

Divergences. The *divergences model* of CSP (often called the failures/divergences model) is important for its use in reasoning about livelock freedom of processes, and because it is fully abstract for processes in CSP. In it, each process is denoted by a pair (F, D) where F is the failures as above and D is the set of traces of the process after which divergence (that is, livelock) may occur. Divergence causes arbitrary behaviour: thus if P diverges at trace t then P can subsequently perform or refuse any event — the extra information in the divergences model is, however, precisely that divergence is explicitly indicated rather than being identified with such arbitrary behaviour.

In the divergences model one process refines another if it is at least as deterministic and 'at least as terminating'.

Thus for universe A, the *divergences model* over A is defined to be the space $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$ whose set is defined

$$\begin{array}{l} \mathcal{D} \quad := \\ \quad \{(F,D) \ \mid \ F \in \mathcal{F} \text{ and } D \text{ is a subset of the domain of } F \text{ satisfying:} \\ \qquad\qquad \text{if } t \in D \text{ and } u \in A^\star \text{ then } tu \in D; \\ \qquad\qquad \text{if } t \in D \text{ and } E \subseteq A \text{ then } (t,E) \in F\}. \end{array}$$

The order of $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$ is set containment of components

$$(F,D) \sqsubseteq_{\mathcal{D}} (G,E) \quad := \quad F \supseteq G \text{ and } D \supseteq E.$$

The least element of $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$ is the process $CHAOS'$ which may perform or refuse any element in A, or diverge, at any interaction. The definition of the divergences of a process expression can be found in [3]. □

The following standard result is our starting point.

Theorem 2.3 Each of the spaces $(\mathcal{T}, \sqsubseteq_{\mathcal{T}})$, $(\mathcal{F}, \sqsubseteq_{\mathcal{F}})$ and $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$ is an *ipo* and the operations of CSP provide continuous functions on each of them. □

In Section 3 we shall formalise the intuition that those models form a hierarchy.

2.3 Compactness

An *ipo* has a topology which is important because it captures the notion of convergence of semantic approximations. We use it now to define compactness and later to define probabilistic lifting.

Definition 2.4 Given an *ipo* $(X, \sqsubseteq)$, a subset S of X is *Scott-open* (that is, an element of the Scott topology on $(X, \sqsubseteq)$) if both:

1. S is *up-closed*: if $x \in S$ and $x \sqsubseteq y$ then $y \in S$
2. S is *inaccessible*: if a directed limit $\sqcup Z$ lies in S then one of its elements must lie in S; that is, for each directed subset Z of X, if $\sqcup Z \in S$ then $z \in S$ for some $z \in Z$.

□

The notion of compactness formalises the idea of 'finite element' in an *ipo*. We shall find it useful in simplifying the study of probabilistic liftings.

Definition 2.5 Let $(X, \sqsubseteq)$ be an *ipo*. For an element x of X, the *cone on* x, written $x{\uparrow}$, is defined to consist of all elements that dominate x

$$x{\uparrow} \quad := \quad \{y \in X \mid x \sqsubseteq y\}.$$

An element x of X is called *compact* iff the cone on x is Scott open; that is, for each directed subset Z of X with $x \sqsubseteq \bigsqcup Z$, there is some $z \in Z$ for which $x \sqsubseteq z$. The set of all compact elements of X is denoted $K(X)$. □

It is straightforward to show (see [5, Lemma 5.3]) that in the *ipo* $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$, a process is compact iff after some finite number of interactions it equals the divergent process. Such processes are often called *finite*. In fact the proof in [5, Lemma 5.3] establishes the same fact for the *ipo* $(\mathcal{F}, \sqsubseteq_{\mathcal{F}})$. For $(\mathcal{T}, \sqsubseteq_{\mathcal{T}})$, refinement is set inclusion rather than set containment; so with the least process being *STOP* rather than *CHAOS*, the proof shows that a process is compact in $(\mathcal{T}, \sqsubseteq_{\mathcal{T}})$ iff it equals *STOP* after some finite number of interactions.

The three models of CSP satisfy two further properties which are common in semantic models and which we require.

Definition 2.6 An *ipo* $(X, \sqsubseteq_X)$ is *algebraic* iff each element is approximated by compact elements: for each $x \in X$, $\{y : K(X) \mid y \sqsubseteq x\}$ is directed and has least upper bound x. (Then since the empty set is bounded above by any element, it must have a least upper bound which is therefore the least element of X.) It *has partial joins* iff each finite set with an upper bound has a least upper bound. □

Clearly the three models of CSP have partial joins by virtue of their actually containing a least upper bound for *any* finite subset. An *ipo* without that property but still having partial joins appears in Section 6. We shall exploit the following result in Section 4.2 to conclude that certain continuous functions are determined by their restrictions to compact elements; we use it also in Section 7.

Theorem 2.7 Each of the three *ipo*'s $(\mathcal{T}, \sqsubseteq_{\mathcal{T}})$, $(\mathcal{F}, \sqsubseteq_{\mathcal{F}})$ and $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$ is algebraic and has partial joins.
Proof: For the space $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$ the result is proved in [5, Lemma 5.4]. It remains to observe here that the same reasoning applies equally to both $(\mathcal{F}, \sqsubseteq_{\mathcal{F}})$ and $(\mathcal{T}, \sqsubseteq_{\mathcal{T}})$, with the identification of compact elements made above. □

2.4 Embedding-Projection Pairs

In this section we recall the basic properties of embedding-projection pairs between *ipo*'s. In Section 3 we shall show how such pairs can be used to relate the models of CSP and, in Section 5, how they can also be used to relate the probabilistic liftings of those models.

An embedding ϕ and a projection ψ can be regarded as weak inverses of each other if they satisfy the following definition.

Definition 2.8 Let $(X, \sqsubseteq_X)$ and $(Y, \sqsubseteq_Y)$ be *ipo*'s. A pair of functions $\phi : X \to Y$ and $\psi : Y \to X$ form a *Galois connection* if and only if they are monotonic and

$$\forall x : X \cdot x \sqsubseteq_X \psi(\phi(x)) \quad \text{and} \quad \forall y : Y \cdot \phi(\psi(y)) \sqsubseteq_Y y,$$

in which case ϕ is called the *embedding* and ψ the *projection* of the pair and we write $gc(\phi, \psi)$. Equivalently ϕ and ψ simply satisfy

$$\forall x : X,\ y : Y \cdot x \sqsubseteq_X \psi(y) \quad \text{iff} \quad \phi(x) \sqsubseteq_Y y.$$

An *embedding-projection pair* (or *ep*-pair for short) is a Galois connection satisfying the stronger condition

$$\forall x : X \cdot x = \psi(\phi(x)) \quad \text{and} \quad \forall y : Y \cdot \phi(\psi(y)) \sqsubseteq_Y y,$$

in which case we write $ep(\phi, \psi)$. □

Galois connections and *ep*-pairs have several properties which are well-known and easy to prove [10], among them: the equivalence of the two forms of definition given above; that each member of a Galois connection determines the other; that if $ep(\phi, \psi)$ then ϕ is injective; and that in any Galois connection ϕ distributes $\sqcup$ and ψ distributes $\sqcap$. Because of its importance we concentrate on just one such property: isotonicity of the embedding in an *ep*-pair.

Lemma 2.9 If $gc(\phi, \psi)$ from the second form of definition above then both ϕ and ψ are monotonic. If $ep(\phi, \psi)$ then ϕ is isotonic: $x \sqsubseteq_X y$ iff $\phi(x) \sqsubseteq_Y \phi(y)$.
Proof: Suppose that ϕ and ψ form a Galois connection from $(X, \sqsubseteq_X)$ to $(Y, \sqsubseteq_Y)$. Then

	$x \sqsubseteq_X y$	
implies	$x \sqsubseteq_X \psi(\phi(y))$	$y \sqsubseteq_X \psi(\phi(y))$ by Def. 2.8
iff	$\phi(x) \sqsubseteq_Y \phi(y)$	Def. 2.8

hence ϕ is monotonic. The proof that ψ is monotonic is similar. If ϕ and ψ form an *ep*-pair then the implication from the first to the second line of the proof becomes an equivalence, which shows that ϕ is isotonic. □

Lemma 2.10 If $gc(\phi, \psi)$ then ϕ distributes $\sqcup$ and ψ distributes $\sqcap$.
Proof: Suppose that ϕ and ψ form a Galois connection from $(X, \sqsubseteq_X)$ to $(Y, \sqsubseteq_Y)$. If Z is a directed subset of X and $y \in Y$ then

	$\phi(\sqcup Z) \sqsubseteq_Y y$	
iff	$\sqcup Z \sqsubseteq_X \psi(y)$	Def. 2.8
iff	$(\forall\ z\colon Z \cdot z \sqsubseteq_X \psi(y))$	
iff	$(\forall\ z\colon Z \cdot \phi(z) \sqsubseteq_Y y)$	Def. 2.8
iff	$\sqcup(\phi(Z)) \sqsubseteq_Y y.$	

Taking first $y = \phi(\sqcup Z)$ and then $y = \sqcup(\phi(Z))$ gives the desired equality. It can be proved in a similar way that ψ distributes $\sqcap$. □

Distribution of $\sqcup$ implies that ϕ is continuous, that is $\phi(\sqcup Z) = \sqcup(\phi(Z))$. However, for ψ to be continuous it also has to distribute $\sqcup$ which does not follow from its being part of a Galois connection. Thus we have the following definition.

Definition 2.11 Let $(X, \sqsubseteq_X)$ and $(Y, \sqsubseteq_Y)$ be *ipo*'s. An *ep*-pair ϕ and ψ is called a *strong* ep-*pair* iff ψ is continuous, in which case we write $sep(\phi, \psi)$. □

We now show that strong *ep*-pairs preserve compactness.

Lemma 2.12 If $sep(\phi, \psi)$ then ϕ preserves compactness in the sense that $x \in K(X)$ iff $\phi(x) \in K(Y)$.
Proof: Suppose that ϕ and ψ form a strong *ep*-pair from $(X, \sqsubseteq_X)$ to $(Y, \sqsubseteq_Y)$. The direct implication uses continuity of ψ; assume x is compact in X and that W is directed in $(Y, \sqsubseteq_Y)$.

	$\phi(x) \sqsubseteq_Y \sqcup W$	
implies	$x \sqsubseteq_X \psi(\sqcup W)$	Galois
implies	$x \sqsubseteq_X \sqcup(\psi W)$	ψ continuous
implies	$\exists w : W \cdot x \sqsubseteq_X \psi(w)$	$x \in K(X)$
implies	$\exists w : W \cdot \phi(x) \sqsubseteq_Y w$	Galois

Thus $\phi(x)$ is compact in Y.

The converse is similar but uses isotonicity of ϕ; assume that $\phi(x)$ is compact in Y and that Z is directed in $(X, \sqsubseteq_X)$.

	$x \sqsubseteq_X \sqcup Z$	
implies	$\phi(x) \sqsubseteq_Y \phi(\sqcup Z)$	ϕ monotonic
iff	$\phi(x) \sqsubseteq_Y \sqcup\phi(Z)$	ϕ continuous
implies	$\exists z : Z \cdot \phi(x) \sqsubseteq_Y \phi(z)$	$\phi(x) \in K(Y)$
implies	$\exists z : Z \cdot x \sqsubseteq_X z$	ϕ isotonic.

Thus x is compact in X. □

Theorem 2.13 If $sep(\phi, \psi)$ and $x \in K(X)$ then $\phi(x)\uparrow$ is Scott-open in $(Y, \sqsubseteq_Y)$ and

$$\phi^{-1}(\phi(x)\uparrow) = x\uparrow.$$

Proof: Suppose that $x \in K(X)$. To show that $\phi(x)\uparrow$ is Scott-open we must show that it is up-closed and inaccessible. The former is immediate; the latter follows from Lemma 2.12.

Next we argue:

$$
\begin{array}{lll}
 & y \in \phi^{-1}(\phi(x)\uparrow) & \\
\text{iff} & \phi(y) \in \phi(x)\uparrow & \\
\text{iff} & \phi(y) \sqsupseteq \phi(x) & \\
\text{iff} & y \sqsupseteq x & \phi \text{ isotonic} \\
\text{iff} & y \in x\uparrow. &
\end{array}
$$

□

3 Embedding-Projection pairs for CSP

Our concern now is to find strong *ep*-pairs between the standard CSP models.

3.1 Embedding Traces in Failures

The usual pair of functions used to relate the traces model and the failures model are the deterministic embedding and the traces projection. The deterministic embedding maps a traces-process to the deterministic failures-process with the same traces; the failures process is thus unable to refuse any event it can perform.

Definition 3.1 The *deterministic embedding* $\widehat{\diamond} : \mathcal{T} \to \mathcal{F}$ assigns to traces-process P the failures process $\widehat{P}$ defined

$$\widehat{P} \quad := \quad \{(t, X) \mid t \in P \wedge (\forall\, a : A \bullet t\langle a\rangle \in P \Rightarrow a \notin X)\}.$$

There and elsewhere the place-holder $\diamond$ assists us in denoting a function written as a decoration on its argument. □

It is easy to see that $\widehat{\diamond}$ is not monotonic, let alone continuous. For whilst $STOP$ is traces-refined by $a \to STOP$, it is not so failures-refined. Since the deterministic embedding cannot be lifted using the techniques of the previous section, we must consider an alternative embedding.

The function $\diamond^+$ maps a traces-process P to the failures-process P^+ which can refuse only those events which P cannot do.

Definition 3.2 The *liberal embedding* $\diamond^+ : \mathcal{T} \to \mathcal{F}$ assigns to traces-process P the failures-process P^+ defined

$$P^+ := \{(s, X) \mid \forall a : A \cdot s\langle a\rangle \in P \Rightarrow a \notin X\}.$$

□

Figure 2 shows some examples of the effect of $\diamond^+$ on processes with alphabet $\{a, b\}$. For brevity we express elements of the semantic spaces algebraically using the standard combinators. Note that *CHAOS*, the bottom process in $\mathcal{F}$, is able, on each interaction, to perform a or b or to deadlock; it is thus expressed algebraically

$$\mu X \cdot STOP \sqcap (a \to X) \sqcap (b \to X).$$

P	P^+
$STOP$	$CHAOS$
$a \to STOP$	$a \to CHAOS \sqcap (a \to CHAOS \,[\!]\, b \to CHAOS)$
$a \to STOP \,[\!]\, b \to STOP$	$a \to CHAOS \,[\!]\, b \to CHAOS$

Figure 2: Effect of $\diamond^+$, over alphabet $\{a, b\}$

As weak inverse to $\diamond^+$ we consider the function $\diamond^-$ which applied to a deterministic failures-process P gives the projection onto its traces; but if P behaves non-deterministically at some point, then P^- is pruned of any events that P can both do and refuse.

Definition 3.3 The *deterministic projection* $\diamond^- : \mathcal{F} \to \mathcal{T}$ assigns to each failures-process the traces-process defined

$$P^- := \{t \mid t \in dom\ P \wedge (\forall\ s : A^\star, a : A \cdot s\langle a\rangle \leq t \Rightarrow (s, \{a\}) \notin P)\}.$$

□

Figure 3 gives some examples of its effect.

Those definitions are justified by the following results.

P	P^-
$a \rightarrow CHAOS \parallel b \rightarrow CHAOS$	$a \rightarrow STOP \parallel b \rightarrow STOP$
$a \rightarrow CHAOS \sqcap b \rightarrow CHAOS$	$STOP$
$a \rightarrow CHAOS \sqcap (a \rightarrow CHAOS \parallel b \rightarrow CHAOS)$	$a \rightarrow STOP$

Figure 3: Effect of $\diamond^-$, over alphabet $\{a, b\}$

Lemma 3.4 $ep(\diamond^+, \diamond^-)$; that is, the functions $\diamond^+$ and $\diamond^-$ form an *ep*-pair from $(\mathcal{T}, \sqsubseteq_{\mathcal{T}})$ to $(\mathcal{F}, \sqsubseteq_{\mathcal{F}})$.

Proof: First we prove that for $P \in \mathcal{T}$, $P^{+-} = P$. By Def. 3.3

$$
\begin{array}{lll}
 & P^{+-} & \\
= & & \text{Def. 3.3} \\
 & \{t \mid t \in dom\ P^+ \ \wedge\ (\forall\ s : A^\star, a : A \cdot s\langle a\rangle \leq t \ \Rightarrow\ (s, \{a\}) \notin P^+)\} & \\
= & \{t \mid \forall s : A^\star, a : A \cdot s\langle a\rangle \leq t \ \Rightarrow\ (s, \{a\}) \notin P^+\} & dom\ P^+ = A^\star \\
= & \{t \mid \forall s : A^\star, a : A \cdot s\langle a\rangle \leq t \ \Rightarrow\ s\langle a\rangle \in P\}\ (s, \{a\}) \notin P^+ \text{ iff } s\langle a\rangle \in P & \\
= & \{t \mid \forall s : A^\star \cdot s \leq t \ \Rightarrow\ s \in P\} & \\
= & \{t \mid t \in P\} & P \text{ prefix closed} \\
= & P. &
\end{array}
$$

Now we prove that for $P \in \mathcal{F}$, $P^{-+} \sqsubseteq P$; that is, $P^{-+} \supseteq P$. We reason by case analysis.

$$
\begin{array}{ll}
 & (t, X) \in P \\
\text{iff} & (t, X) \in P \wedge\ (\forall s \leq t \cdot det(P, s) \\
 & \qquad\qquad \vee \\
 & \qquad\qquad \exists s \leq t \cdot \neg det(P, s)) \\
\text{iff} & \forall s \leq t \cdot det(P, s) \wedge (t, X) \in P \\
 & \vee \\
 & \exists s \leq t \cdot \neg det(P, s) \wedge (t, X) \in P \\
\text{implies} & \qquad\qquad \text{Def. 3.3 and of nondeterminism} \\
 & t \in dom\ P^- \wedge \forall a : A \cdot a \in X \Rightarrow t\langle a\rangle \notin P^- \\
 & \vee \\
 & \exists s \leq t \cdot \exists e : A \cdot \neg det(P, s, e) \wedge (t, X) \in P
\end{array}
$$

$$
\begin{array}{lll}
\text{implies} & (t, X) \in P^{-+} & \text{Def. 3.2} \\
& \vee & \\
& (t, X) \in P^{-+} & \\
\text{iff} & (t, X) \in P^{-+}. &
\end{array}
$$

The result then follows from monotonicity of $\diamond^+$ and $\diamond^-$. $\Box$

By Lemma 2.10 it follows that $\diamond^+$ is continuous; in fact $\diamond^-$ is also continuous.

Theorem 3.5 The functions $\diamond^+$ and $\diamond^-$ form a strong *ep*-pair: $sep(\diamond^+, \diamond^-)$.
Proof: If Z is a directed subset of $\mathcal{F}$, writing Z^- for $\{P^- \mid P \in Z\}$, we must show

$$(\sqcup Z)^- \quad = \quad \sqcup (Z^-).$$

One half of the equality is trivial:

$$
\begin{array}{lll}
& \forall P : Z \cdot P \sqsubseteq \sqcup Z & \\
\text{implies} & \forall P : Z \cdot P^- \sqsubseteq (\sqcup Z)^- & \text{monotonicity of } \diamond^- \\
\text{iff} & \sqcup(Z^-) \sqsubseteq (\sqcup Z)^-. &
\end{array}
$$

To prove the converse we exploit the semantic definition.

$$
\begin{array}{lll}
& (\sqcup Z)^- & \\
= & & \text{Def. 3.3} \\
& \{t \mid t \in dom\ (\sqcup Z) \wedge (\forall\ s : A^\star, a : A \cdot s\langle a \rangle \leq t \Rightarrow (s, \{a\}) \notin \sqcup Z)\} & \\
= & & \text{Def. of } \sqcup \text{ in } \mathcal{F} \\
& \{t \mid\ t \in \bigcap \{dom\ P \mid P \in Z\} & \\
& \quad \wedge & \\
& \quad \forall s : A^\star, a : A \cdot s\langle a \rangle \leq t \Rightarrow (s, \{a\}) \notin \bigcap \{P \mid P \in Z\}\ \} & \\
= & \{t \mid\ \forall P : Z \cdot t \in dom\ P & \\
& \quad \wedge & \\
& \quad \forall s : A^\star, a : A \cdot \exists P : Z \cdot s\langle a \rangle \leq t \Rightarrow (s, \{a\}) \notin P\ \} & \\
= & \{t \mid\ \forall P : Z \cdot t \in dom\ P & \text{see below} \\
& \quad \wedge & \\
& \quad \exists P : Z \cdot \forall s : A^\star, a : A \cdot s\langle a \rangle \leq t \Rightarrow (s, \{a\}) \notin P\ \} & \\
\subseteq & \{t \mid\ \exists P : Z \cdot t \in dom\ P & \\
& \quad \wedge & \\
& \quad \forall s : A^\star, a : A \cdot s\langle a \rangle \leq t \Rightarrow (s, \{a\}) \notin P\ \} &
\end{array}
$$

$$
\begin{array}{lll}
= & \bigcup\{\ \{t \mid t \in dom\ P \wedge (\forall\ s : A^\star, a : A \cdot s\langle a\rangle \leq t \Rightarrow (s, \{a\}) \notin P)\}\ \mid\ P \in Z\} & \\
= & \bigcup\{P^- \mid P \in Z\} & \\
= & \sqcup(Z^-) & \text{Def. of } \sqcup \text{ in } \mathcal{T}
\end{array}
$$

The step marked "see below" is justified as follows. Suppose that for some $P \in Z$ we have $(\langle\rangle, \{a_0\}) \notin P$. Then whenever $P \sqsubseteq R$ the same holds for R. Now suppose that for some $P' \in Z$, $(\langle a_0\rangle, \{a_1\}) \notin P'$. Then if Q refines both P and P', Q contains neither $(\langle\rangle, \{a_0\})$ nor $(\langle a_0\rangle, \{a_1\})$. Clearly we can proceed in this way for all prefixes of (finite) t, until we have found Q such that $(\forall\ s : A^\star, a : A \cdot s\langle a\rangle \leq t \Rightarrow (s, \{a\}) \notin Q)$. □

It can be shown from the definitions that $P^+ \parallel \widehat{P} = \widehat{P}$, that P^+ can be forced to behave as a deterministic embedding of P; from that follows in fact that P^+ is the *weakest* (that is, the least refined) failures-process satisfying that criterion.

Lemma 3.6 If $P \in \mathcal{T}$ and $Q \in \mathcal{F}$ then $P^+ \sqsubseteq Q$ iff $Q \parallel \widehat{P} = \widehat{P}$.
Proof: For the direct implication we reason

$$
\begin{array}{lll}
 & P^+ \sqsubseteq Q & \\
\text{implies} & P^+ \parallel \widehat{P}\ \sqsubseteq\ Q \parallel \widehat{P} & \parallel \text{ monotonic} \\
\text{implies} & \widehat{P}\ \sqsubseteq\ Q \parallel \widehat{P} & P^+ \parallel \widehat{P} = \widehat{P} \\
\text{iff} & \widehat{P} = Q \parallel \widehat{P} & \widehat{P} \text{ deterministic.}
\end{array}
$$

Conversely we argue by contrapositive

$$
\begin{array}{lll}
 & P^+ \not\sqsubseteq Q & \\
\text{iff} & P^+ \not\supseteq Q & \text{Defn. of order in } \mathcal{F} \\
\text{iff} & \exists (t, X) \in Q \setminus P^+ & \\
\text{iff} & \exists (t, X) \cdot (t, X) \in Q\ \wedge\ \exists a : A \cdot t\langle a\rangle \in P\ \wedge\ a \in X & \text{Def. 3.2} \\
\text{implies} & \exists t : A^\star, a : A \cdot (t, \{a\}) \in Q\ \wedge\ t\langle a\rangle \in P & \text{Property of failures} \\
\text{implies} & \exists t : A^\star, a : A \cdot (t, \{a\}) \in Q\ \wedge\ t\langle a\rangle \in \widehat{P}\ \wedge\ (t, \{a\}) \notin \widehat{P} & \text{Def. 3.1} \\
\text{implies} & & \text{Property of failures} \\
 & \exists t : A^\star, a : A \cdot (t, \{a\}) \in Q\ \wedge\ (t, \{\}) \in \widehat{P}\ \wedge\ (t, \{a\}) \notin \widehat{P} & \\
\text{implies} & \exists t : A^\star, a : A \cdot (t, \{a\}) \in (Q \parallel \widehat{P})\ \wedge\ (t, \{a\}) \notin \widehat{P} & \text{Def. of } \parallel \\
\text{implies} & Q \parallel \widehat{P} \neq \widehat{P} &
\end{array}
$$

□

Similarly it can be shown that if Q is a failures-process then Q^- is the *strongest* (or most refined) traces-process which can be forced to behave as the deterministic projection of Q.

Lemma 3.7 If $P \in \mathcal{T}$ and $Q \in \mathcal{F}$ then $P \sqsubseteq Q^-$ iff $\widehat{P} \parallel Q = \widehat{P}$. □

Whilst Lemmas 3.6 and 3.7 combine to show that $\diamond^+$ and $\diamond^-$ form a Galois connection, they are not strong enough to show that they form an *ep*-pair (the conclusion of Lemma 3.4).

3.2 Embedding Failures in Divergences

We now turn to embedding the failures model in the divergences model. The usual way of doing so is by mapping every process in $\mathcal{F}$ to the divergence-free process with the same failures. This is continuous and could therefore be lifted, but it has the disadvantage that no failures-process is mapped to $CHAOS'$, the bottom process of the divergences model. Consequently it does not form a Galois connection with the projection from the divergences onto the failures.

We therefore use a different, more pessimistic, embedding which defines the set of divergences of an embedded process P as the set of traces after which P can do or refuse anything.

Definition 3.8 The *divergent embedding* $\oplus : \mathcal{F} \to \mathcal{D}$ assigns to failures-process P the divergences-process defined

$$P\oplus \quad := \quad (P, D) \text{ where } D := \{s : A^\star \mid \forall t : A^\star, X \subseteq A \bullet (st, X) \in P\}.$$

□

As weak inverse to $\oplus$ we consider the projection from (failures and) divergences onto failures.

Definition 3.9 The *divergent projection* $\ominus : \mathcal{D} \to \mathcal{F}$ assigns to divergences-process (F, D) the failures-process defined

$$(F, D)\ominus \quad := \quad F.$$

□

Theorem 3.10 The functions Φ and Θ form a strong *ep*-pair, $sep(\Phi, \Theta)$.
Proof: Firstly for $P \in \mathcal{F}$, the definitions yield $P\Phi\Theta = P$.

Secondly for $P = (F, D) \in \mathcal{D}$ we must show $P\Theta\Phi \sqsubseteq P$; that is, if $P\Theta\Phi = (G, E)$ then $G \supseteq F$ and $E \supseteq D$. The former is immediate from the definitions; for the latter we reason

$$\begin{array}{lll} & E & \\ = & \{s : A^\star \mid \forall t : A^\star, X \subseteq A \bullet (st, X) \in P\Theta\} & \text{Def. 3.8} \\ = & \{s : A^\star \mid \forall t : A^\star, X \subseteq A \bullet (st, X) \in F\} & \text{Def. 3.9} \\ \supseteq & D & \text{Property of divergences.} \end{array}$$

Thus $ep(\Phi, \Theta)$.

For continuity of Θ suppose that Z is a directed subset of $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$. Then

$$\begin{array}{lll} & \sqcup(Z\Theta) & \\ = & \bigcap\{(F, D)\Theta \mid (F, D) \in Z\} & \text{Def. of order in } \mathcal{D} \\ = & \bigcap\{F \mid (F, D) \in Z\} & \text{Def. 3.9} \\ = & (\bigcap Z)\Theta & \\ = & (\sqcup Z)\Theta & \text{Def. of order in } \mathcal{D}. \end{array}$$

□

Evidently the composition of the two strong *ep*-pairs gives a strong *ep*-pair from the traces model to the divergences model: $sep(\Phi \circ \Diamond^+, \Diamond^- \circ \Theta)$. We will denote the composed embeddings and projections respectively by $\Diamond$ and $\Diamond$.

4 Probabilistic Lifting

In this section we summarise Jones's functorial probabilistic lifting and our application of it to CSP.

4.1 The Lifting Defined

The probabilistic functor applied to an *ipo* $(X, \sqsubseteq)$ lifts it to an *ipo* $(\overline{X}, \sqsubseteq)$, defined in terms of the Scott-topology of $(X, \sqsubseteq)$. We use the same symbol $\sqsubseteq$ for the order of both spaces to avoid clutter; context will distinguish them.

The elements of $\overline{X}$ are evaluations continuous with respect to the Scott topology ([4, pp. 50,66]). An evaluation is like a probability measure, as the conditions below attest, but defined only on the Scott-open sets of $(X, \sqsubseteq)$.

Definition 4.1 If $(X, \sqsubseteq)$ is an *ipo* with Scott topology $\mathcal{S}$, a function E from $\mathcal{S}$ to $[0, 1]$ is said to be a *continuous evaluation* if it satisfies:

1. (*co*)-*strictness*: $E(\{\}) = 0$ and $E(X) = 1$
2. *monotonicity*: for any $Y, Z \in \mathcal{S}$, if $Y \subseteq Z$ then $E(Y) \leq E(Z)$
3. *modularity*: if $Y, Z \in \mathcal{S}$ then $E(Y \cup Z) = E(Y) + E(Z) - E(Y \cap Z)$
4. *continuity*: if $\mathcal{Y}$ is a $\subseteq$-directed subset of $\mathcal{S}$ then $E(\bigcup \mathcal{Y}) = \bigsqcup\{E(Y) \mid Y \in \mathcal{Y}\}$.

We write $\overline{X}$ for the set of all continuous evaluations on $(X, \sqsubseteq)$. The pointwise order on continuous evaluations equips $\overline{X}$ with a partial order, also denoted $\sqsubseteq$, defined for $D, E \in \overline{X}$,

$$D \sqsubseteq E \quad := \quad D(S) \leq E(S) \text{ for each } S \in \mathcal{S}.$$

The resulting space $(\overline{X}, \sqsubseteq)$ is an *ipo*. □

That summarises the action of the probabilistic functor on objects (that is, on *ipo*'s). We write the functor as $\overline{\diamond}$, where the place holder $\diamond$ can be instantiated with an object or a morphism. The effect of $\overline{\diamond}$ on morphisms is to lift a continuous function between *ipo*'s into a function between continuous evaluations on *ipo*'s.

Definition 4.2 If $(X, \sqsubseteq_X)$ and $(Y, \sqsubseteq_Y)$ are *ipo*'s and f is a continuous function from X to Y then $\overline{f}$ from $\overline{X}$ to $\overline{Y}$ is defined, for a continuous evaluation A over X and a Scott-open set S in Y, to assign to S the same probability as the evaluation A assigns to the inverse image of S through f

$$\overline{f}(A)(S) \quad := \quad A(f^{-1}(S)).$$

□

Having recalled the definition of the functor $\overline{\diamond}$, we now justify its description as a lifting: *ipo* $(\overline{X}, \sqsubseteq)$ contains (a copy of) the *ipo* $(X, \sqsubseteq)$. Indeed the function which takes an element of X to the point mass at that element is a continuous injection. We shall abuse notation and also use $\overline{\diamond}$ for that embedding of an *ipo* in its probabilistic lifting.

Definition 4.3 For any $x \in X$, its image $\overline{x} \in \overline{X}$ is the continuous evaluation defined, for any Scott-open set S, by

$$\begin{array}{lll} \overline{x}(S) & := & 1, \text{ if } x \in S \\ & & 0, \text{ otherwise.} \end{array}$$

It is straightforward to show that the embedding is continuous (hence monotonic though not in general isotonic). □

In Section 7 we shall find a weak inverse to $\overline{\diamond}$, so that the two functions together form an *ep*-pair.

4.2 The Lifting Applied to CSP

In the lifted CSP models studied in this paper, the only operators not defined by lifting are probabilistic choice and recursion. The former is defined as the weighted average of two lifted processes, whereas the latter exploits the fact that the lifted model is itself an *ipo* to define a recursive process as a least fixed point.

For the rest of the paper we will use the word *standard* to mean either a member of an unlifted space, or a member of a lifted space $\overline{X}$ which lies in the range of the point-mass embedding. We think of the members of $\overline{X}$ as being probabilistic versions of the members of X. Standard members are those whose probability distribution is concentrated at a single point. We shall thus speak of $(\overline{X}, \sqsubseteq)$ as the *probabilistic lifting* of *ipo* $(X, \sqsubseteq)$.

In [5] the probabilistic lifting of the divergences model $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$ of CSP is studied in some detail and called PCSP. In Section 6 of the present paper, the probabilistic lifting of a subset (without external choice) of the traces *ipo* of CSP, $(\mathcal{T}, \sqsubseteq_{\mathcal{T}})$, will be identified with a previous model of Seidel [8].

Due to Theorem 2.7 it turns out that the three models of CSP have their continuous evaluations determined by their restriction to the set of cones on compact (that is, finite) processes (see [5, Thm.4.7]). Thus for each of those

ipo's we introduce special notation for the value of a continuous evaluation A at the cone $F{\uparrow}$ on a finite process F:

$$F \sqsubseteq\!\!\!\sim A \quad := \quad A(F{\uparrow}).$$

Such notation, (from [5] for the case $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$), is conveniently read as the probability with which probabilistic process A behaves like a refinement of F. In the special case that $A = \overline{P}$ is standard, $F \sqsubseteq\!\!\!\sim \overline{P}$ holds iff P refines F; that justifies the notation. F can also be regarded as a test, in which case $F \sqsubseteq\!\!\!\sim A$ is the probability that A passes the test.

5 The Probabilistic Hierarchy

We are now able to construct mappings between the probabilistic models of Figure 1, which in some sense preserve probabilities. We do so by lifting the strong *ep*-pairs between standard models.

Suppose that f is a continous function between *ipo*'s $(X, \sqsubseteq_X)$ and $(Y, \sqsubseteq_Y)$. From the previous section we know by lifting that $\overline{f}$ is continuous from $(\overline{X}, \sqsubseteq_{\overline{X}})$ to $(\overline{Y}, \sqsubseteq_{\overline{Y}})$. If we now take for $(X, \sqsubseteq_X)$ and $(Y, \sqsubseteq_Y)$ standard models of CSP connected by a strong *ep*-pair, that lifting applied to each member of the pair provides a strong *ep*-pair on the liftings, as illustrated in Figure 4.

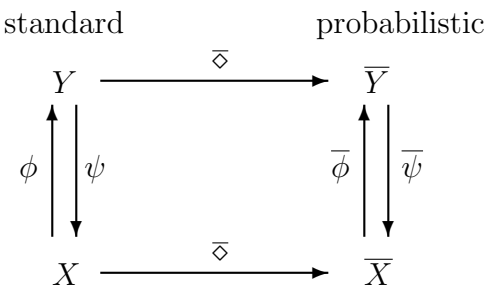

Figure 4: A lifted *ep*-pair

The lifting preserves probabilities in the sense of the following result.

Theorem 5.1 Suppose that $(X, \sqsubseteq_X)$ and $(Y, \sqsubseteq_Y)$ are *ipo*'s with $x \in K(X)$, and that $sep(\phi, \psi)$. Then the test "$x \mathrel{\underset{\sim}{\sqsubseteq}}$" in $\overline{X}$ is equivalent to the test "$\phi(x) \mathrel{\underset{\sim}{\sqsubseteq}}$" in $\overline{Y}$.

Proof: Recall that $A \in \overline{X}$ meets the test $x \mathrel{\underset{\sim}{\sqsubseteq}} A$ with probability $A(x\uparrow)$, which is well defined since x is compact. Similarly $\overline{\phi}(A) \in \overline{Y}$ meets the test $\phi(x) \mathrel{\underset{\sim}{\sqsubseteq}} \overline{\phi}(A)$ with probability $\overline{\phi}(A)((\phi(x))\uparrow)$, which is again well defined since compactness of $\phi(x)$ is ensured by Theorem 2.13. It thus remains to show equality of those two probabilities.

But

$$\begin{array}{lll} & \overline{\phi}(A)((\phi(x))\uparrow) & \\ = & A(\phi^{-1}((\phi(x))\uparrow) & \text{Def. 4.2} \\ = & A(x\uparrow) & \text{Thm. 2.13} \end{array}$$

as desired. □

To summarise the properties of the embedding Φ from $\mathcal{T}$ to $\mathcal{D}$ recall that in the probabilistic models, the meaning of a probabilistic process is determined by the probabilities which it assigns to the cones. The probability of each cone can be viewed as the probability with which the probabilistic process passes a (standard) test. A process refines another if it passes all tests with the same or higher probability. The standard embedding Φ is based on the idea that in the standard traces model refinement is equivalent to testing whether or not a process can be forced to behave as the process it supposedly refines. It translates trace-tests into equivalent tests in $\mathcal{D}$.

The lifted embedding $\overline{\Phi}$ from $\overline{\mathcal{T}}$ to $\overline{\mathcal{D}}$ identifies for each probabilistic traces-process A the probabilistic divergences-process $A\overline{\Phi}$ which will pass all translated tests with the same probability as A. By restricting the kind of test we may make in the probabilistic divergences model, we lose the ability to distinguish certain processes. The difference between the distinctions made by the restricted tests and the full range of tests represents the difference between reasoning in the probabilistic traces model, and the probabilistic failures-divergences model.

For probabilistic divergences-process $A \in \overline{\mathcal{D}}$ define $[A]$ to be the equivalence class of such processes which cannot be distinguished from A by lifted tests:

$$[A] \;:=\; \{A' : \mathcal{D} \mid \forall D : \mathcal{T} \cdot D\Phi \mathrel{\underset{\sim}{\sqsubseteq}} A = D\Phi \mathrel{\underset{\sim}{\sqsubseteq}} A'\}.$$

The Smyth-ordering ([9]) on equivalence classes is

$$[A] \sqsubseteq_S [B] \;\; := \;\; \forall B' \in [B] \cdot \exists A' \in [A] \cdot A' \sqsubseteq B'.$$

Hennessey [2] calls that the *must*-ordering, and indeed it amounts to saying that B can be forced to do more than A. For example, let

$$\begin{array}{rcl} A & = & a \to STOP \;{}_{1/2}\!\oplus (a \to STOP \parallel b \to STOP) \\ B & = & a \to STOP \parallel b \to STOP. \end{array}$$

Then $[A] \sqsubseteq_S [B]$, informally because A can always be forced to behave as $a \to STOP$, but can be forced to behave as $b \to STOP$ only with probability $1/2$, whereas B can always be forced to behave as either. Formally, the equivalence class $[B]$ is the singleton set $\{B\}$, whereas $[A]$ contains the process

$$\begin{array}{rcl} A' := & & (a \to STOP \sqcap (a \to STOP \parallel b \to STOP)) \\ & {}_{1/2}\!\oplus & (a \to STOP \parallel b \to STOP) \end{array}$$

and $A \sqsubseteq A'$.

6 Embedding another Probabilistic Model

In this section we show that one of the models in Seidel's thesis [8] can be viewed as a probabilistic lifting which can be embedded in the lifting $\overline{\mathcal{T}}$ of the traces model via a strong *ep*-pair.

6.1 Seidel's Model $\mathcal{S}$

In Seidel's model a process is defined to be a measure on the space A^ω of infinite traces of events from universe A. Each finite trace is represented by appending to it a tail of τ's, for fixed distinguished element τ. For present purposes we find it more convenient to use instead the isomorphic space

$$\mathcal{O} \;\; := \;\; A^\star \cup A^\omega$$

of finite and infinite traces. Measurable sets are provided by the Borel field $\mathcal{B}$ generated by the sets of traces which are extensions of some finite trace.

Each process is denoted by a measure. For example the atomic process $STOP$ is denoted by the point measure on the empty trace. Probabilistic

choice is denoted by the weighted average of measures. A recursive process is denoted by a fixed point of a recursive equation, which is shown to be unique under certain conditions. All other operators are defined by transformation of measures, in the following way. Given a measure P on the measurable space $(\mathcal{O}, \mathcal{B})$ and a measurable function $f : \mathcal{O} \to \mathcal{O}'$ to some measurable space $(\mathcal{O}', \mathcal{B}')$, the measure P is transformed into a measure P_f on $(\mathcal{O}', \mathcal{B}')$ by setting, for any set $A \in \mathcal{B}'$,

$$P_f\, A \quad := \quad P(f^{-1}A). \tag{1}$$

For example the prefixing operator is defined

$$(a \to P)A \quad := \quad P(f_a^{-1}A)$$

using the measurable transformation $f_a : \mathcal{O} \to \mathcal{O}$ defined, for $u \in \mathcal{O}$,

$$f_a(u) \quad := \quad \langle a \rangle u.$$

The result is a model of CSP with prefixing, probabilistic choice, hiding, parallel composition, sequential composition, interleaving, relabelling, and recursion. We call it $\mathcal{S}$.

6.2 Model $\mathcal{S}$ as a Lifting

To view the model $\mathcal{S}$ as a lifted space we consider the space $(\mathcal{O}, \sqsubseteq_{\mathcal{O}})$ of finite and infinite traces over A, with the prefix ordering: for $t, s \in \mathcal{O}$,

$$t \sqsubseteq s \quad := \quad (t = s \text{ or } t \text{ is a finite prefix of } s).$$

The space $(\mathcal{O}, \sqsubseteq_{\mathcal{O}})$ is an *ipo* because every directed set of traces is a chain and so contains a least upper bound. It also follows as a matter of routine that a trace is compact iff it is finite and that each trace is the least upper bound of a directed set of finite traces, so that $(\mathcal{O}, \sqsubseteq_{\mathcal{O}})$ is algebraic. Since $(\mathcal{O}, \sqsubseteq_{\mathcal{O}})$ clearly also has partial joins we deduce that each evaluation on $\mathcal{O}$ is determined by its restriction to the set of cones $t\uparrow$ on finite traces t. In summary:

Lemma 6.1 $(\mathcal{O}, \sqsubseteq_{\mathcal{O}})$ is an algebraic *ipo* with partial joins; an element is compact iff it is a finite trace. □

Next we identify $\mathcal{S}$ as a lifted space.

Theorem 6.2 There is a construction-preserving bijection from $\mathcal{S}$ to $\overline{\mathcal{O}}$, in the sense that each CSP operator defined in $\mathcal{S}$ by measurable transformation f corresponds to the restriction, to the Scott topology of $\mathcal{O}$, of the lifting to $\overline{\mathcal{O}}$ of f.

Proof: The set of cones on finite traces generates both the Borel field $\mathcal{B}$ with respsect to which $\mathcal{S}$ is defined, and the Scott topology of $\mathcal{O}$. In particular any measure on $(\mathcal{O}, \mathcal{B})$, when restricted to the Scott topology, gives a continuous evaluation on $\mathcal{O}$. Conversely every continuous evaluation extends to a unique measure ([4, Thm. 5.13]).

We thus define $\chi : \mathcal{S} \to \overline{\mathcal{O}}$ by setting $\chi(P)$ to equal the restriction of measure P to the Scott topology of $\mathcal{O}$. By the comments of the previous paragraph, χ is bijective.

Next we observe that for the atomic process $STOP$

$$\chi(STOP) = \overline{\langle\rangle},$$

since each equals the point mass evaluated at $\langle\rangle$.

Now for any unary measurable function f on $\mathcal{O}$ (the case of n-ary f is similar) it follows easily, using the notation P_f from equation (1), that

$$\chi(P_f) = \overline{f}(\chi(P)).$$

Each recursive process is constructed as a least fixed point in the lifted space and as a weak limit in $\mathcal{S}$; and again the two constructions coincide on the Scott topology. That establishes the claim. □

6.3 Embedding $\mathcal{S}$ in $\overline{\mathcal{T}}$

Having shown that the model $\mathcal{S}$ can be identified with the probabilistic lifting $(\overline{\mathcal{O}}, \sqsubseteq)$, we can now embed it into an appropriate subset of probabilistic deterministic CSP $(\overline{\mathcal{T}}, \sqsubseteq)$. We do so by lifting an embedding of the underlying standard models $(\mathcal{O}, \sqsubseteq_{\mathcal{O}})$ into a subset of $(\mathcal{T}, \sqsubseteq_{\mathcal{T}})$.

Recall that the sublanguage $\mathcal{S}$ of CSP has no internal or external choice. The appropriate subset of $\mathcal{T}$ therefore consists of those sets that are the prefix closures of a single trace (finite or infinite), and the embedding projection pair is then trivial: the embedding of a trace from $\mathcal{S}$ is the set of its finite

prefixes; and the projection of an external-choice-free process is $\sqsubseteq$-limit of all its traces, which is trivially continuous.

That completes our identification of model $\mathcal{S}$ with a subset of the probabilistic deterministic model, $\overline{\mathcal{T}}$, of CSP.

7 Weak inverse of $\overline{\diamond}$

We have relied heavily on the embedding $\overline{\diamond}$ of an *ipo* in its probabilistic lifting. In this section we provide a projection, *strip*, with which it forms an *ep*-pair if the underlying space is well behaved.

Let $(X, \sqsubseteq_X)$ be an *ipo* in which each finite subset has a least upper bound. Examples are provided by the three models of CSP.

Definition 7.1 The function $strip : \overline{X} \to X$ is defined, for any $A \in \overline{X}$,

$$strip(A) \quad := \quad \sqcup \{x : K(X) \mid A(x\uparrow) = 1\}.$$

However we must show that *strip* is well defined. Firstly the set above is nonempty since X has a least element (the least upper bound of the empty set) whose cone equals X and is thus mapped to 1 by each $A \in \overline{X}$. It remains to show that the set above is directed. But if $x, y \in K(X)$ then

	$x, y \in \{z : K(X) \mid A(z\uparrow) = 1\}$	
iff	$A(x\uparrow) = 1 = A(y\uparrow)$	
implies	$A(x\uparrow \cap y\uparrow) = 1$	Def. 4.1
implies	$A((x \sqcup y)\uparrow) = 1$	$x \sqcup y$ exists
iff	$x \sqcup y \in \{z : K(X) \mid A(z\uparrow) = 1\}$.	$x \sqcup y$ finite since x, y are

□

Theorem 7.2 If $(X, \sqsubseteq_X)$ is an algebraic *ipo* in which each finite subset has a least upper bound then the functions $\overline{\diamond}$ and *strip* form an *ep*-pair from X to $\overline{X}$: $ep(\overline{\diamond}, strip)$.

Proof: Firstly we show that for each $x \in X$, $strip(\overline{x}) = x$. But for such x,

	$strip(\overline{x})$	
=	$\sqcup\{y : K(X) \mid \overline{x}(y\uparrow) = 1\}$	Def. 7.1

$$
\begin{array}{lll}
= & \sqcup\{y : K(X) \mid y \sqsubseteq x\} & \\
= & x. & X \text{ algebraic}
\end{array}
$$

Next we show that for each $A \in \overline{X}$, $\overline{strip(A)} \sqsubseteq A$. But for such A,

$$
\begin{array}{lll}
 & \overline{strip(A)} & \\
= & \overline{\sqcup\{x : K(X) \mid A(x\uparrow) = 1\}} & \text{Def. 7.1} \\
= & \sqcup\{\overline{x} \mid x \in K(X) \wedge A(x\uparrow) = 1\} & \text{Lemma 5.1} \\
= & \sqcup\{\overline{x} \mid x \in K(X) \wedge \overline{x} \sqsubseteq A\} & \text{Def. 4.1} \\
\sqsubseteq & A. &
\end{array}
$$

□

However *strip* is not continuous, so the *ep*-pair is not strong. To show that, we consider the divergences model $(\mathcal{D}, \sqsubseteq_{\mathcal{D}})$ of CSP in which the effect of *strip* is essentially to convert probabilistic choice into non-deterministic choice.

Lemma 7.3 If $A, B \in \overline{\mathcal{D}}$ and $0 < p < 1$ then $strip(A \,{}_p\!\oplus B) = strip(A \sqcap B)$.
Proof:

$$
\begin{array}{lll}
 & strip(A \,{}_p\!\oplus B) & \\
= & \sqcup\{F : K(\mathcal{D}) \mid (A \,{}_p\!\oplus B)(F\uparrow) = 1\} & \\
= & \sqcup\{F : K(\mathcal{D}) \mid pA(F\uparrow) + (1-p)B(F\uparrow) = 1\} & \text{[5, Lemma 8.2]} \\
= & \sqcup\{F : K(\mathcal{D}) \mid A(F\uparrow) = 1 = B(F\uparrow)\} & \text{convexity} \\
= & \sqcup\{F : K(\mathcal{D}) \mid A(F\uparrow) \times B(F\uparrow) = 1\} & \\
= & \sqcup\{F : K(\mathcal{D}) \mid (A \sqcap B)(F\uparrow) = 1\} & \text{[5, Lemma 9.5]} \\
= & strip(A \sqcap B). & \text{Def. 7.1}
\end{array}
$$

□

Now observe that *strip* is not continuous since in $\mathcal{D}$ process $A \,{}_p\!\oplus B$ approaches A as p approaches 0 yet, for $p \neq 0$, $strip(A \,{}_p\!\oplus B) = A \sqcap B$.

8 Conclusion

We have suggested that the standard hierarchy of models of CSP be defined by strong *ep*-pairs between their members, and have replaced the standard

functions defined between those models by new functions which form strong *ep*-pairs. The $\diamond^+$-embedding maps a traces-process P to the failures-process which can refuse only those events P cannot do. The $\diamond^-$-projection applied to a deterministic failures-process P is just the usual projection onto the traces, but if P behaves non-deterministically at some point then P^- is pruned of any events that P can both do and refuse. The Φ-embedding defines the set of divergences of a failures-process P as the set of traces after which P can do or refuse anything and, unlike the usual embedding in the divergence-free processes of $\mathcal{F}$, forms a strong *ep*-pair with the projection from $\mathcal{D}$.

Lifting provides a functorial and straightforward mechanism for constructing a probabilistic *ipo* from a given *ipo*. By applying it to the standard hierarchy we have obtained a corresponding hierarchy of probabilistic models of CSP again linked by strong *ep*-pairs. We have shown that because the embeddings preserve compactness (which in CSP is finiteness), a test $F \sqsubseteq\!\!\!\sim A$ in one model can be embedded as a test in another model simply by embedding F and lifting A. Furthermore the lifted embeddings preserve probabilities. Such reasoning shows that the different probabilistic models can be characterised by the kinds of test they admit to distinguish processes. The probabilistic traces model only allows tests about what a process must do.

Hierarchies of models of CSP have been constructed by Reed, Schneider and Davies (see eg. [6, 7]) for much the same reason as we consider them here; we now comment on connections between their approach and ours. In their context, timed models of CSP, the process space does not have a complete refinement order and so they are unable to avail themselves of a construction as simple as that of *ep*-pairs. Instead they are forced to use the space of predicates (a space which is complete) to capture behavioural specifications, and their results can then be viewed in a format like ours as follows.

An embedding from untimed specifications into timed specifications can be defined by setting, for predicate S on $\mathcal{T}$, the *strongest timewise refinement of* S to be the predicate

$$\exists P \cdot S(P) \wedge (tstrip(traces(Q)) \subseteq traces(P)).$$

A projection in the reverse direction can be defined by setting, for predicate T on the timed model, the *weakest coarsening of* T to be the predicate

$$\forall Q \cdot (tstrip(traces(Q)) \subseteq traces(P)) \Rightarrow T(Q).$$

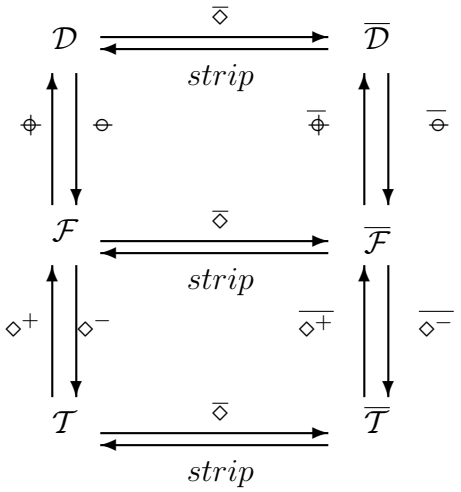

Figure 5: The final hierarchy

Together they form a Galois connection on predicates and an *ep*-pair when restricted to predicates which they call *behavioural specifications*.

Seen from that perspective, the proof rule given by Schneider and Davies ([1]) for hierarchical reasoning about timed processes has the same general form as the one suggested in this paper for hierarchical reasoning about probabilistic processes.

We have also shown that lifting itself forms an *ep*-pair with the projection *strip*, which maps probabilistic processes to standard ones. This projection can be used to prove non-probabilistic properties of probabilistic processes, in the same way that the projections from timed to untimed models were used by Davies and Schneider [1], [7], to prove untimed properties of timed processes.

Acknowledgements

The authors would like to thank Jeff Sanders and Annabelle McIver for their constructive criticism and extensive contributions to this paper.

References

[1] J. Davies and S. Schneider. Using CSP to verify a timed protocol over a fair medium. In *CONCUR 92*, number 630 in LNCS. Springer Verlag, 1992.

[2] M. de Nicola and M. Hennessy. Testing equivalence for processes. *Theoretical Computer Science*, 34, 1984.

[3] C. A. R. Hoare. *Communicating Sequential Processes*. Prentice Hall International, 1985.

[4] C. Jones. Probabilistic nondeterminism. Monograph ECS-LFCS-90-105, Edinburgh Univ. Edinburgh, U.K., 1990. (PhD Thesis).

[5] C. C. Morgan, A. K. McIver, K. Seidel, and J. W. Sanders. Refinement-oriented probability for CSP. *Formal Aspects of Computing*, 8(6):617–647, 1996.

[6] G.M. Reed. *A Uniform Mathematical Theory for Real-Time Distributed Computing*. PhD thesis, Oxford University, 1988.

[7] S. Schneider. Correctness and communication in real-time systems. Technical Monograph PRG-84, Oxford University, 1990. (DPhil Thesis).

[8] K. Seidel. Probabilistic communicating processes. Technical Monograph PRG-102, Oxford University, 1992. (DPhil Thesis).

[9] M. B. Smyth. Power domains and predicate transformers: a topological view. In *Automata, Languages and Programming 10th Colloquium, Barcelona, Spain*, number 154 in LNCS. Springer Verlag, 1983.

[10] J. van Leeuwen. *Handbook of Theoretical Computer Science*. Elsevier Science Publishers, 1990.