# Two Papers on CSP

by

**A.W. Roscoe**



Technical Monograph PRG-67
ISBN 0-902928-49-X
July 1988

Oxford University Computing Laboratory
Programming Research Group
8-11 Keble Road
Oxford OX1 3QD
England

Oxford University Computing Laboratory
Programming Research Group
8-11 Keble Road
Oxford OX1 3QD
England

# An alternative order for the failures model

by A.W. Roscoe[1]

**Oxford University Computing Laboratory,**
**8-11 Keble Road, Oxford OX1 3QD, U.K.**

**0. Introduction.** This paper introduces an alternative, coarser, partial order on the (improved) failures model of [BR] (sometimes called the failures-divergences model). The new order gives *exactly* the same semantics to CSP as the old one. As well as being of intrinsic interest for this reason the new order allows one to establish some interesting new results about the semantics of CSP and also (if desired) to extend the model to encompass certain extra forms of unbounded nondeterminism.

The present failures model $\mathcal{N}$, with its explicit treatment of divergence, was independently introduced in [R,B] to overcome various technical difficulties in the "pure" failures model of [HBR, BHR]. In the earlier model we had based our partial order on nondeterminism

$$P \sqsupseteq Q \Leftrightarrow P \subseteq Q$$

and so it was natural that this idea should be extended to the improved model:

$$P \sqsupseteq Q \Leftrightarrow \mathcal{D}[\![P]\!] \subseteq \mathcal{D}[\![Q]\!] \wedge \mathcal{F}[\![P]\!] \subseteq \mathcal{F}[\![Q]\!] .$$

This order works very well, of course. $P \sqsupseteq Q$ just when $P$ "improves" $Q$, or when $P$ is more deterministic than $Q$. $\sqsupseteq$ is a complete partial order, though

[1]The author gratefully acknowledges that the work reported in this paper was supported by ONR grant N00014-87-G-0242.

when the underlying alphabet of communications is infinite the compactness condition

$$(\forall X \in \wp(Y).(s,X) \in \mathcal{F}[\![P]\!]) \Rightarrow (s,Y) \in \mathcal{F}[\![P]\!]$$

is necessary for this. (This assumption turns out to be somewhat weaker than an assumption of finite nondeterminism.) In this paper we will sometimes refer to this order as the *nondeterminism* order.

There are good intuitive reasons for expecting all CSP operations to be monotonic and continuous with respect to $\sqsupseteq$, and so they are. That the order produces the right semantics for recursion is demonstrated by the congruence of the failures denotational semantics and its operational semantics [BRW] as well as the intnitively attractive idea that the recursive term $\mu P.F(P)$ must, in this order, always denote the most nondeterministic solution to

$$P = F(P).$$

**Summary of definitions and notation.** The reader should consult earlier works, particularly [H], for the syntax of CSP and the meaning of its constructs. This section contains a summary of the technical details of the model $\mathcal{N}$ and the semantics of CSP in that model. It assumes a knowledge of standard notations concerning traces.

$\mathcal{N}$ is defined relative to some non-empty alphabet $\Sigma$ of communications which is fixed for all processes. This parameter will generally be understood rather than mentioned explicitly. Each syntactic CSP process $P$ will be identified with an element of $\mathcal{N}_\Sigma$. Thus we will not use process alphabets ($\alpha P$) as part of the semantics, unlike [H]. The use of process alphabets produces a trivially isomorphic theory where each process is identified with $\mathcal{N}_{\alpha P}$, where $\alpha P$ is the process' own alphabet. The advantage of this, essentially typed, theory is that the parallel operator does not require the explicit mention of alphabets. Its disadvantages for theoretical work like that in this paper are that more mathematical housekeeping is required and that the theory of mutual recursion becomes rather messy.

$\mathcal{N}$ consists of all pairs $P = \langle F, D\rangle$ such that $F \subseteq \Sigma^* \times \mathcal{P}(\Sigma)$ and $D \subseteq \Sigma^*$, and such that for $s, t \in \Sigma^*$ and $X, Y \subseteq \Sigma$

(1) $$F \neq \emptyset \ \wedge \ (st,\emptyset) \in F \Rightarrow (s,\emptyset) \in F$$

(2) $$(t,X) \in F \wedge Y \subseteq X \ \Rightarrow \ (t,Y) \in F$$

(3) $$(t,X) \in F \wedge \forall a \in Y.(t\langle a\rangle,\emptyset) \notin F \ \Rightarrow \ (t, X \cup Y) \in F$$

(4) $$(\forall X' \subseteq X.X' \mathit{finite} \Rightarrow (t,X') \in F) \ \Rightarrow \ (t,X) \in F$$

(5) $$s \in D \ \Rightarrow \ st \in D$$

(6) $$s \in D \ \Rightarrow \ (st,X) \in F.$$

Let $P = \langle F, D\rangle \in \mathcal{N}$ be a process. Define

$$\begin{aligned} \mathcal{F}[\![P]\!] &= F \\ \mathcal{D}[\![P]\!] &= D \\ \mathit{traces}(P) &= \{t \mid (t,\emptyset) \in F\} \\ \mathit{refusals}(P) &= \{X \mid (\langle\rangle, X) \in F\} \end{aligned}$$

and for $s \in \mathit{traces}(P)$ define

$$P \mathit{\ after\ } s = \langle \{(t,B) \mid (st,B) \in F\}, \{t \mid st \in D\}\rangle$$

and for $s \in \Sigma^*$ define

$$\mathcal{R}[\![P]\!]s = \{X \mid (s,X) \in \mathcal{F}[\![P]\!]\}$$

so that $\mathcal{R}[\![P]\!]s = \mathit{refusals}(P \mathit{\ after\ } s)$ when $s \in \mathit{traces}(P)$. For $s \in \mathit{traces}(P)$, $P \mathit{after} s$ is a process.

The determinism order $\sqsubseteq$ is defined on $\mathcal{N}$ by

$$P \sqsubseteq Q \;\Leftrightarrow\; \mathcal{D}[\![P]\!] \supseteq \mathcal{F}[\![Q]\!] \wedge \mathcal{F}[\![P]\!] \supseteq \mathcal{D}[\![Q]\!].$$

$\langle \mathcal{N}, \sqsubseteq\rangle$ is a complete partial order: $\bot = \langle \Sigma^* \times \mathcal{P}(\Sigma), \Sigma^*\rangle$ is the least element, and if $\Delta$ is a directed subset of $\mathcal{N}$ the least upper bound $\bigsqcup \Delta$ is

$$\langle \bigcap \{F \mid \langle F, D\rangle \in \Delta\}, \bigcap \{D \mid \langle F, D\rangle \in \Delta\}\rangle .$$

CSP is given a denotational semantics over $\mathcal{N}$ as follows.

The atomic processes $STOP$ and $SKIP$ are defined

$$STOP = \langle \{(\langle\rangle, X) \mid X \subseteq \Sigma\}, \emptyset\rangle$$

$$SKIP = \langle \{(\langle\rangle, X) \mid \surd \notin X\} \cup \{(\langle\surd\rangle, X) \mid X \subseteq \Sigma\}, \emptyset\rangle .$$

Let $P = \langle F, D\rangle$, $P' = \langle F', D'\rangle$ and for $b \in B \subseteq \Sigma$, $P_b = \langle F_b, D_b\rangle$ all be

processes. Then

$$
\begin{array}{rcl}
\mathcal{D}[\![a \to P]\!] & = & \{\langle a\rangle s \mid s \in D\} \\
\mathcal{F}[\![a \to P]\!] & = & \{(\langle\rangle, X) \mid a \notin X\} \cup \{(\langle a\rangle s, X) \mid (s, X) \in F\} \\
\mathcal{D}[\![x : B \to P_x]\!] & = & \{\langle b\rangle s \mid b \in B \wedge s \in D_b\} \\
\mathcal{F}[\![x : B \to P_x]\!] & = & \{(\langle\rangle, X) \mid B \cap X = \emptyset\} \cup \{(\langle b\rangle s, X) \mid b \in B \wedge (s, X) \in F_b\} \\
\mathcal{D}[\![P \sqcap P']\!] & = & D \cup D' \\
\mathcal{F}[\![P \sqcap P']\!] & = & F \cup F' \\
\mathcal{D}[\![P \,\Box\, P']\!] & = & D \cup D' \\
\mathcal{F}[\![P \,\Box\, P']\!] & = & \{(\langle\rangle, X) \mid (\langle\rangle, X) \in F \cap F'\} \cup \{(s, X) \mid s \neq \langle\rangle \wedge (s, X) \in F \cup F'\} \\
 & & \cup \{(s, X) \mid s \in \mathcal{D}[\![P \,\Box\, P']\!]\} \\
\mathcal{D}[\![P \;{}_B\|_C\; P']\!] & = & \{st \mid s \in (B \cup C)^* \wedge s\restriction B \in D \wedge s\restriction C \in \mathit{traces}(P')\} \\
 & & \cup \{st \mid s \in (B \cup C)^* \wedge s\restriction B \in \mathit{traces}(P) \wedge s\restriction C \in D'\} \\
\mathcal{F}[\![P \;{}_B\|_C\; P']\!] & = & \{(s, (X \cap B) \cup (Y \cap C) \cup Z) \mid Z \cap (B \cup C) = \emptyset \wedge s \in (B \cup C)^* \\
 & & \wedge (s\restriction B, X) \in F \wedge (s\restriction C, Y) \in F'\} \\
 & & \cup \{(s, X) \mid s \in \mathcal{D}[\![P \;{}_B\|_C\; P']\!]\} \\
\mathcal{D}[\![P \,|||\, P']\!] & = & \bigcup \{\mathit{merge}\langle s, t\rangle \mid s \in D \wedge t \in \mathit{traces}(P')\} \\
 & & \cup \bigcup \{\mathit{merge}\langle s, t\rangle \mid s \in D' \wedge t \in \mathit{traces}(P)\} \\
\mathcal{F}[\![P \,|||\, P']\!] & = & \{(s, X) \mid \exists t, u.\, s \in \mathit{merge}\langle t, u\rangle \wedge (t, X) \in F \wedge (u, X) \in F'\} \\
 & & \cup \{(s, X) \mid s \in \mathcal{D}[\![P \,|||\, P']\!]\} \\
\mathcal{D}[\![P; P']\!] & = & \{st \mid s \in D \wedge s \text{ tick-free}\} \\
 & & \cup \{st \mid s\langle\surd\rangle \in \mathit{traces}(P) \wedge t \in D' \wedge s \text{ tick-free}\} \\
\mathcal{F}[\![P; P']\!] & = & \{(s, X) \mid (s, X \cup \{\surd\}) \in F \wedge s \text{ tick-free}\} \\
 & & \cup \{(st, X) \mid s\langle\surd\rangle \in \mathit{traces}(P) \wedge s \text{ tick-free} \wedge (t, X) \in F'\} \\
 & & \cup \{(s, X) \mid s \in \mathcal{D}[\![P; P']\!]\} \\
\mathcal{D}[\![P\backslash a]\!] & = & \{(s\backslash a)t \mid s \in D\} \cup \{(s\backslash a)t \mid \forall n.s\langle a\rangle^n \in \mathit{traces}(P)\} \\
\mathcal{F}[\![P\backslash a]\!] & = & \{(s\backslash a, X) \mid (s, X \cup \{a\}) \in F\} \cup \{(s, X) \mid s \in \mathcal{D}[\![P\backslash a]\!]\} \\
\mathcal{D}[\![f[P]]\!] & = & \{(f(s))t \mid s \in D\} \\
\mathcal{F}[\![f[P]]\!] & = & \{(f(s), X) \mid (s, f^{-1}(X)) \in F\} \cup \{(s, X) \mid s \in \mathcal{D}[\![f[P]]\!]\} \\
\mathcal{D}[\![f^{-1}[P]]\!] & = & \{s \mid f(s) \in D\} \\
\mathcal{F}[\![f^{-1}[P]]\!] & = & \{(s, X) \mid (f(s), f(X)) \in F\} \cup \{(s, X) \mid s \in \mathcal{D}[\![f^{-1}[P]]\!]\}
\end{array}
$$

Each of these operations is continuous with respect to $\sqsubseteq$, and so we may define the meaning of a recursion (single or mutual) by the least fixed point in the usual way.

**1. The new order.** The failures model includes the assumption that once a process *can* diverge (on trace $s$, say) then we are not interested in what it can do (or fail to do) on any extension of $s$. This assumption, which we made for various technical and philosophical reasons, essentially corresponds to an assumption that a divergent process is *undefined*. The new order (which we will call the *definedness* order) is based solely on this principle. $P$ will be weaker

than $Q$ just when $Q$'s divergences are a subset of $P$'s and all of $P$'s convergent behaviour is copied exactly in $Q$. Included in $P$'s convergent behaviour are its *minimal* divergence traces, because the process has completed these traces before it can diverge. We demand that these be included among $Q$'s traces but not necessarily among its divergences. If $X$ is any set of traces we define $\mu(X)$ to be the minimal elements of $X$: $\{s \in X \mid \nexists t \in X. t < s\}$.

$$\begin{aligned} P \leq Q \;\Leftrightarrow\; & \mathcal{D}[\![Q]\!] \subseteq \mathcal{D}[\![P]\!] \wedge \\ & s \notin \mathcal{D}[\![P]\!] \Rightarrow \mathcal{R}[\![P]\!]s = \mathcal{R}[\![Q]\!]s \wedge \\ & \mu(\mathcal{D}[\![P]\!]) \subseteq \mathit{traces}(Q) \end{aligned}$$

Several points are immediately apparent about this order. First, it is *coarser* than the old one, in that

$$P \leq Q \;\Rightarrow\; P \sqsubseteq Q\,,$$

and *strictly* so, in that there are many pairs of processes $P, Q$ such that

$$P \sqsubseteq Q \quad \text{but not} \quad P \leq Q\,.$$

Secondly, the divergence-free processes (i.e., the ones with $\mathcal{D}[\![P]\!] = \emptyset$) are all incomparable and maximal in the order. (No process is strictly above a non-divergent one.) Thirdly, the new order has the same least element as the old one ( $\perp$, the immediately divergent process).

The new order is complete and, where appropriate, has identical least upper bounds for directed sets as the old one. These and some other useful facts about the new order are established in the next Lemma.

**Lemma 1.1**

a) $P \leq Q \Rightarrow P \sqsubseteq Q$

b) $\perp = \langle \Sigma^* \times \mathcal{P}(\Sigma), \Sigma^* \rangle$ is the least element of $\mathcal{N}$ for both orders. The $\leq$-maximal elements are the divergence-free processes.

c) If $P \leq R$ and $P \sqsubseteq Q \sqsubseteq R$, then $P \leq Q$.

d) Any nonempty subset $S$ of $\mathcal{N}$ has greatest lower bounds with respect to both $\leq$ and $\sqsubseteq$. In general, $\bigsqcap_{\leq} S \sqsubseteq \bigsqcap_{\sqsubseteq} S$.

e) Any subset of $\mathcal{N}$ with any $\leq$-upper bound has a least upper bound.

f) Each $\leq$-directed set has a least upper bound.

g) If $\bigsqcup_{\leq} S$ is defined then so is $\bigsqcup_{\sqsubseteq} S$ and the two are equal. Furthermore $\bigsqcup_{\leq} S = P^* = \langle F^*, D^* \rangle$, where $F^* = \bigcap\{F \mid \langle F, D \rangle \in S\}$ and $D^* = \bigcap\{D \mid \langle F, D \rangle \in S\}$.

**Proof.** (a) and (b) and are trivial. For (c), we observe that $P \leq Q$ if and only if $P \sqsubseteq Q$ and

(i) $(s,X) \in \mathcal{F}[\![P]\!] \wedge s \notin \mathcal{D}[\![P]\!] \Rightarrow (s,X) \in \mathcal{F}[\![Q]\!]$, and

(ii) $\mu(\mathcal{D}[\![P]\!]) \subseteq traces(Q)$,

so to prove the result it will be sufficient to prove (i) and (ii). If $(s,X) \in \mathcal{F}[\![P]\!] \wedge s \notin \mathcal{D}[\![P]\!]$ then, since $P \leq R$, we know $(s,X) \in \mathcal{F}[\![R]\!]$. Hence $(s,X) \in \mathcal{F}[\![Q]\!]$ as $Q \sqsubseteq R$. Exactly the same argument applies for (ii).

The $\sqsubseteq$-greatest upper bound of nonempty $S \subseteq \mathcal{N}$ is always given by $\langle \overline{F^*}, D^* \rangle$, where $F^* = \bigcup\{F \mid \langle F,D\rangle \in S\}$, $D^* = \bigcup\{D \mid \langle F,D\rangle \in S\}$ and where $\overline{F}$ is the closure of $F$ with respect to the compactness axiom (4) (i.e., a pair $(s,X)$ is included if all $(s,Y)$ are for finite subsets $Y$ of $X$). The application of this closure operator is not necessary when $S$ or $\Sigma$ is finite. This bound does not work in general for the definedness order $\leq$, however, since one does not in general have $P \in S \Rightarrow \bigsqcap S \leq P$. The greatest lower bound of $S = \{\langle F_i, D_i\rangle \mid i \in \Lambda\}$ is constructed so that it diverges as soon as the behaviour of any two elements of $S$ starts to differ, either by having a different refusal set or a different next communication. We define $\bigsqcap_{\leq} S$ to be $\langle F, D\rangle$, where

- $D = \bigcup\{D_i \mid i \in \Lambda\} \cup \{st \mid \exists i,j.(\exists Y.(s,Y) \in F_i \setminus F_j) \vee (\exists a.(s\langle a\rangle, \emptyset) F_i \setminus F_j)\}$
- $F = \bigcup\{F_i \mid i \in \Lambda\} \cup \{(s,X) \mid s \in D\}$

It is easy to show that this process is in $\mathcal{N}$ and is indeed the $\leq$ greatest lower bound of $S$. That this greatest lower bound is $\sqsubseteq$-less than the other one follows trivially from the fact that $\leq$ is coarser than $\sqsubseteq$.

(e) follows because, as is fairly well-known, *any* partial order which has greatest lower bounds for nonempty sets has this property. The usual argument is repeated here. If $S$ is a set with an upper bound, then $U_S$, the set of upper bounds of $S$ is nonempty and so $x = \bigsqcap U_S$ exists. Since $y \leq z$ whenever $y \in S$ and $z \in U_S$ it follows that each $y \in S$ is a lower bound for $U_S$. As $x$ is the *greatest* lower bound for $S$ it follows that $x \geq y$ for all $y \in S$ and therefore that $x \in U_S$. Plainly $x$ is the least element of $U_S$ and is therefore the least upper bound of $S$.

We now turn to the proof of (f). Notice that if $\Delta$ is a $\leq$-directed set it is also $\sqsubseteq$-directed. Its $\sqsubseteq$-least upper bound

$$\bigsqcup \Delta = \langle \bigcap\{\mathcal{F}[\![P]\!] \mid P \in \Delta\}, \bigcap\{\mathcal{D}[\![P]\!] \mid P \in \Delta\}\rangle$$

is now shown to be an $\leq$-upper bound. If $P \in \Delta$ then $\mathcal{D}[\![P]\!] \supseteq \mathcal{D}[\![\bigsqcup \Delta]\!]$ by the properties of $\sqsubseteq$. If $s \notin \mathcal{D}[\![P]\!]$ and $Q \in \Delta$, it follows from the existence of $R$ such

that $R \geq P$ and $R \geq Q$ that $s \in \mathcal{D}[\![Q]\!]$ or $\mathcal{R}[\![P]\!]s = \mathcal{R}[\![Q]\!]s$. Hence $\mathcal{R}[\![\bigsqcup \Delta]\!]s = \mathcal{R}[\![P]\!]s$. Finally, if $s$ is minimal in $\mathcal{D}[\![P]\!]$ and $Q \in \Delta$ it again follows from the existence of $R \geq P, Q$ that $s \in traces(Q)$. This shows that $s \in traces(\bigsqcup \Delta)$, which completes the proof that $P \leq \bigsqcup \Delta$. Obviously $\bigsqcup \Delta$ is the $\sqsubseteq$-least $\leq$-upper bound.

It follows by part (e) from the fact that $\Delta$ has one $\leq$-upper bound that it has a least one $\bigsqcup_{\leq} \Delta$. Since $\leq$ is coarser than $\sqsubseteq$ it follows that $\bigsqcup_{\leq} \Delta \sqsubseteq \bigsqcup \Delta$. Combining this with the last observation in the previous paragraph proves that the two are the same, which is what we wanted.

The first sentence of (g) follows easily from the second, which is what we prove. We show first that if $P' = \langle F', D' \rangle$ is the actual least upper bound on $S$ then $D^* = D'$, where $D^*$ is as defined in the statement of the Lemma. Trivially $D' \subseteq D^*$ and, since $D'$ satisfies axiom (5), if $D' \neq D^*$ there is $s \in (\mu D^*) \setminus D'$. As $s \in \mu D^*$ there must be some $P_1 = \langle F_1, D_1 \rangle \in S$ such that $s \in \mu D_1$. Since $D_1 \leq D'$ we therefore know that $s \in traces(P')$. It follows that $P'' = \langle F'', D'' \rangle$ defined

$$\begin{aligned} F'' &= F' \cup \{(st, X) \mid t \in \Sigma^* \wedge X \subseteq \Sigma\} \\ D'' &= D' \cup \{st \mid t \in \Sigma^*\} \end{aligned}$$

is a process. But it easy to show that $P \leq P''$ for all $P \in S$ (for example, by part (c) above) and that $P'' < P'$. It follows that $P'$ cannot be the least upper bound on $S$, a contradiction.

It is easy to show that $P^*$ defined in the statement of the Lemma satisfies axioms (1), (2), (5) and (6). We next note that $F^* \supseteq F'$ since $F'$ is a $\sqsubseteq$-upper bound for $S$. Now by the above paragraph those parts of $F^*$ and $F'$ implied by divergence and axiom (5) are equal. Suppose that $s \notin D' = D^*$. Then there is $P = \langle F, D \rangle \in S$ such that $s \notin D$. Necessarily $\mathcal{R}[\![P]\!]s = \mathcal{R}[\![P']\!]s$ as $P \leq P'$. It follows that $\mathcal{R}[\![P']\!]s \supseteq \mathcal{R}[\![P^*]\!]s$. Putting these facts together yields $F' \supseteq F^*$, proving that in fact $F' = F^*$. Since $D' = D^*$ and $F' = F^*$ we have thus proved that $P^*$ is the actual least upper bound, as claimed. This completes the proof of the Lemma. □

It is easy to construct functions which are monotonic in either of our two orders without being monotonic in the other. However all the usual CSP constructs, with their standard definitions over $\mathcal{N}$, turn out to be monotonic in the new order as well as the old. Of course one can prove each case separately (and easily), but this is unenlightening. The underlying reason why they all work is described in the next paragraph.

With the exception of recursion, a special case which will be dealt with later, every CSP operator over $\mathcal{N}$ is defined by mapping the behaviours (failures and divergences) of the argument process(es) to the behaviours to which they correspond in the image. An examination of these definitions will reveal that, for each operator $F$, all convergent behaviours of $F(P)$ (its behaviours on non-divergent

traces plus the facts that its minimal divergences are traces) are consequences of the convergent behaviours of $P$. Hence, if $P \leq Q$ and $s \notin \mathcal{D}[\![F(P)]\!]$, $\mathcal{R}[\![F(P)]\!]s$ is derived only from the non-divergent behaviours of $P$, each of which is also possible for $Q$ by definition of $\leq$. Thus

$$\mathcal{R}[\![F(P)]\!]s \subseteq \mathcal{R}[\![F(Q)]\!]s\,.$$

The reverse inclusion follows from $\sqsubseteq$-monotonicity, as does

$$\mathcal{D}[\![F(P)]\!] \supseteq \mathcal{D}[\![F(Q)]\!]\,.$$

If $s$ is minimal in $\mathcal{D}[\![F(P)]\!]$ then the fact that it is a trace of $F(P)$ is derived from some convergent behaviour of $P$, necessarily present also in $Q$; hence $s$ is a trace of $F(Q)$, completing the proof that $F(P) \leq F(Q)$.

The most interesting operator to look at in conjunction with the preceding paragraph is sequential composition (;). This is the only operator where a minimal divergence trace of an argument can contribute to a non-divergent trace of the result. This happens in $P;Q$ if $P$ can diverge immediately after terminating ($\surd$).

This non-dependence of the convergent behaviour of $F(P)$ upon the divergent behaviour of $P$ is closely related to a property of the operational semantics. The first-step behaviour of $F(P)$ may depend on the behaviour of $P$ or not (the latter is the case with the prefixing operators, for example). However, if it does and $P$ can perform an internal action to become, say, $P'$, $F(P)$ can perform an internal action and become $F(P')$. Thus, when an operated-on process is active, its internal actions occur independently of $F$: they are outside its control. An immediate consequence is that, if $F$ has brought $P$ to a point where it can diverge but is still interested in what it can do, then $F(P)$ can diverge also.

When a function is $\leq$-monotonic and $\sqsubseteq$-continuous, it is $\leq$-continuous. For if $\mathcal{D}$ is $\leq$-directed, then $\{F(P) \mid P \in \mathcal{D}\}$ is also, with least upper bound $\bigsqcup\{F(P) \mid P \in \mathcal{D}\}$ as described above. However, since $F$ is $\sqsubseteq$-continuous and $\mathcal{D}$ is necessarily $\sqsubseteq$-directed, we have

$$F(\bigsqcup \mathcal{D}) = \bigsqcup\{F(P) \mid P \in \mathcal{D}\}\,.$$

This completes the proof as the least upper bound of $\mathcal{D}$ is the same in both orders.

This means that all CSP constructs other than recursion are continuous in the new order, and so we may deal with recursion using least fixed points in the usual way. As is well known, this makes recursion itself a continuous operation in that if $F(P,Q)$ is continuous, then $\mu P.F(P,Q)$ represents a continuous function of $Q$. Therefore all CSP terms represent continuous functions of their free process variables.

The fact that the two orders yield the same semantics for CSP (i.e., ascribe the same value in $\mathcal{N}$ to each term) follows very easily by structural induction once we observe that the only place where the orders are used is recursion, and the two orders yield identical least upper bounds to all sequences that can appear there. (Of course, the fact that the two orders have the same bottom is also used here.)

**2. Consequences for proof rules.** The existence of the definedness order has a number of striking consequences for proofs by recursion induction and unique fixed point rules. I had previously proved some of the following results from the theorem (of [BRW]) expressing the congruence of the operational and denotational semantics for CSP. These earlier proofs, though interesting, were far more difficult and less natural than the following.

**Theorem 2.1.** Suppose the recursive CSP term $\mu P.F(P)$ is divergence-free (i.e., the least solution to $P = F(P)$ has $\mathcal{D}[\![P]\!] = \emptyset$). Then it is the *only* solution to $P = F(P)$.

**Proof.** We know that any solution $Q$ to this equation must satisfy

$$Q \geq \mu P.F(P)$$

but, being divergence-free, $\mu P.F(P)$ is maximal in $\mathcal{N}$ under $\leq$. Hence $Q = \mu P.F(P)$. □

Of course, this result is equally true of mutual recursions. Indeed, one does not need to know that all the mutually defined processes are divergence-free before one can draw useful conclusions.

**Theorem 2.2.** Suppose $\underline{P} = F(\underline{P})$ is a CSP mutual recursion indexed by some set $\Lambda$, and that in the least solution the $\lambda$-component is divergence-free. Then all solutions have identical $\lambda$-components.

**Proof.** If $\underline{P}_\lambda$ is divergence-free then

$$\underline{P} \leq \underline{P'} \Rightarrow P_\lambda = P'_\lambda$$

where $\leq$ has been extended to the product space in the usual way (i.e., coordinatewise). □

The above results are useful, for they allow one to extend the use of the unique fixed point rule to any recursive definition which is known to have a divergence-free least solution. One application is to the analysis of networks where internal communication is hidden. These can often be proved divergence-free by specific techniques. In certain circumstances one can show that the semantic value of the network is the same as the appropriate component of the solution to the mutual recursion over its state space obtained by applying suitable

"expansion theorems". The above theorems show that this "state-recursion" has only one solution, which is extremely useful when proving equality between the (state-space of) the network and some other system. More details of these applications will be found in [R3].

The rest of the results of this section all have the flavour of the above Theorem: extending to general divergence-free recursions results which were already known for constructive recursions. They are all couched in terms of functions that are $\leq$-monotonic or $\leq$-continuous rather than directly as theorems about CSP, for reasons that will become apparent in the next section.

**Lemma 2.3.** Suppose $F$ is a continuous function from $\mathcal{N}^\Lambda$ to itself, and that in the least solution to $\underline{P} = F(\underline{P})$ the $\lambda$-component $P_\lambda$ is divergence-free. Then for each trace $s$ there exists a natural number $n$ such that *any* $\underline{Q} \in \mathcal{P}^\Lambda$ and $m \geq n$,

$$s \notin \mathcal{D}[\![F^m(\underline{Q})_\lambda]\!] \quad \text{and} \quad \mathcal{R}[\![F^m(\underline{Q})_\lambda]\!]s \;=\; \mathcal{R}[\![P_\lambda]\!]s \;.$$

**Proof.** Suppose $\underline{Q}$ and $s$ are as above. We know that the least solution is $\bigsqcup\{F^n(\underline{\bot}^\Lambda) \mid n \in \mathbb{N}\}$. Thus there is some $n$ such that $s \notin \mathcal{D}[\![F^n(\underline{\bot}^\Lambda)_\lambda]\!]$.

An easy induction shows that $F^k(\underline{\bot}^\Lambda) \leq F^k(\underline{Q})$ for all $k$, and that

$$F^n(\underline{\bot}^\Lambda) \;\leq\; F^m(\underline{Q}) \quad \text{when } n \leq m.$$

The definition of $\leq$ and the fact that $F^n(\underline{\bot}^\Lambda)_\lambda \leq P_\lambda$ then easily gives

$$s \notin \mathcal{D}[\![F^m(\underline{Q})_\lambda]\!] \quad \text{and} \quad \mathcal{R}[\![F^m(\underline{Q})_\lambda]\!]s \;=\; \mathcal{R}[\![F^n(\underline{\bot}^\Lambda)_\lambda]\!]s \;=\; \mathcal{R}[\![P_\lambda]\!]s$$

for $m \geq n$, which establishes the Lemma. □

For a number of years I have been attracted to fixed point induction rules of the following form.

**Proforma rule.** If $R$ is a predicate on (vectors of) processes which is satisfiable (i.e., $R(P)$ holds for some $P$) and satisfies some "continuity" condition, and $\mu P.F(P)$ is a recursive definition, satisfying some "well-definedness" condition, such that

$$\forall P.(R(P) \;\Rightarrow\; R(F(P)))$$

then infer $R(\mu P.F(P))$. □

These have been discussed before in, for example, [R1,BHR,Re,RR1,RR2]. Two slots need to be filled in: the conditions on predicates and recursions. In valid versions of the rule one naturally finds that the stronger one condition is, the weaker the other needs to be.

Up to now, by far the most useful versions of the rule have been based on cases where $F$ is a contraction map and $\{P \mid R(P)\}$ a closed subset in some

complete metric space. This applies, for example, over untimed models like $\mathcal{N}$ when $F$ is constructive, that is

$$\forall P.\forall n.F(P\downarrow n)\downarrow n+1 \;=\; F(P)\downarrow n+1$$

and $R$ satisfies

$$(\alpha) \qquad \neg R(P) \;\Rightarrow\; (\exists n.\forall Q.Q\downarrow n = P\downarrow n \Rightarrow \neg R(Q))$$

where $P\downarrow n$ represents the $n$-step behaviour of $P$. That is, over $\mathcal{N}$,

$$\mathcal{D}[\![P\downarrow n]\!] \;=\; \mathcal{D}[\![P]\!]\cup\{st \mid |s| = n \wedge s \in \mathit{traces}(P)\}$$

$$\mathcal{F}[\![P\downarrow n]\!] \;=\; \mathcal{F}[\![P]\!]\cup\{(s,X) \mid s \in \mathcal{D}[\![P\downarrow\ n]\!]$$

Over product spaces $\mathcal{N}^{\Lambda}$, $P\downarrow n$ is taken co-ordinatewise. See [BHR] for more details. Over the timed models of [RR1,RR2,Re] $F$ is an arbitrary recursion (for there all recursions represent contraction maps) and $R$ represents a closed set in the metric spaces used as models.

The proof of validity in these cases is very simple. We know, by the contraction mapping theorem, that $F$ has a unique fixed point in the space of all processes; but, as $F$ preserves $R$ it is also a contraction mapping on the complete metric space $\{P \mid R(P)\}$. As $F$ has *a* fixed point in the subspace, the *unique* fixed point in the whole space must lie in the subspace.

On the assumption that equality with a given process is to be an acceptable predicate $R$, it is clear that any acceptable "well-definedness" condition on recursions must at least imply uniqueness of fixed points. In the context of the *traces* model for CSP I was able to establish (in [R1]) a version of the rule which relied only upon unique fixed points, and not on anything more concrete, even continuity of $F$. Over any partial order we can define the *interval topology* to be the smallest topology in which all the "closed intervals" $[x,y] = \{z \mid x \leq z \leq y\}$ are closed sets. Now the interval topology over the traces model is compact (a very common property for models of computation – see the appendix) so that the following theorem applies. (For one topology, $\mathcal{T}$, to *extend* another, $\mathcal{U}$, simply means every set open (closed) in $\mathcal{U}$ is open (closed) in $\mathcal{T}$. If $\mathcal{T}$ extends $\mathcal{U}$ one sometimes says that $\mathcal{T}$ is *finer* than $\mathcal{U}$. If, in this case, $\mathcal{T}$ is compact then so must be $\mathcal{U}$: hence the conditions of the following theorem require the compactness of the interval topology fortunately this topology is always compact under the conditions of the theorem, as is proved in the appendix.)

**Theorem 2.4 [R1].** Suppose $\langle X,\leq\rangle$ is a complete lattice and that $\mathcal{T}$ is a compact topology on $X$ which extends the interval topology. Suppose also that the monotonic function $f : X \to X$ has a unique fixed point $x$. Then if $Y \subseteq X$ is nonempty, closed in $\mathcal{T}$ and such that $f(Y) \subseteq Y$, we can deduce $x \in Y$.

**Proof.** A straightforward transfinite induction establishes that each of the sets $[f^{\alpha}(\bot), f^{\alpha}(\top)] \cap Y$ (all of which are closed) are all non-empty, where $\bot$ and

$\top$ are respectively the least and greatest elements of $X$. (The limit ordinal case uses the compactness of $\mathcal{T}$.) But the fact that $f$ has a unique fixed point implies that $[f^\alpha(\bot), f^\alpha(\top)]$ is eventually (i.e., for large enough $\alpha$) the singleton set containing the fixed point. The result follows immediately. □

This proof depends crucially on the existence of a top element. Indeed the theorem is not true of more general partial orders. One need only consider the flat truth value domain $\{true, false, \bot\}$ and the continuous function $\neg$. This has unique fixed point $\bot$ and maps the closed set $\{true, false\}$ into itself.

Having observed this it once seemed to me to be unlikely that one could get an analogue of Theorem 2.4 for the failures model. However, the new order will allow us to prove two useful partial analogues, at least. They both have as preconditions the fact that the unique fixed point is divergence-free, which is unlikely to be much of a practical handicap since almost all processes one will tolerate in practice will satisfy this constraint.

First we define three more conditions on predicates.

($\beta$) If $\neg R(P)$ then there is a finite set $T$ of traces such that, for any $Q$, if

$$s \in \mathcal{D}[\![P]\!] \Leftrightarrow s \in \mathcal{D}[\![Q]\!] \quad \text{and} \quad \mathcal{R}[\![P]\!]s = \mathcal{R}[\![Q]\!]s$$

for all $s \in T$, then $\neg R(Q)$.

($\gamma$) If $\neg R(P)$ then there are finite sets $T$ of traces and $F$ of failures such that, for any $Q$, if

$$s \in \mathcal{D}[\![P]\!] \Leftrightarrow s \in \mathcal{D}[\![Q]\!]$$

for all $s \in T$ and

$$(s, X) \in \mathcal{F}[\![P]\!] \Leftrightarrow (s, X) \in \mathcal{F}[\![Q]\!]$$

for all $(s, X) \in F$ then $\neg R(Q)$.

($\delta$) If $\neg R(P)$ then there are finite sets $T$ of traces and $F$ of *finite* failures (failures $(s, X)$ with $X$ finite) such that, for any $Q$, if

$$s \in \mathcal{D}[\![P]\!] \Leftrightarrow s \in \mathcal{D}[\![Q]\!]$$

for all $s \in T$ and

$$(s, X) \in \mathcal{F}[\![P]\!] \Leftrightarrow (s, X) \in \mathcal{F}[\![Q]\!]$$

for all $(s, X) \in F$ then $\neg R(Q)$.

These conditions simply say that any failure to satisfy the predicate is detectable from a process' behaviour on a finite set of traces, or on a finite set of possible divergences and failures, or on a finite set of possible divergences and finite failures. ($\alpha$) says that failure is detectable by some finite length of trace. (In each case the sets or length can depend on the failing process.)

The definitions of the conditions $\beta$, $\gamma$, $\delta$ can all be extended to predicates of product spaces $\mathcal{N}^\Lambda$. One simply insists that failure is detectable from a finite set (or sets) as before from a finite selection of components of the vector of processes. This is different to the extension of $\alpha$ to product spaces, where failure need only be detectable from *all* components up to some fixed length.

($\beta$) If $\neg R(\underline{P})$ then there are finite sets $T$ of traces and $\Phi$ of indices such that, for any $\underline{Q}$, if

$$s \in \mathcal{D}[\![P_\lambda]\!] \Leftrightarrow s \in \mathcal{D}[\![Q_\lambda]\!] \text{ and } \mathcal{R}[\![P_\lambda]\!]s = \mathcal{R}[\![Q_\lambda]\!]s$$

for all $s \in T$ and $\lambda \in \Phi$, then $\neg R(\underline{Q})$.

($\gamma$) If $\neg R(\underline{P})$ then there are finite sets $\Phi$ of indices, $T$ of traces and $F$ of failures such that, for any $\underline{Q}$, if whenever $\lambda \in \Phi$

$$s \in \mathcal{D}[\![P_\lambda]\!] \Leftrightarrow s \in \mathcal{D}[\![Q_\lambda]\!]$$

for all $s \in T$ and

$$(s, X) \in \mathcal{F}[\![P_\lambda]\!] \Leftrightarrow (s, X) \in \mathcal{F}[\![Q_\lambda]\!]$$

for all $(s, X) \in F$ then $\neg R(\underline{Q})$.

($\delta$) If $\neg R(\underline{P})$ then there are finite sets $\Phi$ of indices, $T$ of traces and $F$ of *finite* failures (failures $(s, X)$ with $X$ finite) such that, for any $\underline{Q}$, if whenever $\lambda \in \Phi$

$$s \in \mathcal{D}[\![P_\lambda]\!] \Leftrightarrow s \in \mathcal{D}[\![Q_\lambda]\!]$$

for all $s \in T$ and

$$(s, X) \in \mathcal{F}[\![P_\lambda]\!] \Leftrightarrow (s, X) \in \mathcal{F}[\![Q_\lambda]\!]$$

for all $(s, X) \in F$ then $\neg R(\underline{Q})$.

The conditions $\alpha$, $\beta$, $\gamma$, $\delta$ are clearly successively stronger: for example every $\beta$-predicate is an $\alpha$-predicate. Each of them generates a topology on $\mathcal{N}$ or $\mathcal{N}^\Lambda$ where a set is closed precisely when it has the form $\{P \mid R(P)\}$ for an allowable predicate $R$. Several later results depend on facts about these topologies, but

rather than analyse them in detail here we will state results as required and relegate proofs to the appendix. The following proposition lists some of these.

**Proposition 2.5.**

a) The conditions $\beta$, $\gamma$ and $\delta$ are equivalent if the underlying alphabet $\Sigma$ is finite. If it is infinite $\delta$ is strictly stronger than $\gamma$ and $\gamma$ is strictly stronger than $\beta$. $\alpha$ is equivalent to $\beta$ if the alphabet is finite and the index-set $\Lambda$ is finite. Otherwise $\beta$ is strictly stronger than $\alpha$.

b) For all $\xi \in \{\beta, \gamma, \delta\}$ the $\xi$-topology on $\mathcal{N}^\Lambda$ defined above is well-defined. The topology generated by $\xi$ on $\mathcal{N}^\Lambda$ is the same as the product topology where each copy of $\mathcal{N}$ is given the topology generated by $\xi$.

c) The $\delta$-topology on $\mathcal{N}^\Lambda$ is compact, Hausdorff and zero-dimensional.

d) If $\underline{P} \in \mathcal{N}^\Lambda$, each of the sets $\{\underline{Q} \mid \underline{Q} \geq \underline{P}\}$, $\{\underline{Q} \mid \underline{Q} \leq \underline{P}\}$, $\{\underline{Q} \mid \underline{Q} \sqsupseteq \underline{P}\}$ and $\{\underline{Q} \mid \underline{Q} \sqsubseteq \underline{P}\}$ are closed in the $\delta$-topology (and hence in the others as well). □

Having made all these definitions and stated the above proposition we are now in a position to fulfil the promise made earlier and prove some analogues of Theorem 2.4. The *lower topology* on a partial order is defined to be the smallest topology on which all sets of the form $x\uparrow = \{y \mid x \leq y\}$ are closed (see the appendix for more details).

**Theorem 2.6.** Let $X$ be a complete partial order and let $\mathcal{T}$ be a compact extension of its lower topology. Then, if $f : X \to X$ is a monotonic function whose least fixed point $x$ is maximal (and hence unique) and $Y$, closed in $\mathcal{T}$, is such that $f(Y) \subseteq Y$, we have $x \in Y$.

**Proof.** This is very similar to that of Theorem 2.4. Transfinite induction establishes that the (closed) sets $(f^\alpha(\bot))\uparrow \cap Y$ are all non-empty, where $\bot$ is the least element of $X$. But the fact that the least fixed point of $f$ is maximal implies that $(f^\alpha(\bot))\uparrow$ eventually (i.e., for large enough $\alpha$) contains only the fixed point. The result follows immediately. □

**Corollary.** If $F : \mathcal{N}^\Lambda \to \mathcal{N}^\Lambda$ is monotonic with a divergence-free least fixed point $\underline{P}$ and $R$ is a satisfiable $\delta$-predicate such that $\forall \underline{Q}.R(\underline{Q}) \Rightarrow R(F(\underline{Q}))$ then $R(\underline{P})$.

**Proof.** This follows trivially from Theorem 2.6 and Proposition 2.5 (c) and (d). □

This corollary shows that the proforma rule is valid in any case where the least fixed point can be shown to be divergence free and the predicate satisfies $\delta$. This and the other version given below show that we can do without constructiveness for many applications.

When the function $F$ is continuous one can, because of Lemma 2.3, weaken $\delta$ to $\beta$.

**Theorem 2.7.** If $F : \mathcal{N}^\Lambda \to \mathcal{N}^\Lambda$ is continuous with a divergence-free least fixed point $\underline{P}$ and $R$ is a satisfiable $\beta$-predicate such that $\forall \underline{Q}.(R(\underline{Q}) \Rightarrow R(F(\underline{Q})))$ then $R(\underline{P})$.

**Proof.** If $R(\underline{P})$ did not hold then we could find a finite sets $\Phi$ of indices and $T$ of traces such that for any $\underline{Q}$, if for all $\langle \lambda, s \rangle \in \Phi \times T$ we had $\mathcal{R}[\![P_\lambda]\!]s = \mathcal{R}[\![Q_\lambda]\!]s$ and $s \notin \mathcal{D}[\![Q_\lambda]\!]$, then $R(\underline{Q})$ would not hold. And we know that there exists $\underline{Q} \in \mathcal{N}^\Lambda$ such that $R(\underline{Q})$, and hence $R(F^n(\underline{Q}))$ for all $n$. However, choosing $n$ to be greater than all the $n$s chosen by Lemma 2.3 for the $\langle \lambda, s \rangle \in \Phi \times T$, this gives a contradiction. □

There is a striking similarity between the proof of Theorem 2.7 and the corresponding result for metric spaces and contraction mappings discussed above. Both rely on the fact that iterations begun from an arbitrary point in the space (the process known to satisfy the predicate) converge to the fixed point. For a consequence of Lemma 2.3 is that, if a continuous function $F$ has a divergence-free least fixed point, then every sequence of the form $\langle F^n(Q) \mid n \in \mathbb{N} \rangle$ converges to the fixed point in the $\beta$-topology.

It is mathematically pleasing to have characterisations of when the recursion induction rule is valid that are independent of a specific metric. Even though the great majority of recursions included in CSP programs are constructive, it proves useful to have these abstract forms of the rule when the function is derived from the process and not the other way round: this will be covered in [R3].

**3. Unbounded nondeterminism; hiding infinite sets.** As stated in the introduction, the completeness of the determinism order $\sqsubseteq$ depends on the compactness axiom (4). As an illustration of this, consider the "processes"

$$P_n = \prod_{n \le m} m \to STOP$$

where $\prod$ represents the nondeterministic choice over a possibly infinite set of indexed processes. The natural interpretation of $P_n$ as a set of failures (assuming its alphabet is $\mathbb{N}$) is

$$\{(\langle\rangle, X) \mid \{n, n+1, n+2, \ldots\} \not\subseteq X\} \cup \{(\langle m \rangle, X) \mid m \ge n\}$$

which does not satisfy the compactness axiom. If these processes were allowed we would, as one would expect from their definitions, have $P_n \sqsubseteq P_{n+1}$. However the chain $\langle P_n \mid n \in \mathbb{N} \rangle$ would have no upper bound, for if $P \sqsupseteq P_n$ for all $n$ then

$P$ can neither communicate on its first step nor refuse the whole alphabet – a clear contradiction to axiom (3).

The compactness axiom is not as restrictive as one of finite nondeterminism. The latter would state that, for each $s$, $\mathcal{R}[\![P]\!]s$ contains a finite number of maximal elements, with every $Y \in \mathcal{R}[\![P]\!]s$ contained in one of them. Axiom (4) allows infinite nondeterminism provided it is finitely presented in some sense, for example it allows sets of refusals such as

$$\{X \subseteq \mathbf{N} \mid \forall i.\{2i, 2i+1\} \not\subseteq X\}$$

where there are uncountably many maximal elements. However, as we see from the processes $P_i$ above it prevents us from expressing many natural processes properly.

It is clear that if one wants to model unbounded nondeterminism properly, then we are far better off without this compactness axiom. Fortunately the definedness order $\leq$ does not need it for completeness, and so it may be discarded. (Note that the $P_i$ as defined above are not comparable under $\leq$.) I should remark at this point that an alternative, slightly simpler, form of axiom (3) which is sometimes seen

$$(3') \qquad (s, X) \in \mathcal{F}[\![P]\!] \wedge (s\langle a\rangle, \emptyset) \notin \mathcal{F}[\![P]\!] \Rightarrow (s, X \cup \{a\}) \in \mathcal{F}[\![P]\!]$$

requires axiom (4) in the sense that $(3')+(4) \equiv (3)+(4)$ but $(3')$ alone is strictly weaker than (3). $(3')$ allows such "processes" as $P = \langle\{(<>, X) \mid X \subseteq \mathbf{N} \wedge X \text{ is finite}\}, \emptyset\rangle$ (which cannot refuse the whole set of integers or communicate any). This $P$ is not acceptable as $P\backslash\mathbf{N}$ would not have any traces at all. (See the definition of the infinite hiding operator below.) Thus $(3')$ is not adequate if axiom (4) is omitted.

We will denote by $\mathcal{N}'_\Sigma$ the failures model defined without axiom (4) over some alphabet $\Sigma$. (As before, we will usually suppress the subscript.) Of course $\mathcal{N}'$ only differs from $\mathcal{N}$ when $\Sigma$ is infinite. The orders $\sqsubseteq$ and $\leq$ are defined exactly as before. $\leq$ is complete, but if $\Sigma$ is infinite $\sqsubseteq$ is not (as is illustrated by the example above).

All parts of Lemma 1.1 continue to hold in $\mathcal{N}'$ except that it is now unnecessary (and wrong) to apply the closure operator $\overline{F}$ when finding the $\sqsubseteq$-greatest lower bound of a nonempty set. It is interesting to note that, over $\mathcal{N}'$, $\sqsubseteq$-greatest lower bounds are found by union and $\leq$-least upper bounds (where they exist) are found by intersection, but that neither of the opposite pair of statements hold in general.

All the operators that were used over $\mathcal{N}$ (and defined earlier) may be defined over $\mathcal{N}'$ using exactly the same clauses. In proving the operators well-defined over $\mathcal{N}$, axiom (4) is never used except when proving axiom (4). It follows that

they are all well-defined over $\mathcal{N}'$. We now prove that they are all monotonic and continuous.

If $P \in \mathcal{N}'$, define $P^{\#}$ as follows:

$$\mathcal{D}[\![P^{\#}]\!] = \mathcal{D}[\![P]\!]$$

$$\mathcal{F}[\![P^{\#}]\!] = traces(P) \times \mathcal{P}(\Sigma).$$

This is a process with the same traces as $P$ but which can refuse anything at any time. Clearly $P^{\#} \in \mathcal{N}$ and the function thus represented from $(\mathcal{N}', \leq)$ to $(\mathcal{N}, \sqsubseteq)$ is continuous.

The fact that all operators other than recursion are $\sqsubseteq$-monotonic follows because in all cases the behaviour of the result of an operator depends in a direct and positive way on the behaviour of its arguments. The proof of $\leq$-monotonicity is completed by exactly the same argument we used earlier over $\mathcal{N}$.

The operators' continuity can be proved by using the # operator. If $F$ is one of the operations and $\Delta \subseteq \mathcal{N}'$ is directed, we know by a standard argument using monotonicity that

$$\bigsqcup\{F(P) \mid P \in \Delta\} \leq F(\bigsqcup \Delta).$$

We also have, since # is continuous from $\mathcal{N}'$ to $\mathcal{N}$ and $F$ is continuous over $\mathcal{N}$:

$$\bigsqcup\{F(P^{\#}) \mid P \in \Delta\} = F((\bigsqcup \Delta)^{\#}).$$

It follows (using the same observation as in the last paragraph) that

$$\mathcal{D}[\![\bigsqcup\{F(P) \mid P \in \Delta\}]\!] = \mathcal{D}[\![F(\bigsqcup \Delta)]\!].$$

But if $P \leq Q$ and $\mathcal{D}[\![P]\!] = \mathcal{D}[\![Q]\!]$ it is easy to see that $P = Q$; continuity follows immediately.

I think it is virtually certain that all the usual algebraic laws of CSP (from [BR,H], for example) remain true over the extended model $\mathcal{N}'$, but have not yet systematically checked this. However we will see below that several natural extensions of these do not extend to the infinite hiding operator defined below.

As we know, all of the operators dealt with above map $\mathcal{N}$ to itself, so there is little reason to move to the extended model if they are all one wants to use. However, as hinted earlier, there are interesting operators which we can define over $\mathcal{N}'$ that could not be defined before. Firstly we can now define the nondeterministic composition of an arbitrary nonempty set of processes by union. If $\mathcal{S}$ is any nonempty set of processes

$$\mathcal{D}[\![\sqcap \mathcal{S}]\!] = \bigcup\{\mathcal{D}[\![P]\!] \mid P \in \mathcal{S}\}$$

$$\mathcal{F}[\![\sqcap S]\!] = \bigcup\{\mathcal{F}[\![P]\!] \mid P \in S\}.$$

This is continuous in each process argument individually, but not over infinite vectors of processes. (In considering the monotonicity and continuity of $\sqcap$ it is easier to think of it as a function of vectors rather than sets of processes, for the latter have no obvious partial order.) For example, if we define vectors $\underline{P}^n$ by

$$P^n_m = \begin{cases} \bot & \text{if } n \leq m \\ STOP & \text{otherwise} \end{cases}$$

we have $\underline{P}^n \leq \underline{P}^{n+1}$ but $\sqcap \underline{P}^n = \bot$ for all $n$ while $\sqcap(\bigsqcup\{\underline{P}^n \mid n \in \mathbf{N}\}) = \sqcap(STOP^{\mathbf{N}}) = STOP$.

The existence of the above operator implies that any satisfiable specification defined purely as a property of the traces/failures/divergences of a process (every individual behaviour satisfies some property) has a most nondeterministic solution: simply take the nondeterministic composition of all solutions. (Some properties of this form, like "deadlock-free", do not have most nondeterministic solutions over $\mathcal{N}$.)

Perhaps the most annoying restriction imposed by the original model on the language was the impossibility of hiding an infinite set of events. If we allow infinite alphabets at all, it is strange only to be able to hide finite sets. There is no reason not to define it over $\mathcal{N}'$, however.

$$\begin{aligned} \mathcal{D}[\![P\backslash X]\!] &= \{(s\backslash X)t \mid s \in \mathcal{D}[\![P]\!]\} \\ &\quad \cup\{(s\backslash X)t \mid \exists s_1 < s_2 < \ldots .s = s_1 \wedge \forall i.(s_i\backslash X = s\backslash X \wedge s_i \in traces(P))\} \\ \mathcal{F}[\![P\backslash X]\!] &= \{(s\backslash X, Y) \mid (s, X \cup Y) \in \mathcal{F}[\![P]\!]\} \cup \{(s,Y) \mid s \in \mathcal{D}[\![P\backslash a]\!]\} \end{aligned}$$

Notice that in this definition we insist that the sequence of traces yielding divergence are linearly ordered. This is often not done when $X$ is finite because then König's Lemma proves that, if there are infinitely many $s_i$ with $s_i\backslash X \leq s$ for all $i$ then there is an infinite ordered sequence of the type above.

It is straightforward to show that this operator is well-defined and monotone with respect to $\leq$ and $\sqsubseteq$. However it is not continuous, again because of the introduction of unbounded nondeterminism. Consider the sequence $(P_n \mid n \in \mathbf{N})$, where

$$P_n = (x : \{k \mid k \leq n\} \rightarrow STOP) \Box (x : \{k \mid k > n\} \rightarrow \bot).$$

These form a $\leq$-chain with limit

$$P = x : \mathbf{N} \rightarrow STOP.$$

$P\backslash\mathbf{N} = STOP$ but, for each $n$, $P_n\backslash\mathbf{N} = \bot$.

Thus we have succeeded in modelling unboundedly nondeterministic operators, but with the seemingly inevitable loss of continuity. This means that in

general the least fixed point might not be reached after only $\omega$ iterations of the underlying function. Indeed, one can give a simple if contrived example to show that any ordinal can be required. Given an ordinal $\alpha$ we take it (i.e., $\{\beta | \beta < \alpha\}$) as our alphabet and define

$$P = \beta : \alpha \rightarrow (P \parallel \gamma : \beta \rightarrow STOP) \backslash \alpha .$$

A little thought will reveal that the least fixed point here is

$$\beta : \alpha \rightarrow STOP$$

since every execution generates a strictly decreasing sequence of ordinals, which cannot be infinite. It takes precisely $\alpha$ iterations to reach this fixed point, unless $\alpha$ is finite in which case it takes $\alpha + 1$.

Unbounded nondeterminism seems to appear in two distinct forms: it manifests itself in a finite time, as in

$$Q_1 = \Pi\{n \rightarrow STOP \mid n \in \mathbf{N}\}$$

where the initial refusals of $\mathcal{N}'$ capture it, or takes an infinite amount of time to appear as in

$$Q_2 = \Pi\{P_n \mid n \in \mathbf{N}\} \quad \textit{where} \quad P_0 = STOP,\ P_{n+1} = a \rightarrow P_n .$$

$Q_2$ appears to be able to perform any finite number of $a$s, but not to be able to perform an infinite sequence of them. Thus, intuitively, we would not expect $Q_2 \backslash a$ to diverge. (Note that this form of unbounded nondeterminism can appear where there is a finite alphabet.) Unfortunately, in the $\mathcal{N}'$ semantics, $Q_2 \backslash a$ *can* diverge because it interpolates a process' infinite behaviours from increasing sequences of its finite behaviours. $\mathcal{N}'$ has no means of describing this second, and perhaps more subtle, form of unbounded nondeterminism. (This example illustrates the fact that, in $\mathcal{N}'$, hiding (even finite) is not infinitely distributive over nondeterministic choice.)

The law $P \backslash X \backslash Y = P \backslash Y \backslash X$ fails to hold in general for essentially the same reason. To see this consider the process

$$Q = x : \mathbf{N} \rightarrow P_n$$

where $P_n$ are as defined above. If $a$ is hidden before $\mathbf{N}$ there is no divergence as there is no infinite execution sequence of $a$s following any natural number: $Q \backslash a \backslash \mathbf{N} = STOP$. However if $\mathbf{N}$ is hidden first this information is lost and so divergence is predicted: $Q \backslash \mathbf{N} \backslash a = \bot$.

A way around this problem is to introduce a new component into the model to represent the infinite traces of a process. Thus a process will be described as a triple $(F, D, I)$, where $F$ and $D$ have the same structure as before and $I$ is the

set of infinite traces it can perform. See the sequel [R2] for the construction of this model. In the sequel we will see that the present model and semantics in general give a pessimistic view of CSP, in that if $P$ is any CSP term $P_1$ is the value ascribed to it by the semantics above and $P_2$ is the operationally natural value then $P_1 \leq P_2$. Equality is achieved once one introduces the infinite traces component. We also recover the desirable algebraic properties that were shown to have been lost above.

Thus arguably the model $\mathcal{N}'$ is not really a "proper" model for unboundedly nondeterministic CSP, rather an approximation to the correct one. Nevertheless the fact that it exists and is a complete partial order is certainly interesting. (In the sequel we will see that when it is extended by infinite traces it is not only an incomplete partial order but has no complete partial order consistent with CSP. The existence of the fixed points of recursions becomes much harder to prove.)

**Proof rules over $\mathcal{N}'$.** Perhaps paradoxically we find that the model $\mathcal{N}'$ which was made possible by $\leq$ has slightly less of the attractive properties proved in Section 2 (using $\leq$) than does $\mathcal{N}$.

Theorems 2.1 and 2.2 and Lemma 2.3 hold over $\mathcal{N}'$ for exactly the same reasons as before. One can still define the properties $\alpha$, $\beta$, $\gamma$ and $\delta$ exactly as was done there. Parts (a) and (h) of Proposition 2.5 still hold. The $\delta$-topology on $\mathcal{N}'$ is still compact, but since every open (closed) set clearly contains $\langle \overline{F}, D \rangle$ if and only if it contains $\langle F, D \rangle$ for each $\langle F, D \rangle \in \mathcal{N}'$, it fails even to be $T_0$. (Recall that $\overline{F}$ denotes the closure of $F$ under the compactness axiom.) For this reason it fails to be an extension of either lower topology. The $\gamma$-topology is Hausdorff and satisfies all parts of Proposition 2.5 (d). Unfortunately it is not compact, unless of course $\Sigma$ is finite.

All this means that none of $\alpha - \delta$ satisfies the conditions of Theorem 2.6. The lower topology is itself compact, as it always is over a consistently complete cpo (see the appendix), so at least Theorem 6 is not vacuous. Whether there are any compact extensions of it that are practically useful will be a subject of further research.

Since Lemma 2.3 still holds in the extended model it follows that Theorem 2.7 does as well. However, unlike over $\mathcal{N}$, care is necessary in applying this theorem over $\mathcal{N}'$ since now by no means all functions are continuous. In particular it will be unsafe to use it whenever the function in question involves infinite nondeterminism or infinite hiding.

**4. Conclusions.** We have seen that it is possible for two different orders to give exactly the same semantics to CSP, at first sight a rather surprising result. Since I discovered the new order several of my colleagues have pointed out to me that a very similar alternative order based on definedness has recently been discovered by Greg Nelson for Dijkstra's language of guarded commands [N]. As

in this paper he replaced the usual refinement order by a new one which treats all nondivergent behaviour as incomparable: two processes $P, Q$ are ordered just when $P$ can diverge whenever $Q$ can and all of $P$'s nondivergent behaviour is reflected exactly in $Q$:

$$(\sigma, \bot) \notin P \Rightarrow P(\sigma) = Q(\sigma)\,.$$

Just as was the case in CSP, the new order there provided exactly the same semantics as before but increases the range of constructs that can be considered.

In both of these languages the nondeterminism order must remain the order of refinement and will therefore continue to play a most important role. However just because it has that function does not means that it is the best order on which to base the semantic fixed point theory: we have just seen how by moving to $\leq$ we were able to prove a wide range of new results about fixed points and to extend the model and language.

As I remarked above, the extension $\mathcal{N}'$ of the failures model is only able to cope properly with certain sorts of unbounded nondeterminism. It is perhaps best viewed as a stepping stone to the model incorporating infinite traces that will be introduced in [R2], though where it is adequate it is certainly simpler and easier to analyse than the new one.

Now that we have two different orders on the failures model we are faced with the question of which should be presented as the "standard" order for recursion to those meeting the model for the first time. If we agree that the restriction to finitely nondeterministic operators is undesirable there seem to be two options. One could present this new order (over $\mathcal{N}'$), mentioning the important role of $\sqsubseteq$ in refinement. On the other hand one could continue to use $\sqsubseteq$ over $\mathcal{N}'$ and appeal to the existence of $\leq$ and the results of this paper to assert the existence of fixed points, even though the order wonld not be complete. This is an important question but at this stage it is too early to decide.

**Appendix: the topology of $\mathcal{N}$ and $\mathcal{N}'$.** In this section we justify some of the claims that were made about the topological properties of the various conditions on predicates and about the lower and interval topologies. Of necessity we assume that the reader has a basic knowledge of general topology: essentially up to product spaces and Tychonoff's theorem.

First we establish the various parts of 2.5, with a few observations ahout $\mathcal{N}'$. The fact that $\beta$, $\gamma$ and $\delta$ are equivalent when $\Sigma$ is finite is obvious because there are only finitely many refusal sets and all refusal sets are finite. The case when $\Sigma$ is infinite is illustrated by setting $\Sigma = \mathbb{N}$. Then

$$R_\beta(P) \equiv \exists n.(<>, \{n\}) \in \mathcal{F}[\![P]\!]$$

is a $\beta$-predicate but not a $\gamma$-predicate, while

$$R_\gamma(P) \equiv \forall s.(s, \mathbb{N}) \notin \mathcal{F}[\![P]\!]$$

is a $\gamma$-predicate but does not satisfy $\delta$. ($R_\gamma$ is deadlock-freedom, an important predicate: $\delta$ is the only one of our conditions that does not allow ***all*** predicates that are formed by specifying that all behaviours must satisfy some condition on divergences and failures.)

The fact that $\alpha$ is equivalent to $\beta$ (and hence to $\gamma$ and $\delta$) if the alphabet and $\Lambda$ are both finite follows easily from the fact that there are then only finitely many traces of any length and $\Phi$ (in the definition of $\beta$) can be equal to $\Lambda$. If $\Lambda$ is infinite the predicate

$$R^1_\alpha(\underline{P}) \equiv \exists \lambda . P_\lambda = STOP$$

satisfies $\alpha$ but not $\beta$. The following predicate of a single process also has these properties

$$R^2_\alpha(P) \equiv P \neq STOP$$

when the alphabet $\Sigma$ is infinite.

The fact that $\beta$, $\gamma$ and $\delta$, as defined in section 2 (over either model and over a simple or product space), generate topologies is easily demonstrated. In each case it is trivial that the whole space and the empty set are closed. That an arbitrary intersection of closed sets $\bigcap\{C_\lambda \mid \lambda \in I\}$ is closed follows because if some $P$ fails to be in here then $P \notin C_\lambda$, say. This last fact is witnessed by some finite set(s) (of traces, or traces and failures, etc.). Exactly the same sets witness that $P$ cannot be in the intersection. If $C$ and $D$ are closed then so is $C \cup D$, for if $P \notin C \cup D$ then there are finite sets for each of $C$ and $D$ witnessing this. The union(s) of these witness that $P$ cannot be in $C$ or in $D$, and therefore not in $C \cup D$.

To prove that the topologies produced on $\mathcal{N}^\Lambda$ (and, indeed on $\mathcal{N}'^\Lambda$) are just the products of the topologies on the individual spaces it is helpful to consider bases for the *open* sets of the topologies. We deal here with $\delta$, the others ($\beta$ and $\gamma$) being very similar indeed. An examination of $\delta$ (defined respectively over a single and a product space) reveals that a set $U$ is open (the complement or negation of an open set) if and only for all $P \in U$ there the $\langle T, F\rangle$-ball (respectively $\langle T, F, \Phi\rangle$-ball) about $P$ is contained in $U$ for some finite sets $T, F(, \Phi)$. The $\langle T, F\rangle$-ball is defined to be those processes which cannot be distinguished from $P$ by their divergences in $T$ and their finite failures in $F$. The $\langle T, F, \Phi\rangle$-ball about $\underline{P} \in \mathcal{N}^\Lambda$ is similarly defined to be all those vectors which cannot be distinguished from $\underline{P}$ by inspecting the divergences in $T$ and finite failures in $F$ of their $\lambda$-components for $\lambda \in \Phi$. Denote these balls by

$$N_{\langle T,F\rangle}(P) \quad \text{and} \quad N_{\langle T,F,\Phi\rangle}(\underline{P})$$

respectively. It is easy to see that if $Q \in N_{\langle T,F\rangle}(P)$ then

$$N_{\langle T,F\rangle}(Q) = N_{\langle T,F\rangle}(P)$$

and that the same holds over $\mathcal{N}^\Lambda$. It follows that the balls are in each case a basis for the respective topology.

A basis for the product topology on $\mathcal{N}^\Lambda$ generated by $\delta$ on each component is formed by taking a finite set $S$ of pairs $(\lambda, U)$ where $U$ is a basis element on $\mathcal{N}$ (and, without loss of generality, $(\lambda,U),(\lambda,U') \in S \Rightarrow U = U'$) and letting $U_S = \{\underline{P} \mid (\lambda,U) \in S \Rightarrow P_\lambda \in U\}$. Given $P \in U_S$, we know that for each $(\lambda,U) \in S$ there are $T_\lambda$ and $F_\lambda$ such that $N_{(T_\lambda,F_\lambda)}(P_\lambda) \subseteq U)$. Define

$$\begin{aligned} \Phi &= \{\lambda \mid (\lambda,U) \in S\} \\ T &= \bigcup\{T_\lambda \mid (\lambda,U) \in S\} \\ F &= \bigcup\{F_\lambda \mid (\lambda,U) \in S\}. \end{aligned}$$

It is easy to see that $N_{(T,F,\Phi)}(\underline{P}) \subseteq U_S$. It follows that the $\delta$-topology on $\mathcal{N}^\Lambda$ is finer (an extension of) the product topology. The other way round is easier: given $N_{(T,F,\Phi)}(\underline{P})$, set $S = \{(\lambda, N_{(T,F)}(P_\lambda)) \mid \lambda \in \Phi\}$. It is easy to see that $U_S = N_{(T,F,\Phi)}$.

The compactness axiom (4) in the definition of $\mathcal{N}$ means that each element is completely determined by its divergences and finite failures: there is a unique element of $\mathcal{N}$ corresponding to each pair $\langle ff, D\rangle$ of finite failures and divergences satisfying axioms (1-3),(5),(6). This means that the natural projection from $\mathcal{N}$ to the set $\mathcal{N}_{fin}$ of all such pairs is a bijection. If the $\delta$-topology is defined on $\mathcal{N}_{fin}$ in the same way as over $\mathcal{N}$ it is clear that this map in a homeomorphism. Thus if we could prove that $\mathcal{N}_{fin}$ is compact under $\delta$ we would also have proved that $\mathcal{N}$ is.

It is clear that the $\delta$-toplogy on $\mathcal{N}_{fin}$ is just the subspace topology inherited from the topology equivalently defined on the set $A$ of arbitrary pairs $\langle ff, D\rangle$ of finite failure and divergence sets (i.e., not necessarily satisfying the axioms). It is easy to prove that the latter is just the product of the two topologies on sets of finite failures and traces respectively where bases are given by

$$\begin{aligned} N_X(ff) &= \{ff' \mid ff' \cap X = ff \cap X\} \qquad X \text{ a finite set of finite failures} \\ N_X(D) &= \{D' \mid D' \cap X = D \cap X\} \qquad X \text{ a finite set of traces} \end{aligned}$$

which are in turn homeomorphic to the product of $\Sigma^* \times \wp(\Sigma)$ and $\Sigma^*$ copies respectively of the discrete topology on $\{0,1\}$ – compact by Tychonoff's theorem. It follows that $\mathcal{N}$ is compact if and only if it is a closed subset of $A$. Since the intersection of closed subsets is closed it will be sufficient to check that the sets of pairs satisfying each of (1-3),(5),(6) individually are closed. This is implied by the fact that, in each case, if a pair $\langle ff, D\rangle$ does not satisfy the axiom then this fact is discernable from checking if $\langle ff, D\rangle$ contains each of some finite set of behaviours. For example, if it fails (1) then *either* $(<>,\emptyset) \notin ff$ (one behaviour) or there are $s < t$ such that $(s,\emptyset) \notin ff$ and $(t,\emptyset) \notin ff$ (two behaviours). The most interesting case is axiom (3). Here, because we are restricting ourselves to finite failures, the set $Y$ must (implicitly) be finite itself. Failure to satisfy this

axiom would manifest itself by $(s, X) \in \mathit{ff}$, $(s, X \cup Y) \notin \mathit{ff}$ and $(s\langle a\rangle, \emptyset) \notin \mathit{ff}$ for each $a \in Y$. (It is the need to make (3) closed that has forced us down into finite failures.)

Since the topology on $A$ has been established to be homeomorphic to a Hausdorff one it follows trivially that the toplogies on $\mathcal{N}_{fin}$, and hence on $\mathcal{N}$, are Hausdorff also.

Since $\mathcal{N}$ is compact Hausdorff under $\delta$ it follows from the equivalence of the $\delta$-topology on $\mathcal{N}^\Lambda$ and the product topology that the $\delta$-topology on $\mathcal{N}^\Lambda$ is compact Hausdorff (Tychonoff's theorem again). The topology is zero-dimensional because it has a basis of closed and open (clopen) sets: the ball $N_{(T,F,\Phi)}(\underline{P})$ is closed because (as is easily checked), the complement of this ball is equal to

$$\bigcup\{N_{(T,F,\Phi)}(\underline{Q}) \mid \underline{Q} \notin N_{(T,F,\Phi)}(\underline{Q})\}$$

which is an open set. This completes the proof of 2.5 (c).

Each of the predicates described in (d) have the property that their failure can be demonstrated by at most two behaviours from a single component of $Q$. For example, the failure of $\underline{Q} \geq \underline{P}$ is either demonstrated by some divergence of $Q_\lambda$ not being present in $P_\lambda$, $(s, \emptyset)$ not being in $\mathcal{F}[\![Q]\!]$ for some minimal divergence of $P$ or $(s, X) \in \mathcal{F}[\![Q]\!] \not\Rightarrow (s, X) \in \mathcal{F}[\![P]\!]$ for some $s \notin \mathcal{D}[\![P]\!]$ and $X$ finite. (Here we use the fact that $\mathcal{F}[\![P]\!]$ and $\mathcal{F}[\![Q]\!]$ are determined by their sets of finite failures.)

The $\delta$-topology on $\mathcal{N}'$ is not even $T_0$ because when two processes have the same set of divergences and finite failures there is no open set that contains one without the other. In fact it inherits compactness from $\mathcal{N}_{fin}$ in the same way as above, since its open sets are precisely the inverse images of $\mathcal{N}_{fin}$'s open sets under the natural projection.

When $\Sigma$ is infinite, none of the stronger topologies on $\mathcal{N}$ can be compact because of the theorem that no proper extension of a compact Hausdorff topology can be compact. The $\gamma$-topology on $\mathcal{N}'$ is easily shown to be Hausdorff (any pair of distinct processes differ on some behaviour) but turns out not to be compact, essentially because the infinitary version of axiom (3) is then not closed in the sense described above.

This concludes our discussion of the $\alpha$, $\beta$, $\gamma$ and $\delta$ topologies. We now turn our attention more generally to the subject of the interval and lower topologies. (For a much deeper discussion of these, in particular over lattices, see [CCL].)

The interval topology is always $T_1$, for each singleton set is closed ($[x, x] = \{x\}$) and the lower topology is always $T_0$. The interval topology often satisfies stronger separation properties than this (for example compact Hausdorff) but

need not, as is demonstrated by the lattice with only an infinite set of incomparable points between top and bottom. The lower topology never satisfies $T_1$ in a nontrivial partial order, for if $x < y$ then every open set containing $y$ contains $x$.

Over a consistently complete cpo $X$ (i.e., one where each finite set of elements with an upper bound have a least upper bound) the lower topology is always compact. To prove this we appeal to a theorem of Alexander (see, for example [K]) that a topology is compact if each cover by elements from some sub-basis has a finite subcover. Thus in our case it sufficient to prove that if $S$ is a subset of $X$ such that for all finite $F \subseteq S$ the set $\bigcap\{x\uparrow \mid x \in F\}$ is nonempty, then $\bigcap\{x\uparrow \mid x \in S\}$ is nonempty. The assumptions imply that each finite $F \subseteq S$ has a least upper bound $\bigsqcup F$. The set of these as $F$ varies forms a directed set, the limit of which is the least upper bound of $S$, proving that $\bigcap\{x\uparrow \mid x \in S\}$ is nonempty as desired, for it must contain this limit.

A simple extension of this argument shows that the interval topology over a consistently complete cpo is compact also. (In fact, one proves that the perhaps finer topology with sub-basis of closed sets $\{x\uparrow, x\downarrow \mid x \in X\}$ is compact, where $x\downarrow = \{y \mid y \leq x\}$.) For in that case any subset $G$ of this sub-basis with the finite intersection property splits into two parts $\{x\uparrow \mid x \in S\} \cup \{y\downarrow \mid y \in T\}$. For as before, the least upper bound of $S$ exists, and since $x\uparrow \cap y\downarrow \neq \emptyset$ when $x \in S$ and $y \in T$ we know that each $y \in T$ is an upper bound for $S$, so it follows that $\bigsqcup S \leq y$ for all $y \in T$ and hence that $\bigsqcup S \in \bigcap G$.

**Acknowledgements.** I would like to thank Peter Collins for help in discovering the Alexander sub-basis theorem used in the Appendix. A number of people, in particular Tom Verhoeff, have helped me by pointing out minor errors and stylistic improvements. Several people, including Carroll Morgan, have pointed out the similarity of this work with that of Nelson which was refered to in the conclusion.

**References.**

[B] Brookes, S.D., *A Model for Communicating Sequential Processes*, Oxford University D.Phil. thesis, 1983.

[BHR] Brookes, S.D., Hoare, C.A.R., and Roscoe, A.W., *A Theory of Communicating Sequential Processes*, Journal of the Association for Computing Machinery, vol. 31, no. 3, 560-599.

[BR] Brookes, S.D., and Roscoe A.W., *An improved failures model for communicating processes* , Springer Lecture Notes in Computer Science, vol. 197, 281-305.

[BRW] Brookes, S.D., Roscoe A.W., and Walker, D.J., *An operational semantics for CSP*, Submitted for publication.

[CCL] Gierz, G., Hofmann, K.H., Keimel, K., Lawson, J.D., Mislove, M., and Scott, D.S., *A compendium of continuous lattices*, Springer-Verlag (1980).

[H] Hoare, C.A.R., *Communicating sequential processes*, Prentice-Hall, 1985

[HBR] Hoare, C.A.R., Brookes, S.D., and Roscoe, A.W., *A theory of communicating sequential processes*, Oxford University Computing Laboratory, Programming Research Group, Technical Report PRG-16.

[K] Kelley, J.L., *General Topology*, Springer GTIM27 (1975).

[N] Nelson, G., *A generalisation of Dijkstra's calculus*, Research report 16, Digital Systems Research Center, 1987.

[R1] Roscoe, A.W., *A mathematical theory of communicating processes*, Oxford University D.Phil. thesis, 1982.

[R2] Roscoe, A.W., *Unbounded nondeterminism in CSP*, in this volume.

[R3] Roscoe, A.W., *Induction and fixpoint rules for CSP networks*, in preparation.

[Re] Reed, G.M., *A uniform mathematical theory for real-time distributed computing*, Oxfrod University D.Phil. thesis, 1988.

[RR1] Reed, G.M., and Roscoe, A.W., *A timed model for communicating sequential processes*, Proceedings of ICALP'86, Springer LNCS 226 (1986), 314-323.

[RR2] Reed, G.M., and Roscoe, A.W., *Metric spaces as models for real-time concurrency*, to appear in the proceedings of MFPLS87 (Springer LNCS).

# Unbounded nondeterminism in CSP

by A.W. Roscoe[1]

**Oxford University Computing Laboratory,
8-11 Keble Road, Oxford OX1 3QD, U.K.**

## 0. Introduction

As is well known to the theoretical community, it is generally far easier to model finite nondeterminism (where a process can only choose between finitely many options at any one time) than unbounded nondeterminism (where no such restriction applies). The difficulties encountered with unbounded nondeterminism have hitherto forced us to restrict the language and semantics of CSP to avoid it: the most obvious restrictions being our inability to define the hiding operator $P\backslash B$ when $B$ is infinite and the absence of an infinite nondeterminism operator $\sqcap S$ for arbitrary nonempty sets $S$ of processes.

Both of these restrictions are inconvenient. If we are to allow infinite alphabets at all (and it is often useful to have them) it seems unnatural restricting hiding to finite sets: for example it is impossible to define a piping operator $\gg$ where the alphabet of interaction contains the integers. The fact that there is no infinite nondeterministic composition means that there is sometimes no most nondeterministic process satisfying a specification when one might expect

[1]The author gratefully acknowledges that the work reported in this paper was supported by ONR grant N00014-87-G-0242.

there would be. This is unfortunate because it becomes impossible to treat such specifications as though they were (parts of) programs.

In an earlier paper [R2] I showed how many of the restrictions on unbounded nondeterminism could be lifted by separating the nondeterminism order from the order used for finding fixed points. Unfortunately the structure of the model used there (failures and divergences using only finite traces) means that the semantics given by that model to unboundedly nondeterministic operators is not sufficiently discriminating. That model can successfully model a process which will, on its first step, nondeterministically choose any integer, but cannot tell between a process which can communicate any finite number of $a$s and one which may also choose to communicate an infinite number. One purpose of this paper is to develop a more refined model which can make this sort of distinction. This is done in the first section by adding a component of infinite traces so that any CSP process is represented by $(F, D, I)$ where $F$ is its set of failures (still with finite traces), $D$ is its set of (finite) divergence traces and $I$ is the set of infinite traces it can communicate.

Unfortunately the obvious orders on this new model fail to be complete, though they do have greatest lower bounds for arbitrary nonempty sets, which means that the standard iterative technique will produce the least fixed point of monotone $f$ provided there is any $x$ with $f(x) \leq x$. My first reaction to this failure was to look for a new order coarser than the obvious one which was complete (for this was precisely what I had been able to do in the paper mentioned above for the $(F, D)$ model without the finite subsets axiom). However one can prove that no order which gives the right semantics can be complete. Specifically we find an $\omega$–sequence of CSP-definable processes whose semantic values are provably ordered in any sensible order but which can have no least upper bound.

If recursions are well defined we must therefore find some special property of CSP-definable functions which leads them to have fixed points. In a sense the rest of the paper is devoted to this task, though it achieves far more. The chosen route was to investigate the connections between the abstract semantics for CSP given in Section 1 with an operational semantics given in terms of transition diagrams or synchronisation trees. This had already been done for standard CSP in [BRW].

The second section develops the abstraction maps between transition systems and the abstract models described in Section 1, and proves results about them. It also shows how the map to the infinite traces model can be approximated by a sequence of maps that are produced by iterating a functional.

In [BRW] the congruence proof was proved in two steps by the introduction of an intermediate denotational tree semantics between the term-rewriting tree semantics and the abstract denotational semantics. It turns out that there is little problem in defining an infinitely nondeterministic term-rewriting semantics: this hardly notices any difference between finite and infinite branching. However

the metric theory of trees (plus the contraction mapping theorem) that we used as the foundation of the intermediate semantics before is now of no use at all, for the existence of such a theory is dependent on finite branching and König's Lemma.

It is possible to develop the proof along similar lines by making use of the theory of infinitely branching trees reported in [R3]. However, the relative complexity of that and the discovery of the partial abstraction functions alluded to above, which take over an important function of the intermediate semantics, mean that it is no longer so attractive to do so. Therefore in the present paper the proof is executed in a single step. The final section is devoted to this. The structural induction which proves the congruence simultaneously proves that all CSP-definable functions do indeed have least fixed points. Let $\Phi$ be the natural map from synchronisation trees to the infinite traces model, and $\mathcal{S}$ be the denotational semantic function mapping CSP terms to it. If $P$ is any CSP term and $\rho$ is any binding of the free process variables of $P$ to closed CSP terms (ones without free process variables) it is shown that

$$\mathcal{S}[\![P]\!]\overline{\rho} = \Phi(\mathcal{O}[\![P]\!]\rho)$$

where $\overline{\rho}[\![p]\!] = \Phi(\rho(p))$ for each identifier $p$ and $\mathcal{O}[\![P]\!]\rho$ is the tree produced by the operational semantics when the free variables of $P$ have been substituted by the appropriate $\rho[\![p]\!]$.

Thus, as well as proving the abstract semantics well defined, we have related it to the corresponding operational semantics. To the author, one of the most interesting insights brought by this work has been an understanding of why the denotational fixed points of noncontinuous operators, which often take longer than $\omega$ to reach, are nevertheless operationally correct.

Despite the mathematical complexity of the results alluded to above, the actual definition of the infinite traces model and the semantics of CSP over it are by no means inaccessible. For someone who is willing to take the justification of these on trust, there is no need to read beyond the middle of the first section. Some technical material from the first section, such as the well-definedness of the CSP operators, has been relegated to an appendix.

**A note on alphabets.** The concept of a process' alphabet is of vital importance to the definition of the CSP parallel operator: when two processes are running in parallel, the combination can communicate an event in the combination alphabet if and only if all processes to whose alphabet it belongs are willing to communicate that event. There have been two ways of presenting this. In earlier papers [BHR,B,R1] alphabets were introduced as an explicit parameter of the parallel operator $_B|\!|_C$ . In some more recent works, particularly [H] they have been attributed to all processes: in essence this leads to a *typed* theory of CSP. The latter (alphabetised) theory leads to a more elegant syntax for the description of processes and the presentation of algebraic laws, but is messier when it comes to building abstract models, for there has to be a separate model for

every possible alphabet, including the empty one which is an annoying special case. Since the majority of this paper is concerned with the construction and analysis of models we here adopt the former (unalphabetised) style and use a single, universal, nonempty alphabet $\Sigma$ for all processes. However the two presentations are trivially equivalent and it is easy to determine the value predicted in the alphabetised theory from its value in the unalphabetised theory. Thus, subject to the obvious modifications, everything which is proved in this paper is equally valid in either.

# 1. Adding infinite traces to the failures model

We take as our basis the failures/divergence model with the new "definedness" order developed in [R2] but *without* the bounded nondeterminism axiom

$$\forall Y \subseteq^{\text{fin}} X.(s,Y) \in F \;\Rightarrow\; (s,X) \in F$$

which was needed before that order was developed. This model, as in the earlier paper, will be termed $\mathcal{N}'$. Recall that the definedness order $\leq$ is defined

$$\begin{aligned} P \leq Q \;\Leftrightarrow\; & \mathcal{D}[\![Q]\!] \subseteq \mathcal{D}[\![P]\!] \wedge \\ & s \notin \mathcal{D}[\![P]\!] \Rightarrow \mathcal{R}[\![P]\!]s = \mathcal{R}[\![Q]\!]s \wedge \\ & \mu(\mathcal{D}[\![P]\!]) \subseteq \mathit{traces}(Q) \end{aligned}$$

where $\mu T$ denotes the minimal elements of a set $T$ of finite traces and $\mathcal{R}[\![P]\!]s$ denotes $\{X \mid (s,X) \in \mathcal{F}[\![P]\!]\}$.

As was noted in the introduction, though this model can describe the sort of unbounded nondeterminism which makes itself apparent in a finite time, such as a process which can choose any integer on its first step but cannot deadlock, it cannot describe the type which takes infinitely long to unfold. This is exemplified by a process which, though it can perform every finite prefix of some infinite sequence, cannot perform the whole sequence. Such behaviour cannot arise in the context of finite nondeterminism, essentially because of König's Lemma: that any infiuite but finite branching tree (in our case the tree of ways in which the process can perform a prefix of the infinite trace) has an infinite path. (The fact that the hiding operator is operationally correct for finitely nondeterministic CSP is crucially dependent on this fact.) Thongh one might argue that the infinite behaviours of a process need not concern us, for they cannot be fully observed, the application of operators such as hiding can mean that the set of infinite behaviours influences the finite behaviours. For example, if $P_0 = SKIP$ and $P_{n+1} = a \rightarrow P_n$, need $(\sqcap\{P_n \mid n \in \mathbb{N}\})\backslash\{a\}$ terminate?

This sort of question arises because, when we model a process by its set of possible behaviours of one sort, we are often more interested in its *certain* behaviours of another. Thus, by recording all situations in which a process can diverge or refuse a set we can tell when it *must* accept from a given set in a finite

time. And by knowing all possible infinite sequences one can sometimes deduce that a process will do something in a finite time, as is demonstrated by the example at the end of the last paragraph. Notice how these ideas are consistent with the philosophy of the nondeterminism order $\sqsubseteq$: a process improves as it has less possible behaviours.

One cannot really hope to model this sort of unbounded nondeterminism without a record of the infinite traces that a process might perform. We therefore include such a record in a new model.

All the usual trace notations can be extended in the obvious ways to infinite traces, though of course one cannot concatenate $u$ with $s$ when $u$ is infinite. From here on $u$ will conventionally denote an infinite trace. The set of all infinite sequences of elements of any set $X$ will be written $X^\omega$.

The new model will have the same structure as $\mathcal{N}'$ except that it will have an extra component representing infinite traces. Thus a process $P$ will be a triple $(F, D, I)$, where $F \subseteq \Sigma^* \times \mathcal{P}(\Sigma)$, $D \subseteq \Sigma^*$ and $I \subseteq \Sigma^\omega$. $F$ should be nonempty and the eight axioms must be satisfied. The first seven are tabulated below.

$$
\begin{array}{lrcl}
(1) & (st, \emptyset) \in F & \Rightarrow & (s, \emptyset) \in F \\
(2) & (t, X) \in F \wedge Y \subseteq X & \Rightarrow & (t, Y) \in F \\
(3) & (t, X) \in F \wedge \forall a \in Y.(t\langle a \rangle, \emptyset) \notin F & \Rightarrow & (t, X \cup Y) \in F \\
(4) & s \in D & \Rightarrow & st \in D \\
(5) & s \in D & \Rightarrow & (st, X) \in F \\
(6) & su \in I & \Rightarrow & (s, \emptyset) \in F \\
(7) & s \in D & \Rightarrow & su \in I
\end{array}
$$

Axioms (6) and (7) are both new but straightforward because they are simple extensions to axioms (1) and (4) respectively. One more axiom is required, which can be thought of as an infinite trace analogue to axiom (3). The latter says that anything which, on one step, cannot be refused, must be a possible communication. The new axiom will say that when one, from the finite convergent behaviour, can show that there must be infinite traces, then there are enough of them.

This axiom is very subtle and, perhaps because it is unlike any of the others, proved hard to derive.

One can often prove from the failures of a nondivergent process that some infinite trace is possible because one can formulate a strategy for *forcing* one. The most simple-minded form of strategy is that based on a single infinite trace $u$. If $(s, \{a\}) \notin F$ for all $s\langle a \rangle < u$ then it is intuitively clear that a user single mindedly striving for the infinite trace $u$ must be successful. However there are more subtle versions of this. Consider a process whose failure-set is

$$F_0 = \{(s, X) \mid s \in \{a, b\}^* \wedge \{a, b\} \not\subseteq X\} .$$

Imagine always offering this process the set $\{a, b\}$: it is never refused, so we can guarantee that an infinite trace must arise. However we have no finer control over exactly which infinite trace it is, though on further reflection we can observe that, since every finite sequence $s$ of $a$s and $b$s is possible there must be an infinite trace $su$ extending every such $s$. The necessity of some axiom reflecting the forcing of infinite traces is demonstrated by the definition of the hiding operator below. Studying this will reveal that if a process $P$ with the above failures did not have an infinite trace, then $P\backslash\{a, b\}$ would not have any failures, divergences or infinite traces!

It was true in all versions of the failures model that, modulo divergence, every process was identical to the nondeterministic composition of its deterministic implementations. (There is an extensive discussion of this fact in [B]. An up to date paper which shows the power of this idea is [Blam].) The intuitive argument that applied in these models, that on any particular interaction with a process one cannot tell that it is not deterministic unless it diverges, still applies in our current situation. This property is a consequence of the principle that an external observer cannot tell by experimenting on a process just when it makes a nondeterministic choice. The only circumstances in which an observer could make such a distinction would be if he could copy a process half way through an experiment on it. No CSP operator can do this, so in modelling CSP we generally adopt this principle. It is intimately related to the fact that CSP operators are distribntive over nondeterministic choice, for example,

$$a \rightarrow (P \sqcap Q) = (a \rightarrow P) \sqcap (a \rightarrow Q)\,.$$

Imagine for the moment that we conld take copies of processes in mid execution, and that this is done with the processes above after $a$ has been communicated. Then the various copies of the process on hte left may act variously as $P$ or $Q$, since there is no guarantee that the nondeterministic choice has been made before copying. But all copies of the one on the right must behave the same (all as $P$ or all as $Q$), for there we know the choice has already been made. Thus, if we could take such copies, the above law would not be naturally valid.

One can tell if a process is deterministic by inspecting its set $F$ of failures. They must satisfy

$$(s, \emptyset) \in F \Rightarrow ((s, X) \in F \Leftrightarrow (X \cap \{a \mid (s\langle a\rangle, \emptyset) \in F\} = \emptyset))\,.$$

The infinite traces of a deterministic process are easy to determine: since it can never refuse any event which it can communicate, one can clearly force it to communicate any infinite trace $u$, all of whose finite prefixes are traces. We can thus categorically state that, if $F$ satisfies the above condition then

$$I = \{u \mid \forall s < u.(s, \emptyset) \in F\}\,.$$

In models involving divergence one has the problem that, since no deterministic process can diverge, a process whose set of divergences is nonempty is

not the nondeterministic composition of its deterministic implementations. One can easily get around this by defining a process to be *pre-deterministic* if it is deterministic until it diverges (if it does). A process $\langle F, D\rangle$ or $\langle F, D, I\rangle$ will be said to be pre-deterministic if

$$(s,\emptyset) \in F \Rightarrow ((s,X) \in F \Leftrightarrow s \in D \vee X \cap \{a \mid (s\langle a\rangle,\emptyset) \in F\} = \emptyset)\,.$$

We can determine the infinite traces of such a process exactly as before: we must have

$$I = \{u \mid \forall s < u.(s,\emptyset) \in F\}$$

for either, when trying to force an infinite sequence, the process eventually diverges (in which case the trace should be present by axiom (7) and is in the right hand side above by axiom (4)) or it does not, in which case the argument is as for deterministic processes. Even though we do not know what the last axiom will be, we *do* now know what the set of pre-deterministic elements of the new model is.

The nondeterminism order $\sqsubseteq$ extends trivially to the new model. If $P = \langle F, D, I\rangle$ and $P' = \langle F', D', I'\rangle$ are any two triples we say

$$P \sqsubseteq P' \equiv F \supseteq F' \wedge D \supseteq D' \wedge I \supseteq I'\,.$$

If $P = \langle F, D, I\rangle$ is any triple we define the set of its pre-deterministic *implementations* by

$$imp(P) = \{Q \mid Q \sqsupseteq P \wedge Q \text{ is pre-deterministic}\}\,,$$

noting that this is well-defined by the observation above that the set of pre-deterministic processes is known already.

If $S$ is any nonempty set of processes we define its nondeterministic composition $\sqcap S$ to be $\langle F, D, I\rangle$, where

$$\begin{aligned} F &= \bigcup\{F' \mid \langle F', D', I'\rangle \in S\} \\ D &= \bigcup\{D' \mid \langle F', D', I'\rangle \in S\} \\ I &= \bigcup\{I' \mid \langle F', D', I'\rangle \in S\}\,. \end{aligned}$$

This is just the process which can exhibit any behaviour of any element of $S$.

This allows us to state axiom (8).

(8) $imp\langle F, D, I\rangle$ is nonempty and $\langle F, D, I\rangle = \sqcap imp\langle F, D, I\rangle$.

In other words, every behaviour of a process arises from some pre-deterministic implementation. One should perhaps note that, as we defined pre-deterministic processes above, this axiom in fact implies axioms (6) and (7). However it would not be fair to exclude these as they were certainly taken into account when constructing the definition of pre-deterministic processes. Axiom (8) can be regarded as a statement about what sets $I$ are allowable for given $F$ and $D$

since, for any $F$ and $D$ satisfying (1)-(5) there are sets $I$ satisfying (1)-(8), for example the set of all infinite traces all of whose finite prefixes are traces.

Axiom (8) is both complex and, in style, unlike any other published axiom for a CSP model. The reader who wishes to convince himself that it really is the "right" axiom should look at the Appendix, where several quite different formulations are derived and where it is shown that the various CSP operators defined below preserve the axioms, and also at the proof of Theorem 2.2 where it is shown that every real (i.e., operational) process satisfies it.

The set of all triples satisfying the above eight axioms will be termed $\mathcal{U}$.

The notations of $\mathcal{N}$ and $\mathcal{N}'$ are extended to $\mathcal{U}$ in their obvious ways. If $P = (F.D, I)$, we define $\mathcal{I}[\![P]\!] = I$, $\mathcal{D}[\![P]\!] = D$ and $\mathcal{F}[\![P]\!] = D$. $traces(P)$ will continue to denote the finite traces of $P$; $Traces(P)$ will denote $traces(P) \cup \mathcal{I}[\![P]\!]$.

The main motivation for deriving axiom (8) was to force there to be enough infinite traces to reconcile with what the failures prove the process can be forced to do. We have noted that it places no bound on the finite $(F, D)$ behaviours of a process. One interesting question which is of importance later on is answered by Lemma 1.1.

**Lemma 1.1.** If $\langle F, D, I\rangle \in \mathcal{U}$, $\langle F, D, I'\rangle$ satisfies axioms (1) to (7) (the only one that has to be checked is axiom (6)) and $I' \supseteq I$, then $\langle F, D, I'\rangle \in \mathcal{U}$.

**Proof.** We need to show that there are elements of $imp\langle F, D, I'\rangle$ containing every element of $I' \setminus I$, for every other behaviour is accounted for by an element of $imp\langle F, D, I\rangle (\subseteq imp\langle F, D, I'\rangle)$. Since, if $u \in I' \setminus I$, we have $imp\langle F, D, I \cup \{u\}\rangle \subseteq imp\langle F, D, I'\rangle$ it will be sufficient to show that there exists $P_u \in imp\langle F, D, I \cup \{u\}\rangle$ such that $u \in \mathcal{I}[\![P_u]\!]$. Note that $s < u$ implies $s \notin D$ (or else $u \in I$).

For each $s < u$ and each $a \in \Sigma$ such that $s\langle a\rangle \not< u$ and $(s\langle a\rangle, \emptyset) \in F$, choose $P_{s\langle a\rangle} \in imp\langle F, D, I\rangle$ such that $(s\langle a\rangle, \emptyset) \in \mathcal{F}[\![P_{s\langle a\rangle}]\!]$ (it must exist as $\langle F, D, I\rangle$ satisfies axiom (8)). We will construct a pre-deterministic process by offering all possible options on the next step that $\langle F, D, I\rangle$ can as long as the trace remains a prefix of $u$, but immediately it ceases to be (i.e., has the form $s\langle a\rangle \not< u$ where $s < u$) it behaves like $P_{s\langle a\rangle}$. Formally, $P_u$ is defined to be $\langle F_u, D_u, I_u\rangle$ where

$$
\begin{aligned}
F_u &= \{(s, X) \mid s < u \wedge X \cap \{a \mid (s\langle a\rangle, \emptyset) \in F\} = \emptyset\} \\
&\quad \cup \{(s\langle a\rangle s', X) \mid s < u \wedge s\langle a\rangle \not< u \wedge (s\langle a\rangle, \emptyset) \in F \wedge (s\langle a\rangle s', X) \in \mathcal{F}[\![P_{s\langle a\rangle}]\!]\} \\
D_u &= \{s\langle a\rangle s' \mid s < u \wedge s\langle a\rangle \not< u \wedge (s\langle a\rangle, \emptyset) \in F \wedge s\langle a\rangle s' \in \mathcal{D}[\![P_{s\langle a\rangle}]\!]\} \\
I_u &= \{u\} \cup \{s\langle a\rangle u' \mid s < u \wedge s\langle a\rangle \not< u \wedge (s\langle a\rangle, \emptyset) \in F \wedge s\langle a\rangle u' \in \mathcal{D}[\![P_{s\langle a\rangle}]\!]\}
\end{aligned}
$$

It is elementary to check that $P_u \in imp\langle F, D, I \cup \{u\}\rangle$, which completes the proof of the Lemma. □

One consequence of this result is that we can interpret axiom (8) as saying that there are enough infinite traces to account for the set of failures: the infinite traces for divergences are implied by axiom (7) and this Lemma says that adding

extra infinite traces never implies the addition of yet more. In other words axiom (8) could be replaced by the weaker statement

$$F = \bigcup\{F' \mid \langle F', D', I' \rangle \in imp(P)\}$$

for all $P = (F, D, I)$. This observation is expanded on in the Appendix.

The reader might like to check that the elements of $\mathcal{U}$ with failure set $F_0$ as defined above are precisely $\langle F_0, \emptyset, I \rangle$ where $I$ is a set of nonempty infinite traces such that every element of $\{a, b\}^*$ is a prefix of some element of $I$. This follows in part from the fact that, if $P$ is a process with the given failure set and $Q \in imp(P)$, then $Q$ must have an infinite trace as it is easy to prove that it has arbitrarily long finite traces. Some possible $I$s are $\{a, b\}^\omega$, and $\{su \mid s \in \{a, b\}^*\}$ for any fixed $u \in \{a, b\}^\omega$. Some more examples will be seen a little later.

We have already indicated above how the nondeterminism order is extended to $\mathcal{U}$. It is obvious that the maximum elements are precisely the deterministic processes as defined above and that the least element is the immediately divergent process.

Since divergence (and hence undefinedness) always appears after a finite length of trace, there is no obvious way of extending the idea of definedness to infinite traces. We therefore extend $\leq$ in the same way as above: the order on the infinite traces being by reverse inclusion.

$$\begin{aligned} P \leq Q \;\Leftrightarrow\; & \mathcal{D}[\![Q]\!] \subseteq \mathcal{D}[\![P]\!] \wedge \\ & s \notin \mathcal{D}[\![P]\!] \Rightarrow \mathcal{R}[\![P]\!]s = \mathcal{R}[\![Q]\!]s \wedge \\ & \mu(\mathcal{D}[\![P]\!]) \subseteq traces(Q) \wedge \\ & \mathcal{I}[\![P]\!] \supseteq \mathcal{I}[\![Q]\!] \end{aligned}$$

At first sight it might seem a more natural extension of $\leq$ if we made processes with different infinite traces after convergent behaviour incomparable. However, such an order would be incomplete in a disturbing way because there would be sequences with many minimal upper bounds. As we shall see later in examples of how recursions converge, we do genuinely seem to need the structure of reverse inclusion here.

The following lemma characterises a few useful elementary properties of the two partial orders.

**Lemma 1.2.**

a) $P \sqsubseteq Q$ if, and only if, $imp(P) \supseteq imp(Q)$.

b) $P \leq Q \Rightarrow P \sqsubseteq Q$

c) $\perp = (\Sigma^* \times \mathcal{P}(\Sigma), \Sigma^*, \Sigma^\omega)$ is the least element of $\mathcal{U}$ for both orders.

d) If $P \leq R$ and $P \sqsubseteq Q \sqsubseteq R$, then $P \leq Q$.

e) A process $P$ is pre-deterministic if and only if there is a deterministic $Q$ such that $P \leq Q$.

f) The $\sqsubseteq$-maximal elements of $\mathcal{U}$ are precisely the deterministic processes.

**Proof.** (a), (b) and (c) are trivial. For (d), we observe that $P \leq Q$ if and only if $P \sqsubseteq Q$ and

(i) $(s,X) \in \mathcal{F}[\![P]\!] \wedge s \notin \mathcal{D}[\![P]\!] \Rightarrow (s,X) \in \mathcal{F}[\![Q]\!]$, and

(ii) $\mu(\mathcal{D}[\![P]\!]) \subseteq traces(Q)$,

so to prove the result it will be sufficient to prove (i) and (ii). If $(s,X) \in \mathcal{F}[\![P]\!] \wedge s \notin \mathcal{D}[\![P]\!]$ then, since $P \leq R$, we know $(s,X) \in \mathcal{F}[\![R]\!]$. Hence $(s,X) \in \mathcal{F}[\![Q]\!]$ as $Q \sqsubseteq R$. Exactly the same argument applies for (ii).

Part (e) is elementary once we observe that if $P$ is not pre-deterministic then its nondeterministic convergent behaviour must be present in any $Q$ such that $P \leq Q$.

It is easy to show that if $P$ and $Q$ are both deterministic and $P \sqsubseteq Q$ then $P = Q$. It follows that if $P$ is deterministic then $imp(P) = \{P\}$, and, by (a), that all deterministic processes are maximal. It is easy to see that, for any $P \in \mathcal{U}$, $imp(P)$ contains a deterministic process $Q$ (since any pre-deterministic process is weaker than some deterministic one by (e)). It follows that $P \sqsubseteq Q$ and hence that no nondeterministic process can be maximal. This proves (f). □

All the usual operators may be defined over $\mathcal{U}$. As one would expect, in most cases the finite parts of these definitions are exactly the same as before (with the notable exception of hiding). They are given in full below.

*STOP* and *SKIP* are defined

$$STOP = \langle \{(<>, X) \mid X \subseteq \Sigma\}, \emptyset, \emptyset \rangle$$

$$SKIP = \langle \{(<>, X) \mid \surd \notin X\} \cup \{((\surd), X) \mid X \subseteq \Sigma\}, \emptyset, \emptyset \rangle .$$

Let $P = \langle F, D, I \rangle$, $P' = \langle F', D', I' \rangle$ and, for $b \in B$, $P_b = \langle F_b, D_b, I_b \rangle$ be

processes. Then

$$
\begin{aligned}
\mathcal{D}[\![a \to P]\!] &= \{\langle a\rangle s \mid s \in D\} \\
\mathcal{I}[\![a \to P]\!] &= \{\langle a\rangle u \mid u \in I\} \\
\mathcal{F}[\![a \to P]\!] &= \{(<>, X) \mid a \notin X\} \cup \{(\langle a\rangle s, X) \mid (s, X) \in F\} \\
\\
\mathcal{D}[\![x : B \to P_x]\!] &= \{\langle b\rangle s \mid b \in B \wedge s \in D_b\} \\
\mathcal{I}[\![x : B \to P_x]\!] &= \{\langle b\rangle u \mid b \in B \wedge u \in I_b\} \\
\mathcal{F}[\![x : B \to P_x]\!] &= \{(<>, X) \mid B \cap X = \emptyset\} \cup \{(\langle b\rangle s, X) \mid b \in B \wedge (s, X) \in F_b\} \\
\\
\mathcal{D}[\![P \sqcap P']\!] &= D \cup D' \\
\mathcal{I}[\![P \sqcap P']\!] &= I \cup I' \\
\mathcal{F}[\![P \sqcap P']\!] &= F \cup F' \\
\\
\mathcal{D}[\![P \,\Box\, P']\!] &= D \cup D' \\
\mathcal{I}[\![P \,\Box\, P']\!] &= I \cup I' \\
\mathcal{F}[\![P \,\Box\, P']\!] &= \{(<>, X) \mid (<>, X) \in F \cap F'\} \cup \{(s, X) \mid s \neq <> \wedge (s, X) \in F \cup F'\} \\
&\quad \cup \{(s, X) \mid s \in \mathcal{D}[\![P \,\Box\, P']\!]\} \\
\\
\mathcal{D}[\![P \;{}_B\|_C\; P']\!] &= \{st \mid s \in (B \cup C)^* \wedge s\restriction B \in D \wedge s\restriction C \in \mathit{traces}(P')\} \\
&\quad \cup \{st \mid s \in (B \cup C)^* \wedge s\restriction B \in \mathit{traces}(P) \wedge s\restriction C \in D'\} \\
\mathcal{I}[\![P \;{}_B\|_C\; P']\!] &= \{u \in (B \cup C)^\omega \mid u\restriction B \in \mathit{Traces}(P) \wedge u\restriction C \in \mathit{Traces}(P')\} \\
&\quad \cup \{su \mid s \in \mathcal{D}[\![P \;{}_B\|_C\; P']\!]\} \\
\mathcal{F}[\![P \;{}_B\|_C\; P']\!] &= \{(s, (X \cap B) \cup (Y \cap C) \cup Z) \mid s \in (B \cup C)^* \wedge (s\restriction B, X) \in F \wedge \\
&\quad (s\restriction C, Y) \in F' \wedge Z \cap (B \cup C) = \emptyset\} \\
&\quad \cup \{(s, X) \mid s \in \mathcal{D}[\![P \;{}_B\|_C\; P']\!]\} \\
\\
\mathcal{D}[\![P \,|||\, P']\!] &= \bigcup\{\mathit{merge}\langle s, t\rangle \mid s \in D \wedge t \in \mathit{traces}(P')\} \\
&\quad \cup \bigcup\{\mathit{merge}\langle s, t\rangle \mid s \in D' \wedge t \in \mathit{traces}(P)\} \\
\mathcal{I}[\![P \,|||\, P']\!] &= \bigcup\{\mathit{merge}\langle s, t\rangle \mid s \in \mathit{Traces}(P) \wedge t \in \mathit{Traces}(P') \wedge s \text{ or } t \text{ is infinite}\} \\
\mathcal{F}[\![P \,|||\, P']\!] &= \{(s, X) \mid \exists t, t'.\, s \in \mathit{merge}\langle t, t'\rangle \wedge (t, X) \in F \wedge (t', X) \in F'\} \\
&\quad \cup \{(s, X) \mid s \in \mathcal{D}[\![P \,|||\, P']\!]\} \\
\\
\mathcal{D}[\![P; P']\!] &= \{st \mid s \in D \wedge s \text{ tick-free}\} \\
&\quad \cup \{st \mid s\langle\surd\rangle \in \mathit{traces}(P) \wedge t \in D' \wedge s \text{ tick-free}\} \\
\mathcal{I}[\![P; P']\!] &= \{u \mid u \in I \wedge u \text{ tick-free}\} \\
&\quad \cup \{su \mid s\langle\surd\rangle \in \mathit{traces}(P) \wedge u \in I' \wedge s \text{ tick-free}\} \\
&\quad \cup \{su \mid s \in \mathcal{D}[\![P; P']\!]\} \\
\mathcal{F}[\![P; P']\!] &= \{(s, X) \mid (s, X \cup \{\surd\}) \in F \wedge s \text{ tick-free}\} \\
&\quad \cup \{(st, X) \mid s\langle\surd\rangle \in \mathit{traces}(P) \wedge s \text{ tick-free} \wedge (t, X) \in F'\} \\
&\quad \cup \{(s, X) \mid s \in \mathcal{D}[\![P; P']\!]\}
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{D}[\![P\backslash X]\!] &= \{(u\backslash X)t \mid u \in I \wedge u\backslash X \text{ is finite}\} \cup \{(s\backslash X)t \mid s \in D\} \\
\mathcal{I}[\![P\backslash X]\!] &= \{u\backslash X \mid u \in I \wedge u\backslash X \text{ is infinite}\} \cup \{su \mid s \in \mathcal{D}[\![P\backslash X]\!]\} \\
\mathcal{F}[\![P\backslash X]\!] &= \{(s\backslash X, Y) \mid (s, X \cup Y) \in F\} \cup \{(s, Y) \mid s \in \mathcal{D}[\![P\backslash X]\!]\} \\
\\
\mathcal{D}[\![f[P]]\!] &= \{(f(s))t \mid s \in D\} \\
\mathcal{I}[\![f[P]]\!] &= \{f(u) \mid u \in I\} \cup \{(f(s))u \mid s \in D\} \\
\mathcal{F}[\![f[P]]\!] &= \{(f(s), X) \mid (s, f^{-1}(X)) \in F\} \cup \{(s, X) \mid s \in \mathcal{D}[\![f[P]]\!]\} \\
\\
\mathcal{D}[\![f^{-1}[P]]\!] &= \{s \mid f(s) \in D\} \\
\mathcal{I}[\![f^{-1}[P]]\!] &= \{u \mid f(u) \in I\} \\
\mathcal{F}[\![f^{-1}[P]]\!] &= \{(s, X) \mid (f(s), f(X)) \in F\}
\end{aligned}
$$

We have seen above how the nondeterministic composition of an arbitrary nonempty collection of processes may be defined by component-wise union.

The only definition here that really requires comment is that of hiding. The definition of $\mathcal{D}[\![P\backslash X]\!]$ is rather simpler than before, since a divergence caused by the hiding now arises from a single infinite behaviour rather than from an infinite collection of finite ones. Notice that, with this exception, failures and divergences never depend on the infinite traces of the operands. Some fundamental properties of these operators are summarised in the next theorem.

**Theorem 1.3.** All the operators above are well defined (i.e., preserve the axioms) and monotonic with respect to both orders. All operators are both finitely and infinitely distributive: i.e., $F(\sqcap S) = \sqcap\{F(P) \mid P \in S\}$ for all operators $F$ and nonempty $S \subseteq \mathcal{U}$.

With the exception of hiding the only term that needs to be checked for well-definedness and monotonicity is $I$, for we already know these facts for $\mathcal{N}'$. In these cases monotonicity is trivial, for infinite traces are always constructed positively out of behaviours of the operands. Axioms (6) and (7) are generally easy to check. Some example proofs of axiom (8) are given in the Appendix. Distributivity is a consequence of the fact that all behaviours of $F(P)$ are always deducible from single behaviours of $P$. This is also discussed in the Appendix. We should perhaps note that no claim has been made for the continuity of the operators, which is because many of them are not continuous as a consequence of unbounded nondeterminism. See later examples for discussion of this. The main consequence of this lack of continuity is that the fixed points of recursively defined programs need not have appeared by the $\omega$th iteration from $\perp$ so familiar to computer scientists. However, once we can show that necessary least upper bounds exist there is no problem in defiuing the meaning of any recursive term to be the least fixed point of the appropriate monotone function: it is given by $f^{\alpha}(\perp)$ for sufficiently large $\alpha$. Once one can do this, we can define a semantic function $\mathcal{S} : \mathbf{E} \to \mathit{UEnv} \to \mathcal{U}$, where $\mathbf{E}$ is the set of all CSP terms and $\mathit{UEnv}$ is the set of mappings from process variables to $\mathcal{U}$, in the obvious way. □

**Properties of the partial orders.** We have seen how the partial orders $\leq$ and $\sqsubseteq$ are defined. We cannot hope that $\sqsubseteq$ is complete in general, for it is not complete over $\mathcal{N}'$ when $\Sigma$ is infinite. Unfortunately, *neither* order is complete, even when $\Sigma = \{a, b\}$. It is easy to construct increasing $\leq$-sequences of processes, all with $F = F_0$ as defined above and $D = \emptyset$ which can have no upper bound. As a simple example, let $u_n = ((a)^n\langle b\rangle)^\omega$ be the infinite trace which has $n$ $a$s then a $b$ cyclically. It is clear that the sets $\{su_n \mid s \in \{a,b\}^*\}$ are disjoint as $n$ varies, and therefore that, if we set $I_n = \{su_m \mid s \in \{a,b\}^* \wedge m \geq n\}$, any upper bound for the sequence $\langle\langle F_0, \emptyset, I_n\rangle \mid n \in \mathbb{N}\rangle$ must have an empty set of infinite traces. This is impossible for $\leq$ as, since all the processes are divergence-free, any upper bound must have failure set $F_0$. (And we have already observed that all such elements of $\mathcal{U}$ have nonempty $I$.) It is also impossible for $\sqsubseteq$ since any upper bound must have an implementation $Q$ (necessarily deterministic). $Q$ must also be an implementation of all processes in the sequence and therefore have an infinite trace – a contradiction.

We will return to this incompleteness shortly and show that it is, to some extent at least, inevitable. Before we do this, however, it will be nice to establish a few positive properties.

Theorem 1.4.

a) Any nonempty subset $S$ of $\mathcal{U}$ has greatest lower bounds with respect to both $\leq$ and $\sqsubseteq$. In general, $\bigsqcap_{\leq} S \sqsubseteq \bigsqcap_{\sqsubseteq} S$.

b) In either order, any subset of $\mathcal{U}$ with any upper bound has a least upper bound.

c) If $\bigsqcup_{\leq} S$ is defined then so is $\bigsqcup_{\sqsubseteq} S$ and the two are equal. Furthermore $\bigsqcup_{\leq} S = P^* = \langle F^*, D^*, I^*\rangle$, where $F^* = \bigcap\{F \mid \langle F, D, I\rangle \in S\}$, $D^* = \bigcap\{D \mid \langle F, D, I\rangle \in S\}$ and $I^* = \bigcap\{I \mid \langle F, D, I\rangle \in S\}$.

d) If $S$ is a nonempty set then $\bigsqcup_{\sqsubseteq} S$ exists if and only if $\bigcap\{imp(P) \mid P \in S\}$ is nonempty, and in that case $\bigsqcup_{\sqsubseteq} S = \bigsqcap(\bigcap\{imp(P) \mid P \in S\})$.

e) If $f : \mathcal{U} \to \mathcal{U}$ is a function which is monotone with respect to one of the orders and there is $P \in \mathcal{U}$ such that $f(P) \leq P$ (respectively $f(P) \sqsubseteq P$), then $f$ has a least fixed point given by $f^\alpha(\bot)$ for some ordinal $\alpha$.

f) If $f : \mathcal{U} \to \mathcal{U}$ is monotone with respect to both orders then any least fixed point for one order is also the least fixed point for the other.

**Proof.** It is easy to see that $\bigsqcap S$ is the $\sqsubseteq$-greatest lower bound of any nonempty set $S$. (For an explanation of why $\bigsqcap S$, as defined above, is in $\mathcal{U}$, see the Appendix.) It does not work in general for the definedness order $\leq$, however, since one does not in general have $P \in S \Rightarrow \bigsqcap S \leq P$. The greatest lower bound of $S = \{\langle F_i, D_i, I_i\rangle \mid i \in \Lambda\}$ is, as was the case in $\mathcal{N}'$, constructed so that it diverges as soon as the finite behaviour of any two elements of $S$ starts to differ. We define $\bigsqcap_{\leq} S$ to be $\langle F, D, I\rangle$, where

- $D = \bigcup\{D_i \mid i \in \Lambda\} \cup \{st \mid \exists i, j.(\exists Y.(s, Y) \in F_i \backslash F_j) \vee (\exists a.(s\langle a\rangle, \emptyset) \in F_i \backslash F_j)\}$
- $F = \bigcup\{F_i \mid i \in \Lambda\} \cup \{(s, X) \mid s \in D\}$
- $I = \bigcup\{I_i \mid i \in \Lambda\} \cup \{su \mid s \in D\}$

It is easy to show that this process is in $\mathcal{U}$ and is indeed the $\leq$ greatest lower bound of $S$. Trivially $\bigsqcap_{\leq} S \sqsubseteq \bigsqcap S$. This completes the proof of (a).

(b) follows because, as is fairly well known, *any* partial order which has greatest lower bounds for nonempty sets has this property. The usual argument is repeated here. If $S$ is a set with an upper bound, then $U_S$, the set of upper bounds of $S$ is nonempty and so $x = \bigsqcap U_S$ exists. Since $y \leq z$ whenever $y \in S$ and $z \in U_S$ it follows that each $y \in S$ is a lower bound for $U_S$. As $x$ is the *greatest* lower bound for $S$ it follows that $x \geq y$ for all $y \in S$ and therefore that $x \in U_S$. Plainly $x$ is the least element of $U_S$ and is therefore the least upper bound of $S$.

The first part of (c) follows trivially from the formula which is the second part. However it has an interesting separate proof. Note that, since $Q \leq P \Rightarrow Q \sqsubseteq P$, if $P = \bigsqcap_{\leq} S$ exists then it is a $\sqsubseteq$-upper bound for $S$ and hence $Q = \bigsqcup_{\sqsubseteq} S$ exists and $Q \sqsubseteq P$. Whenever $R \in S$ we then have $R \sqsubseteq Q \sqsubseteq P$ and $R \leq P$. Lemma 1.2 (d) above then tells us that $R \leq Q$. It follows that $Q$ is a $\leq$-upper bound for $S$ and hence that $Q \geq P$. We then have $Q \sqsubseteq P$ and $P \sqsubseteq Q$. The result follows immediately.

For the second part, we show first that if $P' = \langle F', D', I'\rangle$ is the actual least upper bound on $S$ then $D^* = D'$. For trivially $D' \subseteq D^*$ so let $s \in \mu D^*$ (where recall $D^* = \bigcap\{D \mid \langle F, D, I\rangle \in S\}$). Note that there must be $P = \langle F, D, I\rangle \in S$ such that $s \in \mu D$. Since $P \leq P'$ we must have $s \in \mathit{traces}(P')$. If $s \notin D'$ then consider $P'' = \langle F'', D'', I''\rangle$ defined

$$\begin{aligned} F'' &= F' \cup \{(st, X) \mid t \in \Sigma^* \wedge X \subseteq \Sigma\} \\ D'' &= D' \cup \{st \mid t \in \Sigma^*\} \\ I'' &= I' \cup \{su \mid u \in \Sigma^\omega\}\,. \end{aligned}$$

$\mathit{traces}(P'')$ is prefix closed by the observation above. It is thus easy to see that $P''$ is a process, that $P \leq P''$ for all $P \in S$ and that $P' \not\leq P''$. It follows that $P'$ cannot be the least upper bound on $S$, a contradiction. Hence $\mu D^* \subseteq D'$; it easily follows that $D^* \subseteq D'$, so the two are equal as desired.

That $P^*$ defined in the statement of the theorem satisfies axioms (1), (2), (4), (5), (6) and (7) is trivial. We next note that trivially $F^* \supseteq F'$. Now by the above paragraph those parts of $F^*$ and $F'$ implied by divergence and axiom (5) are equal. Suppose that $s \notin D' = D^*$. Then there is $P = \langle F, D, I\rangle \in S$ such that $s \notin D$. Necessarily $\mathcal{R}[\![P]\!]s = \mathcal{R}[\![P']\!]s$ as $P \leq P'$. It follows that $\mathcal{R}[\![P']\!]s \supseteq \mathcal{R}[\![P^*]\!]s$ (for the latter is the intersection of a set containing $\mathcal{R}[\![P]\!]s$). Putting these fact together yields $F' \supseteq F^*$, proving that in fact $F' = F^*$. Note that this implies that $P^*$ satisfies axiom (3).

Since we have now shown that $D^* = D'$ and $F^* = F'$, and it is trivial that $I^* \supseteq I'$ it follows directly from Lemma 1.1 that $P^*$ satisfies axiom (8) and is therefore in $\mathcal{U}$. The fact that it is the $\leq$-least upper bound for $S$ is then trivial. This completes the proof of (c).

(d) follows easily from axiom (8) and (b) above.

(e) is true in any partial order with property (a). By another standard argument, if $f$ is monotonic and $x = \bigsqcap\{P \mid f(P) \leq P\}$ exists in a partial order then it is the least fixed point of $f$. We still have to show that the least fixed point can also be found by iterating $f^\alpha(\bot)$. The only place at which the standard cpo proof of this could go wrong is where, for limit ordinals $\lambda$, one defines $f^\lambda(\bot) = \bigsqcup\{f^\alpha(\bot) \mid \alpha \in \lambda\}$ since this least upper bound might not be defined. But it always is, since it is easy to prove by transfinite induction that all the $f^\alpha(\bot)$ are bounded above by the least fixed point $x$ constructed above so that we can always apply (b) when constructing $f^\lambda(\bot)$.

(f) follows easily from (c) and (e). If $f$ is monotonic with respect to both orders and has any fixed point then it follows easily from (e) that it has least fixed points $f^\alpha_{\leq}(\bot)$ and $f^\beta_{\sqsubseteq}(\bot)$ with respect to these two orders. But one can prove from (c) that if both of these exist then the value of $f^\gamma(\bot)$ is independent of whether it was defined using $\leq$ or $\sqsubseteq$ by an easy transfinite induction on $\gamma$. From this it is easily seen that both processes reach the same fixed point, and do so at the same time.

(f) can alternatively be proved by observing that, by (e), if $f$ has a fixed point then it has a least fixed point with respect to both orders. If $x$ and $y$ denote the $\leq$-least and $\sqsubseteq$-least fixed points respectively, we have $x \leq y$ and hence $x \sqsubseteq y$ by Lemma 1.2. But we know $y \sqsubseteq x$ so it follows that $x = y$. $\square$

We should remark now that all of the properties of the partial orders identified in Lemma 1.2 and Theorem 1.4 extend easily (some of them appropriately amended) to products of $\mathcal{U}$, i.e., $\mathcal{U}^\Lambda (= \Lambda \to \mathcal{U})$ for an arbitrary nonempty set $\Lambda$, with the order $\underline{P} \leq \underline{Q}$ (or $\underline{P} \sqsubseteq \underline{Q}$) if and only if $P_\lambda \leq Q_\lambda$ (or $P_\lambda \sqsubseteq Q_\lambda$) for all $\lambda \in \Lambda$. Some of the more useful properties of these product spaces, which are important in the consideration of mutual recursions and in the definition of the partial abstraction functions later on, are summarised below. All the proofs are either standard or straightforward extensions of what we have already seen.

**Theorem 1.5.**

a) $\bot^\Lambda$ is least element of $\mathcal{U}$ with respect to both orders.

b) Any nonempty subset $S$ of $\mathcal{U}^\Lambda$ has greatest lower bounds with respect to both $\leq$ and $\sqsubseteq$. In general, $\bigsqcap_{\leq} S \sqsubseteq \bigsqcap_{\sqsubseteq} S$. In either case the greatest lower bound's $\lambda$-component is given by $\bigsqcap\{P_\lambda \mid P \in S\}$, where $\bigsqcap$ here denotes the greatest lower bound operator over $\mathcal{U}$ in the appropriate order.

c) In either order, any subset of $\mathcal{U}$ with any upper bound has a least upper bound. In that case its $\lambda$-component is given by $\bigsqcup\{P_\lambda \mid P \in S\}$.

d) If $\bigsqcup_{\leq} S$ is defined then so is $\bigsqcup_{\sqsubseteq} S$ and the two are equal. Furthermore $(\bigsqcup_{\leq} S)_\lambda = P^*_\lambda = \langle F^*_\lambda, D^*_\lambda, I^*_\lambda \rangle$, where $F^*_\lambda = \bigcap\{\mathcal{F}[\![P\lambda]\!] \mid \underline{P} \in S\}$ $D^*_\lambda = \bigcap\{\mathcal{D}[\![P\lambda]\!] \mid \underline{P} \in S\}$ $I^*_\lambda = \bigcap\{\mathcal{I}[\![P\lambda]\!] \mid \underline{P} \in S\}$.

e) If $f : \mathcal{U}^\Lambda \to \mathcal{U}^\Lambda$ is a function which is monotone with respect to one of the orders and there is $\underline{P} \in \mathcal{U}^\Lambda$ such that $f(\underline{P}) \leq \underline{P}$ (respectively $f(\underline{P}) \sqsubseteq \underline{P}$), then $f$ has a least fixed point given by $f^\alpha(\bot^\Lambda)$ for some ordinal $\alpha$.

f) If $f : \mathcal{U}^\Lambda \to \mathcal{U}^\Lambda$ is monotone with respect to both orders then any least fixed point for one order is also the least fixed point for the other. □

These theorems and what we have shown up to now show that $\leq$ and $\sqsubseteq$ are exceptionally well-behaved partial orders. It is interesting to note that $\sqsubseteq$ has its lower bounds given by union and $\leq$ has its upper bounds given by intersection, but that the reverse facts are not true. For example $\sqcap\{a \to STOP, b \to STOP\} = \bot$ or $(a \to STOP) \sqcap (b \to STOP)$ depending on which order is chosen, and $\bigsqcup\{(a \to STOP) \sqcap (b \to STOP, (a \to STOP) \sqcap (b \to SKIP)\} = a \to STOP$ under $\sqsubseteq$ which is not the intersection of the two. Indeed even in cases where $S$ is a chain, $\bigsqcup_{\sqsubseteq} S$ might exist but not be given by component-wise intersection. If $P_n$ is the $n$th process in the chain seen earlier with no upper bound, then if we define

$$Q_n = (c \to STOP) \sqcap (d \to P_n)$$

the least upper bound of this sequence is $c \to STOP$ even though $(\langle d \rangle, \emptyset)$ is a failure of every $Q_n$.

The author's first reaction on finding that the two "natural" partial orders were incomplete was to try to find another one that was but which gave the same semantics. After all, that had been one of the main reasons for the development of the $\leq$ order over $\mathcal{N}'$ since it gave exactly the same least fixed point semantics but was complete, showing that all desired fixed points actually exist. I should perhaps remark at this point that the given orders do actually compute the correct values for CSP definable recursions and that the least upper bounds required to compute them always exist. Of course the proof of these facts will be the subject of much work later, but it is worthwhile seeing some examples here.

**Examples.** Define $P_0 = STOP$ and $P_{n+1} = a \to P_n$. Set $P = \sqcap\{P_n \mid n \in \mathbb{N}\}$, so that $P$ can perform any finite number of $a$s but not an infinite sequence of them. Operationally we can think of $P$ as a process which, as its first action, takes a secret decision on exactly how many $a$s to perform. Now consider the recursively defined process

$$Q = (a \to Q) \|_{\{a\}} P$$

and let $F : \mathcal{U} \to \mathcal{U}$ be the function associated with the right hand side of this recursion. Since the right hand side of the highest level parallel construct initially imposes a bound on the number of $a$s $Q$ can perform, it is clear that $Q$ itself

cannot perform an infinite sequence of them. On the other hand it is clear that $Q$ can perform as large a finite number of $a$s as it pleases. We would therefore expect $P = Q$. However, as is easily verified, $F^{\omega}(\bot)$ can perform an infinite sequence of $a$s (it is equal to $P \sqcap R$, where $R = a \rightarrow R$). On the other hand, $F^{\omega+1}(\bot) = (a \rightarrow (P \sqcap R))_{\{a\}} \|_{\{a\}} P = P$ and $F(P) = P$, so this recursion reaches the operationally correct fixed point at $\omega+1$. Some more examples of recursions, their fixed points and the ordinal required to reach them are summarised below. The reader might enjoy constructing a few of his own.

- If $f : \Sigma \rightarrow \Sigma$ is such that $f^n(a) \neq f^m(a)$ when $n \neq m$ then the recursion

$$Q_1 = STOP \sqcap a \rightarrow ((Q_1 \,_\Sigma\|_\Sigma P) \sqcap f(Q_1))$$

  (with $P$ as above) reaches its fixed point (which is the same as that of the recursion $P' = P \sqcap a \rightarrow f[P']$ which converges in $\omega$ steps), in exactly $\omega.2$ iterations.

- Let $\alpha$ be an infinite ordinal and $\Sigma = \alpha$ (the set of all $\beta < \alpha$). Then the recursion

$$Q_2 = \beta : \alpha \rightarrow ((\gamma : \beta \rightarrow STOP) \,_\Sigma\|_\Sigma Q_2)\backslash\alpha$$

  takes exactly $\alpha$ steps to converge to its fixed point $\beta : \alpha \rightarrow STOP$. $Q_2$ is a process that inputs any element $\beta$ of $\alpha$ and then outputs any element of $\beta$ to a copy of itself or deadlocks if $\beta = 0$. (The fact that this is the natural fixed point is an easy consequence of the fact that there is no infinite descending sequence of ordinals.)

Suppose $\preceq$ is some partial order which does all we want: namely give the same fixed point theory and make $\mathcal{U}$ complete. Clearly it must make all CSP operators monotonic and have the same minimal element $\bot$. To give the same fixed point theory it must have the property that, when $C$ is a linearly ordered subset of $\mathcal{U}$ with respect to $\preceq$ and one of our existing orders, then a least upper bound for $\preceq$ is also a least upper bound for the other. (Note that $\sqsubseteq$ and $\leq$ are in this relationship.) It must also make $P' \prec Q$, where $Q$ is defined as in the example above and $P' = STOP \sqcap a \rightarrow P'$. For $Q$ is a fixed point of this recursion but is distinct from the natural fixed point (by assumption the $\prec$-least) which has the infinite sequence of $a$s. ($P' \prec Q$ can also be proved by looking at the recursion of $Q$, where $P'$ is the $\omega$th iterate.)

From these simple facts and assumptions we will be able to prove that $\preceq$ cannot exist: for there is a sequence of processes in $\mathcal{U}$ which are provably ordered by $\preceq$ but which can have no upper bound. Set $\Sigma = \{a, b\}$. Recall that the set $F_0$ of failures was defined

$$F_0 = \{(s, X) \mid s \in \{a, b\}^* \wedge \{a, b\} \not\subseteq X\} .$$

The corresponding set where a process can refuse anything at any time is

$$F_1 = \{(s, X) \mid s \in \{a, b\}^* \wedge X \subseteq \{a, b\}\} .$$

Recall that the triples $(F_0, \emptyset, I)$ satisfying the axioms were those where $I$ contains an extension of every finite trace. All triples $(F_1, \emptyset, I)$ satisfy the axioms.

We will now construct some subsets of $\{a,b\}^\omega$ to go along with $F_0$ and $F_1$. If $u \in \{a,b\}^\omega$ and $n \in \mathbf{N}$, define $r_n(u)$ to be the ratio of the number of $a$s to the number of $b$s plus one in the first $n$ elements of $u$. (The "plus one" is to make this always defined.) We should perhaps remark that some traces $u$ have $\lim_{n\to\infty} r_n(u)$ existing and some do not. (In fact, there are uncountably many $u$s with any given limit in $[0,\infty)$.) In the author's experience the ratios $r_n(u)$ are very useful when it comes to choosing pathological subsets of $\{a,b\}^\omega$ and similar.

For $n \in \{1,2,3,\ldots\}$ we define

$$I_n = \{u \in \{a,b\}^\omega \mid \exists \epsilon > 0. \exists m. \forall k \geq m. \epsilon < r_k(u) < \frac{1}{n} - \epsilon\} .$$

Thus $u \in I_n$ if and only if the ratios eventually stay within $(0, \frac{1}{n})$ and away from the boundaries of that interval. This last condition means, amongst other things, that $I_n$ contains no sequence with limit 0 or $\frac{1}{n}$. Notice that $u \in I_n$ does not imply that $\lim_{n\to\infty} r_n(u)$ exists. The sets $I_n$ have some interesting properties. First, the $I_n$ all contain elements beginning with any chosen $s \in \{a,b\}^*$ (in fact, uncountably many). Also $I_{n+1} \subseteq I_n$ and $\bigcap\{I_n \mid n \in \{1,2,\ldots\}\} = \emptyset$. Perhaps the most interesting property is that, if $m \leq n$ then

$$\bigcup\{merge\langle s,t\rangle \mid s \in I_n \cup \{a,b\}^* \wedge t \in I_m \cup \{a,b\}^* \wedge s \text{ or } t \text{ is infinite}\} = I_m .$$

Also, the insertion or deletion of finitely many elements of a sequence $u$ does not effect membership of any $I_n$ since the limiting behaviour $r_n(u)$ is not affected by such manipulations. We can now define some processes

$$\begin{array}{lcll} P_n & = & \langle F_0, \emptyset, I_n\rangle & \text{for } n \in \{1,2,3,\ldots\} \\ Q_n & = & \langle F_1, \emptyset, I_n\rangle & \text{for } n \in \{1,2,3,\ldots\} \\ P_0 & = & \langle F_0, \emptyset, \{a,b\}^\omega\rangle & \\ Q_0 & = & \langle F_1, \emptyset, \{a,b\}^\omega\rangle & \\ Q_\infty & = & \langle F_1, \emptyset, \emptyset\rangle & \end{array}$$

We will prove that the $P_n$ are a $\preceq$-increasing sequence.

Now if $f : \Sigma \rightarrow \Sigma$ is defined by $f(a) = f(b) = a$, we have $f^{-1}[P'] = Q_0$ and $f^{-1}[Q] = Q_\infty$, where $P'$ and $Q$ are as described at the start of this discussion. Hence $Q_0 \preceq Q_\infty$ as $f^{-1}$ is monotonic.

Now for all $n$ it is not too hard to see that $P_n \,|||\, Q_0 = P_0$ and $P_n \,|||\, Q_\infty = P_n$. It follows that $P_0 \preceq P_n$ for all $n \geq 1$ as $|||$ is monotonic.

Next, observe that $P_n \,{}_\Sigma\|_\Sigma\, P_m = Q_n$ if $m \leq n$. (The transition from $F_0$ to $F_1$ arises because one side of the parallel may refuse $a$ and the other $b$.) It follows that $Q_m = (P_0 \,{}_\Sigma\|_\Sigma\, P_m) \preceq (P_n \,{}_\Sigma\|_\Sigma\, P_m) = Q_n$ when $m \leq n$.

The property of the $I_n$ described above implies that $P_m \,|||\, Q_n = P_k$, where $k$ is the lesser of $n$ and $m$. Hence, when $m \leq n$, $P_m = P_n \,|||\, Q_m \preceq P_n \,|||\, Q_n = P_n$. This completes the proof that the $P_n$ form an increasing sequence.

The fact the the $P_n$ are $\preceq$-increasing is unsurprising, since they are increasing with respect to $\sqsubseteq$ and $\leq$. We have specified that all $\preceq$ least upper bounds are also $\sqsubseteq$ least upper bounds. Since $\bigcap\{I_n \mid n \in \mathbb{N}\}$ is empty, any $\sqsubseteq$ least upper bound for this sequence has $I = \emptyset$. But there is no element of $\mathcal{U}$ with $F \subseteq F_0$ and $I = \emptyset$. It follows that this sequence has no upper bound with respect to $\preceq$. Therefore $\preceq$ cannot be complete.

We therefore have to give up all bope of a conventional fixed point theory, though note that, by Theorem 1.4, if we can show every CSP term has some fixed point then we essentially have one. One of the reasons the author embarked upon the research set out in the rest of this paper was to prove that every such function has a fixed point. However, it must also be said that a theory like ours which is based on monotonic functions and fixed points which are attained at arbitrary ordinals is by no means obviously right in an intuitive sense. There is even more need now to relate this abstract semantics to an operational semantics than there was for the cousiderably simpler boundedly nondeterministic version if we are to understand how it works (if, indeed it does). It will turn out that by the final section we will be able to show that all CSP definable functions have least fixed points and to prove the congruence result described in the introduction.

Nevertheless it would be very nice to have some simpler argument that every CSP function has a fixed point that did not rest on such a large body of work outside the model. For example, if one could find any partial order which was complete and made all operators monotouic one would be able to show every function we want has a fixed point. Theorem 1.4 would then ensure that the fixed point we actually *want* exists.

## 2. Abstraction functions

When we come to compare abstract and transition system semantics in the final sectiou we will need abstraction functions which map one to the other. The purpose of this section is to introduce and analyse functions from arbitrary transition systems to $\mathcal{U}$ and $\mathcal{N}'$. But before we do this we summarise some facts about transition systems.

**Summary of notation, nomenclature and results.** A *transition system* is a set of states with a binary relation $\xrightarrow{\delta}$ for each element $\delta$ of the set $\Sigma^\tau = \Sigma \cup \{\tau\}$ of transitions, where $\tau$ deuotes an internal transition. We should note that $\Sigma$ (the set of visible actions) is an implicit parameter of almost everything we do from now on, as indeed it was in the last section.

A *morphism* [R1,R3] is a function from one transition system to another which characterises the property of indistinguishability in that no experimenter

who can only see transitions (visible or invisible) should be able to tell $P$ from $F(P)$ if $F$ is a morphism. $F : C \to D$ is said to be a morphism if and only if:

(i) $P \xrightarrow{\delta} Q \Rightarrow F(P) \xrightarrow{\delta} F(Q)$, and

(ii) $F(P) \xrightarrow{\delta} X \Rightarrow \exists Q.P \xrightarrow{\delta} Q \wedge F(Q) = X$.

Morphisms are closely related to the idea of bisimulation but differ mainly in that they treat internal actions in exactly the same rigid way that they treat visible ones.

The *index of nondeterminism* $i(C)$ of a transition system $C$ is the smallest infinite regular cardinal[2] which is strictly larger than $\{Q \mid P \xrightarrow{\delta} Q\}$ for all $P \in C$ and $\delta \in \Sigma^+$.

**The functions.** Given an element $P$ of a transition system, we can construct its sets of failures, divergence and infinite traces in natural ways which are described below.

We first define two multi-step versions of the transition relation. If $P, Q \in C$ and $s = \langle x_i \mid 0 \le i < n \rangle \in (\Sigma^+)^*$ we say $P \overset{s}{\longmapsto} Q$ if there exist $P_0 = P, P_1, \ldots, P_n = Q$ such that $P_k \xrightarrow{x_k} P_{k+1}$ for $k \in \{0, 1, \ldots, n-1\}$. Unlike this first version, the second ignores $\tau$s. For $s \in \Sigma^*$ we write $P \overset{s}{\Longrightarrow} Q$ if there exists $s' \in (\Sigma^+)^*$ such that $P \overset{s'}{\longmapsto} Q$ and $s' \setminus \tau = s$. The following properties of $\overset{s}{\Longrightarrow}$ and $\overset{s}{\longmapsto}$ are all obvious.

**Lemma 2.1.**

a) $P \overset{\langle\rangle}{\Longrightarrow} P \wedge P \overset{\langle\rangle}{\longmapsto} P$

b) $P \overset{s}{\Longrightarrow} Q \wedge Q \overset{t}{\Longrightarrow} R \Rightarrow P \overset{st}{\Longrightarrow} R$

c) $P \overset{s}{\longmapsto} Q \wedge Q \overset{t}{\longmapsto} R \Rightarrow P \overset{st}{\longmapsto} R$

d) $P \overset{st}{\Longrightarrow} R \Rightarrow \exists Q.P \overset{s}{\Longrightarrow} Q \wedge Q \overset{t}{\Longrightarrow} R$

e) $P \overset{st}{\longmapsto} R \Rightarrow \exists Q.P \overset{s}{\longmapsto} Q \wedge Q \overset{t}{\longmapsto} R$

Suppose $C$ is a transition system and $P \in C$. We say $P$ can *diverge*, written $P\uparrow$, if there exist $P_0 = P, P_1, P_2, \ldots$ such that, for all $n \in \mathbb{N}$, $P_n \xrightarrow{\tau} P_{n+1}$.

$$divergences(P) = \{st \mid \exists Q.P \overset{s}{\Longrightarrow} Q \wedge Q\uparrow\}$$

Notice that we have said that $st$ is a divergence trace whenever $s$ is. This is motivated by a desire (inspired by our abstract semantics) to make all possibly divergent processes undefined. (As will be apparent from a careful reading of

[2] A regular cardinal $\lambda$ is one which is not the union of less than $\lambda$ sets all of which are of size less than $\lambda$. There are arbitrarily large regular cardinals, since for example every *successor* cardinal is regular. The combinatorial properties which make *regular* cardinals the natural bounds for nondeterminism are well illustrated in [R3].

the proofs below and in the final section, the fact that our semantic models and functions are strict with respect to divergence is sometimes of great importance.)

Say $P \in C$ is *stable* provided there is no $Q$ such that $P \xrightarrow{\tau} Q$ (in other words, if $P$ cannot make any internal progress). If $B \subseteq \Sigma$ we say $P$ *ref* $B$ if $\forall a \in B \cup \{\tau\}.\neg\exists Q \in C.P \xrightarrow{a} Q$. Thus $P$ *ref* $B$ implies that $P$ is stable. We can now define

$$\mathit{failures}(P) = \{(s,B) \mid \exists Q.P \stackrel{s}{\Longrightarrow} Q \wedge Q \mathit{\ ref\ } B\} \cup \{(s,B) \mid s \in \mathit{divergences}(P)\}\,.$$

The point of these definitions is that a process can properly refuse $B$ only when it is in a stable state, for as long as it is performing internal actions one cannot be sure that it will not come into a state where a desired event is possible. On the other hand, when a process diverges it also refuses (in a different sense perhaps) all communications offered to it. The second part of the definition is also motivated by the desire to make a divergent process undefined.

If $u \in \Sigma^\omega$ is an infinite trace and $P \in C$, we write $P \stackrel{u}{\Longrightarrow}$ if there are $P = P_0, P_1, P_2, \ldots \in C$ and $x_i \in \Sigma^+$ such that $\forall k.P_k \xrightarrow{a_k} P_{k+1}$ and $\langle a_k \mid k \in \mathbb{N} \wedge a_k \neq \tau\rangle = u$. This lets us define

$$\mathit{infinites}(P) = \{u \in \Sigma^\omega \mid P \stackrel{u}{\Longrightarrow}\} \cup \{su \mid s \in \mathit{divergences}(P) \wedge u \in \Sigma^*\}\,.$$

Similarly, if $\langle x_i \mid i \in \omega\rangle = u \in (\Sigma^+)^\omega$ we can write $P \stackrel{u}{\longmapsto}$ if there exist $P = P_0, P_1, P_2, \ldots$ such that, for all $i$, $P_i \xrightarrow{x_i} P_{i+1}$.

Clearly it is possible to define other functions, and to vary these definitions for another definition of divergence. However the above are exactly the required maps to define the abstraction maps into our main abstract models.

**Definition.** If $C$ is any transition system then we define abstraction maps $\Xi : C \to \mathcal{N}'$ (the failures/divergences model with no compactness axiom) and $\Phi : C \to \mathcal{U}$ as follows.

$$\Xi(P) = \langle \mathit{failures}(P),\ \mathit{divergences}(P)\rangle$$
$$\Phi(P) = \langle \mathit{failures}(P),\ \mathit{divergences}(P),\ \mathit{infinites}(P)\rangle$$

We now prove a theorem which establishes some basic properties of $\Phi$ and $\Xi$.

**Theorem 2.2.** The maps $\Xi$ and $\Phi$ are well defined, and furthermore

a) If $F : C \to D$ is a morphism then $\Xi(F(P)) = \Xi(P)$ and $\Phi(F(P)) = \Phi(P)$ for all $P \in C$.

b) If $P \in C$ and $C$ is a sub-system of $D$ (i.e., a subset closed under all the transition relations) then the values $\Phi(P)$ and $\Xi(P)$ do not depend on whether we think of $P$ as an element of $C$ or of $D$.

c) Given any transition system $C$ there is another one $C'$ such that $C$ is a subsystem of $C'$ and the maps $\Phi : C' \to \mathcal{U}$ and $\Xi : C \to \mathcal{N}'$ are onto.

**Proof.** We first prove the well-definedness of the maps $\Xi$ and $\Phi$. Observe that for any process $P$, either $<> \in$ *divergences*$(P)$ or there exists a stable $Q$ such that $P \stackrel{<>}{\Rightarrow} Q$. This and Lemma 2.1 (b) easily imply that the traces of $\Xi(P)$ and $\Phi(P)$ (finite or finite and infinite) are prefix closed. The other axioms of $\mathcal{N}'$ follow trivially once one observes that, if $Q$ *ref* $B$ and $B' \subseteq B$ then $Q$ *ref* $B'$ for any stable $Q \in C$, hence $\Xi(P)$ is well defined.

The only non-trivial thing which remains to be proved about $\Phi(P)$ is axiom (8). We have to prove that every behaviour of $\Phi(P)$ (failure, divergence or infinite trace) belongs to some element of $imp(\Phi(P))$. (The reverse inclusion being trivial.) As we remarked earlier, this axiom is true because on no actual interaction with the process can one tell it is nondeterministic. To prove it true of $\Phi(P)$ it will thus be necessary to isolate the ways in which a process can appear to act on a particular run and to show that these can be regarded as pre-deterministic and include every behaviour of $\Phi(P)$. There are two ways in which a transition system can generate nondeterminism: by executing a $\tau$ action and thereby changing the available actions invisibly and by having more than one possible result from carrying out a given action (visible or invisible) from some state.

Given $P$ in a transition system $C$, we will slightly abuse notation and use the term *pre-deterministic subtree of* $P$ for what is essentially the record of the behaviour of $P$ after it has made its internal decisions of the sorts described above. It will be a set $T$ of pairs $(s, Q)$ where $s \in \Sigma^*$ and $Q \in C$, with the meaning that $Q$ is the state that it might come into immediately on completing $s$. It must satisfy the following conditions

(i) $(<>, P) \in T$

(ii) $(s, Q), (s, Q') \in T \Rightarrow Q = Q'$

(iii) $traces(T) = \{s \mid \exists Q.(s, Q) \in T\}$ is prefix closed.

(iv) $(s, Q), (s\langle a\rangle, Q') \in T \Rightarrow \exists Q''.Q \stackrel{<>}{\Rightarrow} Q'' \wedge Q'' \stackrel{a}{\longrightarrow} Q'$

(v) $(s, Q) \in T \wedge \neg Q\uparrow \Rightarrow \exists Q'.Q \stackrel{<>}{\Rightarrow} Q' \wedge Q'$ *ref* $\{a \in \Sigma \mid s\langle a\rangle \notin traces(T)\}$.

Given such a subtree one can define a pre-deterministic process: $\Phi'(T)$ is defined to be $(F, D, I)$ where

$$\begin{aligned} D &= \{st \mid (s, Q) \in T \wedge Q\uparrow\} \\ F &= \{(s, X) \mid s \in traces(T) \wedge X \cap \{a \in \Sigma \mid s\langle a\rangle \in traces(T)\} = \emptyset\} \cup \{(s, X) \mid s \in D\} \\ I &= \{u \in \Sigma^\omega \mid \forall s < u.s \in traces(T)\} \cup \{su \mid s \in D\} \end{aligned}$$

**Lemma 2.2.1.** If $T$ is a pre-deterministic subtree for $P$ then $\Phi(P) \sqsubseteq \Phi'(T)$.

The proof of this lemma is easy and uninteresting except for the observation that because the state associated with each finite trace is unique one can simply construct a path through the transition system for each required infinite trace by stitching together the pieces for each of its elements produced by (iv) above. □

Every behaviour exhibited by $P$ (of any of the three types) is exhibited by one of its pre-deterministic subtrees. This is a consequence of the next lemma.

A path from $P$ may be defined to be sequences of $P_i$ and $x_i \in \Sigma^+$ such that $P_0 = P$ and $P_i \xrightarrow{x_i} P_{i+1}$. A path can either be finite or infinite. An infinite path is divergent if the $x_i$ are eventually all $\tau$s.

**Lemma 2.2.2.** If $P_0 \xrightarrow{x_0} P_1 \xrightarrow{x_1} P_2 \ldots$ is a nondivergent path from $P$ then there is a pre-deterministic subtree $T$ of $P$ such that

(i) If $x_i \neq \tau$ then $(\langle x_j \mid j \leq i\rangle \backslash \tau, P_{i+1}) \in T$, and

(ii) If $P_i$ is stable and $s = \langle x_j \mid j < i\rangle \backslash \tau$ then $\{a \mid s\langle a\rangle \in \mathit{traces}(T)\} = \{a \mid \exists Q.P_i \xrightarrow{a} Q\}$.

The proof of this lemma simply consists of the tedious construction of $T$ piece by piece. □

It follows from this that $\Phi(P) = \bigsqcap\{\Phi'(T) \mid T \text{ is a pre-deterministic subtree of } P\}$. This completes our proof that $\Phi$ is well defined.

We now turn to the proof of (a), namely that the values of the abstractions are preserved by morpbisms. The following all follow fairly easily from the definition of a morphism.

$$
\begin{aligned}
P \stackrel{s}{\Longrightarrow} Q &\Rightarrow F(P) \stackrel{s}{\Longrightarrow} F(Q) \\
F(P) \stackrel{s}{\Longrightarrow} R &\Rightarrow \exists Q.F(Q) = R \wedge P \stackrel{s}{\Longrightarrow} Q \\
Q \text{ stable} &\Leftrightarrow F(Q) \text{ stable} \\
Q \mathit{\ ref\ } B &\Leftrightarrow F(Q) \mathit{\ ref\ } B \\
Q\uparrow &\Leftrightarrow F(Q)\uparrow \\
Q \stackrel{u}{\longmapsto} &\Leftrightarrow F(Q) \stackrel{u}{\longmapsto} \\
Q \stackrel{u}{\Longrightarrow} &\Leftrightarrow F(Q) \stackrel{u}{\Longrightarrow} .
\end{aligned}
$$

For example, suppose $F(Q)\uparrow$. Then there exist $R_0 = F(Q), R_1, R_2, \ldots$ such that $R_i \xrightarrow{\tau} R_{i+1}$ for all $i$. Claim that there exist $Q_0 = Q, Q_1, Q_2, \ldots$ such that $F(Q_i) = R_i$ and $Q_i \xrightarrow{\tau} Q_{i+1}$. These will be constructed inductively, first setting $Q_0 = Q$. If we have constructed $Q_i$ then, as $F(Q_i) \xrightarrow{\tau} R_{i+1}$ there exists $Q_{i+1}$ such that $F(Q_{i+1}) = R_{i+1}$ and $Q_i \xrightarrow{\tau} Q_{i+1}$. It follows that the desired $Q_i$ exist. This trivially implies $Q\uparrow$.

It is easy to see that the above imply $\mathit{divergences}(P) = \mathit{divergences}(F(P))$, $\mathit{failures}(P) = \mathit{failures}(F(P))$ and $\mathit{infinites}(P) = \mathit{infinites}(F(P))$, which in turn implies part (a).

Part (b) is a trivial consequence of (a) since, if $C$ is a subsystem of $D$, the obvious inclusion map is a morphism.

We now turn to the proof of (c). This is fortunately rather easy given axiom (8). The pre-deterministic processes (which can be identified with the pairs $\langle T, D\rangle$ where, $\emptyset \neq T \subseteq \Sigma^*$ is prefix closed and $D \subseteq T$, may be turned into a transition system $PD$:

$$\begin{array}{ll} \langle T,D\rangle \xrightarrow{\tau} \langle T,D\rangle & \text{if } <>\in D \\ \langle T,D\rangle \xrightarrow{a} \langle \{s \mid \langle a\rangle s \in T\}, \{s \mid \langle a\rangle s \in D\}\rangle & \text{if } \langle a\rangle \in T. \end{array}$$

It is easy to check that $\Phi\langle T,D\rangle$ is precisely the pre-deterministic element of $\mathcal{U}$ that corresponds to $\langle T,D\rangle$.

Now all we have to do is to make up a space $C'$ consisting of (disjoint copies of) $C$, $PD$ and a separate point for each nonempty subset $S$ of $PD$. The transitions of elements of $C$ and $PD$ are those inherited from those systems. If $S$ is one of the subsets then

$$S \xrightarrow{\tau} \langle T,D\rangle \quad \text{for all } \langle T,D\rangle \in S\,.$$

The fact that the $\Phi : C' \to \mathcal{U}$ is onto is now a straightforward consequence of axiom (8). That $\Xi$ is onto follows trivially from the fact that $\Phi$ is and the fact that $\mathcal{N}'$ is exactly the set of all first two components of elements of $\mathcal{U}$.

This completes the proof of Theorem 2.2. □

It might seem a little crious that we have gone to the trouble of extending an *arbitrary* transition to one on which $\Phi$ is onto, especially when the disjoint sum construction is so trivial. The reason for this will become apperent when this result is used in the next section.

For reasons which will be apparent in the final section it is useful to have not only the map $\Phi : C \to \mathcal{U}$ but also a sequence of approximations to it. We will define a map $\Phi_\alpha : C \to \mathcal{U}$ for each ordinal $\alpha$. (Once again, $C$ is here an arbitrary transition system.) It is convenient to define $\Phi_\alpha$ in terms of a functional

$$\mathcal{G} : (C \to \mathcal{U}) \to (C \to \mathcal{U})\,.$$

If $\Psi : C \to \mathcal{U}$ and $P \in C$, we define $\mathcal{G}(\Psi)(P) = \langle F', D', I'\rangle$, where

$$\begin{array}{rcl} F' & = & \{(<>, X) \mid P \; ref \; X\} \\ & & \cup\{(s,X) \mid \exists Q.P \xrightarrow{\tau} Q \wedge (s,X) \in \mathcal{F}[\![\Psi(Q)]\!]\} \\ & & \cup\{(\langle a\rangle s, X) \mid \exists Q.P \xrightarrow{a} Q \wedge (s,X) \in \mathcal{F}[\![\Psi(Q)]\!]\} \\ D' & = & \{s \mid \exists Q.P \xrightarrow{\tau} Q \wedge s \in \mathcal{D}[\![\Psi(Q)]\!]\} \\ & & \cup\{\langle a\rangle s \mid \exists Q.P \xrightarrow{a} Q \wedge s \in \mathcal{D}[\![\Psi(Q)]\!]\} \\ I' & = & \{u \mid \exists Q.P \xrightarrow{\tau} Q \wedge u \in \mathcal{I}[\![\Psi(Q)]\!]\} \\ & & \cup\{\langle a\rangle u \mid \exists Q.P \xrightarrow{a} Q \wedge u \in \mathcal{I}[\![\Psi(Q)]\!]\} \end{array}$$

The following Theorem establishes some useful properties of $\mathcal{G}$.

**Theorem 2.3.**

a) $\mathcal{G}$ is well defined and monotonic with respect to both orders.

b) $\Phi$, as defined earlier in this section, is a fixed point of $\mathcal{G}$.

**Proof.** The whole of part (a) follows immediately from the fact that $\mathcal{G}$ can be re-written entirely in CSP. The operator $P \triangleright Q$ used below is an abbreviation for $(P \Box Q) \sqcap Q$ (the process which can offer the choice between $P$ and $Q$ but which must eventually make an internal transition to become $Q$ if no action occurs). It is a useful operator since it allows more conciseness, and has appeared before in similar circumstances in the literature, e.g. [].

$$\begin{array}{lcll} \mathcal{G}(\Psi)(P) & = & x : P^0 \to \sqcap\{\Psi(Q) \mid P \xrightarrow{x} Q\} & \text{if } \nexists Q.P \xrightarrow{\tau} Q \\ \mathcal{G}(\Psi)(P) & = & ((x : P^0 \to \sqcap\{\Psi(Q) \mid P \xrightarrow{x} Q\}) & \text{otherwise} \\ & & \triangleright \sqcap\{\Psi(Q) \mid P \xrightarrow{\tau} Q\} & \end{array}$$

where $P^0$ denotes $\{a \in \Sigma \mid \exists Q.P \xrightarrow{a} Q\}$. It is easy to see that our two definitions of $\mathcal{G}$ are equivalent. Note that the overall structure of this CSP definition depends only on the transitions within $C$, and is therefore independent of the value of $\Psi$. It is this last fact which proves that $\mathcal{G}$ is monotone with respect to both orders.

Part (b) is intuitively obvious. Consider, for example, the divergence component. It follows immediately from the definition of $\Phi$ that $\mathcal{D}[\![\Phi(P)]\!] = \mathit{divergences}(P)$ is equal to

$$\{st \mid P \xrightarrow{\tau} Q \wedge Q \stackrel{s}{\Longrightarrow} R \wedge R\uparrow\} \cup \{\langle a\rangle st \mid P \xrightarrow{a} Q \wedge Q \stackrel{s}{\Longrightarrow} R \wedge R\uparrow\}$$

which in turn is equal to

$$\{s \mid P \xrightarrow{\tau} Q \wedge s \in \mathit{divergences}(Q)\} \cup \{\langle a\rangle s \mid P \xrightarrow{a} Q \wedge s \in \mathit{divergences}(Q)\}$$

which is $\mathcal{D}[\![\mathcal{G}(\Phi)(P)]\!]$ by definition of $\mathcal{G}$. Both the other cases are similar and depend on this one. The failures case divides into three components rather than two for obvious reasons. □

By Theorem 1.5 applied to the product space $\mathcal{U}^C$ $(= C \to \mathcal{U})$, it follows from the existence of one fixed point that $\mathcal{G}$ has a least fixed point which is equal to $\Phi_\alpha$ for some $\alpha$ where

$$\begin{array}{rcll} \Phi_0(P) & = & \bot & \text{for all } P \in C \\ \Phi_\mu(P) & = & \bigsqcup\{\Phi_\beta(P) \mid \beta \in \mu\} & \text{if } \mu \text{ is a limit ordinal} \\ \Phi_{\beta+1} & = & \mathcal{G}(\Phi_\beta) & \end{array}$$

since $\Phi_0$ is the least element of the product space and $\Phi_\beta = \mathcal{G}^\beta(\Phi_0)$. These $\Phi_\beta$ will play a crucial role in the main congruence theorem in the next section. This is essentially because of the next theorem.

**Theorem 2.4.** $\Phi$ is the least fixed point of $\mathcal{G}$. Hence there exists $\alpha$ such that $\Phi_\alpha = \Phi$.

**Proof.** Let $\Phi_\alpha$ be the least fixed point of $\mathcal{G}$. We know by the above that $\Phi_\alpha \sqsubseteq \Phi$, so it will be sufficient to prove the reverse. In other words we should show that $\mathcal{D}[\![\Phi_\alpha(P)]\!] \subseteq \mathit{divergences}(P)$, $\mathcal{F}[\![\Phi_\alpha(P)]\!] \subseteq \mathit{failures}(P)$ and $\mathcal{I}[\![\Phi_\alpha(P)]\!] \subseteq \mathit{infinites}(P)$ all hold. In each case this is done by taking an arbitrary element of the left hand side and constructing a sequence of processes, either finite or infinite, which demonstrate that it is in the right hand side. We define functions

$$\begin{aligned} f_f : \{(P,(s,X)) \mid P \in C \wedge (s,X) \in \mathcal{F}[\![\Phi_\alpha(P)]\!]\} &\rightarrow (C \times \Sigma^+ \times (\Sigma^* \times \mathcal{P}(\Sigma))) \cup \{\Delta\} \\ f_d : \{(P,s)) \mid P \in C \wedge s \in \mathcal{D}[\![\Phi_\alpha(P)]\!]\} &\rightarrow C \times \Sigma^+ \times \Sigma^* \\ f_i : \{(P,u)) \mid P \in C \wedge s \in \mathcal{I}[\![\Phi_\alpha(P)]\!]\} &\rightarrow C \times \Sigma^+ \times \Sigma^\omega \end{aligned}$$

as follows, where $\Delta$ is some new object. If $(s,X) \in \mathcal{F}[\![\Phi_\alpha(P)]\!] = \mathcal{F}[\![\mathcal{G}(\Phi_\alpha)(P)]\!]$ then by definition of $\mathcal{G}$ one of the following three clauses must hold according to which of the clauses of the failures component of the definition of $\mathcal{G}$ applies.

(i) $P$ may be stable, $s = <>$ and $P$ *ref* $X$. In this case set $f_f(P,(s,X)) = \Delta$. If not then

(ii) there may exist $Q$ such that $P \xrightarrow{\tau} Q$ and $(s,X) \in \mathcal{F}[\![\Phi_\alpha(Q)]\!]$. In this case set $f_f(P,(s,X)) = (Q,\tau,(s,X))$. If not then

(iii) there must exist $Q$, $a$ and $s'$ such that $s = \langle a \rangle s'$ and $P \xrightarrow{a} Q$ and $(s',X) \in \mathcal{F}[\![Q]\!]$. In this case set $f_f(P,(s,X)) = (Q,a,(s',X))$.

If, in (ii) or (iii), there is more than one choice for $Q$ an arbitrary choice is made.

If $s \in \mathcal{D}[\![\Phi_\alpha(P)]\!] = \mathcal{D}[\![\mathcal{G}(\Phi_\alpha)(P)]\!]$ then by definition of $\mathcal{G}$ one of the following two cases must hold according to which of the clauses of the divergences component of $\mathcal{G}$ applies.

(i) There may exist $Q$ such that $P \xrightarrow{\tau} Q$ and $s \in \mathcal{D}[\![\Phi_\alpha(Q)]\!]$. In this case set $f_d(P,s) = (Q,\tau,s)$. If not then

(ii) there must exist $Q$, $a$ and $s'$ such that $s = \langle a \rangle s'$ and $P \xrightarrow{a} Q$ and $s' \in \mathcal{D}[\![Q]\!]$. In this case set $f_d(P,s) = (Q,a,s')$.

If, in either case, there is more than one choice for $Q$ an arbitrary choice is made.

If $u \in \mathcal{I}[\![\Phi_\alpha(P)]\!] = \mathcal{I}[\![\mathcal{G}(\Phi_\alpha)(P)]\!]$ then by definition of $\mathcal{G}$ one of the following two cases must hold according to which of the clauses of the infinite traces component of $\mathcal{G}$ applies.

(i) There may exist $Q$ such that $P \xrightarrow{\tau} Q$ and $u \in \mathcal{I}[\![\Phi_\alpha(Q)]\!]$. In this case set $f_d(P,u) = (Q,\tau,u)$. If not then

(ii) there must exist $Q$, $a$ and $u'$ such that $u = \langle a \rangle u'$ and $P \xrightarrow{a} Q$ and $u' \in \mathcal{I}[\![Q]\!]$. In this case set $f_d(P,u) = (Q,a,u')$.

If, in either case, there is more than one choice for $Q$ an arbitrary choice is made.

These rather cumbersome functions have been defined in such a way that we can deterministically define, in each case, an infinite or finite (the latter only possible in the case of failures) sequence of processes and actions demonstrating membership of the appropriate component of $\Phi(P)$. For example, given $(s,X) \in \mathcal{F}[\![\Phi_\alpha(P)]\!]$ set $P_0 = P$ and $s_0 = s$. If ever $f_f(P_n,(s_n,X)) = \Delta$ then the sequence is finished, otherwise $f_f(P_n,(s_n,X)) = (Q,x,(t,X))$ and we set $x_n = x$, $P_{n+1} = Q$ and $s_{n+1} = t$. This process may terminate, in which case the sequence constructed demonstrates that $P \stackrel{s}{\Longrightarrow} P_n \wedge P_n \; ref \; X$, or it may not, in which case all but finitely many of the $x_i$ are $\tau$ which demonstrates that there exists $s' \leq s$ and $n$ such that $P \stackrel{s'}{\Longrightarrow} P_n \wedge P_n\uparrow$. In either case $(s,X) \in \mathit{failures}(P)$, the second one being because then $s \in \mathit{divergences}(P)$. Each of the other two cases divides into two in just the same way. In those cases, as with the failures one described here, the strictness of our semantics with respect to divergence is a crucial part of the proof.

This completes the proof of Theorem 2.4. □

This result shows the equivalence of the natural operationally defined abstraction function and one which it obtained by iterating a CSP definition through the ordinals. This is exactly what we shall want to do on a much wider scale when we seek to prove the congruence theorem in the final section. It will turn out that this last result is perhaps the most important component of the proof of that theorem.

## 3. Transition system semantics for CSP

This section is devoted to the definition of the operational semantics for CSP and closely related semantics over more general transition systems.

A crucial starting point of the creation of a Plotkin-style semantics is the definition of the programming language. The definition we take is just the usual core CSP extended by unbounded nondeterminism and infinite hiding. For formal reasons we must fix *ab initio* the range of unbounded nondeterminism allowed. However this may be as large as we please. In particular, it is convenient to fix it strictly larger than the cardinality of the alphabet $\Sigma$. Thus the following language is implicitly parameterised both by the alphabet $\Sigma$ of all possible communications and by the bound $\lambda$, an infinite regular cardinal on the unbounded nondeterminism.

Because the unbounded nondeterminism operator (unavoidably) and the guarded choice operator (avoidably at a price) are infinitary operators (take a potentially infinite number of process arguments) one should, for rigour, be rather careful over the definition of the syntax of this version of CSP. On the

one hand we can write down the usual sort of BNF definition.

$$P ::= p \mid STOP \mid SKIP \mid a \rightarrow P \mid x : B \rightarrow g(x) \mid P \Box Q \mid P \sqcap Q \mid$$
$$P \;{}_B\|_C\; Q \mid P \,|||\, Q \mid P;Q \mid P \backslash B \mid f[P] \mid f^{-1}[P] \mid \mu p.P \mid \sqcap S$$

where $g$ is any function from $B$ (a subset of $\Sigma$) to processes, $S$ ranges over nonempty sets of processes smaller than $\lambda$, $f$ ranges over the set $AT$ of (not necessarily finite-to-one) alphabet transformations, $p$ over the set $Var$ of process variables, etc.

When there are infinitary operators in a syntax, like those in this language, the idea of what is defined by a syntax like this one is less obvious than it usually is and should therefore be discussed briefly. If we are to have a principle of structural induction and have a way of defining the semantics of programs we cannot have a program of the form $\sqcap S$ or $x : B \rightarrow g(x)$ which is itself in $S$ or in the range of $g$. One can, of course, regard BNF definitions like the above as fixed point equations, defining the smallest syntactic class which is closed under the various operations on the right. For a language with only finitary constructs this fixed point is reached by $\omega$ iterations (every program is "born on a finite day") but we have to go further, to cater for programs like $n : \mathbb{N} \rightarrow P_n$ where $P_n$ is born on day $n$. The functional implied by the right hand side of the above BNF definition is clearly monotone (the more programs there are, the more it delivers) but since it is not operating over a set (rather over the proper Class of all syntactic objects) it is by no means obvious it even has a fixed point. Fortunately it does, and is guaranteed to reach it by $\lambda$ iterations, where $\lambda$ is the bound on nondeterminism and the size of $\Sigma$ already mentioned. (See [BRW] for some more discussion of this question.) The principle of structural induction is then perfectly valid and corresponds to the principle of transfinite induction on the "birthday" of a term.

To simplify the operational semantics a little it is convenient, as was done in [BRW], to treat the constructs $STOP$, $SKIP$ and $a \rightarrow P$ as special cases of the construct $x : B \rightarrow g(x)$: $STOP$ has $B$ empty, $a \rightarrow P$ has $B = \{a\}$ and $g(a) = P$, and $SKIP = \surd \rightarrow STOP$.

Let $\mathbf{E}$ be the set of all CSP terms defined by the above. An element of $\mathbf{E}$ may have free process variables, in which case it is said to be *open*. If it has none it is said to be *closed*; we denote the set of all closed terms by $\mathbf{P}$. Closed terms are of importance since their meaning is fully determined; there are no slots for processes waiting to be filled in.

If $P, Q \in \mathbf{E}$ and $p \in Var$ then $P[Q/p]$ denotes the term where $Q$ has been substituted for all free occurrences of $p$ in $P$. When $Q$ is not closed (though for us it usually will be) some care will be necessary to prevent $P$ binding any of $Q$'s free variables.

The Plotkin-style semantics regards the set $\mathbf{P}$ of all closed CSP-terms as a transition system, since it describes the set of all actions each closed term

can perform and which new terms it may then become. The clauses of this operational semantics are given in the usual "natural deduction" style below.

Below, $a, b$ range over $\Sigma$ and $x, y$ over $\Sigma^+ = \Sigma \cup \{\tau\}$. Alphabet transformations (functions from $\Sigma$ to $\Sigma$) are extended to $\Sigma^+$ by setting $f(\tau) = \tau$.

$$\frac{}{(x : B \to g(x)) \xrightarrow{b} g(b)} \quad (b \in B)$$

$$\frac{}{P \sqcap Q \xrightarrow{\tau} P} \qquad \frac{}{P \sqcap Q \xrightarrow{\tau} Q}$$

$$\frac{}{\mu p.P \xrightarrow{\tau} P[\mu p.P/p]}$$

$$\frac{P \xrightarrow{\tau} P'}{P \Box Q \xrightarrow{\tau} P' \Box Q} \qquad \frac{Q \xrightarrow{\tau} Q'}{P \Box Q \xrightarrow{\tau} P \Box Q'}$$

$$\frac{P \xrightarrow{a} P'}{P \Box Q \xrightarrow{a} P'} \qquad \frac{Q \xrightarrow{a} Q'}{P \Box Q \xrightarrow{a} Q'}$$

$$\frac{P \xrightarrow{\tau} P'}{P \,{}_B\|_C\, Q \xrightarrow{\tau} P' \,{}_B\|_C\, Q} \qquad \frac{Q \xrightarrow{\tau} Q'}{P \,{}_B\|_C\, Q \xrightarrow{\tau} P \,{}_B\|_C\, Q'}$$

$$\frac{P \xrightarrow{a} P'}{P \,{}_B\|_C\, Q \xrightarrow{a} P' \,{}_B\|_C\, Q} \quad (a \in B - C)$$

$$\frac{Q \xrightarrow{a} Q'}{P \,{}_B\|_C\, Q \xrightarrow{a} P \,{}_B\|_C\, Q'} \quad (a \in C - B)$$

$$\frac{P \xrightarrow{a} P' \quad Q \xrightarrow{a} Q'}{P \,{}_B\|_C\, Q \xrightarrow{a} P' \,{}_B\|_C\, Q'} \quad (a \in B \cap C)$$

$$\frac{P \xrightarrow{x} P'}{P \,|||\, Q \xrightarrow{x} P' \,|||\, Q} \qquad \frac{Q \xrightarrow{x} Q'}{P \,|||\, Q \xrightarrow{x} P \,|||\, Q'}$$

$$\frac{P \xrightarrow{x} P'}{P; Q \xrightarrow{x} P'; Q} \quad (x \neq \surd)$$

$$\frac{\exists P'.\, P \xrightarrow{\surd} P'}{P; Q \xrightarrow{\tau} Q}$$

$$\frac{P \xrightarrow{x} P'}{P \backslash B \xrightarrow{x} P' \backslash B} \quad (x \notin B)$$

$$\frac{P \xrightarrow{a} P'}{P \backslash B \xrightarrow{\tau} P' \backslash B} \quad (a \in B)$$

$$\frac{P \xrightarrow{x} P'}{f[P] \xrightarrow{y} f[P']} \quad (y = f(x))$$

$$\frac{P \xrightarrow{x} P'}{f^{-1}[P] \xrightarrow{y} f^{-1}[P']} \quad (f(y) = x)$$

$$\frac{P \in S}{\sqcap S \xrightarrow{\tau} P}$$

Note at this point that the operationally natural element of $\mathcal{U}$ corresponding to each closed term $P$ is given by $\Phi(P)$, where $\Phi$ is as defined in Section 2 and $P$ is considered to be an element of the transition system $\mathbf{P}$ defined above. Theorem 2.2 shows that this is equal to $\Phi(F(P))$ for any morphism $F$. We can now state the main congruence result that we would like to prove, namely that for all closed CSP terms $P$, $\Phi(P) = \mathcal{S}[\![P]\!]$, where $\mathcal{S}[\![P]\!]$ denotes the value in $\mathcal{U}$ defined by the semantics defined earlier (though we should remember that we still have an obligation to show the existence of fixed points).

There are two structure clashes between the operational and denotational semantics. The first is the obvious one that one is given in terms of transition systems and the other in terms of the abstract model $\mathcal{U}$. But perhaps the more difficult one to resolve is the clash between the term rewriting style of the operational semantics and the denotational style of the other. Of course the latter means that the semantic value of each term is deduced from the semantic value of its subcomponents in a transparent way and that an abstract fixed point theory is used. In the earlier paper on the operational semantics of CSP [BRW] these two issues were resolved separately by creating an intermediate, denotational tree semantics. Unfortunately the complete metric spaces of trees used in that paper no longer exist because of the introduction here of infinite branching.

The main result of [R3] is that, for each infinite regular cardinal $\lambda$, there exists a transition system $T_\lambda$ such that for all transition systems $C$ with $i(C) \leq \lambda$, there exists a unique morphism $H_\lambda : C \to T_\lambda$. Thus $T_\lambda$ is a final object in the category of transition systems with morphisms as arrows. Analogues of the contraction mapping theorem and related results hold which are useful when one uses these systems. $T_\lambda$ can be used to give an intermediate denotational semantics to CSP in the style of [BRW]. However, because of the complexity of this new theory and thanks mainly to the construction of the $\Phi_\alpha$ in the previous section we do not now need to do so.

It is useful to extend the operational space defined above to include non-closed terms with their variables instantiated by elements of an arbitrary transition system.

**Definition.** If $C$ is any transition system then $C^{CSP}$ is the system of CSP syntactic terms over $C$: namely the set of all substitutions by elements of $C$ for all free variables of general terms in the language. All terms are distinct. Note that $C^{CSP}$ contains every closed CSP term and every element of $C$. The transitions of each term are those of $P$ if $P \in C$ (i.e., $P \xrightarrow{\delta} Q$ in $C^{CSP}$ if and only if $P \xrightarrow{\delta} Q$ in $C$). The transitions of proper syntactic terms are determined

from the operational semantic clauses above (from those of their subterms or otherwise).

The stipulation that all terms are distinct means that each possible construction of a term leads to a different element of the system. For example, in $(C^{CSP})^{CSP}$, for each $P \in C$ the terms $a \to a \to \ulcorner P \urcorner$, $a \to \ulcorner a \to P \urcorner$ and $\ulcorner a \to a \to P \urcorner$ are all different, where the syntactic quotes $\ulcorner \cdot \urcorner$ denote the boundary between the inner and outer syntactic construction. However the obvious map from $(C^{CSP})^{CSP}$ to $C^{CSP}$ which "forgets" these boundaries is easily shown to be a morphism.

Note that Theorem 2.2 (a) tells us that the image under $\Phi$ of a closed term $P$ is independent of whether it is considered to belong to the space $\mathbf{P}$ of closed terms or any $C^{CSP}$, since there is an obvious morphism embedding $\mathbf{P}$ into any $C^{CSP}$.

We are now in a position to begin the proof of the main theorem, namely that the $\mathcal{U}$ semantics for CSP is well defined and congruent to the operational semantics. We will eventually complete the proof by performing a structural induction over $C^{CSP}$, but before we do that it is helpful to prove the operational and denotational versions of all the non-recursive operators congruent.

**Theorem 3.1.** The operational versions of the various CSP operators are all congruent to the denotational versions over $\mathcal{U}$. In other words, for each operator $\odot$ and each $P, Q \in C^{CSP}$,

$$\Phi(P \odot Q) = \Phi(P) \odot \Phi(Q) .$$

Furthermore all the operators are well behaved with respect to the partial abstraction functions $\Phi_\alpha$ in the sense that

$$\Phi_\alpha(P \odot Q) \leq \Phi_\alpha(P) \odot \Phi_\alpha(Q)$$

for each $\alpha$. (The form of these clauses is modified suitably when the operator $\odot$ is not binary. The precise statement for each operator in turn can be found in the Lemmas below.)

**Proof.** This theorem is no more nor less than a convenient grouping of a large number of similar though separate results. These are stated below, grouped by operator, plus for each operator a further result which is crucial in the proof of the full congruence part of the Lemma. In each of these Lemmas it is assumed that the given term is an element of $C^{CSP}$ of the given form; the immediate subterms being unrestricted elements of $C^{CSP}$ (i.e., not necessarily elements of $C$ itself).

The operators break into two classes as far as style of proof is concerned: prefixing and nondeterministic choice, which are easiest, and the rest of the operators, which require very similar though more difficult arguments. As usual, recursion is a special case and will be dealt with on its own later. All the Lemmas are stated below but only a few sample proofs are given.

**Lemma 3.1.1 (a).**

(1) $(x : A \to P_x) \stackrel{\langle\rangle}{\Longrightarrow} Q$ iff $Q = (x : A \to P_x)$.

(2) If $s = \langle a \rangle s'$ then $(x : A \to P_x) \stackrel{s}{\Longrightarrow} Q$ iff $a \in A$ and $P_a \stackrel{s'}{\Longrightarrow} Q$.

(3) $(x : A \to P_x)$ *ref* $B$ iff $A \cap B = \emptyset$.

(4) $\neg((x : A \to P_x)\uparrow)$.

(5) If $u = \langle a \rangle u'$ then $(x : A \to P_x) \stackrel{u}{\Longrightarrow}$ iff $a \in A$ and $P_a \stackrel{u'}{\Longrightarrow}$.

**Proof.** These clauses all follow straightforwardly from the operational semantics. □

**Lemma 3.1.1 (b).** For all terms $P_x$ denoting functions from $A$ into $C^{CSP}$, we have

$$\Phi(x : A \to P_x) = x : A \to \Phi(P_x).$$

**Proof.** This is an easy consequence of part (a) above and the definitions of $\Phi$ and the prefixing operator over $\mathcal{U}$. □

**Lemma 3.1.1 (c).** For all terms $P_x$ denoting functions from $A$ to $C^{CSP}$ and all ordinals $\alpha$ we have

$$\Phi_\alpha(x : A \to P_x) \leq x : A \to \Phi_\alpha(P_x).$$

**Proof.** This is proved by transfinite induction on $\alpha$. When $\alpha = 0$ it is trivial, for the left hand side equals $\perp$. If it is true for all $\beta$ less than some limit ordinal $\alpha$ then

$$x : A \to \Phi_\alpha(P_x) \geq \Phi_\beta(x : A \to P_x)$$

for all $\beta < \alpha$ as $\Phi_\gamma(P)$ increases with $\gamma$ and prefixing is monotone. Hence

$$x : A \to \Phi_\alpha(P_x) \geq \bigsqcup\{\Phi_\beta(x : A \to P_x) \mid \beta < \alpha\} = \Phi_\alpha(x : A \to P_x)$$

by properties of least upper bounds and definition of $\Phi_\alpha$.

It only remains to treat the successor ordinal case. Recall that $\Phi_{\beta+1} = \mathcal{G}(\Phi_\beta)$, which is to say that

- if $P$ is stable (has no internal transitions) then

  $$\Phi_{\beta+1}(P) = x : P^0 \to \prod\{\Phi_\beta(P') \mid P \stackrel{x}{\longrightarrow} P'\}$$

  where $P^0$ denotes the initial external transitions possible for $P$, and

- if $P$ is not stable then

  $$\Phi_{\beta+1}(P) = (x : P^0 \to \prod\{\Phi_\beta(P') \mid P \stackrel{x}{\longrightarrow} P'\}) \rhd \prod\{\Phi_\beta(P') \mid P \stackrel{\tau}{\longrightarrow} P'\}$$

  where $P \rhd Q = (P \,\Box\, Q) \sqcap Q$.

The definition of the transitions of $x : A \to P_x$ means that

$$\Phi_{\alpha+1}(x : A \to P_x) = x : A \to \Phi_\alpha(P_x)$$

which, since in general $\Phi_\gamma(Q)$ increases with $\gamma$ and prefixing is monotone, is less than $x : A \to \Phi_{\alpha+1}(P_x)$ as required. (This was a curious induction, since the inductive assumption was only used in the limit ordinal case. In fact it can be eliminated from that as well, but this proof will serve as a model for later ones where induction is really needed.) □

**Lemma 3.1.2 (a).**

(1) $\sqcap S \stackrel{\leq\geq}{\Longrightarrow} R$ iff $R = \sqcap S$ or $\exists P \in S.P \stackrel{\leq\geq}{\Longrightarrow} R$.

(2) If $s \neq <>$ then $\sqcap S \stackrel{s}{\Longrightarrow} R$ iff $\exists P \in S.P \stackrel{s}{\Longrightarrow} R$.

(3) $\neg(\sqcap S \; ref \; B)$ (as it is not a stable process).

(4) $(\sqcap S)\uparrow$ iff $\exists P \in S.P\uparrow$.

(5) $\sqcap S \stackrel{u}{\Longrightarrow}$ iff $\exists P \in S.P \stackrel{u}{\Longrightarrow}$.

**Proof.** These clauses all follow straightforwardly from the operational semantics. □

**Lemma 3.1.2 (b).** For all $S \subseteq C^{CSP}$ (of size less than our bound on nondeterminism) we have

$$\Phi(\sqcap S) = \sqcap\{\Phi(P) \mid P \in S\}.$$

**Proof.** This is an easy consequence of part (a) above and the definitions of $\Phi$ and that of $\sqcap$ over $\mathcal{U}$. □

**Lemma 3.1.2 (c).** For all $P, Q$ in $C^{CSP}$ and all ordinals $\alpha$ we have

$$\Phi_\alpha(\sqcap S) \leq \sqcap\{\Phi_\alpha(P) \mid P \in S\}.$$

**Proof.** This is very similar to that for prefixing. The reader may notice that, so far as the structure of their operational semantics is concerned, prefixing and nondeterminiistic choice have much in common. □

Note that, since binary nondeterministic choice is a special case of the general variety, corresponding results hold for $\sqcap$ as well.

**Lemma 3.1.3 (a).**

(1) $P \Box Q \stackrel{\leq\geq}{\Longrightarrow} U$ iff $\exists P', Q'. P \stackrel{\leq\geq}{\Longrightarrow} P'$, $Q \stackrel{\leq\geq}{\Longrightarrow} Q'$ and $U = P' \Box Q'$.

(2) If $s \neq <>$ then $P \Box Q \stackrel{s}{\Longrightarrow} U$ iff $P \stackrel{s}{\Longrightarrow} U$ or $Q \stackrel{s}{\Longrightarrow} U$.

(3) $P \Box Q \; ref \; B$ iff $P \; ref \; B$ and $Q \; ref \; B$.

(4) $P \Box Q \uparrow$ iff $P \uparrow$ or $Q \uparrow$.

(5) $P \Box Q \stackrel{u}{\Longrightarrow}$ iff $P \stackrel{u}{\Longrightarrow}$ or $Q \stackrel{u}{\Longrightarrow}$.

**Proof.** This follows directly from the definition of the transition relation over $C^{CSP}$. Recall that a term of the form $P \Box Q$ derives its transitions from those of $P$ and $Q$ using the appropriate rules of the operational semantics. (3) above follows since these rules mean that $P \Box Q$ *ref* $B$ if and only if (a) $P$ and $Q$ are both stable and (b) neither $P$ nor $Q$ can perform any action in $B$. All the others follow from the observation that, if $U_0 = P \Box Q$, then $U_i \xrightarrow{x_i} U_{i+1}$ for all $i < \alpha$ ($\alpha \in \omega + 1$) if and only if one of the following applies.

(i) All $x_i$ equal $\tau$ and for each $j$ there exist $P_j$ and $Q_j$ such that, for all $i$, $U_i = P_i \Box Q_i$ and either $P_i = P_{i+1}$ and $Q_i \xrightarrow{x_i} Q_{i+1}$ or $P_i \xrightarrow{x_i} P_{i+1}$ and $Q_i = Q_{i+1}$.

(ii) Not all $x_i$ equal $\tau$ (and $k$ is minimal such that $x_k \neq \tau$), the $U_i$ ($i \leq k$) and $x_i$ ($i < k$) satisfy (i). If $U_k = P_k \Box Q_k$ then $P_k \xrightarrow{x_k} U_{k+1}$ or $Q_k \xrightarrow{x_k} U_{k+1}$. Subsequent transitions are possible for $U_{k+1}$ is $C^{CSP}$.

(1) comes from the case where (i) above applies to a finite sequence of processes and actions, (2) from the case where (ii) applies to a finite sequence, (4) from the case where (i) applies to an infinite sequence and (5) from (ii) applied to an infinite sequence.

**Lemma 3.1.3 (b)** If $P, Q \in C^{CSP}$ then $\Phi(P \Box Q) = \Phi(P) \Box \Phi(Q)$.

**Proof.** This follows more or less immediately from the definition of $\Phi$ and Lemma 3.1.3 (a) above.

**Lemma 3.1.3 (c)** If $P, Q \in C^{CSP}$ and $\alpha$ is any ordinal then $\Phi_\alpha(P \Box Q) \leq \Phi_\alpha(P) \Box \Phi_\alpha(Q)$ (where $\Phi_\alpha$ is as defined in Section 2).

**Proof.** This is a transfinite induction on $\alpha$. The $\alpha = 0$ and limit ordinal cases are the same as with prefixing above, the latter following by monotonicity of $\Box$.

It only remains to prove the result for successor ordinals $\alpha = \beta + 1$. The proof of this clause for all standard operators other than prefixing and nondeterministic choice follows from laws (all theorems of the denotational semantics) which show how processes in each of the two forms produced by the definition of $\mathcal{G}$ combine under the operators in question to produce one of these forms in a way more or less directly analogous to how the operational semantics works. Several laws are required for each operator because of the different cases that arise. In the case of $\Box$ there are three, the first for the case where both arguments are stable, the second for the one where one of the arguments is stable and one unstable, and one for two unstable arguments.

$\Box.1 \quad (x : A \to P_x) \Box (x : B \to Q_x) = x : A \cup B \to R_x,$

$$\text{where } R_x = \begin{cases} P_x & \text{if } x \in A \setminus B \\ Q_x & \text{if } x \in B \setminus A \\ Q_x \sqcap P_x & \text{if } x \in B \cap A \end{cases}$$

$\Box.2$ If $Q = x : B \to Q_x$, then

$$((x : A \to P_x) \rhd P')\Box Q = (x : A \cup B \to R_x) \rhd (P'\Box Q),$$

$$\text{where } R_x = \begin{cases} P_x & \text{if } x \in A \setminus B \\ Q_x & \text{if } x \in B \setminus A \\ Q_x \sqcap P_x & \text{if } x \in B \cap A \end{cases}$$

$\Box.3$ If $P = (x : A \to P_x) \rhd P'$ and $Q = (x : B \to Q_x) \rhd Q'$, then

$$P \Box Q = (x : A \cup B \to R_x) \rhd ((P'\Box Q) \sqcap (P \Box Q'),$$

$$\text{where } R_x = \begin{cases} P_x & \text{if } x \in A \setminus B \\ Q_x & \text{if } x \in B \setminus A \\ Q_x \sqcap P_x & \text{if } x \in B \cap A \end{cases}$$

These laws show that, in any combination of stable and unstable processes, the way in which the operational semantics of $P \Box Q$ are "composed" from those of $P$ and $Q$ is reflected precisely in the abstract semantics. For example consider the case where both $P$ and $Q$ are unstable elements of $C^{CSP}$. Then the operational semantics allows us to deduce that $P \Box Q$ is also unstable and that

$$\begin{aligned}
& \Phi_{\beta+1}(P \Box Q) \\
= \; & (x : P^0 \cup Q^0 \to \textstyle\prod\{\Phi_\beta(R) \mid P \xrightarrow{x} R \vee Q \xrightarrow{x} R\}) \\
& \rhd \textstyle\prod(\{\Phi_\beta(P'\Box Q) \mid P \xrightarrow{\tau} P'\} \cup \{\Phi_\beta(P \Box Q') \mid Q \xrightarrow{\tau} Q'\}) && (1) \\
\leq \; & (x : P^0 \cup Q^0 \to \textstyle\prod\{\Phi_\beta(R) \mid P \xrightarrow{x} R \vee Q \xrightarrow{x} R\}) \\
& \rhd \textstyle\prod(\{\Phi_\beta(P')\Box \Phi_\beta(Q) \mid P \xrightarrow{\tau} P'\} \cup \{\Phi_\beta(P)\Box \Phi_\beta(Q') \mid Q \xrightarrow{\tau} Q'\}) && (2) \\
\leq \; & (x : P^0 \cup Q^0 \to \textstyle\prod\{\Phi_\beta(R) \mid P \xrightarrow{x} R \vee Q \xrightarrow{x} R\}) \\
& \rhd \textstyle\prod(\{\Phi_\beta(P')\Box \Phi_{\beta+1}(Q) \mid P \xrightarrow{\tau} P'\} \cup \{\Phi_{\beta+1}(P)\Box \Phi_\beta(Q') \mid Q \xrightarrow{\tau} Q'\}) && (3) \\
= \; & (x : P^0 \cup Q^0 \to R_x) \\
& \rhd \textstyle\prod\{\Phi_\beta(P')\Box \Phi_{\beta+1}(Q) \mid P \xrightarrow{\tau} P'\} \sqcap \prod\{\Phi_{\beta+1}(P)\Box \Phi_\beta(Q') \mid Q \xrightarrow{\tau} Q'\}) && (4)
\end{aligned}$$

$$\text{where} \quad R_x = \begin{cases} \textstyle\prod\{\Phi_\beta(P') \mid P \xrightarrow{x} P'\} & \text{if } x \in P^0 \setminus Q^0 \\ \textstyle\prod\{\Phi_\beta(Q') \mid Q \xrightarrow{x} Q'\} & \text{if } x \in Q^0 \setminus P^0 \\ \textstyle\prod\{\Phi_\beta(P') \mid P \xrightarrow{x} P'\} & \\ \sqcap \textstyle\prod\{\Phi_\beta(Q') \mid Q \xrightarrow{x} Q'\} & \text{if } x \in Q^0 \cap P^0 \end{cases}$$

$$= \; \Phi_{\beta+1}(P)\Box \Phi_{\beta+1}(Q) \qquad (5)$$

Line (1) comes by inspecting the transitions of $P \Box Q$ and from the definition of $\Phi_{\beta+1} = \mathcal{G}(\Phi_\beta)$. Line (2) comes by induction and the monotonicity of CSP operators. Line (3) comes from the fact that in general $\Phi_{\beta+1}(R) \geq \Phi_\beta(R)$ and monotonicity again. The equality of line (4) with line (3) is a consequence of the fact that if $S$ and $T$ are nonempty sets of processes then $\prod(S \cup T) = \prod S \sqcap \prod T$. The equality of lines (4) and (5) follows from the law $\Box.3$ above and the fact that $\Box$ distributes over $\prod$ so that for example

$$\prod\{\Phi_{\beta+1}(P)\Box \Phi_\beta(Q') \mid Q \xrightarrow{\tau} Q'\} = \Phi_{\beta+1}(P)\Box \prod\{\Phi_\beta(Q') \mid Q \xrightarrow{\tau} Q'\}\,.$$

This establishes the most difficult subcase of the result for $\alpha = \beta + 1$. The other two follow from $\Box.1$ and $\Box.2$ in the same way. This completes the proof of Lemma 3.1.3 (c).

**Lemma 3.1.4 (a).**

(1) $P \setminus X \stackrel{s}{\Longrightarrow} Q$ iff $\exists t. \exists P'. P \stackrel{t}{\Longrightarrow} P' \wedge R = P' \setminus X$ and $s = t \setminus X$.

(2) $P \setminus X$ *ref* $B$ iff $P$ *ref* $B \cup X$.

(3) $P \setminus X \uparrow$ iff $\exists s \in X^*. \exists P'. P \stackrel{s}{\Longrightarrow} P' \wedge P' \uparrow$ or $\exists u \in X^\omega. P \stackrel{u}{\Longrightarrow}$.

(4) $P \setminus X \stackrel{u}{\Longrightarrow}$ iff $\exists u' \in X^\omega. P \stackrel{u'}{\Longrightarrow} \wedge\, u' \setminus X = u$.

**Proof.** The proof is straightforward and is omitted. (There is a characterisation of the possible execution sequences of $P \setminus X$ similar in style to that for $\Box$.) $\Box$

**Lemma 3.1.4 (b).** If $P \in C^{CSP}$ then $\Phi(P \setminus X) = (\Phi(P)) \setminus X$.

**Proof.** This is a straightforward consequence of the part (a) above. (N.B. This particular result is much simpler over this model than over the original failures model [BRW] because divergences are now inferred from single infinite traces rather than infinite sets of finite ones.) $\Box$

**Lemma 3.1.4 (c).** If $P \in C^{CSP}$ and $\alpha$ is any ordinal then $\Phi_\alpha(P \setminus X) \leq \Phi_\alpha(P) \setminus X$.

**Proof.** The central component in this proof is again some laws which show that the denotational semantics reflects the structure of the operational semantics.

$\setminus X.1$ If $A \cap X = \emptyset$ then

$$(x : A \to P_x) \setminus X = x : A \to (P_x \setminus X)$$

$\setminus X.2$ If $A \cap X \neq \emptyset$ then

$$(x : A \to P_x) \setminus X = (x : A \setminus X \to P_x \setminus X) \triangleright \sqcap \{P_x \setminus X \mid x \in A \cap X\}$$

$\setminus X.3$

$$\begin{aligned}((x : A \to P_x) \triangleright P') \setminus X \;=\; & (x : A \setminus X \to P_x \setminus X) \\ & \triangleright \sqcap (\{P' \setminus X\} \cup \{P_x \setminus X \mid x \in A \cap X\})\end{aligned}$$

The proof itself is once again a transfinite induction on $\alpha$. The $\alpha = 0$ and limit ordinal cases are practically the same as for the other operators we have seen. The derivation of the $\alpha = \beta + 1$ case breaks down into cases depending on which of $P$ and $P \setminus X$ are stable (i.e., on which of the laws above applies). We

will omit the proof of the easiest case (both stable). The two others (where $P \setminus X$ is unstable and $P$ either is stable or not) can be covered by a single argument.

$$
\begin{aligned}
\Phi_{\beta+1}(P \setminus X) &= (x : P^0 \setminus X \to \sqcap\{\Phi_\beta(P' \setminus X) \mid P \xrightarrow{x} P'\}) \\
&\quad \triangleright \sqcap\{\Phi_\beta(P' \setminus X) \mid \exists x \in X \cup \{\tau\}.P \xrightarrow{x} P'\} && (1) \\
&\leq (x : P^0 \setminus X \to \sqcap\{\Phi_\beta(P') \setminus X \mid P \xrightarrow{x} P'\}) \\
&\quad \triangleright \sqcap\{(\Phi_\beta(P')) \setminus X \mid \exists x \in X \cup \{\tau\}.P \xrightarrow{x} P'\} && (2) \\
&= (x : P^0 \setminus X \to \sqcap\{\Phi_\beta(P') \setminus X \mid P \xrightarrow{x} P'\}) \\
&\quad \triangleright \sqcap(\{\sqcap\{(\Phi_\beta(P')) \setminus X \mid P \xrightarrow{\tau} P'\} \mid P \text{ is not stable}\} \\
&\quad \cup\{\sqcap\{(\Phi_\beta(P')) \setminus X \mid P \xrightarrow{x} P'\} \mid x \in X \cap P^0\}) && (3) \\
&= (x : P^0 \setminus X \to (\sqcap\{\Phi_\beta(P') \mid P \xrightarrow{x} P'\}) \setminus X) \\
&\quad \triangleright \sqcap(\{(\sqcap\{\Phi_\beta(P') \mid P \xrightarrow{\tau} P'\}) \setminus X \mid P \text{ is not stable}\} \\
&\quad \cup\{(\sqcap\{\Phi_\beta(P') \mid P \xrightarrow{x} P'\}) \setminus X \mid x \in X \cap P^0\}) && (4) \\
&= \Phi_{\beta+1}(P \setminus X) && (5)
\end{aligned}
$$

Line (1) comes by inspecting the transitions of $P \setminus X$ and from the definition of $\Phi_{\beta+1} = \mathcal{G}(\Phi_\beta)$. Line (2) comes by induction and monotonicity. Line (3) is equal to line (2) by associative properties of $\sqcap$. The set to which the outermost $\sqcap$ is applied after $\triangleright$ is guaranteed to be nonempty since we have assumed $P \setminus X$ to be unstable, though either of its two components may be empty. Lines (3) and (4) are equal by distributive properties of $\setminus X$. Lines (4) and (5) are equal by law $\setminus X.2$ if $P$ is stable and by $\setminus X.3$ if $P$ is not. (In either case the law proves the equality of $\mathcal{G}(\Phi_\beta)(P)$ and line (4).) This completes the proof of Lemma 3.1.4 (c). □

Lemmas for the rest of the operators are stated below. In each case part (a) is the major part of the proof of part (b). For part (c) only the laws required for the successor ordinal case are stated. The proof then follows, as in the cases of $\Box$ and hiding, from these laws, the monotonicity and distributivity of the operator and the fact that one step of the behaviour of the operator never requires knowledge of more than one step of the operand(s).

**Lemma 3.1.5 (a).**

(1) $P \,{}_B|\!|_C\, Q \xRightarrow{s} R$ iff $s \in (B \cup C)^*$ and $\exists P', Q'.P \xRightarrow{s'} P' \wedge Q \xRightarrow{s''} Q' \wedge R = P' \,{}_B|\!|_C\, Q'$, where $s' = s\restriction B$ and $s'' = s\restriction C$.

(2) $P \,{}_B|\!|_C\, Q \; ref \; X$ iff $P \; ref \; X \cap B \wedge Q \; ref \; X \cap C$.

(3) $P \,{}_B|\!|_C\, Q\uparrow$ iff $P\uparrow$ or $Q\uparrow$.

(4) $P \,{}_B|\!|_C\, Q \xRightarrow{u}$ iff $u \in (B \cup C)^\omega$ and $P \xRightarrow{u'} \wedge Q \xRightarrow{u''}$, where $u' = u\restriction B$ and $u'' = u\restriction C$. (N.B. One of $u'$ and $u''$ may be finite.)

**Lemma 3.1.5** (b). If $P, Q \in C^{CSP}$ then $\Phi(P \,{}_B|\!|_C\, Q) = \Phi(P) \,{}_B|\!|_C\, \Phi(Q)$.

**Lemma 3.1.5 (c).** If $P, Q \in C^{CSP}$ and $\alpha$ is any ordinal then

$$\Phi_\alpha(P \,{}_B\|_C\, Q) \leq \Phi_\alpha(P) \,{}_B\|_C\, \Phi_\alpha(Q)\,.$$

**Laws.**

$\|.1$ If $P = (x : A \to P_x)$ and $Q = (a : A' \to Q_x)$ then

$$P \,{}_B\|_C\, Q = x : A'' \to R_x$$

$\|.2$ If $P = (x : A \to P_x) \rhd P'$ and $Q = (a : A' \to Q_x)$ then

$$P \,{}_B\|_C\, Q = (x : A'' \to R_x) \rhd (P' \,{}_B\|_C\, Q)$$

$\|.3$ If $P = (x : A \to P_x) \rhd P'$ and $Q = (a : A' \to Q_x) \rhd Q'$ then

$$P \,{}_B\|_C\, Q = (x : A'' \to R_x) \rhd ((P' \,{}_B\|_C\, Q) \sqcap (P \,{}_B\|_C\, Q'))$$

where in each case $A'' = (A \cap (B \setminus C)) \cup (A' \cap (C \setminus B)) \cup (A \cap A' \cap B \cap C)$ and

$$R_x = \begin{cases} P_x \,{}_B\|_C\, Q & \text{if } x \in A \cap (B \setminus C) \\ P \,{}_B\|_C\, Q_x & \text{if } x \in A' \cap (C \setminus B) \\ P_x \,{}_B\|_C\, Q_x & \text{if } x \in A \cap A' \cap B \cap C \end{cases}$$

**Lemma 3.1.6 (a).**

(1) $P \,|||\, Q \stackrel{s}{\Longrightarrow} R$ iff $\exists s', s'', P', Q'. P \stackrel{s'}{\Longrightarrow} P' \wedge Q \stackrel{s''}{\Longrightarrow} Q' \wedge R = P' \,|||\, Q' \wedge s \in merge\langle s', s''\rangle$.

(2) $P \,|||\, Q \; ref\; X$ iff $P \; ref\; X \wedge Q \; ref\; X$.

(3) $P \,|||\, Q\!\uparrow$ iff $P\!\uparrow$ or $Q\!\uparrow$.

(4) $P \,|||\, Q \stackrel{u}{\Longrightarrow}$ iff $\exists u', u''. P \stackrel{u'}{\Longrightarrow} \wedge Q \stackrel{u''}{\Longrightarrow} \wedge u \in merge\langle u', u''\rangle$. (One of $u'$ and $u''$ may be finite.)

**Lemma 3.1.6 (b).** If $P, Q \in C^{CSP}$ then $\Phi(P \,|||\, Q) = \Phi(P) \,|||\, \Phi(Q)$.

**Lemma 3.1.6 (c).** If $P, Q \in C^{CSP}$ and $\alpha$ is any ordinal then

$$\Phi_\alpha(P \,|||\, Q) \leq \Phi_\alpha(P) \,|||\, \Phi_\alpha(Q)\,.$$

**Laws.**

$|||.1$ If $P = (x : A \to P_x)$ and $Q = (a : A' \to Q_x)$ then

$$P \,|||\, Q = x : A \cup A' \to R_x$$

$|||.2$ If $P = (x : A \to P_x) \rhd P'$ and $Q = (a : A' \to Q_x)$ then

$$P \,|||\, Q = (x : A \cup A' \to R_x) \rhd (P' \,|||\, Q)$$

$|||.3$ If $P = (x : A \to P_x) \rhd P'$ and $Q = (a : A' \to Q_x) \rhd Q'$ then

$$P \,|||\, Q = (x : A \cup A' \to R_x) \rhd ((P' \,|||\, Q) \sqcap (P \,|||\, Q'))$$

where in each case

$$R_x = \begin{cases} P_x \,|||\, Q & \text{if } x \in A \setminus A' \\ P \,|||\, Q_x & \text{if } x \in A' \setminus A \\ (P_x \,|||\, Q) \sqcap (P \,|||\, Q_x) & \text{if } x \in A \cap A' \end{cases}$$

**Lemma 3.1.7 (a).**

(1) $P;Q \stackrel{s}{\Longrightarrow} R$ iff $\exists P'.\, P \stackrel{s}{\Longrightarrow} P' \wedge s$ $\surd$-free $\wedge\, R = P';Q$
or $\exists s', s'', P'.s'$ $\surd$-free $\wedge\, P \stackrel{s'\langle\surd\rangle}{\Longrightarrow} P' \wedge Q \stackrel{s''}{\Longrightarrow} R \wedge s = s's''$.

(2) $P;Q$ *ref* $X$ iff $P$ *ref* $X \cup \{\surd\}$.

(3) $P;Q\uparrow$ iff $P\uparrow$ or $\exists P'.P \stackrel{\langle\surd\rangle}{\Longrightarrow} P' \wedge Q\uparrow$.

(4) $P;Q \stackrel{u}{\Longrightarrow}$ iff $P \stackrel{u}{\Longrightarrow} \wedge u$ $\surd$-free or
$\exists s, u', P'.P \stackrel{s\langle\surd\rangle}{\Longrightarrow} P' \wedge s$ $\surd$-free $\wedge\, Q \stackrel{u'}{\Longrightarrow} \wedge u = su'$.

**Lemma 3.1.7 (b).** If $P, Q \in C^{CSP}$ then $\Phi(P;Q) = \Phi(P);\Phi(Q)$.

**Lemma 3.1.7 (c).** If $P, Q \in C^{CSP}$ and $\alpha$ is any ordinal then

$$\Phi_\alpha(P;Q) \leq \Phi_\alpha(P);\Phi_\alpha(Q)\,.$$

**Laws.**

$;.1$ If $P = (x : A \to P_x)$ and $\surd \notin A$ then

$$P;Q = x : A \to P_x;Q$$

$;.2$ If $P = (x : A \to P_x)$ and $\surd \in A$ then

$$P;Q = (x : A \setminus \{\surd\} \to P_x;Q) \rhd Q$$

$;.3$ If $P = (x : A \to P_x) \rhd P'$ and $\surd \notin A$ then

$$P;Q = (x : A \to P_x;Q) \rhd (P';Q)$$

;.4 If $P = (x : A \to P_x) \rhd P'$ and $\surd \in A$ then

$$P;Q = (x : A \setminus \{\surd\} \to P_x;Q) \rhd (Q \sqcap (P';Q))$$

**Lemma 3.1.8 (a).**

(1) $f[P] \stackrel{s}{\Longrightarrow} R$ iff $\exists P', s'.P \stackrel{s'}{\Longrightarrow} P' \wedge R = f[P'] \wedge s = f(s')$ .

(2) $f[P]\, ref\, X$ iff $P\, ref\, f^{-1}(X)$.

(3) $f[P]\uparrow$ iff $P\uparrow$.

(4) $f[P] \stackrel{u}{\Longrightarrow}$ iff $\exists u'.P \stackrel{u'}{\Longrightarrow} \wedge s = f(u')$.

**Lemma 3.1.8 (b).** If $P \in C^{CSP}$ then $\Phi(f[P]) = f[\Phi(P)]$.

**Lemma 3.1.8 (c).** If $P \in C^{CSP}$ and $\alpha$ is any ordinal then

$$\Phi_\alpha(f[P]) \leq f[\Phi_\alpha(P)]\,.$$

**Laws.**

$f[].1$ If $P = (x : A \to P_x)$ then

$$f[P] = x : f(A) \to f[P_x]$$

$f[].2$ If $P = (x : A \to P_x) \rhd P'$ then

$$f[P] = (x : f(A) \to f[P_x]) \rhd f[P']$$

**Lemma 3.1.9 (a).**

(1) $f^{-1}[P] \stackrel{s}{\Longrightarrow} R$ iff $\exists P'.P \stackrel{f(s)}{\Longrightarrow} P' \wedge R = f^{-1}[P']$ .

(2) $f^{-1}[P]\, ref\, X$ iff $P\, ref\, f(X)$.

(3) $f^{-1}[P]\uparrow$ iff $P\uparrow$.

(4) $f^{-1}[P] \stackrel{u}{\Longrightarrow}$ iff $P \stackrel{f(u)}{\Longrightarrow}$.

**Lemma 3.1.9 (b).** If $P \in C^{CSP}$ then $\Phi(f^{-1}[P]) = f^{-1}[\Phi(P)]$.

**Lemma 3.1.9 (c).** If $P \in C^{CSP}$ and $\alpha$ is any ordinal then

$$\Phi_\alpha(f^{-1}[P]) \leq f^{-1}[\Phi_\alpha(P)]\,.$$

**Laws.**

$f^{-1}[].1$ If $P = (x : A \to P_x)$ then

$$f^{-1}[P] = x : f^{-1}(A) \to f^{-1}[P_x]$$

$f^{-1}[].2$ If $P = (x : A \to P_x) \rhd P'$ then

$$f^{-1}[P] = (x : f^{-1}(A) \to f^{-1}[P_x]) \rhd f^{-1}[P']$$

This completes the proof of Theorem 3.1. □

These results provide the building blocks of the proof of the main result, and are put together below. The next Theorem is the main result of the paper. Notice how the well definedness of the denotational semantics, the congruence theorem and the result about the $\Phi_\alpha$ are proved by a simultaneous structural induction.

**Definitions.** Given a CSP term $P$ and a $\rho \in OEnv = Var \to C^{CSP}$, we can define an operational "semantic function": $\mathcal{O}[\![P]\!]\rho \in C^{CSP}$ is defined to be the result of substituting each free variable $p$ in $P$ by $\rho(p)$. (Note that $P$ may have no free variables, finitely many, or infinitely many. This last possibility arises because of the two infinitary operations $\sqcap$ and $x : A \to P_x$.) Given $\rho \in OEnv$ we can define the corresponding element $\overline{\rho}$ of $UEnv = Var \to \mathcal{U}$ by

$$\overline{\rho}[\![p]\!] = \Phi(\rho(p))$$

and also, for each $\alpha$, an approximation

$$\overline{\rho}^\alpha[\![p]\!] = \Phi_\alpha(\rho(p)) .$$

In this theorem we will assume that the basic transition system $C$ is such that $\Phi : C \to \mathcal{U}$ is onto (following Theorem 2.2 (c)). This is helpful in the proof, since it means that for each $\sigma \in UEnv$ there is a $\rho \in OEnv$ such that $\overline{\rho} = \sigma$, but is not in fact necessary, because of Theorem 2.2 (b) and the fact that $C$ may be assumed to include any given system as a subsystem.

**Theorem 3.2.** Suppose $P$ is any CSP term. Then the following all hold.

a) $\mathcal{S}[\![P]\!]\sigma$ is defined for all $\sigma \in UEnv$.

b) $\mathcal{S}[\![P]\!]\overline{\rho} = \Phi(\mathcal{O}[\![P]\!]\rho)$ for all $\rho \in OEnv$.

c) For each ordinal $\alpha$ and each $\rho \in OEnv$ we have $\mathcal{S}[\![P]\!]\overline{\rho}^\alpha \geq \Phi_\alpha(\mathcal{O}[\![P]\!]\rho)$.

**Proof.** This is by structural induction on $P$. Given the sequence of Lemmas above, the cases of all the non-recursive operators are trivial. There is nothing to prove for part (a) since the denotational semantics of the first section only

got into potential trouble at recursive terms. Part (b) follows in each case from the appropriate (b)-Lemma. For example, given a term $P \Box Q$,

$$\begin{array}{rcll} \mathcal{S}[\![P \Box Q]\!]\overline{\rho} & = & (\mathcal{S}[\![P]\!]\overline{\rho}) \Box (\mathcal{S}[\![Q]\!]\overline{\rho}) & \text{by definition of } \mathcal{S} \\ & = & \Phi(\mathcal{O}[\![P]\!]\rho) \Box \Phi(\mathcal{O}[\![Q]\!]\rho) & \text{by induction} \\ & = & \Phi(\mathcal{O}[\![P]\!]\rho \Box \mathcal{O}[\![Q]\!]\rho) & \text{by Lemma 3.1.3 (b)} \\ & = & \Phi(\mathcal{O}[\![P \Box Q]\!]\rho) & \text{by definition of } \mathcal{O}. \end{array}$$

Part (c) follows in each case from the appropriate (c)-Lemma. For example, given a term $P \Box Q$, $\rho \in \mathit{OEnv}$ and an ordinal $\alpha$,

$$\begin{array}{rcll} \mathcal{S}[\![P \Box Q]\!]\overline{\rho}^{\alpha} & = & (\mathcal{S}[\![P]\!]\overline{\rho}^{\alpha}) \Box (\mathcal{S}[\![Q]\!]\overline{\rho}^{\alpha}) & \text{by definition of } \mathcal{S} \\ & \geq & \Phi_{\alpha}(\mathcal{O}[\![P]\!]\rho) \Box \Phi_{\alpha}(\mathcal{O}[\![Q]\!]\rho) & \text{by induction and monotonicity} \\ & \geq & \Phi_{\alpha}(\mathcal{O}[\![P]\!]\rho \Box \mathcal{O}[\![Q]\!]\rho) & \text{by Lemma 3.1.3 (c)} \\ & = & \Phi_{\alpha}(\mathcal{O}[\![P \Box Q]\!]\rho) & \text{by definition of } \mathcal{O}. \end{array}$$

It only remains to consider the case of a recursively defined term $\mu p.P$, where the result is known to hold of $P$. To prove part (a) it is sufficient (by Theorem 1.4 (c)) to show that, given $\sigma \in \mathit{UEnv}$, there is some fixed point of the function $F : \mathcal{U} \to \mathcal{U}$ defined

$$F(X) = \mathcal{S}[\![P]\!]\sigma[X/p]\,.$$

Choose $\rho$ such that $\overline{\rho} = \sigma$, and let $X = \Phi(\mathcal{O}[\![\mu p.P]\!]\rho)$. Now, since $\mathcal{O}[\![\mu p.P]\!]\rho$ has the single $\tau$ transition to $\mathcal{O}[\![P[\mu p.P/p]]\!]\rho$ it follows that

$$X = \Phi(\mathcal{O}[\![\mu p.P]\!]\rho) = \Phi(\mathcal{O}[\![P[\mu p.P/p]]\!]\rho) = \Phi(\mathcal{O}[\![P]\!]\rho[\mathcal{O}[\![\mu p.P]\!]\rho/p])$$

since $p$ is not free in $P[\mu p.P/p]$, by properties of substitution. But induction tells us that the right hand term above equals $\mathcal{S}[\![P]\!]\overline{\rho[\mathcal{O}[\![\mu p.P]\!]\rho/p]}$ which is in turn equal to $\mathcal{S}[\![P]\!]\overline{\rho}[\Phi(\mathcal{O}[\![\mu p.P]\!]\rho)/p] = \mathcal{S}[\![P]\!]\sigma[X/p]$. Hence $X = F(X)$ as required. This proves part (a) for $\mu p.P$.

For part (b), observe that $\mathcal{S}[\![P]\!]\overline{\rho}$ is defined to be the *least* fixed point of $F$ and that $X = \Phi(\mathcal{O}[\![\mu p.P]\!]\rho)$ has been shown to be a fixed point. It follows that $\mathcal{S}[\![P]\!]\overline{\rho} \leq X$. To prove the opposite inequality we show that, for all ordinals $\alpha$,

$$\Phi_{\alpha}(\mathcal{O}[\![\mu p.P]\!]\rho) \leq F^{\alpha}(\bot)\,.$$

When $\alpha = 0$ the result is trivial, for the left hand side equals $\bot$. If $\alpha$ is a limit ordinal and the result holds for all $\beta \in \alpha$ then it holds for $\alpha$ since both sides are simply the least upper bounds of their predecessors. So suppose it holds of $\beta$ and $\alpha = \beta + 1$. Then, since $\mathcal{O}[\![\mu p.P]\!]\rho$ has the single $\tau$ transition to $\mathcal{O}[\![P[\mu p.P/p]]\!]\rho$, it follows that

$$\begin{array}{rcll} \Phi_{\beta+1}(\mathcal{O}[\![\mu p.P]\!]\rho) & = & \Phi_{\beta}(\mathcal{O}[\![P[\mu p.P/p]]\!]\rho) & \\ & = & \Phi_{\beta}(\mathcal{O}[\![P]\!]\rho[\mathcal{O}[\![\mu p.P]\!]\rho/p]) & \\ & \leq & \mathcal{S}[\![P]\!]\overline{\rho[\mathcal{O}[\![\mu p.P]\!]\rho/p]}^{\beta} & \text{by (c) of } P \\ & \leq & \mathcal{S}[\![P]\!]\overline{\rho}^{\beta}[\Phi_{\beta}(\mathcal{O}[\![\mu p.P]\!]\rho)/p] & \\ & \leq & \mathcal{S}[\![P]\!]\overline{\rho}[F^{\beta}(\bot)/p] & \text{induction and monotonicity} \\ & = & F^{\beta+1}(\bot) & \text{definition of } F \end{array}$$

This proves the result for all $\alpha$. However we know by earlier work (Theorem 1.4 and Theorem 2.4) that there is $\alpha$ such that $F^\alpha(\bot) = X$ and $\Phi_\alpha = \Phi$. It follows that $X \leq \mathcal{S}[\![P]\!]\overline{\rho}$, completing the proof of (b).

It only remains to prove (c), in other words that, given $\rho$ and $\alpha$,

$$\Phi_\alpha(\mathcal{O}[\![\mu p.P]\!]\rho) \leq \mathcal{S}[\![\mu p.P]\!]\overline{\rho}^\alpha .$$

Once again we prove this by transfinite induction on $\alpha$. Again the result is easy for $\alpha = 0$ since the left hand side is $\bot$ and also for the limit ordinal case since the left hand side at $\alpha$ is then the least upper bound of the previous left hand sides, and $\mathcal{S}[\![P]\!]$ is monotone. So suppose $\alpha = \beta + 1$ and that the result holds at $\beta$. Then

$$\begin{array}{lcll}
\Phi_{\beta+1}(\mathcal{O}[\![\mu p.P]\!]\rho) & = & \Phi_\beta(\mathcal{O}[\![P[\mu p.P/p]]\!]\rho) & \\
& = & \Phi_\beta(\mathcal{O}[\![P]\!]\rho[\mathcal{O}[\![\mu p.P]\!]\rho/p]) & \\
& \leq & \mathcal{S}[\![P]\!]\rho[\mathcal{O}[\![\mu p.P]\!]\rho/p]^\beta & \text{by (c) of } P \\
& \leq & \mathcal{S}[\![P]\!]\overline{\rho}^\beta[\Phi_\beta(\mathcal{O}[\![\mu p.P]\!]\rho)/p] & \\
& \leq & \mathcal{S}[\![P]\!]\overline{\rho}^\beta[\mathcal{S}[\![\mu p.P]\!]\overline{\rho}^\beta/p] & \text{induction and monotonicity} \\
& = & \mathcal{S}[\![\mu p.P]\!]\overline{\rho}^\beta & \text{as recursions denote fixed points} \\
& \leq & \mathcal{S}[\![\mu p.P]\!]\overline{\rho}^{\beta+1} & \text{by monotonicity}
\end{array}$$

which proves it for $\beta + 1$. This completes the proof of Theorem 3.2. □

### On mutual recursion

The reader may have noticed that this section has not discussed the subject of mutual recursion, where finitely or infinitely many processes are defined in terms of each other. This was for two reasons. First, the formalisation of the syntax of mutual recursion and its operational semantics are rather complex. Furthermore, as is apparent from the above, we would have had to repeat much of the above analysis of single recursion in the mutual case.

Second, there is a simple transformation which converts any mutual recursion into a single one, which makes it all less necessary. Suppose we are defining processes $P_\lambda$ by mutual recursion for all $\lambda \in \Lambda$, where without loss of generality $\Lambda$ is disjoint from the alphabet $\Sigma$. In other words we are identifying each $P_\lambda$ (thought of as an element of $Var$) with a CSP term $F_\lambda(\underline{P})$, which may contain any or all of the variables $P_\mu$. We adjoin $\Lambda$ to $\Sigma$ to obtain a new alphabet $\Sigma'$ and define a new function term $F$ involving a single variable $P$ as follows:

$$F(P) = \lambda : \Lambda \rightarrow F'_\lambda(P) ,$$

where $F'_\lambda$ is the result of substituting each $P_\mu$ in $F_\lambda(\underline{P})$ by $(P_{\Sigma'} \| _{\Sigma'} \mu \rightarrow RUN) \backslash \{\mu\}$, where $RUN = a : \Sigma' \rightarrow RUN$ is the deterministic process which can always communicate anything. Thus $\mu P.F(P)$ denotes a process which on its first step gives the choice of $\Lambda$ and then acts like the $P_\lambda$ which was defined by mutual

recursion. For on each recursive call the correct "component" of $P$ is selected, and the process of selection hidden. (Note that $(P \,{}_{\Sigma'}\|_{\Sigma'} RUN) = P$ for any process $P$.) Note that this transformation does not use any infinite hiding or nondeterminism, and is therefore valid in boundedly nondeterministic CSP as well.

It is intuitively obvious, and can easily be proved, that there is a correspondence between the solutions of the mutually recursive definition and the vectors $\langle (P \,{}_{\Sigma}\|^{\prime}_{\Sigma} \mu \to RUN) \backslash \{\mu\} \mid \mu \in \Lambda \rangle$ for solutions $P$ of the single recursion. Thus we can assert that all mutual recursions do have solutions. We have not proved them congruent to their operational semantics, for the latter have not been defined. But these observations give one great confidence that such a result must be true for any reasonable semantics.

### On $\mathcal{N}'$ and $\Xi$

So far this section has concentrated solely on the semantics of CSP in the new model $\mathcal{U}$ and the corresponding abstraction map $\Phi$. Given the discussion at the start of this paper we would not expect to get such good results for $\mathcal{N}'$ and $\Xi$ since the finer semantics uses infinite traces in a crucial way to determine the finite behaviours when computing the hiding operator. This assessment is correct: we can in fact prove only an inequality rather than a full congruence, though this becomes an equality in the absence of hiding.

Below, $\mathcal{T}$ denotes the semantic function mapping CSP into $\mathcal{N}'$, so that $\mathcal{T} : \mathbf{E} \to NEnv \to \mathcal{N}'$, where $NEnv = Var \to \mathcal{N}'$. If $\rho \in OEnv$, then $\hat{\rho}$ is the corresponding element of $NEnv$: $\hat{\rho}[\![p]\!] = \Xi(\rho(p))$.

**Theorem 3.3.** If $P$ is any CSP term and $\rho$ any element of $OEnv$, then

$$\mathcal{T}[\![P]\!]\hat{\rho} \leq \Xi(\mathcal{O}[\![P]\!]\rho .$$

If the definition of $P$ does not involve the hiding operator, then

$$\mathcal{T}[\![P]\!]\hat{\rho} = \Xi(\mathcal{O}[\![P]\!])\rho .$$

**Proof.** One could prove this result from first principles like we had to do for $\mathcal{U}$. It is, however, much easier to derive it from Theorem 3.2. For this, we need maps between $\mathcal{N}'$ and $\mathcal{U}$. If $\langle F, D, I \rangle \in \mathcal{U}$, then we define its projection into $\mathcal{N}'$ to be $\pi\langle F, D, I \rangle = \langle F, D \rangle$. If $\langle F, D \rangle \in \mathcal{N}'$, we define $\iota\langle F, D \rangle = \langle F, D, I \rangle$, where $I = \overline{\{s \mid (s, \emptyset) \in F\}}$. Note that $\pi \circ \iota$ is the identity map on $\mathcal{N}'$ and that $\iota(\pi(P)) \leq P$ for all $P \in \mathcal{U}$.

Under the conditions of the theorem we know, by Theorem 3.2, that

$$\Xi(\mathcal{O}[\![P]\!]\rho) = \pi(\Phi(\mathcal{O}[\![P]\!]\rho)) = \pi(\mathcal{S}[\![P]\!]\bar{\rho}) .$$

It will therefore be enough to prove that, for all CSP terms $P$ and all $\sigma \in UEnv$,

$$\mathcal{T}[\![P]\!]\bar{\sigma} \leq \pi(\mathcal{S}[\![P]\!]\sigma .$$

where $\tilde{\sigma}[\![p]\!] = \pi(\sigma[\![p]\!])$ for each $p \in Var$, and that this inequality may be replaced by equality when $P$ does not involve hiding. This is a straightforward structural induction. The clause for each non-hiding operator follows simply from monotonicity and the fact that such operators cummute with the projection function $\pi$. For example

$$\pi(P \Box Q) = \pi(P) \Box \pi(Q) \qquad \text{for all } P, Q \in \mathcal{U}.$$

This is just another way of saying that the failures and divergence components of these operators does not rely on infinite traces. We get a weaker result for hiding

$$\pi(P\backslash X) \geq \pi(P)\backslash X \qquad \text{for all } P \in \mathcal{U}.$$

In fact, it is easy to see from the definition that $\pi(P)\backslash X = \pi((\iota(\pi(P))\backslash X)$, since over $\mathcal{N}'$ the hiding operator has to assume that all infinite traces are present all of whose finite prefixes are.

The case of recursion follows from the obvious continuity of $\pi$. If we are given a recursive term $\mu p.P$ and $\sigma \in UEmv$, then if

$$f(X) = \mathcal{T}[\![P]\!]\tilde{\sigma}[X/p] \quad F(Y) = \mathcal{S}[\![P]\!]\sigma[Y/p]$$

we can prove that $\pi(F^\alpha(\bot)) \geq f^\alpha(\bot)$ or $\pi(F^\alpha(\bot)) = f^\alpha(\bot)$ for all $\alpha$ as appropriate. The limit ordinal case is by continuity and the successor case by (structural) induction. The result then follows immediately. □

The inequality for terms involving hiding may well be strict, as is demonstrated by the process $(\sqcap\{P_n \mid n \in \mathbb{N}\})\backslash\{a\}$, where $P_0 = STOP$ and $P_{n+1} = a \rightarrow P_n$. This is identified by $\mathcal{T}$ with $\bot$, but operationally is identical to $STOP$. Theorem 3.3 at least tells us that the value of the operational process is no worse than that predicted by the denotational semantics. It would have been much more dangerous the other way round, since it would then have been possible for an implementation to behave in a way that has been "proved" impossible in the abstract semantics.

## 4. Conclusions

We have seen a long and technical proof that the semantics of CSP in the infinite traces model are well defined, and have simultaneously proved their congruence with an operational semantics. I hope that the fact that these proofs were difficult will not obscure the fact that the concept behind the model – adding infinite traces to the existing failures model – is simple and that the semantic definitions are all straightforward.

In the later sections of this paper almost all the work was cast in terms of the coarser definedness order $\leq$ rather than the nondeterminism order $\sqsubseteq$. The reason for this was that most of our results, stated in terms of $\leq$, trivially imply

the corresponding results for $\sqsubseteq$ but not the reverse. (For example consider the (c)-Lemmas in Theorem 3.1.) In fact all of the work can be recast in terms of $\sqsubseteq$ if desired.

This leads to the same question as was posed in [R2], namely that of which is the natural order to use when presenting the model and semantics, given that both work. Here the arguments are slightly different. On the one hand now neither order is complete (whereas only $\leq$ was over $\mathcal{N}'$). However $\leq$ does still have a nicer theory of least upper bounds than $\sqsubseteq$, for they are always given by intersection where they exist while this is not even true for directed sets for $\sqsubseteq$. On the other, $\sqsubseteq$ is simpler to define and is perhaps more intuitive, but it does not have such a claim over $\mathcal{U}$ to be the "established" order as over $\mathcal{N}$ or $\mathcal{N}'$. This question will be best resolved by time and experience.

On the technical side we have seen in this work that completeness and monotonicity are natural casualties of the introduction of unbounded nondeterminism, but that their absence does not matter unduly except in the sense that proofs become more difficult and require advanced mathematics. To the author the most interesting feature of the proof is the way the approximate abstraction functions $\Phi_\sigma$ show that the least fixed point corresponds with the operationally natural one via a type of "non-destructiveness" argument.

Future work on this model must include a much fuller investigation of its algebraic properties. The ones seen in this paper, namely the infinite distributivity of all operators and the laws of the Lemmas in Theorem 3.1, were simply those needed for the rest of the work. Another issue will be the study of other unboundedly nondeterministic constructs such as fair hiding operators. We should note that it is only permissible to add a new operator (other than one derived from existing operators) to this version of CSP if it can be given an operational semantics and Lemmas of the type seen in Theorem 3.1 proved about it. The work of Barrett [Bar] on the operational semantics of fairness will probably be important here. It will also be interesting to see what use can be made of the infinite traces component in the *specifications* of processes. For example one could add a clause to the usual specification of a buffer which stated that the buffer never does infinitely many inputs without an output, so that anything one puts in is eventually going to come out (even in the presence of an environment which eagerly places as much as possible into the buffer at all times).

The difficulties one encounters when dealing with unbounded nondeterminism, particularly the sort which is only detectable from infinite behaviours, are certainly not restricted to the models seen in this paper. Hopefully some of the work reported here will transfer to other formalisms for concurrency. One place where valuable work could be done is in timed CSP (see [RR1, RR2, Re]). The incorporation of infinite behaviours there (were it possible) would allow more abstract and general expressions of such modalities as "eventually" which appear in some forms of temporal logic.

# Appendix: more details of $\mathcal{U}$

## Alternatives to axiom (8).

The version of axiom (8) seen earlier is in a different style from the others, and from all other axioms of CSP models I have seen. Its discovery was the result of an evolutionary process in which it passed through various incorrect forms (all weaker ones which failed to be compositional) and an equivalent but rather inelegant equivalent formulation. All the earlier forms were expressed in terms of "games" played between the experimenter and the process during which the experimenter tries to force infinite traces out of the process. One incorrect but interesting earlier attempt is described next to show the difficulties which are involved here.

An experimenter who sets out to force an infinite trace out of a process may have decided in advance what his strategy will be. In this case his strategy can be described as a prefix closed nonempty set of traces $T$. At each step, if he has so far succeeded in communicating $t \in T$ with the process, he will next attempt $(T/t)^0$ (the set $\{a \mid t\langle a\rangle \in T\}$). If the process can never refuse any of these sets, it is clear that an infinite trace in $\overline{T}$, the set of all infinite traces all of whose finite prefixes are in $T$, must result. This leads to a property analogous to axiom (8).

$$(8') \quad \begin{array}{l}(s,\emptyset) \in F \wedge T \text{ a nonempty prefix closed set of finite} \\ \text{traces such that } t \in T \Rightarrow (st,(T/t)^0) \notin F\end{array} \Rightarrow \exists u \in \overline{T}.su \in I$$

While this axiom is (or should be) self-evidently true of all real processes, it turns out to be not quite strong enough. It seems that some CSP operators (e.g., both forms of parallel composition) fail to be closed under it, and it is strictly weaker (even in the context of (1-7)) than (8). Consider the following example of a process $P$ with $D = \emptyset$ and $\Sigma = \{a,b,c,d\}$. $P$ cannot at any time refuse $\{a,b\}$ or $\{c,d\}$ (or any superset of either) but can communicate any finite sequence in $\Sigma^*$.

$$F = \{(s,X) \mid s \in \Sigma^* \wedge \{a,b\} \not\subseteq X \wedge \{c,d\} \not\subseteq X\}\,.$$

Forcing strategies of the form seen in (8') are just sets of traces $T$ such that, whenever $t \in T$ then either $\{t\langle a\rangle, t\langle b\rangle\} \subseteq T$ or $\{t\langle c\rangle, t\langle d\rangle\} \subseteq T$ (this starting from any trace $s \in \Sigma^*$.) An element of $imp(P)$ is a deterministic process which, after any trace $s$, must either be able to do $a$ or $b$ and must either be able to do $c$ or $d$. It is possible to include enough infinite traces to satisfy all the strategies implied by (8') yet leaving $imp(P)$ empty because there is an infinite trace missing from every single one of them.

This is shown by a set-theoretic construction which relies on the facts that there are exactly $c$ ($= 2^{\aleph_0}$, the continuum) strategies $(s,T)$, exactly $c$ possible implementations $Q$ and exactly $c$ infinite traces in each such $Q$ and satisfying

each such $(s,T)$. The sets of strategies and implementations are both ordered by the initial ordinal $c$ (i.e., the smallest ordinal with this cardinal) and sets $A_\alpha$ and $B_\alpha$ are constructed recursively for all $\alpha \leq c$ so that $A_\alpha$ contains a representative of each strategy $(s,T)_\beta$ for $\beta < \alpha$, $B_\alpha$ contains a representative of each implementation $Q_\beta$ for $\beta < \alpha$ and $B_\alpha \cap A_\alpha = \emptyset$. We set $A_0 = B_0 = \emptyset$, at limit ordinals $A_\alpha = \bigcup_{\beta\in\alpha} A_\beta$ and $B_\alpha = \bigcup_{\beta\in\alpha} B_\beta$. Since the $A_\alpha$ and $B_\alpha$ always have cardinal less than $c$ for $\alpha < c$, and the sets of infinite traces of $(s,T)_\alpha$ and of $Q_\alpha$ are both of size $c$, it is always possible to find distinct infinite traces $u$ and $u'$ from these sets which are different from all of $A_\alpha \cup B_\alpha$. Then $A_{\alpha+1} = A_\alpha \cup \{u\}$ and $B_{\alpha+1} = B_\alpha \cup \{u'\}$. The set $I = A_c$ is then such that each (8′) strategy is satisfied but $imp(F,\emptyset,I)$ is empty.

Axiom (8′) seems to be weaker because it assumes that the user has decided on his strategy from the start, whereas he might keep on changing his mind. The fact that he changes his mind about what he is *going* to offer the process does not change what *has* been communicated with him before. My first correct axioms were (more complex) modifications of this one which took account of this fact.

On seeing axiom (8) (but not my earlier versions) Stephen Blamey came up with another one which is essentially the same as my first correct one (cast in terms of games) but a lot more elegant. It looks like a "classical" model axiom since it is cast in terms of logic, but the logic is unusual since it has both infinitely nested quantifiers and infinitary use of propositional connectives. There are several different presentations of it, one of which is given below.

$$
\begin{aligned}
(8^*)\quad & \forall s \in traces(P).\forall X_0 \subseteq \Sigma.(s,X_0) \notin F \Rightarrow \\
& \exists a_1 \in X_0.(s\langle a_1\rangle,\emptyset) \in F \wedge \forall X_1 \subseteq \Sigma.(s\langle a_1\rangle,X_1) \notin F \Rightarrow \\
& \quad \exists a_2 \in X_1.(s\langle a_1,a_2\rangle,\emptyset) \in F \wedge \forall X_2 \subseteq \Sigma.(s\langle a_1,a_2\rangle,X_2) \notin F \Rightarrow \\
& \qquad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \\
& \qquad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \\
& \qquad\qquad \Rightarrow s\langle a_i \mid i \geq 1\rangle \in I
\end{aligned}
$$

This axiom can be read as saying that, from all finite traces that a process can reach, any infinite sequence of sets offered to the process will either lead to refusal at some finite stage or to an infinite trace. At first sight this axiom appears very similar to (8′) above. But it is not, because the way the quantifiers nest mean that consistent families of infinite traces are delivered for all decisions an experimenter might make on what sets to offer: if two experiments differ at the $n$th stage, the infinite traces delivered will not differ before the $n$th stage.

The (game-theoretic) proof that this axiom is equivalent to our axiom (8) in the context of the others may be found in [Blam]. The reader may note that (8*) is easily seen to represent only a lower bound on the set of infinite traces, in that adding extra infinite traces cannot invalidate it. This was not so apparent of (8), where we had to prove Lemma 1.1. (Of course a construction similar to

that in the proof of that Lemma is a part of the proof of equivalence of (8) and (8*).)

Blamey has also pointed out that modification of the "modalities" in (8′) above produces a further correct axiom, arguably simpler than either (8) or (8*). If we let $T$ range over all prefix closed nonempty sets of traces, then this new version can be stated:

$$(8^\dagger) \quad (s,\emptyset) \in F \Rightarrow \exists T.(\forall t \in T.(st, \{a \mid t\langle a\rangle \notin T\}) \in F) \wedge \forall u \in \overline{T}.su \in I$$

where again $\overline{T} = \{u \mid \forall s.s < u \Rightarrow s \in T\}$. This is closer in spirit to (8′) than to (8*), for the set $T$ represents no more nor less than a deterministic implementation of $P$ *after* $s$. This axiom is thus easily seen to be implied by the statement "each finite trace $s$ belongs to some deterministic implementation of $P$", which is trivially implied by (8). (For every pre-deterministic process is weaker than a deterministic one with the same traces.) $(8^\dagger)$ implies (8) since an easy consequence of something proved earlier (Lemma 1.1) is that for (8) to be true it is enough for each failure to be present in some deterministic implementation. If, for each $s \in traces(P)$, $T_s$ is given by $(8^\dagger)$ for $s$, and $(r, X)$ is any failure of $P = \langle F, D, I\rangle$, the set of traces

$$\{s \mid s \leq r\} \cup \{s\langle a\rangle t \mid s < r \wedge s\langle a\rangle \not\leq r \wedge s\langle a\rangle \in traces(P) \wedge t \in T_{s\langle a\rangle}\}$$
$$\cup \bigcup \{r\langle a\rangle t \mid t \in T_{r\langle a\rangle} \wedge r\langle a\rangle \in traces(P) \wedge a \notin X\}$$

can be seen to represent a deterministic implementation of $P$ exhibiting $(r, X)$. Note that this argument also shows that $(8^\dagger)$ is *equivalent* to the statement that each trace $s$ belongs to some deterministic implementation.

We observe that, thanks to axioms (4) and (7), the statement of $(8^\dagger)$ can be weakened a little: we can ignore the cases when $s$ is a divergence trace. This gives

$$(8^\ddagger) \quad ((s,\emptyset) \in F \wedge s \notin D) \Rightarrow \exists T.(\forall t \in T.(st, \{a \mid t\langle a\rangle \notin T\}) \in F) \wedge \forall u \in \overline{T}.su \in I$$

which sometimes has shorter proofs than the original version.

The alternative versions (8*), $(8^\dagger)$ and $(8^\ddagger)$ are probably more concise than the original (8), when one takes into account all the discussion of pre-deterministic processes necessary to set (8) up. However (8*) has the disadvantage that the meaning of an infinitary logical expression like the above may be opaque to some. Also, when one is doing technical analysis of the model such as that seen below in the well-definedness proofs of the various operators, (8) seems generally easier to deal with than (8*). We have seen that $(8^\dagger)$ and $(8^\ddagger)$ are technically close in spirit to (8), and in technical manipulations they are similar to use. Which one of them should be stated as *the* axiom (8) will depend on whether one prefers the conciseness of $(8^\dagger)$ or $(8^\ddagger)$ to the fact that in (8) the true structure of this axiom is laid bare rather more clearly.

## Technical properties of CSP operators

We now turn to the proof of Theorem 1.3, namely that all CSP operators are well defined and monotonic with respect to both orders. As was noted in the earlier discussion of Theorem 1.3, with the exception of hiding we can restrict our attention to the infinite traces component since in each case the other components are defined exactly as over the existing model $\mathcal{N}'$. And in each of these cases monotonicity is trivial, and axioms (6) and (7) elementary. Leaving hiding on one side temporarily, it will therefore suffice to prove that (a version of) axiom (8) holds in each case. We have already seen (e.g., in the proof of Theorem 2.2) how (8) itself is used in manipulations. Below we prove the equivalent form ($8^\dagger$), though it convenient to *assume* the formally stronger statement ($8^\dagger$), which is of course permissible since the two versions are equivalent in the presence of (1-7).

The proofs for all operators come down to more or less the same thing. For each operator $F$ we have to create, for each finite nondivergent trace $s$ of $F(P,\ldots)$, the set $T$ required by ($8^\dagger$). In each case the trace $s$ exists in $F(P,\ldots)$ as a consequence of at most one trace from each of the arguments $P,\ldots$ of $F$. The set $T$ is then constructed from the $T$s which are chosen by ($8^\dagger$) in the arguments relative to the traces used to construct $s$.

The individual cases vary in difficulty. Consider the nondeterministic composition operator $\sqcap S$, where $S \subseteq \mathcal{U}$ is nonempty. The validity of axiom ($8^\dagger$) follows from the observation that, if $s \in \mathit{traces}(\sqcap S) \setminus \mathcal{D}[\![\sqcap S]\!]$ then there is $P \in S$ such that $s \in \mathit{traces}(P)$ (and necessarily $s \notin \mathcal{D}[\![P]\!]$). The $T$ which works for $s$ in $P$ will also work in $\sqcap S$, since the failures and infinite traces of $P$ are subsets of those of $\sqcap S$.

The communication operator $x : B \to P_x$ is almost as easy. For any nonempty trace $\langle a \rangle s$, necessarily $a \in B$ and $s$ is a trace of $P_a$. Choose $T$ for $P_a$ relative to $s$. For $s = \langle\rangle$ we simply choose, for each $a \in B$, a $T_a$ relative to $P_a$ and $\langle\rangle$. Then $T = \{\langle\rangle\} \cup \{\langle a \rangle s \mid a \in B \wedge s \in T_a\}$. In either case the requirements for $T$ are easily shown to be met.

Suppose $s$ is a nondivergent trace of $P \,{}_B\|_C\, Q$. Then $s \restriction B$ is a nondivergent trace of $P$ and $s \restriction C$ is a nondivergent trace of $Q$, so there exist prefix-closed nonempty sets $T_P$ and $T_Q$ such that

$$(\forall t \in T_P.((s \restriction B)t, \{a \mid t\langle a \rangle \notin T_P\}) \in F_P) \wedge \forall u \in \overline{T_P}.(s \restriction B)u \in I_P \quad \text{and}$$

$$(\forall t \in T_Q.((s \restriction C)t, \{a \mid t\langle a \rangle \notin T_Q\}) \in F_Q) \wedge \forall u \in \overline{T_Q}.(s \restriction C)u \in I_Q\,.$$

It may be assumed that $T_P, T_Q \subseteq (B \cup C)^*$, since whenever $st$ is a minimal divergence trace of $P$ or $Q$ it is possible to include no extension of $t$ in $T_P$ or $T_Q$. Define $T = \{t \in (B \cup C)^* \mid t \restriction B \in T_P \wedge t \restriction C \in T_Q\}$. Now, if $t \in T$ then $st \restriction B = (s \restriction B)(t \restriction B)$ and $st \restriction C = (s \restriction C)(t \restriction C)$. It follows by definition of $T_P$ and $T_Q$ that

$$(st \restriction B, \{a \mid (t \restriction B)\langle a \rangle \notin T_P\}) \in F_P \quad \text{and} \quad (st \restriction C, \{a \mid (t \restriction C)\langle a \rangle \notin T_Q\}) \in F_Q$$

It follows from the definition of the failures of $P\ {}_B|\!|_C\ Q$ that

$$(st, (\Sigma\backslash(B \cup C)) \cup X \cup Y) \in \mathcal{F}[\![P\ {}_B|\!|_C\ Q]\!]$$

where $X = \{a \in B \mid (t\restriction B)\langle a\rangle \notin T_P\}$ and $Y = \{a \in C \mid (t\restriction C)\langle a\rangle \notin T_Q\}$.

But this is equal to $(st, \{a \mid t\langle a\rangle \notin T\})$ by definition of $T$, so the first requirement for $T$ is met.

Secondly, suppose $u \in \overline{T}$. Whenever $t < u$ we have $t\restriction B \in T_P$ and $t\restriction C \in T_Q$. But $\{t \mid t \leq u\restriction B \wedge t \text{ is finite}\} = \{t\restriction B \mid t < u\}$ and similarly for $C$. It follows that $u\restriction B \in T_P \cup \overline{T_P}$ and $u\restriction C \in T_Q \cup \overline{T_Q}$. Thus $su\restriction B \in \mathit{Traces}(P)$ and $su\restriction C \in \mathit{Traces}(Q)$, so that $su \in \mathcal{I}[\![P\ {}_B|\!|_C\ Q]\!]$ as required.

This was in fact a rather straightforward construction: essentially, the implementation of $P\ {}_B|\!|_C\ Q$ is found by running implementations of $P$ and $Q$ in parallel. This is possible because $\|$ is an operator which, like prefixing, never introduces nondeterminism. The only other operator with this property is inverse image, $f^{-1}[P]$, where a corresponding construction works.

The interleaving parallel operator requires a little more thought since it can introduce nondeterminism: run two deterministic implementations together and the result need not be deterministic. If $s$ is a nondivergent trace of $P\ |||\ Q$ then there must exist traces $s_P, s_Q$ of $P$ and $Q$ respectively such that $s \in \mathit{merge}(s_P, s_Q)$. Choose $T_P$ for $P$ relative to $s_P$ and $T_Q$ for $Q$ relative to $s_Q$. We now build up $T$ and functions $\phi_P : T \to T_P$ and $\phi_Q : T \to T_Q$ simultaneously by recursion on the length of $s \in T$. Initially $\langle\rangle \in T$ and $\phi_P(\langle\rangle) = \phi_Q(\langle\rangle) = \langle\rangle$. If $s \in T$ then $s\langle a\rangle \in T$ for all $a$ such that $\phi_P(s)\langle a\rangle \in T_P$ or $\phi_Q(s)\langle a\rangle \in T_Q$. If $\phi_P(s)\langle a\rangle \in T_P$ but not $\phi_Q(s)\langle a\rangle \in T_Q$ then $\phi_P(s\langle a\rangle) = \phi_P(s)\langle a\rangle$ and $\phi_Q(s\langle a\rangle) = \phi_Q(s)$, and vice-versa. If $\phi_P(s)\langle a\rangle \in T_P$ and $\phi_Q(s)\langle a\rangle \in T_Q$ then an arbitrary choice is made: without loss of generality we define $\phi_P(s\langle a\rangle) = \phi_P(s)\langle a\rangle$ and $\phi_Q(s\langle a\rangle) = \phi_Q(s)$. The important thing is that $\phi_P$ and $\phi_Q$ are monotonic (with respect to the prefix order) functions with the property that, for all $t \in T$, $\phi_P(t) \in T_P$, $\phi_Q(t) \in T_Q$ and $t \in \mathit{merge}(\phi_P(t), \phi_Q(t))$. This means that, given $u \in \overline{T}$, there are elements of $T_P \cup \overline{T_P}$ and $T_Q \cup \overline{T_Q}$ which merge together to form $u$ (the prefix order least upper bounds of $\{\phi_P(s) \mid s < u\}$ and $\{\phi_Q(s) \mid s < u\}$ respectively). This implies that $u \in \overline{T} \Rightarrow su \in \mathcal{I}[\![P\ |||\ Q]\!]$ as required. The way $T$ is constructed also implies that it satisfies the other (first) requirement of $T$ relative to $s$.

This way of resolving the nondeterminism introduced by $|||$ which, in some sense, minimises the set $T$ of traces is not in fact the only one: we could have made the more generous and obvious definition $T = \bigcup\{\mathit{merge}(t_P, t_Q) \mid t_P \in T_P \wedge t_Q \in T_Q\}$. However the argument that $\{su \mid u \in \overline{T}\}$ is a subset of $\mathcal{I}[\![P\ |||\ Q]\!]$ would then have been a delicate argument using König's Lemma: possible in this case because the nondeterminism introduced by $|||$ is always finite: choose left or right. There is the same choice (with easier arguments) in the cases of two of the other operators which can introduce nondeterminism: ; and $\Box$. The final two: $f(P)$ and $P\backslash X$ are different for they can introduce unbounded

nondeterminism: therefore the sort of resolution of nondeterminism seen above now becomes strictly necessary.

$f(P)$ can introduce unbounded nondeterminism when $f$ is not finite-to-one. In the case of a nondivergent trace $f(s)$ of $f(P)$, where $s$ is a trace of $P$, we choose $T_P$ relative to $P$ and $s$ and construct $T$ and a function $\phi$ by firstly including $<>\in T$ and setting $\phi(<>) =<>$. Then, if $s \in T$ we include $s\langle f(a)\rangle$ in $T$ for all $a$ such that $\phi(s)\langle a\rangle \in T_P$. $\phi(s\langle f(a)\rangle)$ is then defined to be $\phi(s)\langle a\rangle$, an arbitrary choice being made if there is more than one such $a$.

As was observed earlier, hiding is unlike the other operators in that the infinite traces of $P$ influence all components of $P\backslash X$ rather than just its infinite traces. It is thus necessary to prove axioms (1)-(5) as well as (6), (7) and (8). The fact that $\mathcal{F}[\![P\backslash X]\!]$ is nonempty and that $\mathit{Traces}(P\backslash X)$ is prefix closed (axioms (1) and (6)) both follow easily from the following Lemma.

**Lemma A.1.** If $s \in \mathit{traces}(P)$ then $s\backslash X \in \mathit{traces}(P\backslash X)$.

**Proof.** Let $T$ be chosen by axiom $(8^\dagger)$ relative to $s$. There are two possibilities we must consider. Either there exists $t \in T$ such that $t \in X^*$ but $\{a \mid t\langle a\rangle \in T\} \cap X = \emptyset$, or no such $t$ exists. In the first case we get $(st, X) \in \mathcal{F}[\![P]\!]$ and hence $(s\backslash X, \emptyset) \in \mathcal{F}[\![P\backslash X]\!]$. In the second case, since $<>\in T \cap X^*$ there must exist $u \in X^\omega \cap \overline{T}$ and hence $su \in \mathcal{I}[\![P]\!]$ so that $s\backslash X = su\backslash X \in \mathcal{D}[\![P\backslash X]\!]$. The result follows immediately. □

Axioms (2), (3), (4), (5) and (7) are all easy to prove. It remains to prove $(8^\ddagger)$. If $s$ is a nondivergent trace of $P\backslash X$ then there is a trace $s_P$ of $P$ such that $s = s_P\backslash X$. Choose $T_P$ relative to $s_P$ in $P$. As was the case for $|||$ and $f(P)$ we construct $T$ for $s$ together with a function $\phi : T \to T_P$. Initially $<>\in T$ and $\phi(<>) =<>$. If $t \in T$ then *either* there is an infinite $u \in X^\omega$ such that $\phi(t)u \in \overline{T_P}$, in which case we define $t$ to have no extensions in $T$, *or* not, in which case we include in $T$ each $t\langle a\rangle$ where $a \notin X$ and there exists $r \in X^*$ such that $\phi(t)r\langle a\rangle \in T_P$. We define $\phi(t\langle a\rangle) = \phi(t)r\langle a\rangle$, once again an arbitrary choice being made if there is any ambiguity. This function $\phi$ works in just the same way as we have seen before: it is trace-monotonic and $\phi(t)\backslash X = t$ for all $t \in T$, which means that for each $u \in \overline{T}$ we have $u = u'\backslash X$, where $u'$ is the least upper bound of $\{\phi(t) \mid t < u\}$. $u'$ is necessarily in $\overline{T_P}$ and hence $su = (s_P u')\backslash X$ is an element of $\mathcal{I}[\![P\backslash X]\!]$ as required. That $T$ also satisfies the first requirement for axiom $(8^\ddagger)$ is easily checked.

All the operators are obviously $\sqsubseteq$-monotone since they all construct the behaviours of the result process positively from the behaviours of the operand(s). As was observed when Theorem 1.3 was stated, this implies that all operators other than hiding are $\leq$-monotone since they are $\leq$-monotone over $\mathcal{N}''$ and the failures/divergence components of those operators are defined exactly as over $\mathcal{N}'$.

The only monotonicity result left to prove is the $\leq$-monotonicity of $P\backslash X$. So suppose $P \leq Q$. Since we know that $P\backslash X \sqsubseteq Q\backslash X$ it is enough to prove

that, whenever $s \notin \mathcal{D}[\![P\backslash X]\!]$, $\mathcal{R}[\![P\backslash X]\!]s \subseteq \mathcal{R}[\![Q\backslash X]\!]s$ and that $\mu(\mathcal{D}[\![P\backslash X]\!]) \subseteq traces(Q)$. The first of these facts follows from the fact that all nondivergent failures of $P\backslash X$ are consequences of nondivergent failures of $P$. If $s \in \mu(\mathcal{D}[\![P\backslash X]\!])$ then *either* there is an element $t$ of $\mu(\mathcal{D}[\![P]\!])$ such that $t\backslash X = s$ *or* there is $u \in \mathcal{I}[\![P]\!]$ such that $u\backslash X = s$ and $t < u \Rightarrow t \notin \mathcal{D}[\![P]\!]$, which implies there is $t \in traces(P)\backslash \mathcal{D}[\![P]\!]$ such that $t\backslash X = s$. In either case $t \in traces(Q)$ by definition of $\leq$ and so $s \in traces(Q\backslash X)$ by Lemma A.1 above.

The remaining part of Theorem 1.3 is its statement that the CSP operators are distributive in the sense that $F(\sqcap S) = \sqcap\{F(P) \mid P \in F\}$ for each operator $F$ and nonempty set $S$. This is a direct consequence of the facts that $\sqcap$ is simply component-wise union and that, for each operator $F$, each single behaviour of $F(P)$ is always attributable to at most one behaviour of each operand of $F$. Thus each behaviour of $F(\sqcap S)$ is the consequence of some behaviour of some element $P$ of $S$, which means that the same behaviour must be present in $F(P)$. This arbitrary distributive law did not hold over $\mathcal{N}'$ in the case of hiding [R2] precisely because the hiding operator there requires more than one behaviour of $P$ (in fact, an infinite number) to deduce some behaviours of $P\backslash X$. A finite distributive law still holds there by a separate argument which relies on König's Lemma.

### Acknowledgements

As will be apparent from the Appendix, Stephen Blamey has put a lot of work into analysing, refining and understanding the axioms for $\mathcal{U}$. In addition this work has been assisted by conversations with a number of colleagues, notably Paul Gardiner, Alan Jeffrey and David Walker.

## References

[B] Brookes, S.D., *A Model for Communicating Sequential Processes*, Oxford University D.Phil. thesis, 1983.

[Bar] Barrett, G., *The semantics and implementation of occam*, Oxford University D.Phil. thesis, forthcoming.

[Blam] Blamey, S.R., *The soundness and completeness of axioms for CSP processes*, forthcoming.

[BHR] Brookes, S.D., Hoare, C.A.R., and Roscoe, A.W., *A theory of communicating sequential processes*, JACM Vol. 31, No. 3 (July 1984) 560-599.

[BRW] Brookes, S.D., Roscoe A.W., and Walker, D.J., *An operational semantics for CSP*, Submitted for publication.

[H] Hoare, C.A.R., *Communicating sequential processes*, Prentice-Hall, 1985

[R1] Roscoe, A.W., *A mathematieal theory of communicating processes*, Oxford University D.Phil. thesis, 1982.

[R2] Roscoe, A.W., *An alternative order for the failures model*, in this volume.

[R3] Roscoe, A.W., *Analysing infinitely branching trees*, in preparation.

[Re] Reed, G.M., *A uniform mathematical theory for real-time distributed computing*, Oxford University D.Phil. thesis, 1988.

[RR1] Reed, G.M., and Roscoe, A.W., *A timed model for communicating sequential processes*, Proceedings of ICALP'86, Springer LNCS 226 (1986), 314-323.

[RR2] Reed, G.M., and Roscoe, A.W., *Metric spaces as models for real-time concurrency*, to appear in the proceedings of MFPLS87 (Springer LNCS).