THE UNIVERSITY OF MICHIGAN

COMPUTING RESEARCH LABORATORY

APPLICATIONS OF TOPOLOGY TO SEMANTICS OF COMMUNICATING PROCESSES

William C. Rounds

CRL-TR-32-84

July 1984

Room 1079, East Engineering Building
Ann Arbor, Michigan 48109
USA
Tel: (313) 763-8000

APPLICATIONS OF TOPOLOGY TO SEMANTICS OF COMMUNICATING PROCESSES

William C. Rounds *
Computer Science and Engineering Division
University of Michigan
Ann Arbor, Michigan 48109

1. Introduction

In this paper we point out some general principles which underlie proofs of correctness for communicating processes. We are concerned with those properties of processes which can be defined with reference to the possible finite behaviors of a process. These include *safety properties*: those assertions which say that nothing undesirable will ever happen.

Hoare, Brookes, and Roscoe, in their paper on CSP [HBR], define *buffers* by means of a collection of safety assertions about buffer behavior. They state some general theorems about properties of processes which can be established using a rule of proof called *fixed-point induction.* We generalize these results to a more refined semantic model than theirs. In doing so, we are led to an interesting application of the *compactness theorem* from first-order logic. This result can be interpreted as saying that the so-called Stone topology on a class of mathematical structures is compact. This topology is induced by defining the closed sets to be classes of structures which satisfy every assertion in a given (consistent) collection of assertions. Our structures represent the possible behaviors of a process, and we show that the definition of buffers can be rephrased as a consistent collection of assertions. Therefore the class of buffers is closed. We also prove that the Stone topology is definable by a metric given naturally in terms of Milner's observational equivalence relations. Convergence in the metric is guaranteed by the hypothesis of the rule of fixed-point induction; in fact, this rule is seen to be nothing more or less than the Banach fixed-point theorem for the space in question.

We view this work as a synthesis of the methods of Hoare, Brookes, and Roscoe with those of de Bakker and Zucker [BaZ], and also those of Milner [M]. Our examples confirm the validity of the idea of *specification semantics* in [OH]. Moreover, the suggestions of Smyth [S] are confirmed because the general notions of topology serve as a guide to formulating theorems about all these models.

The paper is organized into four sections, of which this is the first. Section 2 introduces the basic model of processes–the transition system– and presents the logical system used to describe these models. Section 3 is devoted to buffers and CSP. The final section gives another application of the compactness theorem which solves a problem left open in [Rou] concerning transition systems over a countable alphabet.

2. The operational model

The underlying framework for expressing the behavior of processes will be the familiar *transition system.* This is a tuple

$$S = \langle Q, q, \{\xrightarrow{\sigma} : \sigma \in \Sigma \cup \{\tau\}\}\rangle$$

where Q is a countable set of *states*; $q \in Q$ is the *initial state*; Σ is an alphabet of *event names*; τ is a special event, not in Σ, and denoting the *silent transition.* The relations $\xrightarrow{\sigma}$ are binary relations on Q which for each σ event, denote the possible changes of state which may take place during that event. The elements of Σ are

* Research supported by NSF grant BSR–8301022.

called *visible events*. Typically these might represent the instantaneous transmission of a value along a wire or channel. Generally we will compare and combine systems over the same alphabet.

2.1. Operators on transition systems.

There are several familiar ways of combining transition systems to form new ones. We concentrate on those needed in our examples.

2.1.1. Prefixing.

Let $\Delta \subseteq \Sigma$, and let S_σ be a transition system for $\sigma \in \Delta$. then

$$\sigma : \Delta \to S_\sigma$$

is the system with a new initial state q_0, and a transition on σ from this state to the initial state of S_σ for each $\sigma \in \Delta$. (It is presumed that the state sets in question are disjoint.) The event σ is bound in this construction. A simple special case occurs when Δ is the singleton $\{\sigma\}$. This system is denoted $\sigma \to S$.

2.1.2. Synchronized parallel composition.

Again, let $\Delta \subseteq \Sigma$, and let S and T be transition systems over Σ. This time, the alphabet Δ is regarded as specifying a collection of events which must occur simultaneously in S and T. It is called the ***synchronization alphabet.*** Other events (in $\Sigma \setminus \Delta$) may occur independently in S or T. We define $S \|_\Delta T$ to be the system

$$\langle Q_S \times Q_T, \langle q_S, q_T \rangle, \{\xrightarrow{\sigma}\}\rangle,$$

where $\langle q, r\rangle \xrightarrow{\sigma} \langle q', r'\rangle$ iff either

$$\sigma \in \Delta, \quad q \xrightarrow{\sigma} q', \quad \text{and } r \xrightarrow{\sigma} r'$$

or

$$\sigma \notin \Delta, \quad \text{and either } q \xrightarrow{\sigma} q' \text{ with } r = r'$$
$$\text{or } r \xrightarrow{\sigma} r' \text{ with } q = q'$$

Notice that if Δ is empty we get the "shuffle" of the two systems, while if $\Delta = \Sigma$ we get the direct product.

2.1.3. Hiding or concealment.

Often we wish to ensure that an action a is not available to the environment of a process for synchronization purposes. This is accomplished by renaming the event a to be the silent event τ. This operation will be denoted hide_a, and is formally given by letting the $\xrightarrow{a}$ relation be empty and adding those pairs to the $\xrightarrow{\tau}$ relation.

2.1.4. Renaming.

This operation is used in our examples to direct information being sent on a particular channel to some other channel. It is specified by a renaming map $r : \Sigma \to \Delta$. The result of applying this to a transition system is another system over the alphabet Δ which will allow a transition $q \xrightarrow{b} q'$ iff there is some a with $r(a) = b$ and $q \xrightarrow{a} q'$ (the state set of the new system is unchanged.)

2.2. Logic for process description.

We employ a modal logic L of Hennessy and Milner [HM] modified slightly for our present purposes. The formulas of L are defined to be the least set containing the Boolean constants tt and ff , closed under the usual Boolean operations, and under the rule stating that $\langle a\rangle\phi$ and $\langle\Lambda\rangle\phi$ are formulas whenever $a \in \Sigma$ and ϕ is a formula.

Intuitively, the meaning of these modal formulas is that it is possible to execute a single a event and then be in a state satisfying ϕ; or that it is possible to execute a sequence of silent events and then to satisfy ϕ. To explain the semantics precisely we need to define some auxiliary relations. Let the relation $\stackrel{\Lambda}{\Longrightarrow}$ be the reflexive, transitive closure of the $\xrightarrow{\tau}$ relation. Let the relations $\stackrel{a}{\Longrightarrow}$, for $a \in \Sigma$, be

$$\left(\stackrel{\Lambda}{\Longrightarrow}\right) \circ \xrightarrow{a} \circ \left(\stackrel{\Lambda}{\Longrightarrow}\right).$$

(For $u \in \Sigma^*$ we also have the natural relations $\stackrel{u}{\Longrightarrow}$.)

We next define the *satisfaction predicate* $\models$. Let S be a system and $q \in Q_S$. The predicate $\models$ is the least relation between Q_S and L satisfying:

$$q \models \text{tt} \quad \text{always;}$$
$$q \models \text{ff} \quad \text{never;}$$
$$q \models \phi \vee \psi \text{ iff } q \models \phi \text{ or } q \models \psi;$$

(Similarly for the other Boolean operations)

$$q \models \langle a \rangle \phi \text{ iff } (\exists q')(q \stackrel{a}{\Longrightarrow} q' \wedge q' \models \phi);$$
$$q \models \langle \Lambda \rangle \phi \quad \text{similarly.}$$

Definition 2.2.1 (modal rank.)

Let

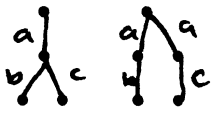
$$|\text{ tt }| = |\text{ ff }| = 0;$$
$$|\phi \vee \psi| = |\phi \wedge \psi| = max(|\phi|, |\psi|);$$
$$|\neg\phi| = |\phi|;$$
$$|\langle a \rangle \phi| = |\langle \Lambda \rangle \phi| = 1 + |\phi|.$$

Definition 2.2.2. Let $L(n, \Delta)$ be the set of all ϕ with $|\phi| \leq n$ and modalities $\langle a \rangle$ with $a \in \Delta$. Two states q and r are n-elementarily equivalent iff $\text{Th}_n(q) = \text{Th}_n(r)$, where

$$\text{Th}_n(q) = \{\phi \in L(n, \Sigma) : q \models \phi\}$$

Let E_n be the relation just defined. Let $q\ E\ r$ iff $q\ E_n\ r$ for all n. Similarly let E_n^Δ and E^Δ be these relations induced by formulas with modalities restricted to Δ.

Example. Consider the two systems



where the initial state is at the root of the tree. These systems are E_1 but not E_2–equivalent.

We also have another useful and closely related definition of equivalence.

Definition 2.2.4. Two states p and q are 0-behaviorally equivalent trivially. They are $(n+1)$-behaviorally equivalent iff for all $\sigma \in \Sigma \cup \{\tau\}$, and for all p' such that $p \stackrel{\sigma}{\Longrightarrow} p'$, we have

$$(\exists q')(q \stackrel{\sigma}{\Longrightarrow} q' \wedge q' \text{ is n-behaviorally equivalent to } p') \text{ and vice versa.}$$

Let B_n denote the relation of n-behavioral equivalence, and let B be the intersection of all the B_n.

Lemma 2.2.5. If Δ is finite, then among the formulas of $L(n, \Delta)$, there are only finitely many logically distinct ones.

Proof. This is found in [GR]; one merely has to put the formulas of $L(n, \Delta)$ into a normal form.

Lemma 2.2.6. (Master formula theorem for L.) For each finite Δ, and each state q, there is a "master formula" $M\phi(n, \Delta, q)$ such that (i) $q \models M\phi(n, \Delta, q)$, and (ii) for all r, if $r \models M\phi(n, \Delta, q)$ then $r\ E_n^\Delta\ q$.

Proof. Let $\theta_1, \ldots, \theta_k$ be the logically distinct representatives of the set of (n, Δ)-formulas satisfied by q. Then we may take

$$M\phi(n, \Delta, q) = \bigwedge_{i=1}^{k} \theta_i.$$

Theorem 2.2.7. If Σ is finite, then $B_n = E_n$ for all n.

Proof. It is easy to show that B_n equivalence implies E_n equivalence. The other direction is by induction on n. This is trivial for $n = 0$. Assume it for n, and let $p\ E_{n+1}\ q$. We want to show $p\ B_{n+1}\ q$. Let $p \stackrel{\sigma}{\Longrightarrow} p'$ and let $\theta = M\phi(n, \Sigma, p')$. Then $p \models \langle\sigma\rangle\theta$. So therefore does q, and Lemma 2.2.6 and the induction hypothesis complete the proof.

Theorem 2.2.8. If Σ is finite, then the E_n relations are congruences with respect to the operations in Section 2.1, except hiding.

Proof. The result is straightforward to show if we use B_n instead of E_n, by induction on n. Applying 2.2.7 then gives the desired conclusion.

The relationship between E_n and B_n is more interesting when Σ is infinite; in this case the relations do not agree. However, we can characterize the distinction as follows.

Definition 2.2.9. For $\Delta \subseteq \Sigma$, let $RUN(\Delta)$ be the transition system with one state q, and a transition $q \stackrel{a}{\Longrightarrow} q$ for each $a \in \Delta$. Let p be a state in an arbitrary.transition system over Σ. We define

$$p \parallel \Delta = p \parallel_\Sigma RUN(\Delta).$$

This operator disallows any transitions of p which are not in the set Δ. (By the transition system p we understand the transition system in which p occurs, redefined to have p as its initial state.)

Lemma 2.2.10. For any finite Δ, and any $n \geq 0$, we have for all states p and q,

$$p\ E_n^\Delta\ q \iff (p \parallel \Delta)\ B_n\ (q \parallel \Delta).$$

Proof. ($\Rightarrow$.) We proceed by induction on n to show that $p\ E_n^\Delta\ q$ implies $(p \parallel \Delta)\ B_n (q \parallel \Delta)$ for all p and q. The case $n = 0$ is trivial. Assume the result for n. Let $(p \parallel \Delta) \stackrel{\sigma}{\Longrightarrow} (q \parallel \Delta)$. The proof is the same whether or not σ is visible. We know from Def. 2.2.9 that σ, if visible, is in Δ, and that there is some p' such that $p \stackrel{\sigma}{\Longrightarrow} p'$. Therefore

$$p \models \langle\sigma\rangle M\phi(n, \Delta, p').$$

Since $p\ E_{n+1}^\Delta\ q$, there is a state q' such that $q \stackrel{\sigma}{\Longrightarrow} q'$ and $q' \models M\phi(n, \Delta, p')$. By Lemma 2.2.6, $q'\ E_n^\Delta\ p'$. By induction hypothesis, $(p' \parallel \Delta)\ B_n\ (q' \parallel \Delta)$. Since $(q \parallel \Delta) \stackrel{\sigma}{\Longrightarrow} (q' \parallel \Delta)$, we have the result for the case $n + 1$.

($\Leftarrow$.) The converse is another **induction on** n. We show: for all p and q, $(p \parallel \Delta)\ B_n\ (q \parallel \Delta)$ implies $p\ E_n^\Delta\ q$. Again the basis is trivial. For the **inductive step**, assume that $(p \parallel \Delta)\ B_{n+1}\ (q \parallel \Delta)$. The conclusion requires $p\ E_{n+1}^\Delta\ q$. This is established by **another induction** over $L(\Delta, n + 1)$ formulas. For each ϕ in $L(\Delta, n + 1)$ we must establish

$$p \models \phi \iff q \models \phi.$$

This is clear when ϕ is tt or ff . If ϕ is a Boolean combination of smaller formulas, the result follows easily from the induction hypothesis for those formulas. Suppose, therefore, that ϕ is $\langle\sigma\rangle\psi$. Let $p \models \phi$. Then there is a p' such that $p \stackrel{\sigma}{\Longrightarrow} p'$ and $p' \models \psi$. If σ is visible then of course $\sigma \in \Delta$, and therefore $(p \parallel \Delta) \stackrel{\sigma}{\Longrightarrow} (p' \parallel \Delta)$. This gives a q' such that $(q \parallel \Delta) \stackrel{\sigma}{\Longrightarrow} (q' \parallel \Delta)$, and $(p' \parallel \Delta)\ B_n\ (q' \parallel \Delta)$. The overall induction hypothesis applies, showing that $p'\ E_n^\Delta\ q'$. Thus $q' \models \psi$, and so $q \models \langle\sigma\rangle\psi$. The reverse implication is the same, so that we have proved the whole lemma.

Theorem 2.2.11. For any states p and q in a transition system over Σ, $p\ E_n\ q$ iff for every finite $\Delta \subseteq \Sigma$, we have $(p \parallel \Delta)\ B_n\ (q \parallel \Delta)$. The same holds for B and E.

Proof. This is immediate from Lemma 2.2.10.

Theorem 2.2.12. The conclusions of Theorem 2.2.8 hold without the hypothesis of finite Σ.

Proof. We show that E_n is a congruence for synchronized parallel composition, leaving the other operators as an exercise. Let Γ be the synchronization alphabet. Suppose $p\ E_n\ q$. We want: for all r, $(p \parallel_\Gamma r)\ E_n\ (q \parallel_\Gamma r)$.

Let Δ be a finite subalphabet of Σ. Then $(p \parallel \Delta)\ B_n\ (q \parallel \Delta)$. These systems can be regarded over the finite alphabet Δ, and so by 2.2.8, we have

$$(p \parallel \Delta) \|_\Gamma r \quad B_n \quad (q \parallel \Delta) \|_\Gamma r$$

for any r. It is easy to see that

$$(p \parallel \Delta) \|_\Gamma r \quad B_n \quad (p \|_\Gamma r) \parallel \Delta,$$

so we have the desired conclusion for all finite Δ. Applying 2.2.11 gives the result.

2.3. The metric space of transition systems.

Using the equivalences in 2.2 we can define the distance between two states in a transition system, and therefore also the distance between two transition systems.

Definition 2.3.1. Let p and q be states in a transition system. The E-distance $d_E(p, q)$ is

$$\inf\{2^{-n} : p\ E_n\ q\}.$$

It is easy to check that d_E makes Υ/E into a metric space, where Υ is the class of (countable) transition systems over some fixed countable alphabet, and Υ/E is the quotient by the E equivalence relation. The same remark holds for the relation B. We wish, however, to introduce yet another topological structure on transition systems. This is done by using the *Stone topology* associated with the logic L. The points of this space are the elementary equivalence classes p/E. We choose as a basis for the space the sets of the form

$$\mathrm{Mod}(\theta) = \{\, p/E : p \models \theta\}$$

where $\theta \in L$. This collection of sets is closed under the Boolean operations and so by itself forms a base for some topology. It will be convenient to consider the *closed sets* in this topology, which by definition are of the form

$$\mathrm{Mod}(\Gamma) = \bigcap_{\theta \in \Gamma} \mathrm{Mod}(\theta).$$

If Σ is finite, the Stone topology is the same as that given by the metric d_E.

Theorem 2.3.2. A subset K of Υ/E is closed iff it is metrically closed in the E-metric, provided Σ is finite.

Proof. Suppose $K = \mathrm{Mod}(\Gamma)$ for some set of formulas Γ. Let $p_n \in K$ and suppose $d(p_n, p) \to 0$. (The relation E will be suppressed from now on.) If $p \notin K$, then for some $\theta \in \Gamma$, $p \models \neg\theta$. Choose n bigger than $|\theta|$, and p_m so that $d(p_m, p) \leq 2^{-n}$. Then $p_m \models \theta$ and $p_m \models \neg\theta$, a contradiction.

Conversely, let K be a closed set in the metric sense. Define

$$\mathrm{Th}(K) = \{\theta : (\forall s \in K)(s \models \theta)\}.$$

Claim: $K = \mathrm{Mod}(\mathrm{Th}(K))$. The inclusion of left in right is trivial. For the other inclusion, suppose $t \in \mathrm{Mod}(\mathrm{Th}(K))$. We will construct a sequence $s_n \in K$ such that $d(s_n, t) \to 0$, which will prove $t \in K$. Fix the value of n. Consider the formulas $M\phi(n, \Sigma, s)$ as s ranges over K. By Lemma 2.2.5, there are only finitely many logically distinct such formulas. Set

$$\phi_n = \bigvee_{s \in K} M\phi(n, \Sigma, s).$$

Now $s \models \phi_n$ for each $s \in K$, which implies $\phi_n \in \mathrm{Th}(K)$. Since $t \in \mathrm{Mod}(\mathrm{Th}(K))$, we have $t \models \phi_n$. Thus for each n, there is an $s_n \in K$ with $t \models M\phi(n, \Sigma, s_n)$. Therefore $t\ E_n\ s_n$, which implies $d(s_n, t) \to 0$, and the proof is complete.

We can now state and apply the Compactness Theorem for L. Its use will be in the next sections, and principally in Section 4.

Theorem 2.3.3 (Compactness). Let Γ be a subset of L. If every finite subset F of Γ has a countable model (i.e. a p such that $p \in \mathrm{Mod}(F)$), then so does Γ. :

Proof. This proof is given in [GR], where it is shown how to translate L into first-order logic. Only minor modifications to that proof are necessary to deal with silent transitions.

Corollary 2.3.4. If Σ is finite, then the space Υ/E with the metric d_E is a compact metric space.

Proof. It is a standard fact from logic that the Compactness Theorem states that the Stone topology is compact. Since the Stone topology agrees with the E-topology, the result follows.

Remark. The compactness theorem itself does not depend on the finiteness of Σ. This will be important in Section 4.

3. Applications to buffer specifications

We present the definition of buffers appearing in Hoare, Brookes, and Roscoe. Their explanation goes roughly as follows. A buffer is a process which accepts messages on its input channel, and emits them in the same order on its output channel. The buffer should not have any other events allowed in its behavior. For the actual specification, we need to impose some structure on the set of events Σ.

Let T be a set of *values*. For example, T might be the set of integers. Let C be a set of *channel names* (e.g., $C = \{\,\mathrm{in}, \mathrm{out}\,\}$.) A pair $\langle c, t\rangle$, to be written $c.t$, denotes the event of passing the message with value t along channel c. For buffer specifications, the channel names will be in the set $\{\,\mathrm{in}, \mathrm{out}\,\}$. We will also use a renaming operator which maps the event $c.t$ to t. This operator is called *restriction* to c, and we write $r(c.t, c) = t$. If an event σ is not in $C \times T$, then $r(\sigma, c) = \sigma$. This operation is extended to strings of events in the usual way.

One other concept is needed about transition systems: for a state q, we define

$$\mathrm{Initials}(q) = \{a \in \Sigma : (\exists q')(q \stackrel{a}{\Longrightarrow} q')\}.$$

Now we can formalize buffers.

Definition 3.1. A *buffer* is a transition system over $\Sigma = \mathrm{in}.T \cup \mathrm{out}.T$, with initial state b, such that for all $s \in \Sigma^*$ and states q such that $b \stackrel{s}{\Longrightarrow} q$, we have
(i.) $r(s, \mathrm{out}) \leq r(s, \mathrm{in})$; ($\leq$ is the prefix relation)
(ii.) If $r(s, \mathrm{out}) = r(s, \mathrm{in})$ then $\mathrm{Initials}(q) = \mathrm{in}.T$;
(iii.) if $r(s, \mathrm{out}) \neq r(s, \mathrm{in})$ then $\mathrm{Initials}(q) \cap \mathrm{out}.T \neq \emptyset$.

The first condition states that outputs appear in the order in which they came in, and that no output appears before it has been input. The second clause states that an empty buffer must be prepared to accept any input in the set T, and the third states that a nonempty buffer must be prepared to output some value in the set T. Note that the condition stating that every value which has come in must eventually come out is not explicitly expressed in this definition; it is possible to input values forever, provided that the buffer has unbounded capacity.

We are going to prove that the class of buffers is a closed set in the Stone topology defined in Section 2. This will have the consequence that the rule of fixed-point induction works to establish properties of buffers. These conclusions appear in [HBR] and [Ros] ; what is new here is the topology and the method of proof using logical specifications.

Lemma 3.1.2. Let K be a closed set in the Stone topology, and let $L \subseteq \Sigma^*$. Then the set K_1 of (equivalence classes of) systems p such that for all $s \in L$ and all q such that $p \stackrel{s}{\Longrightarrow} q$, we have $q \in K$, is also closed.

Proof. We know that $K = \mathrm{Mod}(\Gamma)$ for some set Γ of formulas. Let $\langle s\rangle\phi$ be the formula $\langle\sigma_1\rangle \ldots \langle\sigma_n\rangle\phi$, where $s = \sigma_1 \ldots \sigma_n$, and let $[s]\phi$ be $\neg\langle s\rangle\neg\phi$. Then

$$K_1 = \{[s]\phi : s \in L \text{ and } \phi \in \Gamma\}.$$

Corollary 3.1.3. The set K_1 of processes p such that for all q, if $p \stackrel{s}{\Longrightarrow} q$, then $s \in L$, where $L \subseteq \Sigma^*$, is also closed.

Proof. Let $K = \emptyset$ and let L be replaced by the complement of L in Lemma 3.1.2.

Lemma 3.1.4. The set K_2 of processes q such that Initials$(q) = \Delta$, where $\Delta \subseteq \Sigma$, is a closed set.

Proof. Define

$$\Gamma = \{\langle a\rangle \text{ tt } : a \in \Delta\} \cup \{\neg\langle a\rangle \text{ tt } : a \notin \Delta\}.$$

It is clear that $K_2 = \text{Mod}(\Gamma)$.

Lemma 3.1.5. If Δ is finite, then the set $K_3 = \{q : \text{Initials}(q) \cap \Delta \neq \emptyset\}$ is closed.

Proof. Let Γ consist of the single formula $\langle a_1\rangle \text{ tt } \vee \ldots \vee \langle a_n\rangle \text{ tt }$, where $\Delta = \{a_1, \ldots, a_n\}$. Then $K_3 = \text{Mod}(\Gamma)$.

Theorem 3.1.6. If T is finite, then the class of buffers over T is closed in the Stone topology.

Proof. This is now immediate from Definition 3.1.1 and the foregoing lemmas.

Remark. Lemma 3.1.5 is false for infinite Δ and the Stone topology. This is the only lemma that fails, however. We shall assume that the event set is finite for the remainder of the section.

3.2. Examples in CSP.

Let c be a channel name. We define the operator $hide_c$ to be the renaming of each $c.t$ event to the silent transition τ, and otherwise the identity. Now let p and q be processes over the buffer alphabet. We define the *chaining* of p and q, written $p \gg q$, as follows. Let c be a new channel name. Let p(out $\leftarrow c$) be the process p except that all output events out.t have been renamed to $c.t$. Similarly, let q(in $\leftarrow c$) be the process where all input events in.t have been renamed to $c.t$. Then we define

$$p \gg q = hide_c(\ p(\text{out} \leftarrow c) \parallel_\Delta q(\text{in} \leftarrow c)\ )$$

where $\Delta = c.T$. This has the effect of connecting the output of p directly to the input of q, and then hiding the communications on the new channel. It allows the interaction of p with the environment by means of the input channel, and of q with the environment by means of the output channel. One can prove, as do Hoare, Brookes, and Roscoe, that if p and q are buffers, then so is $p \gg q$.

Now we want to consider the effect of recursive applications of the chaining operator. This will allow the definition of buffers with unbounded capacity. First, consider the one-place buffer B1, which inputs a value of type T, then outputs it, and repeats the process. It is easy to describe B1 as a transition system. In CSP, the process B1 would be expressed

$$*(?x.T \rightarrow !x).$$

As an example of a recursive definition, consider the equation

$$p = ?x.T \rightarrow (p \gg (!x \rightarrow \text{ B1 })).$$

Let $F(p)$ be the function of p denoted by the right-hand side of this equation. In the notation of Section 2, F would be defined

$$F(p) = \sigma : \text{in}.T \rightarrow S(p, \sigma)$$

where

$$S(p, \text{in}.t) = p \gg (\text{out}.t \rightarrow \text{ B1 }).$$

Definition 3.2.1. A function F on transition systems is said to be *constructive* iff whenever $p\ E_n\ q$, it follows that $F(p)\ E_{n+1}\ F(q)$.

Lemma 3.2.2. The function F defined above is constructive. Furthermore, if p is a buffer, so is $F(p)$.

Proof. Omitted here, but see [HBR] for some details. It should be noted that the chaining operator involves hiding, which is in general a nonconstructive operation. Special properties of the function F must be used in the proof.

The significance of constructive functions lies in their metric space formulation.

Definition 3.2.3. Let F be a map from a metric space to itself. F is said to be *contractive* iff there is a number α with $0 \leq \alpha < 1$ such that for all x and y,

$$d(F(x), F(y)) \leq \alpha \cdot d(x, y).$$

Corollary 3.2.4. A function F is contractive in the E-metric iff it is constructive.

Now we recall the well-known Banach fixed-point theorem.

Theorem 3.2.5. Let F be a contractive mapping of a complete metric space to itself. Let B be a nonempty closed subset of the space. Suppose further that whenever $b \in B$, we have $F(b) \in B$. Then F has a unique fixed point which also belongs to B.

Putting all the above results together, and remembering that the space Υ/E is compact, and therefore complete, we deduce that the function F defined above has a unique fixed point, up to E-equivalence, and this fixed point is also a buffer. Many other examples of recursive constructions of buffers can be found in [HBR], and they can all be treated in this way.

4. A cpo for processes over an infinite alphabet

In [Rou] it is shown that the class of *synchronization forests* forms a complete partial order (in fact, a Scott domain) under reverse set inclusion. A synchronization forest is a collection of synchronization trees, and a synchronization tree is just a transition system whose graph is a tree. It is required in [Rou] that a synchronization forest be closed in the B-metric, and that a synchronization tree be over a finite alphabet with no silent transitions. Here we remove the latter two restrictions on synchronization trees, and we state our results for general transition systems.

One of the technical contributions in [Rou] was the construction of cpo-continuous functions from metrically continuous functions in the topology of synchronization trees. Specifically, it was shown that whenever F is a metrically continuous operation (unary, binary, etc.) on a compact metric space X, then the *direct image* map $\lambda K.F[K]$ is a cpo-continuous operation on the class of all nonempty closed subsets of X, ordered by reverse set inclusion. The topology used in [Rou] was the B-metric topology, which is compact if the alphabet is finite. This topology is *not* compact in the case of an infinite alphabet, and so the question arises about the existence of a suitable extension theorem for the direct image mapping in the infinite alphabet case.

Our approach is to use the weaker Stone topology in order to retain the compactness property of the underlying space. Although this topology is metrizable, it is not necessary to use the metric! The topological definition of continuity (the inverse image of a closed set is closed) will suffice for our application, and in fact is extremely natural when used in conjunction with logical specifications.

As an example, we will use the synchronized composition operator. In what follows, the space X will be Υ/E with the Stone topology. Let $\Delta \subseteq \Sigma$ and let

$$F(p, q) = p \parallel_\Delta q.$$

It follows from Theorem 2.2.12 that E is a congruence with respect to this operation, and so F is well-defined on the space $X \times X$.

Lemma 4.1. The function F is a continuous map from $X \times X$ to X.

Proof. Recall that a basis for the topology of X is given by the collection of sets $\mathrm{Mod}(\theta)$, where $\theta \in L$. Let $\mathcal{B}$ be the collection of finite unions of sets of the form $\mathrm{Mod}(\theta_i) \times \mathrm{Mod}(\theta_j)$ as θ_i and θ_j range over L. Then

$\mathfrak{B}$ is a closed basis for the product space. It suffices to show that the inverse image of a basis set for X is a member of $\mathfrak{B}$. We define, for a formula ϕ in L, the set

$$K(\phi) = \{\langle t, u\rangle : F(t, u) \models \phi\}.$$

We are required to show that $K(\phi) \in \mathfrak{B}$ for all ϕ. This we do by induction on formulas. The result is clear for $\phi =$ tt , and if it holds for ϕ and ψ then it clearly holds for $\phi \vee \psi$. Consider the set $K(\neg\phi)$. By induction hypothesis, this is the complement of a set in $\mathfrak{B}$. Therefore this set can be written as a finite intersection of sets of the form $U_i \cup U_j$, where

$$U_i = \{\langle t, u\rangle : t \models \neg\theta_i\}$$

and

$$U_j = \{\langle t, u\rangle : u \models \neg\theta_j\}.$$

By distributivity, the set can be rewritten as a finite union of sets of the form $U_i \cap U_j$. But

$$U_i \cap U_j = \mathrm{Mod}(\neg\theta_i) \times \mathrm{Mod}(\neg\theta_j),$$

and so $K(\neg\phi) \in \mathfrak{B}$.

Finally, consider the case $\phi = \langle a\rangle\psi$. Suppose that $a \in \Delta$. Then

$$F(t, u) \models \langle a\rangle\psi \quad \text{iff} \quad \exists\langle t', u'\rangle : t \stackrel{a}{\Longrightarrow} t',\ u \stackrel{a}{\Longrightarrow} u', \text{ and } F(t', u') \models \psi.$$

Let

$$K(\psi) = \bigcup_{\langle i,j\rangle \in G} \mathrm{Mod}(\theta_i) \times \mathrm{Mod}(\theta_j)$$

where G is a finite set of pairs. Then

$$K(\langle a\rangle\psi) = \bigcup_{\langle i,j\rangle \in G} \mathrm{Mod}(\langle a\rangle\theta_i) \times \mathrm{Mod}(\langle a\rangle\theta_j).$$

Consider the case where $a \notin \Delta$. Then $F(t, u) \models \langle a\rangle\psi$ iff one of two subcases occurs:

$$\exists\langle t', u'\rangle : t \stackrel{a}{\Longrightarrow} t' \text{ and } u \stackrel{\Lambda}{\Longrightarrow} u', \text{ and } F(t', u') \models \psi,$$

or the subcase with the roles of t and u reversed. Let

$$K_1 = \bigcup_{\langle i,j\rangle \in G} \mathrm{Mod}(\langle a\rangle\theta_i) \times \mathrm{Mod}(\langle\Lambda\rangle\theta_j)$$

to account for the first subcase, and

$$K_2 = \bigcup_{\langle i,j\rangle \in G} \mathrm{Mod}(\langle\Lambda\rangle\theta_i) \times \mathrm{Mod}(\langle a\rangle\theta_j)$$

to account for the second. Then $K(\phi) = K_1 \cup K_2$. This completes the proof of Lemma 4.1.

We now state a general extension theorem. The proof is standard.

Theorem 4.2. Let F be a continuous function from a compact Hausdorff space Z to a space Y. Define, for a closed subset K of Z,

$$F[K] = \{F(z) : z \in K\}.$$

Let K_i be a decreasing sequence of closed nonempty subsets of Z. Then the intersection of all the K_i is nonempty, and

$$F\left[\bigcap K_i\right] = \bigcap F[K_i].$$

This result gives us a way to construct continuous functions on the cpo of closed subsets of a compact space like X. For binary operations like parallel composition, we can simply define

$$F(K_1, K_2) = F[K_1 \times K_2]$$

Since the space $X \times X$ is compact, the extension theorem applies with $Z = X \times X$ together with Lemma 4.1 to give cpo-continuous mappings. We have thus generalized the results of [Rou] to the infinite alphabet case.

References

[BaZ] J. W. de Bakker and J. F. Zucker, "Processes and the denotational semantics of concurrency," *Proc. 14th ACM Symp. on Theory of Computing*, 153–158.

[GR] W. Golson and W. Rounds, "Connections between two theories of concurrency: metric spaces and synchronization trees," Tech. Rep. CRL–TR–3–83, Computing Research Laboratory, University of Michigan, 1983 (to appear in *Inf. Control*)

[HM] M. Hennessy and R. Milner, "On observing Nondeterminism and Concurrency," *Proc. 7th ICALP*, LNCS **85** (1980).

[HBR] C. A. R. Hoare, S. D. Brookes, amd W. Roscoe, "A mathematical model for communicating processes," Report PRG–16, Programming Research Group, Oxford (1981). Also to appear in *JACM*.

[M] R. Milner, "A calculus of communicating systems," LNCS **92**, (1980).

[OH] E. Olderog and C. A. R. Hoare, "Specification–oriented semantics for communicating processes," *Proc. 10th ICALP*, LNCS **154** (1983), 561–572.

[Ros] W. Roscoe, "A mathematical theory of communicating processes," D.Phil. thesis, Oxford (1982).

[Rou] W. Rounds, "On the relationships between Scott domains, synchronization trees, and metric spaces," Tech. Rep. CRL–TR–25–83, Computing Research Laboratory, University of Michigan (1983).

[S] M. Smyth, "Power domains and predicate transformers: a topological view," *Proc. 10th ICALP*, LNCS **154** (1983), 662–675.