Deriving Bisimulation Congruences for Reactive Systems

James J. Leifer and Robin Milner

University of Cambridge Computer Laboratory
New Museums Site, Pembroke St., Cambridge CB2 3QG, UK
{James.Leifer, Robin.Milner}@cl.cam.ac.uk

Abstract. The dynamics of reactive systems, e.g. CCS, has often been defined using a labelled transition system (LTS). More recently it has become natural in defining dynamics to use reaction rules — i.e. unlabelled transition rules — together with a structural congruence. But LTSs lead more naturally to behavioural equivalences. So one would like to *derive* from reaction rules a suitable LTS.

This paper shows how to derive an LTS for a wide range of reactive systems. A *label* for an agent a is defined to be any context F which intuitively is just large enough so that the agent Fa ("a in context F") is able to perform a reaction. The key contribution of this paper is a precise definition of "just large enough", in terms of the categorical notion of *relative pushout* (RPO), which ensures that bisimilarity is a congruence when sufficient RPOs exist. Two examples — a simplified form of action calculi and term-rewriting — are given, for which it is shown that sufficient RPOs indeed exist. The thrust of this paper is, therefore, towards a general method for achieving useful behavioural congruence relations.

1 Purpose

The semantics of interactive systems is in a state of flux, inevitably so because new models for such systems are constantly appearing. Frequently, a calculus is developed to model certain features (e.g. communication, mobility and security) and the behaviour of agents is described in terms of state transition rules, also called reduction rules, rewriting rules, firing rules, etc.; we shall call them *reaction rules*. The question of behavioural equivalence between two agents immediately arises.

A sledgehammer approach to behavioural equivalence is in terms of *contexts*. It is often easy to determine, for a calculus, the class of all possible contexts C in which agents may appear; then we can declare that two agents a and b are *contextually equivalent* — here written $a \sim b$ — iff for all contexts C the agents Ca and Cb have the same reaction pattern (which may be defined differently for different kinds of equivalence). This definition has the advantage of making $\sim$ a congruence ($a \sim b$ implies $Ca \sim Cb$), and the disadvantage that to check equivalence one has to consider *all* contexts.

A common and more practical approach has been to define (by rules) not only the *reactions* $a \longrightarrow\!\!\triangleright\, a'$ of each agent, but also a system of *labelled transitions* $a \xrightarrow{\lambda}\!\!\triangleright\, a'$, where the *label* λ is drawn from some tractable set representing all the “ways” in which an agent may interact with its environment. We may then define $a \sim b$ to mean that a and b have the same pattern (traces, bisimilarity, ...) of labelled transitions, not merely the same pattern of reactions. But we are still faced with proving a congruence property with respect to some class of contexts; this proof may be hard, and is often ad hoc for each calculus.

This paper offers a general method for deriving a labelled transition system (LTS) whose labels are a restricted class of contexts. The crux of the paper is that these labels are defined in terms of the categorical notion of *relative pushout* (RPO), and that the induced bisimulation equivalence (either strong or weak) is guaranteed to be a congruence when sufficient RPOs exist.

2 Background and outline

Since the early days of process calculi, the question of behavioural equivalence has been central. It has usually been operationally defined, and often centred upon an LTS; this was the case with CCS [15]. There have indeed been notable exceptions to the use of LTSs as the defining method: Hoare's CSP [12] was given an elegant denotational semantics, the *failures* model; in the Process Algebra [3] which originated with Bergstra and Klop the emphasis was upon an algebraic theory rather than upon transitions. But LTSs have been prominent, and they led to an intense study of the different equivalences they induce [9], and of their congruential properties [10, 24].

With the π-calculus [18] the LTS methodology became strained because the passage of names as messages required a somewhat ad hoc structure in the labels. For this reason Milner [16], inspired by the Chemical Abstract Machine of Berry and Boudol [4], devised an alternate semantics based upon structural congruence and reaction rules, with specific definitions of behavioural equivalence and specific congruence proofs, often based upon barbed bisimulation [19].

Simultaneously, action calculi [17] were proposed as a framework embracing a wide variety of process calculi. Many calculi, including the λ-calculus, the π-calculus, Petri nets and the Ambient calculus can be presented as action calculi, which employ a uniform notion of structural congruence. Thus arose the challenge to find a general way of deriving LTSs, and thence behavioural congruences, from reaction rules all expressed within action calculi.

Sewell [22] has derived an LTS for several classes of reactive system, and in each case proved the induced bisimilarity to be a congruence. He also proposed a notion of *colouring* to keep track of component occurrences, and thereby to yield satisfactory congruences. This work has given guidance on what a uniform approach might be, and on which congruences it should yield. We here offer a uniform approach applying to any reactive system which forms a category possessing *relative pushouts*; in our ongoing work we aim to demonstrate that action calculi enjoy this property.

Outline In the next section we discuss the derivation of LTSs and motivate the use of contexts as labels. In Section 4 we define the notion of *relative pushout* (RPO) and the sister notion of an *idem pushout* (IPO) — a self-relative RPO. In

Section 5, we define the labelled transitions of an LTS in terms of IPOs. We then prove that the associated strong bisimilarity is a congruence; we also show that a weak bisimilarity is a congruence. In Sections 6 and 7 we study two examples: a simple class of graphs related to action calculi [17], and term-rewriting with "multi-hole" contexts, in comparison with Sewell's study [22]. Current and future work is discussed in Section 8.

3 Motivation

We wish to answer two questions about arbitrary reactive systems consisting of agents (whose syntax may be quotiented by a structural congruence) and a reaction relation $\longrightarrow$ (generated by reaction rules):

1. Can we *derive* a labelled transition relation $\xrightarrow{\lambda}$ where λ comes from a small set of labels that intuitively reflect how an agent interacts with its environment?
2. Under what general conditions is bisimulation over $\xrightarrow{\lambda}$ a congruence?

We can begin to address question 1 by considering CCS [15]. Let a, b range over agents (processes), C, D, F range over agent contexts (processes with a hole), and x range over names. The usual labelled transitions $\xrightarrow{\lambda}$ for $\lambda ::= \bar{x} \mid x \mid \tau$ reflect an agent's *capability* to engage in some behaviour, e.g. $\bar{x}.a|b$ has the labelled transition $\xrightarrow{\bar{x}}$ because $\bar{x}.a|b$ can perform an output on x. However, if we shift our emphasis from characterising the capabilities of an agent to the *contexts* that cause the agent to react, then we gain an approximate answer to question 1, namely we define

$$a \xrightarrow{F} a' \qquad \text{iff} \qquad Fa \longrightarrow a' \tag{1}$$

for all contexts F. (We denote context composition and application by juxtaposition throughout.) Instead of observing that $\bar{x}.a|b$ "can do an $\bar{x}$" we might instead see that it "interacts with an environment that offers to input on x, i.e. reacts when placed in the context $-|x$". Thus, $\bar{x}.a|b \xrightarrow{-|x} a|b$.

The definition of labelled transition in (1) is attractive when applied to an arbitrary process calculus because it in no way depends upon the presence or absence of structural congruence. Furthermore, it is generated entirely from the reaction relation $\longrightarrow$ (question 1); and, bisimulation over $\xrightarrow{F}$ is a congruence, (question 2). The proof of the latter is straightforward: let C be an arbitrary context and suppose $a \sim b$; we show that $Ca \sim Cb$. Suppose $Ca \xrightarrow{F} a'$; by definition, $FCa \longrightarrow a'$, hence $a \xrightarrow{FC} a'$. Since $a \sim b$, there exists b' such that $b \xrightarrow{FC} b'$ and $a' \sim b'$. Hence $Cb \xrightarrow{F} b'$, as desired.

Nonetheless, the definition in (1) is unsatisfactory: the label F comes from the set of *all* agent contexts — not the "small set" asked for in question 1 — thus making bisimulation proofs intolerably heavy. Also, the definition fails to capture its intended meaning that $a \xrightarrow{F} a'$ holds when a *requires* the context F to enable a reaction: there is nothing about the reaction $Fa \longrightarrow a'$ that forces all of F — or indeed any of F — to be used. In particular, if $a \longrightarrow a'$ then for all contexts F that preserve reaction, $Fa \longrightarrow Fa'$, hence $a \xrightarrow{F} Fa'$; thus a has many labelled transitions that reflect nothing about the behaviour of a itself.

Let us unpack (1) to understand in detail where it goes wrong. Consider an arbitrary reactive system equipped with a set Reacts of reaction rules; the reaction relation $\longrightarrow\!\!\!\triangleright$ contains Reacts and is preserved by all contexts:

$$l \longrightarrow\!\!\!\triangleright r \quad \text{if } (l,r) \in \mathsf{Reacts} \qquad\qquad \frac{a \longrightarrow\!\!\!\triangleright a'}{Ca \longrightarrow\!\!\!\triangleright Ca'} \quad .$$

Expanding (1) according to this definition of $\longrightarrow\!\!\!\triangleright$ we have:

$$\begin{aligned} a \xrightarrow{F}\!\!\!\triangleright a' \quad & \text{iff} \quad Fa \longrightarrow\!\!\!\triangleright a' \\ & \text{iff} \quad \exists (l,r) \in \mathsf{Reacts}, D. \quad Fa = Dl \;\&\; a' = Dr \quad . \end{aligned} \tag{2}$$

The requirement $Fa = Dl$ in (2) is rendered by a commuting square (as shown) in some category whose arrows are the agents and contexts of the reactive system. This requirement reveals the flaw described earlier: nothing in (2) forces F and D to be a "small upper bound" on a and l.

$$\begin{tikzcd} \cdot \arrow[r, "a"] \arrow[d, "l"'] & \cdot \arrow[d, "F"] \\ \cdot \arrow[r, "D"'] & \cdot \end{tikzcd}$$

For the past few years we have been searching for a result for action calculi which we call a "dissection lemma", having roughly the form: given $Fa = Dl$, there exists a "maximum" C, such that for some F' and D' we have $F'a = D'l$, $F = CF'$ and $D = CD'$. Sewell's already cited congruence proofs [22] indeed used dissection lemmas, even though they did not assert maximality. To capture our intuition of maximality, we construct below a category-theoretic framework, in which we then obtain an elegant and general proof of congruence for the induced bisimulation equivalence.

4 Relative pushouts

The standard way of characterising that F and D are a "least upper bound" for a and l is to assert that the square for (2) is a *pushout*, i.e. has the property: $Fa = Dl$, and for every F' and D' satisfying $F'a = D'l$ there exists a unique G such that $GF = F'$ and $GD = D'$, as shown.

$$\begin{tikzcd} \cdot \arrow[r, "a"] \arrow[d, "l"'] & \cdot \arrow[d, "F"] \arrow[ddr, bend left, "F'"] & \\ \cdot \arrow[r, "D"'] \arrow[drr, bend right, "D'"'] & \cdot \arrow[dr, dotted, "G"] & \\ & & \cdot \end{tikzcd}$$

Unfortunately, pushouts rarely exist in the categories that interest us. Consider, for example, a category of term contexts over a signature Σ; its objects consist of 0 and 1; its arrows $0 \to 1$ are terms over Σ; its arrows $1 \to 1$ are one-hole contexts over Σ; there are no arrows $1 \to 0$ and exactly one arrow $\mathsf{id}_0 : 0 \to 0$. Now, if Σ contains only constant symbols, say $\Sigma = \{\alpha, \alpha'\}$, then there is no pushout completing Fig. 1(1) because there are no contexts other than the identity. If we introduce a 2-place function symbol β into Σ, we can construct an upper bound for α and α' but still no pushout (Fig. 1(2)).

We now define *relative pushouts* (RPOs) which exist, unlike pushouts, in many categories of agent contexts (illustrated in later sections). Let $\mathbf{C}$ be an arbitrary category whose arrows and objects we denote by f, g, h, k and m, n; in pictures we omit labels on the objects when possible.

Definition 1 (RPO). *Given a commuting square (Fig. 2(1)) consisting of* $g_0 f_0 = g_1 f_1$*, an* RPO *is a triple* h_0, h_1, h *satisfying two properties:*

commutation: $h_0 f_0 = h_1 f_1$ *and* $h h_i = g_i$ *for* $i = 0, 1$ *(Fig. 2(2));*

$$\begin{tikzcd}
0 \arrow[r, "\alpha"] \arrow[d, "\alpha'"'] & 1 \arrow[d, dotted, "\nexists"] \\
1 \arrow[r, dotted, "\nexists"'] & 1
\end{tikzcd}$$

1(1)

$$\begin{tikzcd}
0 \arrow[rr, "\alpha"] \arrow[dd, "\alpha'"'] & & 1 \arrow[dl, "\beta(\alpha',-)"'] \arrow[dd, "\beta(-,\alpha')"] \\
& 1 \arrow[dr, dotted, "\nexists"] & \\
1 \arrow[ur, "\beta(-,\alpha)"] \arrow[rr, "\beta(\alpha,-)"'] & & 1 \arrow[ul, dotted, bend left, "\nexists"]
\end{tikzcd}$$

1(2)

Figure 1. Non-existence of pushouts

$$\begin{tikzcd}
\cdot \arrow[r, "f_0"] \arrow[d, "f_1"'] & \cdot \arrow[d, "g_0"] \\
\cdot \arrow[r, "g_1"'] & m
\end{tikzcd}$$

2(1)

$$\begin{tikzcd}
\cdot \arrow[rr, "f_0"] \arrow[dd, "f_1"'] & & \cdot \arrow[dl, "h_0"'] \arrow[dd, "g_0"] \\
& \cdot \arrow[dr, "h"] & \\
\cdot \arrow[ur, "h_1"] \arrow[rr, "g_1"'] & & m
\end{tikzcd}$$

2(2)

$$\begin{tikzcd}
\cdot \arrow[rrr, "f_0"] \arrow[ddd, "f_1"'] & & & \cdot \arrow[dll, "h_0"'] \arrow[dl, "h'_0"] \arrow[ddd, "g_0"] \\
& \cdot \arrow[r, dotted, "k"] \arrow[ddrr, "h"] & \cdot \arrow[ddr, "h'"] & \\
& & & \\
\cdot \arrow[uur, "h_1"] \arrow[uurr, bend right, "h'_1"'] \arrow[rrr, "g_1"'] & & & m
\end{tikzcd}$$

2(3)

Figure 2. Construction of an RPO

---

universality: for any h'_0, h'_1, h' satisfying $h'_0 f_0 = h'_1 f_1$ and $h' h'_i = g_i$ for $i = 0, 1$, there exists a unique k such that $h'k = h$ and $kh_i = h'_i$ (Fig. 2(3)).

(An RPO for Fig. 2(1) is just a pushout in the slice category of $\mathbf{C}$ over m.)

A square is called an *idem pushout* (IPO) if it has an RPO of a special kind:

Definition 2 (IPO). *The commuting square in Fig. 2(1) is an IPO if the triple g_0, g_1, id_m is an RPO.*

The difference between a pushout and an IPO is clearest in a partial order category: a pushout is a least upper bound (i.e. less than any other upper bound) and an IPO is a minimal upper bound (i.e. not greater than any other upper bound). IPOs form the basis of our abstract definition of labelled transition and, by the following proposition, their existence follows from that of RPOs:

Proposition 1 (IPOs from RPOs). *If Fig. 2(2) is an RPO diagram then the square in Fig. 3(1) is an IPO.*

IPOs can be pasted together as shown by the following proposition, analogous to the standard pasting result for pushouts:

$$\begin{tikzcd} \cdot \arrow[r, "f_0"] \arrow[d, "f_1"'] & \cdot \arrow[d, "h_0"] \\ \cdot \arrow[r, "h_1"'] & \cdot \end{tikzcd}$$

3(1)

$$\begin{tikzcd} \cdot \arrow[rr, "f_0"] \arrow[dd, "x"'] & & \cdot \arrow[d, "g_0"] \\ & & \cdot \arrow[d, "z"] \\ \cdot \arrow[r, "f_1"'] & \cdot \arrow[r, "g_1"'] & m \end{tikzcd}$$

3(2)

$$\begin{tikzcd} \cdot \arrow[r, "f_0"] \arrow[d, "x"'] & \cdot \arrow[r, "g_0"] \arrow[d, "y"] & \cdot \arrow[d, "z"] \\ \cdot \arrow[r, "f_1"'] & \cdot \arrow[r, "g_1"'] & m \end{tikzcd}$$

3(3)

Figure 3. IPO lemmas

Proposition 2 (IPO pasting). *Suppose that both squares in Fig. 3(3) commute and that Fig. 3(2) has an RPO.*

(i) *If the two squares of Fig. 3(3) are IPOs then so is the big rectangle.*
(ii) *If the big rectangle and the left square of Fig. 3(3) are IPOs then so is the right square.*

5 Labelled transitions and bisimulation congruence

We have set up in the previous section the categorical technology we need. We now give a formal definition of a "reactive system" and proceed to derive therefrom a labelled transition system. We then prove that, subject to the existence of sufficiently many RPOs, the associated bisimulation equivalence is a congruence.

Definition 3 (reactive system). *A* reactive system *consists of a category* $\mathbf{C}$ *with added structure. We let* m, n *range over objects.* $\mathbf{C}$ *has the following extra components:*

- *a distinguished object* 0 *(not necessarily initial);*
- *a set* $\mathsf{Reacts} \subseteq \bigcup_m \mathbf{C}(0, m)^2$ *of* reaction rules*;*
- *a subcategory* $\mathbf{D}$ *of* $\mathbf{C}$*, whose arrows are the* reactive contexts*, with the property that* $D_1 D_0 \in \mathbf{D}$ *implies* $D_1, D_0 \in \mathbf{D}$.

We think of the arrows of $\mathbf{C}$ as agents and contexts; the reactive contexts $\mathbf{D}$ are those in which reaction will be permitted. We write e.g. $D \in \mathbf{D}$ to mean that D is an arrow of $\mathbf{D}$. We let C, D, F range over arrows; we use a, b, l, r for arrows (the "agents") with domain 0. Note that if $(l, r) \in \mathsf{Reacts}$ then $l, r : 0 \to m$ for some m. The objects m of $\mathbf{C}$ represent interfaces between contexts. At this level of abstraction we specify no structure on objects, except to distinguish 0.

The reaction relation $\longrightarrow\!\!\!\!\triangleright$ is generated from Reacts by closing up under all reactive contexts:

Definition 4 (reaction). $a \longrightarrow\!\!\!\!\triangleright a'$ *iff there exists* $(l, r) \in \mathsf{Reacts}$ *and* $D \in \mathbf{D}$ *such that* $a = Dl$ *and* $a' = Dr$.

We now give our main definition. We replace the commuting square of (2) (Section 3) with an IPO, defining labelled transitions as follows:

$$
\begin{tikzcd}
0 \arrow[r, "a"] \arrow[d, "l"'] & {} \arrow[d, "F"] \\
{} \arrow[r, "D"'] & {}
\end{tikzcd}
\qquad
\begin{tikzcd}
0 \arrow[r, "a"] \arrow[d, "l"'] & {} \arrow[d, "R'F"] \\
{} \arrow[r, "D"'] & {}
\end{tikzcd}
$$

4(1) 4(2)

Figure 4. IPO squares for labelled transitions

Definition 5 (labelled transition). *$a \xrightarrow[1]{F}\!\!\triangleright\, a'$ iff there exists $(l, r) \in \mathsf{Reacts}$ and $D \in \mathbf{D}$ such that Fig. 4(1) is an IPO and $a' = Dr$.*

This definition assures that F, D provides a minimal upper bound on a and l, as required in Section 3. For suppose there is another upper bound F', D', i.e. $F'a = D'l$, and also $F = RF'$ and $D = RD'$ for some R. Then the IPO property for Fig. 4(1) ensures that for some R' (with $RR' = \mathsf{id}$) we have $F' = R'F$ and $D' = R'D$ — so F, D provides a "lesser" upper bound than F', D' after all.

Proposition 3. *For all contexts F we have that $a \xrightarrow[1]{F}\!\!\triangleright\, a'$ implies $Fa \longrightarrow\!\!\triangleright\, a'$.*

The converse fails in general (which is good, given the remarks made in Section 3 about the tentative definition (1) of labelled transitions). We return to the converse property later in the special case that F is an isomorphism.

Bisimulation over $\xrightarrow[1]{F}\!\!\triangleright$ follows its usual scheme [21]:

Definition 6 (bisimulation over $\xrightarrow[1]{F}\!\!\triangleright$). *Let $\mathcal{S} \subseteq \bigcup_m \mathbf{C}(0, m)^2$. $\mathcal{S}$ is a* simulation *over $\xrightarrow[1]{F}\!\!\triangleright$ iff for $(a, b) \in \mathcal{S}$, if $a \xrightarrow[1]{F}\!\!\triangleright\, a'$ then there exists b' such that $b \xrightarrow[1]{F}\!\!\triangleright\, b'$ and $(a', b') \in \mathcal{S}$. $\mathcal{S}$ is a* bisimulation *iff $\mathcal{S}$ and $\mathcal{S}^{-1}$ are simulations. Let $\sim_1$ be the largest bisimulation over $\xrightarrow[1]{F}\!\!\triangleright$.*

We now state and prove the central result of this paper: if $\mathbf{C}$ has a sufficiently rich collection of RPOs then $\sim_1$ is a congruence.

Definition 7 (redex-RPOs). *We say that $\mathbf{C}$* has all redex-RPOs *if for all $(l, r) \in \mathsf{Reacts}$ and arrows a, F, D such that $D \in \mathbf{D}$ and $Fa = Dl$, the square in Fig. 4(1) has an RPO.*

Theorem 1 (strong congruence). *If $\mathbf{C}$ has all redex-RPOs then $\sim_1$ is a congruence, i.e. $a \sim_1 b$ implies $Ca \sim_1 Cb$ for all C.*

Proof. It is sufficient to show that the following relation is a bisimulation:

$$\mathcal{S} \triangleq \{(Ca, Cb) \mid a \sim_1 b\} \quad .$$

The proof falls into three parts, each of which is an implication as illustrated in Fig. 5(1). Dashed lines connect pairs of points contained within the relation annotating the line. Each arrow "$\Downarrow$" is tagged by the part of the proof below that justifies the implication.

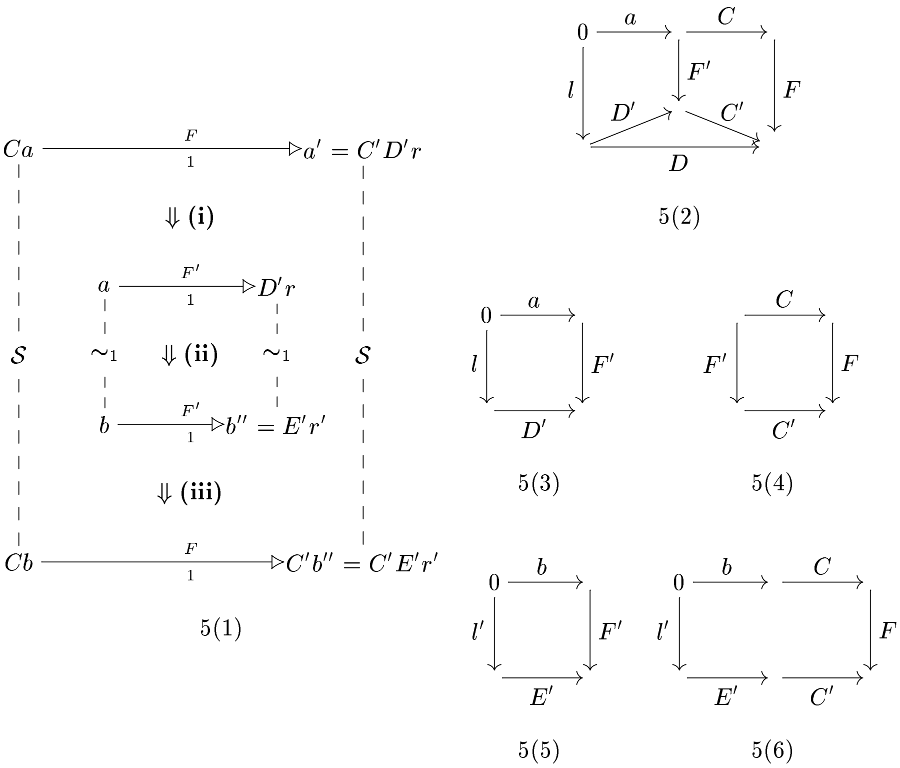

Figure 5. Congruence proof

(i) If $Ca \xrightarrow[1]{F} a'$ then, by definition, there exists $(l, r) \in \mathsf{Reacts}$ and $D \in \mathbf{D}$ such that the big rectangle in Fig. 5(2) is an IPO and $a' = Dr$. Because $\mathbf{C}$ has all redex-RPOs, there exists F', D', C' forming an RPO as in Fig. 5(2); moreover, $D', C' \in \mathbf{D}$ since $C'D' = D \in \mathbf{D}$. By Prop. 1, Fig. 5(3) is an IPO. Because $\mathbf{C}$ has all redex-RPOs, Prop. 2 implies that Fig. 5(4) is an IPO too. By definition, $a \xrightarrow[1]{F'} D'r$ and $a' = C'D'r$.

(ii) Since $a \sim_1 b$, there exists b'' such that $b \xrightarrow[1]{F'} b''$ and $D'r \sim_1 b''$. By definition there exists $(l', r') \in \mathsf{Reacts}$ and $E' \in \mathbf{D}$ such that Fig. 5(5) is an IPO and $b'' = E'r'$.

(iii) Because $\mathbf{C}$ has all redex-RPOs, Prop. 2 implies that we can paste Fig. 5(5) with Fig. 5(4) (both IPOs) along F' and conclude that Fig. 5(6) is an IPO. Hence $Cb \xrightarrow[1]{F} C'E'r'$ and $(C'D'r, C'E'r') \in \mathcal{S}$ because $D'r \sim_1 E'r'$, as desired.

The crux of the above proof is that Fig. 5(4), which mediates between an F'-labelled transition of a and an F-labelled transition of Ca, can be pasted onto a new diagram, serving the same function for b and Cb. This essential idea appears to be robust under variation both of the definition of labelled transition and of the congruence being established.

We now define two variants of $\xrightarrow[1]{F}$ for which transitions labelled by an isomorphism F recover the reaction relation, i.e. $a \xrightarrow[i]{F} a'$ iff $Fa \longrightarrow a'$ for $i = 2,3$ (cf. Prop. 3): here the isomorphisms play the role of the τ-label in π-calculus. The first is defined by brute-force case analysis:

Definition 8. $a \xrightarrow[2]{F} a'$ *iff* $\begin{cases} Fa \longrightarrow a' & \text{, if } F \text{ is an isomorphism} \\ a \xrightarrow[1]{F} a' & \text{, otherwise.} \end{cases}$

The second involves the existence of a retraction, a pair R, R' with $RR' = \mathsf{id}$, that adds just enough flexibility to the IPO condition:

Definition 9. $a \xrightarrow[3]{F} a'$ *iff there exists* $(l, r) \in \mathsf{Reacts}$*,* $D \in \mathbf{D}$*,* R*, and* R' *such that Fig. 4(2) is an IPO,* $a' = RDr$*, and* $RR' = \mathsf{id}$*.*

Finally, let $\sim_4$ be the bisimulation induced by the definition of labelled transition given in (1) (Section 3). The induced bisimulations of the different labelled transition relations are congruences and related as follows:

Theorem 2. *If* $\mathbf{C}$ *has all redex-RPOs then* $\sim_i$ *is a congruence for* $i = 2,3,4$ *and* $\sim_1 \subseteq \sim_2 \subseteq \sim_3 \subseteq \sim_4$*.*

We expect that some of these congruences coincide in specific applications. We shall also seek category theoretic conditions under which they provably coincide.

This theory generalises smoothly both to weak bisimulation [15] and to trace equivalences. For weak bisimulation, we think of the isomorphism labels as "silent moves":

Definition 10 (weak labelled transition).

$$a \xRightarrow{F} a' \quad \text{iff} \quad \begin{cases} Fa \longrightarrow^* a' & \text{, if } F \text{ is an isomorphism} \\ a \xrightarrow[1]{F}\longrightarrow^* a' & \text{, otherwise} \end{cases}$$

Let $\approx$ be the largest bisimulation over $\xRightarrow{F}$.

Theorem 3 (weak congruence). *If* $\mathbf{C}$ *has all redex-RPOs then* $\approx$ *is a congruence.*

The original weak bisimilarity of CCS employed a looser definition of $\xRightarrow{F}$, in which (using current notation) the compound transition $\xrightarrow[1]{F}\longrightarrow^*$ was replaced by $\longrightarrow^*\xrightarrow[1]{F}\longrightarrow^*$. It was not a congruence for CCS, though it was preserved by the CCS equivalent of reactive contexts. Interestingly, if the above replacement is made in Def. 10, then $\approx$ is preserved by reactive contexts.

6 Example 1: Wiring and bunches

In this section and the next we present two examples of reactive systems in which RPOs exist. They are not chosen to represent practical systems, but to illustrate clearly the three main features of action calculi [17]: *parallel composition*, *wiring*, and *nesting of agents*.

Our first example is motivated by parallel composition and wiring. We study a simple class of agents which we call *bunches*, that exhibit some of the variety

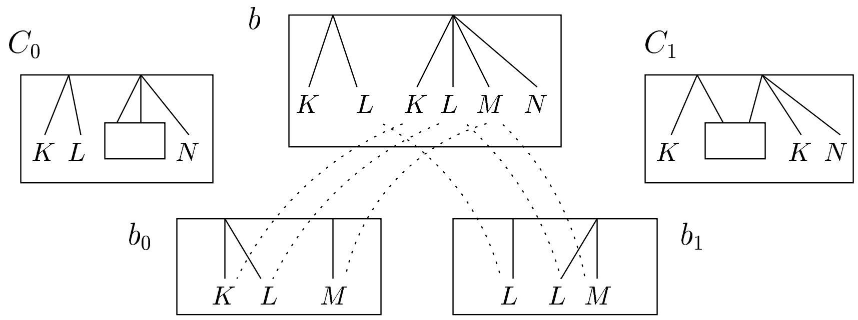

Figure 6. Composition of bunch contexts and bunches

of copied wiring (naming) inherent in, for example, the π-calculus and clearly apparent in a graphical presentation. A bunch is a finite ordered set of unordered trees of depth one, each leaf vertex possessing a character from a fixed *character set* $\mathcal{K} = \{K, L, M, \dots\}$. Three bunches b, b_0, b_1 are shown in Fig. 6 (ignore the dotted lines for now). Two *bunch contexts* C_0, C_1 are also shown, each with a single hole; putting b_0 into C_0 and b_1 into C_1 yields $b = C_0 b_0 = C_1 b_1$. We define **Bun** formally as follows:

Definition 11 (interfaces and bunch contexts). *The bunch category* **Bun** *has* interfaces (U, m) *as objects, where* U *is a finite set of* vertices *and the ordinal* $m = \{0, \dots, m-1\}$ *represents an ordered set of* roots. *An arrow of* **Bun** *is a* bunch context $C = (t, root, char) : (U_0, m_0) \to (U_1, m_1)$ *whose components, where* $V = U_1 - t(U_0)$ *is the* vertex set, *are:*

$$t : U_0 \rightarrowtail U_1 \qquad \textit{the } \text{trail } \textit{(injective)}$$
$$root : V \oplus m_0 \twoheadrightarrow m_1 \qquad \textit{the } \text{parent map } \textit{(surjective)}$$
$$char : V \to \mathcal{K} \qquad \textit{the } \text{character map}.$$

If $(U_0, m_0) = (\emptyset, 0)$ *we call* C *a* bunch*; we typically use* b *for a bunch. In* **Bun**, *every context is reactive.*

(In this example we use "$\oplus$" to combine disjoint sets and functions with disjoint domains.) Composition of contexts is easy to understand graphically. Formally:

Definition 12 (identities and composition). *The* identity *context* $\mathsf{id}_{(U,m)} \triangleq (id_U, id_m, \emptyset)$. *For two contexts* $C_i = (t_i, root_i, char_i) : (U_i, m_i) \to (U_{i+1}, m_{i+1})$ *(*$i = 0, 1$*), their* composition $C_1 C_0 \triangleq (t, root, char) : (U_0, m_0) \to (U_2, m_2)$ *is determined as follows, where* V_i *are the vertex sets of* C_i *and* $V = V_1 \oplus t_1(V_0)$*:*

$$t \triangleq t_1 \circ t_0$$
$$root : V \oplus m_0 \twoheadrightarrow m_2 \triangleq root_1 \circ (id_{V_1} \oplus (root_0 \circ (t_1^{-1} \oplus id_{m_0})))$$
$$char : V \to \mathcal{K} \triangleq char_1 \oplus (char_0 \circ t_1^{-1}).$$

This definition yields a category. To see how interfaces and trails work, consider C_0 in Fig. 6. When a graph with 3 vertices and 2 roots is placed in the hole, the

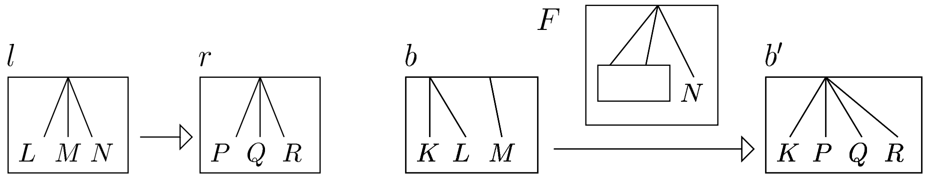

Figure 7. A reaction rule $l \longrightarrow\!\!\!\triangleright\, r$ and a labelled transition $b \xrightarrow{F}\!\!\triangleright\, b'$

result is a graph with 6 vertices and 2 roots. Naming the vertices suitably we have $C_0 : (\{v_0, v_1, v_2\}, 2) \to (\{v_0, \ldots, v_5\}, 2)$. Then the trail of C_0 is $t_0 : v_i \mapsto v_{i+2}$ $(i = 0, 1, 2)$; this is indicated in Fig. 6 by the dotted lines from the vertices of b_0 to those of b. Trails are a version of Sewell's notion of colouring. They assure the following:

Theorem 4. Bun *has all RPOs, hence all redex-RPOs.*

There is a natural alternative version of **Bun** without trails; we lack space for it, but a counter-example indicates that the RPO property is then lost.

The labels obtained via IPOs in **Bun** are pleasantly simple. A reaction rule $l \longrightarrow\!\!\!\triangleright\, r$ is shown in Fig. 7, with an example of a corresponding labelled transition $b \xrightarrow{F}\!\!\triangleright\, b'$; the label context F supplies the parts of l which are missing in b, both leaves and wiring, required to create an instance of l. By specifying a vertex set U in the interface (U, m), we have fixed the size of bunch which can fit in a hole. Current work promises to relax this condition, by allowing contexts to retain their trail components but to be *polymorphic*, in that they apply to holes of any size. We do not treat this generalisation here.

7 Example 2: Term-rewriting and multi-hole contexts

Our second example is motivated by the nesting of agents, which occurs in its most familiar form in term-rewriting. In Section 4 we argued that pushouts do not exist for free term contexts, thus motivating the exploration of RPOs in the abstract. We now address the specific properties of RPOs for term contexts. If we apply our theory to the category of one-hole contexts, then RPOs exist, as a corollary of Sewell's dissection result for terms (Lemma 1 in [22]). Consequently, all the definitions of labelled transition in Section 5 induce bisimulation equivalences that are congruences for term rewriting systems. The resulting labels are unnecessarily heavy, though. For consider the reaction rule $(\gamma(\alpha), \alpha')$; we have $\alpha \xrightarrow{\gamma(-)}\!\!\triangleright\, \alpha'$ which corresponds to our intuition that α needs $\gamma(-)$ to react. Unfortunately, we also have a labelled transition where the label contains a complete copy of the redex:

$$\alpha' \xrightarrow{\beta(-,\gamma(\alpha))}\!\!\triangleright\, \beta(\alpha', \alpha') \quad .$$

To attack this problem we consider multi-hole contexts where we shall find that this transition is prohibited. We then modify the definition of labelled transition and the statement of the congruence theorem to cater explicitly for multi-hole contexts in any reactive system. We end by asserting that multi-hole term contexts satisfy the hypotheses of this new congruence theorem.

$$
\begin{tikzcd}
0 \arrow[rr, "a"] \arrow[dd, "l"'] & & m_0 \arrow[dl, "F'"'] \arrow[dd, "F"] \\
& u \arrow[dr, "G"] & \\
m_1 \arrow[ur, "D'"] \arrow[rr, "D"'] & & m
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
0 \arrow[r, "a_0"] \arrow[d, "a_1"'] & m_0 \arrow[d, "\mathsf{id}_{m_0} \otimes a_1"] \\
m_1 \arrow[r, "a_0 \otimes \mathsf{id}_{m_1}"'] & m_0 \otimes m_1
\end{tikzcd}
$$

8(1) 8(2)

Figure 8. Redex-RPOs and tensor-IPOs

Definition 13 (multi-hole term contexts). *Given a* signature Σ *of function symbols then the category of* multi-hole term contexts $\mathbf{T}^*(\Sigma)$ *over* Σ *is constructed as follows: the objects are the natural numbers; an arrow $j \to k$ is a k-tuple of terms over the signature $\Sigma \cup \{-_1, \dots, -_j\}$ containing exactly one use of each hole $-_i$ $(1 \le i \le j)$. The* identities *are:* $\mathsf{id}_j \triangleq \langle -_1, \dots, -_j \rangle$. *For $f = \langle a_1, \dots, a_k \rangle : j \to k$ and $g : k \to m$, their* composition *is the substitution*

$$gf \triangleq \{a_1/-_1, \cdots, a_k/-_k\}g \quad .$$

(When $j = 1$ we write $-_1$ as $-$.)

We now refine the abstract definitions of reactive system and of labelled transition to cater explicitly for multi-hole contexts in *any* reactive system, not just $\mathbf{T}^*(\Sigma)$. This refinement is part of our programme to express abstractly the phenomena of real reactive systems.

We require the notion of a *strict monoidal category* $(\mathbf{C}, \otimes, 0)$, a category $\mathbf{C}$ equipped with a functor $\otimes : \mathbf{C} \times \mathbf{C} \to \mathbf{C}$ and an object 0 such that $\otimes$ is strictly associative and has unit 0.

The role of the tensor $\otimes$ in the definition of reactive system is to "tuple" objects (e.g. $1 \otimes 1 = 2$ in $\mathbf{T}^*(\Sigma)$) and arrows.

Definition 14 (multi-hole reactive system). *A* reactive system *consists of a strict monoidal category* $(\mathbf{C}, \otimes, 0)$ *and the following added structure:*

- *a subset Z of objects (we use $m, m', \dots$ to range over Z);*
- *a set* $\mathsf{Reacts} \subseteq \bigcup_{m \in Z} \mathbf{C}(0, m)^2$ *of* reaction rules*;*
- *a subcategory $\mathbf{D}$ of $\mathbf{C}$, whose arrows are the* reactive contexts*, with two properties: $D_1 D_0 \in \mathbf{D}$ implies $D_1, D_0 \in \mathbf{D}$; and $a \otimes \mathsf{id}_m \in \mathbf{D}$ for $a : 0 \to m'$.*

The *agents* of a reactive system are arrows $0 \to m$ and the *agent contexts* are arrows $m \to m'$, for $m, m' \in Z$. (Thus, for example, in $\mathbf{T}^*(\Sigma)$, we take $Z = \{1\}$ to mark out the singleton terms.) We adapt the definition of labelled transition:

Definition 15 (labelled transition). $a \xrightarrow{F} a'$ *iff a, a' are agents, F an agent context, and there exists $(l, r) \in \mathsf{Reacts}$ and $D \in \mathbf{D}$ such that Fig. 4(1) is an IPO and $a' = Dr$.*

Two conditions replace "redex-RPOs" (Def. 7):

Definition 16 (redex-RPOs). **C** has all redex-RPOs *if for all* $(l, r) \in \mathsf{Reacts}$ *and arrows* a, F, D*, where* a *is an agent,* F, D *agent contexts,* $D \in \mathbf{D}$*, and* $Fa = Dl$*, then the square in Fig. 8(1) has an RPO, as shown, such that either* $u \in Z$*, or there exists an isomorphism* $k : u \to m_0 \otimes m_1$ *such that* $kF' = \mathsf{id}_{m_0} \otimes l$ *and* $kD' = a \otimes \mathsf{id}_{m_1}$*.*

Definition 17 (tensor-IPOs). **C** has all tensor-IPOs *if Fig. 8(2) is an IPO square for all* $a_i : 0 \to m_i$ *with* $m_i \in Z$ *and* $i = 0, 1$*.*

Theorem 5 (congruence). *If* **C** *has all redex-RPOs and all tensor-IPOs then* $\sim$ *is preserved by all agent contexts.*

Let us return to the category $\mathbf{T}^*(\Sigma)$ and see how we may apply Theorem 5 to it. We have a choice of Z; we here confine ourselves to the case $Z = \{1\}$. Also, we may choose any subcategory of $\mathbf{T}^*(\Sigma)$ to be the reactive contexts, subject to the conditions in Def. 14. Then

- an agent of $\mathbf{T}^*(\Sigma)$ is a term $a : 0 \to 1$;
- an agent context of $\mathbf{T}^*(\Sigma)$ is a term context $C : 1 \to 1$, i.e. a term containing a single hole.

The labels of $\mathbf{T}^*(\Sigma)$ depend, of course, on the reaction rules. Once these are specified, we have determined the labelled transition relation $\xrightarrow{F}\!\triangleright$ over $\mathbf{T}^*(\Sigma)$, and hence the induced bisimulation $\sim$. Formally we have:

Theorem 6. *If we take* $Z = \{1\}$ *then* $\mathbf{T}^*(\Sigma)$ *has all redex-RPOs and all tensor-IPOs. Hence from Theorem 5 the induced bisimilarity* $\sim$ *over* $\mathbf{T}^*(\Sigma)$ *is preserved by all term contexts.*

Let us now revisit the reactive system whose only reaction rule is $(\gamma(\alpha), \alpha')$. It contains exactly the following labelled transitions:

$$\begin{array}{rl} D(\gamma(\alpha)) \xrightarrow{-}\!\triangleright D(\alpha') & \qquad \text{for all reactive term contexts } D \\ \alpha \xrightarrow{\gamma(-)}\!\triangleright \alpha' & \end{array}$$

These agree with the transitions found by Sewell in the case of ground-term rewriting. We believe that the labels in our reactive system $\mathbf{T}^*(\Sigma)$ coincide exactly with Sewell's. Note that the heavy transition mentioned earlier is absent. We indicate why this is so with the help of the diagram below.

$$\begin{tikzcd}
0 \arrow[rr, "\alpha'"] \arrow[dd, "\gamma(\alpha)"'] & & 1 \arrow[dl, "{\langle -, \gamma(\alpha) \rangle}"] \arrow[dd, "{\beta(-, \gamma(\alpha))}"] \\
& 2 \arrow[dr, "{\beta(-_1, -_2)}"] & \\
1 \arrow[ur, "{\langle \alpha', - \rangle}"] \arrow[rr, "{\beta(\alpha', -)}"'] & & 1
\end{tikzcd}$$

If we work in the category of *one-hole* contexts then the outer square is an IPO, which gives rise the transition $\xrightarrow{\beta(-,\gamma(a))}$ mentioned earlier. By admitting multi-hole contexts we have given the outer-square a simpler RPO.

Note also that, though working with multi-hole contexts, we have been free to choose the set Z as small as we wish, and thus obtain a simpler requirement for the existence of RPOs.

8 Current and future work

We have presented here a single key idea which allows labelled transitions, and thence behavioural congruences, to be derived for reactive systems. This is part of a larger programme of work, aiming at a theory of behavioural equivalence relevant to calculi designed for a wide range of practical purposes.

We are at present identifying those action calculi, or subcategories thereof, which possess RPOs; this will greatly clarify the status of action calculi as a framework. In particular, sharing graphs [1, 11] and Gardner's closed action calculi [8] should be addressed. We also have to develop the notion of "polymorphic" context alluded to in Section 6.

Calculi already equipped with LTSs and congruence proofs need to be checked to see how close our uniformly derived LTSs and equivalences come to theirs. We would like to study in generality the situation in which redexes themselves are contexts, as Sewell [22] has done for term rewriting with parallel composition. Jeffrey and Rathke [13] have recently studied the relationship between contextual equivalence and labelled transitions for the ν-calculus of Pitts and Stark [20]; this will provide a good test for our uniform derivation of LTSs. We do not expect yet to achieve the fine-tuning present in some calculi; but we see no obvious limit to what can be achieved using general categorical (and other) methods as distinct from working in each individual calculus. More generally, links with other lines of research must be explored. Our method does not appear to overlap with the categorical approach by Joyal, Nielsen and Winskel [14] in defining bisimulation from open maps, but one should attempt to integrate the category-theoretic study of LTS-based equivalences. Categorical methods have also been productive in graph-rewriting; for example, in 1991 Corradini and Montanari [6] were already combining categorical and algebraic methods in concurrent graph-rewriting. Such work has developed further and should be related to ours.

In the longer term, we believe that graphical models represent the best hope for a theory of interactive systems — including the internet — which design engineers and analysts can actually use, with the help of computerised visualisation backed by rigorous machine-assisted verification. We hope the present work will provide part of the necessary theoretical background for this development.

Acknowledgements The authors thank Luca Cattani, Philippa Gardner, Georges Gonthier, Martin Hyland, Ole Jensen, Jean-Jacques Lévy, Andrew Pitts, and Peter Sewell for stimulating discussions, and the anonymous referees for their helpful comments. Leifer was supported by an NSF Graduate Research Fellowship and a Trinity College Senior Rouse Ball Studentship.

References

1. Ariola, Z.M. and Klop, J.W., Equational term graph rewriting. Fundamentae Informaticae, 26(3,4), pp. 207–240, 1996.
2. Abadi, M. and Gordon, A.D., A calculus for cryptographic protocols: the spi calculus. Proc. Fourth ACM Conf. on Computer and Communications Security, Zürich, ACM Press, pp. 36–47, 1997.
3. Baeten, J.C. and Weiland, W.P., Process algebra. CUP, 1990.
4. Berry, G. and Boudol, G., The chemical abstract machine. Theor. Comp. Sci. 96, pp. 217–248, 1992.
5. Cardelli, L. and Gordon, A.D., Mobile ambients. Foundations of System Specification and Computational Structures, LNCS 1378, pp. 140–155, 1998.
6. Corradini, A. and Montanari, U., An algebra of graphs and graph rewriting. Proc. Fourth Biennial Conf. on Category Theory and Computer Science, LNCS 530, pp. 236–260, 1991.
7. Fournet, C., Gonthier, G., Lévy, J.-J., Maranget, L., and Rémy, D., A calculus of mobile agents. Proc. CONCUR'96, LNCS 1119, pp. 406–421, 1996.
8. Gardner, P., Closed action calculi. Theor. Comp. Sci. 228(1,2), pp. 77–103, 1999.
9. van Glabbeek, R.J., The linear time - branching time spectrum. Proc. CONCUR'90, LNCS 458, pp. 278–297, 1990.
10. Groote, J.F. and Vaandrager, F.W., Structural operational semantics and bisimulation as a congruence. Information and Computation 100(2), pp. 202–260, 1992.
11. Hasegawa, M. Models of sharing graphs (a categorical semantics of let and letrec). PhD thesis, LFCS, University of Edinburgh, 1997.
12. Hoare, C.A.R., Communicating Sequential Processes. Prentice Hall, 1985.
13. Jeffrey, A. and Rathke, J., Towards a theory of bisimulation for local names. Proc. LICS'99, IEEE Press, pp. 56–66, 1999.
14. Joyal, A., Nielsen, M. and Winskel, G., Bisimulation from open maps. Information and Computation 127(2), pp. 164–185, 1996.
15. Milner, R., Communication and Concurrency. Prentice Hall, 1989.
16. Milner, R., Functions as processes. Mathematical Structures in Computer Science 2(2), pp. 119–141, 1992.
17. Milner, R., Calculi for interaction. Acta Informatica 33(8), pp. 707–737, 1996.
18. Milner, R., Parrow, J. and Walker, D., A calculus of mobile processes, Parts 1 and 2. Information and Computation 100(1), pp. 1–77, 1992.
19. Milner, R. and Sangiorgi, D., Barbed bisimulation. Proc. ICALP'92, LNCS 623, pp. 685–695, 1992.
20. Pitts, A.M. and Stark, I.D.B., Observable properties of higher order functions that dynamically create local names, or: What's new? Proc. MFCS, LNCS 711, pp 122–141, 1993.
21. Park, D., Concurrency and automata on infinite sequences. LNCS 104, 1980.
22. Sewell, P., From rewrite rules to bisimulation congruences. Proc. CONCUR'98, LNCS 1466, pp. 269–284, 1998. [Revised version to appear in a special issue of Theor. Comp. Sci.]
23. Sewell, P., Global/local subtyping and capability inference for a distributed pi-calculus. Proc. ICALP'98, LNCS 1443, pp. 695–706, 1998.
24. Turi, D. and Plotkin, G. Towards a mathematical operational semantics. Proc. LICS'97, IEEE Press, pp. 280–291, 1997.