Available online at www.sciencedirect.com
SCIENCE DIRECT®
Theoretical Computer Science 333 (2005) 297–327
Theoretical Computer Science
www.elsevier.com/locate/tcs

Locating reaction with 2-categories[☆]

Vladimiro Sassone[a,*], Paweł Sobociński[b]
[a] *Department of Informatics, University of Sussex, Brighton BN1 9QH, UK*
[b] *BRICS, University of Aarhus, Denmark*

Abstract

Groupoidal relative pushouts (GRPOs) have recently been proposed by the authors as a new foundation for Leifer and Milner's approach to deriving labelled bisimulation congruences from reduction systems. In this paper, we develop the theory of GRPOs further, proving that well-known equivalences, other than bisimulation, are congruences. To demonstrate the type of category theoretic arguments which are inherent in the 2-categorical approach, we construct GRPOs in a category of 'bunches and wirings.' Finally, we prove that the 2-categorical theory of GRPOs is a generalisation of the approaches based on Milner's precategories and Leifer's functorial reactive systems.
© 2004 Elsevier B.V. All rights reserved.

0. Introduction

It has become increasingly common to view modern foundational process calculi as being, at their core, *reduction systems*. Starting from their common ancestor, the λ-calculus, most recent calculi consist of a reduction system together with a contextual equivalence (built out of basic observations, viz. barbs). The strength of such an approach resides in its intuitiveness. In particular, we need not invent labels to describe the interactions between systems and their possible environments, a procedure that may present a degree of

☆ Research supported by '*DisCo*: Semantic Foundations of Distributed Computation, EU IHP 'Marie Curie' contract HPMT-CT-2001-00290, and *BRICS*, Basic Research in Computer Science, funded by the Danish National Research Foundation.

* Corresponding author. Tel.: +44 1273 873828; fax: +44 1273 671320.
E-mail address: vs@susx.ac.uk (V. Sassone).

0304-3975/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2004.10.025

arbitrariness, (cf. early and late semantics of the π calculus) and may prove quite complex (cf. [1,3–5], for instance).

By contrast, reduction semantics suffer at times by their lack of compositionality, and have complex semantic theories because contextual equivalences usually involve quantification over an infinite set of contexts. Labelled bisimulation congruences based on *labelled transition systems* (LTS) may in such cases provide fruitful proof techniques; in particular, bisimulations provide the power and manageability of coinduction, while the closure properties of congruences provide for compositional reasoning.

A well-behaved LTS associated with a reduction system should involve a compositional system of labels, with silent moves (or τ-actions) reflecting the original reductions and labels describing potential external interactions. Ideally, the resulting bisimulation should be a congruence, and should be at least included in the original contextual reduction equivalence. Proving bisimilarity is then enough to prove reduction equivalence.

Sewell [24] and Leifer and Milner [14,12] set out to develop a theory to perform such derivations using general criteria; a meta-theory of *deriving bisimulation congruences*. The basic idea behind their construction is to use contexts as labels. To exemplify the idea, in a CCS-like calculus one would for instance derive a transition

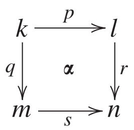
$$a.P \xrightarrow{-|\bar{a}.Q} P \mid Q$$

because term $a.P$ in context—$|\ \bar{a}.Q$ reacts to become $P \mid Q$; in other words, the context is a trigger for the reduction.

The first hot spot of the theory is the selection of the right triggers to use as labels. The intuition is to take only the '*smallest*' contexts which allow a given reaction to occur. As well as reducing the size of the LTS, this often makes the resulting bisimulation equivalence finer and often closer to operational intuitions. Sewell's method is based on dissection lemmas which provide a deep analysis of a term's structure. A generalised, more scalable approach was later developed in [14], where the notion of 'smallest' is formalised in categorical terms as a *relative-pushout* (RPOs). More precisely, as we shall see, a context is selected as a label for the transition system if it makes a certain categorical diagram be a pushout. Both theories, however, do not seem to scale up to calculi with non-trivial *structural congruences*. Already in the case of the monoidal rules that govern parallel composition, things become rather involved.

The fundamental difficulty brought about by a structural congruence $\equiv$ is that working up to $\equiv$ loses too much information about terms for the RPO approach to work as expected. RPOs do not usually exist in such cases, because the fundamental indication of exactly which occurrences of a term constructor belong to the redex becomes blurred when terms are quotiented by $\equiv$. A very simple, yet significant example of this is the category **Bun** of bunch contexts considered in [14], and similar problems arise in structures such as action graphs [15] and bigraphs [17].

In [19,21], we therefore proposed a framework in which term structure is not explicitly quotiented, but the equality of terms is taken up to $\equiv$. Precisely, to give $rp \equiv sq$ one must exhibit a proof α of structural congruence. Thinking of terms as arrows in categories where objects represent term arities (e.g. as induced by a signature Σ), the equation $rp \equiv sq$ can be recast categorically as a commuting diagram together with a 2-cell α (constructed from the rules generating ≡ and closed under all contexts), as in the diagram below.



Since such proofs are naturally isomorphisms, we were led to consider *groupoid-enriched* categories (*G-categories* for short), i.e. 2-categories where all 2-cells are iso, and initiated the study of *G-relative pushouts* (GRPOs), as a suitable generalisation of RPOs from categories to G-categories. The idea of using 2-cells to represent generalised structural congruence was first suggested by Sewell [23].

The purpose of this paper is to continue the development of the theory of GRPOs. We aim to show that, while adding little further complication (cf. Sections 2 and 3), GRPOs advance the field by providing a convenient solution to simple, yet important problems (cf. Sections 4 and 5). GRPOs indeed promise to be part of an elegant foundation for a meta-theory of 'deriving bisimulation congruences'.

This paper presents two main technical results in support of our claims. Firstly, we prove that the case of the aforementioned category **Bun** of bunch contexts, problematic for RPOs, can be treated in a natural way using GRPOs. Secondly, we show that the notions of precategory and functorial reactive system, theories introduced to deal with the problems solved by GRPOs, can be encompassed in the GRPO-based approach.

The notion of *precategory* is proposed in [12,13] inspired by the examples of Leifer in [12], Milner in [17] and, most recently, of Jensen and Milner in [8]. It consists of a category appropriately decorated by so-called '*support sets*' which identify syntactic elements so as to keep track of them under arrow composition. Such supported structures are no longer categories—arrow composition is partial—which bring us away from the well-known world of categories and their established theory, and requires an ad hoc development. The intensional information recorded in precategories, however, allows one to generate a category 'above' where RPOs exist, as opposed to the category of interest 'below,' say $\mathbb{C}$, where they do not. The category 'above' is related to $\mathbb{C}$ via a well-behaved functor, used to map RPOs diagrams from the category 'above' to $\mathbb{C}$, where constructing them would be impossible. (Here, 'well-behaved' means that the functor satisfies technical conditions which guarantee the transport of relevant properties to $\mathbb{C}$.) These structures take the name of *functorial reactive systems*, and give rise to a theory developed in [12] to generate labelled bisimulation congruences.

This paper presents a technique for mapping precategories to G-categories so that the LTS generated using GRPOs is the same (i.e. it has *exactly* the same labels) as the LTS generated using the above-mentioned approach. The translation derives from the precategory's support information a notion of homomorphism, specific to the particular structure in hand, which constitutes the 2-cells of the derived G-category. We claim that this yields a mathematically elegant approach, potentially more general and in principle more direct than precategories, in that it allows for arbitrary structural isomorphisms to be considered, and fits well within existing category theory, with no need for new frameworks. In particular, one advantage of G-categories is that one may apply standard categorical constructions without translations or alterations. Further supporting evidence for GRPOs is provided in [22], where we apply their theory to graphs and graph rewriting. It remains to be seen, of course, whether future developments, e.g. for the analysis of specific LTSs obtained through our constructions, will point towards the need of additional structure on G-categories.

Structure of the paper: In Section 2 we review definitions and results presented in our previous work [19,21]; Section 3 shows that, analogously to the 1-dimensional case, trace and failures equivalence are congruences provided that enough GRPOs exist. In Section 4, we show that the category of bunch contexts is naturally a 2-category where GRPOs exist; Section 5 shows how precategories are subsumed by our notion of GRPOs. The exposition ends with a few concluding remarks; Section 1 recalls basic notions of 2-categories, and can be safely skipped by those readers acquainted with the standard notations.

An extended abstract of this work appeared as [20]. Here we additionally develop the theory of weak operational congruences, and illustrate the role of the notion of extensive category in the construction of GRPOs in **Bun**.

1. Preliminaries

Throughout the paper, we assume a moderate knowledge of category theory and related terminology. In this section, we fix notations and recall the basic elements of 2-categories we need to state our definitions and prove our results. For a thorough introduction to 2-categories, the reader is referred to [10].

We use **Ord** to denote the category of finite ordinals. The objects of this category are the natural numbers $0, 1, 2, \ldots$. The morphisms from m to n are the all the functions from the m-element set $[m] = \{1, 2, \ldots, m\}$ to $[n] = \{1, 2, \ldots, n\}$. Composition is the usual compositions of functions. The category is skeletal, in that we have $n \cong n'$ if and only if $n = n'$. We assume that **Ord** has chosen coproducts, namely ordinal addition $\oplus$. One possible way to define this is to let, on objects, $m \oplus n = m + n$, while on arrows, given $f : m \to m'$ and $g : n \to n'$, let $f + g : m+n \to m'+n'$ be the function $(f+g)(x) = f(x)$ for $1 \leqslant x \leqslant m$ and $(f+g)(x) = g(x-m) + m'$ otherwise. Intuitively, $f+g$ is constructed by putting f and g side-by-side.

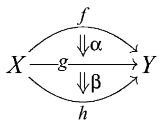
For any finite set x, let $ord(x)$ be the finite ordinal of the same cardinality and $t_x \colon x \to ord(x)$ be a chosen isomorphism. There is an equivalence of categories $F \colon \mathbf{Set}_f \to \mathbf{Ord}$. On objects it sends x to $ord(x)$; on morphisms, it maps $f \colon x \to y$ to $t_y f t_x^{-1} \colon ord(x) \to ord(y)$.

A 2-*category* $\mathbb{C}$ is a category where homsets (that is the collections of arrows between any pair of objects) are categories and, correspondingly, whose composition maps are functors. Explicitly, a 2-category $\mathbb{B}$ consists of the following:

- A class of *objects* $X, Y, Z, \ldots$.
- For any $X, Y \in \mathbb{C}$, a category $\mathbb{C}(X, Y)$. The objects $\mathbb{C}(X, Y)$ are called 1-*cells*, or simply arrows, and denoted by $f \colon X \to Y$. Its morphisms are called 2-*cells*, are written $\alpha \colon f \Rightarrow g \colon X \to Y$ and drawn as

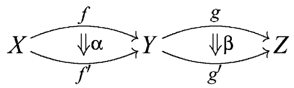
$$X \underset{g}{\overset{f}{\Downarrow\alpha}} Y.$$

Composition in $\mathbb{C}(X, Y)$ is denoted by $\bullet$ and referred to as '*vertical*' composition. Identity 2-cells are denoted by $\mathbf{1}_f \colon f \Rightarrow f$. Isomorphic 2-cells are occasionally denoted as $\alpha\colon f \cong g$. As an example of vertical composition, consider 2-cells $\alpha : f \Rightarrow g$ and $\beta : g \Rightarrow h$ as below.



They can be composed, yielding $\beta \bullet \alpha : f \Rightarrow h$.

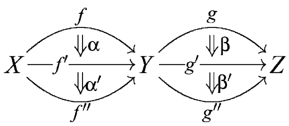
- For each X, Y, Z there is a functor $.\colon \mathbb{C}(Y, Z) \times \mathbb{C}(X, Y) \to \mathbb{C}(X, Z)$, the so-called '*horizontal*' composition, which we often denote by mere juxtaposition. Horizontal composition is associative and admits $\mathbf{1}_{\mathrm{id}_X}$ as identities. As an example, consider 2-cells $\alpha : f \Rightarrow f'$ and $\beta : g \Rightarrow g'$, as illustrated below.



They can be composed horizontally, obtaining $\beta\alpha : gf \Rightarrow g'f'$.

As a notation, we write αf and $g\alpha$ for, respectively, $\alpha\mathbf{1}_f$ and $\mathbf{1}_g\alpha$. We follow the convention that horizontal composition binds tighter than vertical composition.

In 2-categories, the order of composition of 2-cells is not important. This is a consequence of the horizontal composition being a functor, and can be axiomatised with the so called *middle-four interchange law*: for $f, f', f''\colon X \to Y$ and $g, g', g''\colon Y \to Z$ and $\alpha\colon f \Rightarrow f'$, $\alpha'\colon f' \Rightarrow f''$, $\beta\colon g \Rightarrow g'$ and $\beta'\colon g' \Rightarrow g''$, as illustrated by

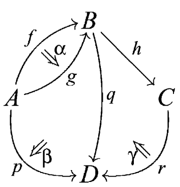


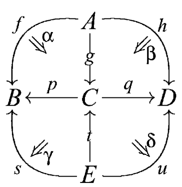
we have

$$\beta'\alpha' \bullet \beta\alpha = (\beta' \bullet \beta)(\alpha' \bullet \alpha).$$

As a consequence, it can be shown that a diagram of 2-cells defines at most one composite 2-cell; that is, all the possible different ways to combine together vertical and horizontal composition, yield the same composite 2-cell. This primitive operation is referred to as *pasting*.

In order to illustrate the notion of pasting, we shall consider the following diagrams.





The left diagram features 2-cells $\alpha : f \Rightarrow g$, $\beta : qg \Rightarrow p$ and $\gamma : rh \Rightarrow q$. They can be pasted together uniquely to obtain a 2-cell $rhf \Rightarrow p$. This 2-cell can be written as either $\beta \bullet q\alpha \bullet \gamma f : rhf \Rightarrow p$, or equally, $\beta \bullet \gamma g \bullet rh\alpha : rhf \Rightarrow p$. Now consider the right diagram with 2-cells $\alpha : f \Rightarrow pg$, $\beta : h \Rightarrow qg$, $\gamma : pt \Rightarrow s$ and $\delta : qt \Rightarrow u$. There is no way of composing these 2-cells.

The canonical example of a 2-category is **Cat**, the 2-category of categories, functors and natural transformations.

Two objects C, D of a 2-category $\mathbb{C}$ are *equivalent* when there are arrows $f : C \to D$, $g : D \to C$ and isomorphic 2-cells $\alpha : \mathrm{id}_C \Rightarrow gf$, $\beta : fg \Rightarrow \mathrm{id}_D$. We refer to f and g as equivalences.

2. Reactive systems and GRPOs

Lawvere theories [11] provide a canonical way to recast term algebras as categories, and open the way to the categorical treatment of related notions. For Σ a signature, the (free) Lawvere theory on Σ, say $\mathbf{C}_\Sigma$, has the natural numbers for objects and a morphism $t : m \to n$, for t a n-tuple of m-holed terms. Composition is substitution of terms into holes. For instance, for Σ the signature for arithmetics, term $(-_1 \times x) + -_2$ is an arrow $2 \to 1$ (two holes yielding one term) while $\langle 3, 2 \times y \rangle$ is an arrow $0 \to 2$ (a pair of terms with no holes). Their composition is the term $(3 \times x) + (2 \times y)$, an arrow of type $0 \to 1$.

Generalising from term rewriting systems on $\mathbf{C}_\Sigma$, Leifer and Milner formulated a definition of *reactive system* [14], and defined a technique to extract labelled bisimulation congruences from them. In order to accommodate calculi with non-trivial structural congruences, as explained in the Introduction, we refine their approach as follows.

Definition 1. A *G-category* is a 2-category where all 2-cells are isomorphisms.

A G-category is a thus a category enriched over **Gp**, the category of groupoids.

We shall adopt the convention of not indicating the direction of 2-cells when working with G-categories. This will considerably simplify notation while not causing much confusion; our 2-cells $\alpha : p \Rightarrow q$ will always be isomorphic.

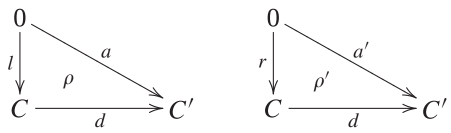
Definition 2. A *G-reactive system* **C** consists of

(1) a G-category $\mathbb{C}$,
(2) a collection $\mathbb{D}$ of arrows of $\mathbb{C}$ which shall be referred to as the *reactive contexts*; it is required to be closed under 2-cells and reflect composition,
(3) a distinguished object $0 \in \mathbb{C}$,
(4) a set of pairs $\mathcal{R} \subseteq \bigcup_{C \in \mathbb{C}} \mathbb{C}(0, C) \times \mathbb{C}(0, C)$ called the *reaction rules*.

The reactive contexts are those contexts inside which evaluation may occur. By composition-reflecting we mean that $dd' \in \mathbb{D}$ implies $d \in \mathbb{D}$ and $d' \in \mathbb{D}$, while the closure property means that given $d \in \mathbb{D}$ and $\alpha : d \Rightarrow d'$ in $\mathbb{C}$ implies $d' \in \mathbb{D}$. The reaction relation $\longrightarrow\!\!\!\!\!\triangleright$ is defined by taking

$$a \longrightarrow\!\!\!\!\!\triangleright a' \quad \text{if there exists } \langle l, r \rangle,\ d \in \mathbb{D} \text{ and } \rho : dl \Rightarrow a,\ \rho' : a' \Rightarrow dr.$$

As illustrated by the diagram below, this represents the fact that, up to structural congruence (as witnessed by ρ), a is the left-hand side l of a reaction rule in a reactive context d, while a' is, up to structural congruence (witness ρ'), the corresponding right-hand side r of the reaction rule in the reactive context d.



The set $\mathcal{R}$ of reaction rules is, therefore, a set of base rules with which one generates the reaction relation $\longrightarrow\!\!\triangleright$ by closure under suitable contexts. For pragmatic reasons, we choose not to stipulate that $\mathcal{R}$ is to be closed under structural congruence; that is, in our formalism, under 2-cells. More precisely, we do not require that $\langle l', r' \rangle \in \mathcal{R}$ if there exist $\langle l, r \rangle \in \mathcal{R}$ and 2-cells $\alpha : l \Rightarrow l'$, $\beta : r \Rightarrow r'$. Indeed, modern process calculi often have very simple reaction rules and the closure under structural congruence comes at the point of defining the reaction relation. For example, the standard textbook definition of CCS [16] lists the single reaction rule

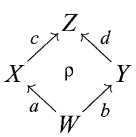
$$\overline{a.P + P' \mid \overline{a}.Q + Q' \longrightarrow\!\!\triangleright P \mid Q}$$

without listing, additionally, all of its structurally congruent variants. It is easy to check that, if we did choose to impose this condition ($\mathcal{R}$ closed under 2-cells) then the reaction relation $\longrightarrow\!\!\triangleright$, as well as the canonical labelled transition system (Definition 10) would remain unchanged.

The notion of GRPO formalises the idea of a context being the 'smallest' that enables a reaction in a G-reactive system, and is a conservative 2-categorical extension of Leifer and Milner's RPOs [14] (cf. [19,21] for a precise comparison).

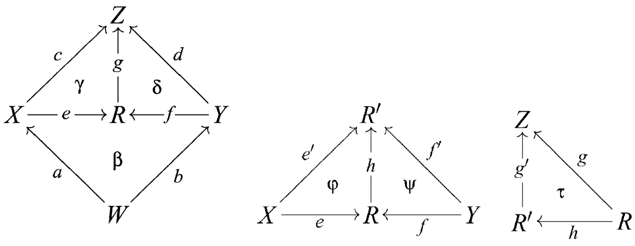
For readers acquainted with 2-dimensional category theory, GRPOs are defined in Definition 3. This is spelled out in elementary categorical terms in Proposition 4, taken from [19,21].

Definition 3 (*GRPOs*). Let $\rho\colon ca \Rightarrow db\colon W \to Z$ be a 2-cell (cf. diagram below) in a G-category $\mathbb{C}$. A *G-relative pushout* (GRPO) for ρ is a bipushout (cf. [9]) of the pair of arrows $(a, \mathbf{1}) : ca \to c$ and $(b, \rho) : ca \to d$ in the pseudo-slice category $\mathbb{C}/Z$.



(1)

Proposition 4. *Let $\mathbb{C}$ be a G-category. A candidate GRPO for $\rho: ca \Rightarrow db$ as in diagram* (1) *is a tuple $\langle R, e, f, g, \beta, \gamma, \delta \rangle$ such that $\delta b \bullet g\beta \bullet \gamma a = \rho$ –* cf. *diagram* (i).



(i) (ii) (iii)

A GRPO for ρ is a candidate which satisfies a universal property (viz. to be the 'smallest' such candidate). Namely, for any other candidate $\langle R', e', f', g', \beta', \gamma', \delta' \rangle$ there exists a quadruple $\langle h, \varphi, \psi, \tau \rangle$ where $h: R \to R'$, $\varphi: e' \Rightarrow he$ and $\psi: hf \Rightarrow f'$— cf. diagram (ii)—and $\tau: g'h \Rightarrow g$— diagram (iii)—which makes the two candidates compatible after the obvious pasting, i.e.

$$\tau e \bullet g'\varphi \bullet \gamma' = \gamma, \quad \delta' \bullet g'\psi \bullet \tau^{-1} f = \delta, \quad \psi b \bullet h\beta \bullet \varphi a = \beta'.$$

Such a quadruple, which we shall refer to as *mediating morphism*, must be *essentially unique*, that is unique up to a unique iso. Namely, for any other mediating morphism $\langle h', \varphi', \psi', \tau' \rangle$ there must exist a *unique* two cell $\xi: h \to h'$ which makes the two mediating morphisms compatible, i.e.:

$$\xi e \bullet \varphi = \varphi', \quad \psi \bullet \xi^{-1} f = \psi', \quad \tau' \bullet g'\xi = \tau.$$

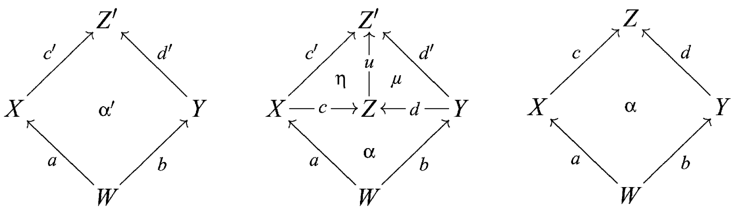
Observe that whereas RPOs are defined up to isomorphism, GRPOs are defined up to equivalence, as they are bicolimits.

The definition below plays an important role in the following development.

Definition 5 (*GIPO*). Diagram (1) of Definition 3 is said to be a *G-idem-pushout* (GIPO) if $\langle Z, c, d, \mathrm{id}_Z, \rho, \mathbf{1}_c, \mathbf{1}_d \rangle$ is its GRPO.

The next two lemmas explain the relationships between GRPOs and GIPOs.

Lemma 6 (*GIPOs from GRPOs*). *If $\langle Z, c, d, u, \alpha, \eta, \mu \rangle$ is a GRPO for* (i) *below, as illustrated in* (ii), *then* (iii) *is a GIPO*.

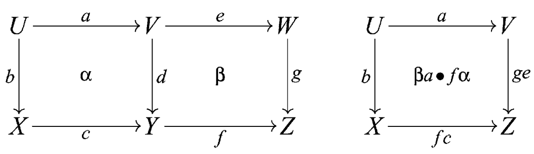


(i) (ii) (iii)

Lemma 7 (*GRPOs from GIPOs*). *If square* (iii) *above is a GIPO,* (i) *has a GRPO, and* $\langle Z, c, d, u, \alpha, \eta, \mu\rangle$ *is a candidate for it as shown in* (ii), *then* $\langle Z, c, d, u, \alpha, \eta, \mu\rangle$ *is a GRPO for* (i).

The following technical lemmas from [19,21] state the basic properties of GRPOs, upon which the congruence theorems below rest.

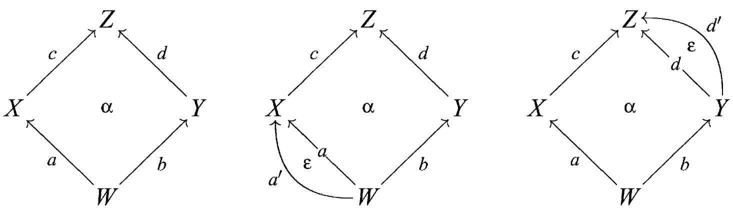
Lemma 8. *Suppose that diagram* (ii) *below has a GRPO.*



(i) (ii)

(1) *If both squares in* (i) *are GIPOs then the rectangle of* (i) *is a GIPO*;
(2) *If the left square and the rectangle of* (i) *are GIPOs then so is the right square.*

Lemma 9. *Suppose that diagram* (i) *below is a GIPO.*



(i) (ii) (iii)

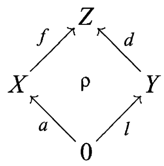
Then the regions obtained by pasting the 2-cells in (ii) and (iii) are GIPOs. Note that the proof relies on the fact that ε is, in both diagrams (i) and (ii), an isomorphism.

The previous lemma in particular implies that the following definition of labelled transition system derived from a G-reactive system is well defined.

Definition 10 (*LTS*). For **C** a G-reactive system whose underlying category $\mathbb{C}$ is a G-category, define GTS(**C**) as follows:

- the states GTS(**C**) are iso-classes of arrows $[a]\colon 0 \to X$ in **C**;
- for $a, a' : 0 \to X$ and $f : X \to Z$, there is a transition $[a] \xrightarrow{[f]} [a']$ if $fa \longrightarrow\!\triangleright a'$ via a GIPO; that is, if there exists a reaction rule $\langle l, r\rangle \in \mathcal{R}$, a reactive context $d \in \mathbb{D}$,

and 2-cells $\rho : fa \Rightarrow dl$ and $\rho' : dr \Rightarrow a'$ such that diagram (2) below is a GIPO.



(2)

Notice that this amounts to consider an *f*-labelled transition from *a* only if *f* is the 'smallest' context—in the technical sense defined by the universal property of GRPOs—to induce a particular reaction in *a*. The role of ρ is absolutely fundamental here: by determining the correspondence (isomorphism) between *a* and *dl*, it determines exactly the '*location*' of the redex being reduced, and therefore the reaction being fired. We will remark again on this with specific examples in later sections.

Henceforth we shall abuse notation and leave out the square brackets when writing transitions; i.e. we shall write simply $a \xrightarrow{f} a'$ instead of $[a] \xrightarrow{[f]} [a']$. Note that, taking into account the conclusions of Lemma 9, this abuse is quite harmless. Indeed, from a transition $[a] \xrightarrow{[f]} [a']$, we can conclude that $fa \longrightarrow\!\!\triangleright\, a'$ (working with the "concrete" underlying representatives) and that there exists a reaction rule $\langle l, r\rangle \in \mathcal{R}$ and a GIPO $\rho : fa \Rightarrow dl$ with $dr \cong a'$. In particular, it does not matter which representatives of equivalence classes one starts with.

Categories can be seen as a discrete G-categories, where the only 2-cells are the identities. Using this observation, each G-concept introduced above reduces to the corresponding 1-categorical concept. For instance, a GRPO (resp. GIPO) in a category is exactly a RPO (resp., IPO) of [14].

3. Congruence results for GRPOs

The following notion is the precondition needed to prove the congruence theorem.

Definition 11 (*Redex GRPOs*). A G-reactive system **C** is said to have *redex GRPOs* if its underlying G-category $\mathbb{C}$ has GRPOs for all squares like (2), where l is the left-hand side of a reaction rule $\langle l, r\rangle \in \mathcal{R}$, and $d \in \mathbb{D}$.

Observe that this means that there exists a GRPO for each possible interaction between a term and a context. We are therefore able to determine a 'smallest' label *f* to capture each of them in GTS(**C**). The main theorem of [19,21] is then expressed as follows.

Theorem 12 (*cf. Sassone and Sobociński [19,21]*). *Let* **C** *be a G-reactive system which has redex GRPOs. Then the largest bisimulation* $\sim$ *on* GTS(**C**) *is a congruence.*

The next three subsections complement this result by proving the expected corresponding theorems for trace and failure semantics, and by lifting them to the case of weak equivalences. Theorems and proofs in this section follow closely [12], as they are meant to show that GRPOs are as viable a tool as RPOs are.

3.1. Traces preorder

Trace semantics [18] is a simple notion of equivalence which equates processes if they can engage in the same sequences of actions. Even though it lacks the fine discriminating power of branching time equivalences, viz. bisimulations, it is nevertheless interesting because many safety properties can be expressed as conditions on sets of traces.

We say that a sequence $f_1 \cdots f_n$ of labels of $\mathrm{GTS}(\mathbf{C})$ is a trace of a if

$$a \xrightarrow{f_1} \cdots \xrightarrow{f_n} a_{n+1}$$

for some $a_1, \ldots, a_n$. The trace preorder $\lesssim_{\mathrm{tr}}$ is then defined as $a \lesssim_{\mathrm{tr}} b$ if all traces of a are also traces of b.

Theorem 13 (*Trace congruence*). *$\lesssim_{\mathrm{tr}}$ is a congruence.*

Proof. Assume $a \lesssim_{\mathrm{tr}} b$. We shall prove that $ca \lesssim_{\mathrm{tr}} cb$ for all contexts $c \in \mathbb{C}$. Suppose that

$$ca = \bar{a}_1 \xrightarrow{f_1} \bar{a}_2 \cdots \bar{a}_n \xrightarrow{f_n} \bar{a}_{n+1}.$$

We first prove that there exists a sequence, for $i = 1, \ldots, n$,

$$\begin{array}{ccccc} \cdot & \xrightarrow{a_i} & \cdot & \xrightarrow{c_i} & \cdot \\ {\scriptstyle l_i}\downarrow & \alpha_i & \downarrow{\scriptstyle g_i} & \beta_i & \downarrow{\scriptstyle f_i} \\ \cdot & \xrightarrow[d_i]{} & \cdot & \xrightarrow[d_i']{} & \cdot \end{array}$$

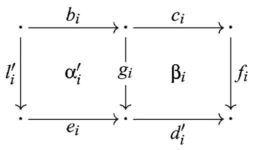
where $a_1 = a$, $c_1 = c$, $c_{i+1} = d_i'$, $\bar{a}_i = c_i a_i$, and each square is a GIPO.[1] The ith induction step proceeds as follows. Since $\bar{a}_i \xrightarrow{f_i} \bar{a}_{i+1}$, there exists $\gamma_i \colon f_i c_i a_i \Rightarrow \bar{d}_i l_i$, for some $\langle l_i, r_i \rangle \in \mathcal{R}$ and $\bar{d}_i \in \mathbb{D}$, with $\bar{a}_{i+1} = \bar{d}_i r_i$. Since $\mathbf{C}$ has redex GIPOs (cf. Definition 11), this can be split in two GIPOs: $\alpha_i \colon g_i a_i \Rightarrow d_i l_i$ and $\beta_i \colon f_i c_i \Rightarrow d_i' g_i$ (cf. diagram above). Take $a_{i+1} = d_i r_i$, and the induction hypothesis is maintained. In particular, we obtain a trace

$$a = a_1 \xrightarrow{g_1} a_2 \cdots a_n \xrightarrow{g_n} a_{n+1}$$

and, by the inductive hypothesis, $a \lesssim_{\mathrm{tr}} b$ must be matched by a corresponding trace of b. This means that, for $i = 1, .., n$, there exist GIPOs $\alpha_i' \colon g_i b_i \Rightarrow e_i l_i'$, for some $\langle l_i', r_i' \rangle \in \mathcal{R}$ and $e_i \in \mathbb{D}$, once we take b_{i+1} to be $e_i r_i'$. We can then paste each of such GIPOs together with

[1] Since the fact is not likely to cause confusion, we make no notational distinction between the arrows of $\mathbf{C}$ (e.g. in GRPOs diagrams) and the states and labels of $\mathrm{GTS}(\mathbf{C})$, where the latter are iso-classes of the former.

the corresponding $\beta_i \colon f_i c_i \Rightarrow d'_i g_i$ obtained above ($i = 1, \ldots, n$) and, using Lemma 8, conclude that there exist GIPOs $f_i c_i b_i \Rightarrow d'_i e_i l'_i$, as in the diagram below,



which means $\quad c_i b_i \xrightarrow{f_i} d'_i e_i r'_i$.

As $cb = c_1 b_1$, in order to construct a trace $cb = \bar{b}_1 \xrightarrow{f_1} \cdots \xrightarrow{f_n} \bar{b}_{n+1}$ and complete the proof, we only need to verify that for $i = 1, \ldots, n$, we have that $d'_i e_i r'_i = c_{i+1} b_{i+1}$. This follows at once, as $c_{i+1} = d'_i$ and $b_{i+1} = e_i r'_i$. $\square$

3.2. Failures preorder

Failure semantics [6] enhances trace semantics with limited branch-inspecting power. More precisely, failure sets allow to test when processes deplete the capability of engaging in certain actions.

Formally, for a a state of GTS($\mathbf{C}$), a *failure* of a is a pair $(f_1 \cdots f_n, X)$, where $f_1 \cdots f_n$ and X are, respectively, a non-empty sequence and a set of labels, such that:

- $f_1 \cdots f_n$ is a trace of a, $a \xrightarrow{f_1} \cdots \xrightarrow{f_n} a_{n+1}$;
- a_{n+1}, the final state of the trace, is *stable*, i.e. $a_{n+1} \not\longrightarrow\!\triangleright$;
- a_{n+1} *refuses* X, i.e. $a_{n+1} \not\xrightarrow{x}$ for all $x \in X$.

The failure preorder $\lesssim_f$ is defined as $a \lesssim_f b$ if all failures of a are also failures of b.

Theorem 14 (*Failures congruence*). *$\lesssim_f$ is a congruence.*

Proof. Assume $a \lesssim_f b$ to prove that $ca \lesssim_f cb$ for all contexts $c \in \mathbb{C}$. The proof extends the previous one of Theorem 13.

Let $(f_1 \cdots f_n, X)$, $n > 0$, be a failure of ca. We proceed exactly as above to determine a matching trace $cb = \bar{b}_1 \xrightarrow{f_1} \cdots \xrightarrow{f_n} \bar{b}_{n+1}$. In addition, we contextually need to prove that $\bar{b}_{n+1}$ is stable and refuses X, exploiting the corresponding hypothesis on $\bar{a}_{n+1}$.

First, we claim that a_{n+1} is stable. In fact, were it not, it would follow from $c_{n+1} \in \mathbb{D}$ (which equals d'_n) that also $\bar{a}_{n+1} = c_{n+1} a_{n+1} \longrightarrow\!\triangleright$. But this is impossible, since $\bar{a}_{n+1}$ is stable. Secondly, a_{n+1} refuses both

$$Y = \{g \mid \text{there exists a GIPO } \delta_g \colon x c_{n+1} \Rightarrow dg, \text{ for } x \in X,\ d \in \mathbb{D}\} \text{ and}$$
$$Z = \{g \mid \text{there exists a 2-cell } \varepsilon_g \colon dg \Rightarrow c_{n+1}, \text{ for } d \in \mathbb{D}\},$$

which can be seen as follows. If $a_{n+1} \xrightarrow{g}$ for $g \in Y$, then there exists a GIPO $\alpha \colon g a_{n+1} \Rightarrow d'l$, for some rule $\langle l, r \rangle$, which could be pasted together with δ_g to yield a GIPO $x c_{n+1} a_{n+1} \Rightarrow d d' l$, which is impossible since it means that $\bar{a}_{n+1} \xrightarrow{x}$, for $x \in X$. Similarly, if $a_{n+1} \xrightarrow{g}$ for $g \in Z$, pasting the corresponding GIPO with ε_g, we see that $\bar{a}_{n+1} \longrightarrow\!\triangleright$, contradicting the hypothesis that $\bar{a}_{n+1}$ is stable.

If follows then from the hypothesis $a \lesssim_f b$ that b_{n+1} is stable and refuses $Y \cup Z$. It is then easy to complete the proof by transferring stability and X-refusal to $\bar{b}_{n+1}$. First, suppose that $\bar{b}_{n+1} \longrightarrow\!\!\triangleright$. This means that there exists a 2-cell $dl \Rightarrow \bar{b}_{n+1}$. Since **C** has redex-GRPOs, we can factor c_{n+1} out and obtain from this a GRPOs $\alpha\colon gb_{n+1} \Rightarrow d'l$ together with a 2-cell $d''g \Rightarrow c_{n+1}$. But this would mean that $b_{n+1} \xrightarrow{g}$, for $g \in Z$, which is a contradiction.

Suppose finally that $\bar{b}_{n+1} \xrightarrow{x}$, for $x \in X$. Again, by definition of the transition relation, and exploiting the existence of redex-GRPOs, we find GRPOs $xc_{n+1} \Rightarrow d''g$ and $gb_{n+1} \Rightarrow d'l$, which mean that $b_{n+1} \xrightarrow{g}$, for $g \in Y$. □

3.3. Weak equivalences

Theorems 12–14 can be extended to weak equivalences, as below.

For f a label of GTS(**C**) define a *weak transition* $a \overset{f}{\Longrightarrow} b$ to be a mixed sequence of transitions and reductions $a \longrightarrow\!\!\triangleright^* \xrightarrow{f} \longrightarrow\!\!\triangleright^* b$. Observe that this definition identifies silent transitions in the LTS with reductions. As a consequence, care has to be taken to avoid interference with transitions of the kind $\xrightarrow{equi}$, synthesised from GRPOs and labelled by an equivalence. These transitions have essentially the same meaning as silent transitions (i.e. no context involved in the reduction), and must therefore be omitted in weak observations. The following lemma makes the reasoning above precise.

Lemma 15. *Suppose that* **C** *is a G-reactive system. If* $a \xrightarrow{e} b$ *with* e *an equivalence, then there exists* b' *such that* $a \longrightarrow\!\!\triangleright b'$. *Moreover,* $b' = e'b$, *where* e' *is the pseudo-inverse of* e.

Proof. Suppose that $\rho : dl \Rightarrow fa$ is a GIPO and f is an equivalence, that is, there exist isomorphisms $\alpha : \mathrm{id}_X \Rightarrow gf$ and $\beta : fg \Rightarrow \mathrm{id}_Y$. Then $\alpha^{-1}a \bullet g\rho : gdl \Rightarrow a$ and it remains to show that $gd \in \mathbb{D}$. But $\beta d : fgd \cong d$ and since $\mathbb{D}$ is closed under 2-cells, $fgd \in \mathbb{D}$. Then $gd \in \mathbb{D}$ since $\mathbb{D}$ is composition-reflecting. □

We may now consider the weak counterparts of the preorders and equivalences studied earlier.

Definition 16 (*Weak traces and failures*). A sequence $f_1 \cdots f_n$ of *non-equivalence* labels of GTS(**C**) is a weak trace of a if

$$a \overset{f_1}{\Longrightarrow} a_1 \cdots a_{n-1} \overset{f_n}{\Longrightarrow} a_n$$

for some $a_1, \ldots, a_n$. The weak trace preorder is then defined accordingly.

A *weak failure* of a is a pair $(f_1 \cdots f_n, X)$, where $f_1 \cdots f_n$ and X, are, respectively, a sequence and a set of *non-equivalence* labels, such that $f_1 \cdots f_n$ is a weak trace of a reaching a final state which is stable and refuses X. The weak trace preorder is defined accordingly.

Definition 17 (*Weak bisimulation*). A symmetric relation $\mathcal{S}$ on GTS(**C**) is a weak bisimulation if for all $a \; \mathcal{S} \; b$

$$a \xrightarrow{f} a' \quad f \text{ not an equivalence,} \quad \text{implies } b \overset{f}{\Longrightarrow} b' \text{ with } a' \; \mathcal{S} \; b';$$
$$a \longrightarrow\!\triangleright a' \quad \text{implies } b \longrightarrow\!\triangleright^{*} b' \text{ with } a' \; \mathcal{S} \; b'.$$

Using the definitions above Theorems 12–14 can be lifted, respectively, to weak traces, failures and bisimulation.

It is worth remarking that the congruence results, however, only hold for contexts $c \in \mathbb{D}$, as it is well known that non-reactive contexts—i.e. those c where $ca \longrightarrow\!\triangleright cb$ does not follow from $a \longrightarrow\!\triangleright b$, as e.g. the CCS context $c = - + c_0$, do not preserve weak equivalences. Alternative definitions of weak bisimulations are investigated in [12], and they are applicable *mutatis mutandis* to GRPOs.

4. Bunches and wires

In this section we consider an example of a simple G-category, recasting in the present framework the notion of bunch context first due to Leifer and Milner [14]. We will recall the notion of extensive category [2] and proceed to construct GRPOs in the G-category of bunches. The construction will only make use of the fact that **Ord**, the category whose objects are the node sets of our bunches, is extensive and has pushouts.

4.1. Category of bunch contexts

The category of 'bunches and wires' was introduced in [14] as a skeletal algebra of shared wirings, abstracting over the notion of *names* in, e.g. the π-calculus. Although elementary, its relevance resides in representing the simplest possible form of naming. In any case, its structure is complex enough to lack RPOs.

A bunch context of type $m_0 \to m_1$ consists of an ordered set of m_1 trees of depth one containing exactly m_0 holes. Leaves are labelled from an alphabet $\mathcal{K}$. These data represent m_1 bunches of unspecified controls (the leaves), together with m_0 places (the holes) where further bunch contexts can be plugged to. Before illustrating this graphically, let us proceed with the formal definition of Leifer and Milner's category of bunch contexts.

Definition 18. Let m_0 and m_1 be finite ordinals. A *concrete bunch context* $c : m_0 \to m_1$ is a tuple $c = \langle X, \mathsf{char}, \mathsf{rt} \rangle$, where X is a finite carrier, $\mathsf{rt}\colon m_0 + X \to m_1$ is a surjective function linking leaves (X) and holes (m_0) to their roots (m_1), and $\mathsf{char}\colon X \to \mathcal{K}$ is a leaf labelling function.

Given concrete bunch contexts $c_0\colon m_0 \to m_1$ and $c_1\colon m_1 \to m_2$, we can compose them to obtain a concrete bunch context $c_1 c_0\colon m_0 \to m_2$. Roughly, this involves 'plugging' the m_1 trees of c_0 orderly into m_1 holes of c_1; leaves and holes of c_0 are 'wired' to the roots of

c_1, alongside c_1's leaves. Formally, c_1c_0 is $(X, \mathsf{rt}, \mathsf{char})$ with

$$X = X_0 + X_1, \qquad \mathsf{rt} = \mathsf{rt}_1(\mathsf{rt}_0 + \mathrm{id}_{X_1}), \qquad \mathsf{char} = [\mathsf{char}_0, \mathsf{char}_1],$$

where $+$ and $[\_, \_]$ are, respectively, coproduct and copairing.

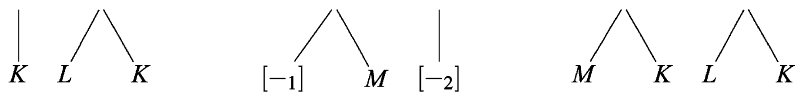
A *homomorphism* of concrete bunch contexts $\rho : c \Rightarrow c' : m_0 \to m_1$ is a function $\rho : X \to X'$ which respects rt and char, i.e. $\mathsf{rt}'\rho = \mathsf{rt}$ and $\mathsf{char}'\rho = \mathsf{char}$. An isomorphism is a bijective homomorphism.

Definition 19. The category of *bunch contexts* $\mathbf{Bun}_0$ has

- objects the finite ordinals (cf. Section 1), written as $m_0, m_1, \ldots$
- arrows from m_0 to m_1 are isomorphism classes $[a] : m_0 \to m_1$ of concrete bunch contexts.

Given an object m_0, the identity is (the isomorphism class of) $\langle \emptyset, !, \mathrm{id} \rangle : m_0 \to m_0$. Isomorphic bunch contexts are equated, making composition associative and $\mathbf{Bun}_0$ a category.

The pictures below illustrate the concept of bunch context. The leftmost diagram represents a bunch context $[a] : 0 \to 2$ with $X = 3$, $\mathsf{char}(1) = \mathsf{char}(3) = K$, $\mathsf{char}(2) = L$, $\mathsf{rt}(1) = 1$ and $\mathsf{rt}(2) = \mathsf{rt}(3) = 1$. The middle diagram represents a bunch context $[b] : 2 \to 2$ with $X = \{*\}$, $\mathsf{char}(*) = M$, $\mathsf{rt}(1) = \mathsf{rt}(*) = 1$ and $\mathsf{rt}(2) = 1$.

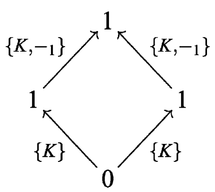


[a]: 0→2 [b]: 2→2 [ba]: a→2

The final diagram represents $[ba] : 0 \to 2$, the result of composing a and b.

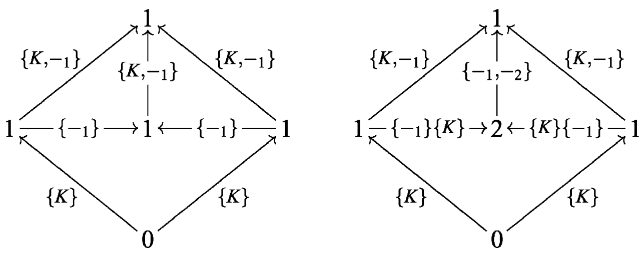
A bunch context $[c] : m_0 \to m_1$ can alternatively be depicted as a string of m_1 non-empty multisets on $\mathcal{K} + m_0$ (the bunches of leaves and holes connected to the same root), with the proviso that elements m_0 must appear exactly once in the string. In the examples, we represent elements of m_0 as numbered holes $-_i$. For instance, the three pictures above can be written, respectively, as $\{K\}\{L, K\}$, $\{-_1, M\}\{-_2\}$, and $\{M, K\}\{L, K\}$.

As we mentioned before, RPOs do not exist in $\mathbf{Bun}_0$. Indeed, consider (i) below.



(i)

The following diagrams show two candidate RPOs (ii) and (iii) which are easily proved not to have a common 'lower bound' candidate.



(ii) (iii)

The point here is that by taking the arrows of $\mathbf{Bun}_0$ up to isomorphism we lose information about *how* bunch contexts equal each other. Diagram (i), for instance, can be commutative in two different ways: the K in the bottom left part may correspond either to the one in the bottom right or to the one in the top right, according to whether we read $\{K, -_1\}$ or $\{-_1, K\}$ for the top rightmost arrow. The point is therefore exactly which *occurrences* of K correspond to each other. The fundamental contribution of G-categories is to equip our structures of interest with an explicit mechanism (viz. the 2-cells) to track such correspondences. Fed into the categorical machinery of relative pushouts, this gives GRPOs the power to '*locate*' reaction beyond the blurring effect of a structural congruence (in this case, the commutation of elements inside a multiset). To illustrate our ideas concretely, let us grant $\mathbf{Bun}_0$ its natural 2-categorical structure.

Definition 20. The 2-category of bunch contexts **Bun** has:

- objects the finite ordinals (cf. Section 1), denoted $m_0, m_1, \ldots$
- arrows $c = (x, \mathsf{char}, \mathsf{rt})\colon m_0 \to m_1$ consist of a finite ordinal x, a surjective function $\mathsf{rt}\colon m_0 \oplus x \to m_1$ and a labelling function $\mathsf{char}\colon x \to \mathcal{K}$.
- 2-cells ρ are isomorphisms between bunches' carriers which preserve the structure, that is respect char and rt.

Composition of arrows and 2-cells is defined in the obvious way. Notice that since $\oplus$ is associative, composition in **Bun** is associative. Therefore **Bun** is a G-category.

Replacing the carrier set X with a finite ordinal x allows us to avoid the unnecessary burden of working in a bicategory, which would arise because sum on sets is only associative up to isomorphism. Observe that this simplification is harmless since the set-theoretical identity of the elements of the carrier is irrelevant. We remark, however, that GRPOs are naturally a bicategorical notion and would pose no particular challenge in that setting. In particular, in [22] we use a bicategorical framework in order to apply the theory of GRPOs to derive bisimulation congruence for generic graph rewriting systems.

4.2. Extensive categories

When constructing GRPOs, we have tried to use general categorical constructions defined using universal properties. This not only simplifies the proofs, freeing one from unnecessary set-theoretical detail but also makes them more robust in that the proofs lift relatively easily to other models.

In particular, in the proof of Theorem 23 below, we use only the fact that **Ord** is an extensive [2] category with pushouts. An extensive category can be thought of roughly as a category where coproducts are in many ways 'well-behaved,' where the paradigm for good behaviour comes from the category of sets and functions. For the reader's convenience we reproduce a definition below.

Definition 21. A category **C** is *extensive* when

- it has finite coproducts,
- it admits pullbacks along injections of binary coproducts,
- given a commutative diagram,

$$\begin{array}{ccccc} C_1 & \xrightarrow{c_1} & C & \xleftarrow{c_2} & C_2 \\ {\scriptstyle g_1}\downarrow & & {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g_2} \\ A & \xrightarrow[i_1]{} & A+B & \xleftarrow[i_2]{} & B \end{array}$$

where the bottom row is a coproduct, the two squares are pullbacks if and only if the top row is a coproduct diagram.

In order to provide the reader with some intuition for the good behaviour of coproducts in extensive categories, we recall below some properties of extensive categories. Notice that these simply express expected properties of coproducts in **Set**, the category of sets and functions.

Lemma 22. *Let* **C** *be an extensive category. Then,*

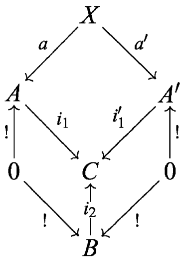
(i) *sums are disjoint*; *that is, the pullback of the two injections of a binary coproduct is the initial object,*
(ii) *coproduct injections are mono,*
(iii) *if* $A \xrightarrow{i_1} C \xleftarrow{i_2} B$ *and* $A' \xrightarrow{i_1'} C \xleftarrow{i_2} B$ *are coproduct diagrams, then there exists a unique isomorphism* $\varphi : A \to A'$ *such that* $i_1'\varphi = i_1$,
(iv) *suppose that* $\varphi : A+C \to B+C$ *is an isomorphism such that* $\varphi i_2 = i_2 : C \to B+C$; *then there exists a unique isomorphism* $\psi : A \to B$ *so that* $\varphi = \psi + C$,

Proof. We begin by proving (i) and (ii). In the following diagram, the bottom row and the top row are coproduct diagrams,

$$\begin{array}{ccccc} 0 & \xrightarrow{!} & B & \xleftarrow{\mathrm{id}} & B \\ {\scriptstyle !}\downarrow & & {\scriptstyle i_2}\downarrow & & \downarrow{\scriptstyle \mathrm{id}} \\ A & \xrightarrow[i_1]{} & A+B & \xleftarrow[i_2]{} & B \end{array}$$

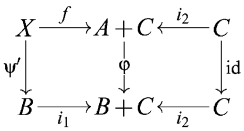
and the two squares are clearly commutative. Using the definition of extensivity, the two squares are, therefore, pullbacks. The left square being a pullback means that coproducts are disjoint. The fact that the right-hand side is a pullback implies that i_2 is mono. By a similar argument, i_1 is also mono.

We shall now proceed with (iii). Consider the following diagram:



using (i), we deduce that the two lower regions are pullbacks. Let the upper region be a pullback. Using extensivity, $0 \xrightarrow{!} A \xleftarrow{a} X$ and $0 \xrightarrow{!} A' \xleftarrow{a'} X$ are coproduct diagrams, and therefore, it follows that a and a' are isomorphisms. Let $\varphi = a'a^{-1}$, which satisfies $i'_1\varphi = i_1$, as required. Given another such φ', we have $i'_1\varphi = i_1 = i'_1\varphi$. We can now use (ii) to deduce that i'_1 is mono, and therefore, that $\varphi = \varphi'$.

It remains to prove (iv). Consider the diagram below, where

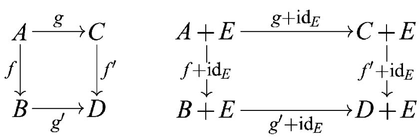


the right square can be verified to be pullback, using the fact that φ is mono. Suppose that the left-square is a pullback. Note that ψ' is an isomorphism, since it is a pullback of an isomorphism. Using extensivity, the resulting top row is a coproduct diagram, and using part (iii), we can deduce that there exists an isomorphism $\varphi : A \to X$ such that $f\varphi = i_1 : A \to A + C$. Letting $\psi = \psi'\varphi$, we obtain $\varphi = \psi + C$. The fact that $i_1 : B \to B + C$ is mono implies uniqueness. □

Examples of extensive categories include **Set**, and more generally any topos. The category of topological spaces and continuous functions **Top** is extensive. Any category with freely generated coproducts is extensive [2].

The following simple fact will prove useful for us later in this section. It holds in any category, that is, it does not require the assumption of extensivity.

Proposition 23. *Suppose that the diagram* (i)*, below, is a pushout. Then diagram* (ii) *is also a pushout.*



(i) (ii)

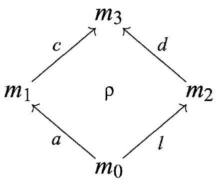
4.3. Construction of GRPOs

Theorem 23. **Bun** *has GRPOs.*

Proof. The proof is divided into two parts. In the first part we give the construction, and in the second part we verify that the universal property holds. □

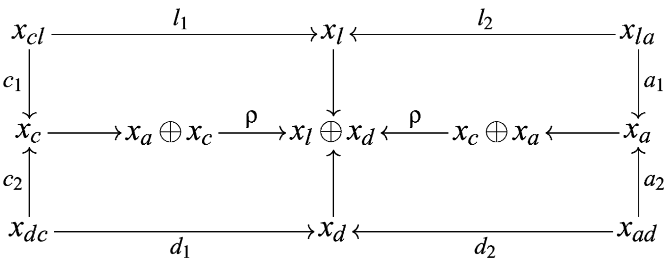
4.4. Construction of GRPO

Suppose that we have an isomorphic 2-cell $\rho\colon ca \Rightarrow dl$ as illustrated below.



The intuition here is that, for an 'agent' a and a left-hand side l of some reaction rule, we are given bunch contexts c and d so that ca is dl, up to ρ (in symbols, $ca \cong_\rho dl$). We shall find the smallest upper bound of a and l which 'respects' ρ.

Using $\rho : x_a \oplus x_c \to c_c \oplus x_d$ and the injections into the chosen coproduct in **Ord** (which in the diagrams below we leave unlabelled, or denote generically with i_1 and i_2), we take four pullbacks obtaining the following diagram. Due to the extensivity of **Ord**, each of l_i, a_i, d_i and c_i where $i \in \{1, 2\}$ is a coproduct injection.



(i)

Here, one can think of x_{dc} as the nodes common to bunch contexts d and c, when ca is translated, via ρ, to dl. Similarly, x_{cl} are the nodes common to c and l, x_{la} are the nodes common to l and a, while x_{ad} are the nodes common to a and d. We shall show that x_{cl} and x_{ad} form the nodes of the minimal candidate.

Let $x_e = x_{cl}$, $x_f = x_{ad}$ and $x_g = x_{dc}$. Using the morphisms from the diagram above as building blocks, we can construct bijections $\gamma\colon x_c \to x_e \oplus x_g$, $\delta\colon x_f \oplus x_g \to x_d$

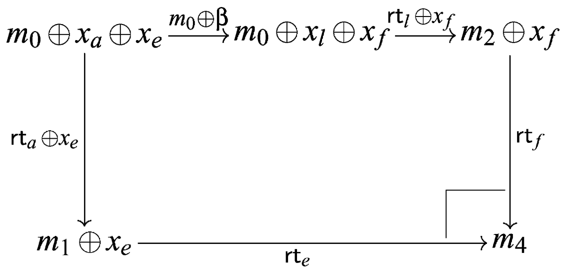
and $\beta: x_a \oplus x_e \to x_l \oplus x_f$ such that

$$x_l \oplus \delta.\beta \oplus x_g.x_a \oplus \gamma = \rho, \tag{3}$$

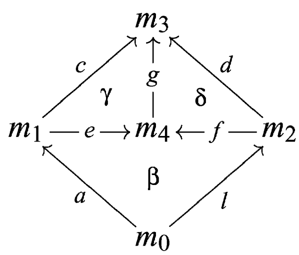
more precisely, $\gamma = [c_1, c_2]^{-1}$, $\delta = [d_2, d_1]$ and β is the following composition,

$$x_a \oplus x_e \xrightarrow{[a_2,a_1]^{-1}\oplus x_e} x_f \oplus x_{la} \oplus x_e \xrightarrow{x_f \oplus [l_2,l_1]} x_f \oplus x_l \xrightarrow{tw} x_l \oplus x_f,$$

where $tw : x_f \oplus x_l \to x_l \oplus x_f$ is the 'twist' isomorphism. Let rt_e and rt_f be morphisms making (ii) below a pushout diagram.



(ii)

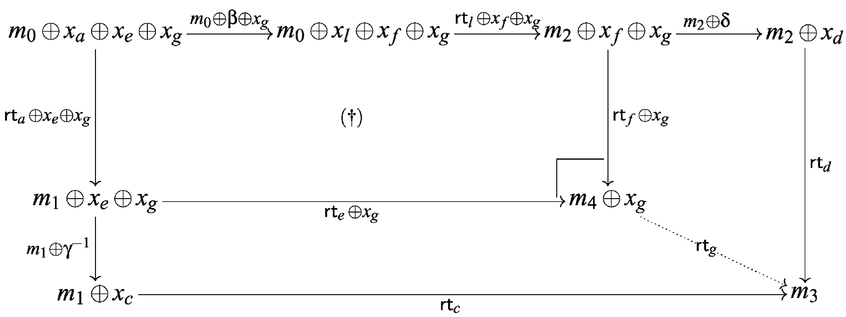


(iii)

We can then define char_e, char_f and char_g (from γ, δ, char_c and char_d) so as to form bunch contexts *e*, *g* and *f* which make (iii) above a candidate GRPO. Notice that the commutativity of (ii) implies that β is a bunch homomorphism.

It remains to define rt_g and prove that γ and δ are bunch homomorphisms.

Consider the diagram (iv), below.



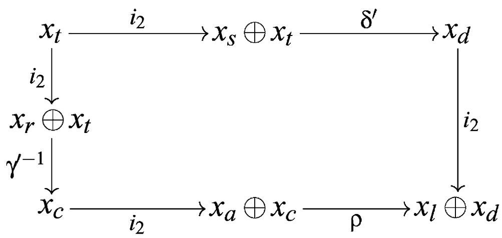
(iv)

The exterior of (iv) is commutative since ρ is a bunch homomorphism, this can be verified by precomposing with $m_0 \oplus x_a \oplus \gamma : m_0 \oplus x_a \oplus x_c \to m_0 \oplus x_a \oplus x_e \oplus x_g$ and using (3). Now, since (ii) is a pushout, an application of Lemma **??** yields that (†) is a pushout. We obtain a morphism $\mathsf{rt}_g : m_4 \oplus x_g \to m_3$ which makes the remaining regions of (iv) commute. These two remaining regions imply that γ and δ are bunch homomorphisms. We can deduce that rt_g is epi since $\mathsf{rt}_g.\mathsf{rt}_f \oplus x_g = \mathsf{rt}_d.m_2 \oplus \delta$, rt_d is epi and $m_2 \oplus \delta$ is an isomorphisms.

Thus, diagram (ii) is indeed a candidate GRPO for the 2-cell $\rho\colon ca \Rightarrow dl$.

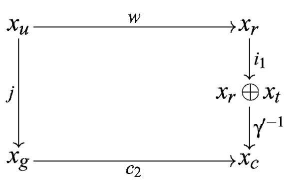
4.5. *Verification of the universal property*

Suppose that $\langle m_5, r, s, t, \beta', \gamma', \delta'\rangle$ is another candidate GRPO for ρ, i.e. $\delta' l \bullet t\beta' \bullet \gamma' a = \rho$. A diagram chase shows that the diagram (v), below, is commutative.

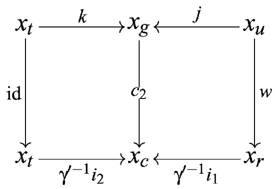


(v)

Since x_g with maps $c_2 : x_g \to x_c$ and $d_1 : x_g \to x_d$ is a pullback of $\rho i_2 : x_c \to x_l \oplus x_d$ and $i_2 : x_d \to x_l \oplus x_d$ – cf. (i) –, there exists a monomorphisms $k\colon x_t \to x_g$ such that $\gamma'^{-1} i_2 = c_2 k$ and $\delta' i_2 = d_1 k$.



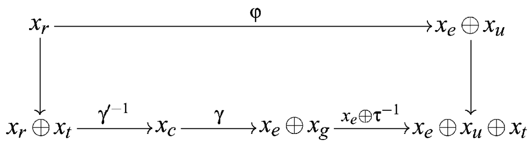
(vi)



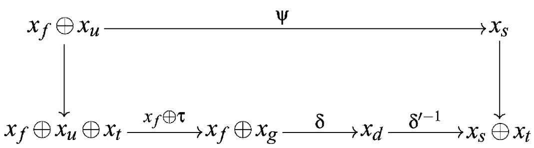
(vii)

Take the pullback (vi). Using extensivity, $x_u \xrightarrow{\;j\;} x_g \xleftarrow{\;k\;} x_t$ is a coproduct diagram, as shown by (∇), where the square on the right-hand side is (vii). The commutative square on the left-hand side can be verified to be a pullback since c_2, being a coproduct injection in an extensive category, is mono. We shall show that x_u is the set of nodes of a mediating bunch context $u : m_4 \to m_5$.

Let τ denote the isomorphism $[j, k]\colon x_u \oplus x_t \to x_g$. By the definition of τ, the composites at the bottom edges of diagrams (viii) and (ix), below, act as the identity on the second injections (x_t). Applying part (iv) of Lemma 22,



(viii)



(ix)

we obtain isomorphisms φ and ψ so that diagrams (viii) and (ix) are pullbacks.

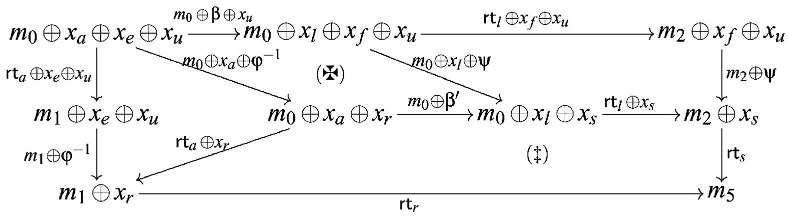
The commutativity of these pullback diagrams implies that

$$x_e \oplus \tau . \varphi \oplus x_t . \gamma' = \gamma \tag{4}$$

and $\delta' . \psi \oplus x_t . x_f \oplus \tau^{-1} = \delta$. These two equations, together with those which relate ρ to β, γ, δ and β', γ',δ', give

$$x_l \oplus \psi . \beta \oplus x_u . x_a \oplus \varphi = \beta'. \tag{5}$$

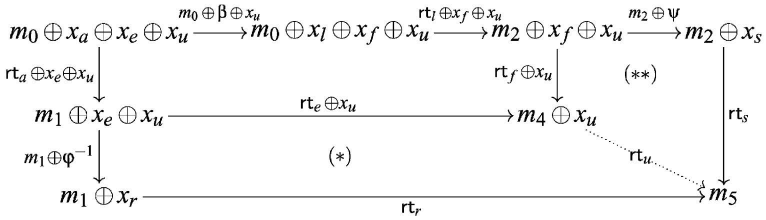
Now consider diagram (x), below.



(x)

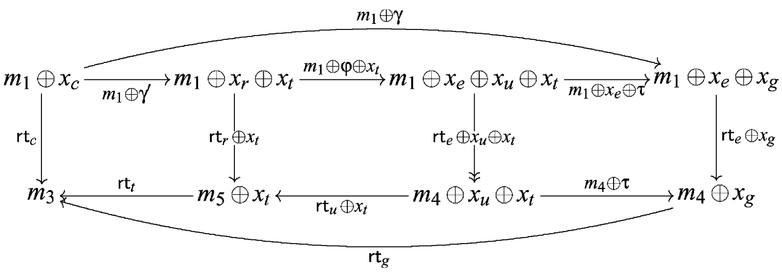
The commutativity of region (✠) follows from Eq. (5). Region (‡) is commutative because β' is a bunch homomorphism. Thus, the entire diagram is commutative. The commutativity of (x) implies that the outside of diagram (xi) is commutative. Applying the conclusion of Lemma **??** to diagram (ii) implies that the inner region is a pushout diagram, and therefore, that there exists a unique morphism $\mathsf{rt}_g : m_4 \oplus x_g \to m_3$ which renders regions (*)

and (**) commutative.



(xi)

Thus $u: m_4 \to m_5$ is a bunch context. Regions (*) and (**) of (xi) imply that $\varphi: r \Rightarrow ue$ and $\psi: uf \Rightarrow s$, respectively, are homomorphisms. To see that $\tau: tu \Rightarrow g$ is a homomorphism, consider diagram (xii), below.



(xii)

The two rectangles on the left are commutative since γ' and φ are homomorphisms. Using (4), the top row is equal to $m_1 \oplus \gamma$. Using the fact that γ is a homomorphism (the commutativity of the outside region) and the surjectivity of the marked arrow in the above diagram, we conclude that $\mathsf{rt}_g.m_4 \oplus \tau = \mathsf{rt}_t.\mathsf{rt}_u \oplus x_t$. Thus $\langle u, \varphi, \psi, \tau\rangle$ is a mediating morphism.

Now consider any other mediating morphism $\langle u', \tau', \varphi', \psi'\rangle$. We have that

$$x_e \oplus \tau'.\varphi' \oplus x_t.\gamma' = \gamma, \tag{6}$$

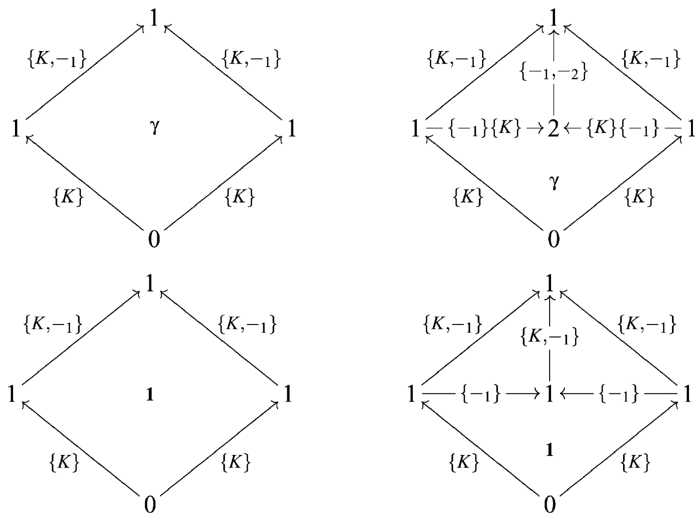
$\delta'.\psi' \oplus x_t.x_f \oplus (\tau')^{-1} = \delta$ and $x_l \oplus \psi'.\beta \oplus x_{u'}.x_u \oplus \varphi' = \beta'$.

Using (4) and (6), we have $x_e \oplus \tau.\varphi \oplus x_t.\gamma' = \gamma = x_e \oplus \tau'.\varphi' \oplus x_t.\gamma'$ and therefore $x_e \oplus \tau.\varphi \oplus x_t = x_e \oplus \tau'.\varphi' \oplus x_t$. Precomposing with the second injection $i_2 : x_t \to x_r \oplus x_t$ allows us to deduce $\tau i_2 = \tau' i_2 : x_t \to x_g$. Thus, we have coproduct diagrams $x_u \xrightarrow{\tau i_1} x_g \xleftarrow{\tau i_2} x_t$ and $x_{u'} \xrightarrow{\tau' i_1} x_g \xleftarrow{\tau i_2} x_t$. Using (iii) of Lemma 22, we obtain a unique isomorphism $\xi : x_u \to x_{u'}$ such that $\tau' i_1 \xi = \tau i_1$, and therefore $\tau'.\xi \oplus x_t = \tau$.

Now, using (4) and (6) again, $x_e \oplus \tau'.\varphi' \oplus x_t = \gamma.(\gamma')^{-1} = x_e \oplus \tau.\varphi \oplus x_t = x_e \oplus \tau'.x_e \oplus \xi \oplus x_t.\varphi \oplus x_t$, from which follows $\varphi' \oplus x_t = (x \oplus \xi.\varphi) \oplus x_t$. A straightforward application of part (iv) of Lemma 22 yields that $\varphi' = x \oplus \xi.\varphi$. Similarly, one may derive $\psi.x_f \oplus \xi^{-1} = \psi'$.

We remark again that the proof relies nicely on the fact that **Ord** is an extensive category with pushouts, and it goes through unchanged for any other such category.

Examples. Let $\gamma\colon 2 \to 2$ be the function taking $1 \mapsto 2$ and $2 \mapsto 1$. We give below on the right the GRPOs for the squares on the left.



Of course, the ambiguity in $\mathbf{Bun}_0$ about 'how' the diagrams commute—which ultimately leads to $\mathbf{Bun}_0$ failing to have RPOs—is resolved here by the explicit presence of **1** or γ. And in both cases, GRPOs exist.

5. 2-categories vs. precategories

Other categories which, besides $\mathbf{Bun}_0$, lack RPOs include the closed *shallow action contexts* [12,13] and *bigraph contexts* [17,8]. The solution adopted by Leifer [13] and later by Milner [17] is to introduce a notion of a *well-supported precategory*, where the algebraic structures at hand are decorated by finite 'support sets'. The result is no longer a category—since composition of arrows is defined only if their supports are disjoint—but from any such precategory one can generate two categories which jointly allow the derivation of a bisimulation congruence via a *functorial reactive system*. These categories are the so-called *track* category, where support information is built into the objects, and the *support quotient* category, where arrows are quotiented by the support structure. The track category has enough RPOs and is mapped to the support quotient category via a 'well-behaved' *functor*, so as to transport RPOs adequately. We remark that Jensen and Milner [7] have recently simplified the theory by developing their arguments internally in precategories, in order to bypass working with the track category.

In this section we present a general translation from arbitrary precategories to G-categories. Our main result shows that the LTS derived using precategories and functorial reactive systems is identical to the LTS derived using GRPOs. We begin with a brief recapitulation of the definitions from [13], to which the reader is referred for motivations and details.

Definition 24. A *precategory* $\mathbb{A}$ consists of the same data as a category. The composition operator $\circ$ is, however, a partial function which satisfies:
(1) for any arrow $f : A \to B$, $\mathrm{id}_B \circ f$ and $f \circ \mathrm{id}_A$ are defined and $\mathrm{id}_B \circ f = f = f \circ \mathrm{id}_A$;
(2) for any $f : A \to B$, $g : B \to C$, $h : C \to D$, $(h \circ g) \circ f$ is defined iff $h \circ (g \circ f)$ is defined and then $(h \circ g) \circ f = h \circ (g \circ f)$.

Definition 25. Let $\mathbf{Set}_f$ be the category of finite sets. A *well supported precategory* is a pair $\langle \mathbb{A}, |-| \rangle$, where $\mathbb{A}$ is a precategory and $|-|$ is a map from the arrows of $\mathbb{A}$ to $\mathbf{Set}_f$, the so-called support function, satisfying:
(1) $g \circ f$ is defined iff $|g| \cap |f| = \emptyset$, and if $g \circ f$ is defined then $|g \circ f| = |g| \cup |f|$;
(2) $|\mathrm{id}_A| = \emptyset$.
For any $f : A \to B$ and any injective function ρ in $\mathbf{Set}_f$ the domain of which contains $|f|$ there exists an arrow $\rho \cdot f : A \to B$ called the *support translation* of *f* by ρ. The following axioms are to be satisfied.

1. $\rho \cdot \mathrm{id}_A = \mathrm{id}_A$;
2. $\mathrm{id}_{|f|} \cdot f = f$;
3. $\rho_0 |f| = \rho_1 |f|$ implies $\rho_0 \cdot f = \rho_1 \cdot f$;
4. $\rho \cdot (g \circ f) = \rho \cdot g \circ \rho \cdot f$;
5. $(\rho_1 \circ \rho_0) \cdot f = \rho_1 \cdot (\rho_0 \cdot f)$;
6. $|\rho \cdot f| = \rho |f|$.

We illustrate these definitions giving a precategorical definition of bunches and wiring (cf. Section 4).

Example 26 (*Bunches*). The precategory of bunch contexts **A-Bun** has objects as in $\mathbf{Bun}_0$. However, differently from $\mathbf{Bun}_0$, arrows are concrete bunch contexts, they are not isomorphism classes. The support of $c = (X, \mathsf{char}, \mathsf{rt})$ is *X*. Composition $c_1 c_0 = (X, \mathsf{char}, \mathsf{rt})$: $m_0 \to m_2$ of $c_0 \colon m_0 \to m_1$ and $c_1 \colon m_1 \to m_2$ is defined if $X_0 \cap X_1 = \emptyset$ and, if so, we have $X = X_0 \cup X_1$. Functions char and rt are defined in the obvious way. The identity arrows are the same as in $\mathbf{Bun_0}$. Given an injective function $\rho \colon X \to Y$, the support translation $\rho \cdot c$ is $(\rho X, \mathsf{char}\, \rho^{-1}, \mathsf{rt}\, (\mathrm{id}_{m_0} + \rho^{-1}))$. It is easy to verify that this satisfies the axioms of precategories.

The definitions below recall the construction of the track and the support quotient categories from a well-supported precategory $\mathbb{A}$. The track has the support information built into the objects. On the contrary, the support quotient consists of isomorphism classes of arrows with respect to support translation. Both constructions yield categories relevant to $\mathbb{A}$. The track category, in particular, is concrete enough to admit RPOs in important cases. We shall question shortly the relationship between these constructions and our notion of G-categories.

Definition 27. The *track* of $\mathbb{A}$ is a category $\widehat{\mathbb{C}}$ with
- objects: pairs $\langle A, M \rangle$ where $A \in \mathbb{A}$ and $M \in \mathbf{Set}_f$;
- arrows: $\langle A, M \rangle \xrightarrow{f} \langle B, N \rangle$ where $f \colon A \to B$ is in $\mathbb{A}$, $M \subseteq N$ and $|f| = N \backslash M$.

Composition of arrows is as in $\mathbb{A}$. Observe that the definition of $|f|$ ensures that composition is total. We leave it to the reader to check that the data defines a category (cf. [13]).

Definition 28. The *support quotient* of $\mathbb{A}$ is a category $\mathbb{C}$ with
- objects: as in $\mathbb{A}$;
- arrows: equivalence classes of arrows of $\mathbb{A}$, where f and g are equated if there exist a bijective ρ such that $\rho \cdot f = g$.

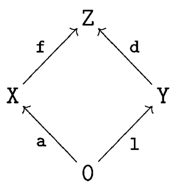
Example 29 (*Bunches*). The support quotient of **A-Bun** is $\mathbf{Bun}_0$.

There is an obvious functor $F : \widehat{\mathbb{C}} \to \mathbb{C}$, the *support-quotienting functor*. There is a straightforward way of defining a reactive system over a well-supported precategory, akin to the definition of G-reactive system for a G-category (Definition 2).

Definition 30. A *reactive system* **A** over a well-supported precategory $\mathbb{A}$ consists of
(1) a collection $\mathbb{D}$ of arrows of $\mathbb{A}$, the reactive contexts; it is required to be closed under support translation and to be composition-reflecting,
(2) a distinguished object $0 \in \mathbb{A}$,
(3) a set of pairs $\mathcal{R} \subseteq \bigcup_{A \in \mathbb{A}} \mathbb{A}(0, A) \times \mathbb{A}(0, A)$ called the reaction rules. These are required to be pointwise closed under support translation, that is, given $\langle l, r \rangle \in \mathcal{R}$ and support translations ρ, ρ' whose domains contain, respectively, $|l|$ and $|r|$, we require that $\left\langle \rho \cdot l, \rho' \cdot r \right\rangle \in \mathcal{R}$.

In the following we use the typewriter font for objects and arrows of $\widehat{\mathbb{C}}$. We make the notational convention that any $\mathtt{A}$ and $\mathtt{f}$ in $\widehat{\mathbb{C}}$ are such that $F(\mathtt{A}) = A$ and $F(\mathtt{f}) = f$.

Definition 31. Let **A** be a reactive system over a well-supported precategory $\mathbb{A}$. Let $\widehat{\mathbb{C}}$ and $\mathbb{C}$ be the corresponding track and support quotient. The LTS $\mathrm{FLTS}^c(\mathbf{A})$ has
- States: arrows $a : 0 \to X$ in $\mathbb{C}$;
- Transitions: $a \xrightarrow{f} dr$ if and only if there exist $\mathtt{a}, \mathtt{l}, \mathtt{f}, \mathtt{d}$ in $\hat{\mathbb{C}}$ with $\langle \mathtt{l}, \mathtt{r} \rangle \in \mathcal{R}$, $\mathtt{d} \in \mathbb{D}$, and such that



is an IPO in $\widehat{\mathbb{C}}$.

It is proved in [13] that the support-quotienting functor F satisfies the conditions required by the theory of functorial reactive systems [12,13]. Thus, if the category $\widehat{\mathbb{C}}$ has enough RPOs, then the bisimulation on $\mathrm{FLTS}^c(\mathbf{A})$ is a congruence.

All the theory presented so far can be elegantly assimilated into the theory of GRPOs. In [13], Leifer predicted that instead of precategories, one could consider a bicategorical notion of RPO in a bicategory of supports. This is indeed the case, with GRPOs being the bicategorical notion of RPO. However, working with ordinals for support sets we can avoid bicategories and, as in the case of **Bun**, stay within the realm of 2-categories. It is worth noticing, however, that a bicategory of supports as above and the G-category we introduce below would be biequivalent (in the sense of, e.g. [25]). In the following, we make use of a chosen isomorphism $t_x \colon x \to ord(x)$, as defined in Section 1.

Definition 32 (*G-Category of supports*). Given a well-supported precategory $\mathbb{A}$, the G-category of supports $\mathbb{B}$ has

- objects – as in $\mathbb{A}$;
- arrows – $f \colon A \to B$ where $f \colon A \to B$ is an arrow of $\mathbb{A}$ and $|f|$ is an ordinal;
- 2-cells – ρ: $f \Rightarrow g$ for ρ a 'structure preserving' support bijection, i.e. $\rho \cdot f = g$ in $\mathbb{A}$.

Composition is defined as follows. Given $f : A \to B$ and $g : B \to C$,

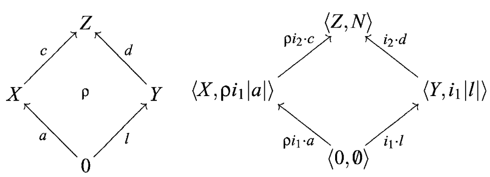
$$g \circ_{\mathbb{B}} f = i_2 \cdot g \circ_{\mathbb{A}} i_1 \cdot f,$$

where $|f| \xrightarrow{i_1} |f| \oplus |g| \xleftarrow{i_2} |g|$ is the chosen coproduct diagram in **Ord**. Given an arrow f in $\mathbb{A}$, we use $\widetilde{f} = t_{|f|} \cdot f$ in $\mathbb{B}$, the 'canonical representative' of f in $\mathbb{B}$. To simplify the notation in the following we write t_f for $t_{|f|}$. Observe that, with these conventions, $t_f \colon |f| \to |\widetilde{f}|$.

Notice that the translation can be easily extended to reactive systems. That is, starting with a reactive system **A** over a well-supported precategory $\mathbb{A}$, one uses the translation of Definition 32 to obtain a G-reactive system **B** over the G-category of supports $\mathbb{B}$. Observe that such structure gives a concise representation of both the quotient, via the 2-structure, and the support, with no need to include the latter explicitly in the objects. The following theorem guarantees that the LTS generated is the same as the one generated with the theory of functorial reactive systems.

Theorem 33. *Let* **A** *be a reactive system over a well-supported precategory* $\mathbb{A}$, *and let* **B** *and* $\mathbb{B}$ *be, respectively, the G-reactive system and G-category obtained as above. Then,* $\mathrm{FLTS}^c(\mathbf{A}) = \mathrm{GTS}(\mathbf{B})$.

Proof. Let $\widehat{\mathbb{C}}$ be the track of $\mathbb{A}$. It is enough to present a translation between GIPOs in $\mathbb{B}$ and IPOs in $\widehat{\mathbb{C}}$ which preserves the resulting label in the derived LTS. Suppose that (i) below is a GIPO.



(i) (ii)

Then we claim that (ii) above is an IPO in $\widehat{\mathbb{C}}$, for $N = |l| \oplus |d|$ and i_1, i_2 injections into coproducts in **Ord**. (Observe that the i's in the two sides of the diagram refer to different coproducts; we trust this will not cause confusion.) Note that (ii) is commutative since ρ is by definition a structure-preserving support bijection and, therefore, $\rho(i_2 \cdot c \circ i_1 \cdot a) = i_2 \cdot d \circ i_1 \cdot l$.

Suppose that $\langle \langle R, M \rangle, e, f, g \rangle$ is a candidate for (ii). We then show how to find β, γ and δ such that $\langle R, \widetilde{e}, \widetilde{f}, \widetilde{g}, \beta, \gamma, \delta \rangle$ is a candidate GRPO for (i). This amounts to require that β, γ, and δ are such that their pasting composite yields ρ, and that each of them is a structure-preserving bijection.

Let β represent the following composite:

$$|a| \oplus |\widetilde{e}| \xrightarrow{[\rho i_1, t_e^{-1}]} |\rho i_1 \cdot a| \cup |e| = |i_1 \cdot l| \cup |f| \xrightarrow{\left[i_1 i_1^{-1}, i_2 t_f\right]} |l| \oplus |\widetilde{f}|$$

and similarly let γ and δ be, respectively,

$$|c| \xrightarrow{\rho i_2} |\rho i_2 \cdot c| = |g \circ e| = |e| \cup |g| \xrightarrow{[i_1 t_e, i_2 t_g]} |\widetilde{e}| \oplus |\widetilde{g}|$$

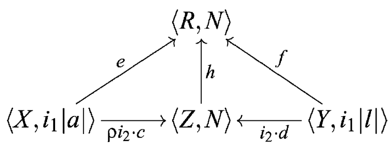
and

$$|\widetilde{f}| \oplus |\widetilde{g}| \xrightarrow{\left[t_f^{-1}, t_g^{-1}\right]} |f| \cup |g| = |g \circ f| = |i_2 \circ d| \xrightarrow{i_2^{-1}} |d|.$$

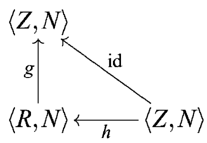
It is easy to check that the pasting of γ, β and δ as in the GRPO diagram yields ρ. We show that γ is a structure-preserving bijection The argument for the other morphisms is similarly trivial. Since $\rho i_2 \cdot c = g \circ e$ we have $[i_1 t_e, i_2 t_g] \rho i_2 \cdot c = [i_1 t_e, i_2 t_g] \cdot (g \circ e)$ and so $\gamma \cdot c = \widetilde{g} \circ \widetilde{e}$.

Indeed, $\langle R, \widetilde{e}, \widetilde{f}, \widetilde{g}, \beta, \gamma, \delta \rangle$ is a candidate GRPO for (i). Thus there exists $h: Z \to R$ and 2-cells (structure-preserving support bijections) $\varphi: \widetilde{e} \Rightarrow hc$, $\psi: hd \Rightarrow \widetilde{f}$ and $\tau: \widetilde{g}h \Rightarrow \mathrm{id}_Z$.

From the existence of τ and the definition of well-supported category, we can deduce that $|\widetilde{g}| = |g| = \emptyset$ and $|h| = \emptyset$. Note that $\tau = \mathrm{id}$, since there is only one endofunction on $\emptyset$. We can therefore conclude that also $M = N$ and $\widetilde{g} = g$.



(i)

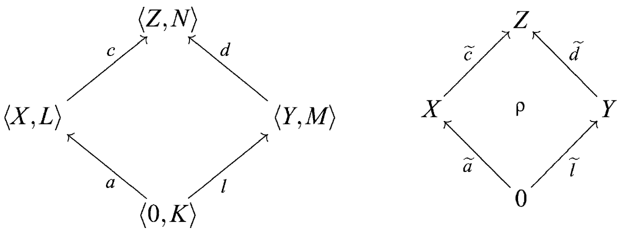


(ii)

We also get immediately that (ii) above commutes. We show that the left triangle of (i) commutes, the proof for the right one is similar. From the definition of GRPO, we have that $\mathrm{id}_c = \tau c \bullet \widetilde{g}\varphi \bullet \gamma = g\varphi \bullet \gamma$ which then implies that $\varphi = \gamma^{-1}$. Using the definition of γ, $\rho i_2 \bullet \varphi \bullet t_e = \mathrm{id}$ which amounts to saying that the triangle is commutative.

Uniqueness in $\widehat{\mathbb{C}}$ easily follows from essential uniqueness in $\mathbb{B}$ (which is in this case the same as uniqueness, since there is only one endofunction on the $\emptyset$).

Going the other way, suppose that (iii) below



(iii) (iv)

is a RPO. Then (iv) is a GRPO where ρ is

$$|\widetilde{a}| \oplus |\widetilde{c}| \xrightarrow{[t_a^{-1}, t_c^{-1}]} |a| \cup |c| = |l| \cup |d| \xrightarrow{[i_1 t_l, i_2 t_d]} |\widetilde{l}| \oplus |\widetilde{d}|.$$

It is trivial to show that ρ is structure-preserving, i.e. $\rho \cdot (\widetilde{c}\,\widetilde{a}) = \widetilde{d}\,\widetilde{l}$. Now consider a candidate $\langle R, e, f, g, \beta, \gamma, \delta \rangle$ for (ii), above. Since the pasting composite of γ, β and δ yields ρ, we have that $t_c^{-1}\gamma^{-1}i_2 \cdot g = t_d^{-1}\delta i_2 \cdot g = g'$. Let $V = N \backslash |g'|$. Let $e' = t_c^{-1}\gamma^{-1}i_1 \cdot e$ and $f' = t_d^{-1}\delta i_1 \cdot f$. It is easy but tedious to check that $\langle \langle R, V \rangle, e', f', g' \rangle$ is a candidate for (i). By assumption, there exists an arrow $h \colon \langle Z, N \rangle \to \langle R, V \rangle$ which satisfies $hc = e'$, $hd = f'$ and $g'h = f'$. This can be translated in the by-now standard way into a mediating morphism $\langle h, \varphi, \psi, \tau \rangle$ where τ is again the unique endofunction on the Ø. Uniqueness again follows by laborious, yet not challenging, work. □

Example 34 (*Bunches*). The 2-category of supports of the precategory **A-Bun** is **Bun**. Note that a 'structure preserving' support bijection is exactly a bunch homomorphism. Indeed, $\rho \colon (X, \mathsf{char}, \mathsf{rt}) \Rightarrow (X', \mathsf{char}', \mathsf{rt}')$ if $X' = \rho X$, $\mathsf{char}' = \mathsf{char}\,\rho^{-1}$ and $\mathsf{rt}' = \mathsf{rt}(\mathrm{id} \oplus \rho^{-1})$ which is the same as saying $\mathsf{char} = \mathsf{char}'\,\rho$ and $\mathsf{rt} = \mathsf{rt}'(\mathrm{id} \oplus \rho)$.

In other words, our general construction translating from well-supported precategories to G-categories applied to the particular case of 'bunches and wirings,' extracts **Bun** out of **A-Bun**. This confirms the results obtained by Leifer and Milner on this specific subject, and supports our claim of appropriateness of the structures we have introduced. It is worth remarking how in Definition 32 precategories' support-translation isomorphisms are subsumed in G-categories as 2-cells. Further study is of course necessary to verify the usefulness of GRPOs in the presence of more complex terms. The results we obtained recently in the case of graph rewriting and bigraphs are indeed encouraging [22].

6. Conclusion

We have extended our theory of GRPOs initiated in previous work in order to strengthen existing techniques for deriving operational congruences for reduction systems in the presence of non-trivial structural congruences. In particular, this paper has shown that previous theories can be recast using G-reactive systems and GRPOs at no substantial additional complexity. Also, we proved that the theory is powerful enough to encompass several examples considered in the literature, as a precise consequence of the fact that any precategory or functorial reactive system yields a corresponding G-category in a direct, systematic way. Therefore, we believe that it constitutes a natural starting point for future investigations towards a fully comprehensive theory, which we started to explore further in [22].

It follows from Theorem 33 that G-categories are at least as expressive as well-supported precategories. A natural consideration is whether a reverse translation may exist. We believe that this is not the case, as general G-categories appear to carry more information than precategories. This may turn out to have an impact in dealing with complex structural congruences, such the one arising from the replication axiom $P \equiv P \mid !P$.

Acknowledgements

The authors are indebted to the referees, both of the extended abstract [20], and of this article, for their helpful comments and suggestions. Thanks also go to Robin Milner for his comments.

References

[1] M. Bugliesi, S. Crafa, M. Merro, V. Sassone, Communication interference in mobile boxed ambients, in: Foundations of Software Technology and Theoretical Computer Science, FST&TCS‘02, Lecture Notes in Computer Science, Vol. 2556, Springer, Berlin, 2002, pp. 71–84.
[2] A. Carboni, S. Lack, R.F.C. Walters, Introduction to extensive and distributive categories, J. Pure Appl. Algebra 84 (2) (1993) 145–158.
[3] G. Castagna, F. Zappa Nardelli, The Seal calculus revisited, in: Foundations of Software Technology and Theoretical Computer Science, FST&TCS‘02, Lecture Notes in Computer Science, Vol. 2556, Springer, Berlin, 2002, pp. 85–96.
[4] J.C. Godskesen, T. Hildebrandt, V. Sassone, A calculus of mobile resources, in: Internat. Conf. on Concurrency Theory, CONCUR‘02, Lecture Notes in Computer Science, Vol. 2421, Springer, Berlin, 2002, pp. 272–287.
[5] M. Hennessy, M. Merro, Bisimulation congruences in safe ambients, in: Principles of Programming Languages, POPL‘02, ACM Press, New York, 2002, pp. 71–80.
[6] C.A.R. Hoare, Communicating Sequential Processes, Prentice-Hall, Englewood Cliffs, NJ, 1985.
[7] O.H. Jensen, R. Milner, Bigraphs and mobile processes, Technical Report 570, Computer Laboratory, University of Cambridge, 2003.
[8] O.H. Jensen, R. Milner, Bigraphs and transitions, in: Principles of Programming Languages, POPL‘03, ACM Press, New York, 2003.
[9] G.M. Kelly, Elementary observations on 2-categorical limits, Bull. Austral. Math. Soc. 39 (1989) 301–317.
[10] G.M. Kelly, R.H. Street, Review of the elements of 2-categories, Lecture Notes Math. 420 (1974) 75–103.
[11] F.W. Lawvere, Functorial semantics of algebraic theories, Proc. Natl. Acad. Sci. 50 (1963) 869–873.
[12] J. Leifer, Operational congruences for reactive systems, Ph.D. Thesis, University of Cambridge, 2001.
[13] J. Leifer, Synthesising labelled transitions and operational congruences in reactive systems, parts 1 and 2, Technical Report RR-4394 and RR-4395, INRIA Rocquencourt, 2002.
[14] J. Leifer, R. Milner, Deriving bisimulation congruences for reactive systems, in: Internat. Conf. on Concurrency Theory, CONCUR‘00, Lecture Notes in Computer Science, Springer, Berlin, 2000, pp. 243–258.
[15] R. Milner, Calculi for interaction, Acta Inform. 33 (8) (1996) 707–737.
[16] R. Milner, Communicating and Mobile Systems: the Pi-calculus, Cambridge University Press, Cambridge, 1999.
[17] R. Milner, Bigraphical reactive systems: basic theory, Technical Report 523, Computer Laboratory, University of Cambridge, 2001.

[18] A.W. Roscoe, The Theory and Practice of Concurrency, Prentice-Hall, Englewood Cliffs, NJ, 1997.
[19] V. Sassone, P. Sobociński, Deriving bisimulation congruences: a 2-categorical approach, Electron. Notes Theoret. Comput. Sci. 68 (2) (2002).
[20] V. Sassone, P. Sobociński, Deriving bisimulation congruences: 2-categories vs. precategories, in: Proc. Found. of Software Science and Computation Structures, FOSSACS'03, Lecture Notes in Computer Science, Vol. 2620, Springer, Berlin, 2003, pp. 409–424.
[21] V. Sassone, P. Sobociński, Deriving bisimulation congruences using 2-categories, Nordic J. Comput. 10 (2003) 163–183.
[22] V. Sassone, P. Sobociński, Coinductive reasoning for contextual graph rewriting, Manuscript, 2004.
[23] P. Sewell, Working note PS12, February 2000, Unpublished note.
[24] P. Sewell, From rewrite rules to bisimulation congruences, Theoret. Comput. Sci. 274 (1–2) (2002) 183–230.
[25] R.H. Street, Fibrations in bicategories, Cahiers Topologie Géom. Différentielle XXI-2 (1980) 111–159.