Undecidability of the Free Adjoint Construction

R. J. MacG. Dawson *
Saint Mary's University
email: rdawson@stmarys.ca

R. Paré
Dalhousie University
email: pare@mscs.dal.ca

D.A. Pronk
Dalhousie University
email: pronk@mscs.dal.ca

August 1, 2002

Abstract

In this paper we discuss some aspects of categories obtained by freely adding right adjoints to all arrows in a category. We will give a description of the arrows and 2-cells in such a category and show how the equivalence relation on the 2-cells for an appropriately chosen category $\mathcal{C}_{\mathbb{A}}$ can be used to simulate a 2-register abacus $\mathbb{A}$, so that deciding whether two 2-cells with different representatives are equal becomes equivalent to solving the halting problem for the abacus. In particular, this implies that (in general) equality of 2-cells in such categories is undecidable.

Keywords: free adjoint construction, 2-category, undecidability, abacus

AMS classification: 18A40, 18D05

*All three authors are suppported by NSERC grants.

1 Introduction

In the 2-category **Cat**, an arrow (that is, a functor) $f\colon A \to B$ is said to have a right adjoint $u\colon B \to A$ if there exist 2-cells $\varepsilon\colon f \circ u \Rightarrow i_B$ and $\eta\colon i_A \Rightarrow u \circ f$, such that the composite $f \stackrel{f\circ\eta}{\Rightarrow} f \circ (u \circ f) = (f \circ u) \circ f \stackrel{\varepsilon\circ f}{\Rightarrow} f$ is equal to i_f, and the composite $u \stackrel{\eta\circ u}{\Rightarrow} (u \circ f) \circ u = u \circ (f \circ u) \stackrel{u\circ\varepsilon}{\Rightarrow} u$ is equal to i_u. This definition can obviously be extended to any 2-category; in most cases, as in **Cat** itself, not all arrows have adjoints.

There will always be some adjunctions, however. Identity arrows are self-adjoint, and more generally any inverse pair f, f^{-1} constitutes an adjunction with identity 2-cells as unit and counit. Another special class of adjunctions, more general than inverse pairs, are the *equivalences*, pairs for which the unit and counit are isomorphisms.

If a 2-category does not have a certain type of adjunction, there may be a construction that adds these. Some examples are:

- Any semigroup may be turned into a group in a canonical way by adding inverses. The constructions of the integers from the natural numbers, and of the nonzero rationals from the nonzero integers, are familiar examples.
- Gabriel and Zisman's *category of fractions* [6] adds inverses to a class of arrows satisfying certain conditions. (This construction is not explicitly 2-categorical, because the units and counits are identities; however, as shown in [9], it is useful to consider it in this context.)
- Quillen's homotopy categories [10] add inverses to the weak equivalences of a model category.
- Pronk's *bicategory of fractions* [9] add equivalences to a suitable class of arrows, thereby transforming the original arrows into equivalences as well.
- In [11] Schanuel and Street add a right adjoint to the single nonidentity arrow of $\mathbf{2}$ The structure of the resulting 2-category is surprisingly rich; it contains both Δ and Δ^{op} as hom-categories.
- In [3], the authors develop a construction $\Pi_2(\mathcal{C})$ which adds general adjoints to all arrows in any category.
- In [5], it is shown that if the original category has pullbacks, and adjoints are added subject to the Beck-Chevalley condition, the 2-category $\mathrm{Span}(\mathcal{C})$ is obtained.

In all of the above examples, there are equivalence relations on the resulting structures; the basic principle is illustrated by the equation $2 - 4 = 3 - 5$ in the first example above. In most of these cases, it is fairly straightforward (if we know the structure of the original category) to determine whether two compositions of arrows or 2-cells are equal. In this paper, however, we show that the equality of 2-cells in $\Pi_2(\mathcal{C})$ is undecidable for some choices of $\mathcal{C}$, even when $\mathcal{C}$ is presented in such a fashion that all compositions can be evaluated.

It should be noted that in general there is more than one way to add adjoints. For instance consider the category $\mathbb{2} = (0 \xrightarrow{f} 1)$. To give f a right adjoint, we must add an arrow $u\colon 1 \to 0$, and 2-cells ε and η. However, the requirement that $f \dashv u$ does not fully define the new bicategory. For instance, it does not tell us whether $ufuf = uf$. In the construction of Schanuel and Street, this identity - and any other not required by the adjunction - does not hold, making **Adj** initial among all 2-categories extending $\mathbb{2}$ in which f has an adjoint, or, equivalently, the *free* construction of this type.

In [3] we extend this construction, adding right adjoints to an arbitrary category $\mathcal{C}$ to obtain a 2-category $\Pi_2(\mathcal{C})$. The adjoint arrows correspond to arrows of $\mathcal{C}^{op}$; thus, a general arrow of $\Pi_2(\mathcal{C})$ corresponds to a zigzag of arrows of $\mathcal{C}$. The 2-cells of $\Pi_2(\mathcal{C})$ are generated by the identity cells on the arrows of $\mathcal{C}$ and their adjoints, and by the units and counits. Every composition of these 2-cells corresponds to a planar diagram of a special type, called a *fence* (these are described below); vertical and horizontal composition of fences are defined. These compositions do not satisfy the middle-four interchange law; to obtain a 2-category we must identify all "vertical-first" compositions of fences $(\alpha*\beta)\circ(\gamma*\delta)$ with the corresponding "horizontal-first" compositions $(\alpha \circ \gamma) * (\beta \circ \delta)$. The resulting 2-category is $\Pi_2(\mathcal{C})$.

The equivalence relation mentioned above is shown in [3] to be generated by a class of local substitutions based upon compositions and factorizations within $\mathcal{C}$. Fences that can be obtained from each other by a single such operation will be called *directly equivalent*. Both the definition of a fence, and the equivalence relation, will be given in detail in Section 2 below; for proofs, the reader is referred to the paper cited.

In this paper, we show that the equivalence relation on fences that yields $\Pi_2(\mathcal{C})$ is undecidable for certain choices of $\mathcal{C}$. To do this, we follow the usual strategy of modelling a known undecidable problem within the one we wish to prove undecidable. Common choices for the known undecidable problem are the *word problem for groups* and the *halting problem for Turing machines.* Both of these involve structures (strings of generators in one case, the "tape" of the Turing machine in the other) that can grow without bound. As will be shown below, any two equivalent fences in $\Pi_2(\mathcal{C})$ must have the same number of arrows. We will therefore use a different (though related) undecidable problem, the *halting problem for abacuses.*

Abacuses, introduced by Lambek [7] and (under the name "register machines") by Minsky [8], are universal models for computation similar to Turing machines. Like Turing machines, they have a finite number of internal program states, a halting state, and change state in a deterministic manner, branching on some instructions based upon the contents of a specified memory location. However, where the Turing machine has infinitely many registers, each able to take a finite number of values, the abacus has finitely many registers, each able to hold an arbitrarily large natural number. It was shown by Minsky *(op. cit.)*

that any abacus, and any Turing machine, may be simulated by an appropriate 2-register abacus; in particular, there exist universal 2-register abacuses, for which the halting problem is undecidable.

Undecidability occurs in several places in higher-dimensional category theory. For instance, in [1] it was shown that the word problem for a free double category could be undecidable. In [4] the authors show that various free extensions of double categories and 2-categories involve an undecidable word problem for composable arrangements of 2-cells.

2 Freely Adding Adjoints

In [3] we have studied the structure of a 2-category $\Pi_2(\mathcal{C})$ obtained by freely adding adjoints to all arrows in a category $\mathcal{C}$. We start by summarizing the results of [3] for this special case.

2.1 The 2-category $\Pi_2(\mathcal{C})$.

Let $\mathcal{C}$ be any category. We write $\Pi_2(\mathcal{C})$ for the following 2-category:

1. The **objects** of $\Pi_2(\mathcal{C})$ are the same as the objects of $\mathcal{C}$, so $\Pi_2(\mathcal{C})_0 = \mathcal{C}_0$.

2. The **arrows** in $\Pi_2(\mathcal{C})$ are zig-zags in $\mathcal{C}$, i.e. an arrow in $\mathcal{C}(A, B)$ is of the form

$$A = C_1 \xrightarrow{f_1} D_1 \xleftarrow{g_1} C_2 \xrightarrow{f_2} D_2 \longleftarrow \cdots \longrightarrow D_{n-1} \xleftarrow{g_{n-1}} C_n \xrightarrow{f_n} D_n = B.$$

   We denote such an arrow by

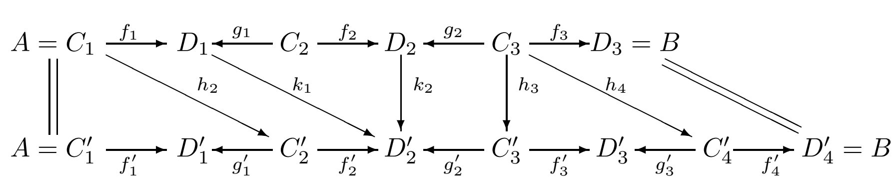
$$(g_1, \cdots, g_{n-1}; f_1, \cdots, f_n).$$

3. A **2-cell**

$$(g_1, \cdots, g_{n-1}; f_1, \cdots, f_n) \Rightarrow (g'_1, \cdots, g'_{m-1}; f'_1, \cdots, f'_m)$$

   is an equivalence class of a special kind of diagrams, called *fences.*

An example of a fence is



Specifically, fences are determined by an adjoint pair of order-preserving index functions

$$\phi\colon \{1,\cdots,m\} \to \{1,\cdots,n\} \text{ and } \psi\colon \{1,\cdots,n\} \to \{1,\cdots,m\}$$

such that $\psi\phi(i) \geq i$ and $\phi\psi(j) \leq j$, together with families of arrows

$$k_j\colon D_j \to D'_{\psi(j)} \text{ and } h_i\colon C_{\phi(i)} \to C'_i,$$

such that all resulting squares commute, *i.e.*

1. For every $1 \leq i < n$

$$f'_i h_i = \begin{cases} g'_i h_{i+1} & \text{if } \varphi(i) = \varphi(i+1) \\ k_{\varphi(i)+1} f_{\varphi(i)} & \text{otherwise} \end{cases}$$

2. For every $1 \leq j < m$

$$k_j g_j = \begin{cases} k_{j+1} f_{j+1} & \text{if} \psi(j) = \psi(j+1) \\ g'_{\psi(j)} h_{\psi(j)+1} & \text{otherwise} \end{cases}$$

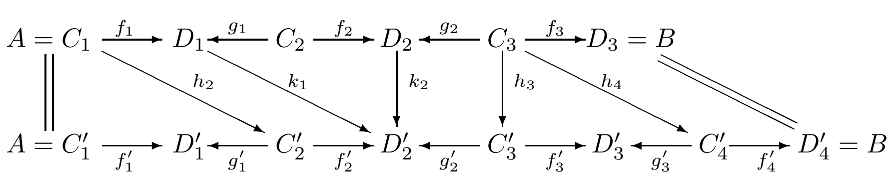
Finally we require that $h_1 = \mathrm{id}_A$ and $k_n = \mathrm{id}_B$. We denote such a representative for a 2-cell by $(\phi;\psi;k_1,\ldots,k_{n-1};h_2,\ldots,h_m)$. Note that in the example above, $\psi(1) = 2$, $\psi(2) = 2$ and $\psi(3) = 4$, and $\phi(1) = 1$, $\phi(2) = 1$, $\phi(3) = 3$, and $\phi(4) = 3$.

2.2 Composition

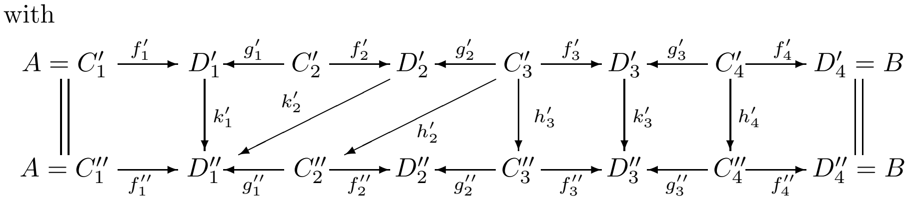
Composition of arrows in $\Pi_2(\mathcal{C})$ is defined by concatenating of the zig-zags and composing the two composable morphisms:

$$(g_1,\cdots,g_{n-1},f_1,\cdots,f_n) \circ (h_1,\cdots,h_{m-1},k_1,\cdots,k_n) = \\ (h_1,\cdots,h_{m-1},g_1,\cdots,g_{n-1},k_1,\cdots,k_{n-1},f_1 \circ k_n, f_2,\cdots,f_n)$$

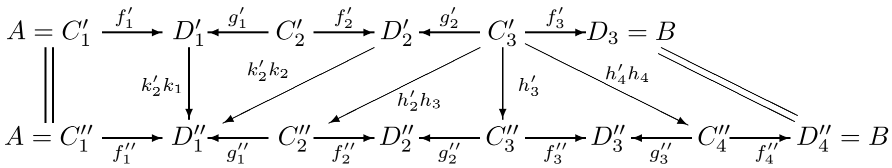
Vertical composition of fences can be calculated by drawing the diagrams above one another and constructing the compositions of the arrows when that is possible. For example, composing



with



gives

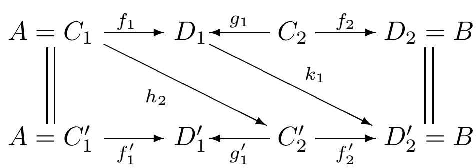


To be precise, the index functions of the new 2-cell are the compositions of the index functions of the original 2-cells, and the arrows in the diagrams are the compositions of the corresponding arrows:

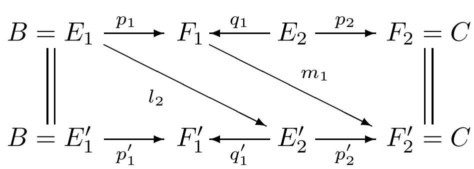
$$\begin{aligned} &[\phi', \psi', (k_j'), (h_i')] \cdot [\phi, \psi, (k_j), (h_i)] = \\ = \quad &[\phi \circ \phi', \psi' \circ \psi, (\overline{k}_j), (\overline{h}_i)], \end{aligned}$$

where $\overline{k}_j = k_{\psi(j)} k_j$ and $\overline{h}_i = h_i h'_{\phi'(i)}$.

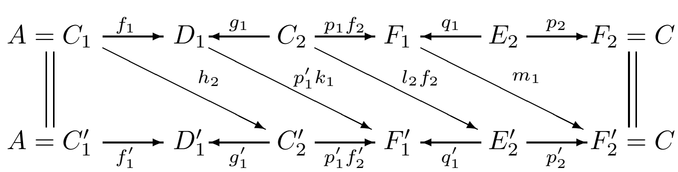
Horizontal composition of fences is defined by concatenation, together with some compositions in the category $\mathcal{C}$. This is a tensor-like construction over $\mathcal{C}$. For example, horizontal composition of



with



gives



To be precise, if

$$\begin{aligned} &[\overline{\phi}; \overline{\psi}; \overline{k}_1, \ldots, \overline{k}_{n-1}; \overline{h}_2, \ldots, \overline{h}_m] \colon \\ &\qquad (g_1, \ldots, g_{n-1}; f_1, \ldots, f_n) \Rightarrow (g_1', \ldots, g_{m-1}'; f_1', \ldots, f_m') \end{aligned}$$

and

$$\begin{aligned} &[\tilde{\phi}; \tilde{\psi}; \tilde{k}_1, \ldots, \tilde{k}_{p-1}; \tilde{h}_2, \ldots, \tilde{h}_q] \colon \\ &\qquad (u_1, \ldots, u_{p-1}; v_1, \ldots, v_p) \Rightarrow (u_1', \ldots, u_{q-1}'; v_1', \ldots, v_q') \end{aligned}$$

are 2-cells such that $\mathrm{cod}(f_n) = \mathrm{dom}(v_1)$ (*i.e.*, that are composable), then the horizontal composition is defined as

$$[\tilde{\phi};\tilde{\psi};\tilde{k}_1,\dots,\tilde{k}_{p-1};\tilde{h}_2,\dots,\tilde{h}_q]\circ[\overline{\phi};\overline{\psi};\overline{k}_1,\dots,\overline{k}_{n-1};\overline{h}_2,\dots,\overline{h}_m] = [\phi;\psi;k_1,\dots,k_{n-1};h_2,\dots,h_m],$$

where the index functions

$$\psi\colon \{1\dots,n+p-1\}\to\{1,\dots,m+q-1\}$$
$$\phi\colon \{1,\dots,m+q-1\}\to\{1,\dots,n+p-1\}$$

are

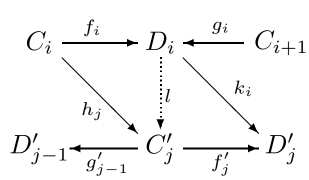
$$\phi(i) = \begin{cases} \overline{\phi}(i) & \text{for} \quad i\le m \\ \tilde{\phi}(i-m+1)+n-1 & \text{for} \quad i>m \end{cases}$$
$$\psi(j) = \begin{cases} \overline{\psi}(j) & \text{for} \quad j\le n-1 \\ \tilde{\psi}(j-n+1)+m-1 & \text{for} \quad j\ge n \end{cases}$$

and

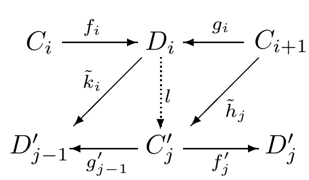
$$k_j = \begin{cases} \overline{k}_j & \text{if} \quad j<n \text{ and } \overline{\psi}(j)<m \\ v_1'\overline{k}_j & \text{if} \quad j<n \text{ and } \overline{\psi}(j)=m \\ \tilde{k}_{j-n+1} & \text{if} \quad j\ge n \end{cases}$$
$$h_i = \begin{cases} \overline{h}_i & \text{if} \quad i\le m \\ \tilde{h}_{i-m+1}f_n & \text{if} \quad i>m \text{ and } \tilde{\phi}(i-m+1)=1 \\ \tilde{h}_{i-m+1} & \text{if} \quad i>m \text{ and } \tilde{\phi}(i-m+1)>1 \end{cases}$$

2.3 The equivalence relation

The pair of composition operations given above does not have the middle-four exchange property ([3], Example 1). If we identify every composition $(\alpha\circ\beta)*(\gamma\circ\delta)$ with the corresponding composition $(\alpha*\gamma)\circ(\beta*\delta)$, we obtain (*op. cit.*, section 3) the equivalence relation generated (through symmetry and transitivity) by the following relationship: Two diagrams θ and ω representing 2-cells in $\mathcal{C}^*$ are called *directly equivalent* if they are everywhere the same except for two parts of the following shapes: the cell ω contains



and the cell θ contains



and there exists an arrow $l\colon D_i \to C'_j$ which factors both diagrams.

Both vertical and horizontal composition are well-defined on equivalence classes of 2-cells ([3], section 2), and together they have the middle-four property. The resulting 2-category is given the name $\Pi_2\mathcal{C}$. The reader is referred to section 4 of [3] for a description of the universal properties of this construction.

3 From abacus to category

In this section we will construct a category, presented in such a way that any composition may be computed trivially, but such that the equivalence relation on fences models an undecidable problem in the theory of computation. There are various undecidable problems that could be used; as the length of fences does not vary within an equivalence class, the abacus (which has a fixed number of registers) is a more convenient model than the Turing machine (the tape of which must be capable of indefinite extension) or the word problem for groups (which again may involve intermediate words of arbitrarily great length). As Minsky has shown that a universal abacus may have as few as two registers, we shall restrict our attention to such a machine.

We shall construct, for any 2-register abacus $\mathbb{A}$, a category $\mathcal{C}_{\mathbb{A}}$, containing three sets of arrows which are indexed by the program state, X register, and Y register of $\mathbb{A}$ respectively. We then exhibit a family of fences over $\mathcal{C}_{\mathbb{A}}$, each containing one arrow from each of these three sets, and therefore indexed by the complete internal state of the abacus. Finally, we show that two such fences are directly equivalent if and only if the abacus passes directly from the state corresponding to one fence to that corresponding to the other. It follows that, if $\mathbb{A}$ is a universal 2-register abacus, any algorithm to determine the equivalence of fences over $\mathcal{C}_{\mathbb{A}}$ would also solve the halting problem for $\mathbb{A}$; the equivalence relation is therefore undecidable.

3.1 Definition of an abacus

Definition 1 *An (n-register)* abacus *consists of*
(i) A finite set of states S*;*
(ii) Variables $X_1, \cdots, X_n$ *in* $\mathbb{N}$ *which are considered as the contents of the registers;*
(iii) A function $\mathbf{Inst}\colon S \to \{\mathtt{INCX}, \mathtt{INCY}, \mathtt{DECX}, \mathtt{DECY}, \mathtt{HALT}\}$*;*
(iv) A starting state $s_0 \in S$*;*
(v) Transition functions $\sigma\colon S \to S$ *and* $\sigma'\colon S \to S$*.*

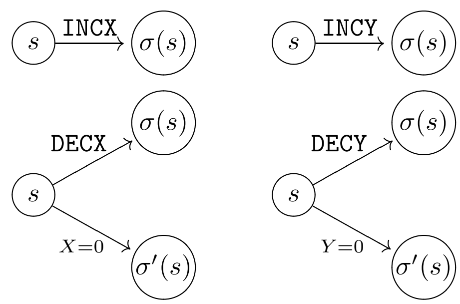
In this paper we will only be interested in the notion of a 2-register abacus, so from now on we will take $n = 2$ in the definition above, and use X and Y as variables for the registers. The behaviour of the abacus is a partial function $S \times \mathbb{N} \times \mathbb{N} \to S \times \mathbb{N} \times \mathbb{N}$

$$\mathfrak{S} = (s, X, Y) \overset{\Sigma}{\mapsto} \mathfrak{S}' = (s', X', Y')$$

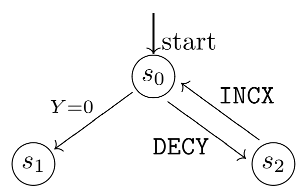
defined as follows

$$(s', X', Y') = \begin{cases} (\sigma(s), X+1, Y) & \text{if } \mathbf{Inst}(s) = \texttt{INCX} \\ (\sigma(s), X, Y+1) & \text{if } \mathbf{Inst}(s) = \texttt{INCY} \\ (\sigma(s), X-1, Y) & \text{if } \mathbf{Inst}(s) = \texttt{DECX} \text{ and } X > 0 \\ (\sigma'(s), X, Y) & \text{if } \mathbf{Inst}(s) = \texttt{DECX} \text{ and } X = 0 \\ (\sigma(s), X, Y-1) & \text{if } \mathbf{Inst}(s) = \texttt{DECY} \text{ and } Y > 0 \\ (\sigma'(s), X, Y) & \text{if } \mathbf{Inst}(s) = \texttt{DECY} \text{ and } Y = 0 \\ \text{undefined} & \text{if } \mathbf{Inst}(s) = \texttt{HALT}. \end{cases}$$

We can represent this by a graph whose nodes are the elements of S and whose edges are of the form



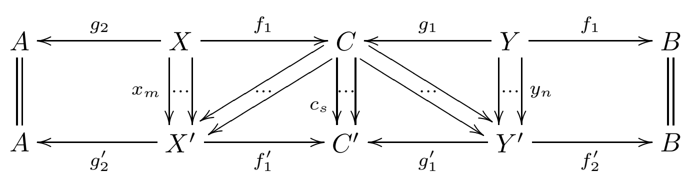
So the σ and σ' permit branching at nodes s with $\mathbf{Inst}(s) \in \{\texttt{DECX}, \texttt{DECY}\}$. Here is a simple example which adds the contents of the X-register and the Y-register and puts the sum in the X-register and 0 in the Y-register.



If one starts with $X = 4$ and $Y = 3$, this abacus would go through the following states: $(s_0, 4, 3) \mapsto (s_2, 4, 2) \mapsto (s_0, 5, 2) \mapsto (s_2, 5, 1) \mapsto (s_0, 6, 1) \mapsto (s_2, 6, 0) \mapsto (s_0, 7, 0) \mapsto (s_1, 7, 0)$ at which point it halts.

3.2 A graph representation for $\mathcal{C}_{\mathbb{A}}$

Let A be a 2-register abacus. The category $\mathcal{C}_{\mathbb{A}}$ is generated by a graph of the following form:



(1)

So the set of objects of the category $\mathcal{C}_{\mathbb{A}}$ is

$$C_0 = \{A, X, X', C, C', Y, Y', B\},$$

and the horizontal arrows shown in (1) are the only arrows in their hom-sets. The diagonal and vertical arrows in (1) represent hom-sets containing more than one element. These hom-sets are symmetric in the sense that the right side of the diagram is a mirror image of the left side.

The hom-sets that are indicated by vertical arrows in the diagram (1) are:

$$\begin{array}{rcl} \mathrm{Hom}(C, C') & = & \{c_s | s \in S\} \\ \mathrm{Hom}(X, X') & = & \{x_m | m \in \mathbb{N}\} \\ \mathrm{Hom}(Y, Y') & = & \{y_n | n \in \mathbb{N}\} \end{array}$$

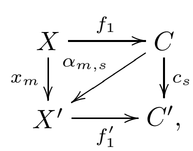
These satisfy the following relations:

$$\begin{array}{rcl} g_2' x_m & = & g_2 \text{ for all } m \in \mathbb{N}, \\ c_s f_1 & = & f_1' x_m \text{ for all } s \in S \text{ and } m \in \mathbb{N}, \\ f_2' y_n & = & f_2 \text{ for all } n \in \mathbb{N}. \end{array}$$

The hom-sets indicated by the diagonal arrows in the diagram (1) are

$$\begin{array}{rcl} \mathrm{Hom}(C, X') & = & \{\alpha_{m,s}, \beta_{m,s} | m \in \mathbb{N},\ s \in S \text{ with } \mathbf{Inst}(s) \in \{\mathtt{INCX}, \mathtt{DECX}\}\} \\ \mathrm{Hom}(C, Y') & = & \{\gamma_{n,s}, \delta_{n,s} | n \in \mathbb{N},\ s \in S \text{ with } \mathbf{Inst}(s) \in \{\mathtt{INCY}, \mathtt{DECY}\}\}. \end{array}$$

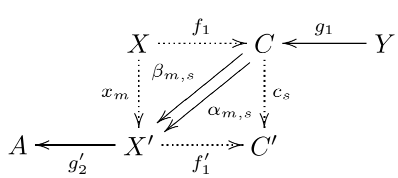
The arrow $\alpha_{m,s}$ makes the following diagram commute



i.e.,

$$x_m = \alpha_{m,s} \circ f_1 \text{ and } f_1' \circ \alpha_{m,s} = c_s. \tag{2}$$

We also require that in the diagram



we have

$$\alpha_{m,s} \circ g_1 = \beta_{m,s} \circ g_1 \text{ and } g_2' \circ \alpha_{m,s} = g_2' \circ \beta_{m,s}. \tag{3}$$

The $\beta_{m,s}$ need to satisfy the following equations:

$$\beta_{m,s} \circ f_1 = \begin{cases} x_{m+1} & \text{if } \mathbf{Inst}(s) = \texttt{INCX} \\ x_{m-1} & \text{if } \mathbf{Inst}(s) = \texttt{DECX} \text{ and } m > 0 \\ x_0 & \text{if } \mathbf{Inst}(s) = \texttt{DECX} \text{ and } m = 0 \end{cases} \tag{4}$$

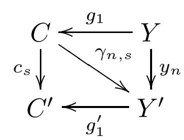
and

$$f_1' \circ \beta_{m,s} = \begin{cases} c_{\sigma(s)} & \text{if } \mathbf{Inst}(s) = \texttt{INCX} \\ c_{\sigma(s)} & \text{if } \mathbf{Inst}(s) = \texttt{DECX} \text{ and } m > 0 \\ c_{\sigma'(s)} & \text{if } \mathbf{Inst}(s) = \texttt{DECX} \text{ and } m = 0 \end{cases} \tag{5}$$

We also write

$$\kappa(x_m, c_s) := \beta_{m,s} \circ f_1 \text{ and } \lambda(x_m, c_s) := f_1' \circ \beta_{m,s}. \tag{6}$$

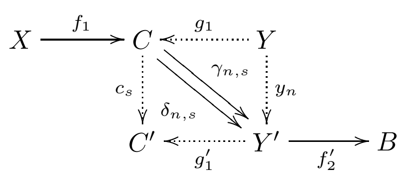
Dually, the arrow $\gamma_{n,s}$ makes the following diagram commute



i.e.,

$$y_n = \gamma_{n,s} \circ g_1 \text{ and } g_1' \circ \gamma_{m,s} = c_s. \tag{7}$$

We also require that in the diagram



we have that

$$\gamma_{n,s} \circ f_1 = \delta_{n,s} \circ f_1 \text{ and } f_2' \circ \gamma_{n,s} = f_2' \circ \delta_{n,s}. \tag{8}$$

The $\delta_{m,s}$ need to satisfy the following equations:

$$\delta_{n,s} \circ g_1 = \begin{cases} y_{n+1} & \text{if } \mathbf{Inst}(s) = \texttt{INCY} \\ y_{n-1} & \text{if } \mathbf{Inst}(s) = \texttt{DECY} \text{ and } n > 0 \\ y_0 & \text{if } \mathbf{Inst}(s) = \texttt{DECY} \text{ and } n = 0 \end{cases} \tag{9}$$

and

$$g_1' \circ \delta_{n,s} = \begin{cases} c_{\sigma(s)} & \text{if } \mathbf{Inst}(s) = \texttt{INCY} \\ c_{\sigma(s)} & \text{if } \mathbf{Inst}(s) = \texttt{DECY} \text{ and } n > 0 \\ c_{\sigma'(s)} & \text{if } \mathbf{Inst}(s) = \texttt{DECY} \text{ and } n = 0 \end{cases} \tag{10}$$

We also write

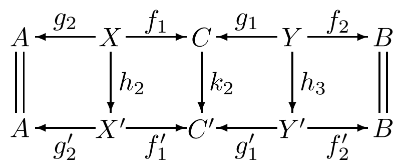
$$\kappa(c_s, y_n) := g_1' \circ \delta_{n,s} \text{ and } \lambda(c_s, y_n) := \delta_{n,s} \circ g_1. \tag{11}$$

3.3 The equivalence relation on $\Pi_2(\mathcal{C}_{\mathbb{A}})_2$

In this section we will discuss the equivalence relation for representatives of 2-cells in $\Pi_2(\mathcal{C}_{\mathbb{A}})_2$ of the form

$$(\phi, \psi, (k_j), (h_i)) \colon (g_2, g_1; I_A, f_1, f_2) \Rightarrow (g_2', g_1'; I_A, f_1', f_2')$$

with $\phi = \psi = \mathrm{id}$ and $k_1 = \mathrm{id}_A$. Since this implies that the leftmost square of the diagram for this 2-cell only consists of identities, we will not draw it:



Note: For the rest of this paper we will denote such a representative by an ordered list

$$(h_2, k_2, h_3)$$

of its vertical arrows, and we will call such a diagram *rectangular*.

If for any consecutive pair of vertical arrows, i.e. h_2, k_2 or k_2, h_3, the functions γ, δ, κ, and λ from the previous subsection are defined, the equations (2) to (3), (6) to (8), and (11) imply that

$$(h_2, k_2, h_3) \sim (\kappa(h_2, k_2), \lambda(h_2, k_2), h_3)$$

and

$$(h_2, k_2, h_3) \sim (h_2, \kappa(k_2, h_3), \lambda(k_2, h_3)).$$

For this particular instance of the equivalence relation, we say that $(\kappa(h_2, k_2), \lambda(h_2, k_2), h_3)$ and $(h_2, \kappa(k_2, h_3), \lambda(k_2, h_3))$ are *successor equivalent* to, or are *successors* of (h_2, k_2, h_3). We will also say that a diagram D *follows from* a diagram D' if it can be obtained from D by repeatedly taking successors. This yields a rewrite system with certain restrictions: all rewrite rules replace one adjacent pair of letters by another pair.

Note that all vertical diagrams are of the form (x_m, c_s, y_n). We write $D_{\mathfrak{S}} = (x_m, c_s, y_n)$ when $\mathfrak{S} = (s, m, n)$. It follows directly from equations (4), (5), (9), and (10), that

Lemma 1 *The diagram $D_{\mathfrak{S}'}$ is a successor of $D_{\mathfrak{S}}$ if and only if $\mathfrak{S}' = \Sigma(\mathfrak{S})$. In particular, any diagram of the form (x_m, c_s, y_n) has a unique successor if* $\mathbf{Inst}(s) \neq \mathtt{HALT}$*, and it has no successors if* $\mathbf{Inst}(s) = \mathtt{HALT}$.

Definition 2 *We call a diagram D' a* predecessor *of a diagram D when D is a successor of D'. Predecessors of a diagram (x_m, c_s, y_n) correspond to internal states $\mathfrak{S}$ such that $\Sigma(\mathfrak{S}) = (s, m, n)$.*

4 Properties of the Simulation

In this section we show that the halting problem for the abacus A is equivalent to determining whether there exist natural numbers n and m and a state s with $\mathbf{Inst}(s) = \texttt{HALT}$, such that the diagrams $D_{\mathfrak{S}_1}$ and (x_m, c_s, y_n) represent the same 2-cell in $\Pi_2(\mathcal{C}_{\mathbb{A}})$.

This quantification over the natural numbers is not actually necessary. Given an abacus A that can halt in a state (h, m, n) with m or n nonzero, it is always possible to modify it to obtain a new abacus A° that 'clears the registers' before halting but otherwise behaves identically. We create two new 'pre-halt' states p_1 and p_2 with the following properties:

$$\begin{array}{lll} \mathbf{Inst}(p_1) = \texttt{DECX}, & \sigma(p_1) = p_1, & \sigma'(p_1) = p_2 \\ \mathbf{Inst}(p_2) = \texttt{DECY}, & \sigma(p_2) = p_2, & \sigma'(p_2) = h \end{array}$$

Whenever $\sigma(s) = h$ or $\sigma'(s) = h$ in $\mathbb{A}$, the corresponding successor in $\mathbb{A}^\circ$ is p_1; otherwise $\mathbb{A}$ and $\mathbb{A}^\circ$ are identical. Thus, the halting problem for the original abacus $\mathbb{A}$ is equivalent to determining whether there is a state s with $\mathbf{Inst}(s) = \texttt{HALT}$ such that the diagrams $D_{\mathfrak{S}_1}$ and $D_{(x_m, c_s, y_n)}$ represent the same 2-cell in $\Pi_2(\mathcal{C}_{\mathbb{A}^\circ})$.

Theorem 1 *Let $\mathbb{A}$ be an abacus and $\mathcal{C}_{\mathbb{A}}$ the category described in the previous section. There exist natural numbers n and m and a state s with $\mathbf{Inst}(s) = \texttt{HALT}$ such that the diagrams $D_{\mathfrak{S}_1}$ and (x_m, c_s, y_n) are equivalent as 2-cells in $\Pi_2(\mathcal{C}_{\mathbb{A}})$ if and only if the abacus is able to reach a halting state.*

Proof: Suppose that the abacus is able to reach the halting state s, say $\Sigma^n(\mathfrak{S}_1) = (s, m, n)$ with $\mathbf{Inst}(s) = \texttt{HALT}$. Then it follows immediately from Lemma 1 that diagram $D_{\mathfrak{S}_1}$ is equivalent to diagram (x_m, c_s, y_n).

Conversely, suppose that there exist natural numbers n and m and a state s with $\mathbf{Inst}(s) = \texttt{HALT}$ such that the diagrams $D_{\mathfrak{S}_1}$ and (x_m, c_s, y_n) are equivalent. We will first show that in this case diagram $D_H := (x_m, c_s, y_n)$ follows from diagram $D_{\mathfrak{S}_1}$. Let

$$D_{\mathfrak{S}_1} = D_0 \sim \cdots \sim D_i \sim \cdots \sim D_{2n} = D_H \tag{12}$$

be a shortest list of direct equivalences which shows that $D_{\mathfrak{S}_1} \sim D_H$ (*i.e.*, this is a list so that no sublist would form a sequence of direct equivalences with the same beginning and ending diagrams). Note that every second diagram in this sequence is vertical. So for each $i \in \{0, \ldots, n\}$, the diagram D_{2i} is either a successor or a predecessor of D_{2i-2}. Since D_H does not have any successors, it follows that D_H is the successor of D_{2n-2}. Now let D_{2j} be the last diagram after which this sequence consists only of pairs of direct equivalences which correspond to successors. Then D_{2j-2} would have to be a successor of D_{2j}. However, since successors are unique by Lemma 1, this implies that $D_{2j-2} = D_{2j+2}$ and there exists a shorter list of direct equivalences from $D_{\mathfrak{S}_1}$ to D_H. This contradicts our assumption that the list (12) is minimal. We conclude that D_H follows from $D_{\mathfrak{S}_1}$.

Since (12) forms a list of successors, and successors are unique, it follows that $D_{2i} = D_{\Sigma^i(\mathfrak{S}_1)}$. We conclude that the even indexed diagrams in (12) encode the successive internal states of the abacus as it progresses to the halting state.

5 An Alternative Construction

The construction given in the previous section does not only yield a category for which the equivalence relation on $\Pi_2(\mathcal{C})_2$ is undecidable, but it shows rather concretely how any computation on an abacus may be translated into an equivalence problem for fences over an appropriate category. (A Java applet illustrating this for the Collatz $(3n+1)$-problem can be found on **cs.stmarys.ca/∼dawson/abacus.html**.) In this section, we will examine an alternative construction (first described in [2]) which, while rather less concrete, is shorter and yields undecidable equivalence problems for fences over some familiar categories such as **Set** and **Grp**. Our first proposition is probably a "folk theorem".

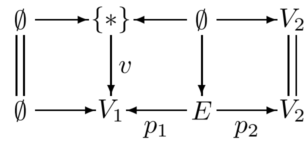
Proposition 1 *There exists a bipartite graph $\mathcal{G} = (V_1, V_2, E)$ with the following properties:*

1. *The vertex sets V_1 and V_2, and the set E of edges, are each indexed by a subset of $F \times \mathbb{N} \times \mathbb{N}$ for a finite set F; for each of the sets V_1, V_2, and E, there is a bounded-time algorithm to determine whether (s, x, y) is an index of an element of that set.*
2. *Given any two vertices of $\mathcal{G}$ there is a bounded-time algorithm to determine whether they are connected by an edge; and given any edge and any vertex there is a bounded-time algorithm to determine whether the edge contains the vertex.*
3. *The problem of determining whether two vertices are in the same component of $\mathcal{G}$ is undecidable.*

Proof: We construct $\mathcal{G}$ from a universal 2-register abacus $\mathbb{A}^\circ$, as described above, that halts only in the state $(h, 0, 0)$. One set of vertices V_1 corresponds to the complete internal states $\mathfrak{S} = (s, x, y)$ of the abacus; the other set, V_2, corresponds to the legal transitions $[\mathfrak{S} \to \Sigma(\mathfrak{S})]$. (As there is at most one legal transition out of any given state, the first claim follows.) There are edges joining each of the vertices $\mathfrak{S}$ and $\Sigma(\mathfrak{S})$ to the vertex $[\mathfrak{S} \to \Sigma(\mathfrak{S})]$, and no other edges; the second claim thus follows trivially. Finally, it is evident that determining whether (s, x, y) and $(h, 0, 0)$ are in the same component is equivalent to solving the halting problem for $\mathbb{A}^\circ$, from which the third claim follows.

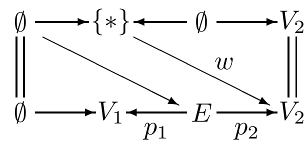
Theorem 2 *For any bipartite graph $\mathcal{G} = (V_1, V_2, E)$, there exists a family of fences over* **Set***, all with the same domain and the same codomain and indexed by the vertices of $\mathcal{G}$, such that two vertices are in the same component of $\mathcal{G}$ if and only if the corresponding fences are equivalent.*

Proof: For $i = 1, 2$, let $p_i : E \to V_i$ be the function that sends an edge to its endpoint in V_i. Consider the class of fences of the form



(13)

We note that these are in 1-1 correspondence with the functions $v : \{*\} \to V_1$, that is, with the elements of V_1. The only other fences with this domain and codomain are of the form



(14)

where $t(*)$ is a transition $x \to y$ in V_2.

If two such fences are directly equivalent, there exists a function $u : \{*\} \to E$ such that $p_1 u = v$ and $p_2 u = t$. Thus, $u(*)$ must be of the form (v, t), where $t(*) = (v \to w)$ or $(w \to v))$. Conversely, a fence of form (14) in which $*$ is mapped to the transition $(x \to y)$ can only be directly equivalent to a fence of form (13) in which $*$ is mapped to x or y. It follows that two fences of the form (13), in which $*$ is mapped to v and v' respectively, are equivalent if and only if v and v' are in the same connected component of $\mathcal{G}$.

The main theorem of this section follows immediately from this:

Theorem 3 *The equivalence of fences over* **Set** *is undecidable.*

Corollary 1 *If* **Set** *is a full subcategory of* $\mathcal{C}$*, then the equivalence of fences over* $\mathcal{C}$ *is undecidable.*

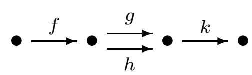
This corollary covers many familiar categories such as topological spaces, categories, posets, graphs, and several common variants of these (*e.g.*, Hausdorff spaces and directed graphs). Substituting the one-point set for the empty set in the construction lets us extend this result to other categories, such as pointed sets, and pointed topological spaces. A further variation, given in [2], models the word problem for groups to show that the equivalence of fences over **Grp** is also undecidable.

6 Decidability Results

In the previous two sections we have given examples of categories $\mathcal{C}$ for which the equivalence relation on $\Pi_2(\mathcal{C})_2$ is undecidable. But this is not a problem for *all* categories. In this section we discuss two types of categories for which the equivalence relation is decidable. We have already observed one condition on the category $\mathcal{C}$ that makes the equality of 2-cells in $\Pi_2(\mathcal{C})$ decidable.

Proposition 2 *If the category $\mathcal{C}$ is locally finite, equality of 2-cells in $\Pi_2(\mathcal{C})$ can be decided in bounded time.*

The construction in the previous section makes heavy use of the fact that the arrows in the category $\mathcal{C}$ don't cancel. If a category $\mathcal{C}$ satisfies the condition that for any diagram of the form



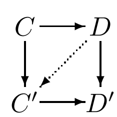
we have

$$kg = kh \text{ and } gf = hf \text{ implies } g = h \qquad (15)$$

the word problem for 2-cells in $\Pi_2(\mathcal{C})$ becomes decidable in bounded time:

Proposition 3 *If the category $\mathcal{C}$ satisfies the condition (15), equality of 2-cells in the category $\Pi_2(\mathcal{C})$ can be determined in bounded time: maximally $(m-1)(n-1)$ searches for the existence of a factorization and $m+n-2$ checks for equality of two arrows in $\mathcal{C}$ are required to determine whether two representatives for a 2-cell $(g_1, \cdots, g_{n-1}; f_1, \cdots, f_n) \Rightarrow (g'_1, \cdots, g'_{m-1}; f'_1, \cdots, f'_m)$ are equivalent.*

Proof:Note that in this condition is equivalent to requiring that every square of the form



has maximally 1 factorization. Consequently, a situation as described in the previous section can not happen: if two representatives have the same index functions, they are only equivalent when they are the same. So the only work that needs to be done in checking whether two 2-cells are equivalent, is to see whether the appropriate factorizations exist to change the index functions of one of them into the index functions of the other, and then one needs to check whether the corresponding arrows in the resulting diagrams are equal. There exist maximally $(m-1)(n-1)$ factorizations for a set of diagrams representing one 2-cell, and $m+n$ equality checks are needed.

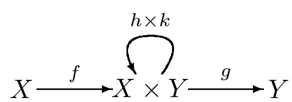
6.1 Examples

1. If the category $\mathcal{C}$ has only monic or only epic maps, $\mathcal{C}$ satisfies this condition.

2. If $\mathcal{C}$ is a category with only monic maps and $\mathcal{D}$ has only epic maps, the category $\mathcal{C} \times \mathcal{D}$ satisfies this condition.

3. For any category $\mathcal{C}$, the subcategory of the category of arrows $\mathcal{C}^{\rightarrow}$ consisting of squares where the top arrow is monic and the bottom is epic (or vice versa) satisfies this condition.

4. The category generated by the following diagram in **Set** satisfies this condition:



where $f(x) = (x, y_0)$, h is an isomorphism and k is an isomorphism that keeps y_0 fixed, $y_0 \in Y$ is a chosen fixed point and $g(x, y) = y$.

References

[1] R. J. MacG. Dawson, R. Paré, What is a double category like?, *Jour. Pure Appl. Alg.* 168 (2002), pp. 19-34.

[2] R. J. MacG. Dawson, R. Paré, D. A. Pronk, Undecidability and free adjoints, in *Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics 2001*, Volume XIV, N. Callaos, F. G. Tinetti, J. M. Champarnaud, J. K. Lee (Eds), International Institute of Informatics and Systemics, Orlando, 2001, pp. 156-161

[3] R. J. MacG. Dawson, R. Paré, D. A. Pronk, Adjoining adjoints, to appear in *Adv. in Math.*

[4] R. J. MacG. Dawson, R. Paré, D. A. Pronk, Free extensions of double categories, in preparation.

[5] R. J. MacG. Dawson, R. Paré, D. A. Pronk, Free adjoints and spans, in preparation.

[6] P. Gabriel and M. Zisman, *Calculus of Fractions and Homotopy Theory*, Springer-Verlag, New York, 1967.

[7] J. Lambek, How to program an infinite abacus, *Canad. Math. Bull.* 4 (1961), pp. 295-302.

[8] M. L. Minsky, Recursive unsolvability of Post's problem of 'tag' and other topics in the theory of Turing machines, *Annals of Math.* 74 (1961), pp. 437-455.

[9] D. A. Pronk, Etendues and stacks as bicategories of fractions, *Comp. Math.* 102 (1996), pp. 243-303.

[10] D. Quillen, *Homotopical Algebra*, LNM 43, Springer Verlag, New York, 1967.

[11] S. Schanuel, R. Street, The free adjunction, *Cahier Topologie Géom. Differentielle Catégoriques* 27 (1986), pp. 81-83.