Reversing graph transformations*

Paweł Sobociński

Computer Laboratory, University of Cambridge, United Kingdom

We present an application of the framework developed in [1] to the theory of graph transformation systems. Here we say "graph" to mean an object of an arbitrary category with pushouts along monos where the local Church-Rosser theorem holds. An example is an adhesive category [6].

In order to establish the basic concepts we shall consider a concrete example, working in the category $\mathbf{C}$ of directed graphs whose vertices are tagged with the elements of a set; the presheaf topos $\mathbf{C} = \mathbf{Set}^{\cdot \rightarrow \cdot \leftleftarrows}$. The edges of such graphs will represent physical proximity of entities represented by the vertices. The elements with which vertices may be tagged represent the internal state of the entities. Let T be the graph illustrated in Fig 1. Then $\mathbf{C}/T$ is the adhesive category [6] of graphs typed over T.

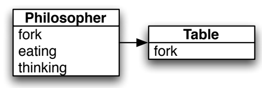

Fig. 1. Type graph T.

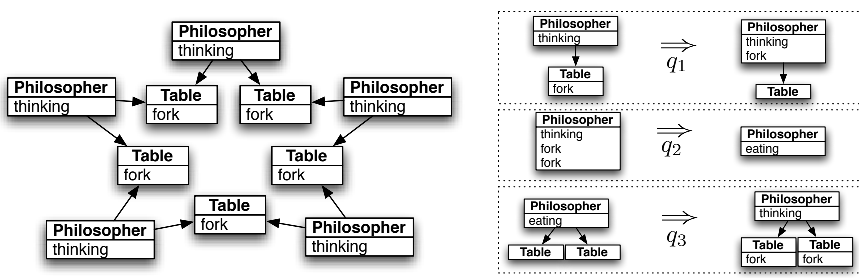

Fig. 2. Start graph S and productions q_1, q_2 and q_3 of $\mathcal{P}$.

Our example models Hoare's dining philosophers problem [5]. Let $\mathcal{P}$ be the DPO grammar over $\mathbf{C}/T$ with start graph S and the three productions q_1, q_2 and q_3 as illustrated in Fig 2. Note that we only illustrate the left and the right hand sides of the productions, their interface is the obvious one in each case. Each thinking philosopher may claim a fork next to her using the production q_1. Once a thinking philosopher has two forks in her possession, she may start eating via the production q_2. Finally, an eating philosopher can release her forks at any time and return to thinking using production q_3. The dining philosophers problem is famous not least for the fact that it succinctly illustrates the fundamental issue of *deadlock* in parallel programming. If each of the philosophers picks up the fork to her

* Research partially supported by EPSRC grant EP/D066565/1.

left then no further productions are possible and the philosophers starve to death. To solve this problem, one may choose to view of the behaviour of each philosopher as a series of *transactions* – where each transaction consists of a series of actions (two instances of q_1) which lead to a state in which a desired action (q_2) can be performed. In order to avoid deadlock, one specifies that each of the initial actions can be reversed.

The naive solution of simply adding a reversed production $q_{1_\star}$, illustrated in Fig 3 is unsatisfactory. Indeed, a philosopher can now begin by picking up her left fork with q_1 and placing it via $q_{1_\star}$ together with her right fork. This sequence of actions results in a state not in the problem specification. One can think of several ways to fix this: for example, one can label the edges out of each philosopher with l and r, replace the rule q_1 with two rules q_{1l}, q_{1r} and add their inverses $q_{1l_\star}$, $q_{1r_\star}$, thus disallowing the aforementioned errant behaviour. There are two apparent problems with such an ad-hoc solution: firstly, one has to prove that the transactions are indeed modelled correctly (trivial in this case, but not always so); secondly, there is a danger of making the model too complex to be of use. There is a *general* solution, described in [1], which we briefly outline below.

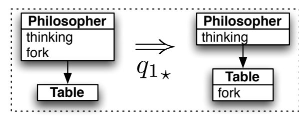

Fig. 3. Reversed rule $q_{1_\star}$.

Let the *category of computations* $\mathbf{Cp}\,\mathcal{G}$ of a grammar $\mathcal{G}$ be the category with objects those of $\mathbf{C}$ and arrows finite (possibly empty) paths of direct derivations modulo switch-equivalence. The arrows of $\mathbf{Cp}\,\mathcal{G}$ are thus the concurrent computations of $\mathcal{G}$. See [4, Ch 4] for a more in depth presentation and a proof that such a category also arises as a free construction.

In the problem specification, the set of productions P of $\mathcal{G}$ is partitioned into sets of reversible productions R and irreversible productions I. In our example, $R = \{q_1\}$ and $I = \{q_2, q_3\}$. Let $\mathcal{R}$ be the subcategory of $\mathbf{Cp}(\mathcal{G})$ with arrows the derivations consisting of only the reversible productions. Let $\mathcal{I}$ be the subcategory of $\mathbf{Cp}(\mathcal{G})$ consisting of the irreversible computations – roughly, those where the last action in each thread is irreversible. The arrows of $\mathcal{I}$ can be considered as paths of transactions modulo concurrency - where a transaction is a causal sequence of reversible steps followed by an irreversible step. Since $\mathcal{I}$ is a subcategory, it is clear that we allow empty transactions.

It is not difficult to verify that $\langle \mathcal{I}, \mathcal{R} \rangle$ is a factorisation system [2] on $\mathbf{Cp}(\mathcal{G})$ – each computation can be factorised into a (possibly empty) irreversible component followed by a (possibly empty) reversible component, moreover, such factorisation is essentially unique. As well as the notion of a factorisation system, we shall need the notion of a category of fractions [3]. Given a set of morphisms $\mathcal{R}$ of a category $\mathbf{C}$, the category of fractions $\mathbf{C}[\mathcal{R}^{-1}]$ is the category resulting from $\mathbf{C}$ by "freely" adding inverses to the arrows of $\mathcal{R}$. We obtain a canonical functor $\Phi : \mathbf{C} \to \mathbf{C}[\mathcal{R}^{-1}]$ which sends each arrow in $\mathcal{R}$ to an isomorphism.

Let $h(\mathbf{Cp}\,\mathcal{G}, \mathcal{R})$ be the category of histories. The objects of this category are arrows in $\mathcal{R}$ (reversible computations), while the arrows are commutative diagrams, as illustrated, where f is in $\mathbf{Cp}\,\mathcal{G}$ and f' is in $\mathcal{I}$. Clearly, *any* computation $f : Q_1 \to Q_2$ leads to a (unique up to isomorphism) object $g_2 \colon P_2 \to Q_2$ of $h(\mathbf{Cp}\,\mathcal{G}, \mathcal{R})$ and f' resulting in a map $g_1 \to g_2$ – here $g_2 f'$ is simply the $\langle \mathcal{I}, \mathcal{R} \rangle$-factorisation.

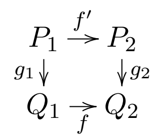

Fig. 4. $h(\mathbf{Cp}\,\mathcal{G}, \mathcal{R})$.

Let $h_\star(\mathbf{Cp}\,\mathcal{G}, \mathcal{R})$ be the category of reversible histories. It has the same objects as $h(\mathbf{Cp}\,\mathcal{G}, \mathcal{R})$; arrows are formal diagrams, as illustrated, where $f_\star$ is in $\mathbf{C}[\mathcal{R}^{-1}]$ and $f_\star \Phi(g_1) = \Phi(g_2 f)$. Roughly speaking, this category is as $h(\mathbf{Cp}\,\mathcal{G}, \mathcal{R})$ but histories can be backtracked [1]. Note that while we constructed the history categories for our particular category $\mathbf{Cp}\,\mathcal{G}$, in fact the constructions rely only on the presence of a factorisation system. The main result of [1] states that there is an equivalence of categories $h_\star(\mathbf{Cp}\,\mathcal{G}, \mathbf{C}) \simeq \mathcal{I}$. In other words, to simulate transactions for a graph grammar $\mathcal{G}$, one replaces the category of computations $\mathbf{Cp}\,\mathbf{C}$ with $h_\star(\mathbf{Cp}\,\mathbf{C}, \mathcal{R})$. More concretely, one: (i) replaces the states of a computation; instead of a graph Q, a state is a reversible computation $f \colon P \to Q$; (ii) adds an inverse production $q_\star$ for each reversible production q. The resulting computations are "weakly" (modulo reversible moves) equivalent to the transactions of the original grammar.

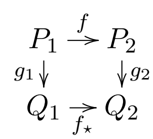

Fig. 5. $h_\star(\mathbf{Cp}\,\mathcal{G}, \mathcal{R})$.

References

1. V. Danos, J. Krivine, and P. Sobociński. General reversibility. In EXPRESS '06, Electronic Notes in Theoretical Computer Science. Elsevier, 2006. To appear.
2. P. J. Freyd and G. M. Kelly. Categories of continuous functors, I. *Journal of Pure and Applied Algebra*, 2:169–191, 1972.
3. P. Gabriel and M. Zisman. *Calculus of fractions and homotopy theory.* Springer-Verlag, 1967.
4. R. Heckel. *Open Graph Transformation Systems: A New Approach to Compositional Modelling of Concurrent and Reactive Systems.* PhD thesis, TU Berlin, 1998.
5. C. A. R. Hoare. Towards a theory of parallel programming. In *Seminar at Queen's University, Belfast, Northern Irelend.* Academic Press, 1972.
6. S. Lack and P. Sobociński. Adhesive and quasiadhesive categories. *Theoretical Informatics and Applications*, 39(3):511–546, 2005.