Axiomatic Rewriting Theory I: A Diagrammatic Standardization Theorem

Paul-André Melliès

Equipe Preuves, Programmes et Systèmes,
CNRS, Université Paris 7 Denis Diderot

Abstract. By extending *nondeterministic* transition systems with *concurrency* and *copy mechanisms*, Axiomatic Rewriting Theory provides a uniform framework for a variety of rewriting systems, ranging from higher-order systems to Petri nets and process calculi. Despite its generality, the theory is surprisingly simple, based on a mild extension of transition systems with independence: an axiomatic rewriting system is defined as a 1-dimensional transition graph $\mathcal{G}$ equipped with 2-dimensional transitions describing the *redex permutations* of the system, and their orientation. In this article, we formulate a series of elementary axioms on axiomatic rewriting systems, and establish a diagrammatic standardization theorem.

Foreword by the Author

Many concepts of Rewriting Theory started in the λ-calculus — which is by far the most studied rewriting system in history. A remarkable illustration is the ***confluence theorem***. The theorem was formulated by A. Church and J.B. Rosser in the early years of the λ-calculus [7]. The theorem was then generalized and applied extensively to other rewriting systems. It became eventually an object of study in itself, in a line of research pioneered by H.-B. Curry and R. Feys in their book on Combinatory Logic (1958). This culminated in a series of beautiful papers by G. Huet, J. W. Klop, and J.-J. Lévy published at the end of the 1970s and beginning of the 1980s. Today, more than half a century after its appearance in the λ-calculus, the confluence property is universally accepted as the theoretical principle underlying *deterministic* computations.

The article is concerned with another key property of the λ-calculus: the ***standardization theorem***, which was discovered by A. Church and J.B. Rosser quite at the same time as the confluence property. We advocate in this article that, in the same way as confluence underlies deterministic computations, standardization guides *causal* computations. It is worth clarifying here what kind of causality we have in mind, since the concept has been used in so many different ways. First of all, by *computation*, we mean a rewriting path

$$M_1 \xrightarrow{u_1} M_2 \xrightarrow{u_2} M_3 \longrightarrow \cdots \longrightarrow M_{n-1} \xrightarrow{u_n} M_n$$

in which every term M_k describes a particular state of the system, and in which every redex u_k describes a particular transition on states, for $1 \leq k \leq n$. Then,

A. Middeldorp et al. (Eds.): Processes... (Klop Festschrift), LNCS 3838, pp. 554–638, 2005.

by *causal computation*, we mean a computation in which every transition u_k is enabled by a chain or cascade of previous transitions. We are particularly interested in situations where the chain of causality leading to u_k is not necessarily the whole rewriting path

$$M_1 \xrightarrow{u_1} M_2 \xrightarrow{u_2} M_3 \longrightarrow \cdots \longrightarrow M_{k-1} \xrightarrow{u_{k-1}} M_k. \tag{1}$$

At this point, we advise the reader to practice the following spiritual exercise: think of today as a particular sequence of transitions (1) starting from your bedroom (state M_1) and leading you to the current position in the day (state M_k). Then, call $v = u_k$ the transition consisting in reading this very article:

$$v \;=\; u_k \quad : \quad M_k \longrightarrow M_{k+1}.$$

You must admit that some transitions performed today among the $u_1, \ldots, u_{k-1}$ are not necessary to read this article. And that it seems particularly difficult to disentangle the necessary transitions from the unnecessary ones. This is the point of this article: we investigate how to perform this task in Rewriting Theory by *permuting* transitions — in the spirit of true concurrency and Mazurkiewicz traces. Suppose for instance that your last action $u = u_{k-1}$ today has been to drink coffee:

$$u \;=\; u_{k-1} \quad : \quad M_{k-1} \longrightarrow M_k.$$

Do you really need that coffee to read these lines? The simplest way to answer is to check whether the transition v may be permuted before the transition u. If this is the case, then coffee is not necessary. Of course, you may reply that you have already drunk your coffee ten minutes ago, and thus, that it is far too late *now* to permute the order of events! You are certainly right... but this is not what matters here: the very fact that permuting the transition v before the transition u is possible *in principle* is sufficient to establish that performing transition u is not necessary in order to perform transition v.

Suppose on the other hand that your last action u has been to fetch this article from the library. In that case, performing the transition u is absolutely necessary in order to perform the transition v. There is no way indeed (either in reality or in principle) to permute the order of the two transitions... and this is precisely the reason why you went to the library on the first hand!

Of course, separating the necessary transitions from the unnecessary ones may involve more than just one permutation. Suppose for instance that you have drunk coffee just before fetching the article from the library. In that case, it takes two permutations (permute your coffee time after your visit to the library, and then after your exploration of the article) in order to demonstrate that drinking coffee is not necessary.

Everyday life shows that chains of causality may be reconstructed by applying relevant series of permutations on transitions. Now, Rewriting Theory complicates matters by implementing a symbolic universe in which computations may be *erased* or *duplicated* at will. New situations arise, which often defy common