Least Upper Bounds on the Size of Church-Rosser Diagrams in Term Rewriting and $\lambda$-Calculus

Jeroen Ketema$^1$ and Jakob Grue Simonsen$^2$

$^1$ Faculty EEMCS, University of Twente
P.O. Box 217, 7500 AE Enschede, The Netherlands
j.ketema@ewi.utwente.nl

$^2$ Department of Computer Science, University of Copenhagen (DIKU)
Universitetsparken 1, 2100 Copenhagen Ø, Denmark
simonsen@diku.dk

Abstract. We study the Church-Rosser property—which is also known as confluence—in term rewriting and $\lambda$-calculus. Given a system $R$ and a peak $t^* \leftarrow s \rightarrow^* t'$ in $R$, we are interested in the length of the reductions in the smallest corresponding valley $t \rightarrow^* s' \leftarrow t'$ as a function $v_{s_R}(m, n)$ of the size $m$ of $s$ and the maximum length $n$ of the reductions in the peak. For confluent term rewriting systems (TRSs), we prove the (expected) result that $v_{s_R}(m, n)$ is a computable function. Conversely, for every total computable function $\varphi(n)$ there is a TRS with a single term $s$ such that $v_{s_R}(|s|, n) \geq \varphi(n)$ for all $n$. In contrast, for orthogonal term rewriting systems $R$ we prove that there is a constant $k$ such that $v_{s_R}(m, n)$ is bounded from above by a function exponential in $k$ and independent of the size of $s$. For $\lambda$-calculus, we show that $v_{s_R}(m, n)$ is bounded from above by a function contained in the fourth level of the Grzegorczyk hierarchy.

1 Introduction

The Church-Rosser property—also called confluence—is a property of rewriting systems which states that any peak $t^* \leftarrow s \rightarrow^* t'$ has a corresponding valley $t \rightarrow^* s' \leftarrow t'$. The valley and the term $s'$ are said to complete the diagram.

In functional programming, the Church-Rosser property ensures that different ways of evaluating a program will always yield the same end result (modulo non-termination): The outcome will be independent of the evaluation order or reduction strategy. In logic, if a deductive system has the Church-Rosser property, the system will be consistent: No statement can both hold and not hold.

While the Church-Rosser property has been shown to hold for a wide variety of rewrite systems, there has, to our knowledge, never been an investigation into the number of reduction steps in a valley that completes the diagram of a peak of a given size (see Figure 1). Succinctly: The question “How large is the valley as a function of the peak?” has apparently never been asked.
We believe the above question to be *intrinsically* interesting from a theoretical point of view, as Church-Rosser-type results are ubiquitous. We also believe the practical implications in mainstream functional programming to be limited: Standard functional languages like ML and Haskell employ a fixed evaluation strategy such as call-by-value or call-by-need, and there seems to be little interest in performing optimisations by switching strategies (modulo non-termination). However, for more specialised languages, like declarative DSLs where the evaluation order may not be fixed, there may be practical implications: If, for small peaks, the size of the smallest corresponding valley is so large that a term completing the Church-Rosser diagram cannot be computed using realistic resources, then it matters *very much* what kind of reduction strategy is used: Choosing the ‘wrong’ evaluation strategy (say, call-by-value) and performing just a few steps of computation could result in a very long reduction before a result is reached—better to backtrack to the original term and try another strategy. Apparently, there is no prior research concerning this problem in the foundational basis of declarative programming—\(\lambda\)-calculus and term rewriting. There does exist some literature on length of shortest and longest reductions to normal form for certain classes of systems \[14,8,16\], but the Church-Rosser theorem does not concern normal forms: It also applies to systems where some (or all) terms may fail to have normal forms.

In this paper, we perform the first fundamental study of the size of peaks and valleys for systems having the Church-Rosser property; specifically we study how the size of a peak affects the *valley size* of the smallest corresponding valley. We consider three very general settings: That of (arbitrary) first-order term rewriting systems, of orthogonal term rewriting systems (roughly corresponding to first-order functional programs that have no fixed evaluation order), and untyped \(\lambda\)-calculus. We believe that these three areas cover most of the non-specialised areas where the Church-Rosser property occurs; the most significant omission is the case of general higher-order rewrite systems (including general higher-order functional programs and logics with bound variables)—we expect general upper bounds in that case to be difficult to derive (and, likely, to be astronomical), as is foreshadowed by our treatment of \(\lambda\)-calculus in Section 5.