Improving Lazy Non-Deterministic Computations by Demand Analysis

Michael Hanus

Institut für Informatik, CAU Kiel, D-24098 Kiel, Germany.
`mh@informatik.uni-kiel.de`

Abstract. Functional logic languages combine lazy (demand-driven) evaluation strategies from functional programming with non-deterministic computations from logic programming. The lazy evaluation of non-deterministic subexpressions results in a demand-driven exploration of the search space: if the value of some subexpression is not required, the complete search space connected to it is not explored. On the other hand, this improvement could cause efficiency problems if unevaluated subexpressions are duplicated and later evaluated in different parts of a program. In order to improve the execution behavior in such situations, we propose a program analysis that guides a program transformation to avoid such inefficiencies. We demonstrate the positive effects of this program transformation with KiCS2, a recent highly efficient implementation of the functional logic programming language Curry.