

Braßel, Bernd; Hanus, Michael; Huch, Frank

Encapsulating non-determinism in functional logic computations. (English) Zbl 1084.68511
J. Funct. Log. Program. 2004, Spec. Iss. 1, Article 6, 28 p. (2004).

Summary: One of the key features of the integration of functional and logic languages is the access to non-deterministic computations from the functional part of the program. In order to ensure the determinism of top-level computations in a functional logic program, which is usually a monadic sequence of I/O operations, one has to encapsulate the non-determinism (i.e., search for solutions) occurring in logic computations. However, an appropriate approach to encapsulation can be quite subtle if subexpressions are shared, as in lazy evaluation strategies. In this paper we examine the current approaches to encapsulate non-deterministic computations for the declarative multi-paradigm language Curry, show their relative advantages and the problems they induce. Furthermore, we present a new approach which combines the advantages but avoids the problems. Our proposal is based on providing a primitive I/O action for encapsulation from which various specialized search operators can be derived. In order to provide a formal foundation for this new approach to encapsulation, we define the operational semantics of this new primitive.

MSC:

[68N17](#) Logic programming

[68N18](#) Functional programming and lambda calculus

Cited in **6** Documents

Keywords:

[operational semantics](#)

Software:

[PAKCS](#)

Full Text: [Link](#)