

②5 MLOG: A Strongly Typed Confluent Functional
Language with Logical Variables

V.Poirriez(INRIA,フランス)

発表要旨

関数型言語と論理的な道具とを組み合わせ、強力な言語を作るために多くの試みがなされているが、本稿ではその一つとしてMLOGという言語を設計した。この言語は、logical variableを加えてMLを保守的拡大した言語である。また、MLOGのコンパイラとして“CAML Light FLUO”を作った。

MLOGでは、評価の順番にある程度の自由度はあるものの、logical variableは一度しか値が代入できないようになっていること、また値が与えられていないと止まってしまう(suspension mechanism)などの特徴がある。

MLOGの操作的意味は、“A strict calculus with environment”によって与えられる。これは、Church Rosser Propertyが成り立つような体系であり、reductionの仕方に依存しないでその値が一意に定まることが保証されている。しかし、その拡張のために、logical typeからnon-logical typeへ写すような関数は使うことができないという制限が加えられることになる。さもなければ、もはや合流性は満たされなくなってしまう。

質疑応答

質問：Church Rosser Propertyが成り立つことを定理として示しましたが、具体的にその証明はどのようにして行なうのでしょうか。例えば良く使われるようなスケッチによる方法で証明できるのでしょうか。

回答：無解答