

Paper #8

Author: Kamen Yotov et al.

Title: A Comparison of Empirical and Model-driven Optimization

Reviewer: Ilie Savga

Summary:

The growing interest to the use of empirical program optimizers can be explained by impressive efficiency of generated code that is highly tuned to particular architectures. As opposite to conventional compilers, which use models of programs and machines to find the best values of key optimization parameters, empirical optimizers rely on generation of different program versions with running them on the actual hardware. The scope of the paper is to evaluate the relative effectiveness of empirical and model-driven optimizations. For this, ATLAS - an existing system for empirical optimization - is used as well as a purposely-developed one with a model-based optimization engine. The performance using both systems is measured on three different platforms. The results show that though the model-based approach is generally less efficient, it still can be surprisingly acceptable, especially on particular machine architectures.

The main contributions

The paper provides the first quantitative evaluation of the differences between empirical and model-driven optimizations. It also shows the results of the performed comparison between these two kinds of optimizers and discusses possible improvements of evaluation techniques.

Another important contribution is that they show how a model-based optimization engine can replace an existing empirical one.

Merits and weaknesses

The paper is well written, though contains sometimes many technical details hard to understand by a novice. The authors describe an empirical program optimizer by exploring its strategy of code generation and optimization parameters' discovery. Also, a model-based approach is thoroughly presented in stages. The analysis of the performance for each system and possible improvement of evaluation techniques clearly summarizes the paper.

While describing technical aspects of comparison, the paper does not discuss why and when the need in such evaluation may appear. It also does not mention the advantages and disadvantages of these two kinds of optimizers except for the efficiency of the generated code.

Numerical rating

Significance 5

Originality 5

Interest to a journal on programming languages and compiler technology 1

Quality of experimental evaluation 9

Overall organization 8

Presentation (style and language) 10

Length appropriate 9

References appropriate 10

Overall evaluation 7

Recommendation Weak accept

Confidence 2

Comments and suggestions for improvement:

The paper is generally well written; still, it contains a lot of technical discussion that is sometimes hard to follow. In addition, you should address the following issues:

- You present the comparison techniques between empirical and model-driven optimizers, but do not provide the discussion why and when such evaluation might be needed. This seems to be the main drawback of the paper as clearly no reason for your work is given. So, please, correct this by extending some sections of the paper (at least, abstract and introduction).

- a short discussion of the advantages and disadvantages between the two kinds of systems also seems reasonable in the introduction.