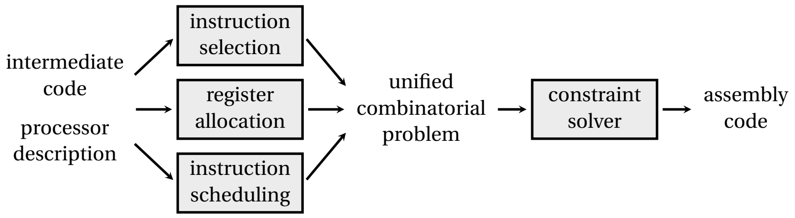
Unison
Assembly Code Generation Using Constraint Programming

Roberto Castañeda Lozano[1], Gabriel Hjort Blindell[2], Mats Carlsson[1] and Christian Schulte[2,1]

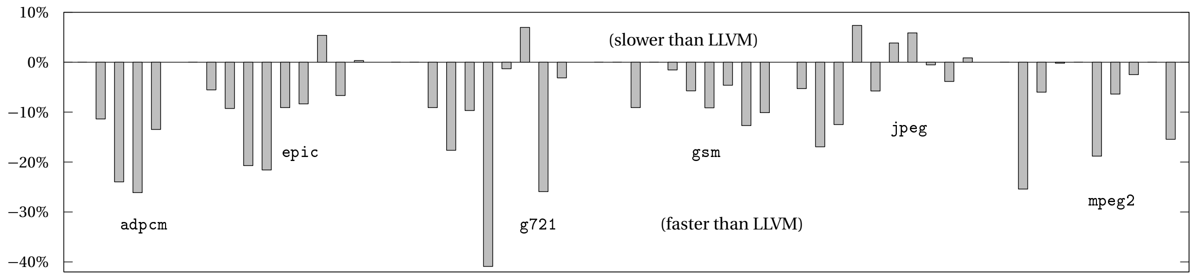
[1]SCALE, SICS (Swedish Institute of Computer Science), Sweden {rcas,matsc}@sics.se
[2]SCALE, School of ICT, KTH Royal Institute of Technology, Sweden {ghb,cschulte}@kth.se

Code generation translates the intermediate representation of a program into assembly code for a particular hardware architecture. This translation has a critical impact on the quality of a compiler's output. State-of-the-art compilers such as GCC and LLVM decompose code generation into multiple tasks (instruction selection, instruction scheduling and register allocation) and solve each task with heuristic algorithms. This scheme disregards the typically strong interdependencies among tasks and their combinatorial nature.

We demonstrate *Unison* – a simple, flexible and potentially optimal code generator based on a radically different scheme. In Unison, interdependent code generation tasks are translated into combinatorial problems and solved together with constraint programming, a modern combinatorial optimization method:



Unison's combinatorial models capture a wide array of hardware architectures ranging from out-of-order, general-purpose to VLIW digital signal processors (DSPs). Ongoing experiments with MediaBench functions for Hexagon (a DSP ubiquitous in modern mobile platforms) show that Unison consistently improves the speed of the code generated by LLVM, sometimes by as much as 40%:



Unison reuses the state-of-the-art LLVM compiler infrastructure for the platform-independent compilation tasks and can be easily integrated into the regular LLVM toolchain as well as accessed from other compilers.

Further reading

■ R. Castañeda Lozano, M. Carlsson, F. Drejhammar, and C. Schulte. Constraint-based register allocation and instruction scheduling. In *CP*, volume 7514 of *LNCS*, pages 750–766. Springer, 2012.

■ R. Castañeda Lozano, G. Hjort Blindell, M. Carlsson, F. Drejhammar, and C. Schulte. Constraint-based code generation. In *SCOPES*, pages 93–95. ACM, 2013.

This research has been partially funded by LM Ericsson AB and the Swedish Research Council (VR 621-2011-6229).