Design Space Exploration and Optimization of Embedded Cache Systems via a Compiler

Rodric M. Rabbah
Joint work with Krishna V. Palem
Center for Research on Embedded Systems and Technology
Georgia Institute Of Technology
http://www.crest.gatech.edu

The research presented here is funded in part by DARPA contract Nos. F33165-99-1-1499 (DIS) and F30602-00-2-0564 (PACC), HP Labs and Yamacraw
Design Space Exploration and Optimization of Embedded Cache systems via a Compiler

The original document contains color images.
High Performance Embedded Processors
Conventional Design Flow

- Current strategy is ad hoc engineering
- **The system engineer is on the critical path**
- High engineering costs
  - How can the lessons learned from one design be used in new contexts?
- Slow time-to-market
Desiderata – A New Paradigm

- Is it possible to capture, quantify, and characterize the virtual program behavior?
- Is it possible to track or measure the program behavior as a result of the applied compiler optimizations?
- Is it possible to interpret the virtual program behavior in light of an architectural context?
  - Evaluation of the architecture without actual synthesis

Our Fundamental Philosophy

Virtual Program Behavior → Realized Architecture Behavior

The Past
- Human design based on benchmarks
- High NRE Slow-time-to-market

The Future
- Automated exploration and architecture evaluation based on quantitative measures
- Low NRE Fast-time-to-market
- Optimized workload specific design

Realized Architecture Behavior

compiler
Example Design Space Exploration
Via Data Remapping

- Remapping reduces the size of the working set and decreases the demand bandwidth of the application

- We may leverage the benefits afforded by data remapping to optimize the architecture
  - Tradeoff cache size and performance
  - Halving of the cache requirements as well as the power consumed while preserving performance goal

- Systematic exploration of the design space based on quantitative measures