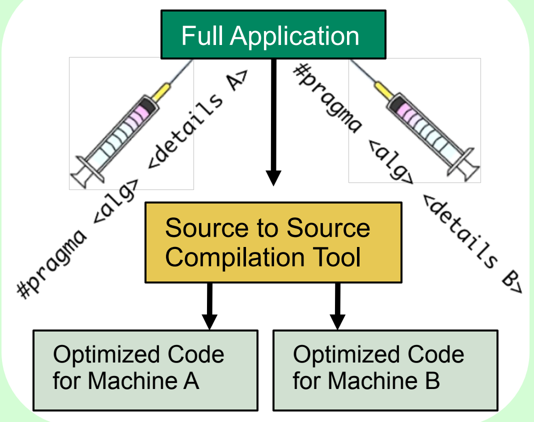
SAIMI: Separating Algorithm and Implementation via Programming Model Injection

Problem: Algorithms in programs are obfuscated due to implementation details and performance tuning done per machine and by hand.
Solution: Use simpler, more restricted programming models to express important sub-computations. Express implementation details as transformations of sub-computations.
Status: We are evaluating separation in existing programming models and developing look-up table, grid, and task graph injectable programming models.

Illustration of the SAIMI concept.



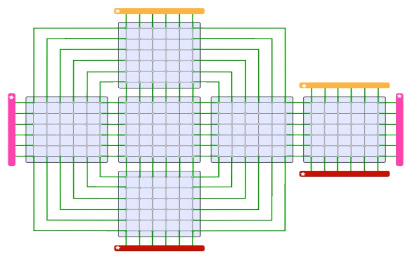
Full Application
#pragma <alg> <details A>
#pragma <alg> <details B>
Source to Source Compilation Tool
Optimized Code for Machine A
Optimized Code for Machine B

This won't hurt a bit!

Separating Grid Details

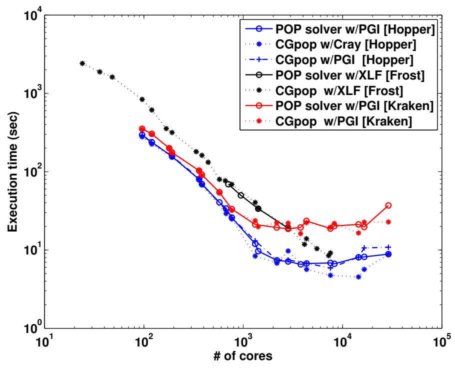
Earth simulation applications often have grid details tangled with algorithm and communication code. The GridWeaver project aims to separate these details with a library interface where grids are described declaratively. Code generation techniques are used to replace library calls with more efficient code.

Modeled grids in the GridWeaver library are defined as a series of regular subgrids connected together. The following picture represents a cube-sphere grid. The blue boxes represent regular subgrids, the green edges represent connections between these subgrids.



Grid-Based Computation

Many simulation applications become tangled with the underlying discretization grid. We are developing ways to express grid details orthogonally from the computation performed on the grid. We will evaluate our solutions with CGPOP, which is a miniapp that models the conjugate gradient solver in the Parallel Ocean Program. The figure below shows that the 3000 line CGPOP mini app behaves as a performance proxy for the 75K line POP application.



Early Career Award, DOE grant DE-SC3956
April 2010 through March 2015

U.S. DEPARTMENT OF ENERGY

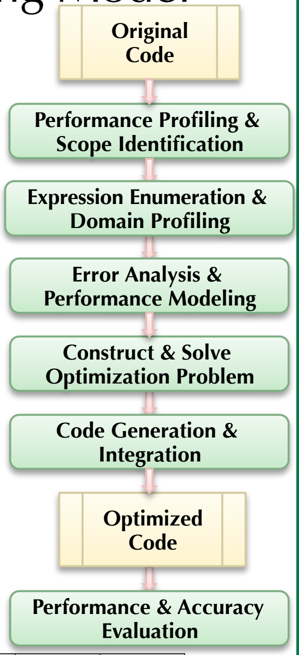
http://www.cs.colostate.edu/hpc/SAIMI/

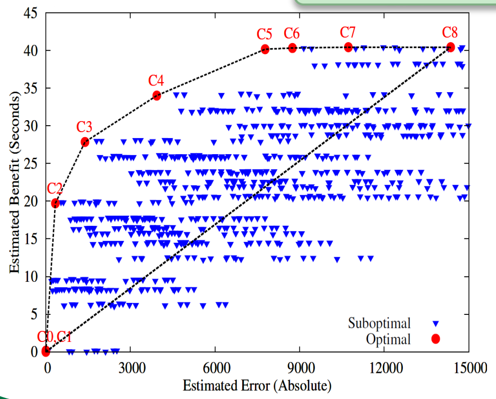
Colorado State University

SAIMI: Separating Algorithm and Implementation via Programming Model Injection

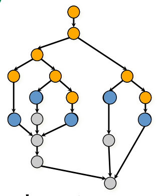
Lookup Table Injectable Programming Model

Developed source to source translation tool called Mesa for applying LUT optimization to scientific codes. Mesa uses ROSE to perform source code analysis and program transformation to achieve up to a 6.8x speedup on set of 5 applications. It provides the user with a set of pareto optimal expression sets that benefit the most from LUT transformation while introducing minimal error.


Original Code
Performance Profiling & Scope Identification
Expression Enumeration & Domain Profiling
Error Analysis & Performance Modeling
Construct & Solve Optimization Problem
Code Generation & Integration
Optimized Code
Performance & Accuracy Evaluation



Libraries for Sparse Tiling



Currently, sparse tiling (also called communication avoiding) is applied to algorithms in an ad hoc fashion. Hand coding these techniques creates a significant barrier to their adoption. Regardless of the specific application, full sparse tiling requires the same building blocks – a **work unit**, e.g. the updating of an atom during one timestep of a simulation or one row of a sparse matrix computation, a **tiler** that aggregates these work items together while respecting intra-tile data dependences, a tile **sequencer** that generates partially ordered task graphs, and an **engine** that efficiently schedules and executes the task graphs using available compute resources. We are developing a C++ library that provides these building blocks, reducing barriers to using FST techniques and increasing efficiency.

Future Directions

We plan incorporate the injectable programming models we are investigating into our existing source-to-source compilation tool and evaluate this approach in the context of more DOE applications.

Participant	**Role**
Michelle Strout	Principal Investigator and developer of sparse polyhedral framework.
Christopher Krieger	Ph.D. student investigating dynamic task graphs.
Andrew Stone	Ph.D. student investigating orthogonal specification of atmosphere grids.
Christopher Wilcox	New Ph.D. who developed source to source look-up table optimizations tool Mesa.
John Dennis	NCAR collaborator on CGPOP mini app. Provided CGPOP vs. POP graph.

U.S. DEPARTMENT OF ENERGY

http://www.cs.colostate.edu/hpc/SAIMI/

Colorado State University