SUPER Performance Autotuning

Khalid Ahmad, Mary Hall, Anand Venkat
University of Utah

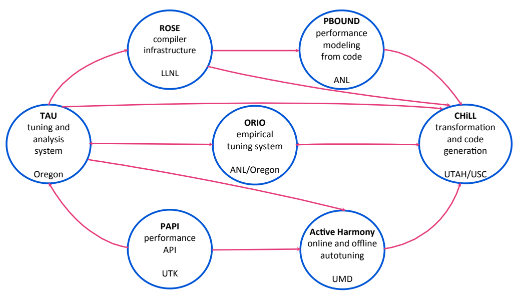
Samuel Williams
Lawrence Berkeley National Laboratory

Ray Chen, Jeffrey Hollingsworth
University of Maryland

SUPER Researchers Develop State-of-the-Art Performance Tools

	Commercial Tools	SUPER Tools
Target Workload	General-purpose workloads (FP↓, CF↑, RE↓, SY↑)	Scientific simulation (FP↑, CF↓, RE↑, SY↓)
Driver	Programmer productivity, primarily	Performance, primarily
Performance Measurement	Focuses on execution time	Extensive, pinpoints opportunities for improvement
Optimization	Conservative (static, architecture independent)	Aggressive (dynamic, autotuning, architecture specific)
Riskiness	Proven Technology	State-of-the-art Technology

Performance Tool Integration

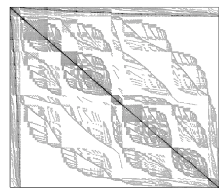


- **Key:** Edge (A->B) implies Tool B uses information from Tool A.
- Tools track architectural changes and support application requirements.

Impact

- ***State-of-the-art tools are applied to SciDAC applications.***
- ***Application development informed by results derived by SUPER tools.***
- ***Tool development informed by application requirements.***

MFDn Application (NUCLEI)



- Many Fermion Dynamics: Nuclear (**MFDn**) is used to calculate the properties of light atomic nuclei.
- Nominally, this requires an eigensolver like **Lanczos** which requires applying the operator repeatedly.
- MFDn forms the very large (**half a billion nonzeros per process**) symmetric configuration interaction matrix explicitly.
- Repeated application of the operator is **bandwidth-intensive**.
- Symmetry further complicates this as it necessitates high performance SpMV and SpMV_T (transpose)

LOBPCG and CSB Representation (2014)

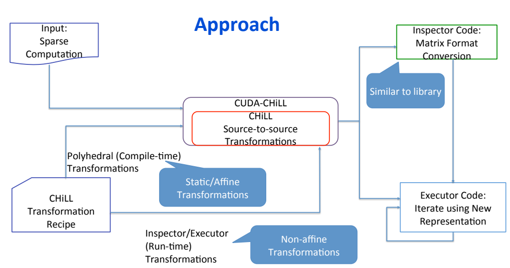
- **LOBPCG is a block eigensolver** that restructures computation into a series of Sparse Matrix-Dense Matrix Multiplications (**SpMM**) where the dense matrix is tall and skinny (500,000 x 16 per process).
- Once again, symmetric necessitates SpMM_T (transpose)
- SpMM reuses matrix, reduces memory bandwidth requirements.
- **SUPER collaborated with FastMath to model, analyze, and optimize these operations.**
- **Optimization Strategy employs Compressed Sparse Block (CSB) matrix representation.**

H. M. Aktulga, A. Buluc, S. Williams, C. Yang, **"Optimizing Sparse Matrix-Multiple Vector Multiplication for Nuclear Configuration Interaction Calculations"**, *International Parallel and Distributed Processing Symposium (IPDPS 2014), May 2014*

Automation in CHiLL (2015)

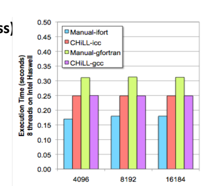
- **GOAL**: Extend CHiLL compiler to perform domain-specific transformations to both sparse matrix representation and associated code.
- Inspector/Executor: Inspector identifies indices of nonzeros and (optionally) derives new matrix representation.
- Incorporates 3 new transformations into CHiLL: *make-dense, compact, compact-and-pad*

Approach



Apply to NUCLEI Computation (Work in Progress)

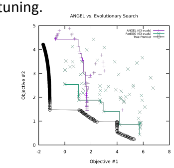
- CHiLL inspector automates conversion to CSB sparse matrix representation
- CHiLL transformations derive optimized SpMV executor for CSB representation (SpMV_T analogous)
- Next step: generate SpMM executor



A. Venkat, M. Hall, M. Strout, **"Loop and Data Transformations for Sparse Matrix Codes", Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation** *(PLDI 2015), June 2015*

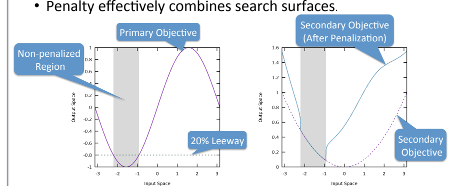
ANGEL: Multi-Objective Autotuning

- Evolutionary algorithms favor exploration.
  - Attempt to approximate the performance "landscape."
- The ANGEL search algorithm favors exploitation.
  - Attempts to move toward the goal with each evaluation.
  - Mitigates the overhead of evaluating a bad configuration.
  - Meets the demands of online autotuning.
- Priority and leeway specify the goal.
  - "Allow a **30%** drop in my **primary** objective in order to improve my **secondary** objective."



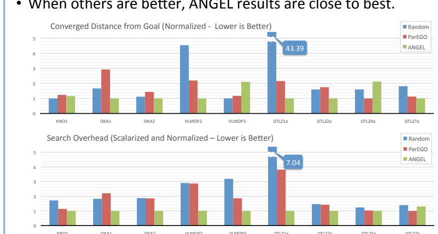
Penalty Function

- ANGEL penalty function considers multiple objectives.
  - Penalty in proportion to leeway violations of higher objectives.
  - Prevents "needle-in-a-haystack" search surfaces.
  - Penalty effectively combines search surfaces.



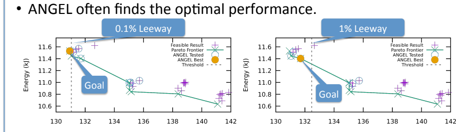
ANGEL: Comparison with Testsuite

- ANGEL almost always performs better than others.
- When others are better, ANGEL results are close to best.

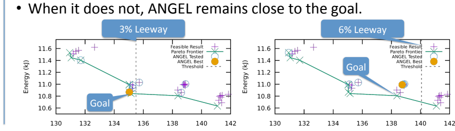


LULESH2 Kernel Time/Energy Results

- ANGEL selects solutions close to, but not violating the leeway.
- As a larger leeway is given, more energy is saved.
- ANGEL often finds the optimal performance.



- When it does not, ANGEL remains close to the goal.



Support for this work was provided through the Scientific Discovery through Advanced Computing (SciDAC) program funded by the U.S. Department of Energy Office of Advanced Scientific Computing Research