





Finding and Optimizing Phases in Parallel Programs

FRSIT

Ray Chen <rchen@cs.umd.edu>

Jeffrey K. Hollingsworth <hollings@cs.umd.edu>









- HPC programs often contain "phases"
 - Dynamic execution context (like a stack trace for performance)
 - Each have distinct performance traits
- Particularly disruptive if inside a timestep loop
 - Short phases confound tools
 - Difficult to analyze a rapidly changing landscape
 - Worse if phases are nested





LULESH2 MPI Call Trace







Automatic Phase Identification

• Prior art (chosen completely at random)







- Key: Automatic identification is hard
 - Rely on experts for annotations



-IPS-2



Guided Phase Identification







Performance Landscape





8/2/16 Find



Cross-Domain Analysis

Finding and Optimizing Phases in Parallel Programs: Scalable Tools Workshop

• Utilize experts during development

My application

has three phases

8/2/16

- Library writers specify tuning variables
- Application writers specify code regions

Phase dictates different performance context

• Even though the same function is being called

MPI performance

I know what

variables affect

FFTW

performance

know what

variables affect

variables affect

BLAS

performance

7



Integration Work

- Special annotation types identify:
 - Tunable variables
 - Code regions that should enable tuning
- New Caliper tuning service
 - Listens for and reacts to special annotations
 - Calls Active Harmony to perform search





3D Fast Fourier Transform

- FFT in 3 dimensions
 - Composed of three 1 dimensional FFT's
 - Data is redistributed among processes between FFT's







Computation/Communication Overlap





FFTz

A2A1 (blocking)



FFTy



A2A2 (blocking)



FFTx







Auto-tuning Opportunities









- Block size during A2A transfer is tunable
 - Relatively independent from other variables
 - May be tuned as a nested sub-phase
- Outer and inner phases run in tandum







Online Auto-Tuning





8/2/16



Offline Auto-Tuning Cost









Online vs. Offline Tuning

- Improvements over offline tuning
 - Nested phases simplifies search complexity
 - Reduce search dimensions from 24 to 16
 - 40% fewer search steps needed to converge
 - Equivalent performance after convergence
- Eliminates need for training runs
 - Don't allocate thousands of nodes to train









- Phases are key for HPC analysis tools
 - Rely on human guidance through annotations
- Annotations unite cross-domain expertise
 - Libraries annotate variables to analyze
 - Application annotate regions to analyze
- Currently analyzing other HPC codes
 - HPGMG has natural phases to exploit
 - AMR codes are next in line

