A. Calotoiu\textsuperscript{1} (GRS), T. Hoefler (ETH), M. Poke (GRS), F. Wolf (GRS)

Using Automated Performance Modeling to Find Scalability Bugs in Complex Codes
Latent Scalability Bugs

System size

Execution time
Analytical performance modeling

Disadvantages
- Time consuming
- Danger of overlooking unscalable code

Identify kernels
- Parts of the program that dominate its performance at larger scales
- Identified via small-scale tests and intuition

Create models
- Laborious process
- Still confined to a small community of skilled experts
Scalability bug detector

main() {
    foo()
    bar()
    compute()
}

Instrumentation
- All functions

Performance measurements (profiles)
- $p_1 = 128$
- $p_2 = 256$
- $p_3 = 512$
- $p_4 = 1,024$
- $p_5 = 2,048$
- $p_6 = 4,096$

Ranking:
- Asymptotic
- Target scale $p_t$

Input

Output
Primary focusing on scaling trend

![Graph showing scaling trend]

**Ranking:**

1. $F_2$
2. $F_1$
3. $F_3$
Primary focusing on scaling trend

![Graph showing scaling trend with ranking]

**Ranking:**

1. $F_2$
2. $F_1$
3. $F_3$
Primary focusing on scaling trend

Ranking:

1. $F_2$
2. $F_1$
3. $F_3$
Model building blocks

Computation

LU
\[ t(p) \sim c \]

FFT
\[ t(p) \sim \log_2(p) \]

Naïve N-body
\[ t(p) \sim p \]

Samplesort
\[ t(p) \sim p^2 \log_2(p) \]

... 
\[ t(p) \sim ... \]

Communication

LU
\[ t(p) \sim c \]

FFT
\[ t(p) \sim c \]

Naïve N-body
\[ t(p) \sim p \]

Samplesort
\[ t(p) \sim p^2 \]

... 
\[ t(p) \sim ... \]
Performance model normal form

\[ f(p) = \sum_{k=1}^{n} c_k \times p^{i_k} \times \log^j_k(p) \]

\( n = 1 \)
\( I = \{0,1,2\} \)
\( J = \{0,1\} \)
Performance model normal form

\[ f(p) = \sum_{k=1}^{n} c_k \times p^{i_k} \times \log_{2}^{j_k}(p) \]

\( n = 2 \)
\( I = \{0, 1, 2\} \)
\( J = \{0, 1\} \)
Requirements modeling

Program

Computation
- FLOPS
- ... (Load)
- Store

Communication
- P2P
- Collective

Disagreement may be indicative of wait states

Time
Workflow

- Statistical quality control
- Kernel refinement
- Performance measurements
- Performance profiles
- Model generation
- Scaling models
- Performance extrapolation
- Ranking of kernels
- Model generation
- Scaling models
- Accuracy saturated?
  - Yes
  - No
- Model refinement
**Model refinement**

- **Input data**
  
  \[ n = 1; R_0 = \]

- **Hypothesis generation; hypothesis size \( n \)**
  
  \( c_1 \times p \)

- **Hypothesis evaluation via cross-validation**
  
  \( c_1 \times p^2 \times \log(p) \)

- **Computation of \( R_n^2 \) for best hypothesis**
  
  \[ R_n^2 = 1 - \frac{\text{residualSumSquares}}{\text{totalSumSquares}} \]

  \[ R_n^2 = 1 - (1 - R^2) \cdot \frac{n}{6-n-1} \]

- **Scaling model**
  
  \[ I = \{0,1,2\}; J = \{0,1\}; n_{\text{max}} = 2 \]
Evaluation

\[
I = \{ \frac{0}{2}, \frac{1}{2}, \frac{2}{2}, \frac{3}{2}, \frac{4}{2}, \frac{5}{2}, \frac{6}{2} \}
\]

\[
J = \{0, 1, 2\}
\]

\[n = 5\]
Sweep3D

Solves neutron transport problem
- 3D domain mapped onto 2D process grid
- Parallelism achieved through pipelined wave-front process

LogGP model for communication developed by [1].

\[
t_{\text{comm}} = \left[ 2(p_x + p_y) - 2 \right] + 4(n_{\text{sweep}} - 1) \times t_{\text{msg}}
\]

\[
t_{\text{comm}} = c \times \sqrt{p}
\]

[1] Hoisie et al.: Performance analysis of wavefront algorithms on very-large scale distributed systems; Workshop on Wide Area Networks and High Performance Computing, 1999
Sweep3D (2)

<table>
<thead>
<tr>
<th>Kernel [2 of 40]</th>
<th>Runtime[%]</th>
<th>Model [s]</th>
<th>Predictive error [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>sweep-&gt;MPI_Recv</td>
<td>65.35</td>
<td>4.03\sqrt{p}</td>
<td>5.10</td>
</tr>
<tr>
<td>sweep</td>
<td>20.87</td>
<td>582.19</td>
<td>0.01</td>
</tr>
</tbody>
</table>
MILC

MILC/su3_rmd – code from MILC suite of QCD codes with performance model manually created by [2].

- Time per process should remain constant except for a rather small logarithmic term caused by global convergence checks

<table>
<thead>
<tr>
<th>Kernel [3 of 479]</th>
<th>Model [s] ( t = f(p) )</th>
<th>Predictive Error [%] ( p_t = 64k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>compute_gen_staple_field</td>
<td>( 2.40 \times 10^2 )</td>
<td>0.43</td>
</tr>
<tr>
<td>g_vecdoublesum( \triangleright )MPI_Allreduce</td>
<td>( 6.30 \times 10^{-6} \times \log_2(p) )</td>
<td>0.01</td>
</tr>
<tr>
<td>mult_adj_su3_fieldlink_lathwec</td>
<td>( 3.80 \times 10^3 )</td>
<td>0.04</td>
</tr>
</tbody>
</table>

\[ P_i \leq 16k \]

HOMME

Core of the Community Atmospheric Model (CAM)
- Spectral element dynamical core
  on a cubed sphere grid

<table>
<thead>
<tr>
<th>Kernel [3 of 194]</th>
<th>Model [s] t = f(p)</th>
<th>Predictive error [%] p_t = 130k</th>
</tr>
</thead>
<tbody>
<tr>
<td>Box_rearrange-&gt;MPI_Reduce</td>
<td>$3.63 \times 10^{-6} p \times \sqrt{p} + 7.21 \times 10^{-13} p^3$</td>
<td>30.34</td>
</tr>
<tr>
<td>Vlaplace_sphere_vk</td>
<td>$24.44 + 2.26 \times 10^{-7} p^2$</td>
<td>4.28</td>
</tr>
<tr>
<td>Compute_and_apply_rhs</td>
<td>49.09</td>
<td>0.83</td>
</tr>
</tbody>
</table>

$$P_i \quad 43k$$

The G8 Research Councils Initiative on Multilateral Research Funding
Interdisciplinary Program on Application Software towards Exascale Computing for Global Scale Issues
HOMME (2)
Mass-producing performance models

- Is feasible
- Offers insight
- Requires low effort
- Improves code coverage

Future work
- Integration into Scalasca
- Strong scaling
- Asymptotic requirements characterization
Acknowledgements

- John Dennis and Rich Loft
  National Center For Atmospheric Research
- Marc-André Hermanns
  German Research School for Simulation Sciences
Cost of first prediction

Assumptions

- Input experiments at scales \( \{2^0, 2^1, 2^2, \ldots, 2^m\} \)
- Target scale at \( 2^{m+k} \)
- Application scales perfectly

<table>
<thead>
<tr>
<th>k</th>
<th>Full scale [%]</th>
<th>Input [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100</td>
<td>&lt;100</td>
</tr>
<tr>
<td>2</td>
<td>100</td>
<td>&lt;50</td>
</tr>
<tr>
<td>3</td>
<td>100</td>
<td>&lt;25</td>
</tr>
<tr>
<td>4</td>
<td>100</td>
<td>&lt;12.5</td>
</tr>
</tbody>
</table>

Jitter may require more experiments per input scale, but to be conclusive experiments at the target scale would have to be repeated as well.