Performance Analysis
Scalability on PajeNG

Prof. Lucas M. Schnorr
INF/UFRGS, GPPD, Porto Alegre
schnorr@inf.ufrgs.br

Fourth Brazil-France Workshop (CNPq – Inria)
Gramado, September 16th, 2014
Context

- Large computational systems
  - High performance computing
  - Large distributed systems

- Large parallel and distributed applications
  - In space: thousands (or even millions) of processes
  - In time: many events
Large computational systems

Milkyway-2, 3.12 M cores

Tesla K40, 2888 cores

Xeon Phi, 61 cores

MPPA, 256 cores
Performance analysis

- Run applications faster considering platform constraints
- Performance analysis cycle (*post-mortem*)
- Reproducibility (Provenance)
- Clock synchronisation
- Indeterminism
- Intrusion
- Strategies for behavior data collection
  - Sampling, Profiling
  - Tracing (important events registered during runtime)
Challenges and motivation

- Large-scale applications
  - Supercomputers with more than 3 millions cores
  - Exascale expectation \(\rightarrow\) billions of cores
- Low-intrusion tracing techniques
  - Buffering, hardware support
Space/Time trace size explosion

- **Ondes3D**: Seismic wave propagation in 3D
  - 32p, 50s run, 100K events
- **LU.A.32**: Lower-upper gauss-seidel solver
  - 32p, 4.79s run, about 7 million events (142 Mbytes raw)
- **Naïve Particle Simulator**: (BSP-based) quadratic impl.
  - 32p, 6.26s run, about 200 million events (2.5 Gbytes)

- “Big Data” problem
How to extract useful knowledge from traces?

- Combination of
  - Scalable analysis software system
  - Well-defined analysis methods

- PajeNG
Outline

Context and motivation

What is PajeNG?

Technical efforts

Analysis methods

Intl. Cooperation

Conclusion
What is PajeNG?

- **PajeNG → Paje Next Generation**
  - Complete re-write in C++ of the original Paje
  - Free software, GPL’ed, component-based implementation
  - Available at [http://github.com/schnorr/pajeng/](http://github.com/schnorr/pajeng/)

- Generic framework for performance analysis
  - Open and extensible trace file format

- Features and components
  - Trace event simulator (PajeSimulator)
  - Space/Time visualization tool
  - Unix-like tools (dump, extract, export)
Research general overview (around PajeNG)

- Technical
  - Binary file format
  - Parallelization and distribution

- Analysis methods
  - Spatio/Temporal aggregation
  - Trace visualization
  - Trace comparison
  - Automatic analysis
PajeNG
Technical efforts
Binary file format  (by undergrad Vinicius H.)

- Original Paje file format is textual
  - Very large trace files
    - Particle Simulator – 7 secs – 8.8 GBytes
    - NAS.CG.A.64 – 20 secs – 2 GBytes
    - NAS.LU.B.64 – 310 secs – 750 Mbytes
  - Takes a lot of time to read and parse

Define a binary file format

- GNU Flex/Bison, librastro
- Improve the reading of large trace files
Parallel and distributed PajeNG (By grad Jonas K.)

- Paje simulator recreates the parallel application state
  - Takes each registered event and replay its effect
- **Problem**: simulator is sequential, not scalable
  - Single trace file as input
  - Events are simulated one by one

**Distribute PajeNG**

- Use the distributed platform for the analysis
PajeNG
Analysis methods
Trace comparison (by undergrad Alef F.)

- Performance optimization cycle
  - Execution → perf. analysis → optimization → · · ·
  - Compare traces from different runs is useful
    - Confirm if an optimization is effective

- Related work
  - DNA alignment, process to process (Vampir)
  - Lack of global comparison and visualization

Propose a global trace comparison methodology

- Should we use the same bio-inspired algorithm?
  - Original Needleman–Wunsch algorithm?
  - Optimization in the form of the Hirschberg’s algorithm

- Propose a global comparison algorithm
- Diff visualization
Trace visualization

- Create a visual representation of the traces
  - Interactive investigation

- Traditional technique: a space/time representation
  - Vertical axis → Processes (the observed entities)
  - Horizontal axis → Time (their behavior along time)
  - Causality check
Space/Time view - The Sweep3D case-study

- **Sweep3D**
  - "It solves a 1-group time-independent discrete ordinates (Sn) 3D cartesian (XYZ) geometry neutron transport problem."
  - Very small messages, small states (millions of them)
Sweep3D – maximum zoom
Sweep3D – zooming out 1
Sweep3D – zooming out 2
Sweep3D – zooming out 3 and right shift
Sweep3D – full execution

- Observe the synthetic perturbation

Problems with the space/time view

- Scales badly
  - horizontal *versus* vertical
- Platform topology?
  - It might explain a lot of application behavior
Squarified treemap view

- Observe outliers, differences of behavior
- Hierarchical aggregation

B  Hierarchy: Site (10) - Cluster (10) - **Machine** (10) - Processor (100)
Hierarchical graph view

- Correlate application behavior to network topology
- Pin-point resource contention
  - Grid5000 platform topology, application on top
Temporal aggreg. evaluation (by PhD student Damien D.)

- Aggregation is a possible solution to scale the analysis
  - May mislead the analysis → smooth or hide behavior

Using entropy to evaluate **temporal** aggregation

- Kullback-Leibler divergence
- Shannon entropy
- Example of NAS CG A 64
Spatio/Temporal aggr. eval. (by PhD student Damien D.)

- Analyst looks for a tradeoff between
  - Information loss
  - Complexity reduction

### Spatio/Temporal aggregation evaluation

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **a.** The microscopic model (12 resources and 20 microscopic time periods)
- **b.** A non-optimal spatial and temporal aggregation (3 clusters and 4 time periods)
- **c.** An optimal spatial and temporal aggregation (3 clusters and 6 time periods)
- **d.** An optimal spatiotemporal aggregation (56 spatiotemporal areas)
- **e.** A higher-level spatiotemporal aggregation (15 spatiotemporal areas)
- **f.** Visual aggregation of 3.d (21 data aggregates and 7 visual aggregates)
The Ocelotl tool (by PhD student Damien D.)
Automatic performance analysis (by grad Flavio A.)

Hang on
See Flavio’s presentation in a moment
International cooperation

- Software panorama

- Scientific context
  - ANR USS-SIMGRID and INFRA-SONGS projects
  - LICIA Laboratory (INF/UFRGS with CNRS-LIG)
  - ExaSE FAPERGS-Inria Equipe Associée (Mescal/Moais)
Conclusion

- Performance analysis scalability
  - Very hard to obtain
    - Technical
    - Theoretical
  - Multi-technique strategy, complementary

- Many cooperation possibilities
- Different scenarios
  - Application
  - Performance analysis
Thank you for your attention

▶ A Spatiotemporal Data Aggregation Technique for Performance Analysis of Large-scale Execution Traces. Damien Dosimont, Robin Larmarche-Perrin, Lucas Mello Schnorr, Guillaume Huard, Jean-Marc Vincent. Accepted for IEEE Cluster 2014.

http://www.inf.ufrgs.br/~schnorr/
schnorr@inf.ufrgs.br