Benchmark Precision and Random Initial State

Tomas Kalibera, Lubomir Bulej, Petr Tuma

DISTRIBUTED SYSTEMS RESEARCH GROUP
http://nenya.ms.mff.cuni.cz

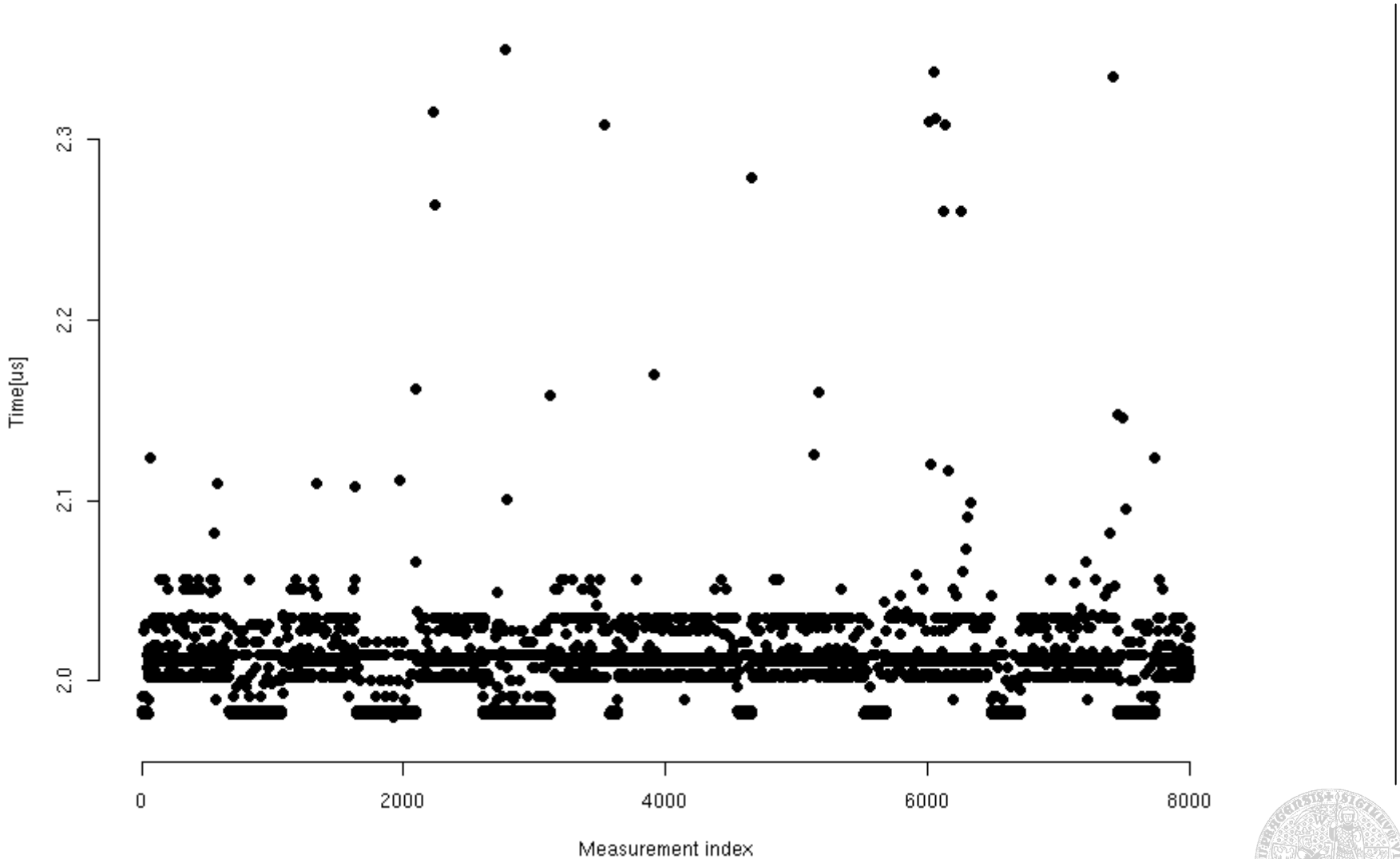
CHARLES UNIVERSITY PRAGUE
Faculty of Mathematics and Physics

Goal: Tool for improving software performance.

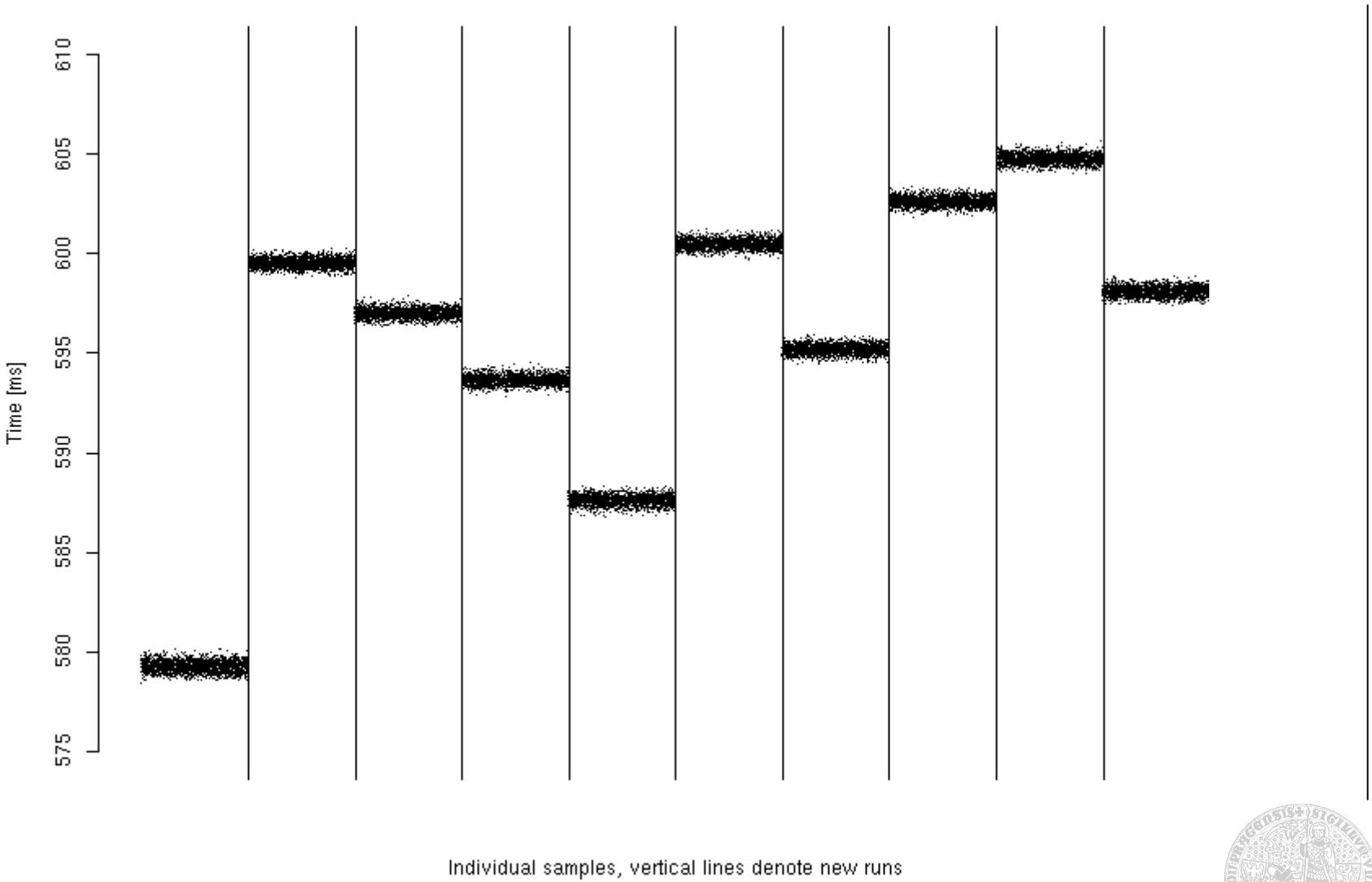
- Regular automatic benchmarking
  - Incorporate into regression testing
- Automated detection of regressions
  - Detect changes in benchmark results
- Fixing important regressions
  - Automatically find suspect modifications
  - (Manually) fix regressions if possible

http://nenya.ms.mff.cuni.cz/projects/mono
Proceedings: pg. 853

Benchmarks are unstable.



Benchmark results differ in each execution.



Random state is integral part of real systems.

- Differences in results from different executions cannot be removed by
  - Shutting down non-related services
  - Disconnecting network, unloading drivers
  - Turning off randomization of virtual addresses
  - Rebooting before each benchmark execution
  - Excessively long warm-up phase in each execution

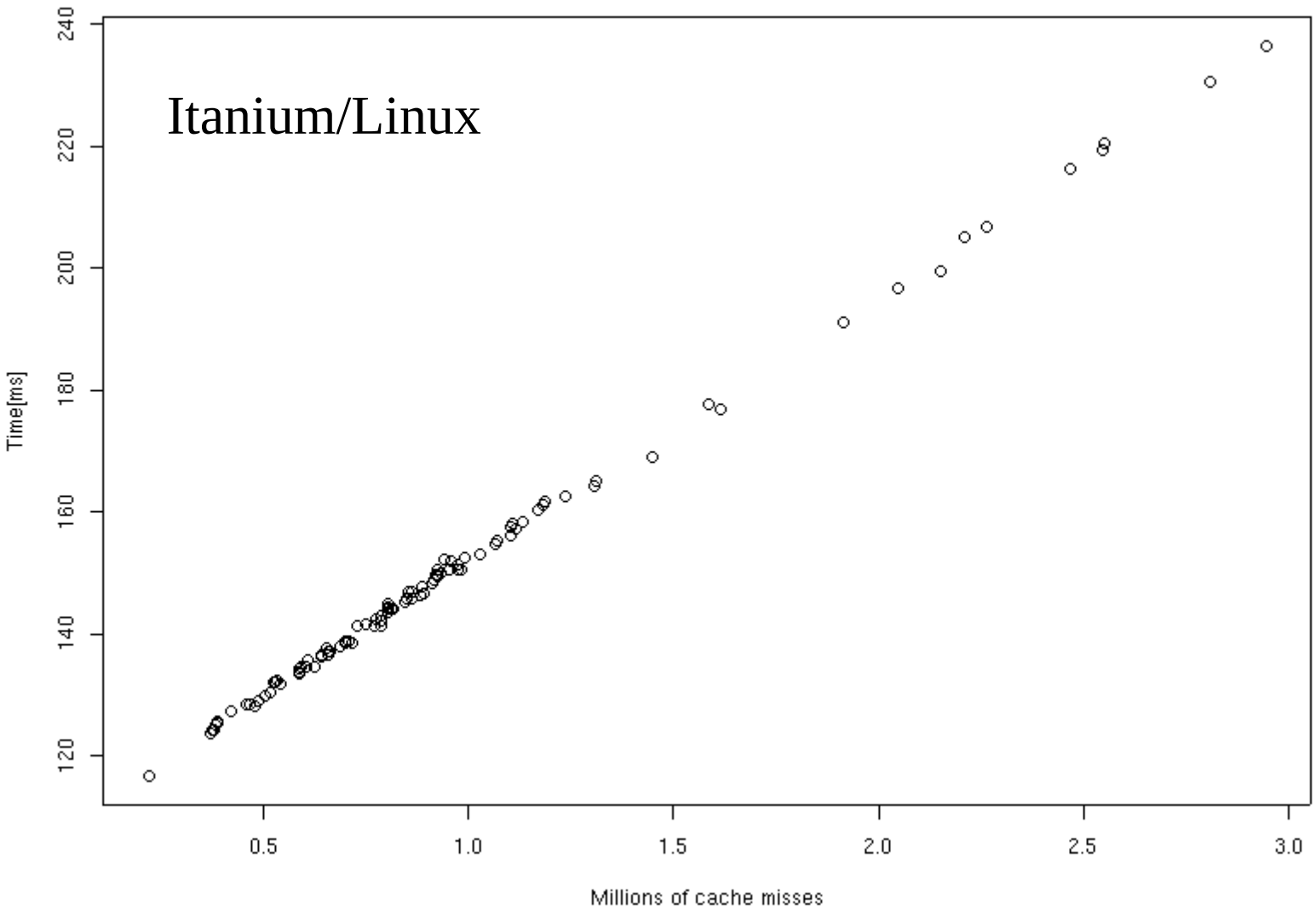
The problem can be quantified.

- Impact factor of random initial state
  - Robust to non-normality, outliers
  - Calculated from benchmark results by simple statistical simulation
- Defined as ratio of variability in data from different runs to variability in data from the same run
  - Values ≥1, 1 means no impact

Impact of random state is system dependent.

Benchmark	Platform	Impact Factor
FFT	Pentium/Windows	94.74
FFT	Itanium/Linux	35.91
FFT	Pentium/Linux	25.81
FFT	Pentium/DOS	1.06
RPC Marshaling	Pentium/Linux	2.61
RPC Ping	Pentium/Linux	1.10
RUBiS	Pentium/Linux	1.01

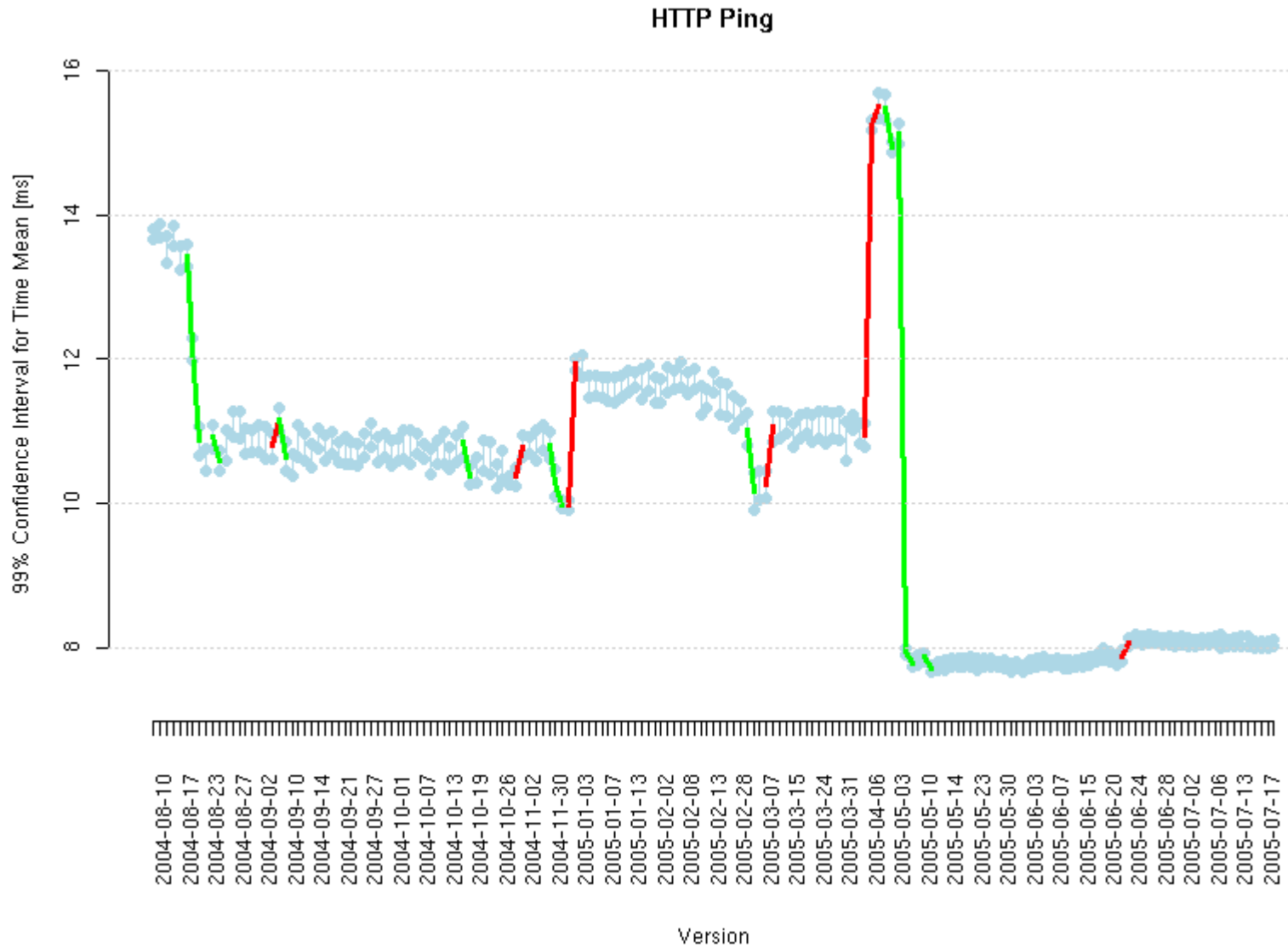
Differences in results are due to cache misses.



Conclusion: Benchmarking is still possible.

- Random initial state is a reality
- Implications for benchmarking
  - Need to run more times, possibly re-compile
  - Non-trivial statistical evaluation required
- Current status
  - Simple hierarchical model
  - Allows precision estimation, experiment planning
  - http://nenya.ms.mff.cuni.cz/benchmark

Mono Regression Benchmarking Project



Regression benchmarking publications

- Kalibera, T., Bulej, L., Tuma, P.: ***Quality Assurance in Performance: Evaluating Mono Benchmark Results***, accepted as a full paper on Second International Workshop on Software Quality (SOQUA 2005), Erfurt, Germany
- Kalibera, T., Bulej, L., Tuma, P.: ***Automated Detection of Performance Regressions: The Mono Experience***, accepted as a full paper on 13th Annual Meeting of the IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems (MASCOTS 2005), Atlanta, GA, USA
- Bulej, L., Kalibera, T., Tuma, P.: ***Repeated Results Analysis for Middleware Regression Benchmarking***, Performance Evaluation: An International Journal, Performance Modeling and Evaluation of High-Performance Parallel and Distributed Systems, Elsevier, 2005
- Bulej, L., Kalibera, T., Tuma, P.: ***Regression Benchmarking with Simple Middleware Benchmarks***, proceedings of IPCCC 2004 Mid-dleware Performance Workshop, IEEE 2004