Evaluation of Parallel Application-Level Behavioral Attributes

ICS - First International Workshop on Characterizing Applications for Heterogeneous Exascale Systems *CACHES 2011*

Jeffrey J. Evans and Charles E. Lucas

Purdue University
PC Krause and Associates, Inc.

June 4, 2011

Agenda

1. Motivation
   - Executive Order 13423
   - Gaps in our Understanding
2. Technical Approach
3. Experiments
   - Attribute Evaluation
   - Verification
   - Surprises
4. Concluding Remarks

PURDUE UNIVERSITY

PCKA

Table of Contents

Systemic Performance Management

- Improve *systemic performance consistency*
  - **Goal:** Minimize application run time *extension* and *variability*
- Prior work on the T_{comm} component of parallel applications
  - Application level perspective
  - Monolithic (single processor, single core) compute nodes
  - Message preparation, Congestion, Errors
  - Methods and metrics to quantify application *run time sensitivity* to network performance variation
- Nature of communication has changed
  - Multi-core complicates things (hybrid DMP, SMP)
- Q: Do the models and metrics still hold?

PURDUE UNIVERSITY

PCKA

Energy Management

- Improve energy efficiency
- Reduce greenhouse gases
- Reduction in energy intensity by 3 percent/yr or
- 30% by 2015 based on 2003 baseline
- Energy intensity = consumption per sq ft building space
- Q: Can better performance management improve energy efficiency?

Research Opportunity

- Combine systemic performance and energy management
  - Use application behavioral performance data
  - Develop standard interfaces for HPC/data center management
- Challenges
  - Conflicting subsystem operational objectives
  - Subsystem interactions can trigger autonomous subsystem behavior
  - Cost of state transition
  - Centralized control of semi-autonomous subsystems
  - Need to understand subtleties at core, processor, node, system levels
  - OS, libraries, applications may not behave as assumed

Table of Contents

Application-Level Behavioral Attributes

- Describe run time performance affects due to subsystem interactions
- Major Subsystems
  - Resource Manager, Scheduler
  - Communication Subsystem
  - Set of executing applications

Application-Level Behavioral Attributes

- Describe run time performance affects due to subsystem interactions
- Major Subsystems
  - Resource Manager, Scheduler
  - Communication Subsystem
  - Set of executing applications
- Resource advice tuple

$$R = \begin{cases} \{S, D, E, \ldots\} & : \quad S = \{1 \ldots\} \\ & : \quad D = \{1 \ldots\} \\ & : \quad E = \{1 \ldots\} \\ & : \quad \ldots = TBD \end{cases}$$

Emulating Parallel Applications - PACE

$$T_{app} = \sum_{1}^{n} T_{cycle}, \quad T_{cycle} = T_{comm} + T_{comm}$$

$$T_{comm} = \alpha + \beta n, \quad T_{comp} = \frac{T_{comm} \cdot 100}{L} - T_{comm}$$

- Emulate parallel simulation (cycles or run time) PACE
  - Simulated T_{comp}, Real T_{comm}
  - Measure T_{comm} parameters just prior to run
  - Determine T_{comp} from T_{comm} and load L
  - Determine T_{run} from T_{comm}, T_{comp}, and cycles or
  - Determine cycles from T_{comm}, T_{comp}, and T_{run}

Attribute Evaluation - PARSE

- Parallel Application Run time Sensitivity Evaluation - PARSE
- Python wrapper
- PARSE Manages
  - Process allocation on processors - depreciated
  - Synchronization of PACE & App. Under Test (AUT) execution
  - Progression of attribute evaluation tests
- PARSE Performs
  - Post processing
    - Sensitivity
    - Disruptiveness
    - Intermediate values and supplementary statistics

PURDUE UNIVERSITY

PCKA

Quantifying AUT Sensitivity

- Compare baseline vs. perturbed run times
- Perturb AUT using PACE ⇒ measure AUT runtime
- Combine calculated coefficients of mean and variation between operational scenarios

$$COM_{ij} = \left(\frac{\left(\frac{\sum^n T_{run}}{n} \right)_i}{\left(\frac{\sum^n T_{run}}{n} \right)_j} \right), \quad COV_k = \left(\frac{\sigma_k}{\bar{x}_k} \cdot 100 \right)$$

$$ROV_{ij} = \left(\frac{COV_i}{COV_j} \right)$$

$$S_{ij} = COM_{ij} \cdot ROV_{ij}$$

$$\lim_{P[d] \to 0} S_{i0} = \lim_{i \to 0} S_{00} = COM_{00} \cdot ROV_{00} = 1.$$

Quantifying AUT Disruptiveness

- Compare baseline vs. perturbed run times
- Perturb PACE using AUT $\Rightarrow$ measure PACE runtime
- Combine calculated coefficients of mean and variation between operational scenarios

$$COM_{ij} = \left(\frac{\left(\frac{\sum^n T_{run}}{n} \right)_i}{\left(\frac{\sum^n T_{run}}{n} \right)_j} \right), \quad COV_k = \left(\frac{\sigma_k}{\bar{x}_k} \cdot 100 \right)$$

$$ROV_{ij} = \left(\frac{COV_i}{COV_j} \right)$$

$$D_{ij} = COM_{ij} \cdot ROV_{ij}$$

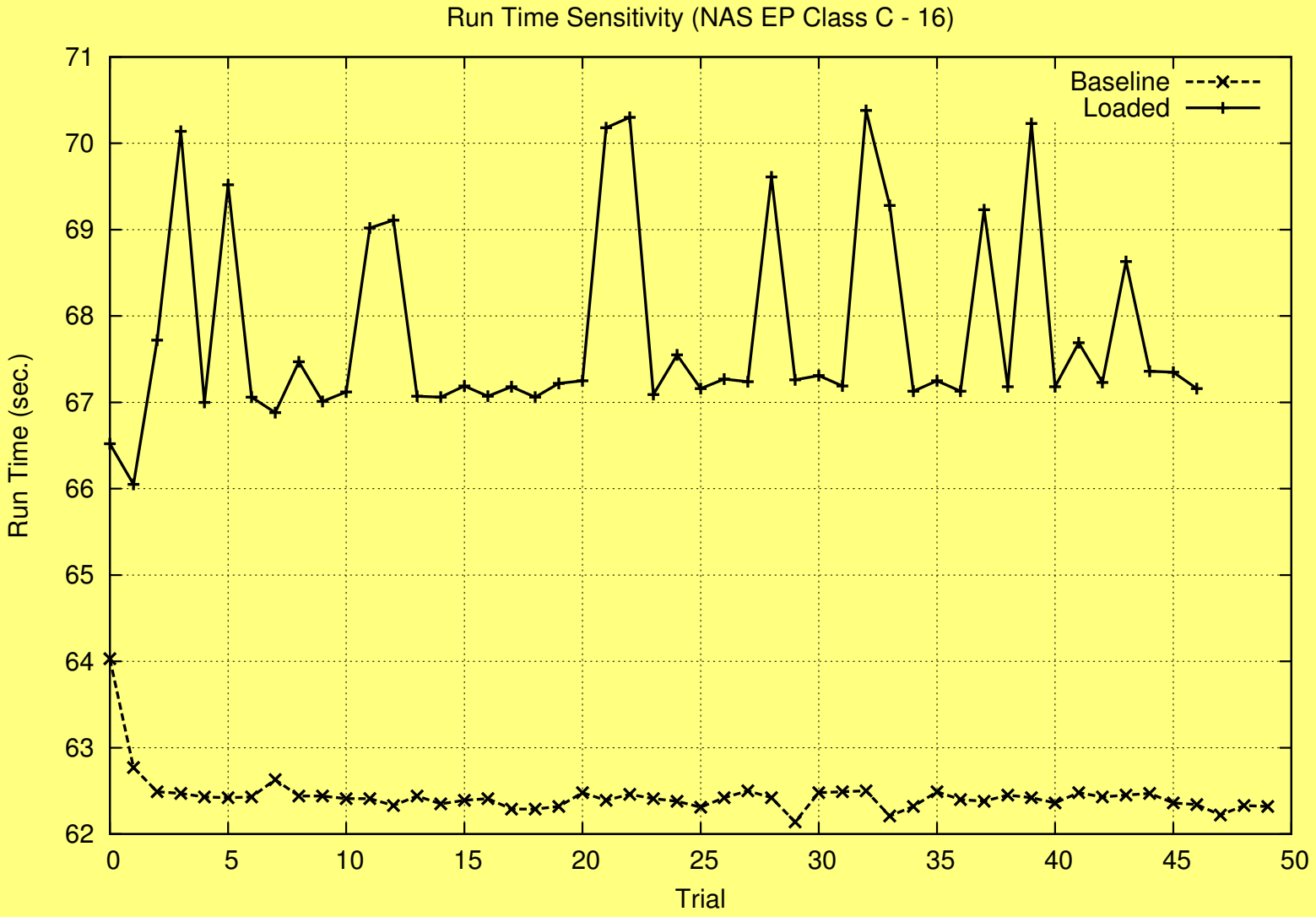
$$\lim_{P[d] \to 0} D_{i0} = \lim_{i \to 0} D_{00} = COM_{00} \cdot ROV_{00} = 1.$$

Table of Contents

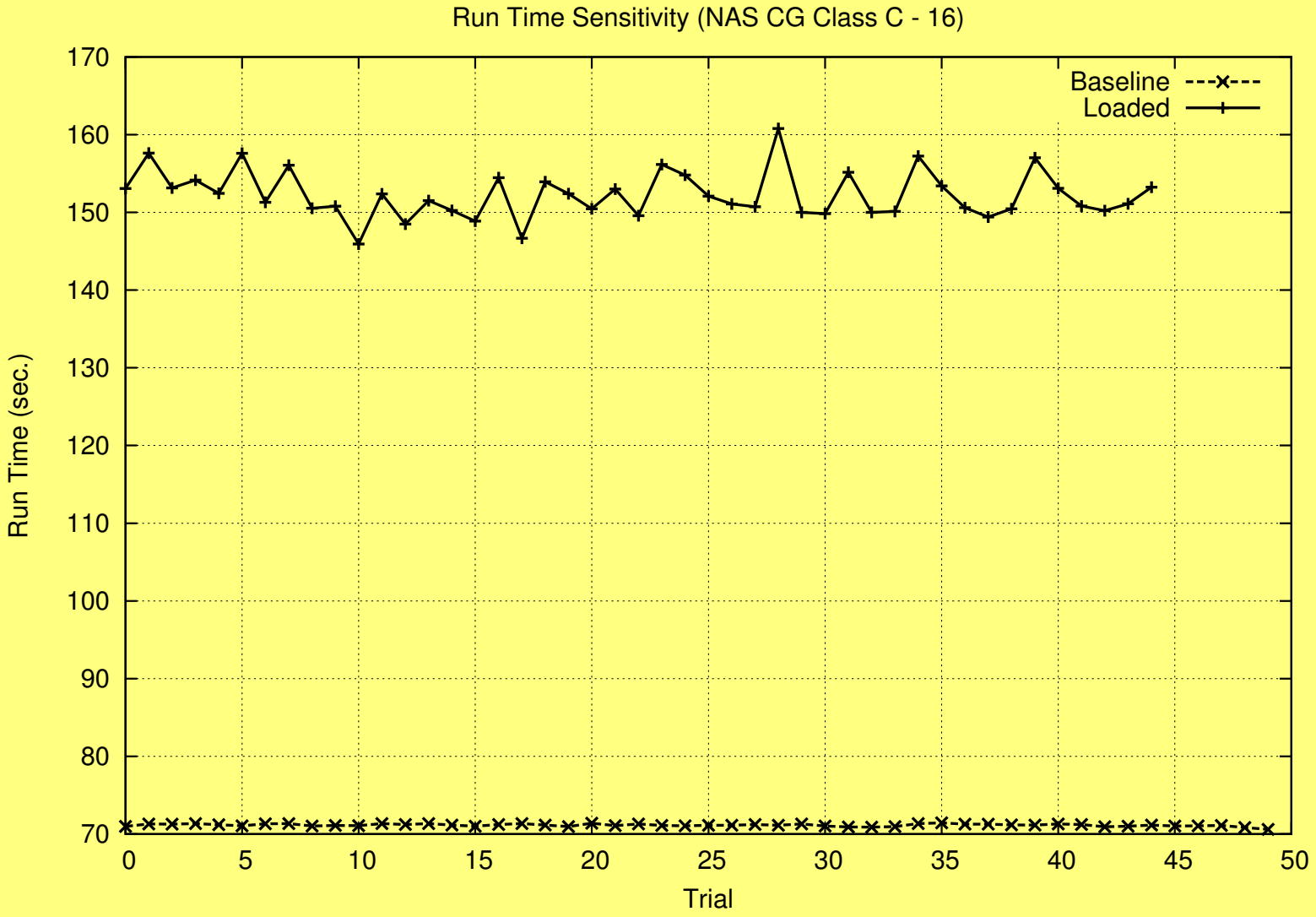
Platform

- 10 node, 80-core HP Proliant DL165 G5p Cluster
  - 2 Quad-core AMD Opteron 2384 (Shanghai) per node
  - 16GB RAM, 500GB HD
  - 2 1000Mbps Ethernet
  - Red Hat Enterprise Linux 5.4 (2.6.18-194.26.1.el5)
  - gcc 4.4.0, PGI, Intel compilers
  - MPICH2 communication libraries (1.2.1p1, 1.3b1)

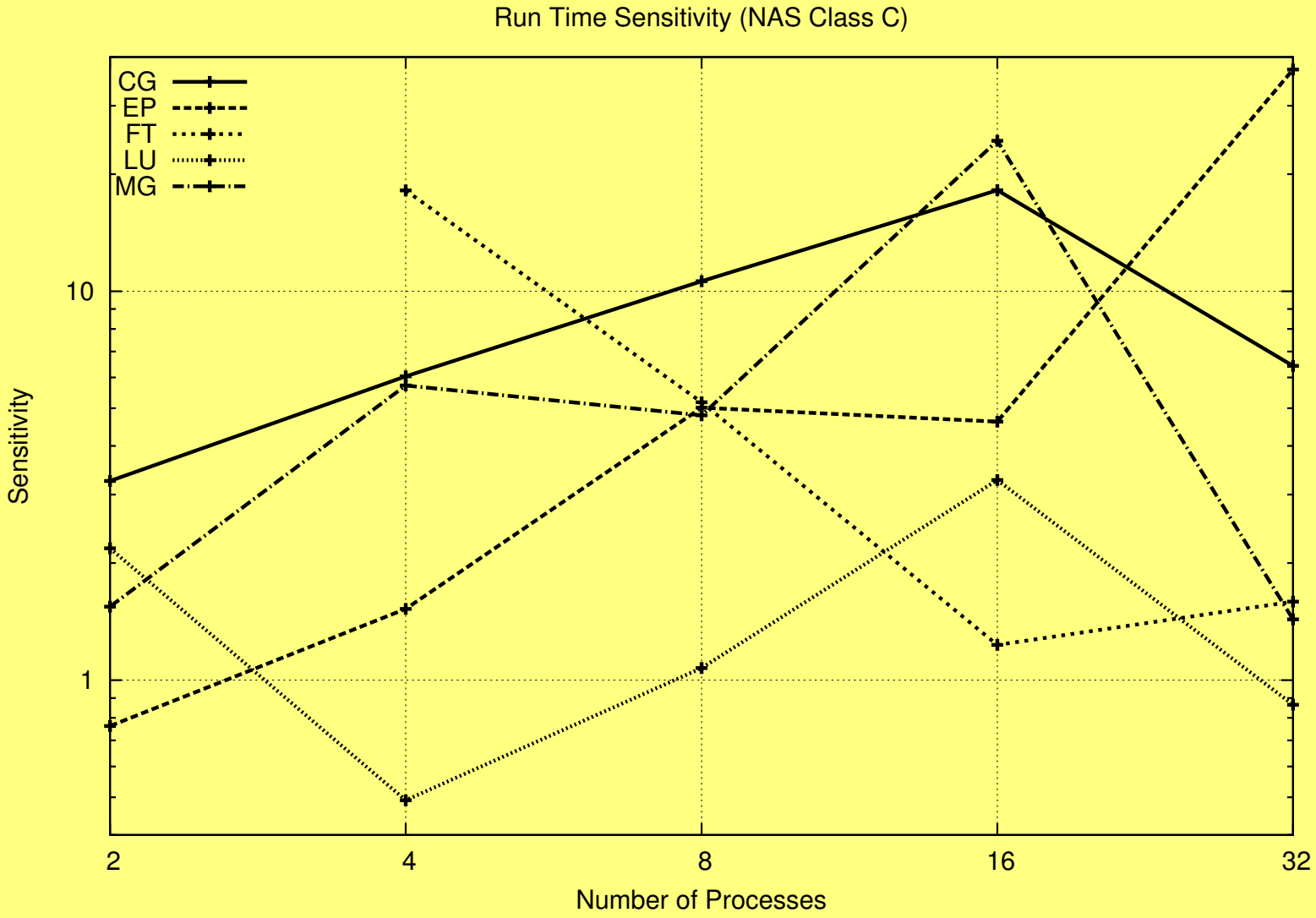
Sensitivity Trials



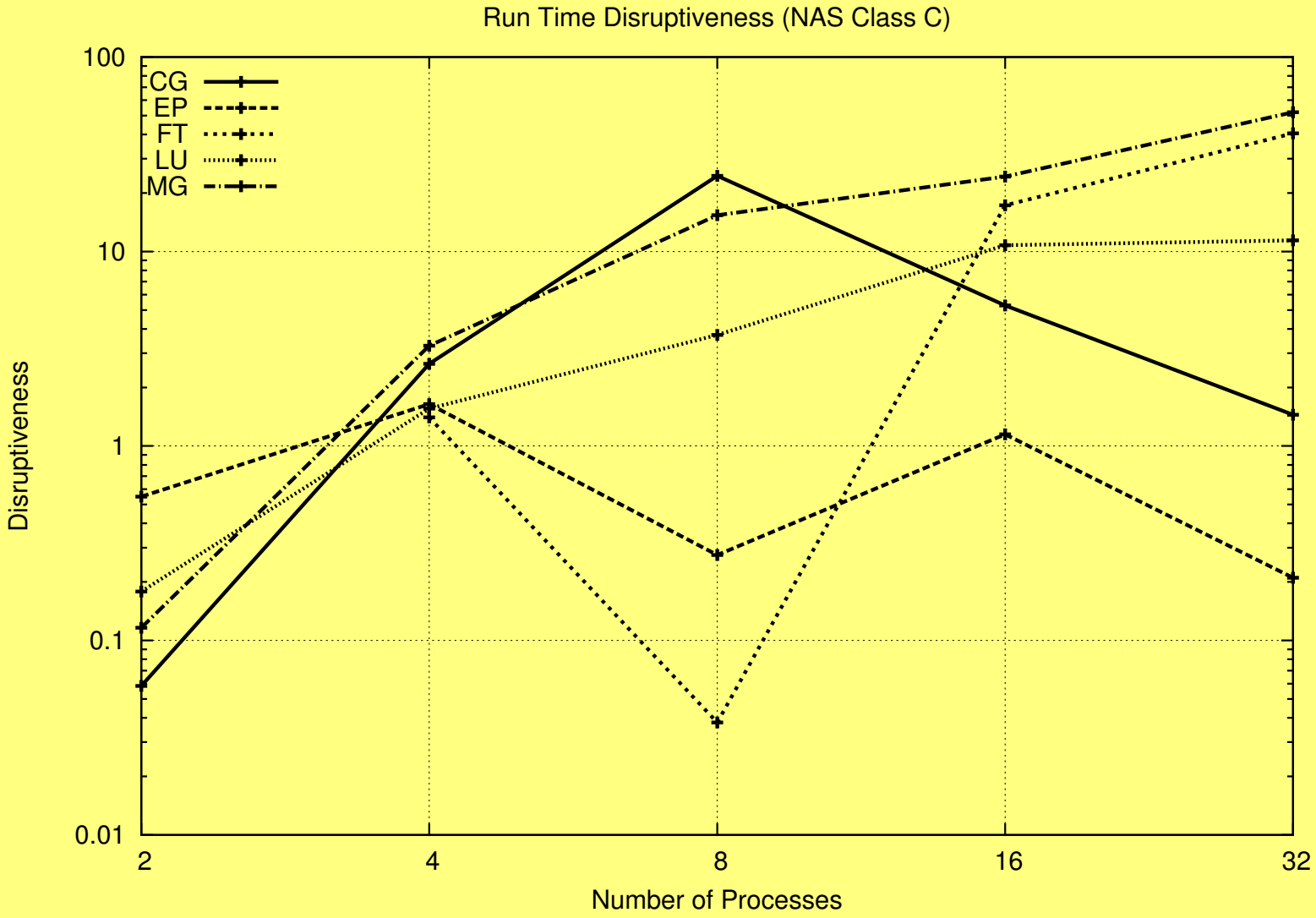
Sensitivity Trials



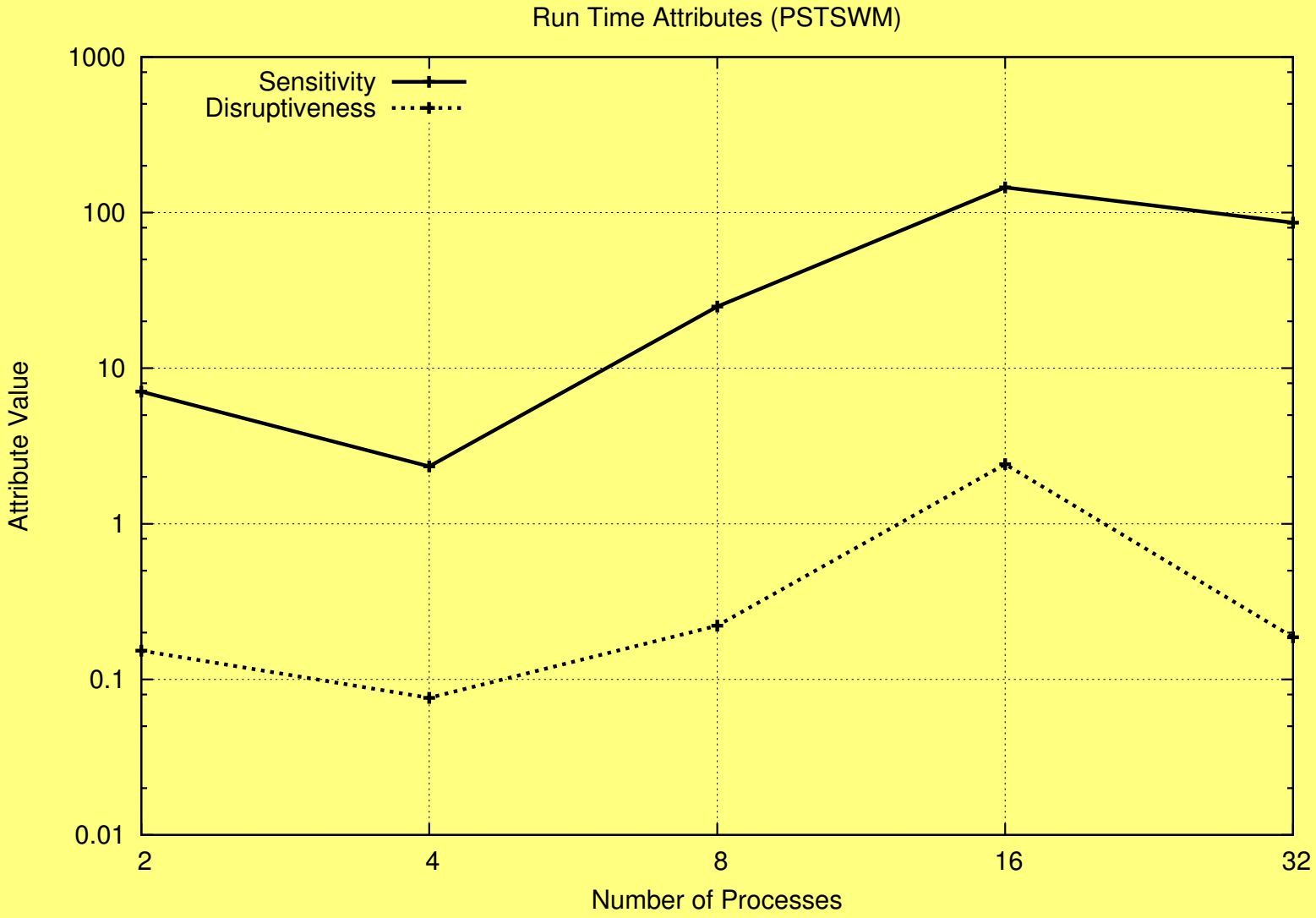
NAS Class C Attributes



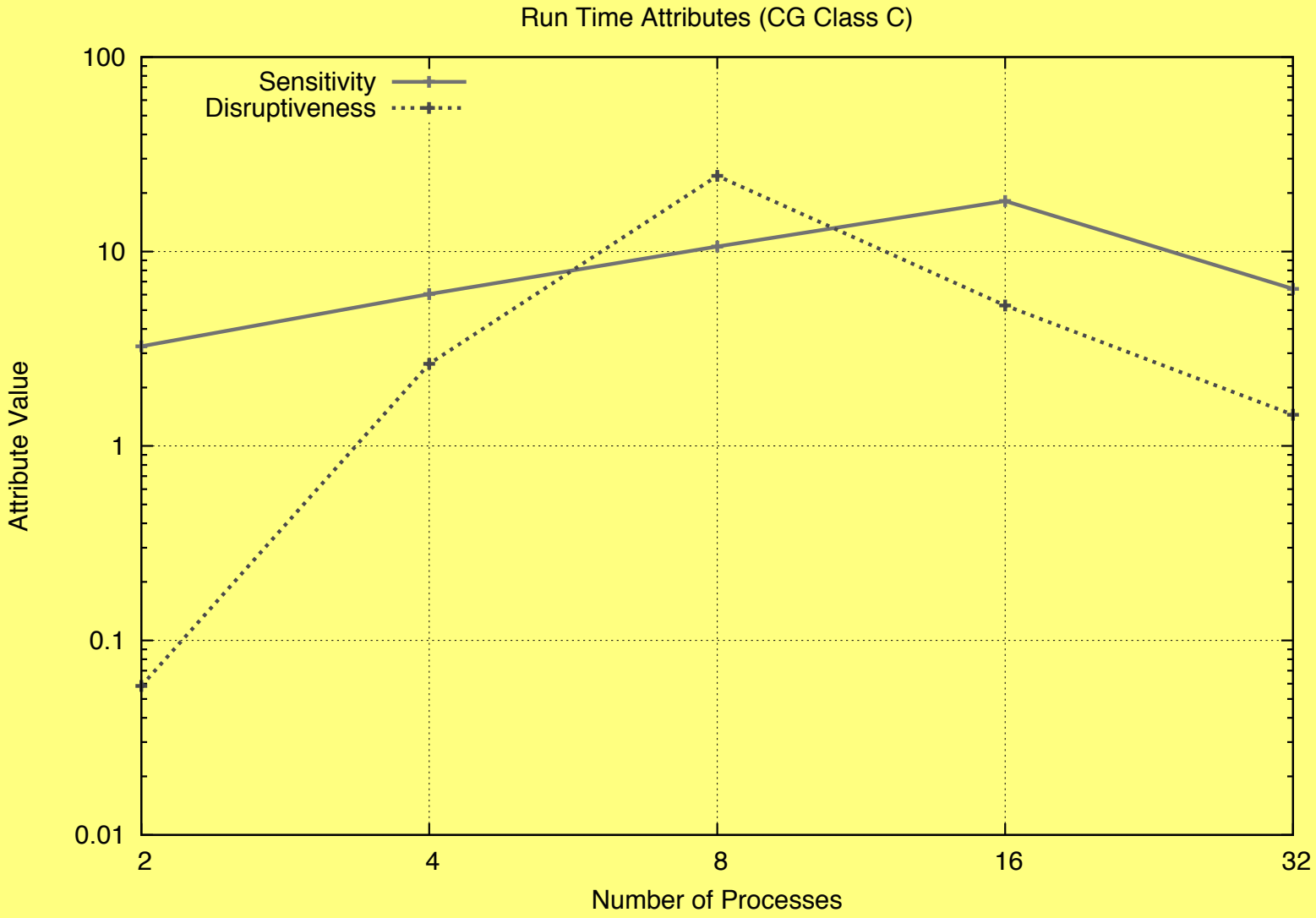
NAS Class C Attributes



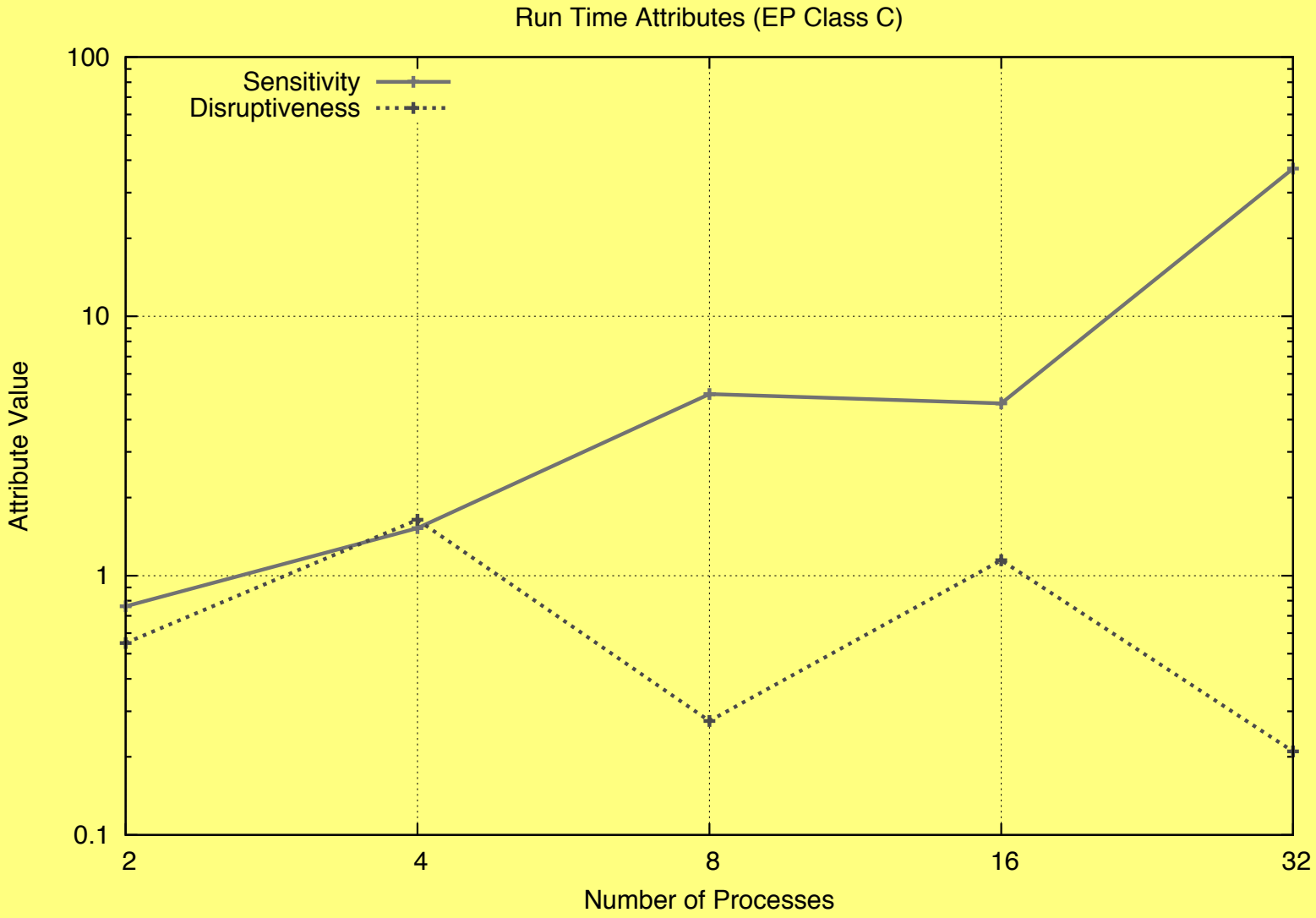
PSTSWM


Run Time Attributes (PSTSWM)
Sensitivity
Disruptiveness
Attribute Value
1000
100
10
1
0.1
0.01
2
4
8
16
32
Number of Processes

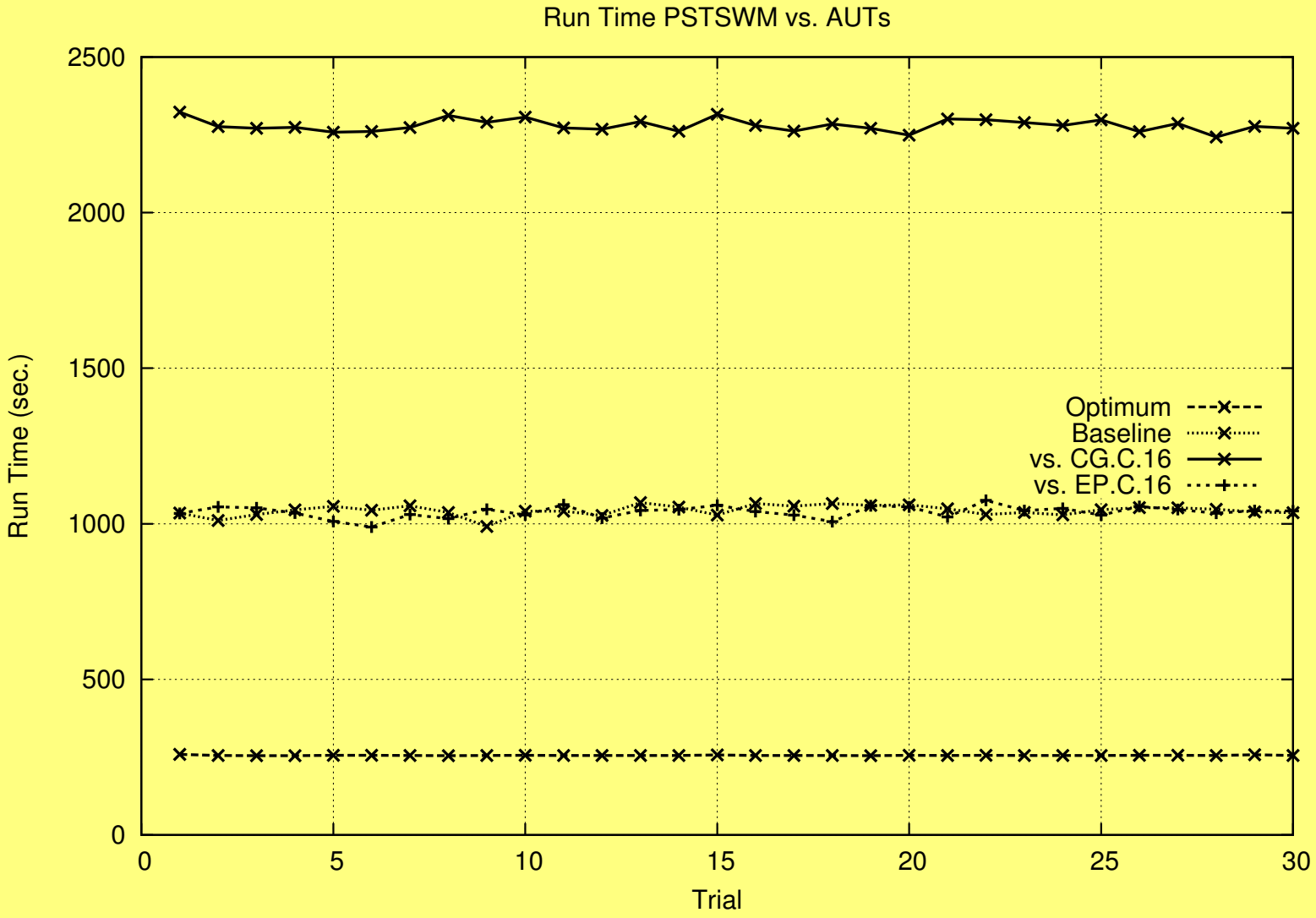
Concurrent Applications



Concurrent Applications



Concurrent Applications



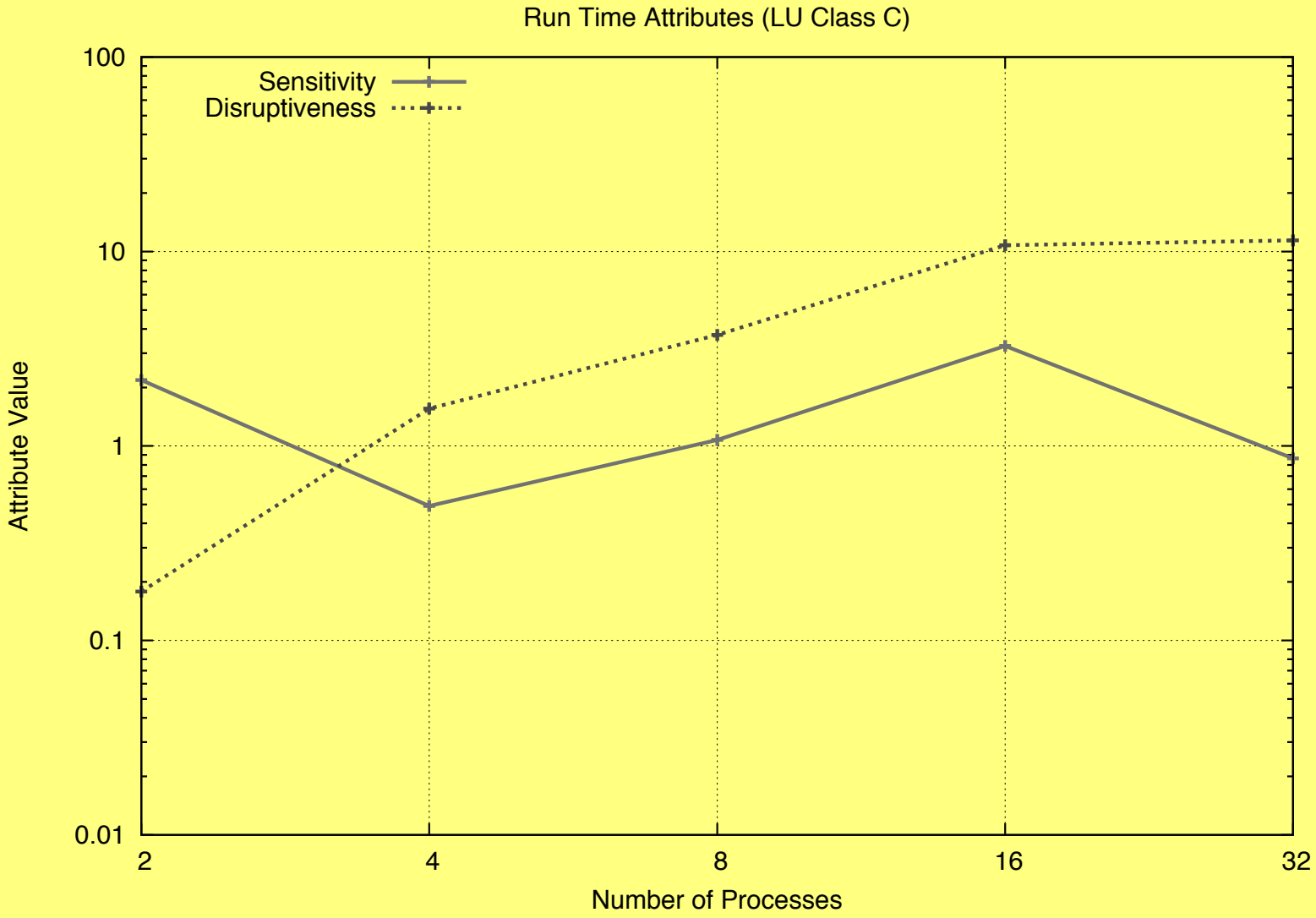
Process Binding

- MPICH2-(pre 1.3x) vs. MPICH2-1.3x
  - Hydra process manager: specified at build time vs. default
  - core-binding interface: non-functional vs. functional
  - verified using `top`
- core-binding (repeatable process placement) makes a significant difference
- Ramifications to PACE, PARSE
  - None to PACE
  - PARSE process placement (stride) in multi-core, multi-processor cases - depreciated

Process Locality

- Optimum process location
  - Best case: Co-locating processes to adjacent cores
  - Some applications better off spread across processors, nodes
  - Verified during AUT baseline tests
- 71% run time (= energy) difference observed in some cases
- Ramifications
  - Work in progress

NAS LU-4 Class C



NAS LU-4 Class C

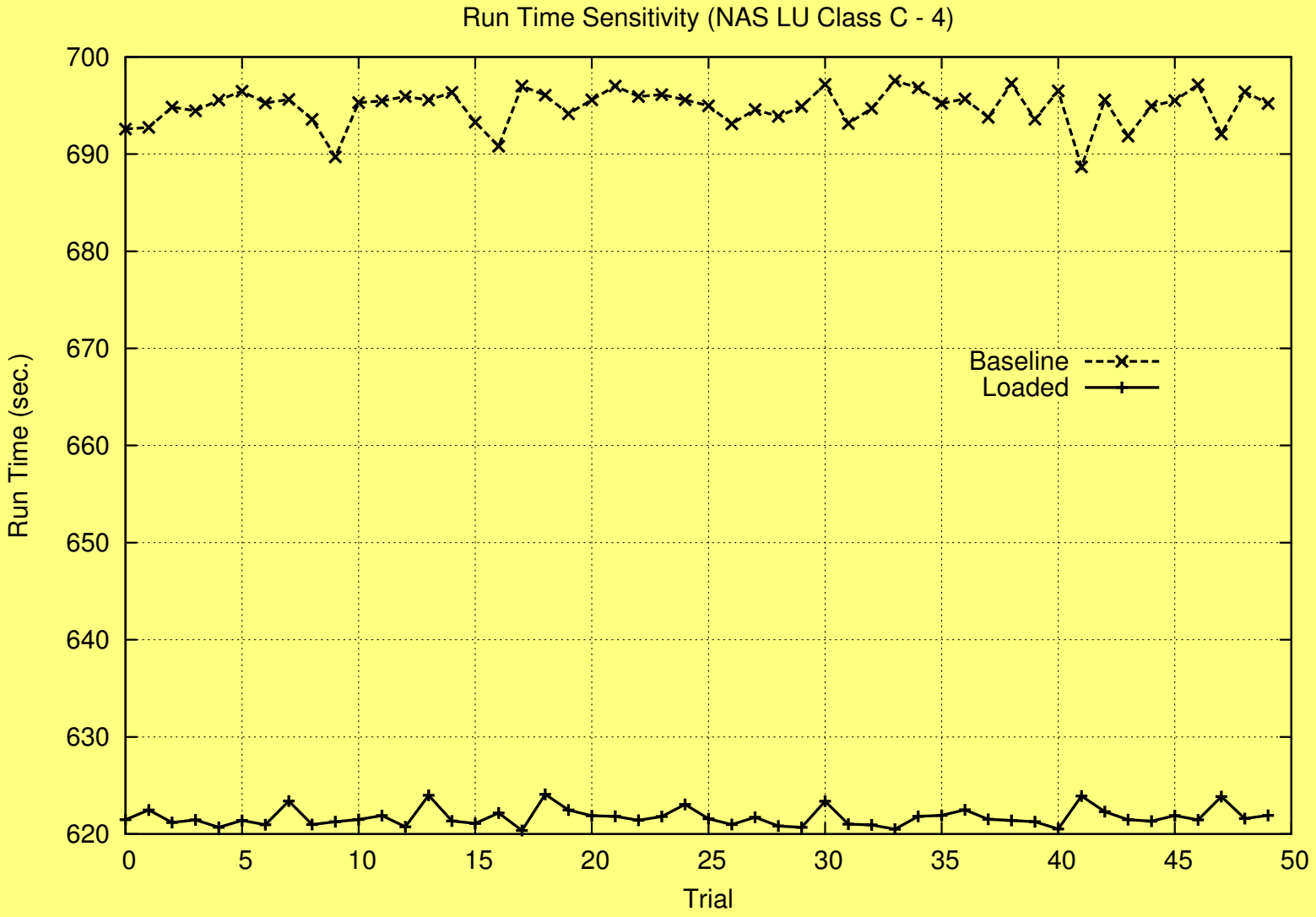


Table of Contents

Contributions

- Disruptiveness Attribute
  - Inverse of sensitivity
  - Application's tendency to disrupt others
- Verification
  - PSTSWM vs. non-disruptive application (EP)
  - PSTSWM vs. disruptive application (CG)
- Surprises
  - Process binding
  - Post MPICH2-1.3b1 "Hydra" implementation binds processes
  - Process location - now more of a sensitivity issue

Ongoing Work

- Characterize sensitivity to process locality
  - Optimum baseline configuration
  - Ramifications to other attributes
- Investigate disruptiveness assessment more rigorously
  - $D_{ij} < 1 \Rightarrow$ Investigate *COM* and *ROV* components
  - Optimize PACE parameters for disruptiveness evaluation
- Investigate attributes at scale
  - Scientific applications
  - Hybrid architectures - accelerators (GPU)
- Correlation to managing both performance and energy efficiency
  - Attribute components (*COM* and *ROV*)
  - Standard interfaces to HPC and HVAC systems

Acknowledgements

- U.S. Department of Energy
  - Contract #DE-SC0004596
- PC Krause and Associates, Inc.
  - http://www.pcka.com
- Purdue University
  - http://www.purdue.edu

Thank You

Questions?

Jeffrey J. Evans
jje@purdue.edu
evans@pcka.com

References

1. Evans, J.J. and Hood, C.S., A network performance sensitivity metric for parallel applications, *Int. J. High Performance Computing and Networking*, Vol. 7, No. 1, pp. 8-18.
2. Evans J., Lucas, C., Evaluation of parallel application-level behavioral attributes, *Proceedings of the 25th International Conference on Supercomputing, First International Workshop on Characterizing Applications for Heterogeneous Exascale Systems, CACHES 2011*, June 2011.
3. Evans J., Lucas, C., PARSE 2.0: a tool for parallel application run time behavior evaluation, *Proceedings of the 31st International Conference on Distributed Computing Systems*, June 2011.
4. Veeraraghavan, P., Evans J., Parallel Application Communication Performance on Multi-Core High Performance Computing Systems, *Proceedings of the IASTED International Conference on Parallel and Distributed Computing and Systems, PDCS 2010,* November 2010.
5. Evans J., On Performance and Energy Management in High Performance Computing Systems, *Proceedings of the Second International Workshop on Green Computing GreenCom 2010, 39th International Conference on Parallel Processing, GreenCom 2010,* pgs. 445-452, September 2010. (Invited Paper and Presentation).

References

6. Hu, F., and Evans, J., Power and environment aware control of beowulf clusters. *Cluster Computing*, Vol. 12, No. 3, pages 299-308, DOI: 10.1007/s10586-009-0085-z, March, 2009.
7. Evans J., Hood C., A Network performance sensitivity metric for parallel applications, *Proceedings of the Fifth International Symposium on Parallel and Distributed Processing and Applications, ISPA07,* August 2007, (Best Paper Award).
8. Hu, F., Evans J., Linux kernel improvement: Toward dynamic power management of beowulf clusters, *Proceedings of the 8th LCI International Conference on High-Performance Clustered Computing,* May 2007, CDROM.
9. Evans J., Hood C., PARSE: A tool for parallel application run time sensitivity evaluation, *Proceedings of the IEEE International Conference on Parallel and Distributed Systems, ICPADS2006,* pgs. 475-484, July 2006.
10. Evans J., Hood C., Network performance variability in NOW clusters, *Proceedings of the 5th IEEE/ACM conference on cluster computing and the Grid (CCGrid2005),* May, 2005.
11. Evans J., Hood C., Application communication emulation for performance managment of NOW clusters, *Proceedings of the 9th IFIP/IEEE International Symposium on Integrated Network Management,* May, 2005.

References

12. Evans J., Hood C., Baik, S., Kroculik, J. Network adaptability in clusters and Grids, *Proceedings from the Conference on Advances in Internet Technologies and Applications (CAITA),* July, 2004.
13. Evans J., Hood C., Gropp W. Exploring the relationship between parallel application run-time variability and network performance in clusters, *Workshop on High-Speed Local Networks (HSLN), IEEE Conference on Local Computer Networks (LCN),* pages 538-547, October, 2003.
14. Evans J., Baik S., Hood C., Gropp W. Toward understanding soft faults in high performance cluster networks, *IFIP/IEEE International Symposium on Integrated Network Management,* pages 117-121, March, 2003.