Predicting Parallel Application Performance using LogGP

Presenter: Michael J. Brim
CS747: Advanced Computer Systems Modeling
May 10, 2002

Outline

- Motivation
- LogP & LogGP Models
- MPI Send & Recv Models
- NPB LU Application Overview
- Constructing the LU Model
- Validation

Motivation

- Move towards commodity clusters for high-performance computing
- Do previous models for MPPs such as IBM SP & SGI Origin hold for this new paradigm?
- Will current applications scale with the ever-increasing size of clusters?

LogP : A Model for Parallel Apps

- Seeks to represent the following application characteristics:
  - Computational bandwidth
  - Communication bandwidth & delay
  - Effects of work partitioning
- Four Parameters
  - L : latency of the network
  - o : overhead of processing messages
  - g : communication gap – time between messages
  - P : number of processors

LogGP: LogP + Large Messages

- Original LogP model has no support for architectures which provide higher communication bandwidth for larger messages
- Solution: add another parameter
  - G : communication gap per byte

Model Construction - MPI

- Analysis of LAM/MPI
  - Message: Header + Data
  - Short vs. Long Messages
    - Short: < 65536 bytes
    - Long: >= 65536 bytes
  - Asynchronous Sends
    - Short: send header & data at same time
    - Long: send header, wait for ack, send data
  - Blocking Receives

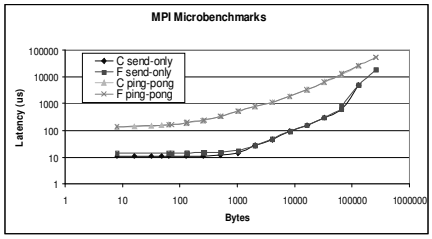
Model Construction – MPI

- Short Message
  - send header & data at same time

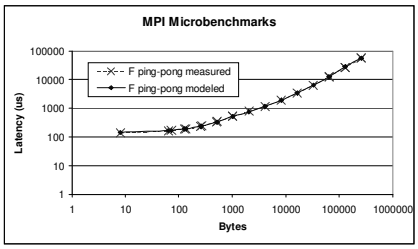
  $T_{comm} = o_s + L + o_s + (G_s * \text{msg-size})$
- Long Message
  - send header, wait for ack, send data

  $T_{comm} = (o_s + L + o_s) + (o_s + L) +$
  $(o_l + L + o_l + G_l * \text{msg-size})$

MPI MicroBenchmarks



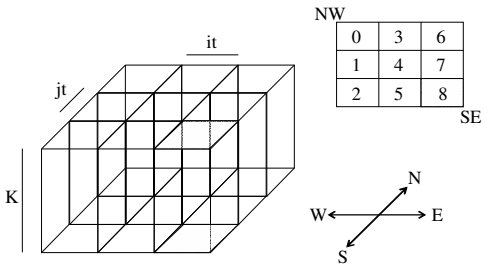
MPI Model Verification



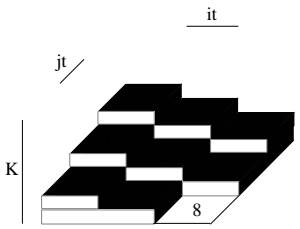
NAS Parallel Benchmark: LU

- 3-Dimensional Computational Fluid Dynamics (CFD) Application
- "Wavefront" communication pattern
- Problem Grid Sizes:
  - W: 33 x 33 x 33
  - A: 64 x 64 x 64
  - B: 102 x 102 x 102

LU : Partitioning the Grid



LU : Diagonal Pipelining



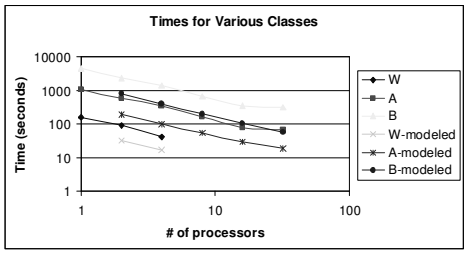
Model Construction - LU

- 3 phases
  - Initialization
  - SSOR: when wavefront happens
  - Verification of results
- Model
  - $T_{total} = T_{init} + T_{SSOR} + T_{verify}$
  - Get T_{init} & T_{verify} from measurement
  - Predict T_{SSOR} using LogGP & measurement

Model Construction - LU

- $W_{i,j} = it * jt * W_{gridpt}$
- $StartP_{i,j} =$
  $\max(StartP_{i-1,j} + W_{i-1,j} + T_{send} + T_{comm},$
  $StartP_{i,j-1} + W_{i,j-1} + T_{comm} + T_{recv})$
- $T_{blts} = StartP_{n,m} + K*W_{n,m} + (K-1)*(2* T_{recv})$
- $T_{buts} = T_{blts}$ (only in opposite direction)
- $T_{SSOR} = \#iterations * (T_{blts} + T_{buts})$

LU Model Verification



References

- D. Culler et al, "LogP: Towards a Realistic Model of Parallel Computation". In *Fourth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, May 1993.
- A. Alexandrov et al, "LogGP: Incorporating Long Messages into the LogP Model". In *Proc. 7th Annual ACM Symposium on Parallel Algorithms and Architectures*, July 1995.
- D. Sundaram-Stukel and M. K. Vernon, "Predictive Analysis of a Wavefront Application using LogGP". In *Proc. 7th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, May 1999.
- LAM/MPI, http://www.lam-mpi.org
- D. Bailey et al, "The NAS Parallel Benchmarks 2.0", Technical Report NAS-95-020, December 1995.