Low-Overhead LogGP Parameter Assessment for Modern Interconnection Networks

T. Hoefler, A. Lichei, W. Rehm

Open Systems Lab
Indiana University
Bloomington, USA

Computer Architecture Group
Technical University of Chemnitz
Chemnitz, Germany

IPDPS'07 - PMEO-PDS'07 Workshop
Long Beach, CA, USA
30th March 2007

Introduction

- network performance prediction is important
- assess the runtime of parallel algorithms
- optimize communication patterns (e.g., collective)
- runtime message scheduling (heterogeneous interfaces)

Our approach

We propose a contention-free LogGP parameter assessment method to be used in (changing) runtime environments.

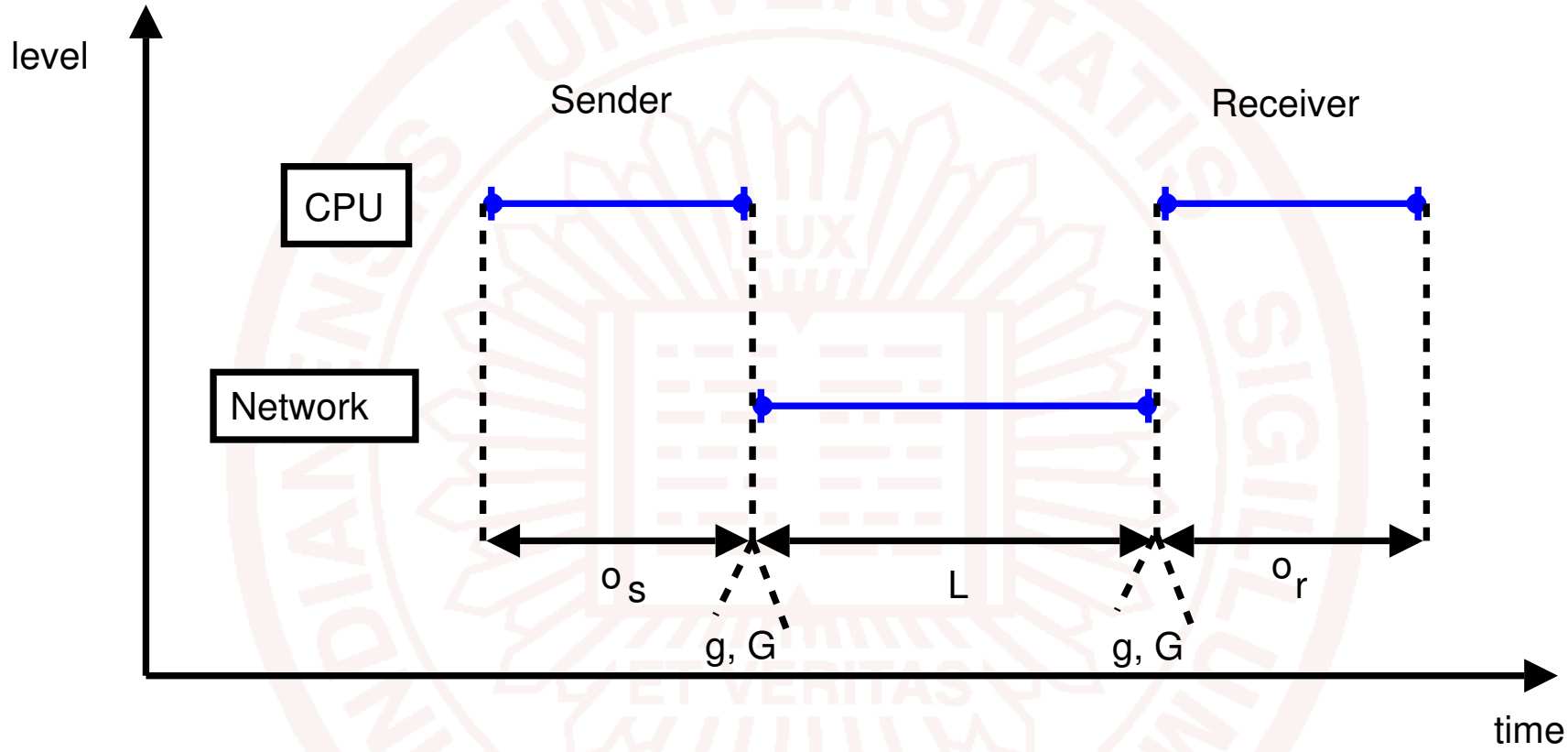
Introduction

- network performance prediction is important
- assess the runtime of parallel algorithms
- optimize communication patterns (e.g., collective)
- runtime message scheduling (heterogeneous interfaces)

Our approach

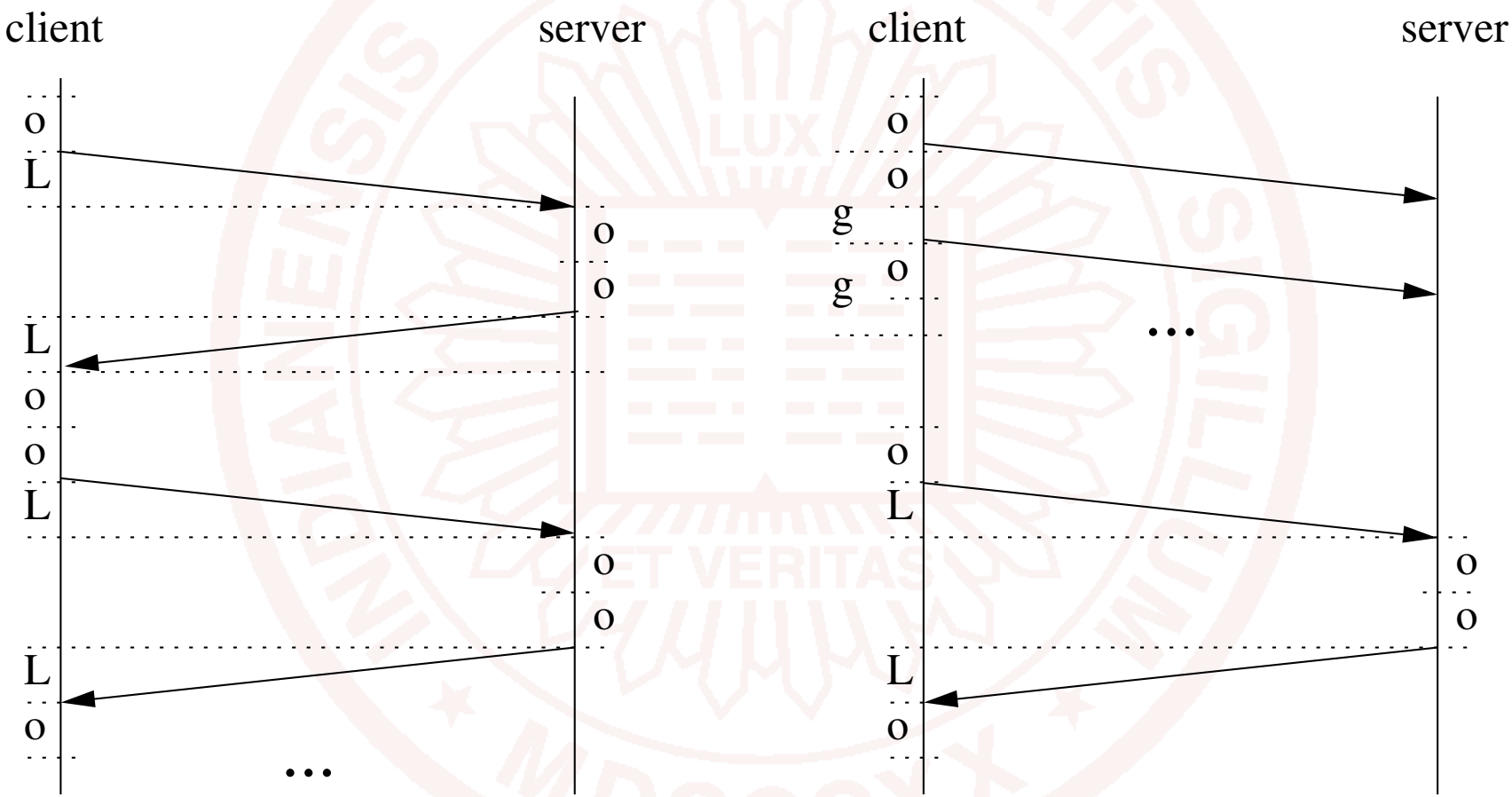
We propose a contention-free LogGP parameter assessment method to be used in (changing) runtime environments.

The LogGP Model



The Tools to Measure

no central clock → measurements on one host only



Related Work

Culler et al. / Ianello et al.

- differentiates between o_s and o_r
- o_s: issue small number (n) of sends and divide by n
- o_r: delay between messages, larger as RTT, substract o_s
- g: flood network
- L: RTT/2 - o_r - o_s (third order errors)

Kielmann et al.

- changes the model to pLogP
- o_s: time for a single send
- o_r: time to copy the message from the receive buffer
- g: flood network (if accurate)
- L: (RTT(0)-2g(0))/2 (second order errors)

Related Work

Culler et al. / Ianello et al.

- differentiates between o_s and o_r
- o_s: issue small number (n) of sends and divide by n
- o_r: delay between messages, larger as RTT, substract o_s
- g: flood network
- L: RTT/2 - o_r - o_s (third order errors)

Kielmann et al.

- changes the model to pLogP
- o_s: time for a single send
- o_r: time to copy the message from the receive buffer
- g: flood network (if accurate)
- L: (RTT(0)-2g(0))/2 (second order errors)

Related Work

Bell et al.

- differentiates between o_s and o_r
- o_s: uses delay between message sends (adjust delay until $d + o = g + (s-1)G$ (multiple measurements) $\Rightarrow$ $o_s = g + (s-1)G - d$ (second order errors)
- o_r: similar to Culler et al.
- g: flood network (similar to Kielmann et al.)
- L: not measured (network effects)
- EEL: end-to-end latency (RTT)

Our Approach

Definition of $PRTT(n, d, s)$

- parametrized round-trip-time
- n - number of successive messages
- d - delay between messages
- s - message size

$PRTT(n, d, s)$ and LogGP

- $PRTT(1, 0, s) = 2 \cdot (L + 2o + (s-1)G)$
- $G_{all} = g + (s-1)G$
- $PRTT(n, 0, s) = 2 \cdot (L + 2o + (s-1)G) + (n-1) \cdot G_{all}$
- $PRTT(n, 0, s) = PRTT(1, 0, s) + (n-1) \cdot G_{all}$
- $PRTT(n, d, s) = PRTT(1, 0, s) + (n-1) \cdot max\{o + d, G_{all}\}$

Our Approach

Definition of $PRTT(n, d, s)$

- parametrized round-trip-time
- n - number of successive messages
- d - delay between messages
- s - message size

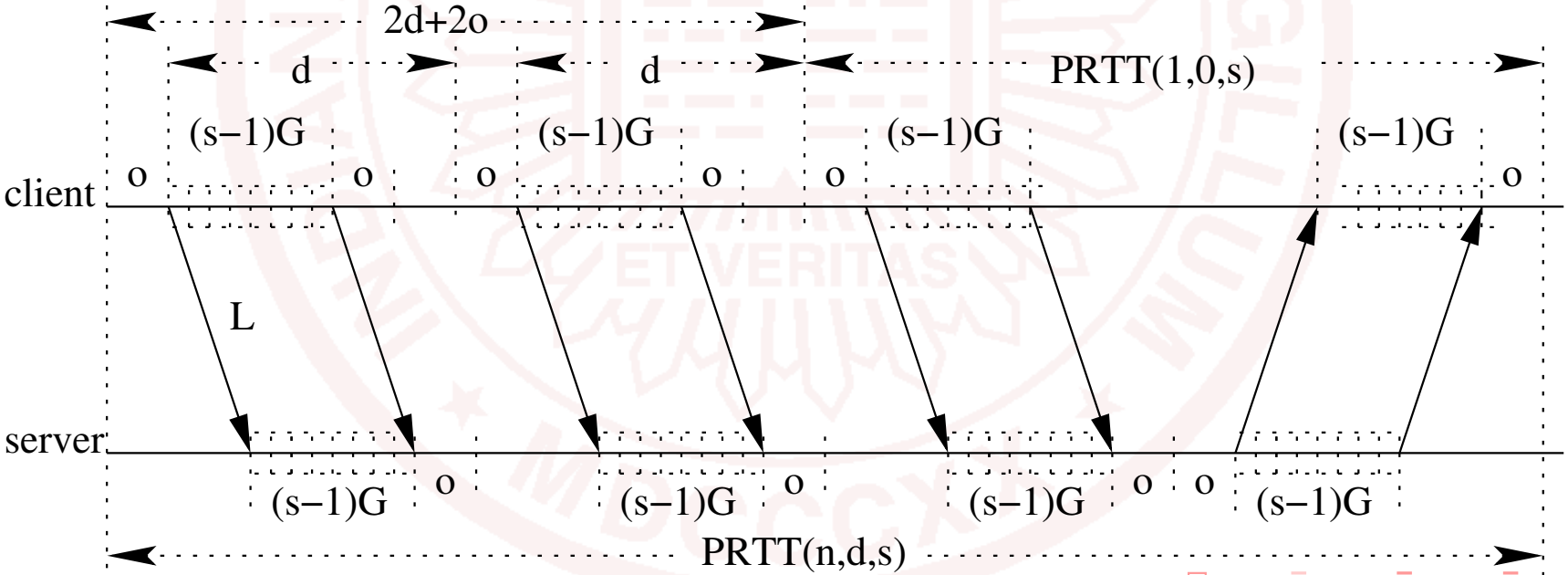
$PRTT(n, d, s)$ and LogGP

- $PRTT(1, 0, s) = 2 \cdot (L + 2o + (s-1)G)$
- $G_{all} = g + (s-1)G$
- $PRTT(n, 0, s) = 2 \cdot (L + 2o + (s-1)G) + (n-1) \cdot G_{all}$
- $PRTT(n, 0, s) = PRTT(1, 0, s) + (n-1) \cdot G_{all}$
- $PRTT(n, d, s) = PRTT(1, 0, s) + (n-1) \cdot max\{o + d, G_{all}\}$

Assessment of the Overhead

Assessing o

- $\frac{PRTT(n,d,s)-PRTT(1,0,s)}{n-1} = max\{o+d, G_{all}\}$
- we chose $d > G_{all}$
- $\frac{PRTT(n,d,s)-PRTT(1,0,s)}{n-1} = o+d$
- we chose $d = PRTT(1,0,s)$ (fall back to $d = PRTT(2,0,s)$ for high gaps)



Assessment of the Gaps

Assessing g and G

- $G(s-1)+g = \frac{PRTT(n,0,s)-PRTT(1,0,s)}{n-1}$
- polynomial of degree 1 (linear function)
- $\Rightarrow$ measue $PRTT(n,0,s)$ and $PRTT(1,0,s)$ for different s
- more measurement values increase accuracy ($\Rightarrow$ least squares linear fit)

Detecting Protocol Changes

- comm. subsystems use data-size dependent protocols
- different parameters
- autodetection possible
- changes in the mean least squares deviation

Assessment of the Gaps

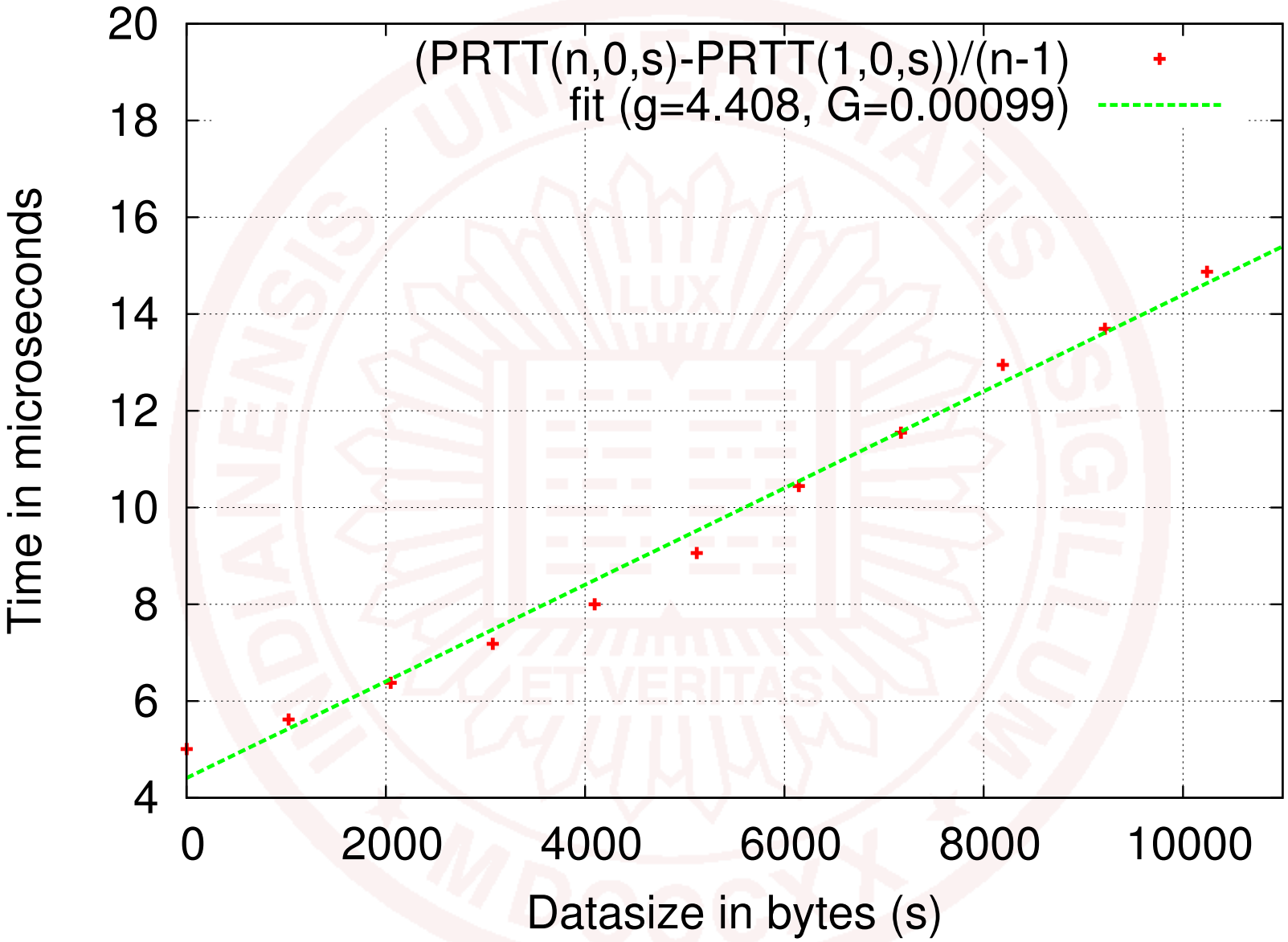
Assessing g and G

- $G(s-1)+g = \frac{PRTT(n,0,s)-PRTT(1,0,s)}{n-1}$
- polynomial of degree 1 (linear function)
- $\Rightarrow$ measue $PRTT(n,0,s)$ and $PRTT(1,0,s)$ for different s
- more measurement values increase accuracy ($\Rightarrow$ least squares linear fit)

Detecting Protocol Changes

- comm. subsystems use data-size dependent protocols
- different parameters
- autodetection possible
- changes in the mean least squares deviation

Assessment of the Gaps - Example Fit



Results

Netgauge

- support for multiple low-level Interfaces
- e.g., TCP, UDP, IB, GM, SCI, MPI
- $\Rightarrow$ low-level and MPI measurements (lib overhead)
- different communication patterns (different measuements)
- implemented new pattern: **loggp**

MPI Benchmark Environment

- MPICH2 1.0.3 for Gigabit Ethernet
- NMPI for SCI
- Open MPI 1.1 for InfiniBand^{TM}/OFED
- Open MPI 1.1 for Myrinet/GM

Results

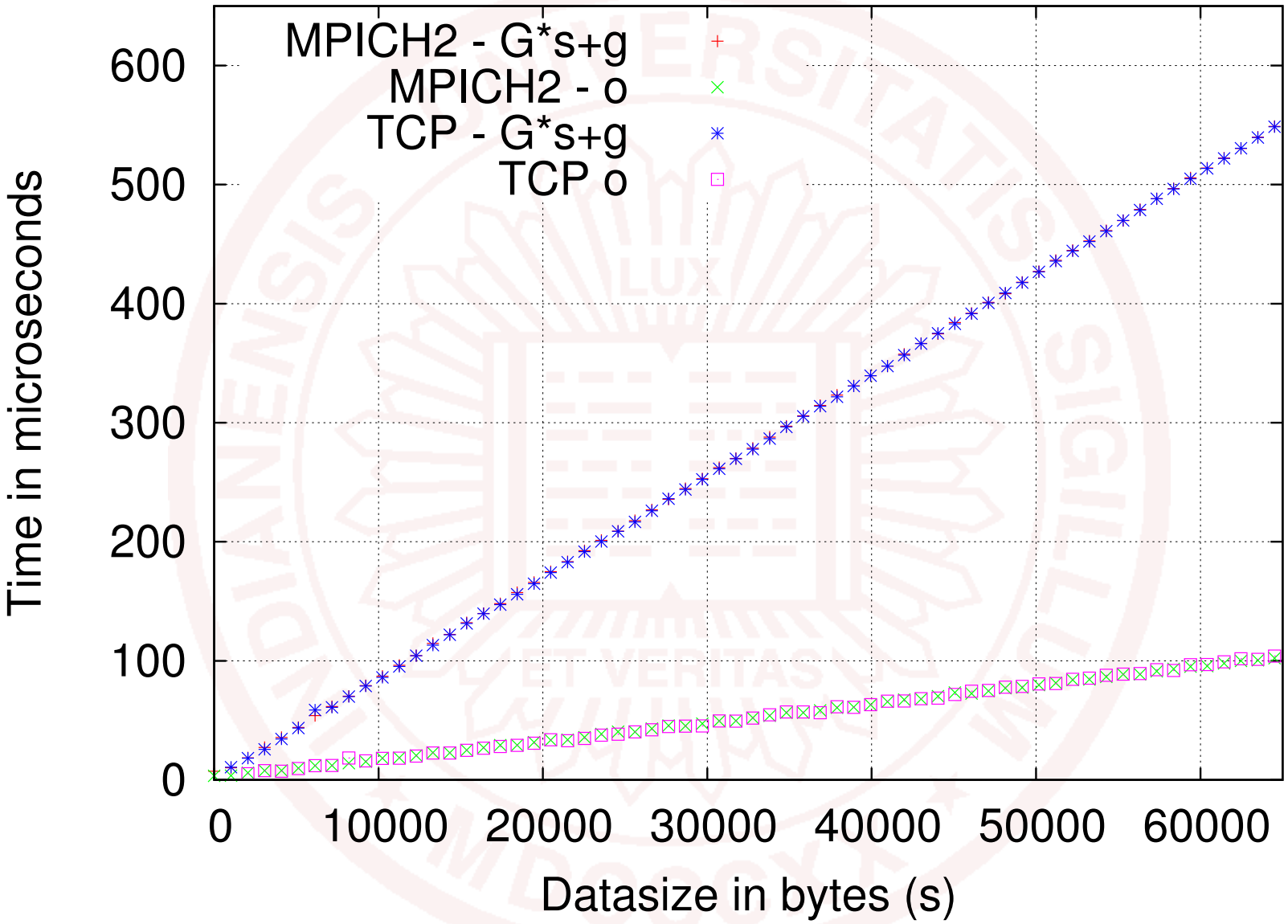
Netgauge

- support for multiple low-level Interfaces
- e.g., TCP, UDP, IB, GM, SCI, MPI
- $\Rightarrow$ low-level and MPI measurements (lib overhead)
- different communication patterns (different measuements)
- implemented new pattern: **loggp**

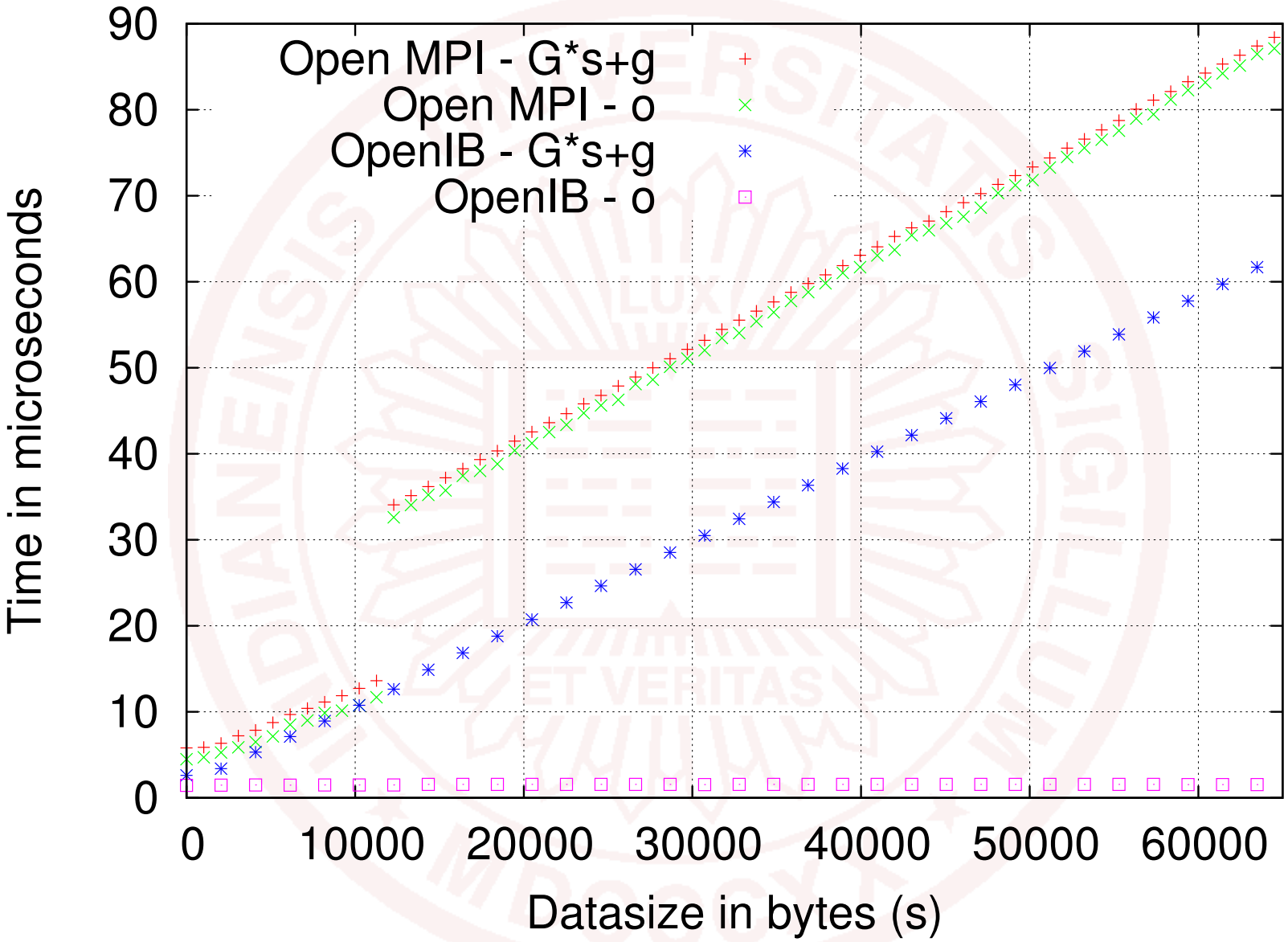
MPI Benchmark Environment

- MPICH2 1.0.3 for Gigabit Ethernet
- NMPI for SCI
- Open MPI 1.1 for InfiniBand^{TM}/OFED
- Open MPI 1.1 for Myrinet/GM

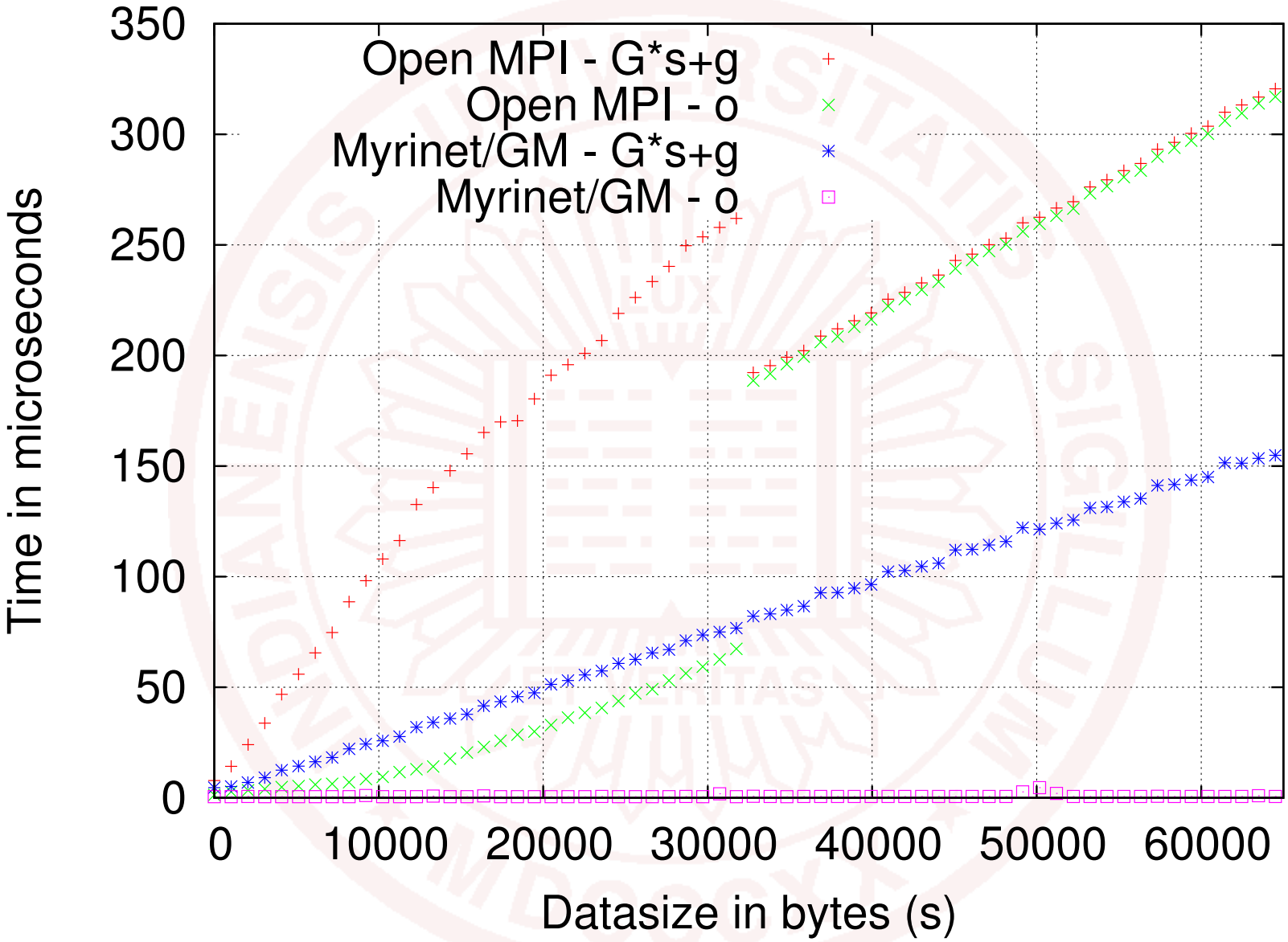
TCP/IP over Gigabit Ethernet



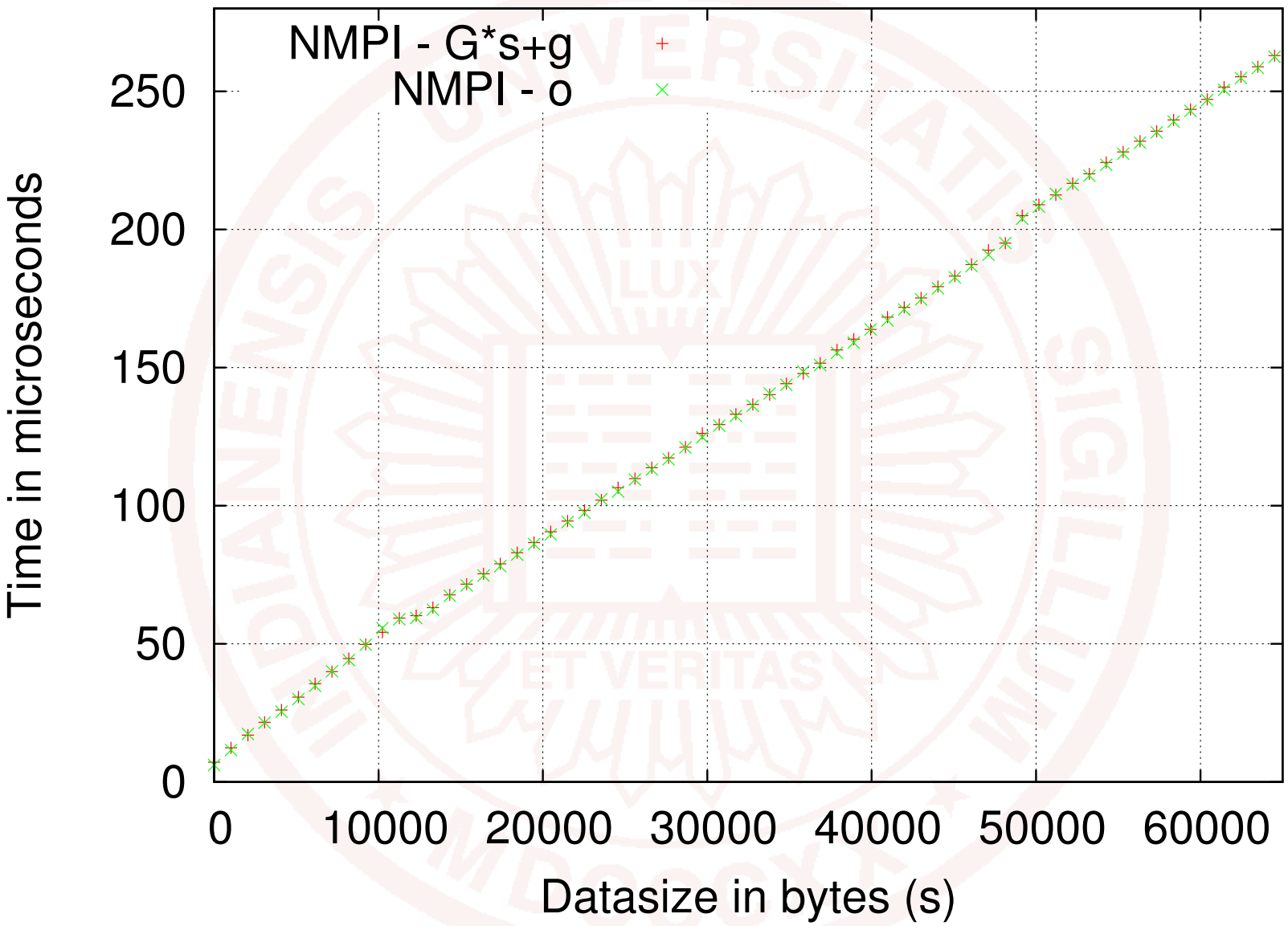
InfiniBand™/OFED



Myrinet/GM


Open MPI - G*s+g
Open MPI - o
Myrinet/GM - G*s+g
Myrinet/GM - o
Time in microseconds
Datasize in bytes (s)
0
50
100
150
200
250
300
350
0
10000
20000
30000
40000
50000
60000

SCI (MPI only)


NMPI - G*s+g
NMPI - o
Time in microseconds
0
50
100
150
200
250
0
10000
20000
30000
40000
50000
60000
Datasize in bytes (s)

Numeric Results

Trans.	**Prot. Int.**	**L (μs)**	**o(1) (μs)**	**g (μs)**	**G ($\mu s/byte$)**
TCP	$1 \leq s$	45.74	3.46	0.915	0.00849
SCI	$1 \leq s < 12289$	5.48	6.10	7.78	0.0045
	$12289 \leq s$	5.48	6.10	13.34	0.0037
OFED	$1 \leq s < 12289$	5.96	4.72	5.14	0.00073
	$12289 \leq s$	5.96	4.72	21.39	0.00103
GM	$1 \leq s < 32769$	10.53	1.27	9.44	0.0092
	$32769 \leq s$	10.53	1.27	52.01	0.0042

Conclusions and Future Work

Conclusions

- new measurement technique for LogGP parameters
- congestion-free
- low overhead
- accurate (no higher-order errors, multiple datapoints)
- detects protocol changes automatically

Future Work

- apply scheme to heterogeneous NIC scheduling
- analyze communication schemes
- measure non-blocking MPI communication

Conclusions and Future Work

Conclusions

- new measurement technique for LogGP parameters
- congestion-free
- low overhead
- accurate (no higher-order errors, multiple datapoints)
- detects protocol changes automatically

Future Work

- apply scheme to heterogeneous NIC scheduling
- analyze communication schemes
- measure non-blocking MPI communication