Fast Classification of MPI Applications Using Lamport's Logical Clocks

Zhou Tong
Florida State University
tong@cs.fsu.edu

Scott Pakin and Mike Lang
Los Alamos National Lab
{mlang, pakin}@lanl.org

Xin Yuan
Florida State University
xyuan@cs.fsu.edu

Introduction

Fast classification and identification of performance limiting factors in MPI applications

- One simulation, predict application performance for many network configurations by replaying the traces
- Use extended Lamport's logical clocks to maintain time counters that track computation, bandwidth, latency and wait time
- Classify 9 DOE's full applications and miniapps into bandwidth-bound (BW), Latency-bond (L), computation-bound (Comp.) or load-imbalance-bound (Imb.)
- Performance evaluation with NAS benchmarks show that our tool is fast and scalable

Key Ideas

- Trace-driven, MPI-based modelling tool
- Classify application based on the performance summary on a range of configurations (Bandwidth and Latency) of given interconnect technology

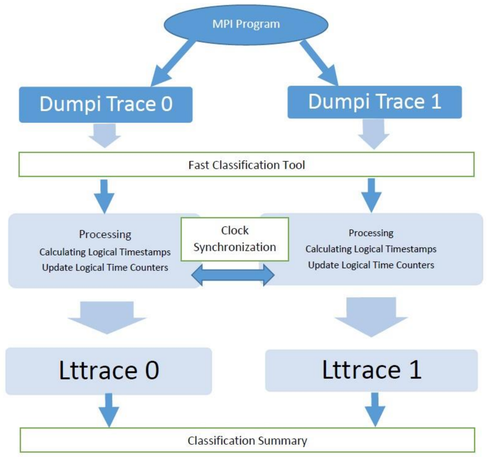
Modelling Assumptions:

- Model Communication with Eager Protocol as default. Rendezvous protocol is available if necessary.
- P2P: Hockney's model $\alpha + n \times \beta$, where α is the communication latency, n is the message size and latency β is the per-byte bandwidth speed
- Collective: global synchronization for all processes to be ready for the operation. Then, it is treated as a sequence of individual P2P communications.

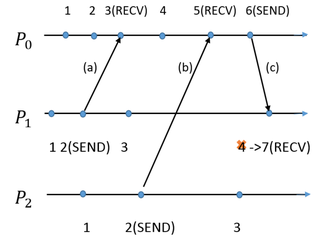
Benchmarks

AMR	Adaptive Mesh Refinement cosmology
BigFFT	3D Fast Fourier Transform solver
CLAMR	Cell-based adaptive mesh refinement
CR	Nek5000
FB	Halo update PDE solver code
MG	Geometric Multigrid elliptic solver
MiniFE	Finite element mini-application
PARTISN	Neutral-particle transport
NPB	NAS Parallel Benchmarks

Fast Classification Tool



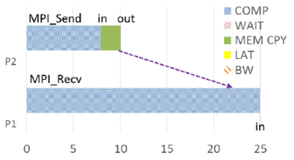
Basic Lamport's Clock



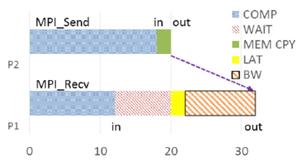
Basic Lamport's logical clock ensures partial ordering of causally-related events with increment of 1 unit of time for both computation and communication.

Extended Lamport's Logical Clocks

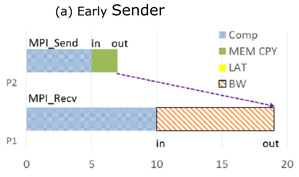
P2P and collective routines are modeled by computation, wait, latency and bandwidth counters with regards to computation time and non-unit latency.



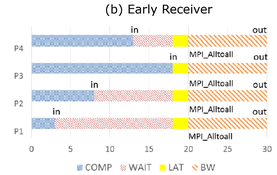
(a) Early Sender



(b) Early Receiver



(c) Concurrent Sender & Receiver



(d) Collective

Validation

Table II: Predicted and measured communication and total application time (in second) of 64-rank CLAMR, CR and FB on Cielito

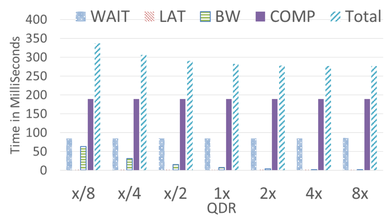
	CLAMR		CR		FB		
	eag.	eag.*	eag.	eag.*	eag.	eag.*	rend.
Comp.	0.23	0.23	0.26	0.26	0.86	0.86	0.86
Pred. Comm.	0.76	0.90	0.07	0.05	0.15	0.19	0.35
Act. Comm.	0.89	0.89	0.06	0.06	0.36	0.36	0.36
Comm. Err. %	-14.61	+1.12	+16.67	-16.67	-58.33	-47.22	-2.78
Pred. Tot.	0.99	1.13	0.33	0.31	1.01	1.05	1.21
Act. Tot.	1.12	1.12	0.32	0.32	1.22	1.22	1.22
Pred. Err. %	-11.61	+0.89	+3.13	-3.13	-17.21	-13.93	-0.82

For each benchmark:
1st column: (eag.) shows the predicted performance using default eager protocol with Hockney's model.

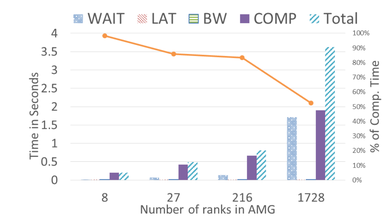
2nd column: (eag.*) shows that the predicted performance using the default eager protocol with the more accurate look-up table improves prediction errors of both communication and overall execution time. The overall prediction errors of CLAMR and CR are within 3%. Message sizes in CLAMR and CR are under 4KB.

For FB:
3rd column: shows prediction results with rendezvous protocol and look-up table further improves the prediction rate because message sizes in FB are under 4MB.

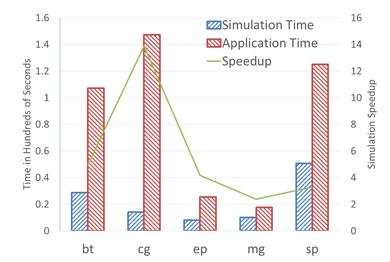
Preliminary Results



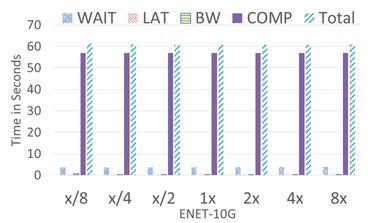
(a) BigFFT(100) is load-imbalanced-sensitive as wait time accounts for 30% of total time with QDR communication model.



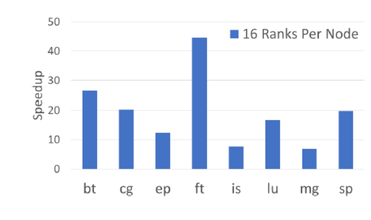
(c) Percentage of computation decreases as the problem size and the number of ranks in AMG increases.



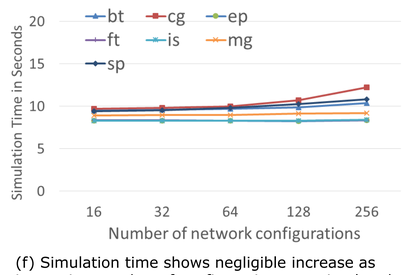
(e) Simulation time shows 2-15 time speedup for 4096-rank runs of NPB benchmarks.



(b) MiniFe(1152) is computation-bound as 90% time on computation with 10G Etherent Communication model.



(d) Simulation time shows 3-45 time speedup for 4096-rank runs of NPB benchmarks.



(f) Simulation time shows negligible increase as increasing number of configurations are simulated in one run.

Conclusions

Trace-based and communication-centric fast classification tool for MPI programs provide insights on application performance.

- Our innovation is to use a modified Lamport logical clock scheme that uses non-unit computation and communication times to predict overall time
- By maintaining multiple independent logical clocks that are parameterized differently but honor the same happens-before relationship, the tool can predict execution time on many network configurations in nearly the same time needed to predict time for a single configuration.
- This multiple-prediction capability enables new analyses to be performed that would be too computationally expensive to perform with traditional, one-configuration-at-a-time simulation.
- Classification results could be used to assist code optimization and better overlap of communication and computation.

Table I: Interconnect in current production systems.

Configurations	Latency (μ)	BW (Gbps)
1G Ethernet (E1G)	50	1
10G Ethernet (E10G)	5	10
InfiniBand QDR (QDR)	1.3	32

Table III: Configurations for classification

Model	ENET-1G	ENET-10G	QDR
Latency	(1, 6.25)	(10, 0.625)	(32, 0.1625)
	(1, 12.5)	(10, 1.25)	(32, 0.325)
	(1, 25)	(10, 2.5)	(32, 0.65)
	(1, 50)	(10, 5)	(32, 1.3)
	(1, 100)	(10, 10)	(32, 2.6)
	(1, 200)	(10, 20)	(32, 5.2)
	(1, 400)	(10, 40)	(32, 10.4)
BW	(.125, 50)	(1.25, 5)	(4, 1.3)
	(.25, 50)	(2.5, 5)	(8, 1.3)
	(.5, 50)	(5, 5)	(16, 1.3)
	(1, 50)	(10, 5)	(32, 1.3)
	(2, 50)	(20, 5)	(64, 1.3)
	(4, 50)	(40, 5)	(128, 1.3)
	(8, 50)	(80, 5)	(256, 1.3)
COMM	(0.125, 400)	(1.25, 40)	(4, 10.4)
	(0.25, 200)	(2.5, 20)	(8, 5.2)
	(0.5, 100)	(5, 10)	(16, 2.6)
	(1, 50)	(10, 5)	(32, 1.3)
	(2, 25)	(20, 2.5)	(64, 0.65)
	(4, 12.5)	(40, 1.25)	(128, 0.325)
	(8, 6.25)	(80, 0.625)	(256, 0.1625)

Table IV: Classification results (number of MPI ranks shown in parenthesis)

	ENET-1G	ENET-10G	QDR
AMG(8)	Comp.	Comp.	Comp.
AMG(27)	Imb.-s	Imb.-s	Imb.-s
AMG(216)	Imb.-s	Imb.-s	Imb.-s
AMG(1728)	Imb.	Imb.	Imb.
AMR(64)	Latency-s	Comp.	Comp.
BigFFT(100)	Comm.	Comm.	Imb.
CLAMR(64)†	Latency	Latency	Imb.
CR(64)†	Comm.	Comm.	Comp.
FB(64)†	BW-s	Comp.	Comp.
FB(125)	BW-s	Imb.-s	Imb.-s
MG(1000)	Imb.-s	Comp.	Comp.
MiniFE(1152)	Comp.	Comp.	Comp.
PARTISN(168)	Imb.	Imb.	Imb.
IS(64)†	Comm.	Imb.-s	Imb.-s

-s: denotes for sensitive but not bounded.

Acknowledgements

Special thanks to my mentor Scott Pakin, Mike Lang and my academic advisor Xin Yuan. This work is supported by Ultra-Scale Research Center, New Mexico Consortium, Los Alamos National Lab.