Trace Synchronization of multi-level, multi-core distributed system

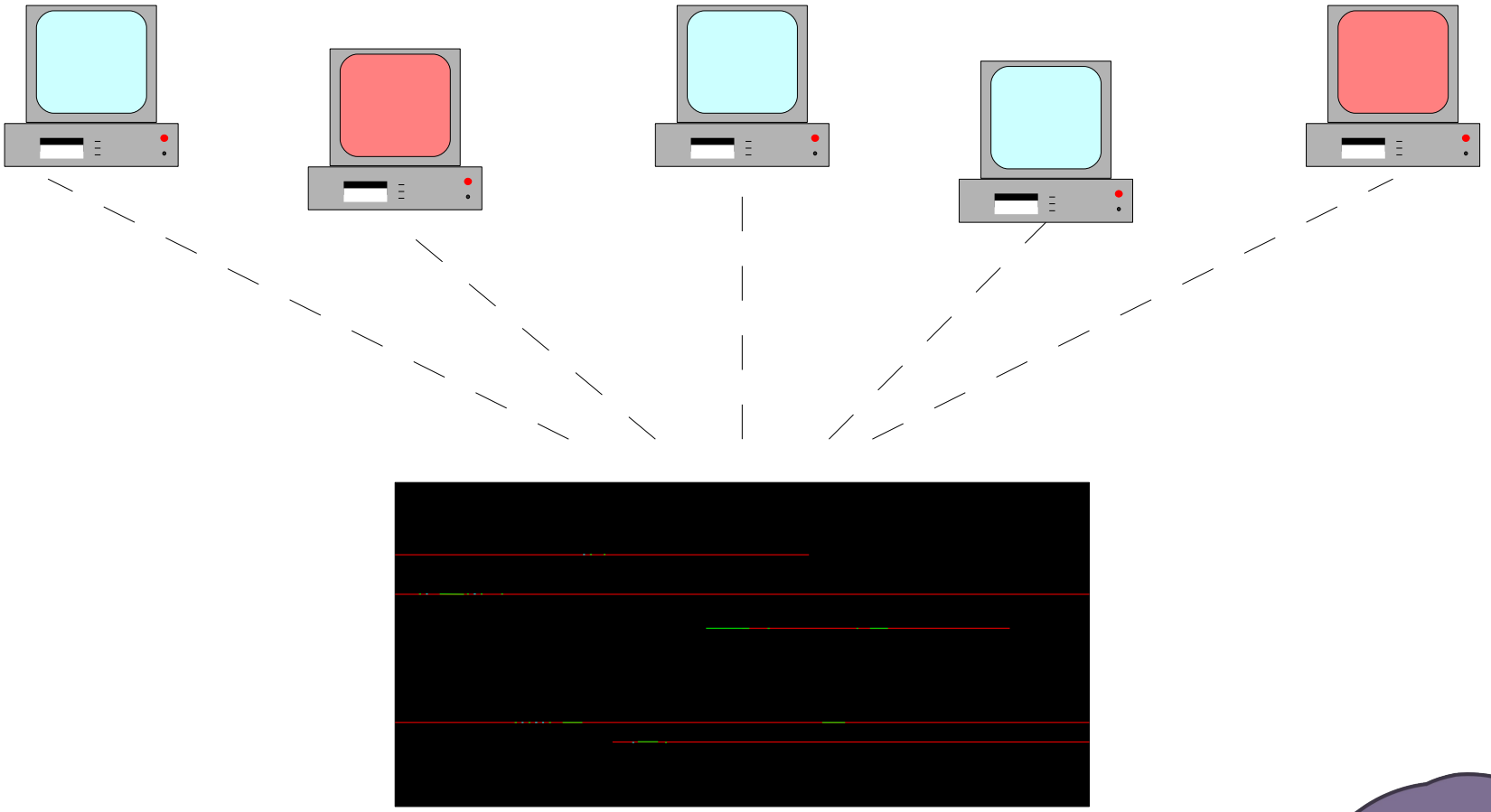
Masoume Jabbarifar
masoume.jabbarifar@polymtl.ca
DORSAL

29 June 2010
École Polytechnique, Montreal

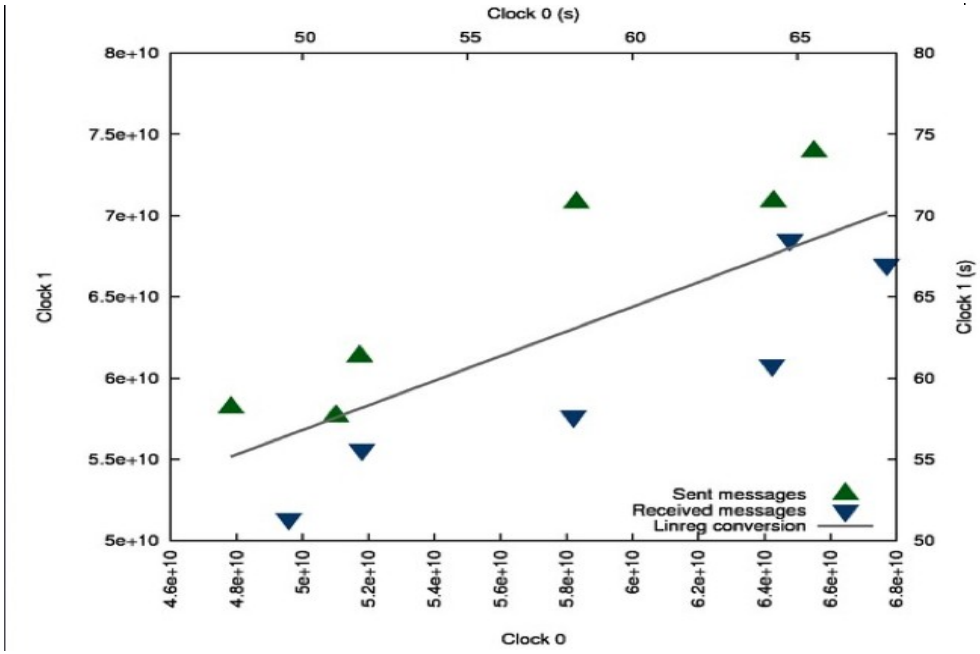
Content

- **Why synchronization**
- **Synchronization methods**
- **Architecture**
- **Processing module**
- **Matching module**
- **Sync based on accuracy**
- **Sync based on time**
- **Comparison accuracy vs. time**
- **Future work**
- **References**

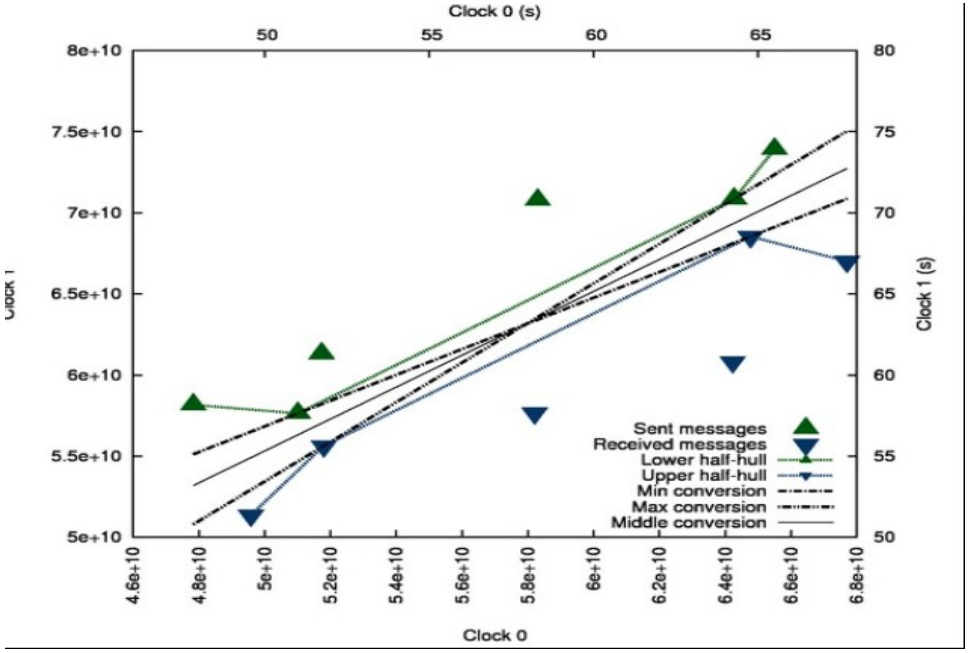
Why synchronization?



Synchronization Methods



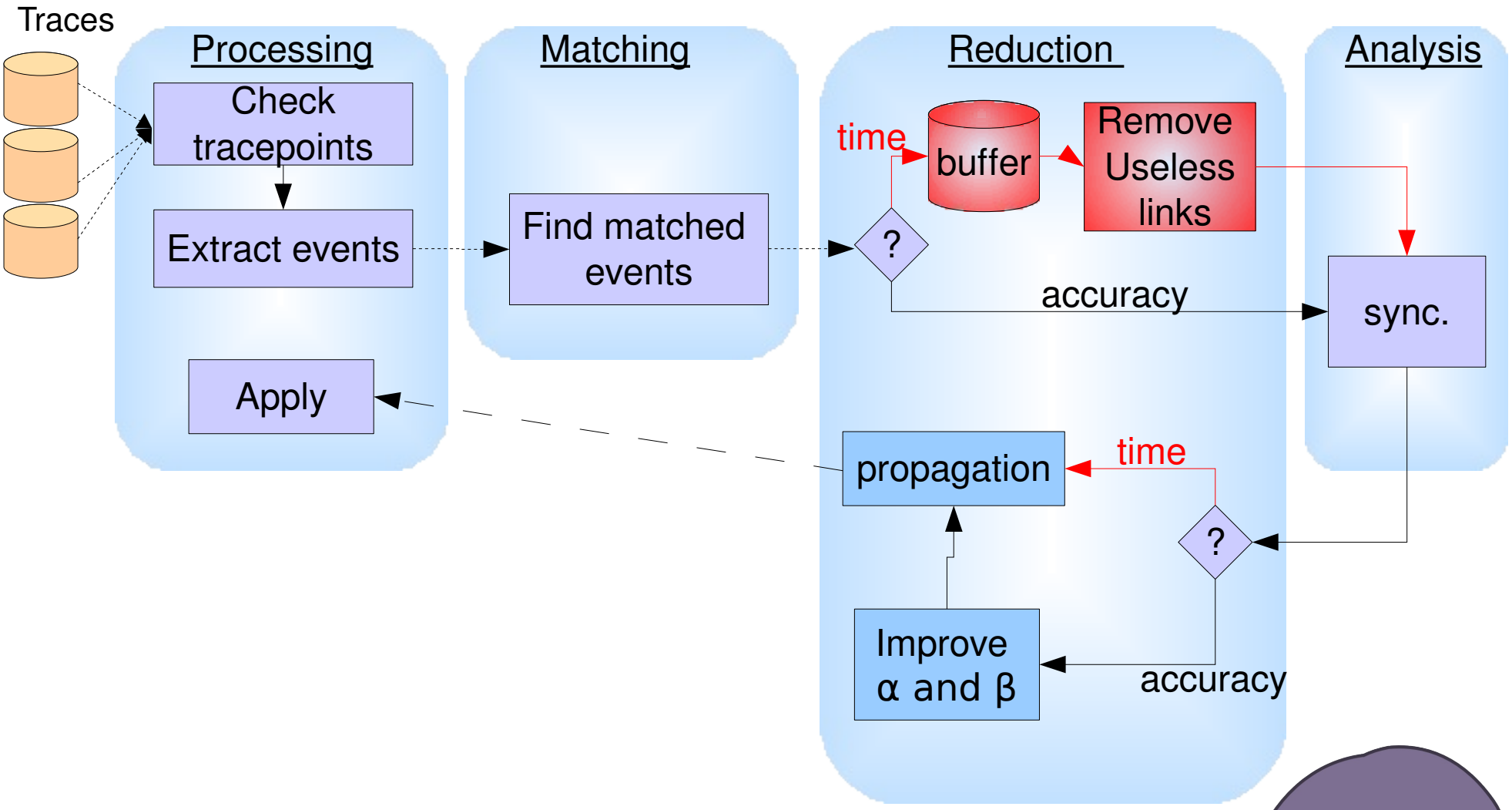
Linear regression



Convex Hull

$$clock1 = \alpha + \beta clock0$$

Architecture

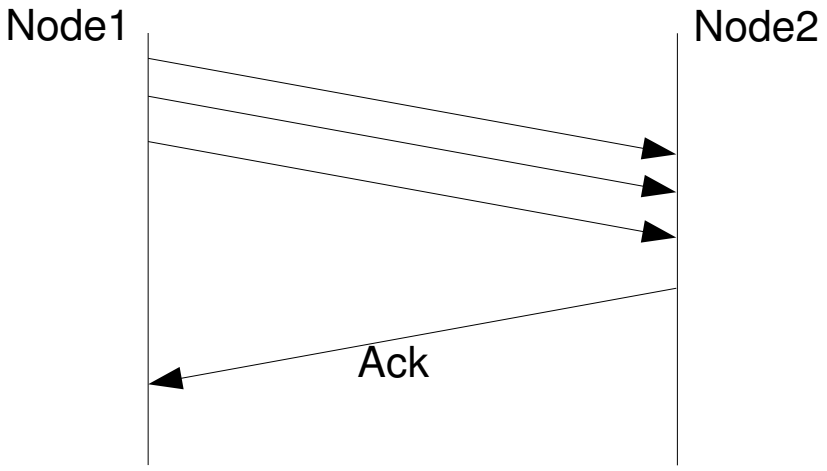


Processing module

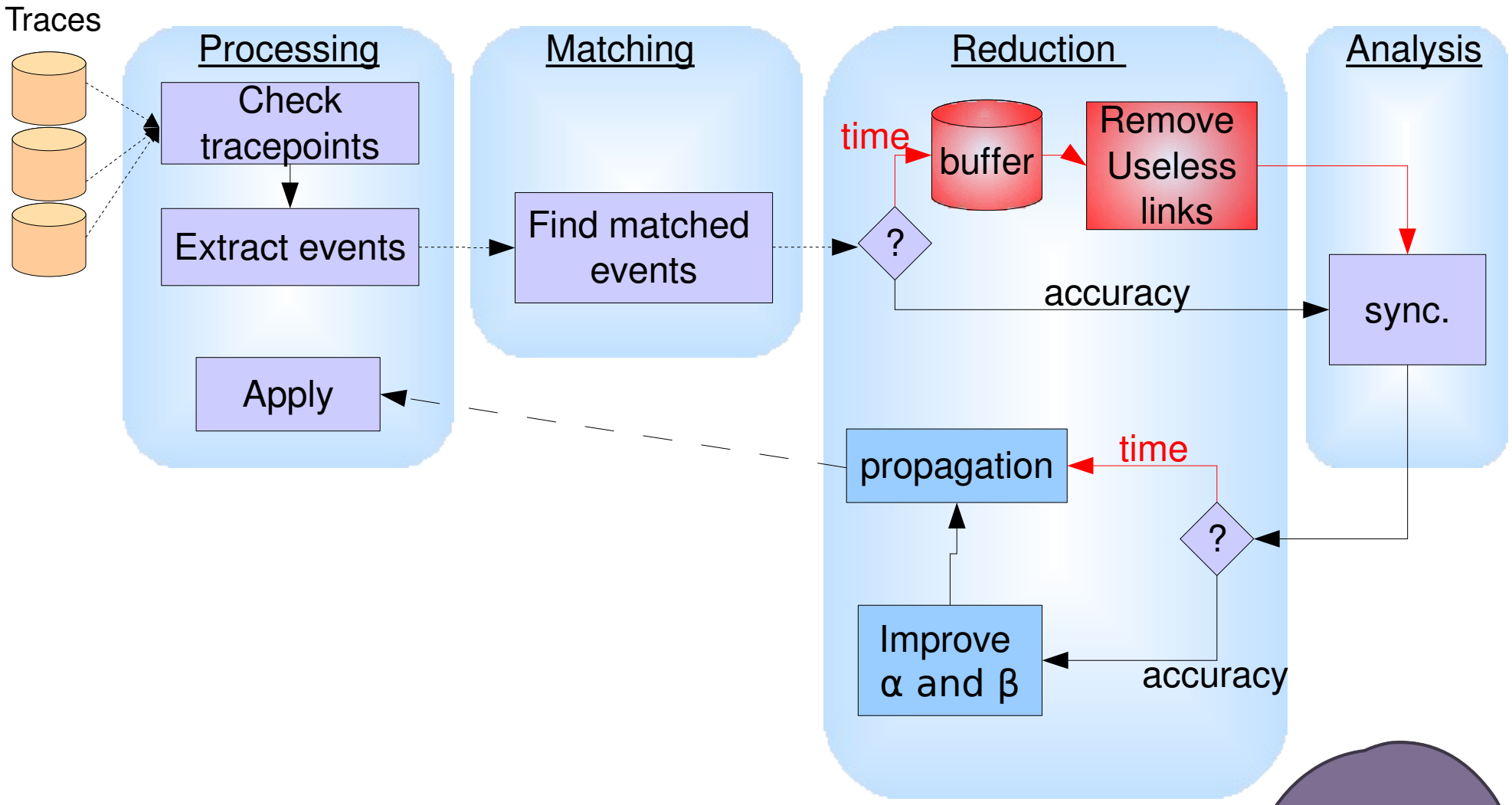
- Insure that there is network traffic exchanged between nodes
- Checks tracepoints are enabled
- Dispatch those events to the matching module

Matching module

- Forming events as a group
  - TCP: one to one
  - UDP: one to many
  - MPI: mix

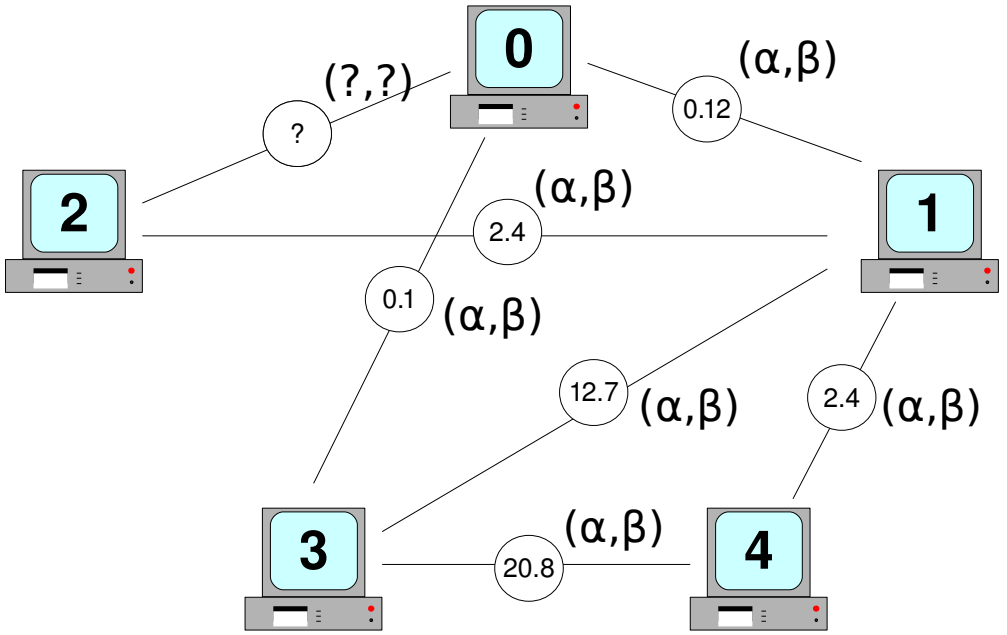


Architecture



Synchronization based on accuracy

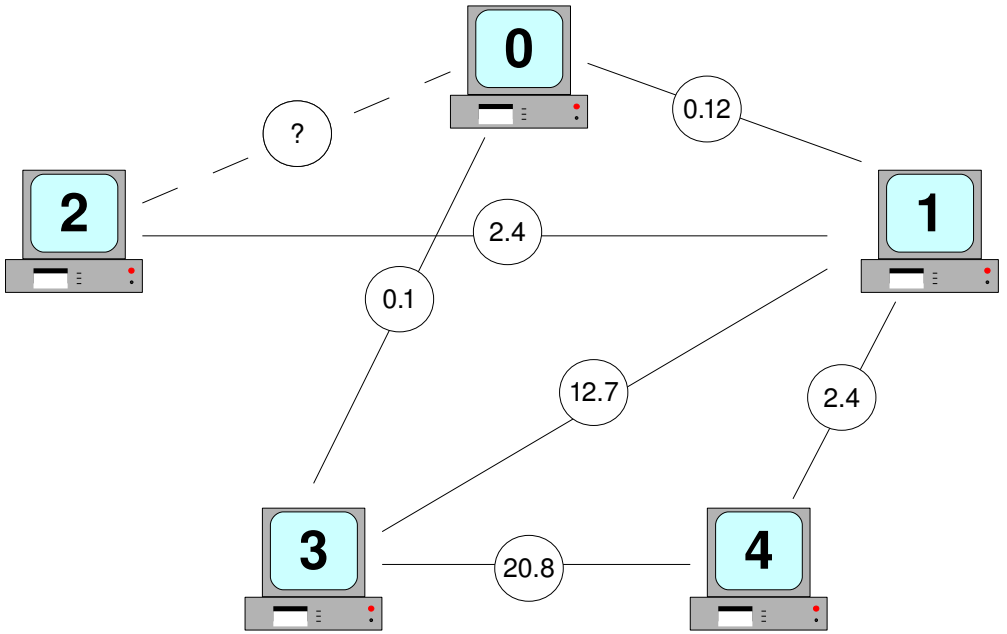
Phase 1) All connected nodes have to be synchronized and accuracies will be estimated



0 - 1	: sent 5	received 5
0 - 2	: sent 3	received 2
0 - 3	: sent 5	received 5
0 - 4	: sent 0	received 0
1 - 2	: sent 8	received 7
1 - 3	: sent 2	received 3
1 - 4	: sent 13	received 12
2 - 3	: sent 0	received 0
2 - 4	: sent 0	received 0
3 - 4	: sent 7	received 8

Synchronization based on accuracy

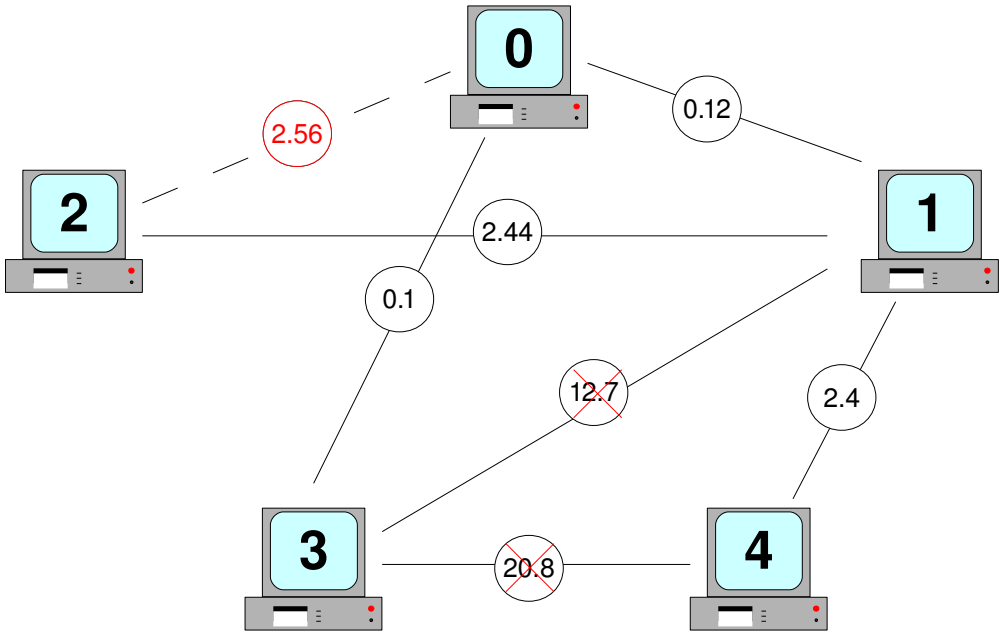
Phase 1) All connected nodes have to be synchronized and accuracies will be estimated



	N0	N1	N2	N3	N4
N0	0	0.12	?	0.1	?
N1	0.12	0	2.44	12.7	2.4
N2	?	2.44	0	?	?
N3	0.1	12.7	?	0	?
N4	?	2.4	?	?	0

Synchronization based on accuracy

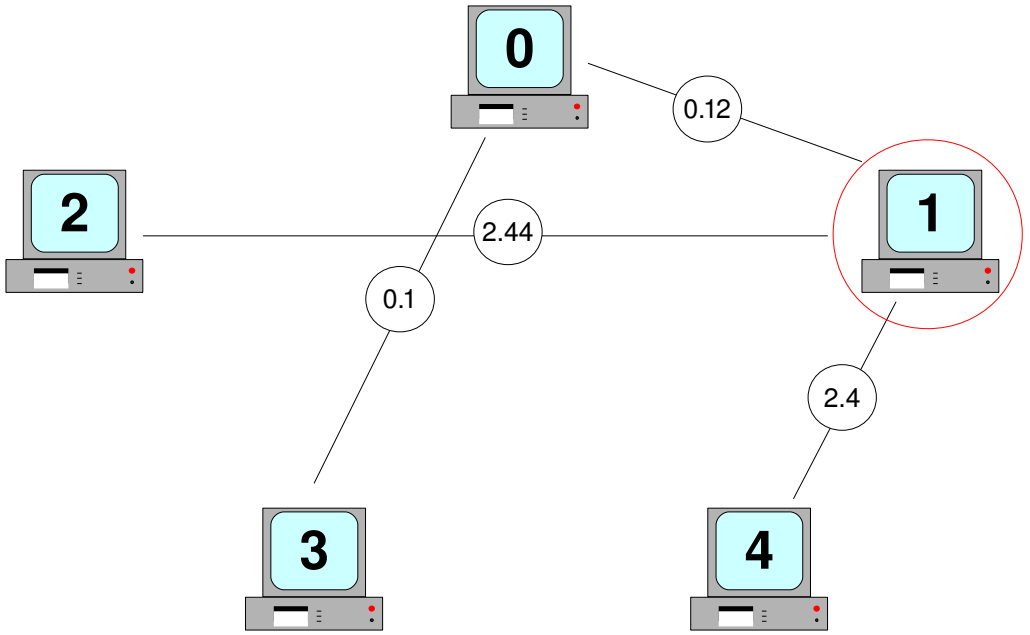
Phase 2) The best accuracy will be achieved by an indirect route combining the drift/offset of two links



	N0	N1	N2	N3	N4
N0	0	0.12	2.56	0.1	2.52
N1	0.12	0	2.44	0.22	2.4
N2	2.56	2.44	0	2.66	4.84
N3	0.1	0.22	2.66	0	2.62
N4	2.52	2.4	4.84	2.62	0

Synchronization based on accuracy

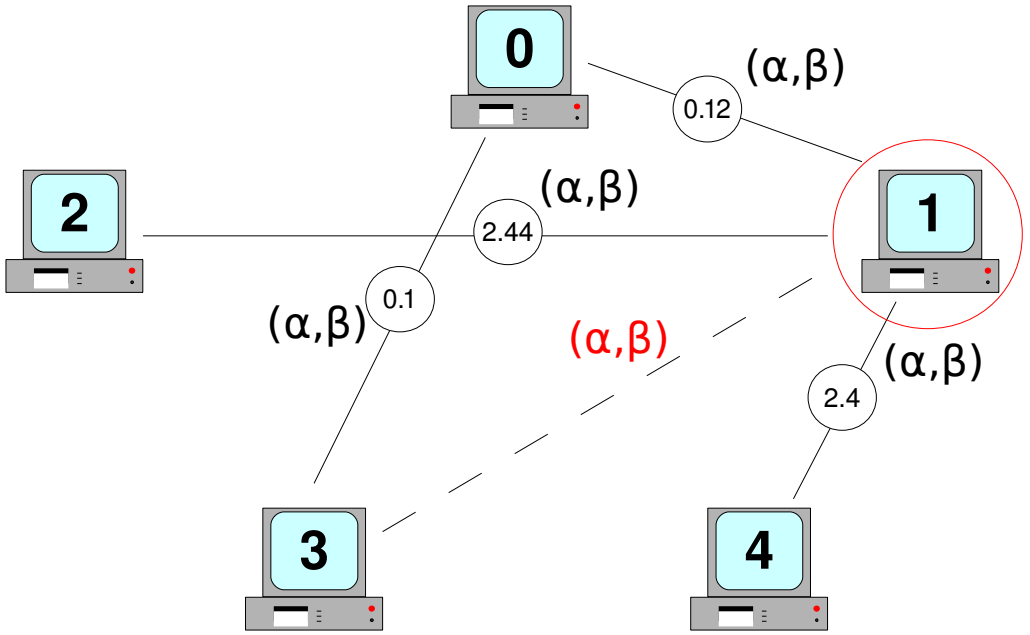
Phase 3) A reference is determined by using a shortest path search based on the accuracy of the approximation



	N0	N1	N2	N3	N4	Sum
N0	0	0.12	2.56	0.1	2.52	5.31
N1	0.12	0	2.44	0.22	2.4	5.19
N2	2.56	2.44	0	2.66	4.84	12.53
N3	0.1	0.22	2.66	0	2.62	5.63
N4	2.52	2.4	4.84	2.62	0	12.4

Synchronization based on accuracy

Phase 4) Recalculate the offset and drift based on reference node path

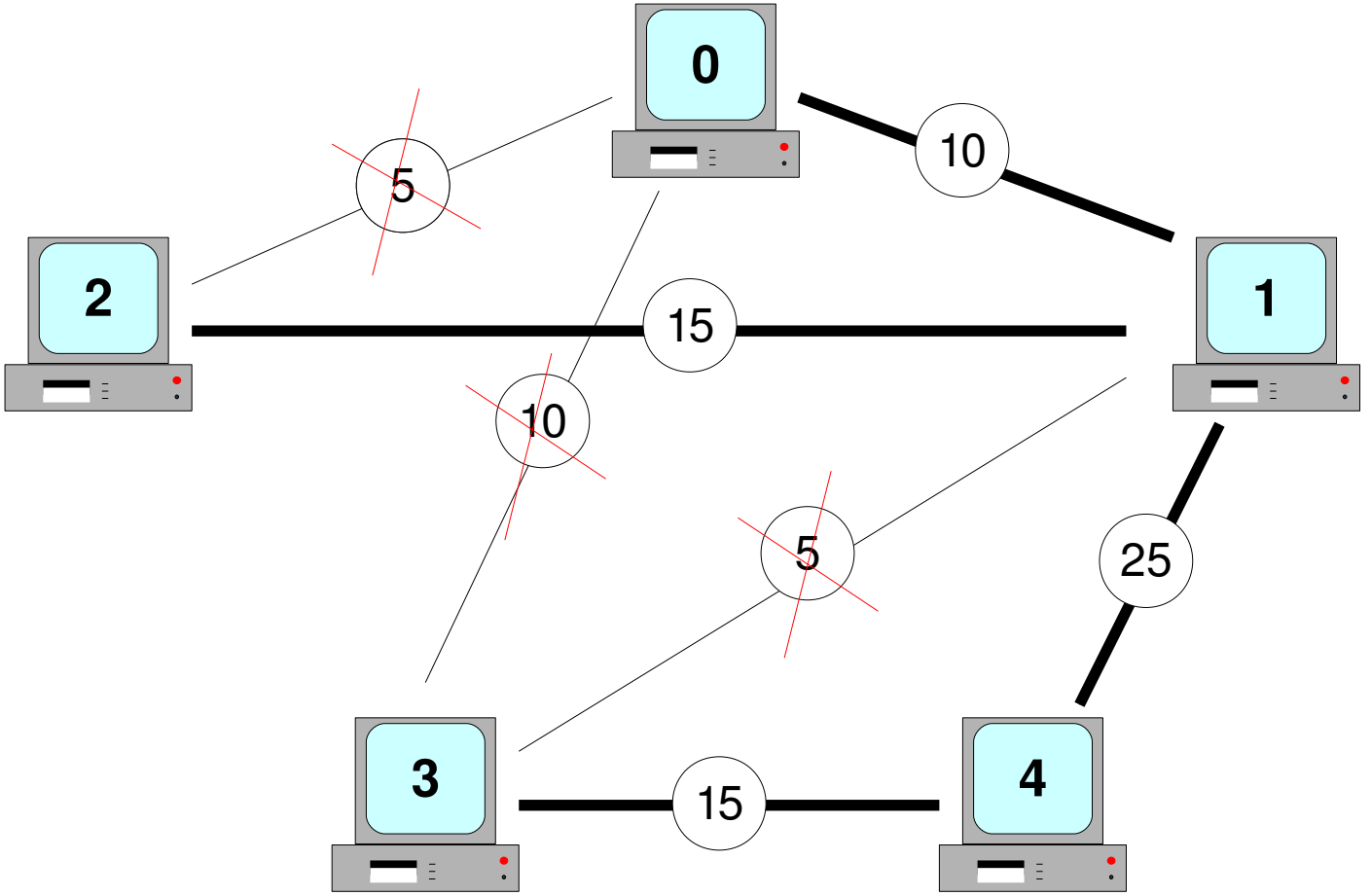


	drift	offset
N0	0.996567	5.3994e+9
N1	1	0
N2	0.561756	3.57928e+10
N3	-6.69055	4.60771e+11
N4	4.54918	-2.45135e+11

Total time 0.292372

Synchronization based on time

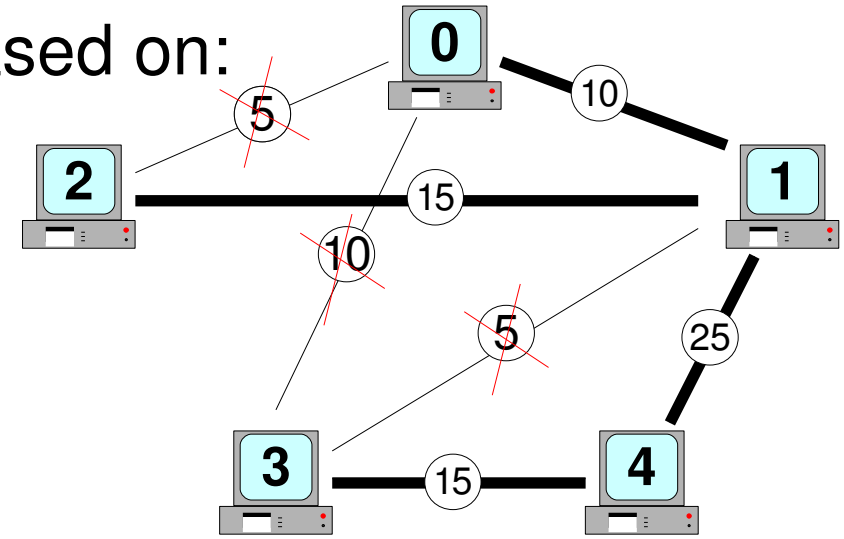
Phase 1) Find Maximum Spanning Tree based on the number of network traffic exchanged packet



Synchronization based on time

Phase 2) Find reference node based on:

- MST
- Main graph



	N0	N1	N2	N3	N4	Sum
N0	0	10	0	0	0	10
N1	10	0	15	0	25	50
N2	0	15	0	0	0	15
N3	0	0	0	0	15	15
N4	0	25	0	15	0	40

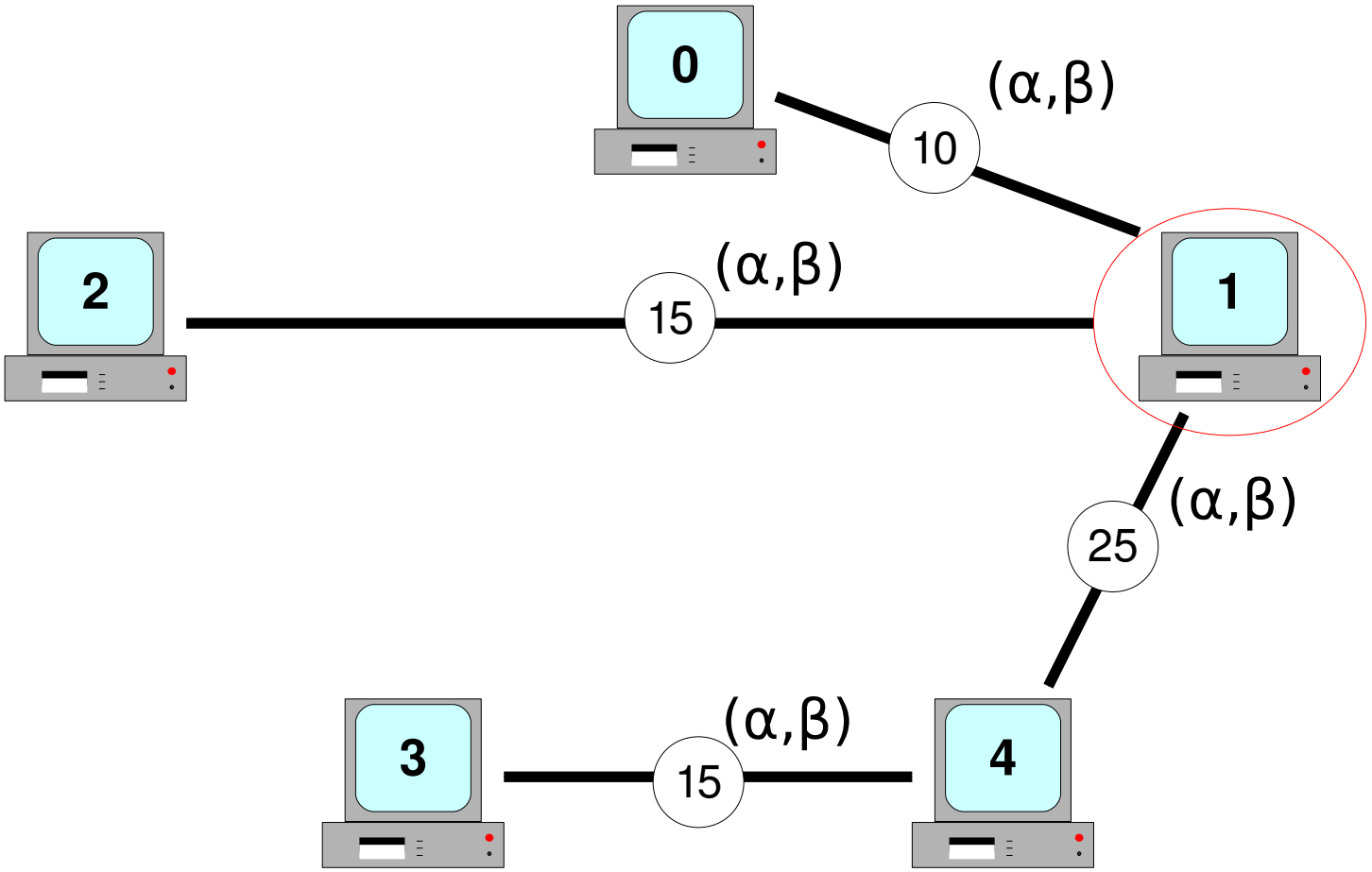
MST = {N1}

	N0	N1	N2	N3	N4	Sum
N0	0	10	5	10	0	25
N1	10	0	15	5	25	55
N2	5	15	0	0	0	15
N3	10	5	0	0	15	30
N4	0	25	0	15	0	40

Main graph = {N1}

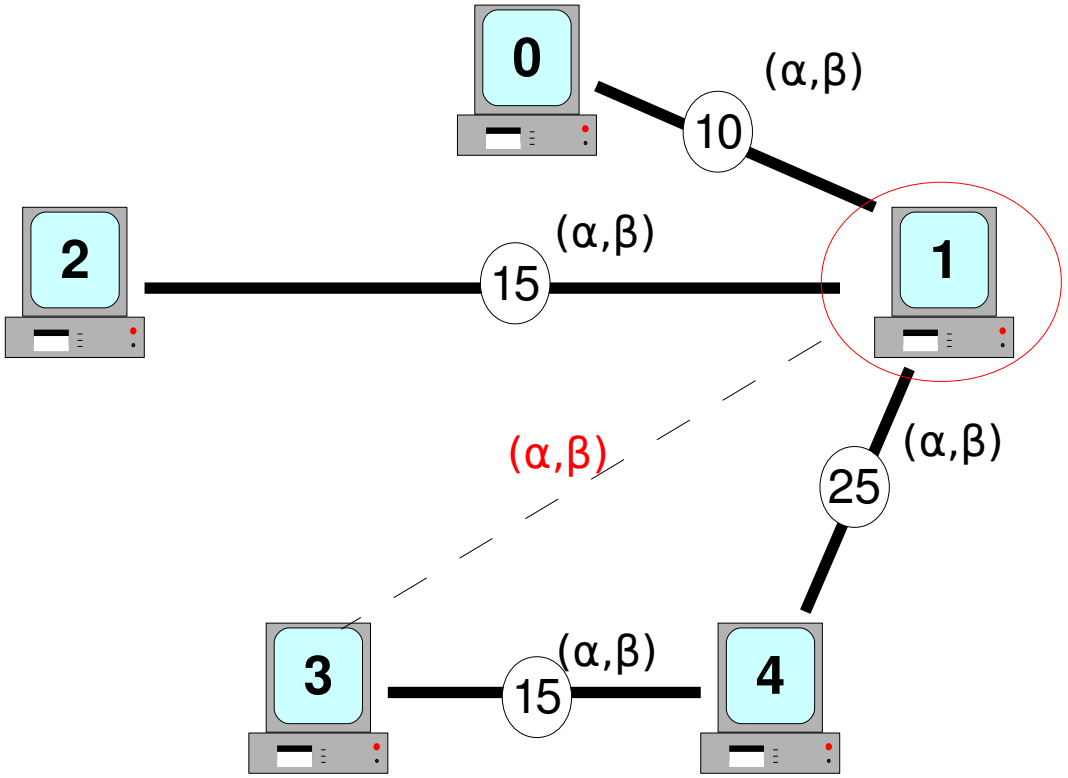
Synchronization based on time

Phase 3) Calculate offset and drift between nodes



Synchronization based on time

Phase 4) Recalculate the offset and drift based on reference node path



	drift	offset
N0	0.996567	5.3994e+9
N1	1	0
N2	0.561756	3.57928e+10
N3	-1.05667	2.08485e+11
N4	4.54918	-2.45135e+11

Total time 0.241690

Comparison Accuracy vs. Time

	Accuracy		Time	
	drift	offset	drift	offset
N0	0.996567	5.3994e+9	0.996567	5.3994e+9
N1	1	0	1	0
N2	0.561756	3.57928e+10	0.561756	3.57928e+10
N3	-6.69055	4.60771e+11	-1.05667	2.08485e+11
N4	4.54918	-2.45135e+11	4.54918	-2.45135e+11
Total time	**0.292372**		**0.241690**	

Comparison Accuracy vs. Time

	Phase 1	Phase 2	Phase 3	Phase 4
Accuracy	Sync connected nodes	Find better accuracy	Reference Node	Recaculate Drift & Offset
Time	MST	Reference Node	Sync connected nodes	Recaculate Drift & Offset

	Advantages	Disadvantages
Accuracy	very low buffering	time
Time	time	buffering

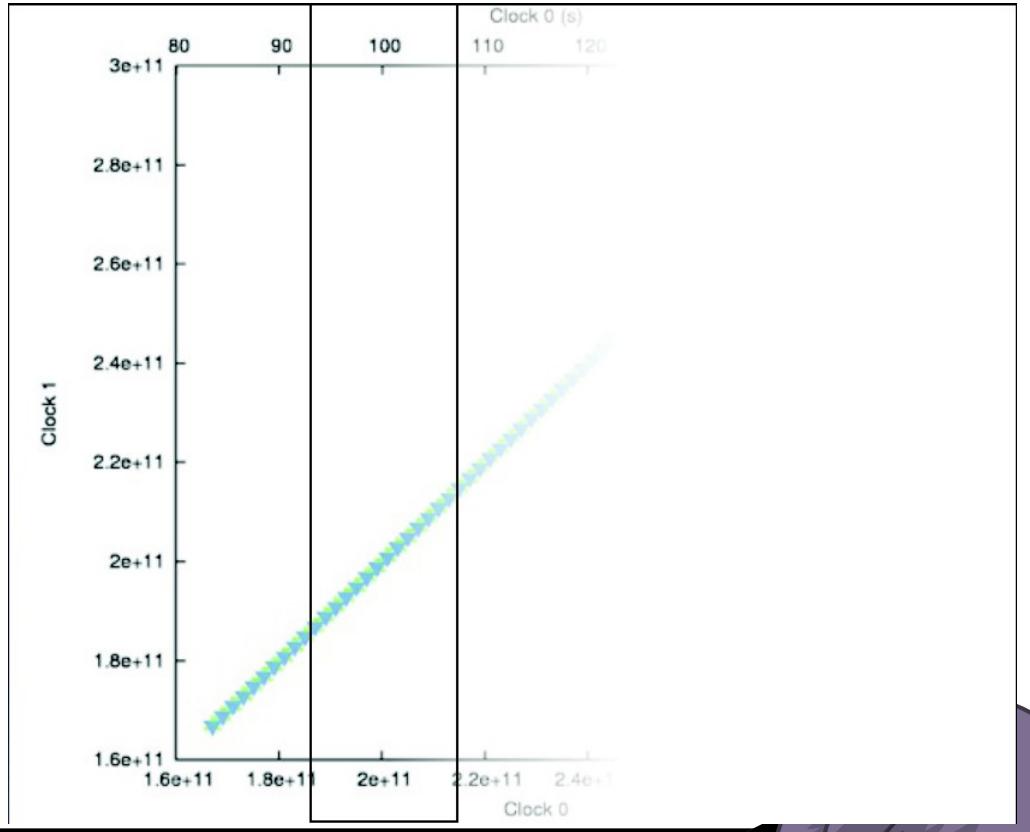
Future work

- Real computer cluster
- Effect of physical distances and network latency

Future work

Streaming Trace Synchronization

- Sliding window
  - Combine with convex hull
  - Window size



References

[1]	B. Poirier, R. Roy and M. Dagenais, “Accurate offline synchronization of distributed traces using kernel-level events, 2010.
[2]	J. H. Deschenes, M. Desnoyers and M. Dagenais. “Tracing Time Operating System State Determination,” The Open Software Engineering Journal, vol. 2, 2008, pp. 40-44.
[3]	A. D. Ksehmkalyani and M. Singhal, “Logical time,” in Distributed Computing: Principles, Algorithms, and Systems, 1st ed., USA: Cambridge University Press, 2008, pp. 50-84.
[4]	H. Khlifi and J. C. Gregorie, " Low-complexity offline and online clock skew estimation and removal," The International Journal of Computer and Telecommunications Networking, vol. 50, no. 11, pp. 1872-1884, 2006.
[5]	L. Chai, Q. Gao and D. K. Panda, “Understanding the Impact of Multi-Core Architecture in Cluster Computing: A Case Study with Intel Dual-Core System,” Proceedings of the Seventh IEEE International Symposium on Cluster Computing and the Grid, Rio De Janeiro, Brazil, 2007, pp. 471-478.
[6]	J. M. Jezequel and C. Jard, “Building a global clock for observing computations in distributed memory parallel computers,” Concurrency: Practice and Experience, vol 2, no. 1, 1996, pp. 71-89
[7]	E. Betti, M. Cesati, R Gioiosa and F. Piermaria, “A global operating system for HPC clusters,” IEEE International Conference on Cluster Computing and Workshops, 2009.
[8]	R. Sirdey and F. Maurice, “A linear programming approach to highly precise clock synchronization over a packet network,” 4OR: A Quarterly Journal of Operations Research, vol. 6, no. 4, 2008, pp. 393-401.
[9]	C. N. Keltcher, K. J. McGrath, A. Ahmed, and P. Conway, “The amd opteron processor for multiprocessor servers,” IEEE Micro, vol. 23, no. 2, 2003, pp. 66–76.

Thank you