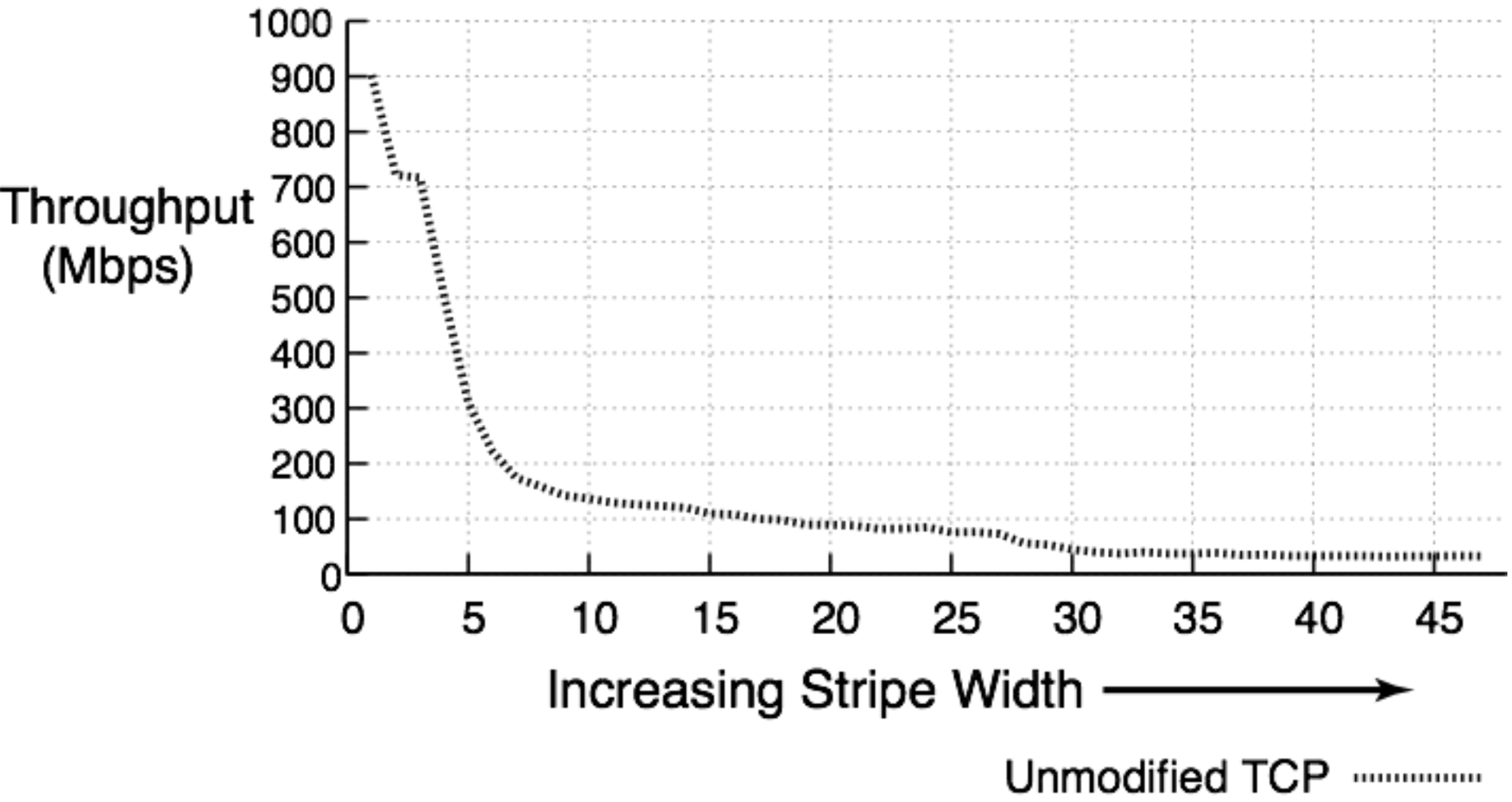
Solving TCP Incast in Cluster Storage Systems

Vijay Vasudevan

Hiral Shah, Amar Phanishayee, Elie Krevat
David Andersen, Greg Ganger, Garth Gibson

Carnegie Mellon University

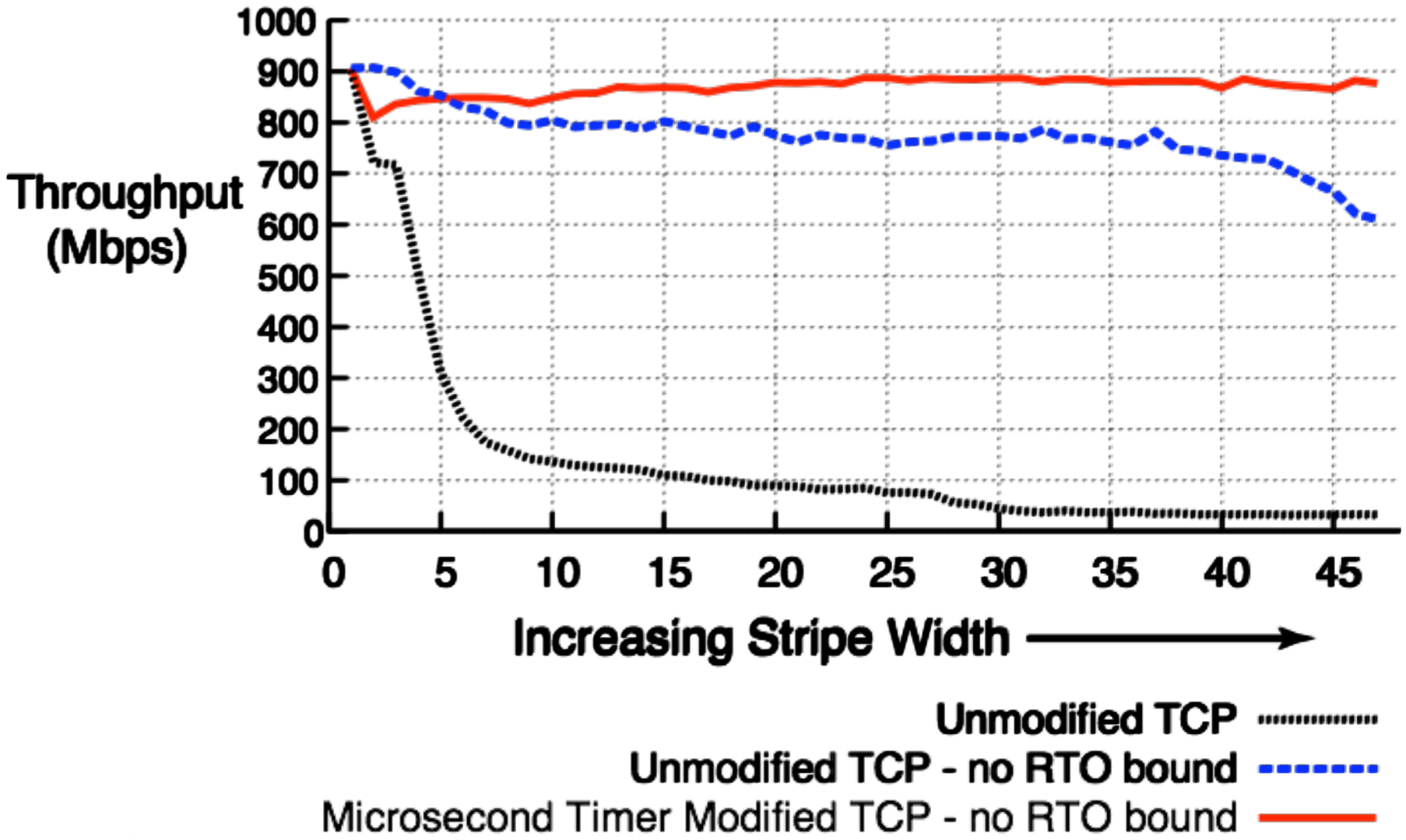
TCP Throughput Collapse



Cluster Environment
1Gbps Ethernet
100us Delay
200ms RTO
S50 Switch
1MB Block Size

- [Nagle04] called this Incast; provided app-level workaround
  - Cause of throughput collapse: 200ms TCP timeouts
- Promising Solution: Reduce Retransmission Timeout (RTO)
  - Is it effective, practical and safe in the real world?

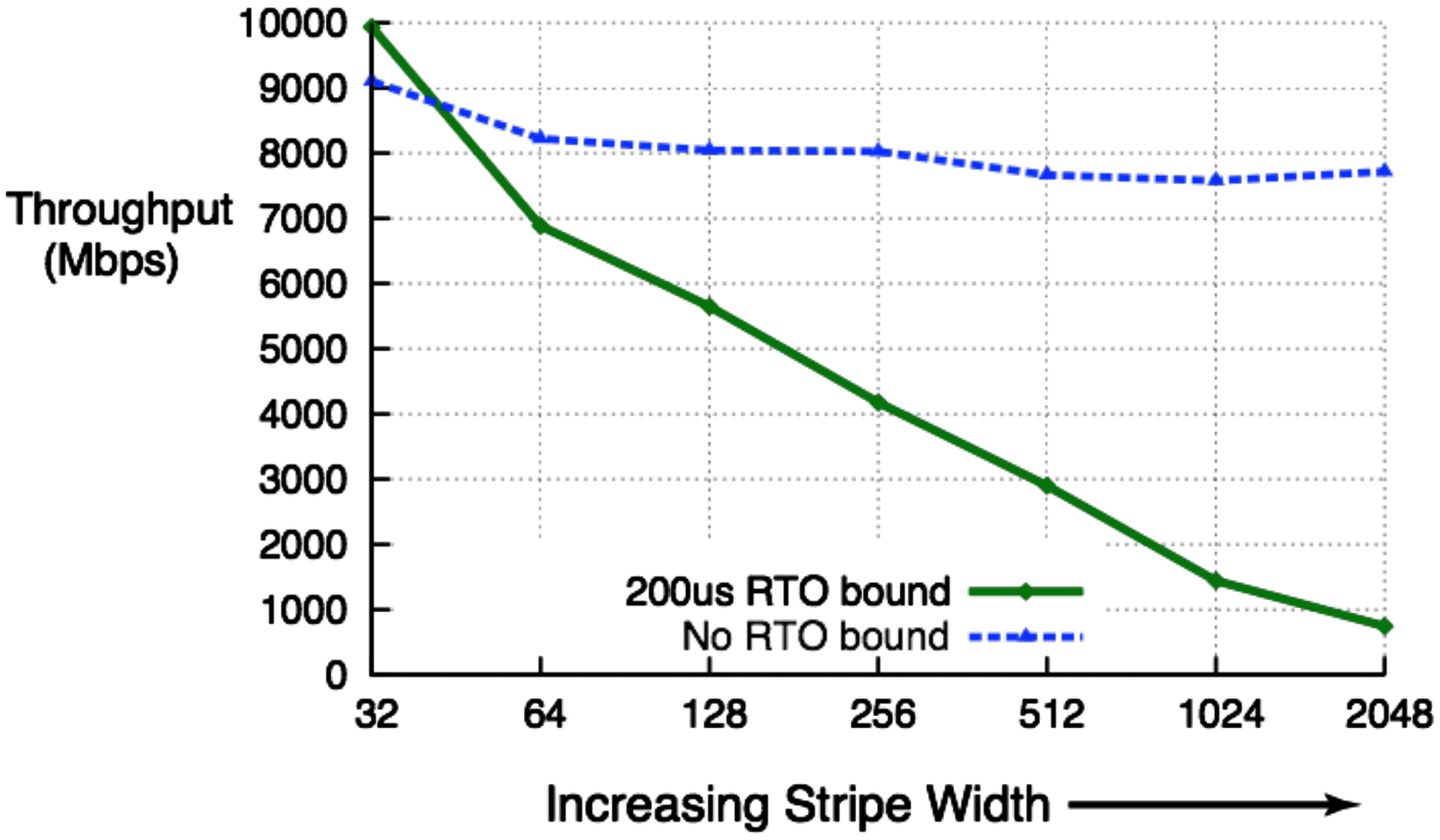
Mitigating Incast



Cluster Environment
1Gbps Ethernet
100us Delay
S50 Switch
1MB Block Size

- Eliminating RTO bound: 5ms timeouts - clock granularity limit
- Eliminating RTO bound + Microsecond TCP timeouts avoids Incast
  - Single line, server only change

The need for Microsecond Timeouts



Simulation Environment
10Gbps Ethernet
20us Delay
40MB Block Size

- Future datacenters: More bandwidth, less delay, more servers
  - Retransmission timeouts should not be bounded

Conclusions

Technique	Effectiveness
Unmodified TCP	1-10% Throughput
Unmodified TCP, Eliminate RTO bound (one-line, server change)	70-80% Throughput (current networks) 10-70% Throughput (future networks)
Microsecond TCP timers Eliminate RTO bound	80-100% Throughput

Latest results and analysis can be found in CMU PDL tech report: http://tinyurl.com/incast