Power Consumption Due to Data Movement in Distributed Programming Models

Siddhartha Jana (UH), Oscar Hernandez (ORNL),
Stephen Poole (ORNL), Barbara Chapman (UH)

Green HPC – EuroPar14
August 28th, 2014

Outline

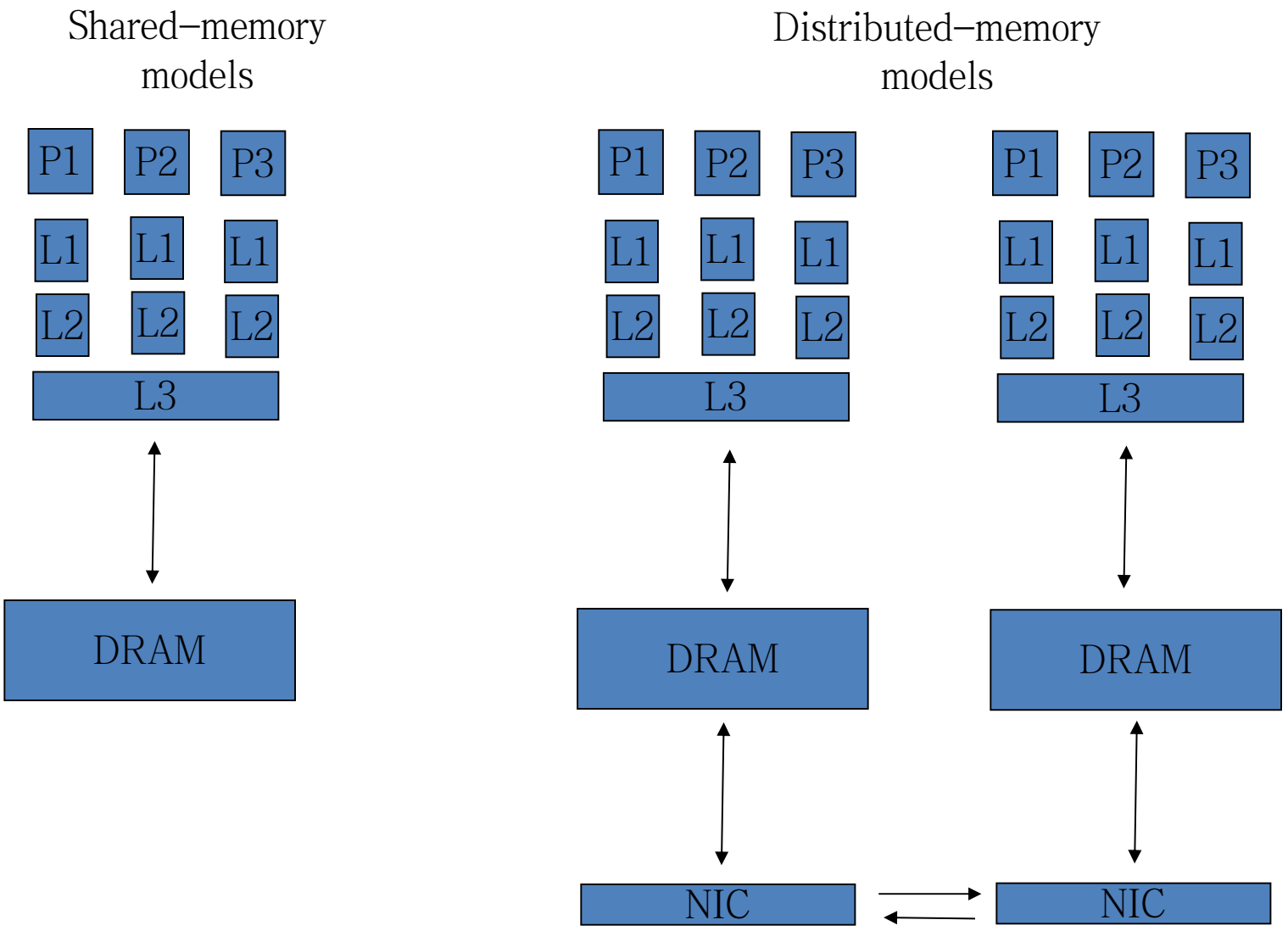
- Motivation
  - Impact of data movement

- Factors impacting energy and power consumption across the h/w - s/w stack
  - Design of data-transfer protocols
  - Choice of transport layer

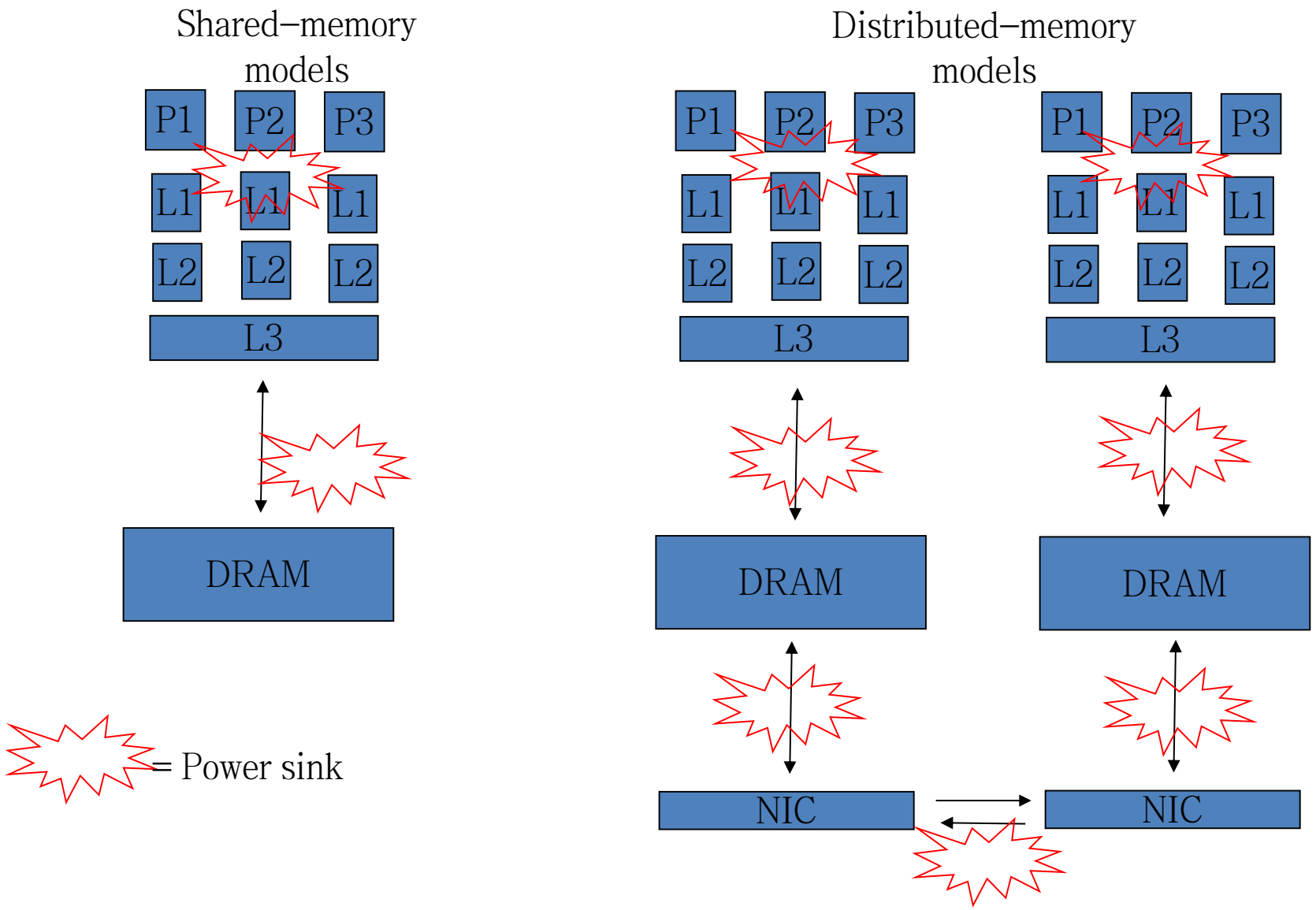
- Impact of remote data transfers
  - Experimental setup
  - Observations

- Conclusions and Future Work

Impact of Data Movement



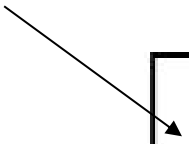
Impact of Data Movement



Factors impacting energy and power consumption

Impact across the hardware-software stack

MPI_Send and MPI_Recv



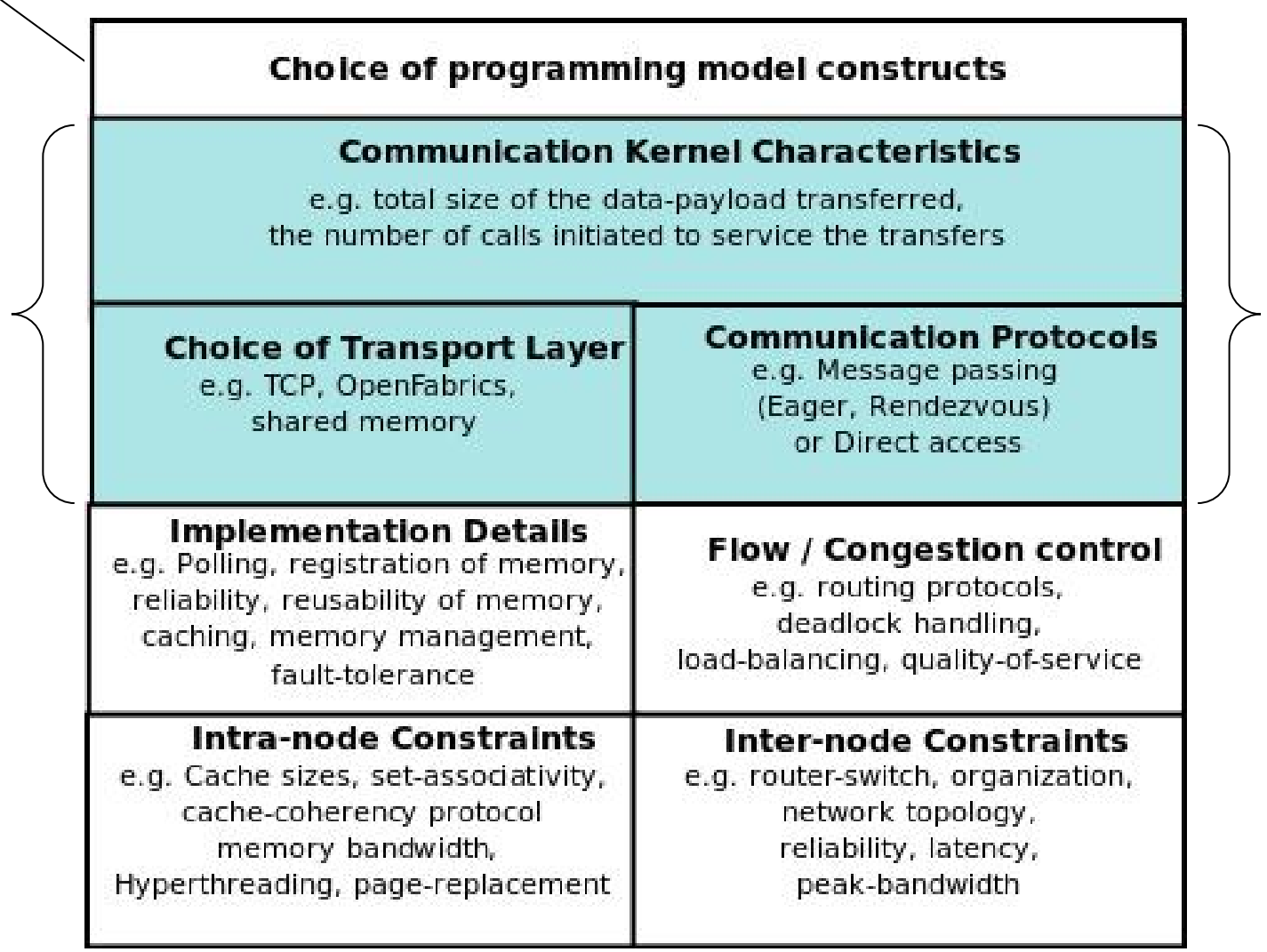
Choice of programming model constructs	
Communication Kernel Characteristics e.g. total size of the data-payload transferred, the number of calls initiated to service the transfers	
Choice of Transport Layer e.g. TCP, OpenFabrics, shared memory	**Communication Protocols** e.g. Message passing (Eager, Rendezvous) or Direct access
Implementation Details e.g. Polling, registration of memory, reliability, reusability of memory, caching, memory management, fault-tolerance	**Flow / Congestion control** e.g. routing protocols, deadlock handling, load-balancing, quality-of-service
Intra-node Constraints e.g. Cache sizes, set-associativity, cache-coherency protocol memory bandwidth, Hyperthreading, page-replacement	**Inter-node Constraints** e.g. router-switch, organization, network topology, reliability, latency, peak-bandwidth

Factors impacting energy and power consumption

Impact across the hardware-software stack

MPI_Send and MPI_Recv

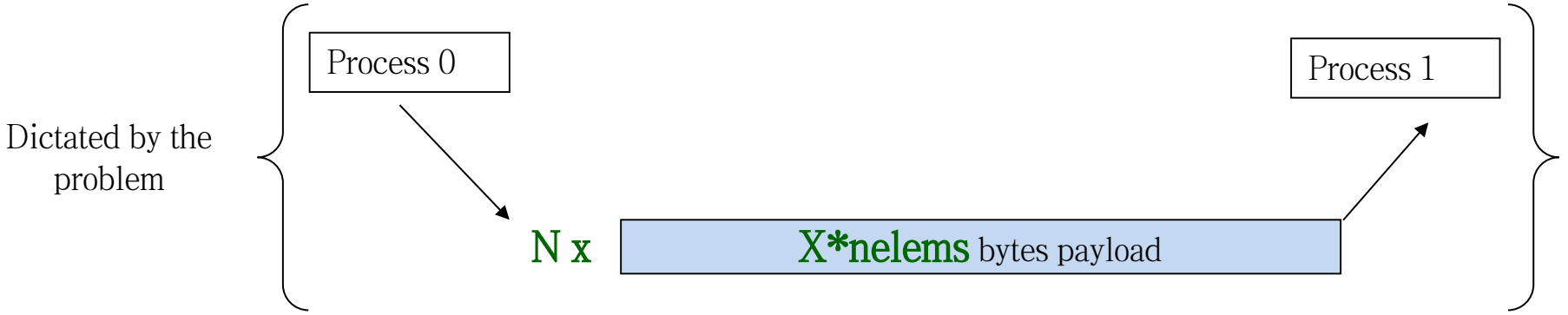
Scope of this talk



Energy and power efficiency - software stack

Remote data transfers

Consider the following case for a given problem:

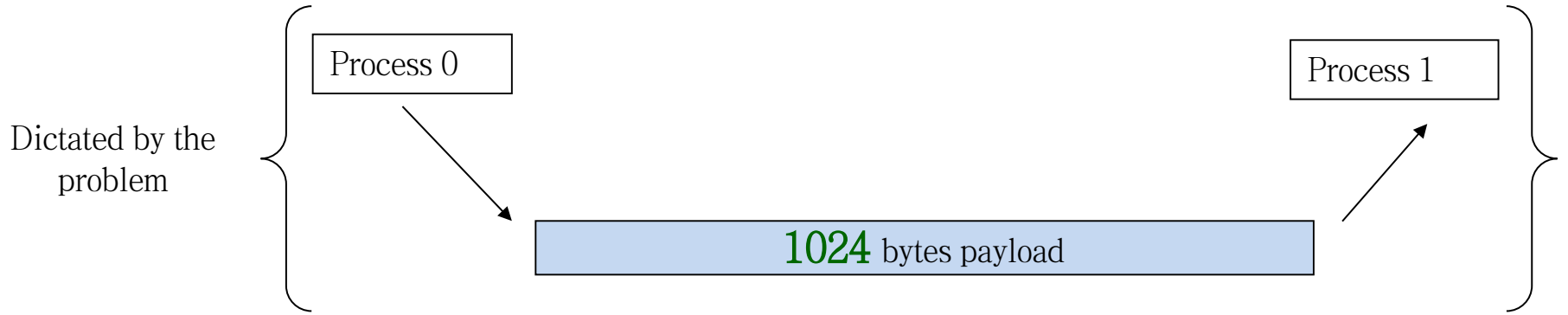


N ***** MPI_Send(const void *buf, int **count**, MPI_Datatype **datatype**, int dest,..)

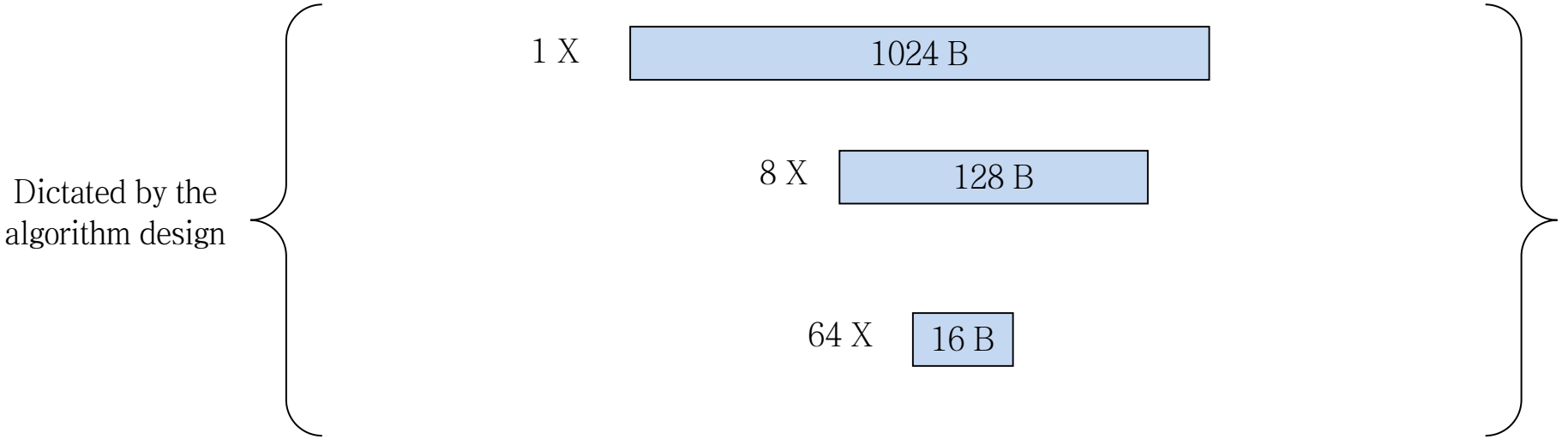
Energy and power efficiency - software stack

Identifying factors at the application layer

Consider the following case for a given problem:



This can be transferred as:



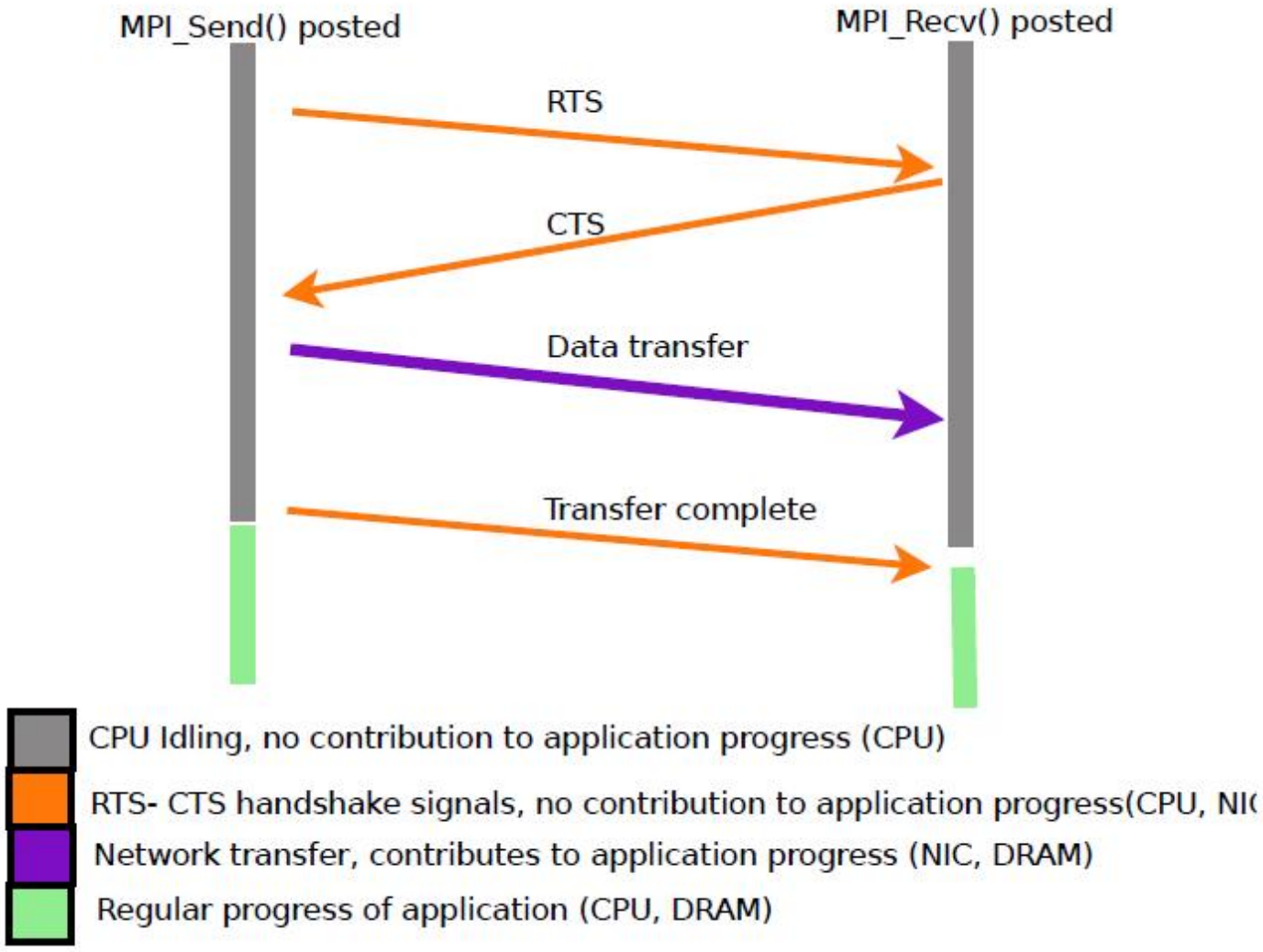
Middleware design factors

- TCP
    - OS dependant
    - impact on energy signatures

- OFED (OpenFabrics Enterprise Distribution)
    - OpenIB
    - RDMA capable
    - OS kernel bypass

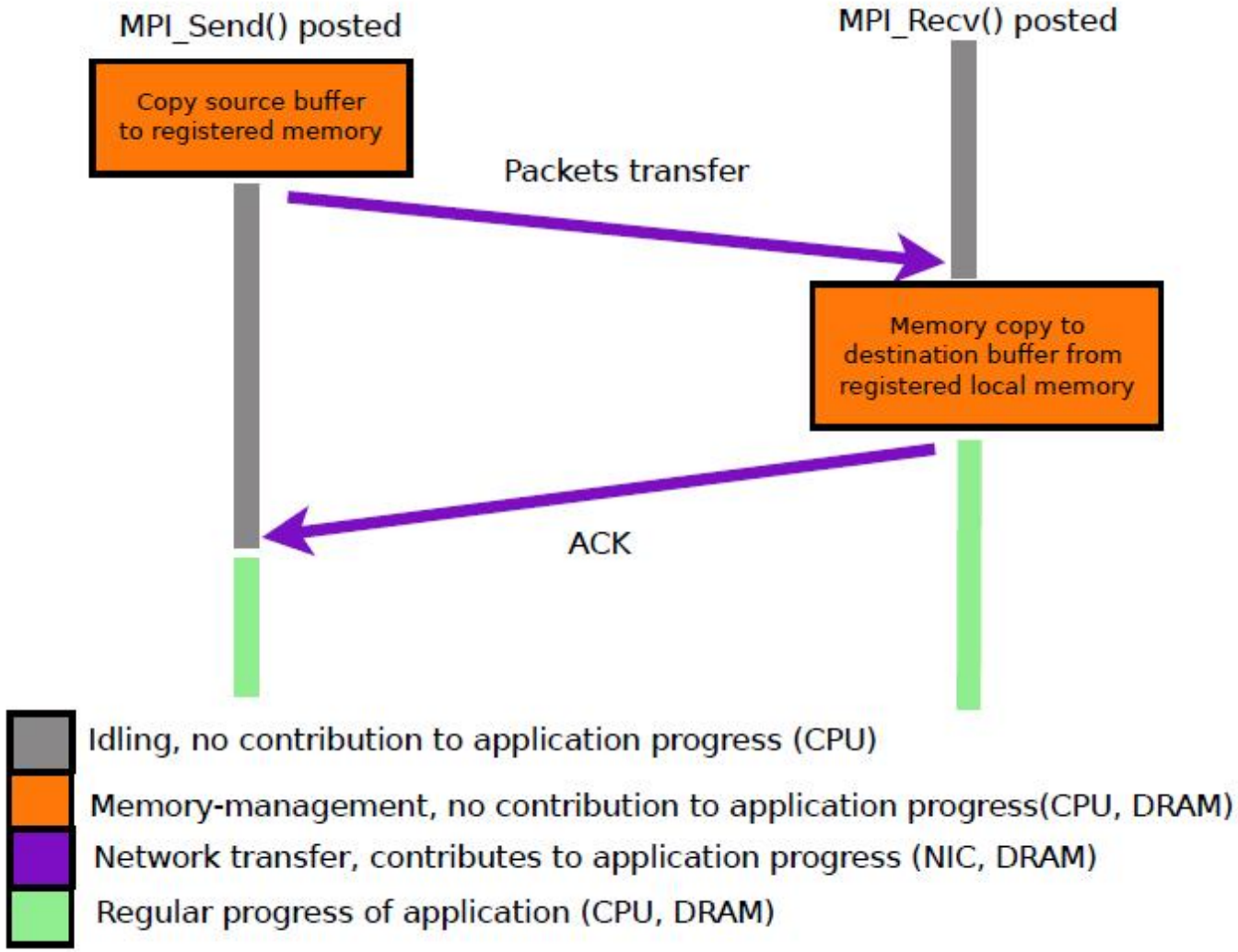
Data Transfer Protocols

Rendezvous protocol



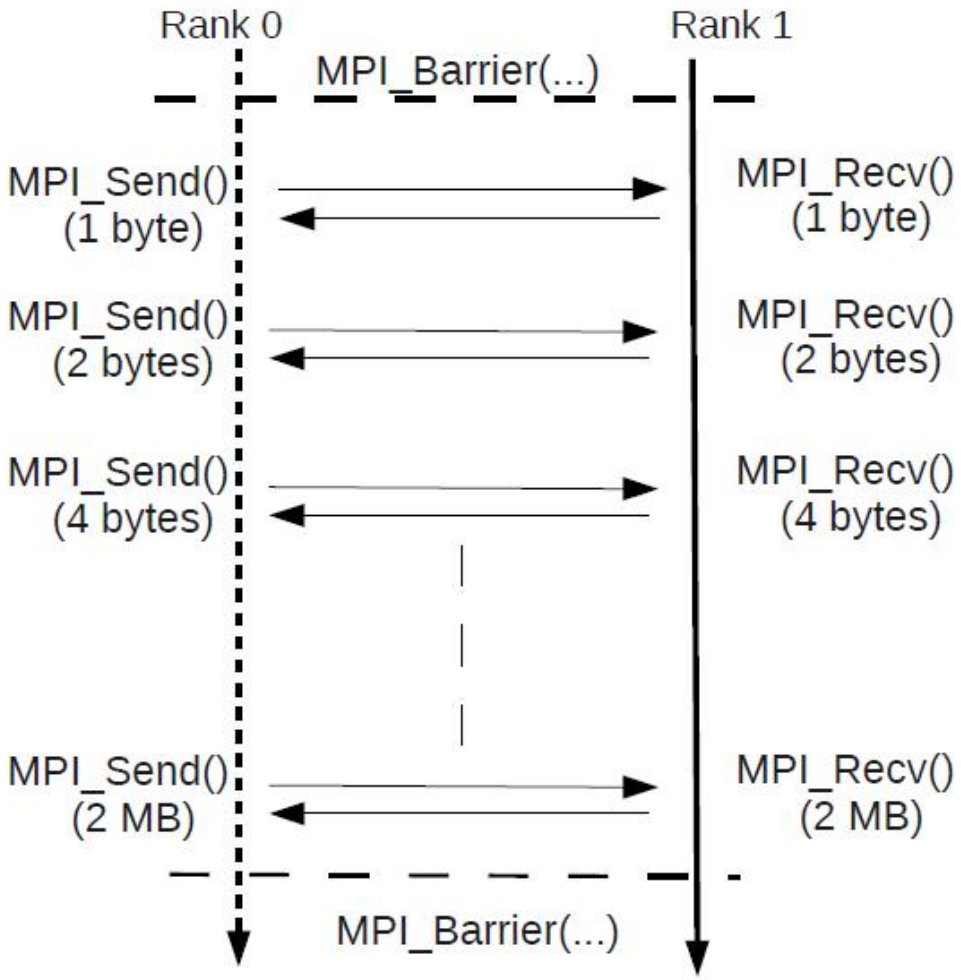
Data Transfer Protocols

Eager protocol



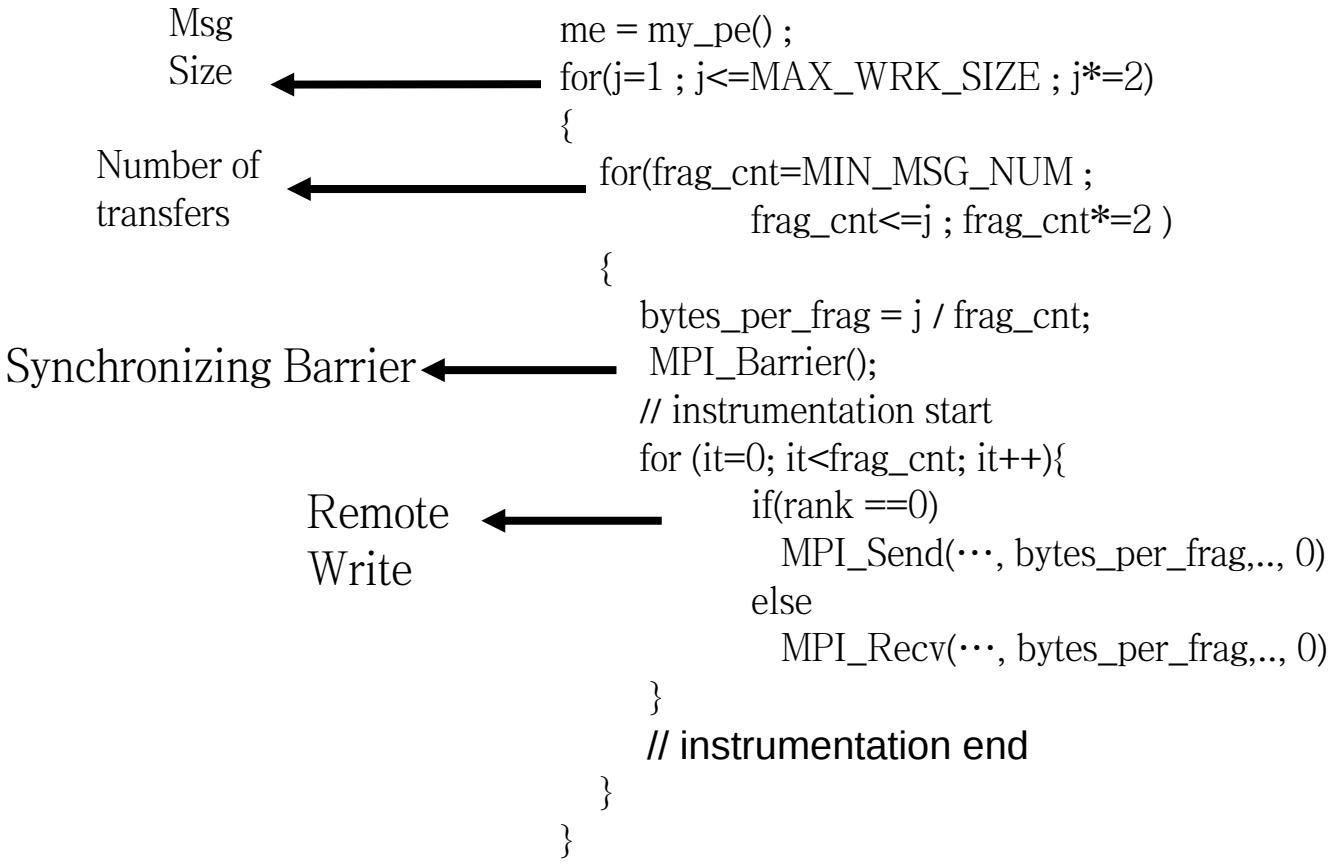
The Impact of Remote Transfers

Microbenchmarks



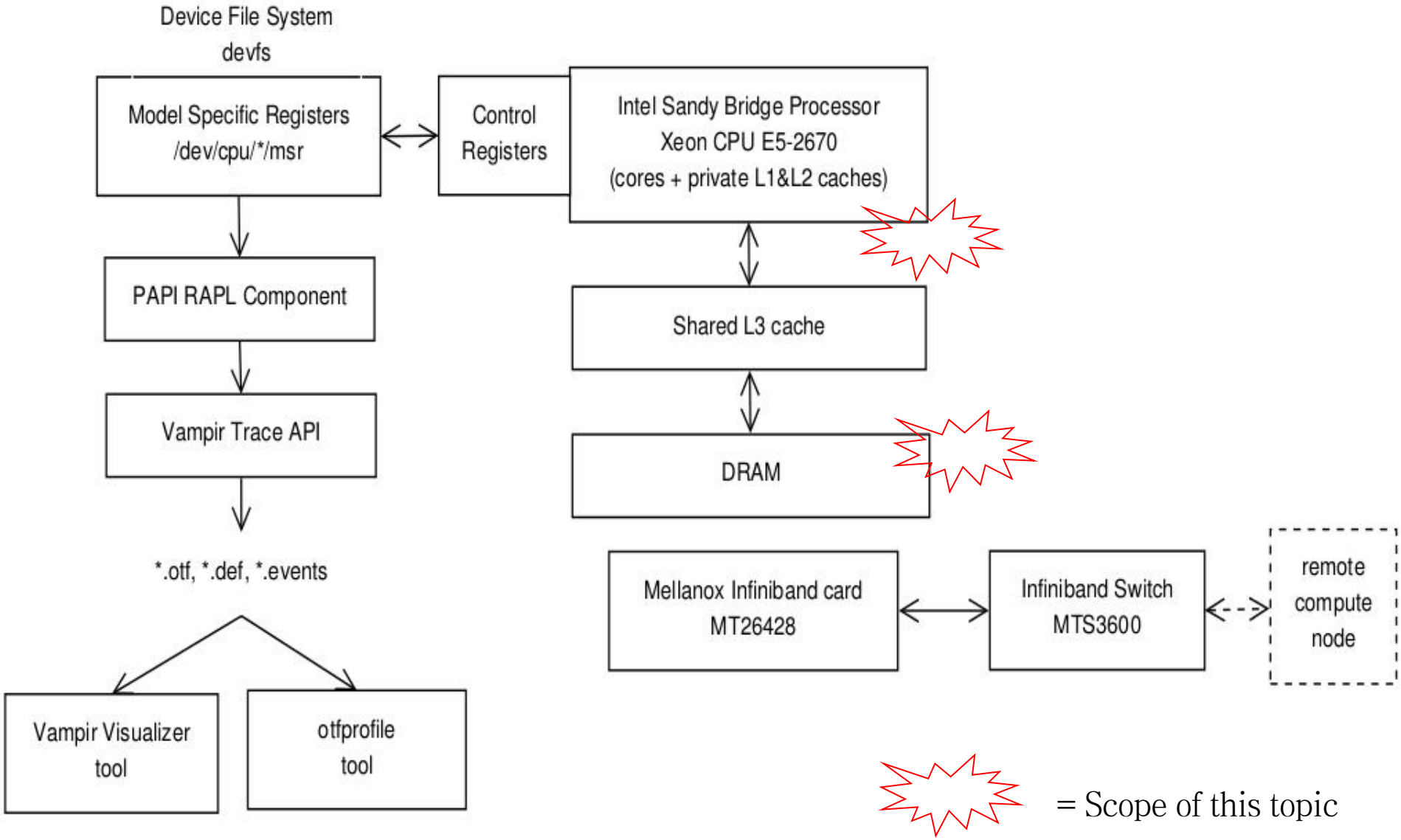
The Impact of Remote Transfers

Microbenchmarks



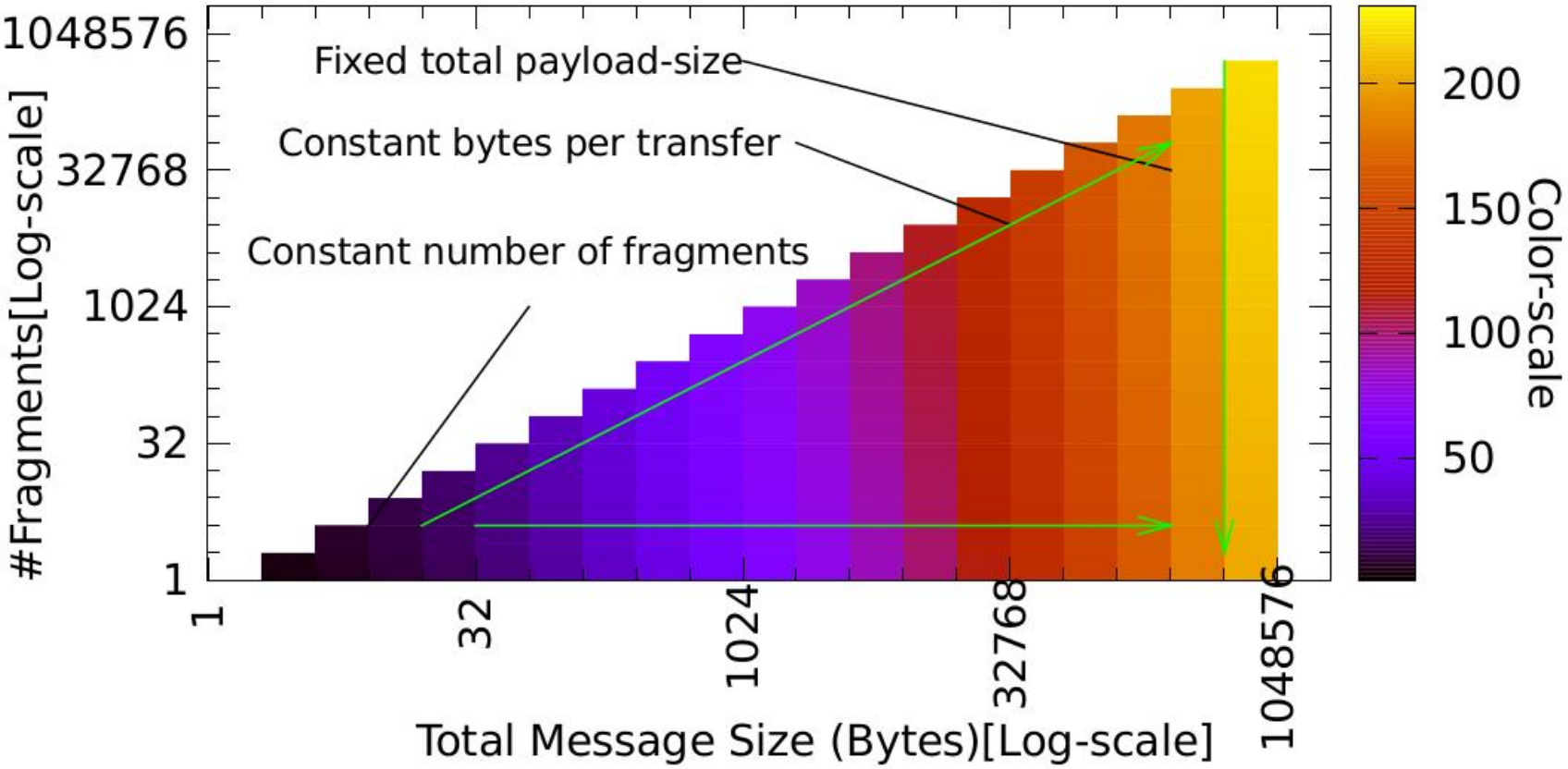
The Impact of Remote Transfers

Experimental Setup

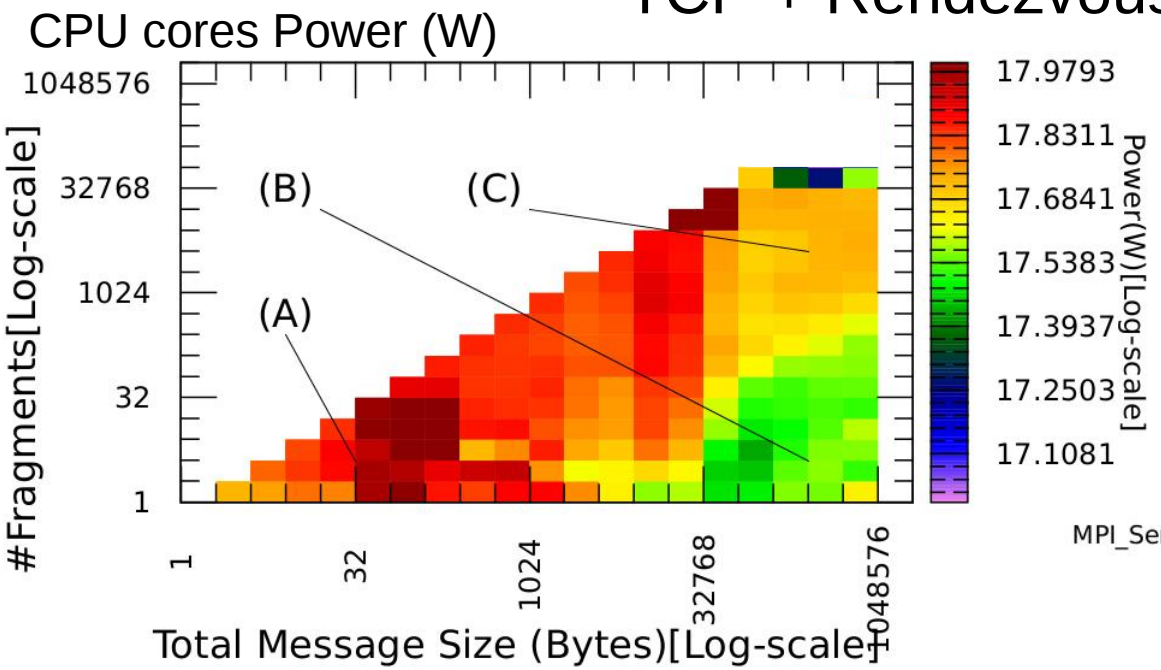


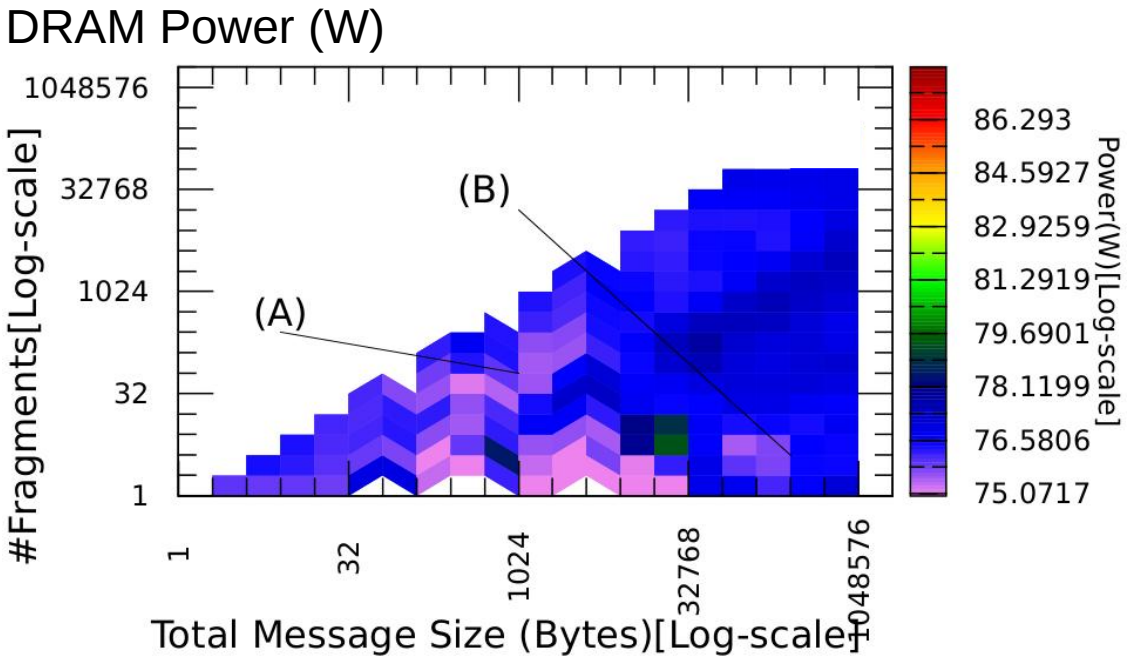
The Impact of Remote Transfers

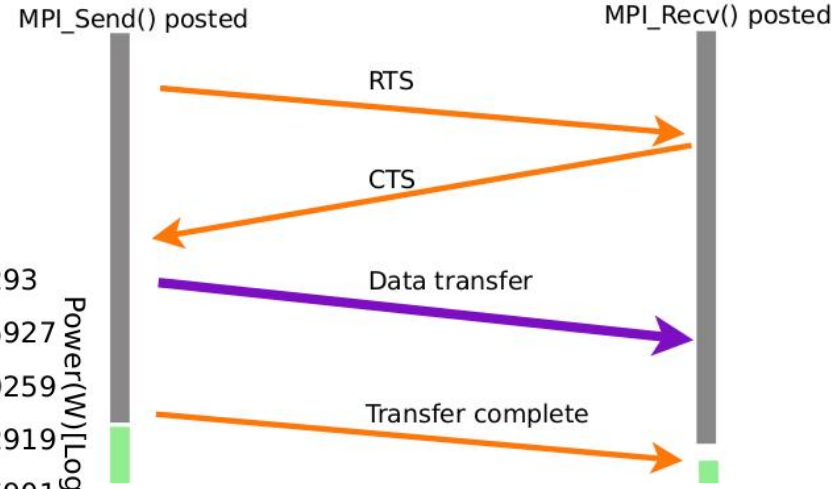
A note on reading the plots



TCP + Rendezvous protocol (sender)

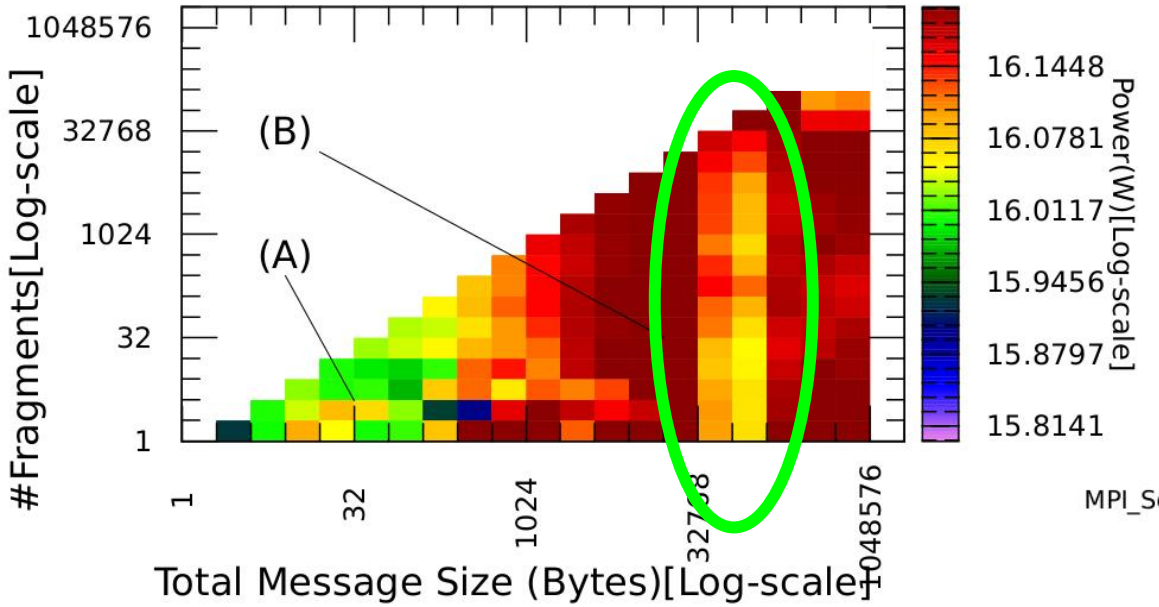




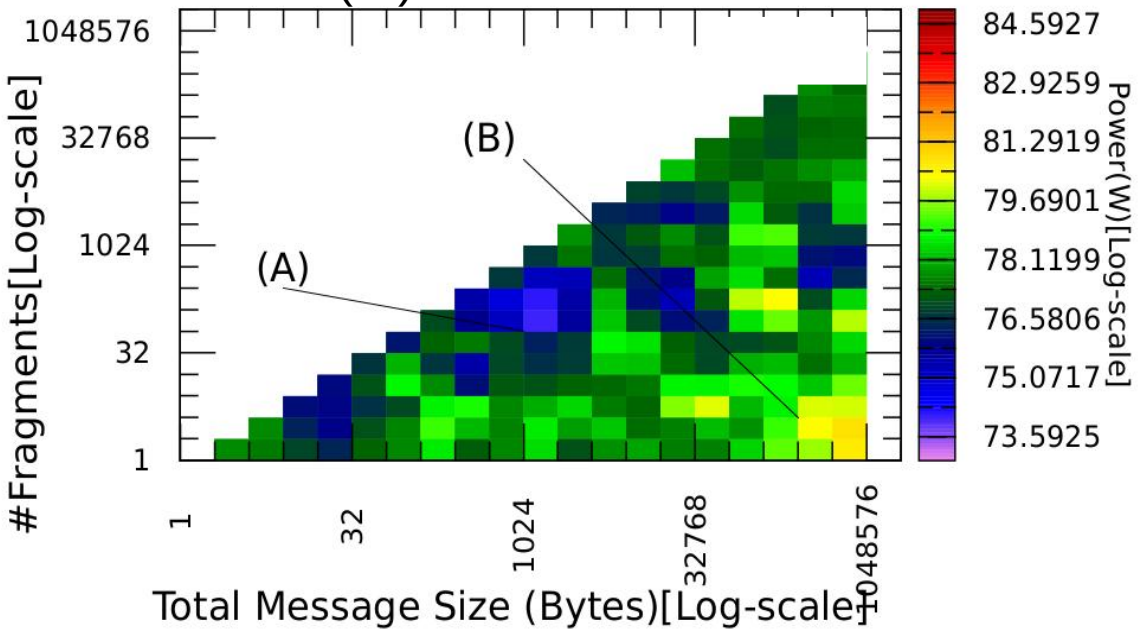


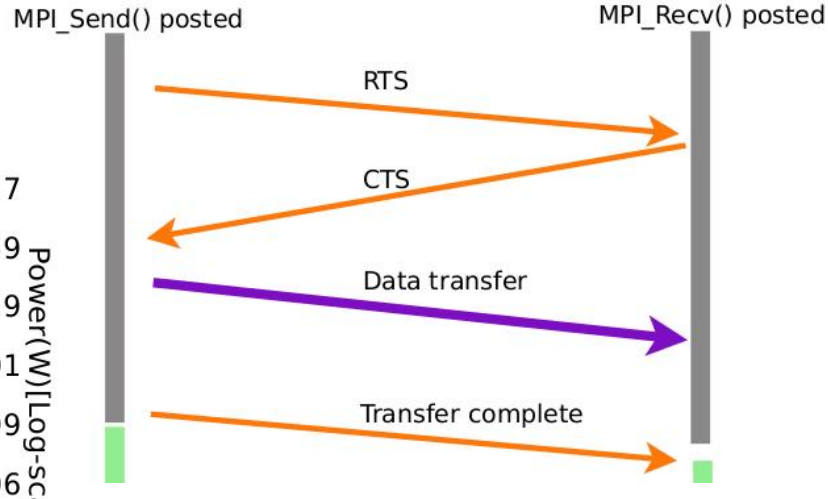
TCP + Rendezvous protocol (receiver)

CPU cores Power (W)



DRAM Power (W)





TCP + Rendezvous protocol

Lessons learned

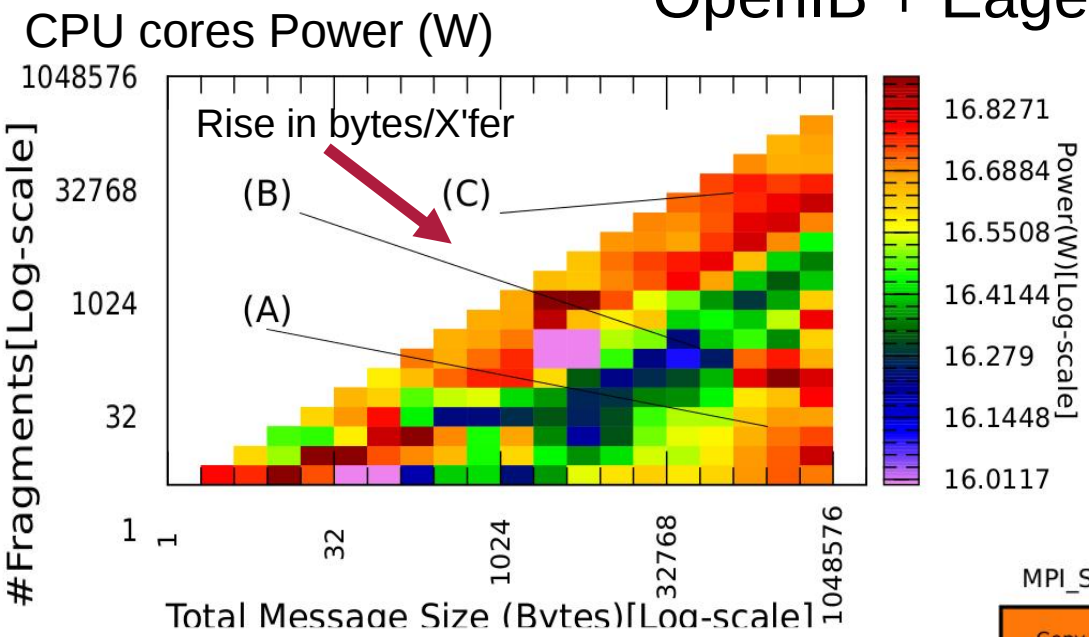
- (Sender:) Relying on the traditional protocol for small sized transfers => higher CPU core power consumption

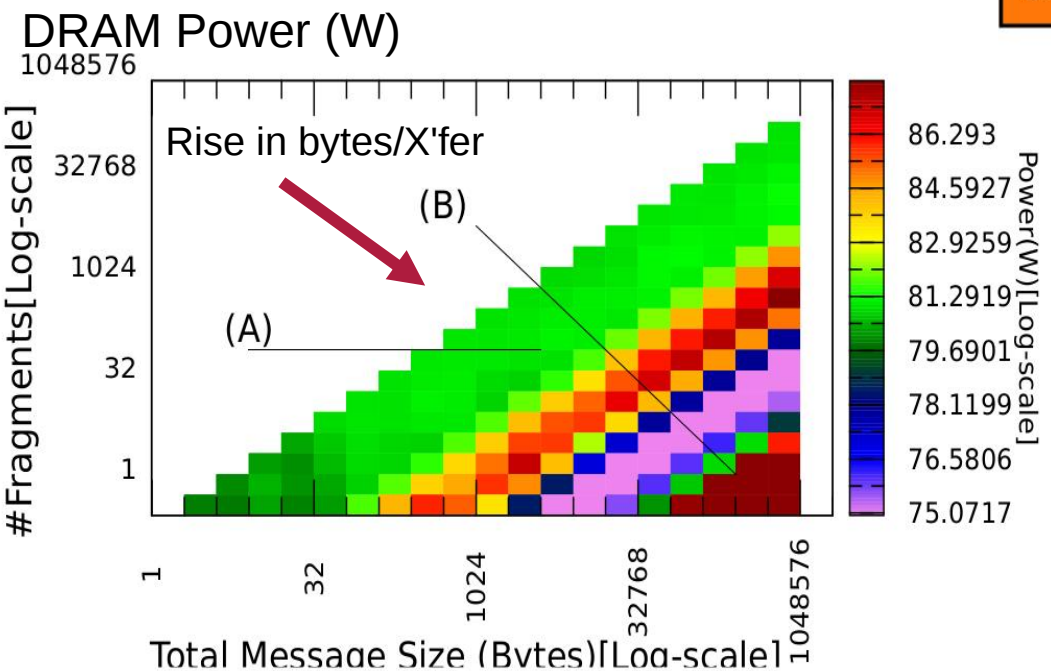
- Drop in power consumption for bulk sized transfers for the sender. The opposite for the receiver => Participation in handshaking is a dominant factor

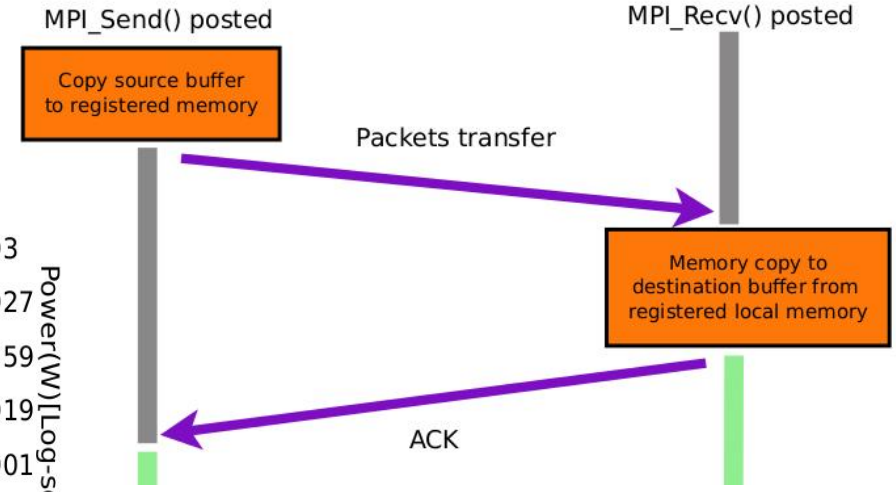
- (Sender:) Chunking of bulk transfers lead to a rise in power consumption

- DRAM power is primarily influenced by the total payload size

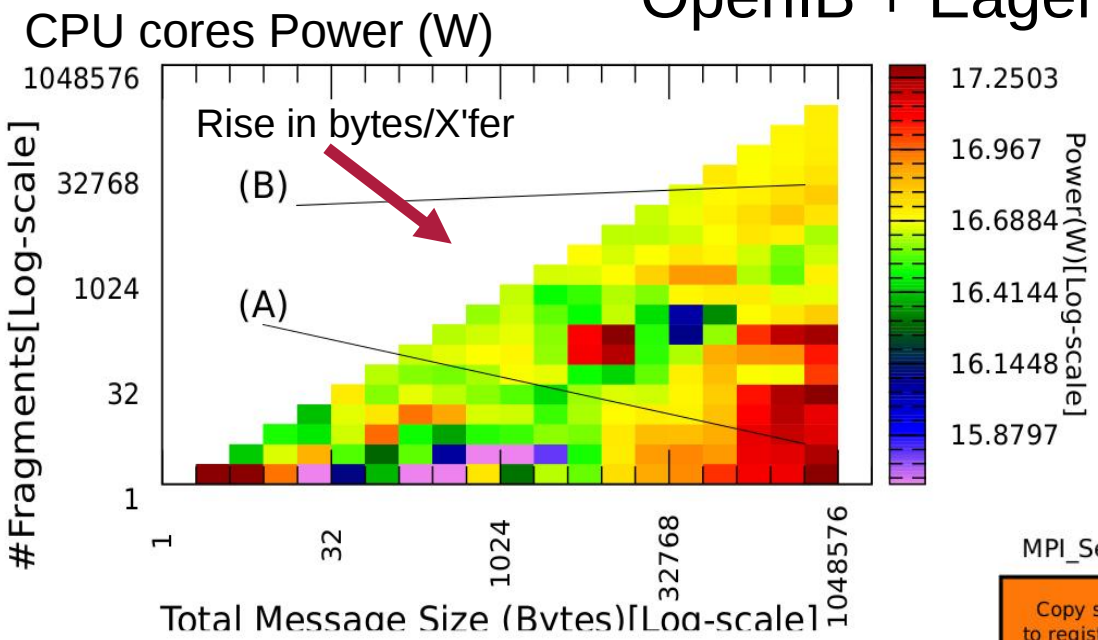
OpenIB + Eager protocol (sender)

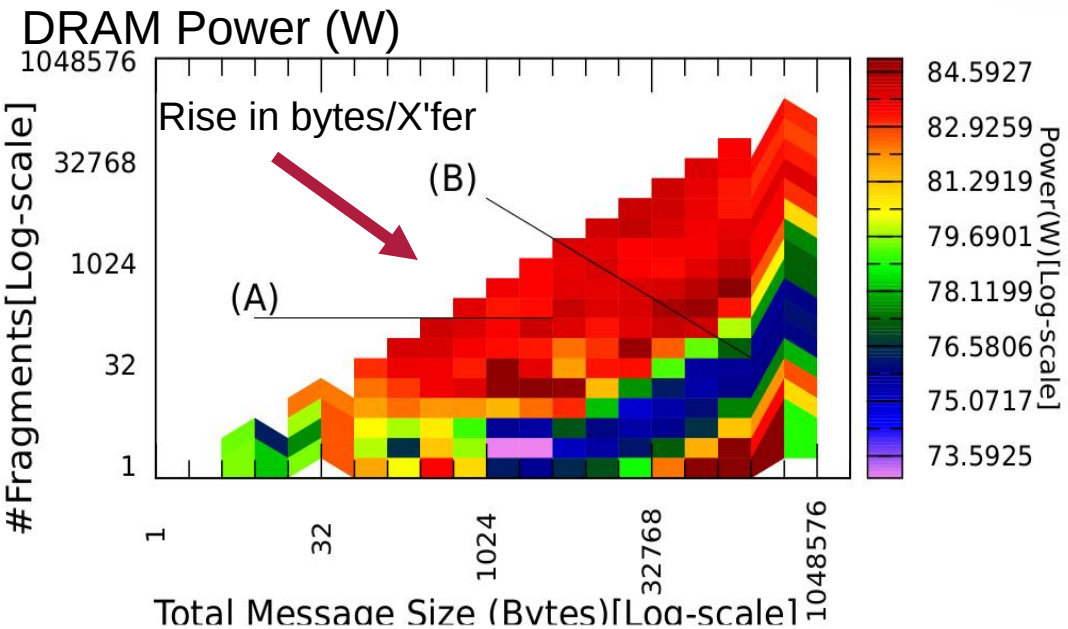


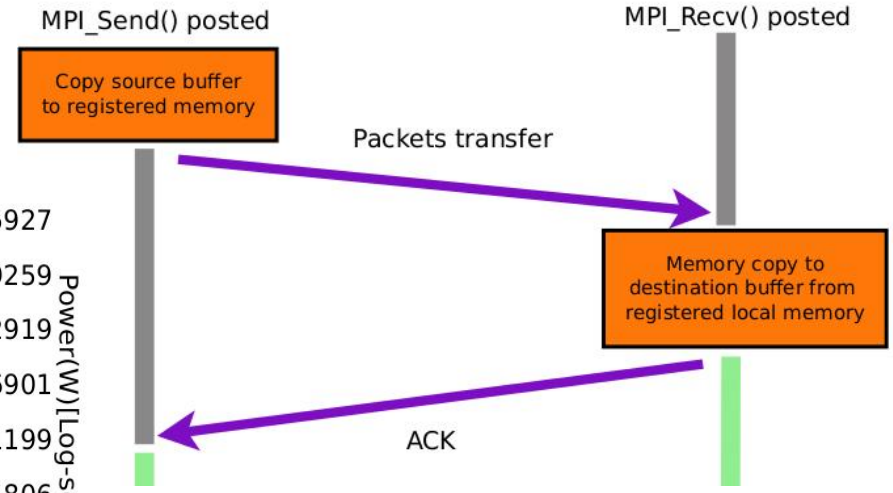




OpenIB + Eager protocol (receiver)







OpenIB+ Eager protocol

Lessons learned

- The bytes transferred per chunk plays a dominant role.

- A sweet spot where the memory bandwidth is utilized efficiently. Opportunities for fine-tuning libraries. This can be attributed to the switch between using pre-registered buffers and dynamic registration of memory.

- Overall, the range of power consumed by the system while using OpenIB is lower than using TCP.

Bytes transferred per joule

Bandwidth Achieved

Total Bytes transferred

$$\frac{Bw}{P_{net}} = \frac{Bw}{(P_{s,cpu}+P_{s,mem}+P_{r,cpu}+P_{r,mem})} = \frac{B_{payload}}{\Delta E_s+\Delta E_r} \quad (\frac{Bytes}{Joule})$$

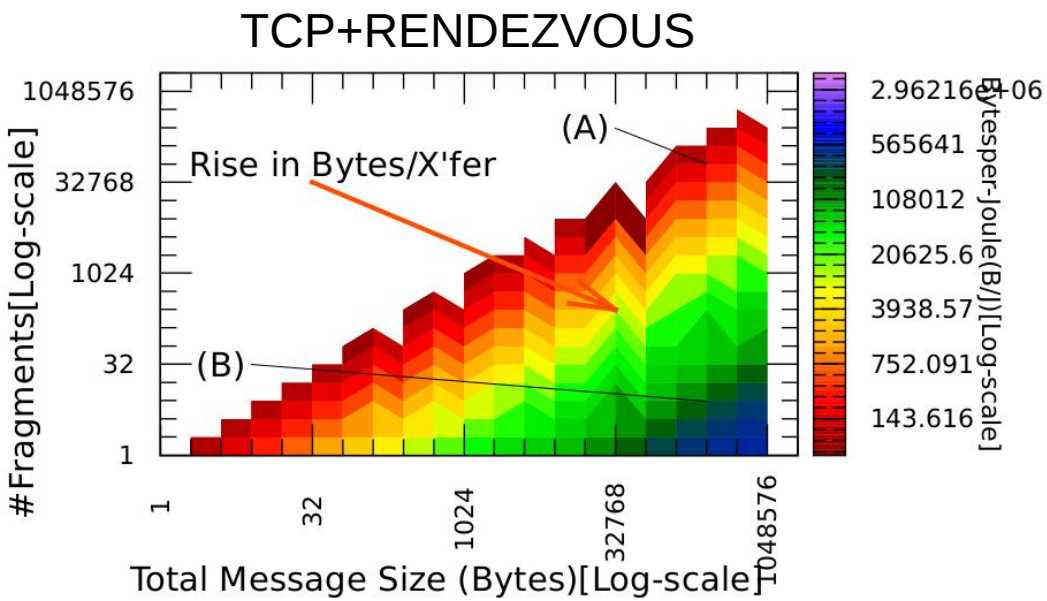
Power @ sender CPU+DRAM

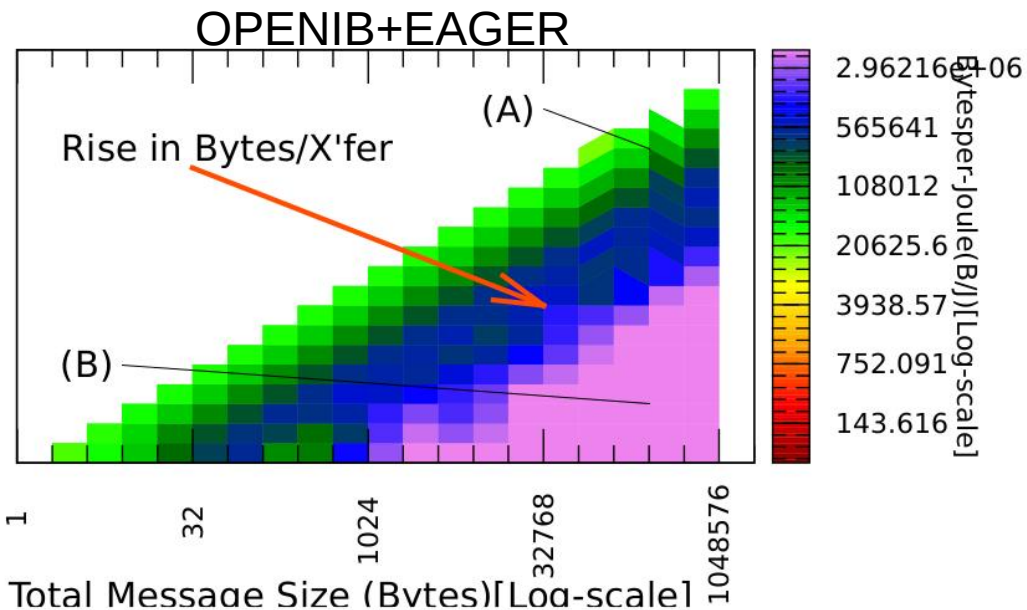
Power @ receiver CPU+DRAM

Total Energy Consumed

Overall impact (TCP+Rendezvous v/s OpenIB+Eager)

Metric: Bytes Transferred per Joule (Two node system)

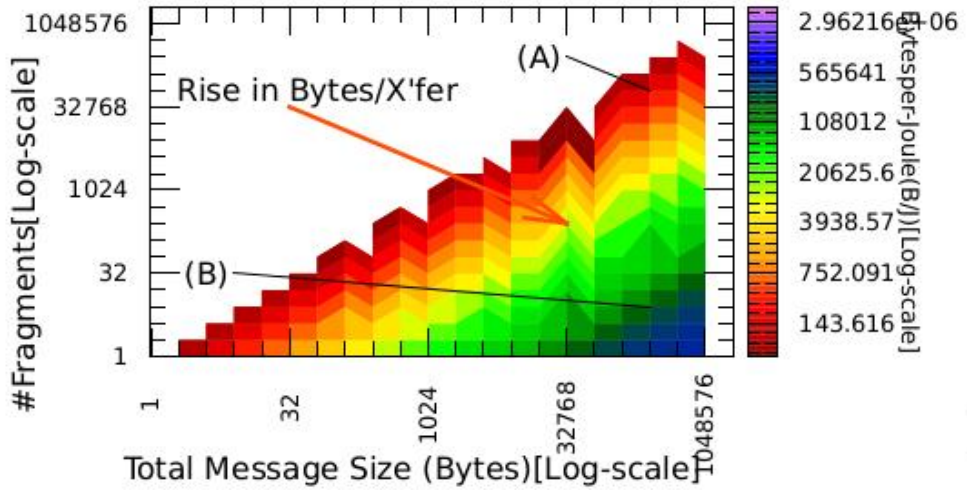




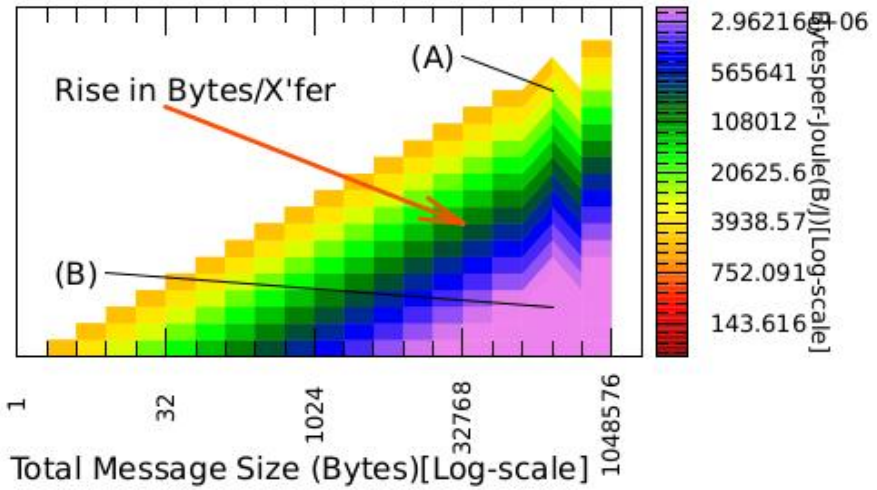
Overall impact (all configurations)

Metric: Bytes Transferred per Joule (Two node system)

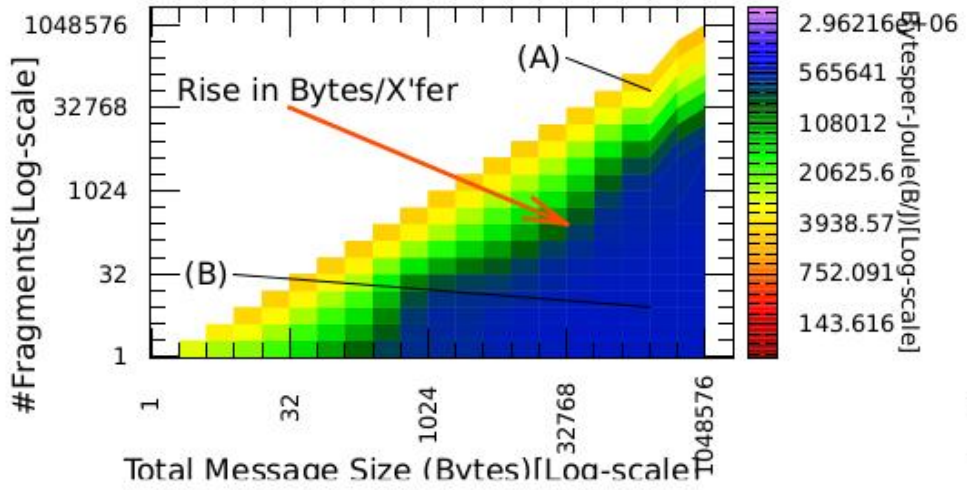
TCP+RENDEZVOUS: Bytes per Joule (B/J)



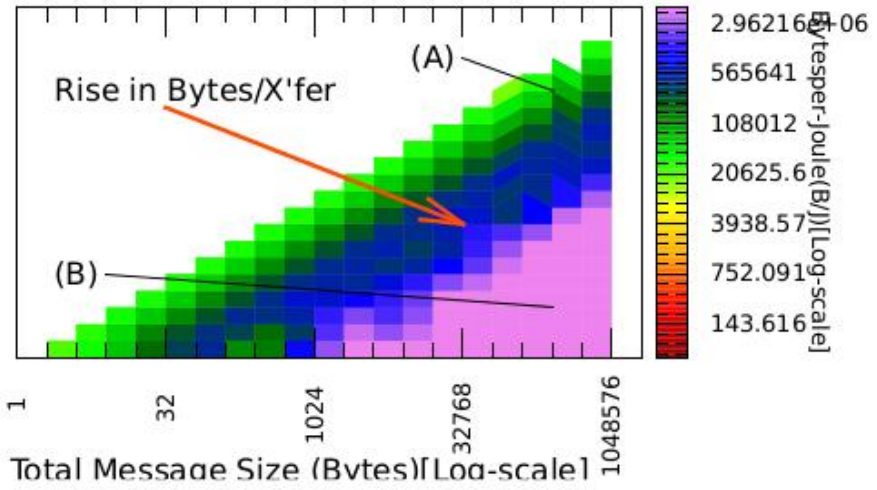
OPENIB+RENDEZVOUS: Bytes per Joule (B/J)



TCP+EAGER: Bytes per Joule (B/J)



OPENIB+EAGER: Bytes per Joule (B/J)



Overall impact (all configurations)

Lessons learned

- Main factors dominate energy efficiency of transfers
  - Total achievable bandwidth by the underlying interconnect
  - The overhead of the data transfer protocols

Conclusions

(Tips for achieving energy-efficient transfers)

- Data movement is costly
  - Impact of design of communication kernels
    - Total data transfer size
    - Total number of fragments / chunks
  - Underlying design factors of middleware
    - Data transfer protocols
    - Choice of transport layer
- Empirical results indicate:
  - Aggregating smaller chunks of buffers into larger contiguous memory buffers
  - Small-sized transfers should be preferred
    - Algorithm dependent
    - If used, the overhead of the data transfer protocol need to be accounted for.

Acknowledgments

•Funding supported by the U.S. DOD

•Resources supported by ORNL, TN

•Facility supported by U.S. DOE through
  •Contract No. DE-AC05-00OR22725.

•Project supported by HPCTools @ University of Houston

•Thanks are due to support teams of:
  •PAPI @ University of Tennessee, Knoxville
  •VampirTrace @ Technische Universität Dresden
  •PowerPack @ Virginia Tech

Thank You !

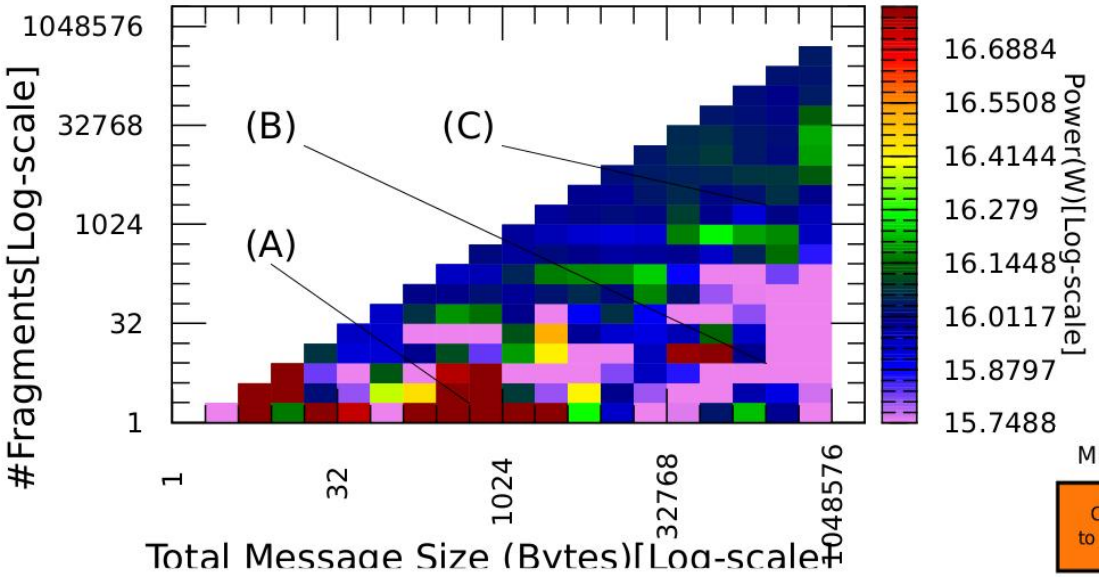
Speaker:
Siddhartha Jana (sidjana@cs.uh.edu)
University of Houston, USA

Questions ?

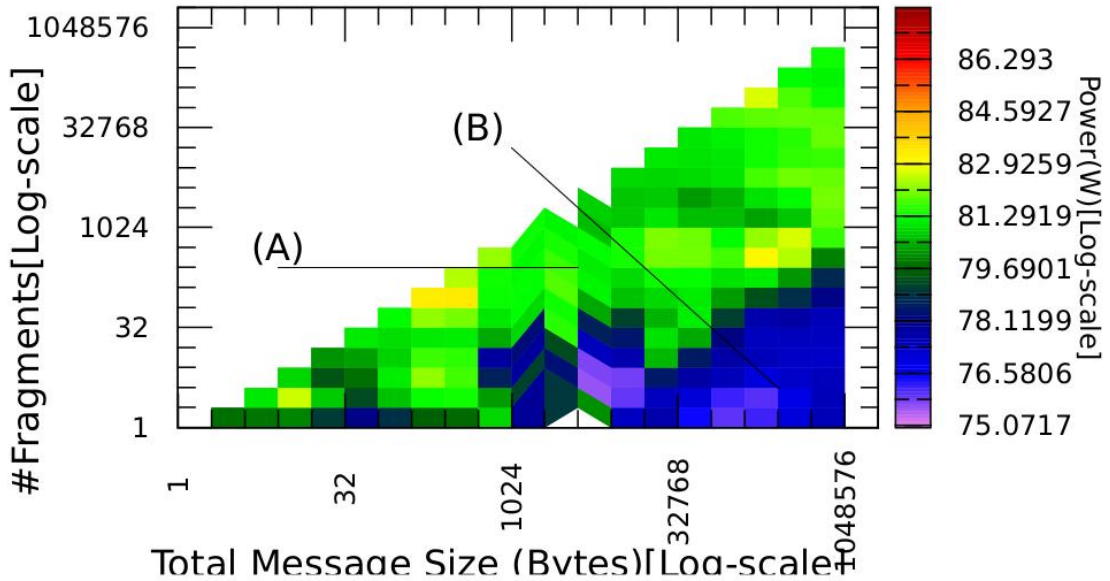
Backup slides

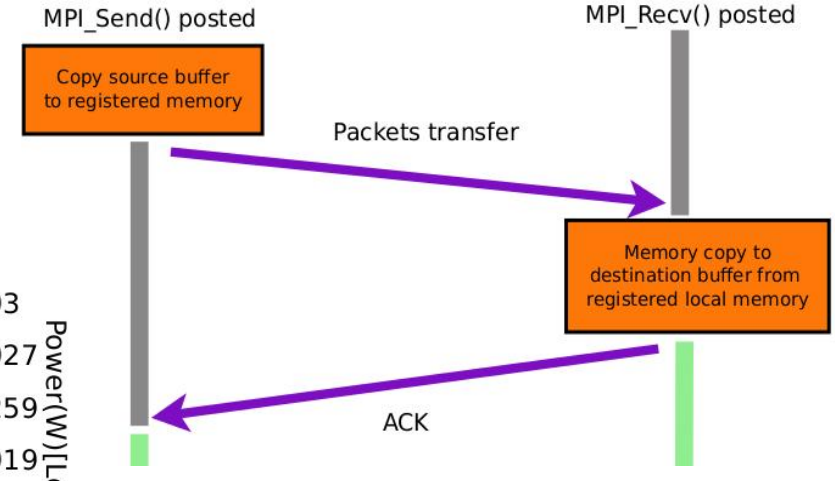
TCP + Eager protocol

CPU cores Power (W)



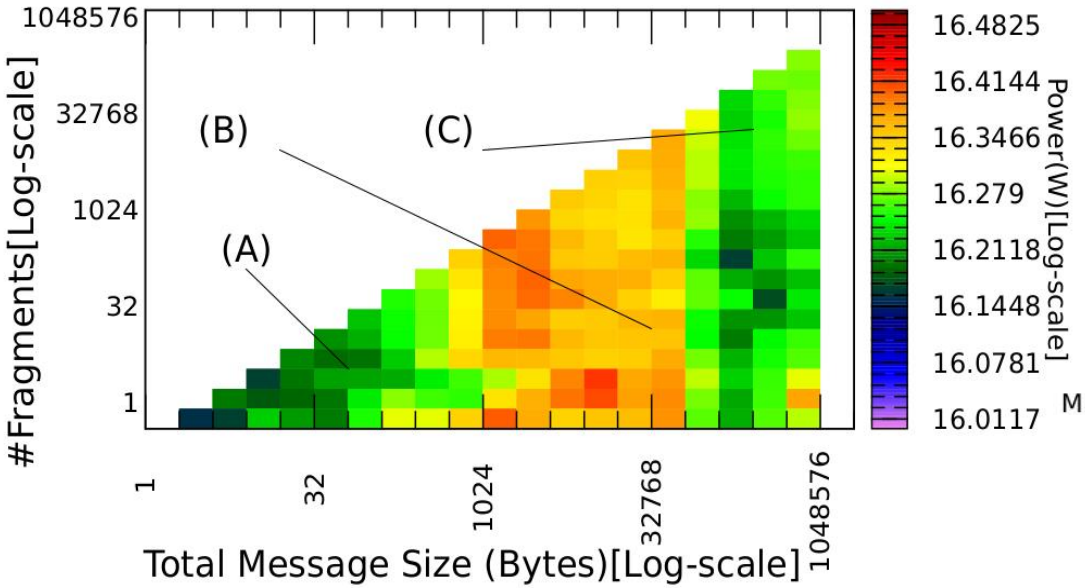
DRAM Power (W)





OpenIB + Rendezvous protocol

CPU cores Power (W)



DRAM Power (W)