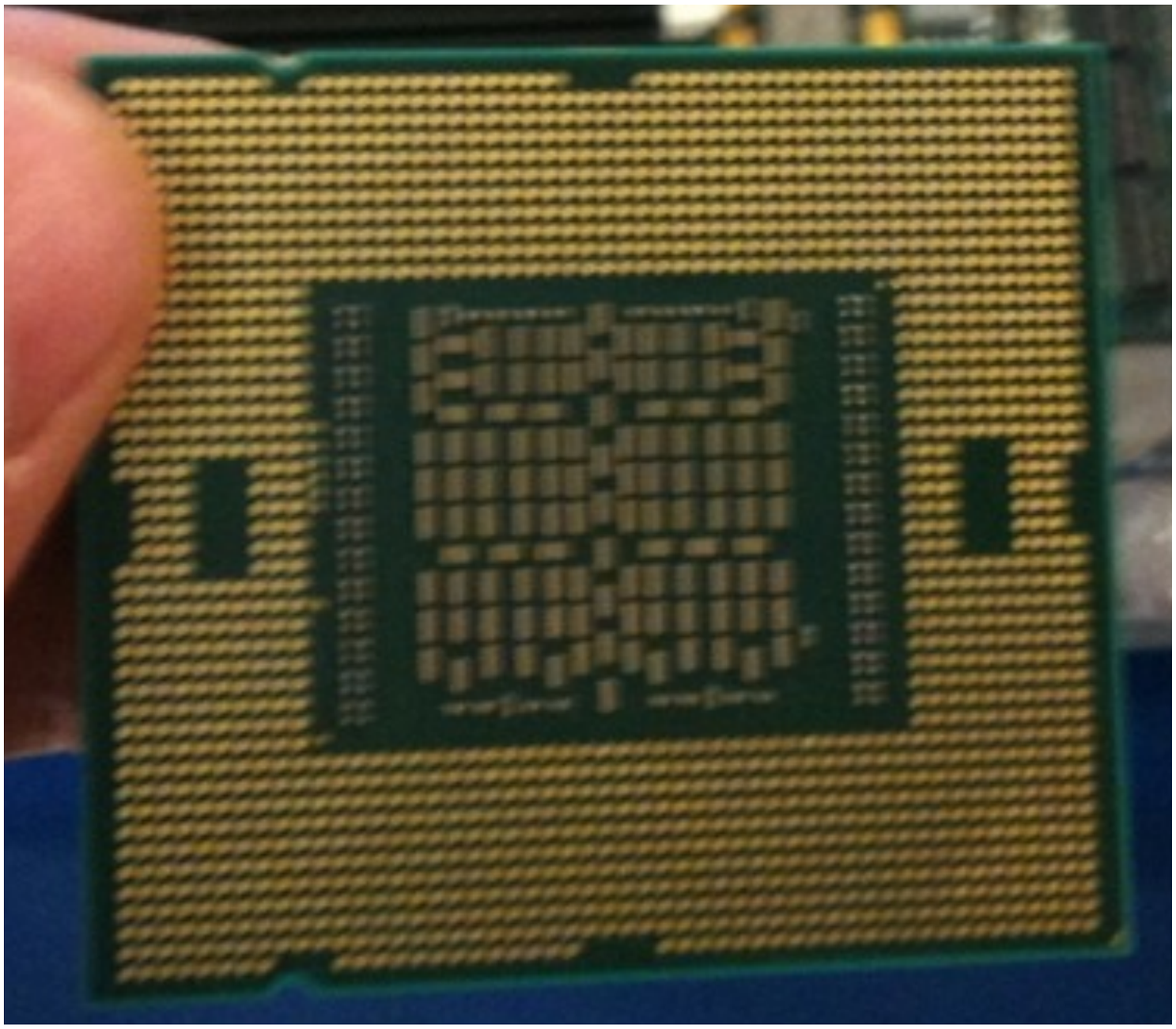
X10 on the Single-chip Cloud Computer (SCC)

Keith Chapman, Ahmed Hussein, Antony Hosking

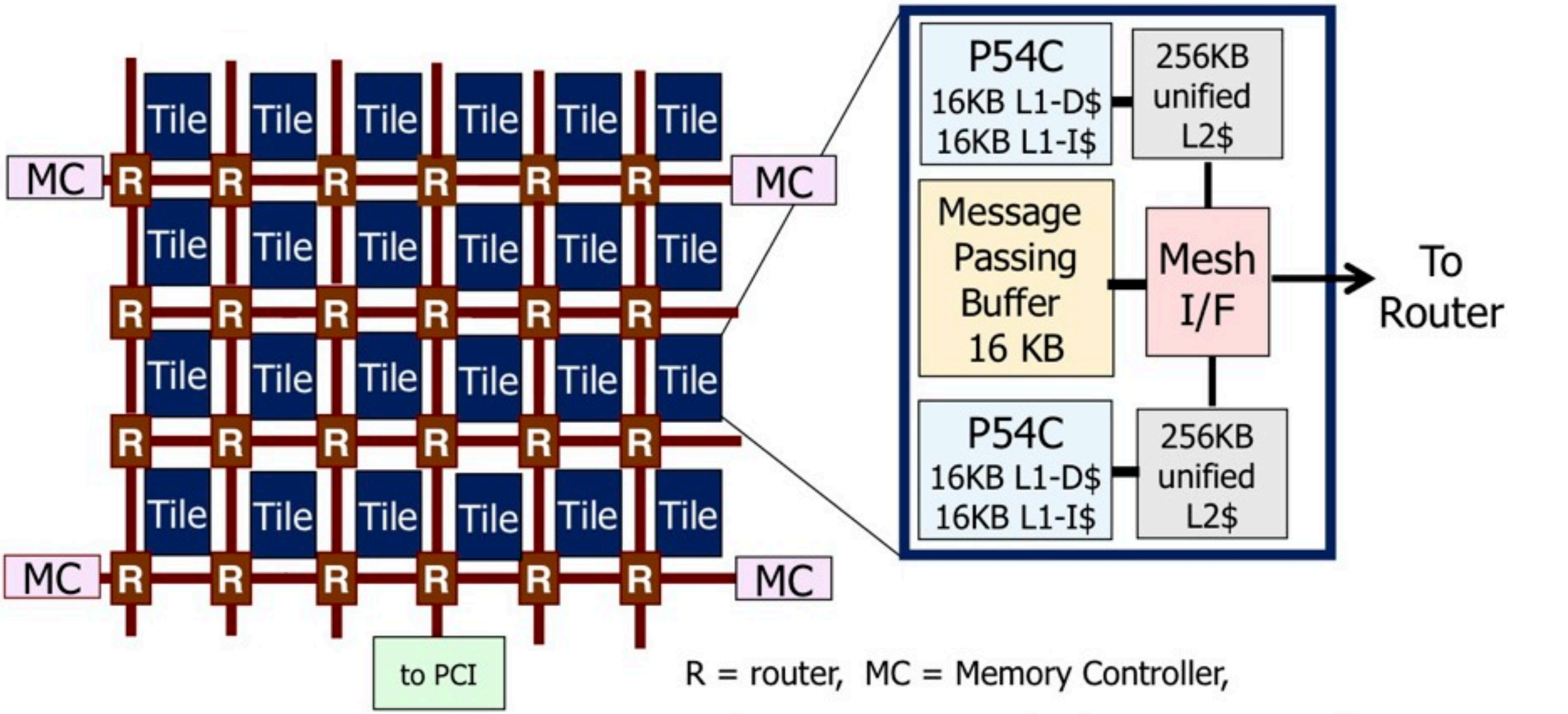
PURDUE
UNIVERSITY™

What is the SCC



PURDUE
UNIVERSITY

What is the SCC

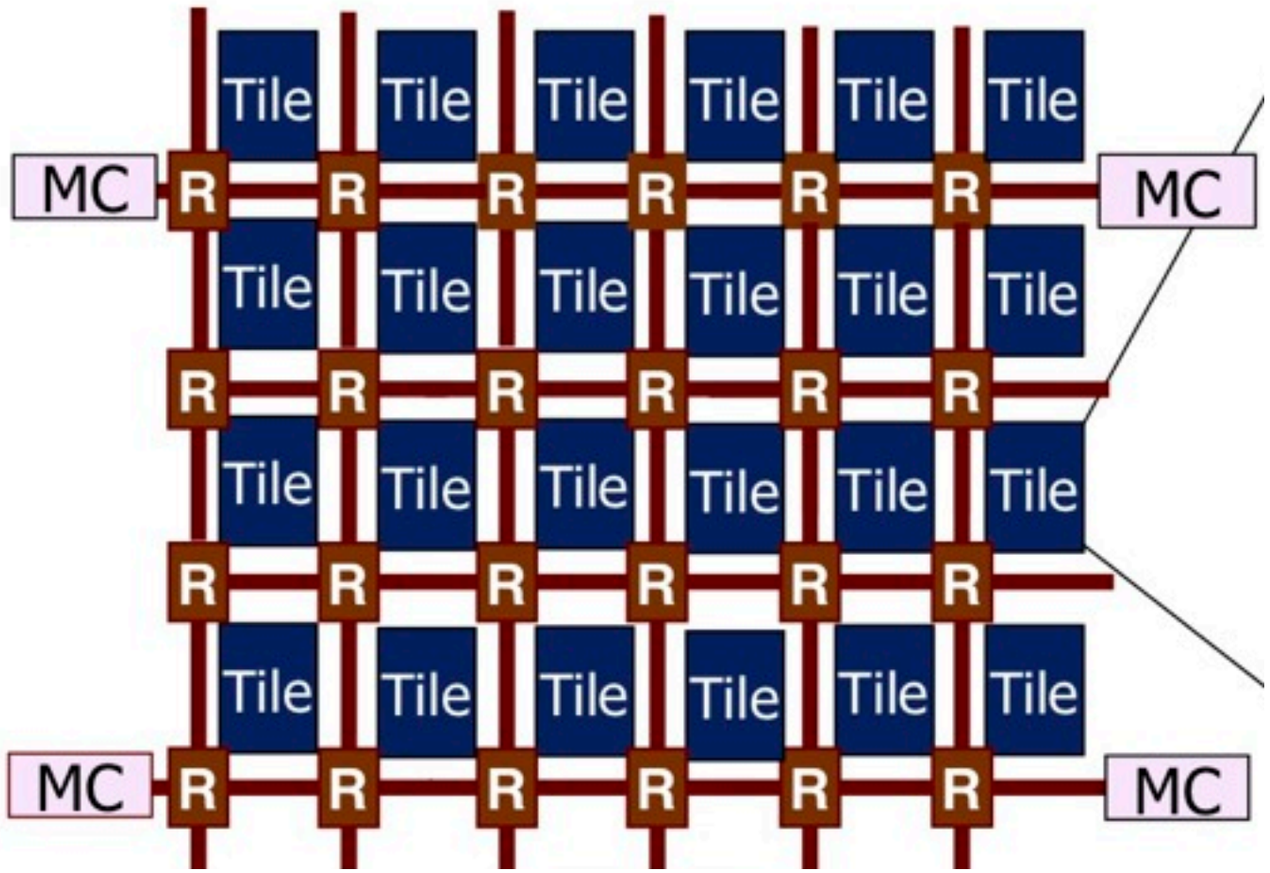


R = router, MC = Memory Controller,

P54C = second generation Pentium® core, CC = cache cntrl.

PURDUE UNIVERSITY

What is the SCC

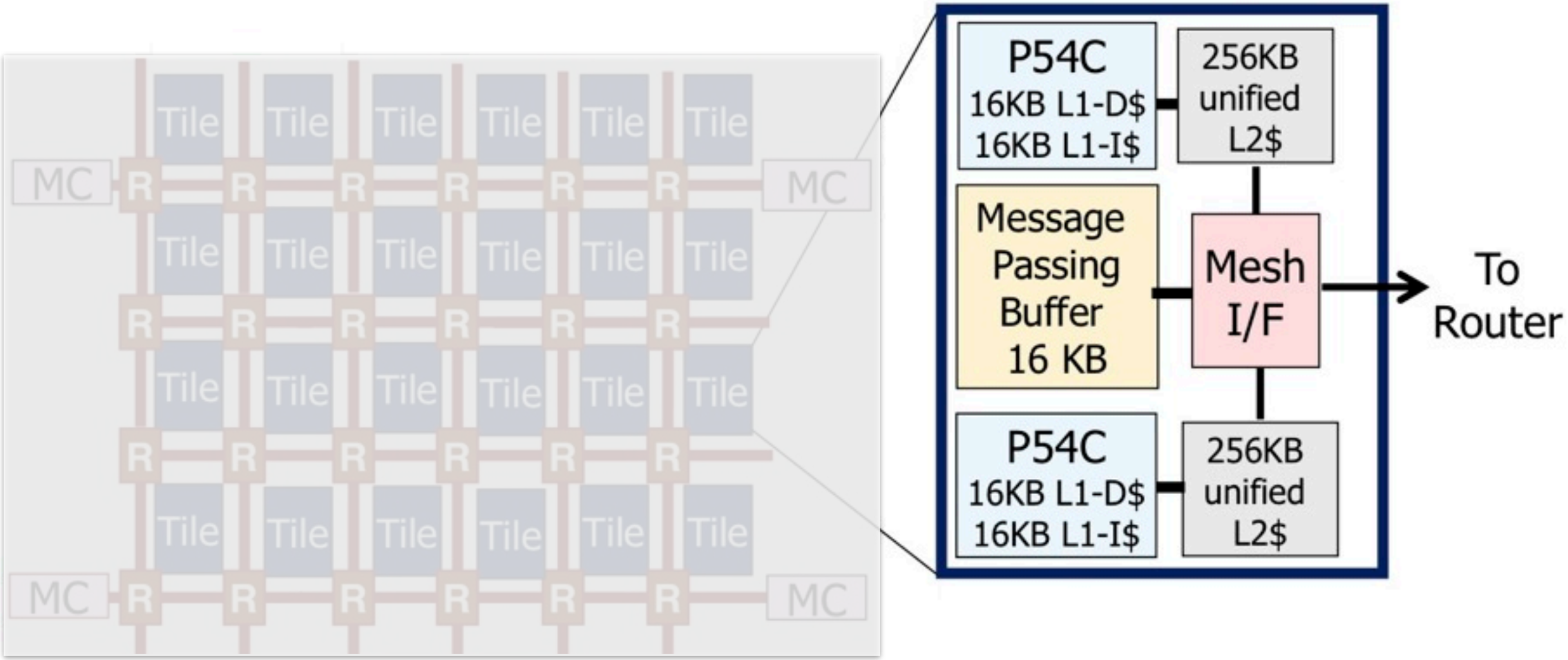


PURDUE UNIVERSITY

What is the SCC

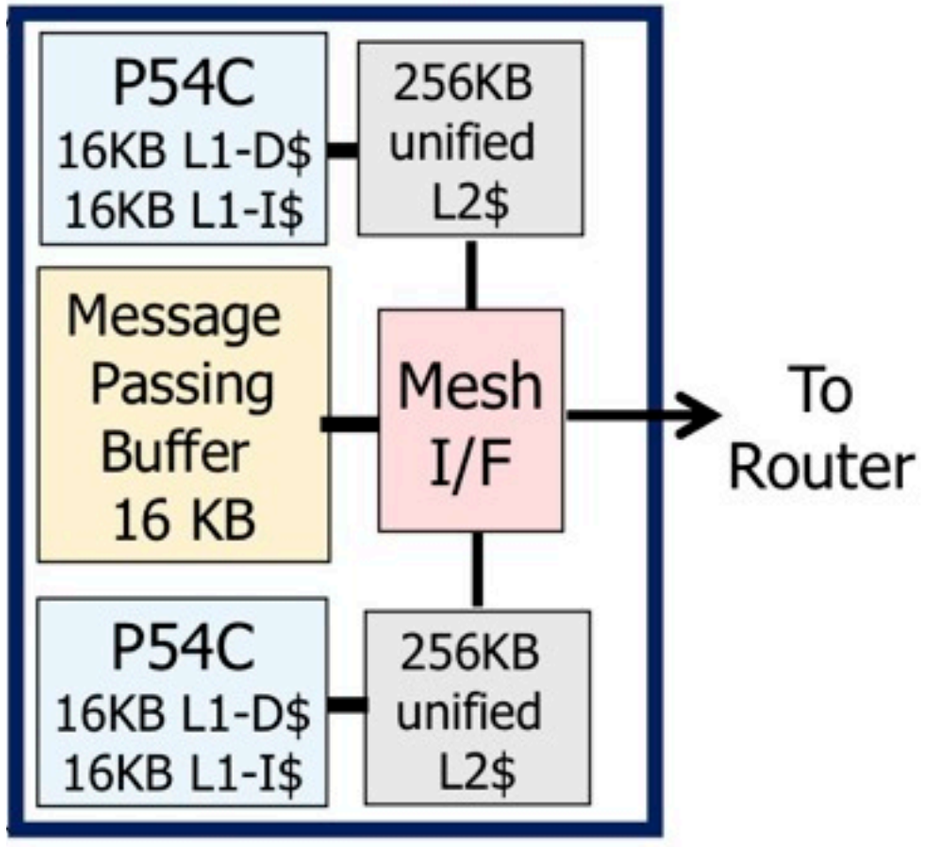
PURDUE UNIVERSITY

What is the SCC



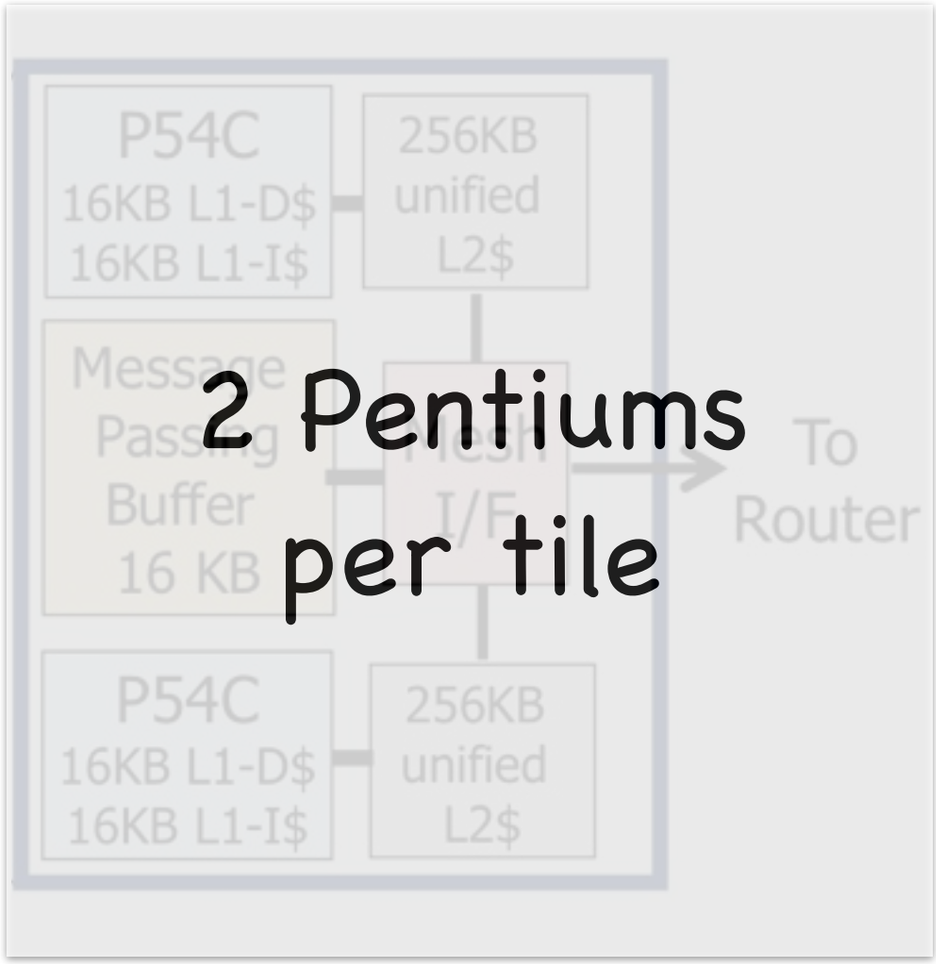
PURDUE
UNIVERSITY

What is the SCC



PURDUE
UNIVERSITY

What is the SCC

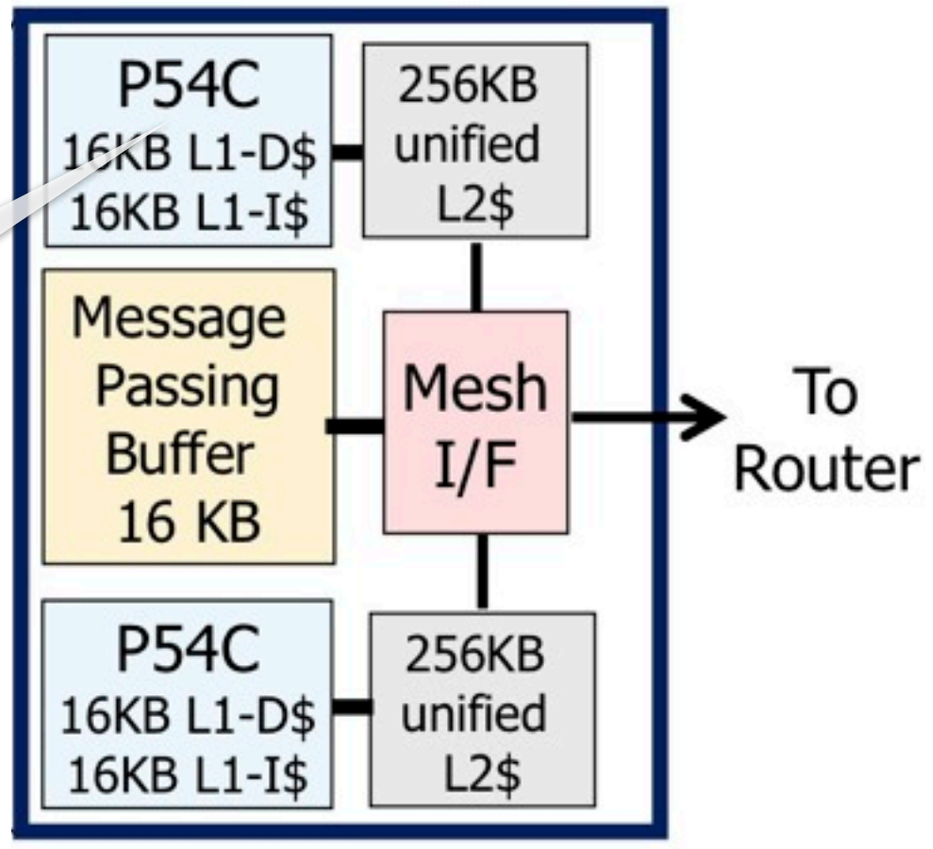


2 Pentiums per tile

PURDUE UNIVERSITY

What is the SCC

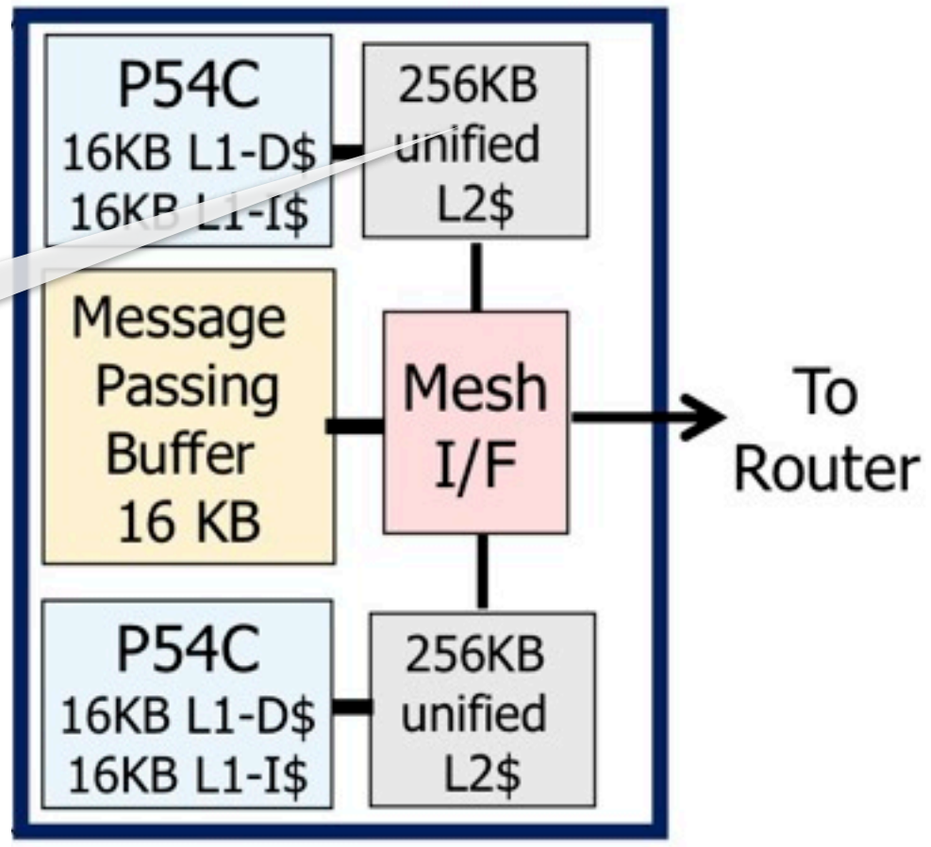
16 KB L1 D$ + I$



PURDUE
UNIVERSITY

What is the SCC

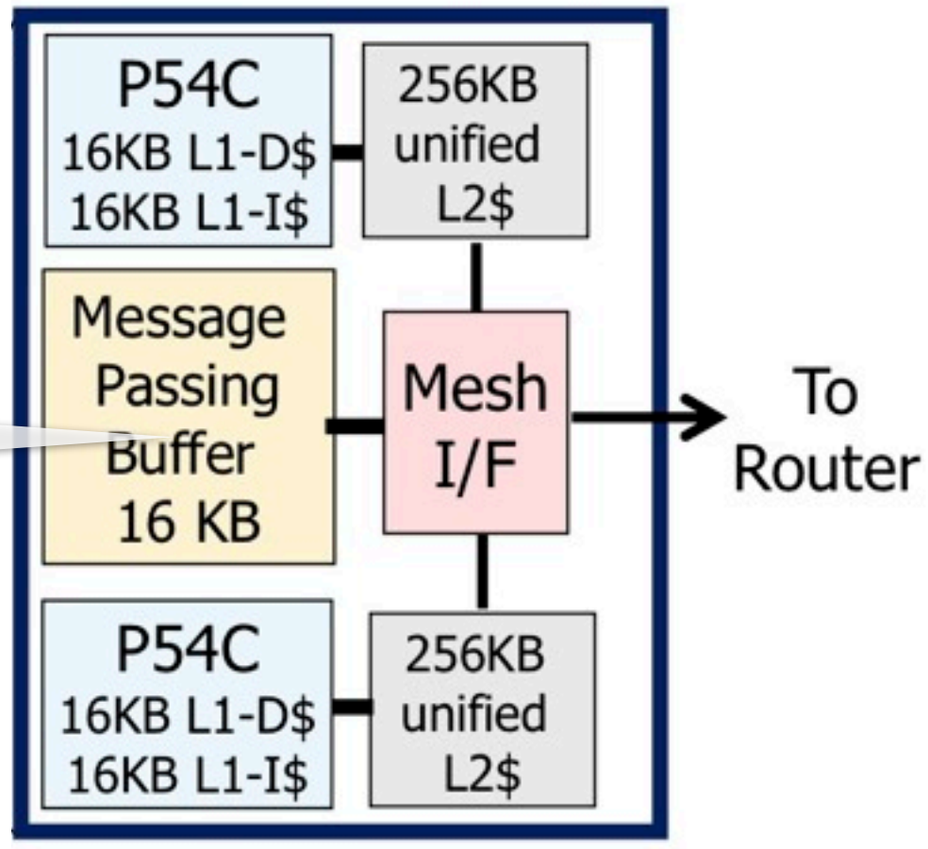
256 KB L2$



PURDUE
UNIVERSITY

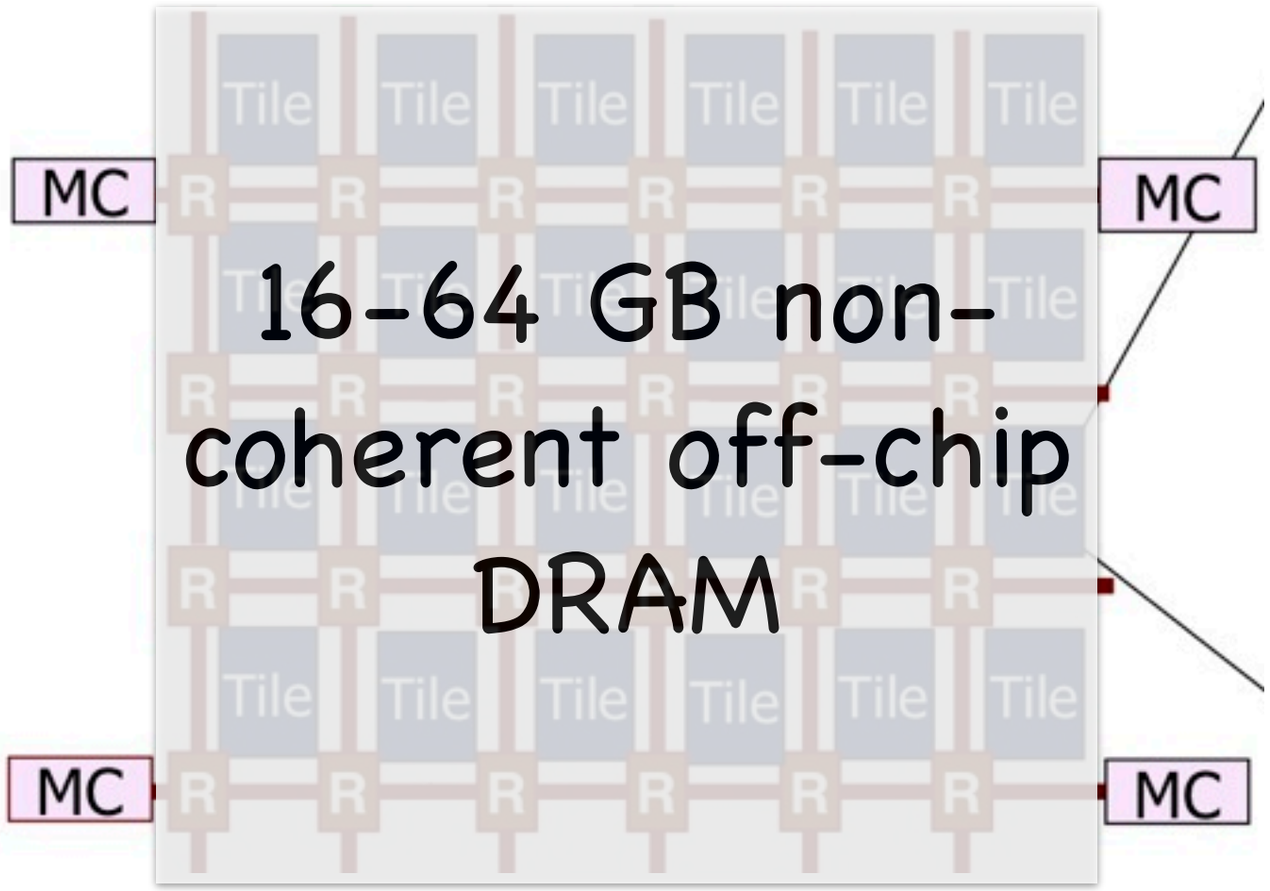
What is the SCC

16 KB message passing buffer



PURDUE UNIVERSITY

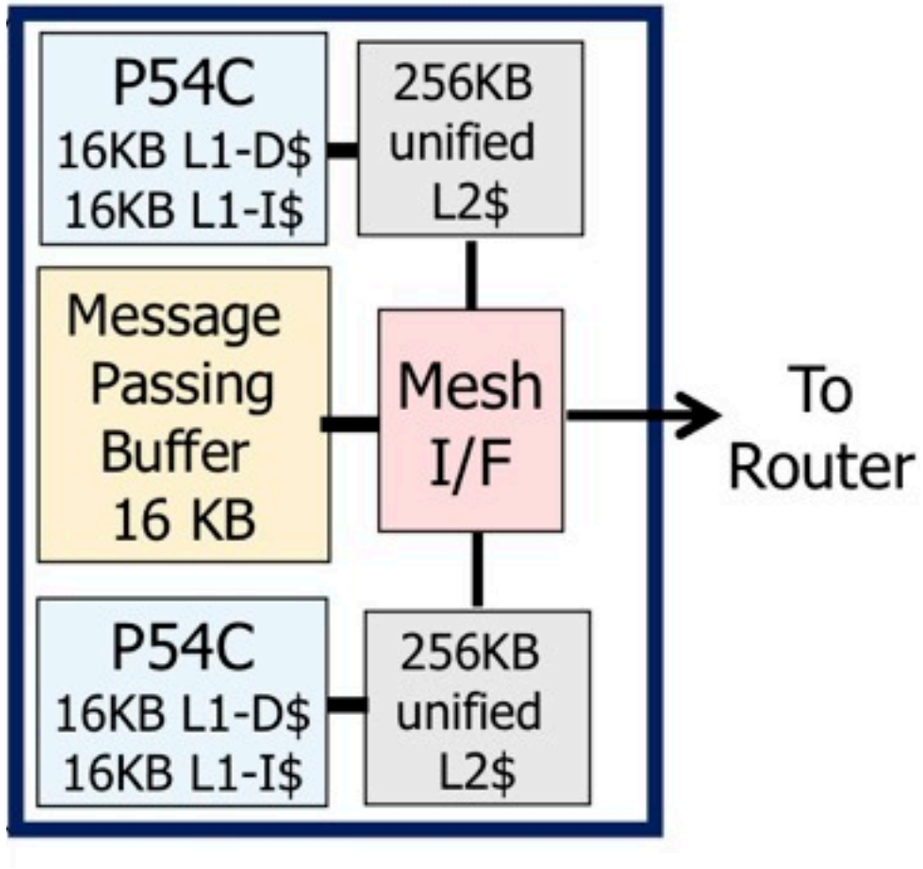
What is the SCC



PURDUE UNIVERSITY

Message Passing Buffer

- Shared on-chip SRAM
- Mapped by all cores
- Cached in L1
- Tagged for fast invalidate
- 8 KB per core
- 384 KB total accessible



PURDUE UNIVERSITY

RCCE native messaging

- Fast, thin layer over MPB
- One-sided *put/get* API
- RCCE flags allocatable in each MPB for coordination (one T&S register per core)
- Two-sided *send/recv* built on one-sided API using RCCE Flags

PURDUE UNIVERSITY

X10 (C++) on the SCC

- Relatively straightforward
  - cross-compile using SCC tool chain
  - MPI or sockets over MPB out of the box
- What about RCCE?

PURDUE UNIVERSITY

X10RT API

- Used by X10 runtime to communicate between *places*
- Send serialized object graphs
- Registered receiver callbacks for different types of message
- Probe occasionally for incoming messages
- Don't block waiting for messages to arrive

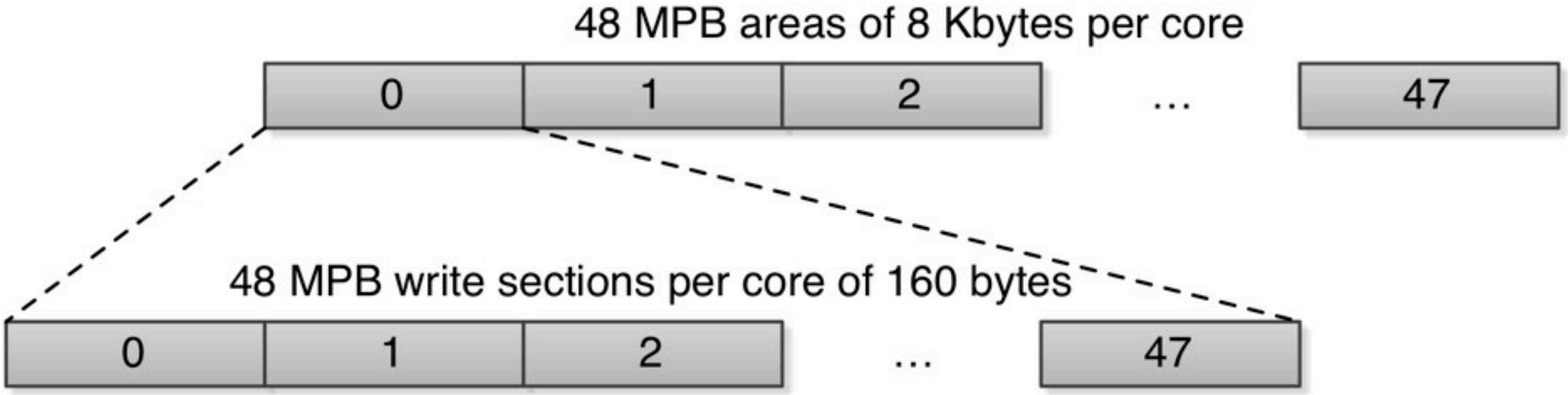
PURDUE UNIVERSITY

X10RT + MPI + SCC

- X10RT with MPI binding
- SCC MPI library with alternative channels
  - mpi-mpb: direct MPB
  - mpi-sock: sockets via mesh TCP/IP driver
  - mpi-shm: shared memory

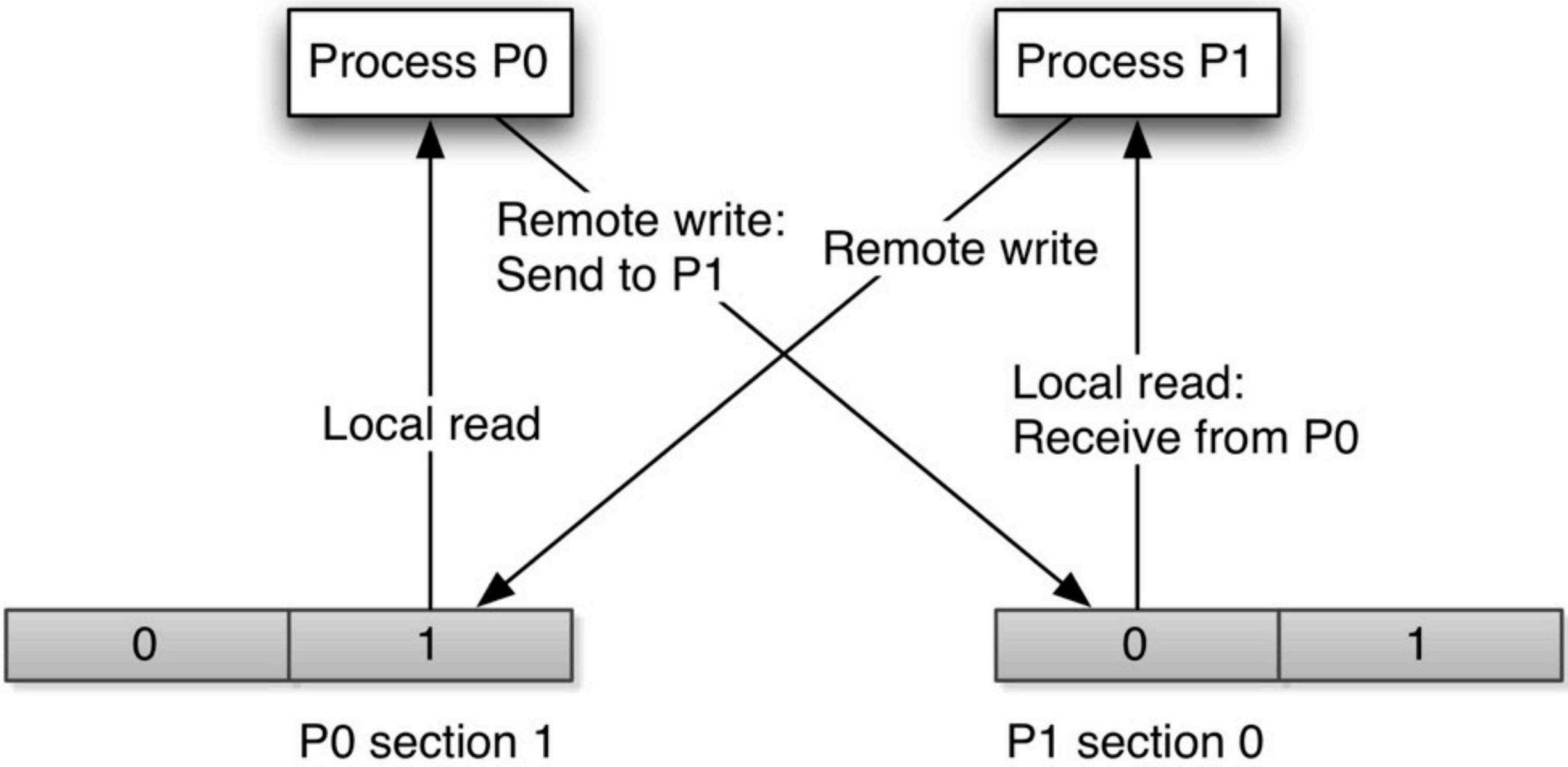
PURDUE UNIVERSITY

mpi-mpb MPB management

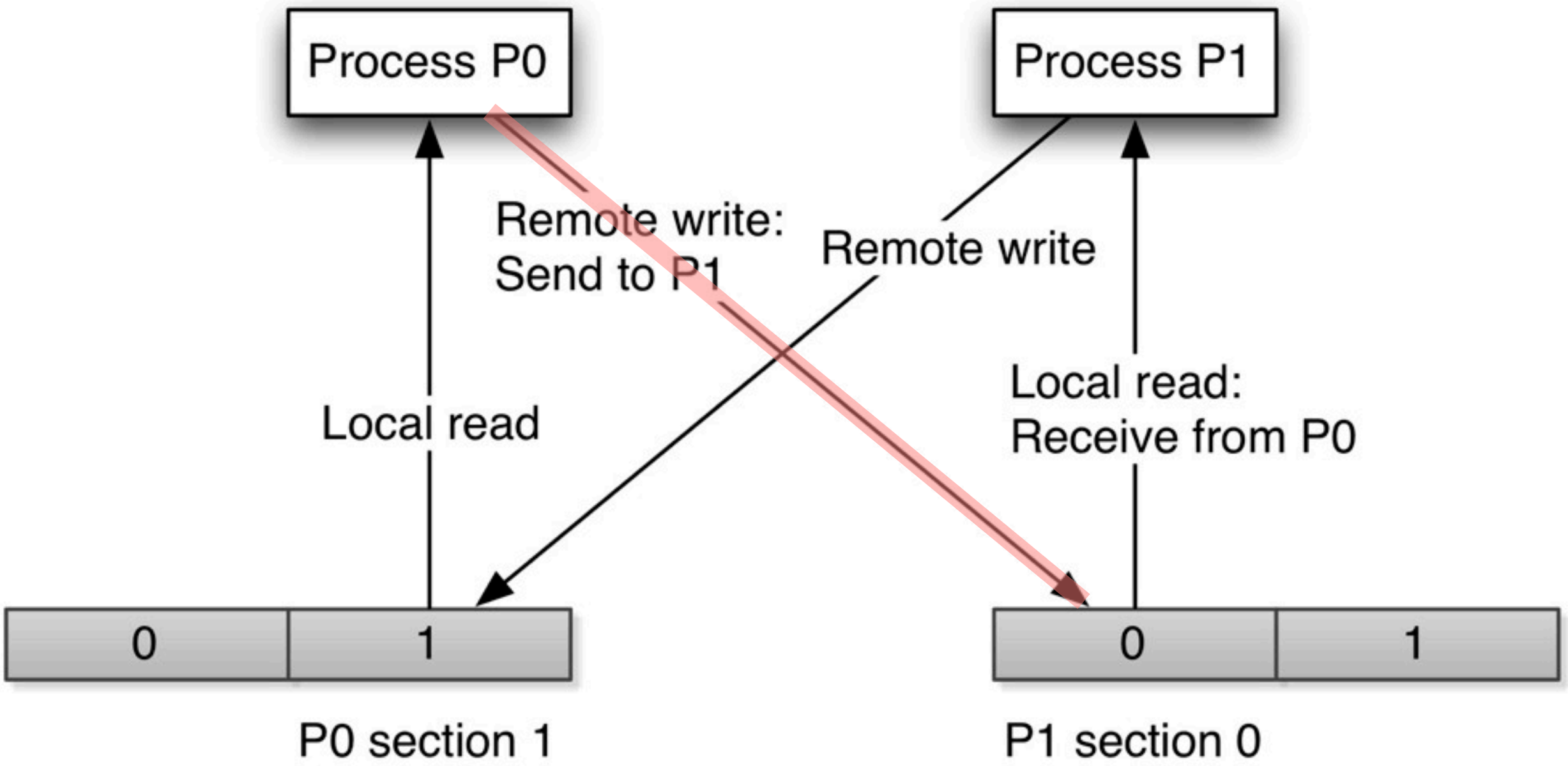


PURDUE UNIVERSITY

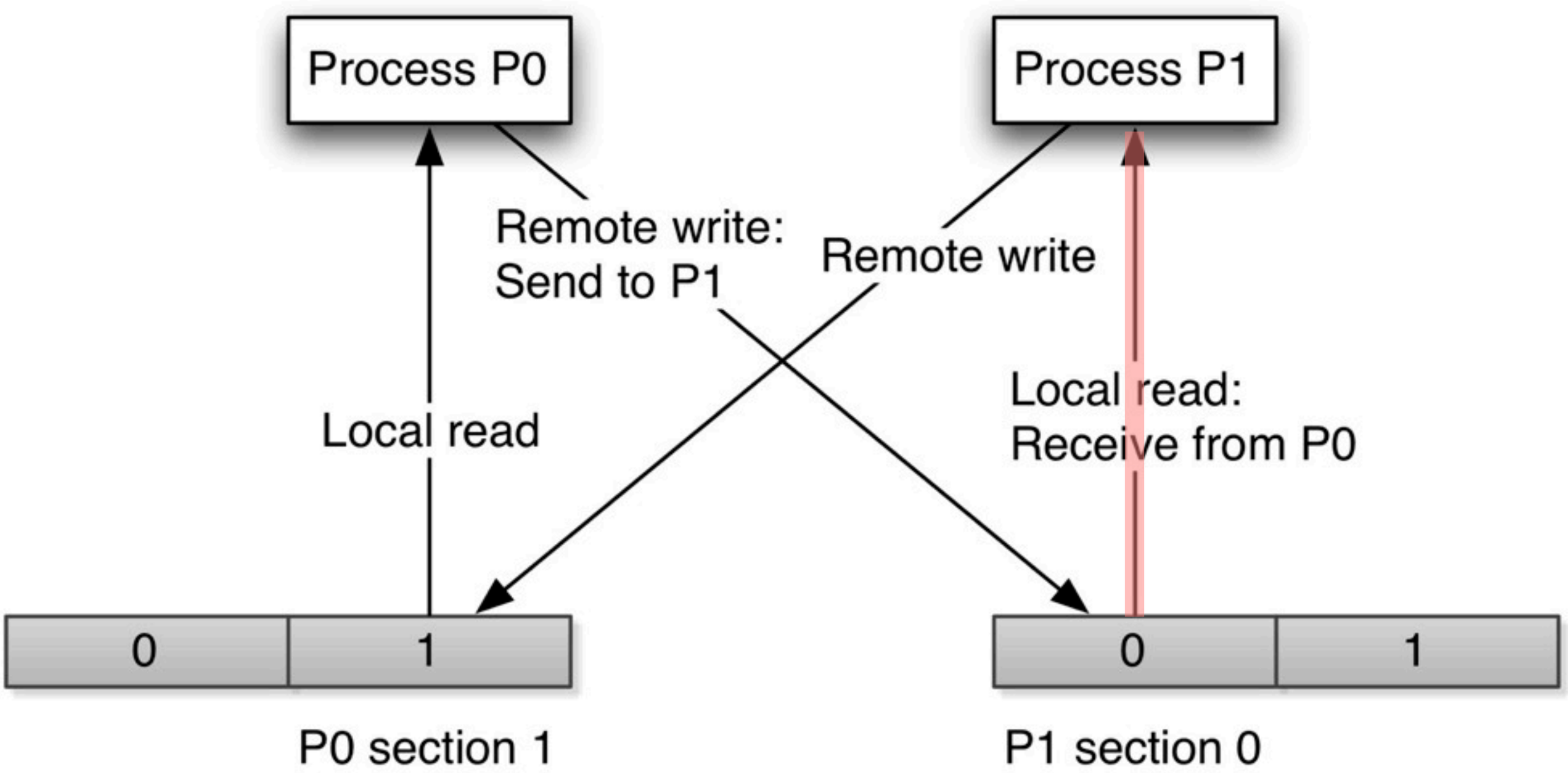
mpi-mpb send/receive



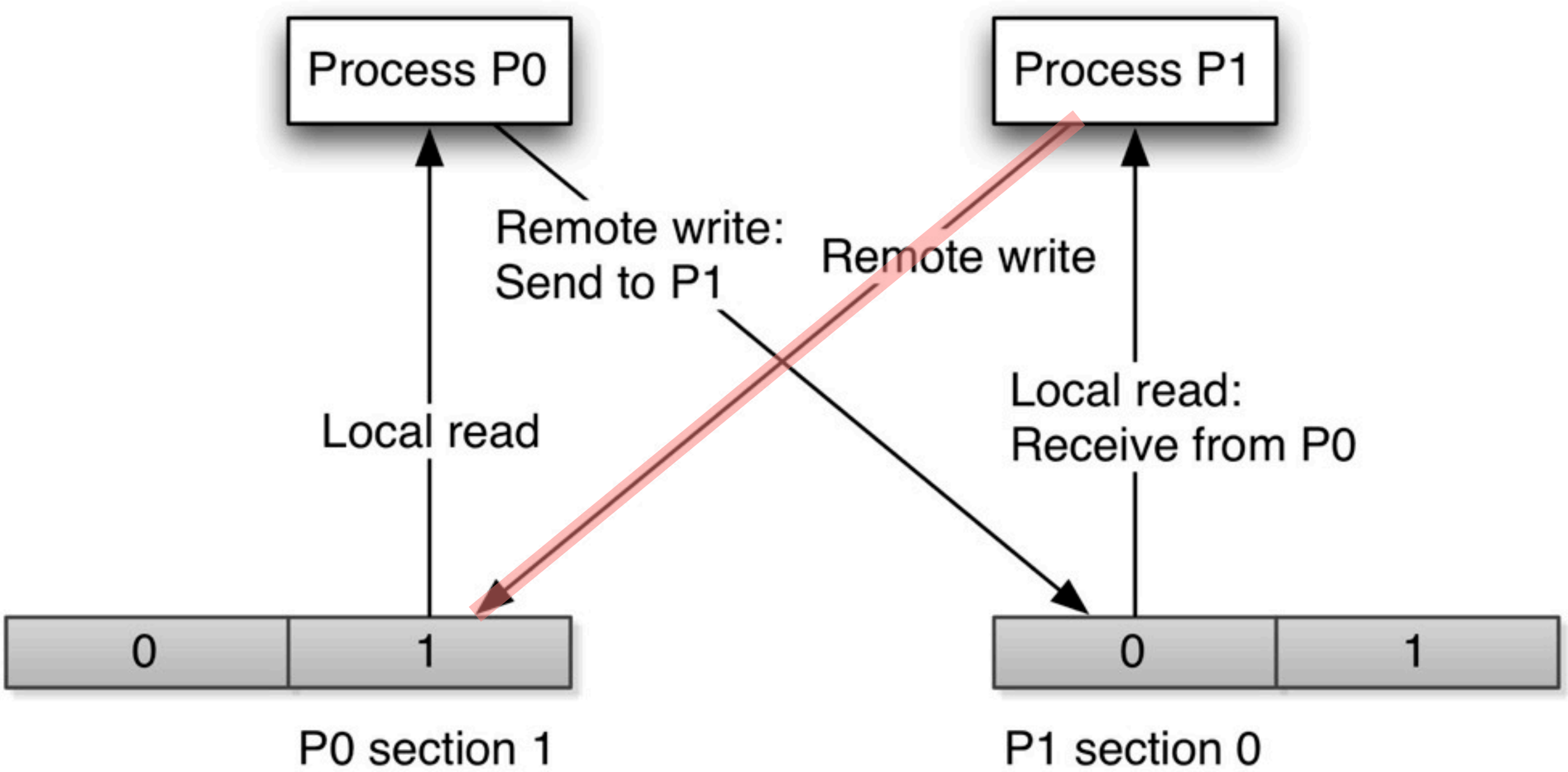
mpi-mpb send/receive



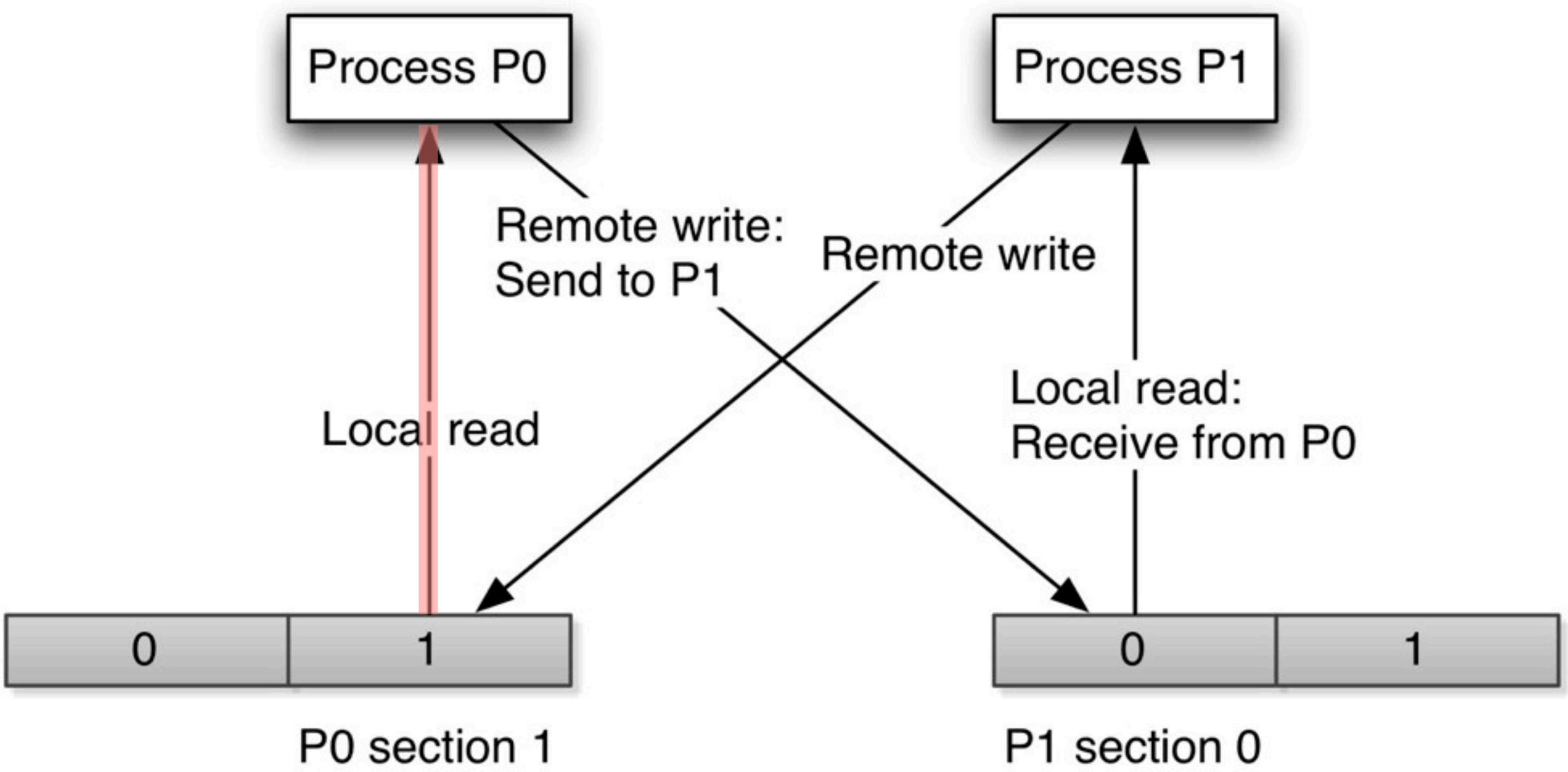
mpi-mpb send/receive



mpi-mpb send/receive



mpi-mpb send/receive



X10RT+"RCCE"+SCC

- RCCE send/recv are blocking
- RCCE non-blocking recv_test requires knowing sender to test
- RCCE does not support X10RT tags

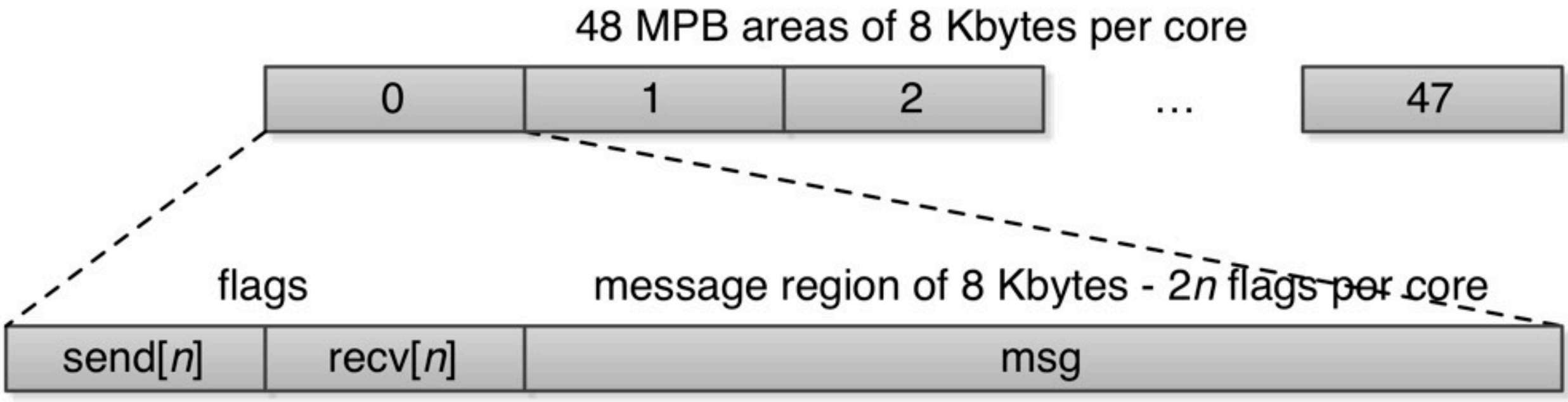
PURDUE UNIVERSITY

X10RT+"RCCE"+SCC

- Re-implement as RCCE-X10
  - one-sided non-blocking send
  - non-blocking probe

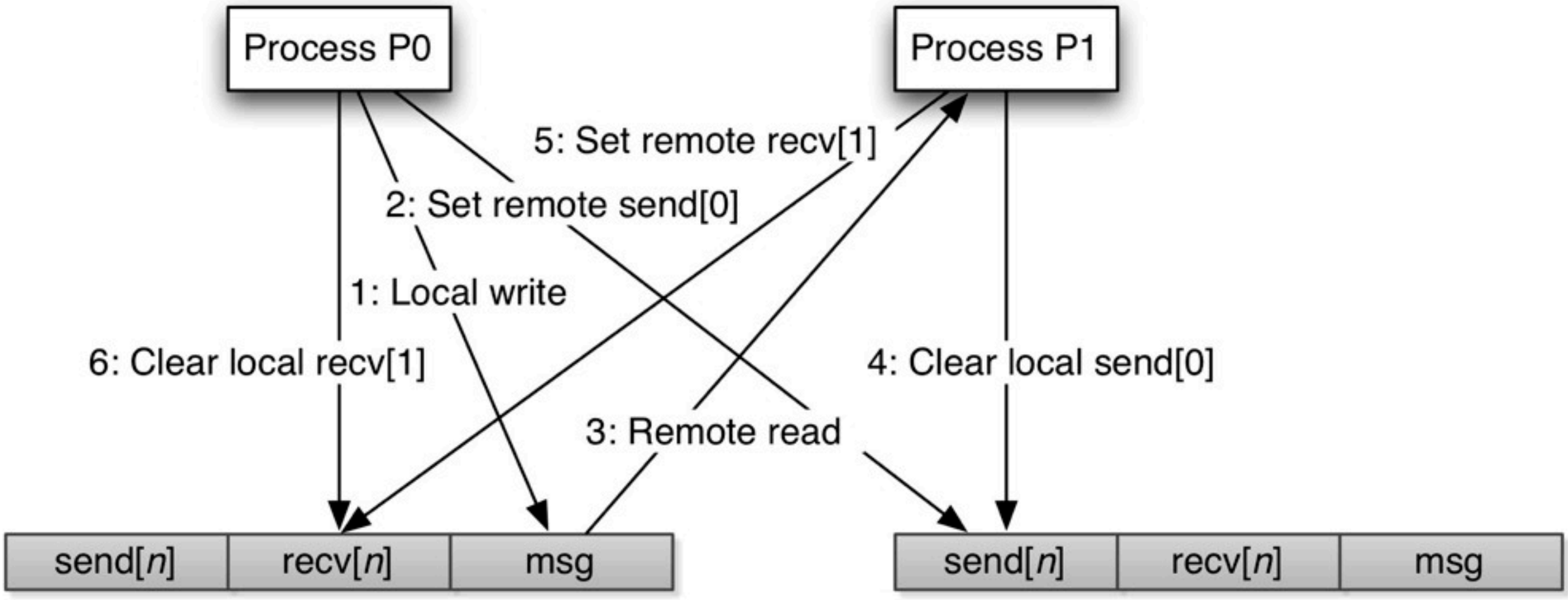
PURDUE UNIVERSITY

RCCE-X10 MPB Management



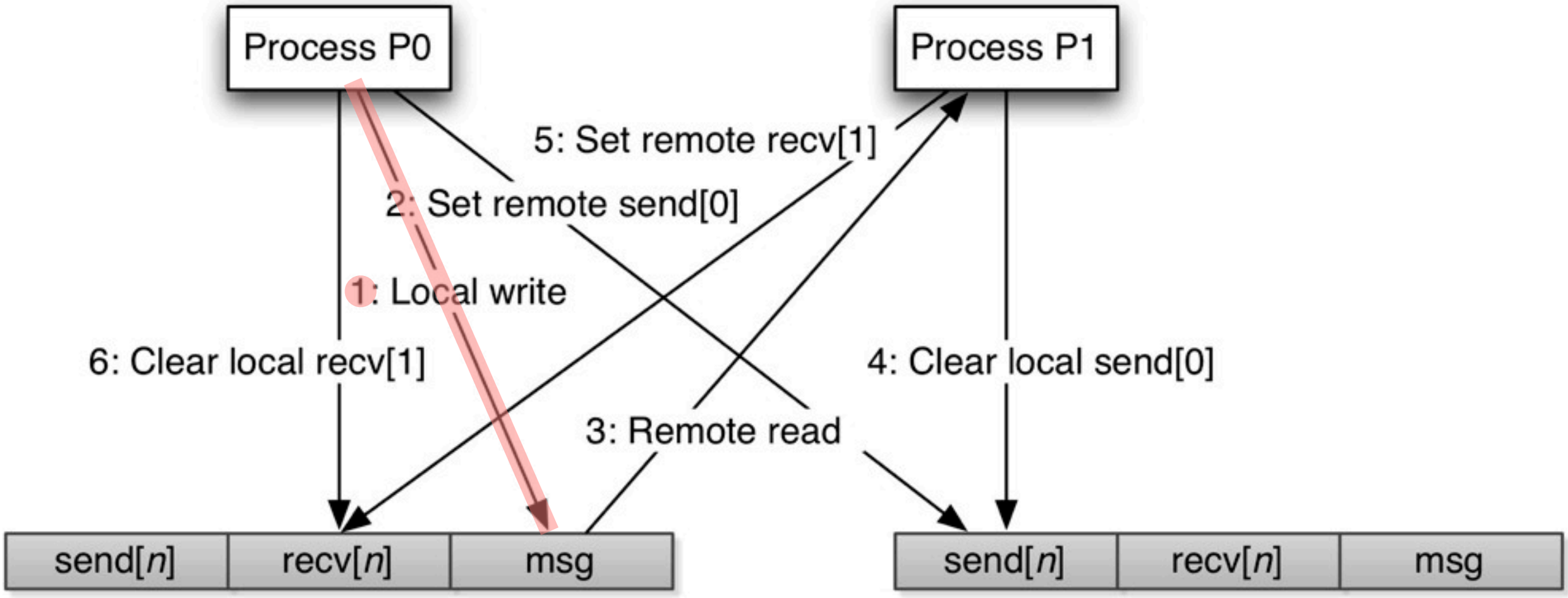
PURDUE UNIVERSITY

RCCE-X10 send/ receive



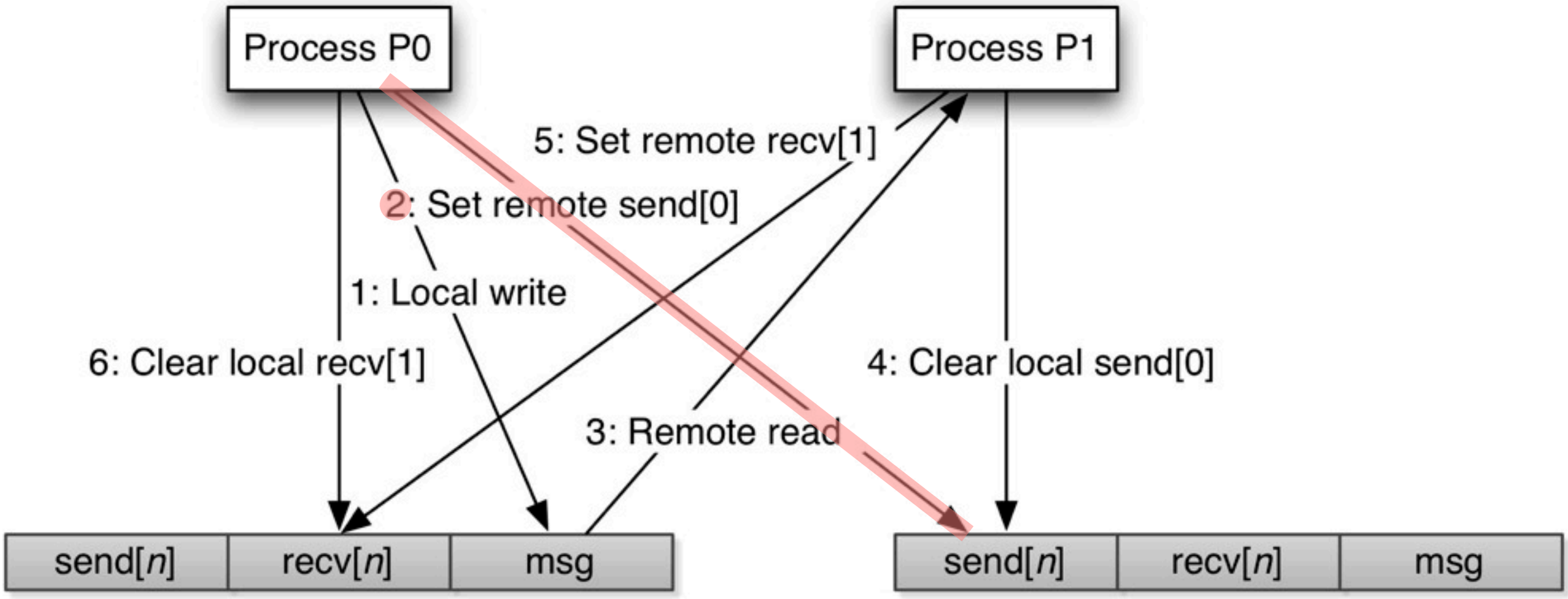
PURDUE UNIVERSITY

RCCE-X10 send/ receive



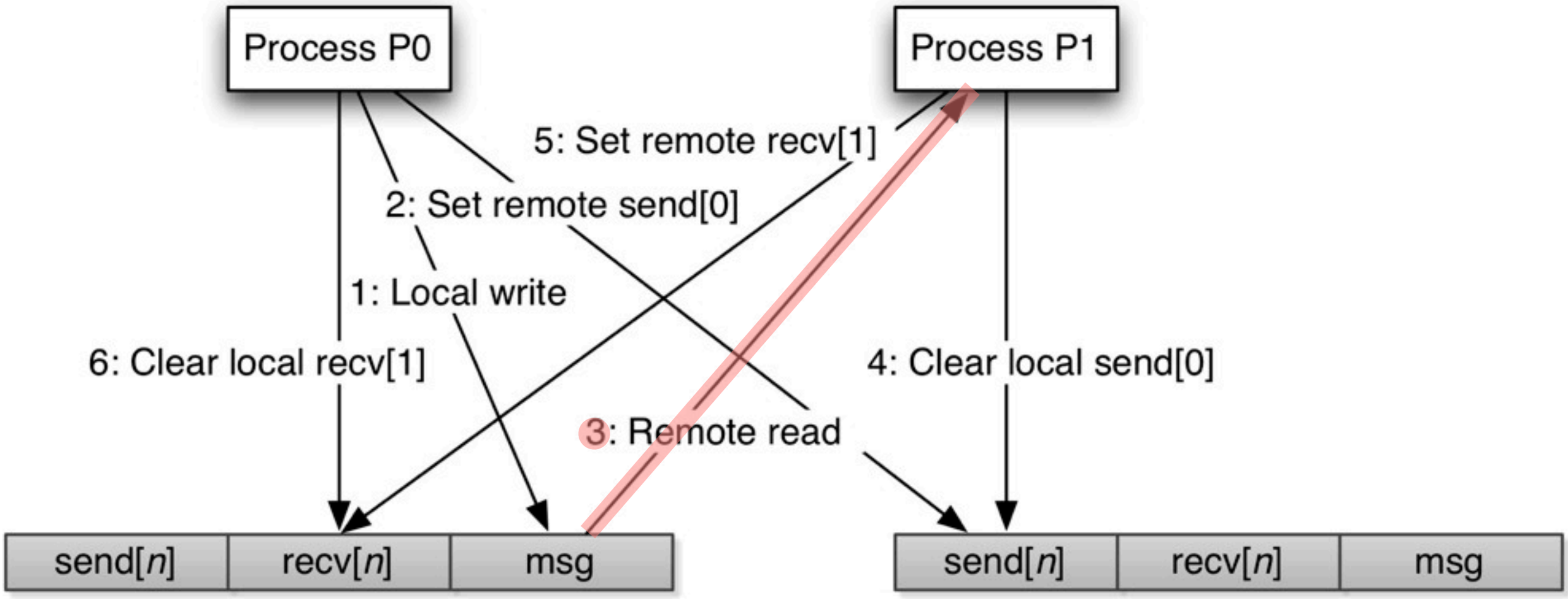
PURDUE UNIVERSITY

RCCE-X10 send/ receive



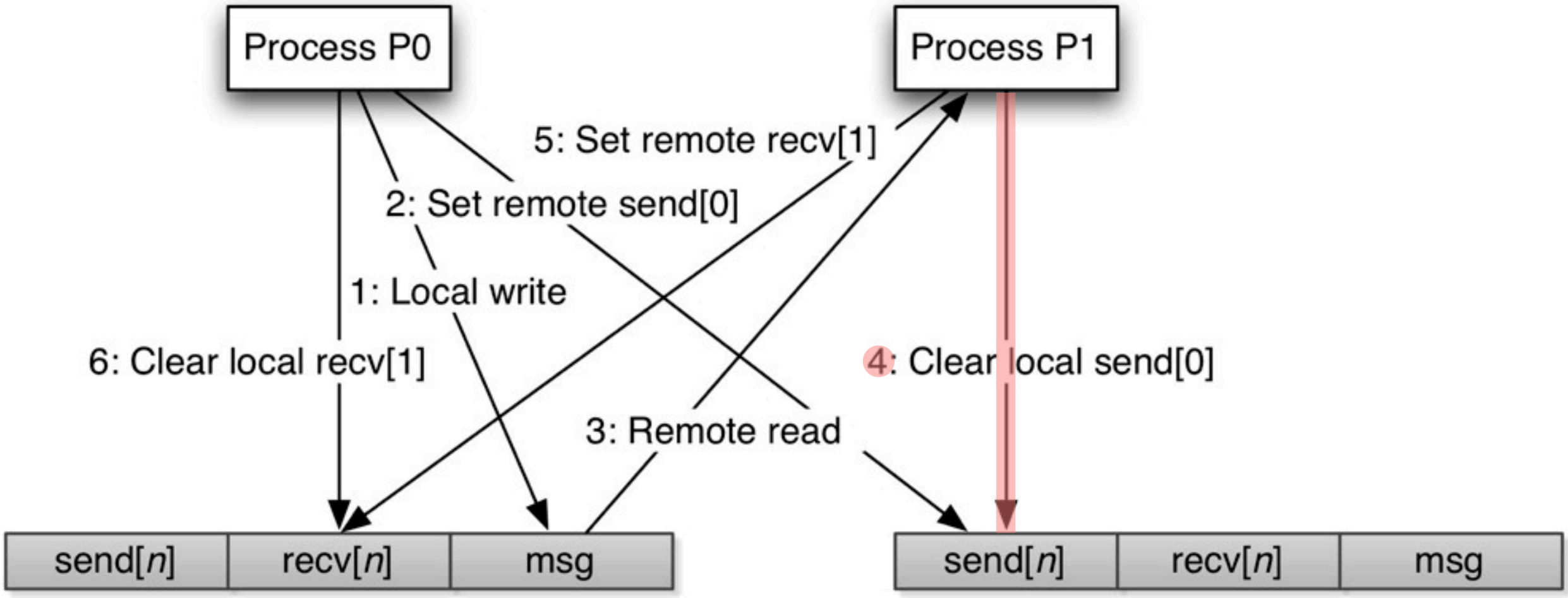
PURDUE UNIVERSITY

RCCE-X10 send/ receive



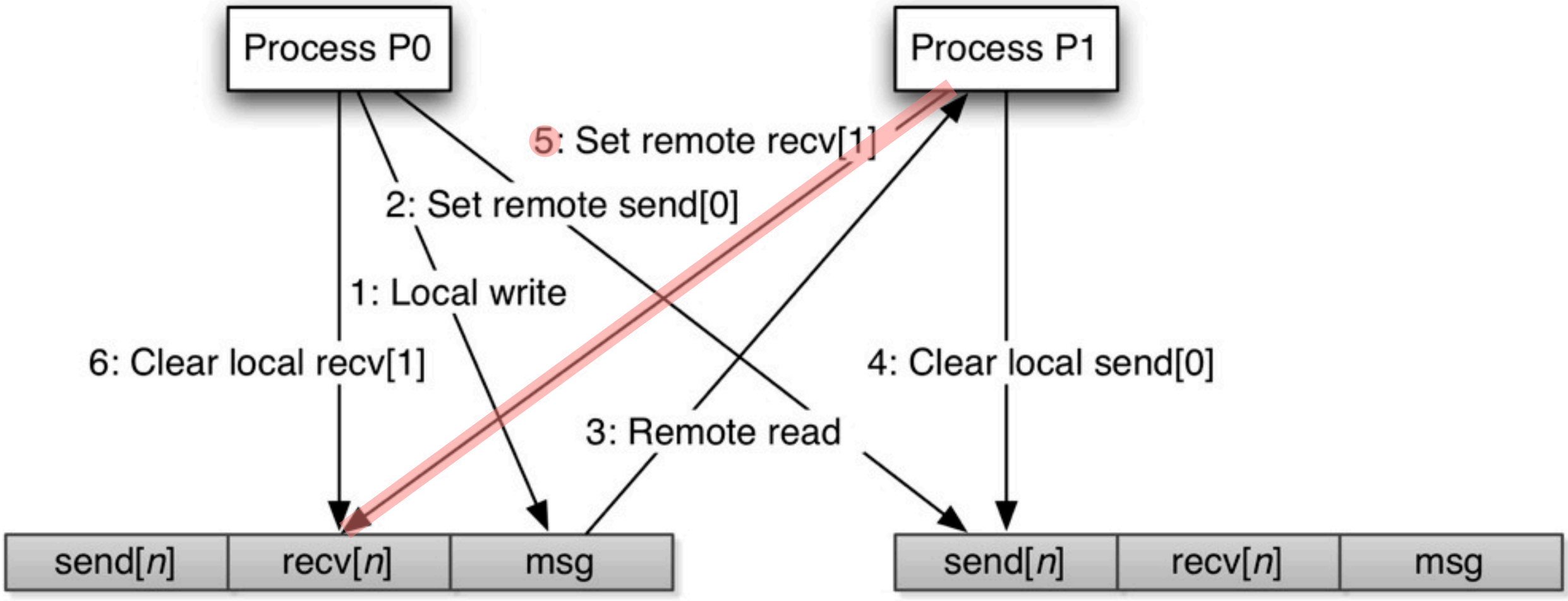
PURDUE UNIVERSITY

RCCE-X10 send/ receive



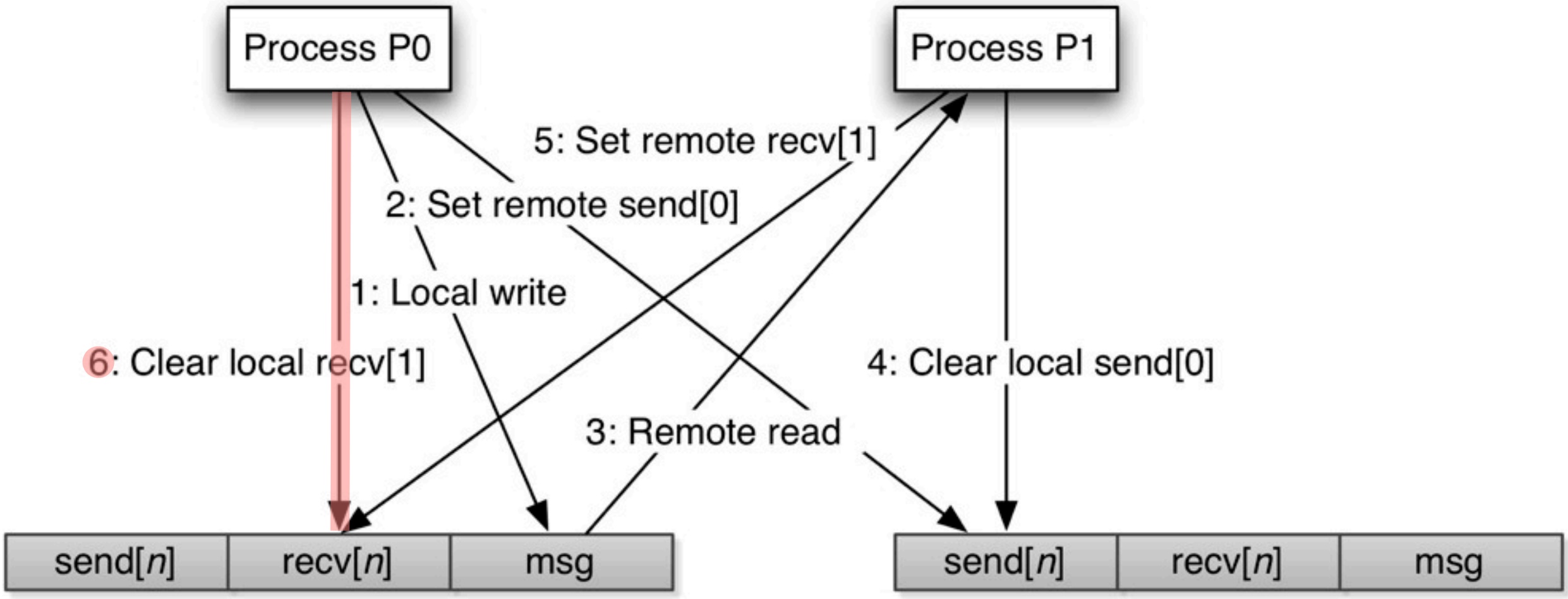
PURDUE UNIVERSITY

RCCE-X10 send/ receive



PURDUE UNIVERSITY

RCCE-X10 send/ receive



PURDUE UNIVERSITY

Experiments

- One place per SCC core
- One worker thread per place
- Results in paper without GC
- 322 MB private memory per core
- 2 benchmarks: ANUChem-HF and BC
- X10 post-2.1.1 trunk dated January 13th

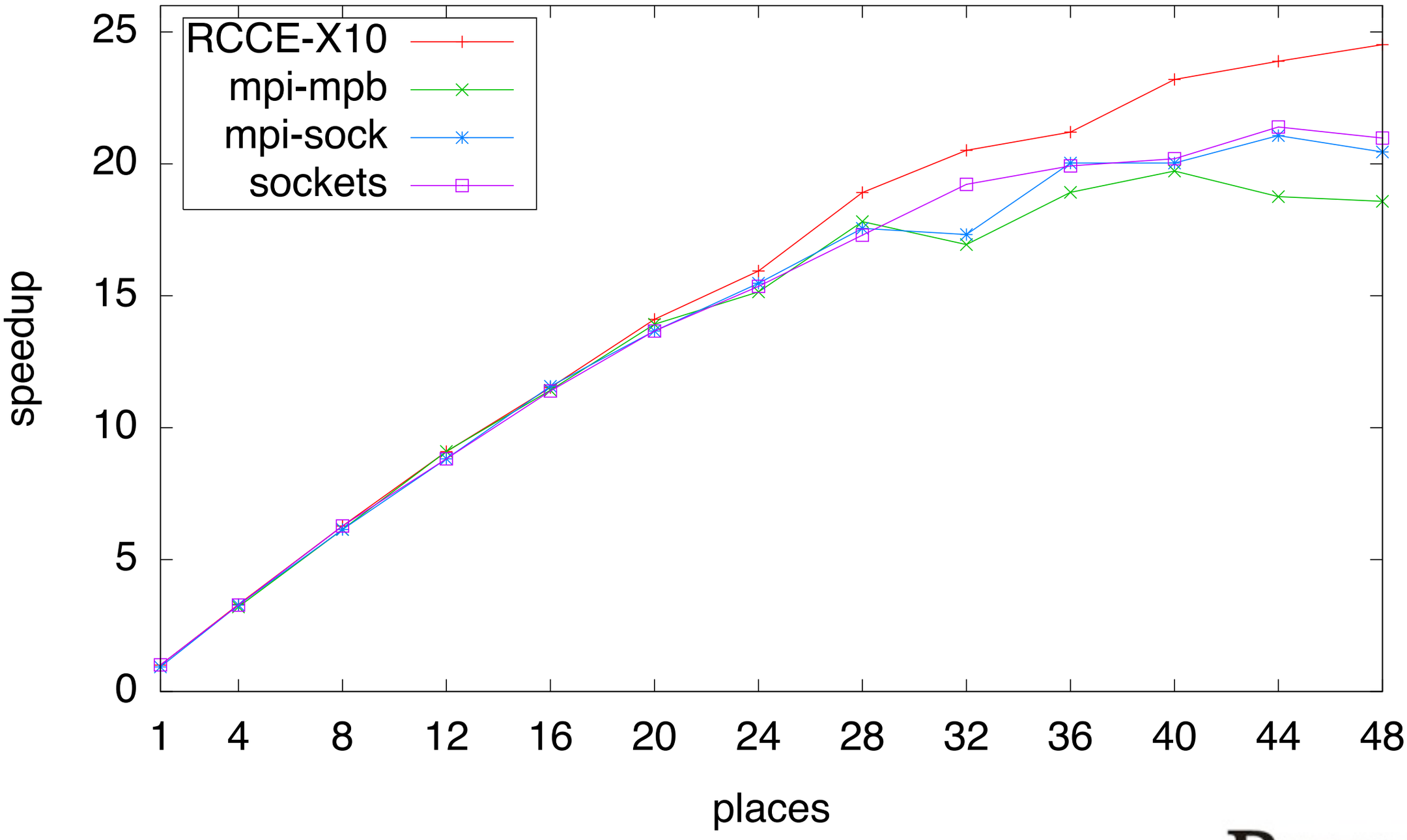
PURDUE UNIVERSITY

Benchmarks

- ANUChem Hartree-Fock (HF) quantum chemistry benchmark
  - Benzene STO-3G workload (in paper)
  - Benzene 3-21G with GC
- Betweenness Centrality (BC) social network actor centrality benchmark
  - Dijkstra's shortest path algorithm
  - small, medium, large workloads

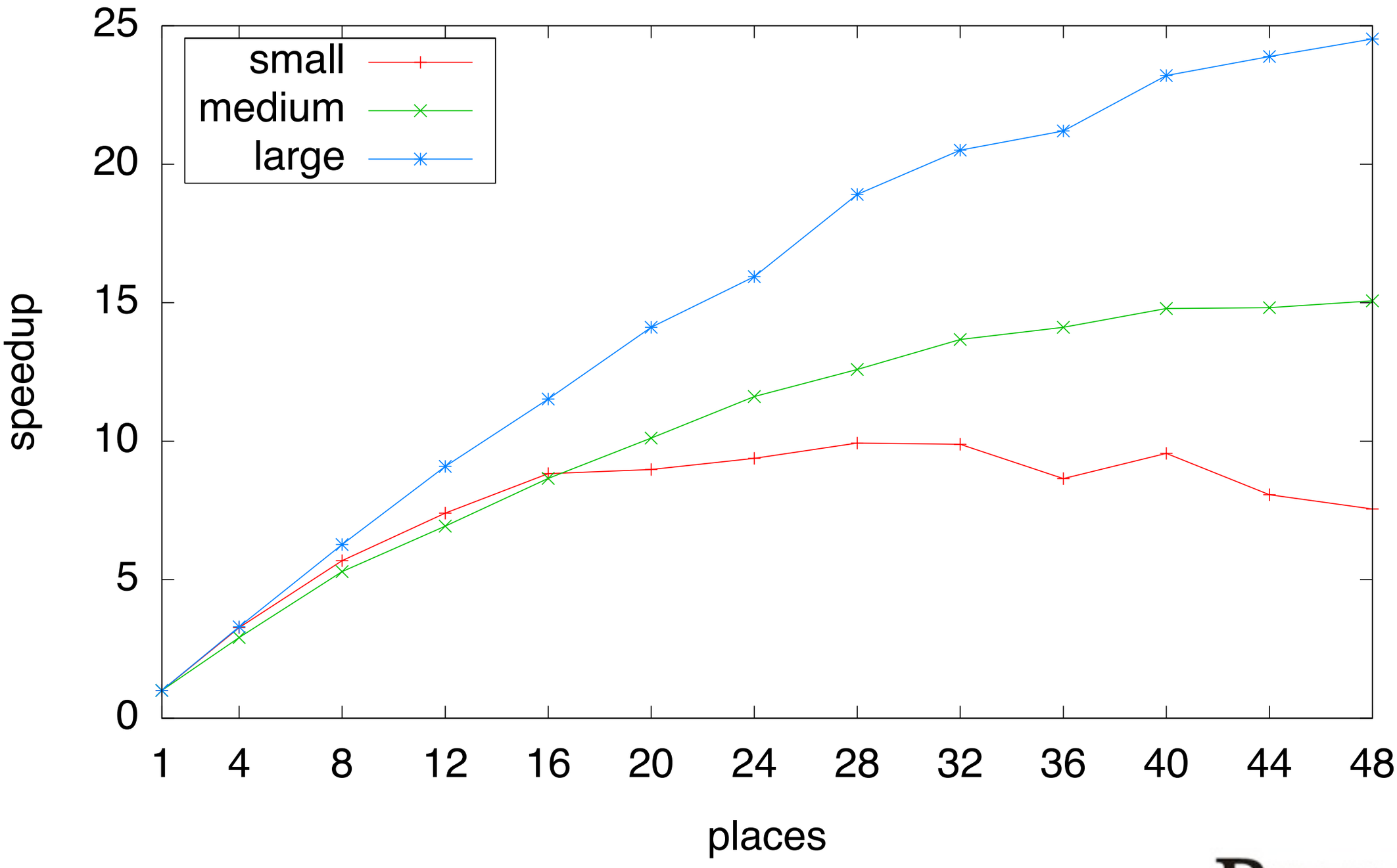
PURDUE UNIVERSITY

BC large workload



PURDUE UNIVERSITY

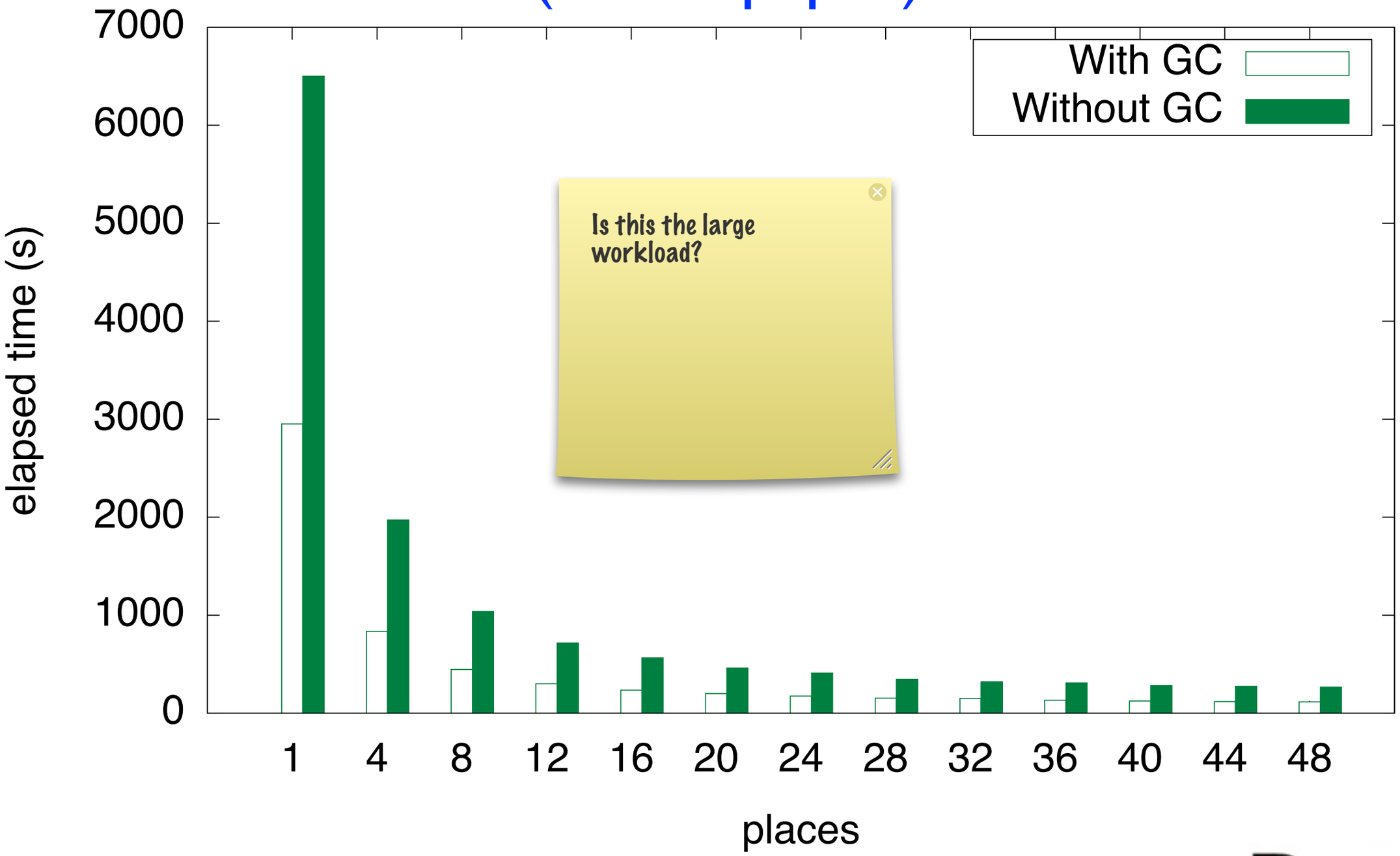
BC varying workload



PURDUE UNIVERSITY

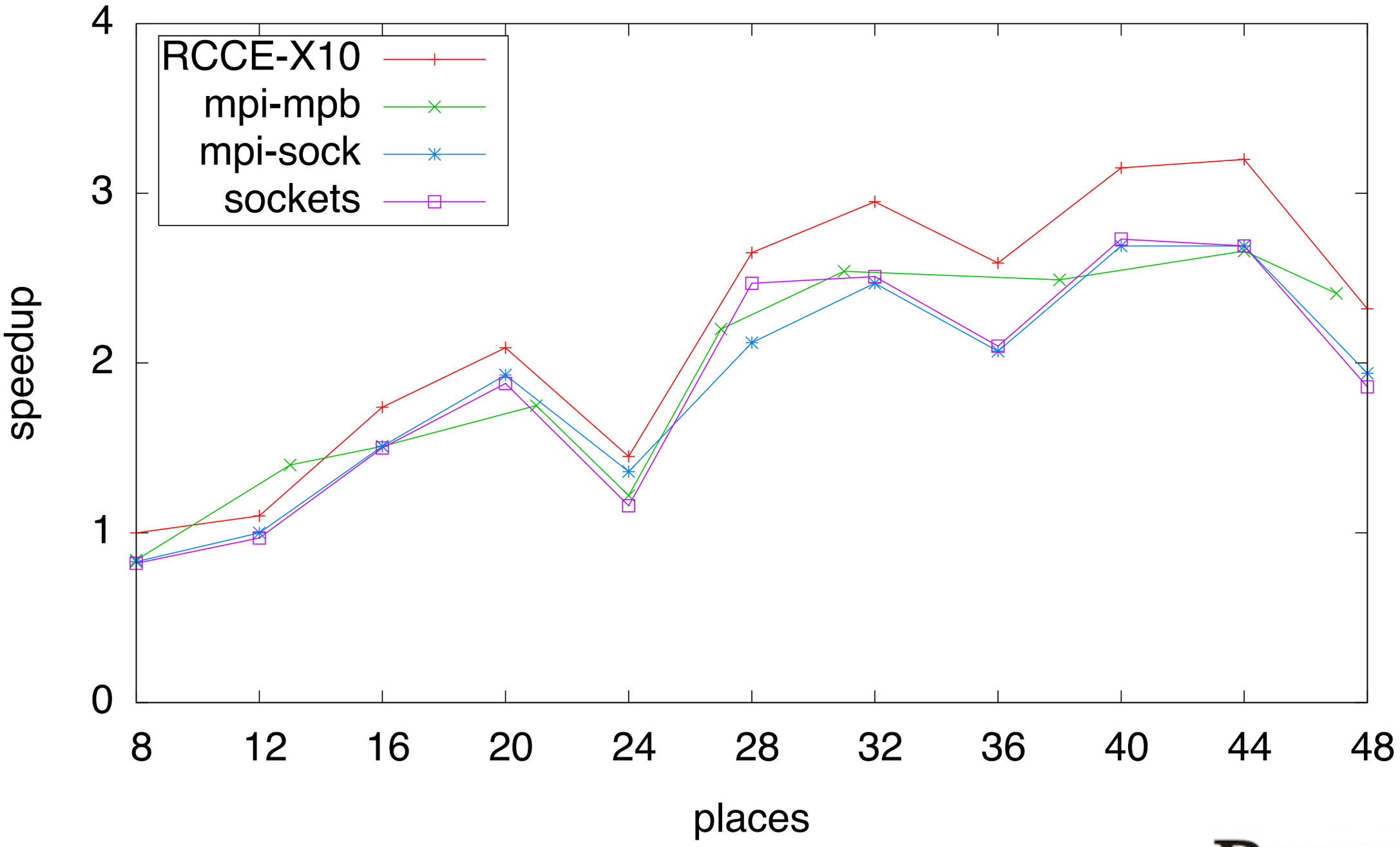
BC with RCCE-X10

(not in paper)



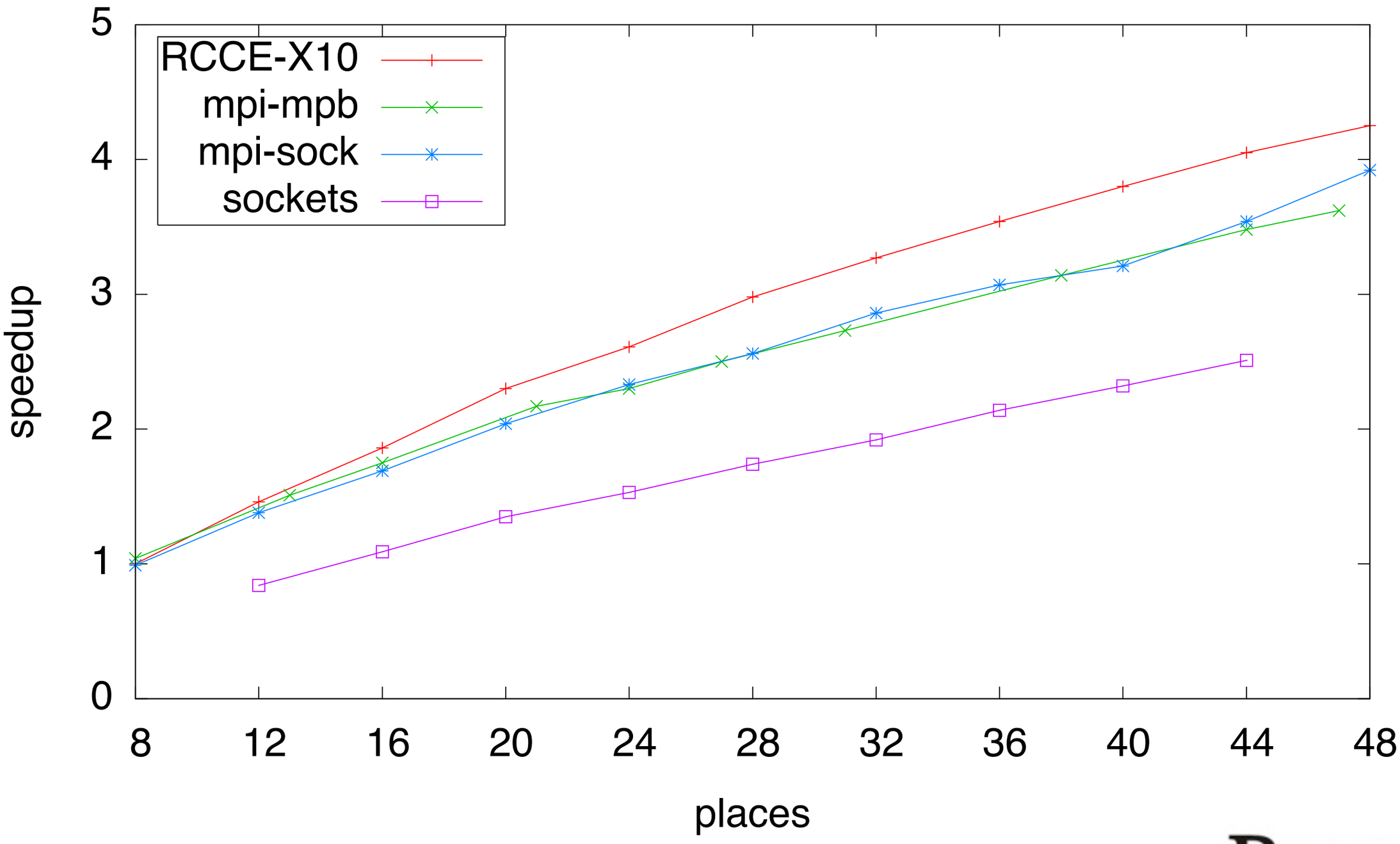
PURDUE UNIVERSITY

HF static balancing



PURDUE UNIVERSITY

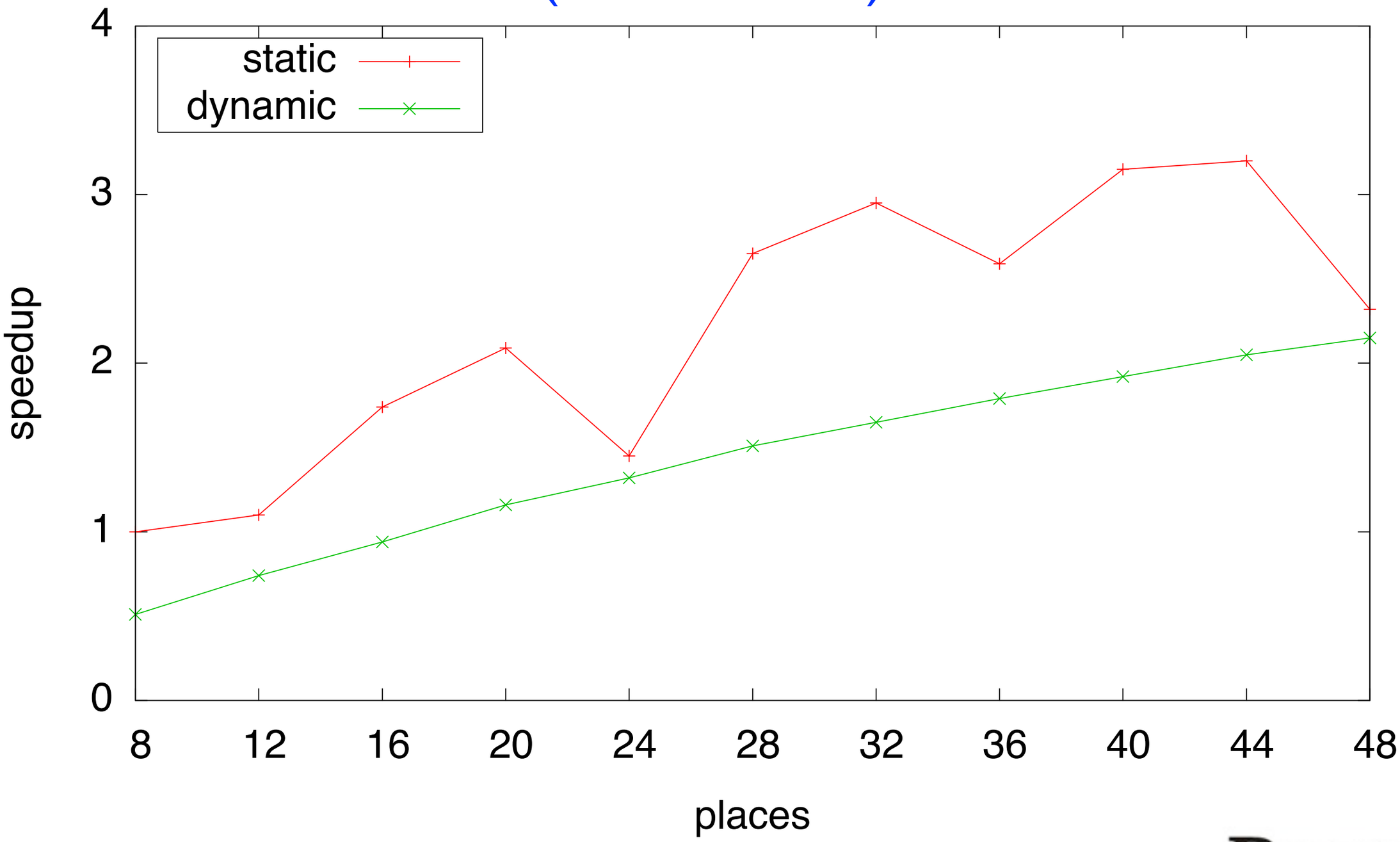
HF dynamic balancing



PURDUE UNIVERSITY

HF dynamic vs static

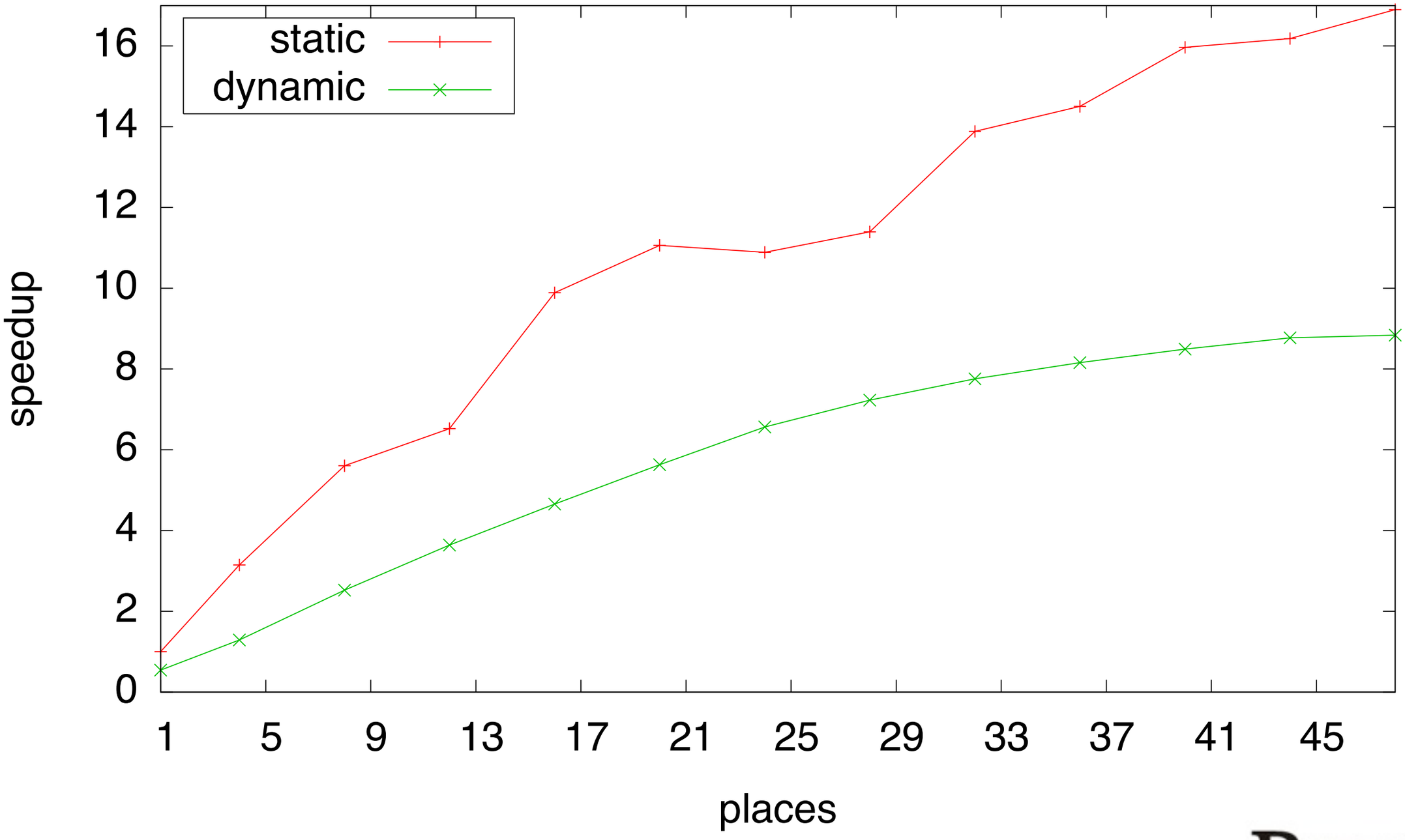
(RCCE-X10)



PURDUE UNIVERSITY

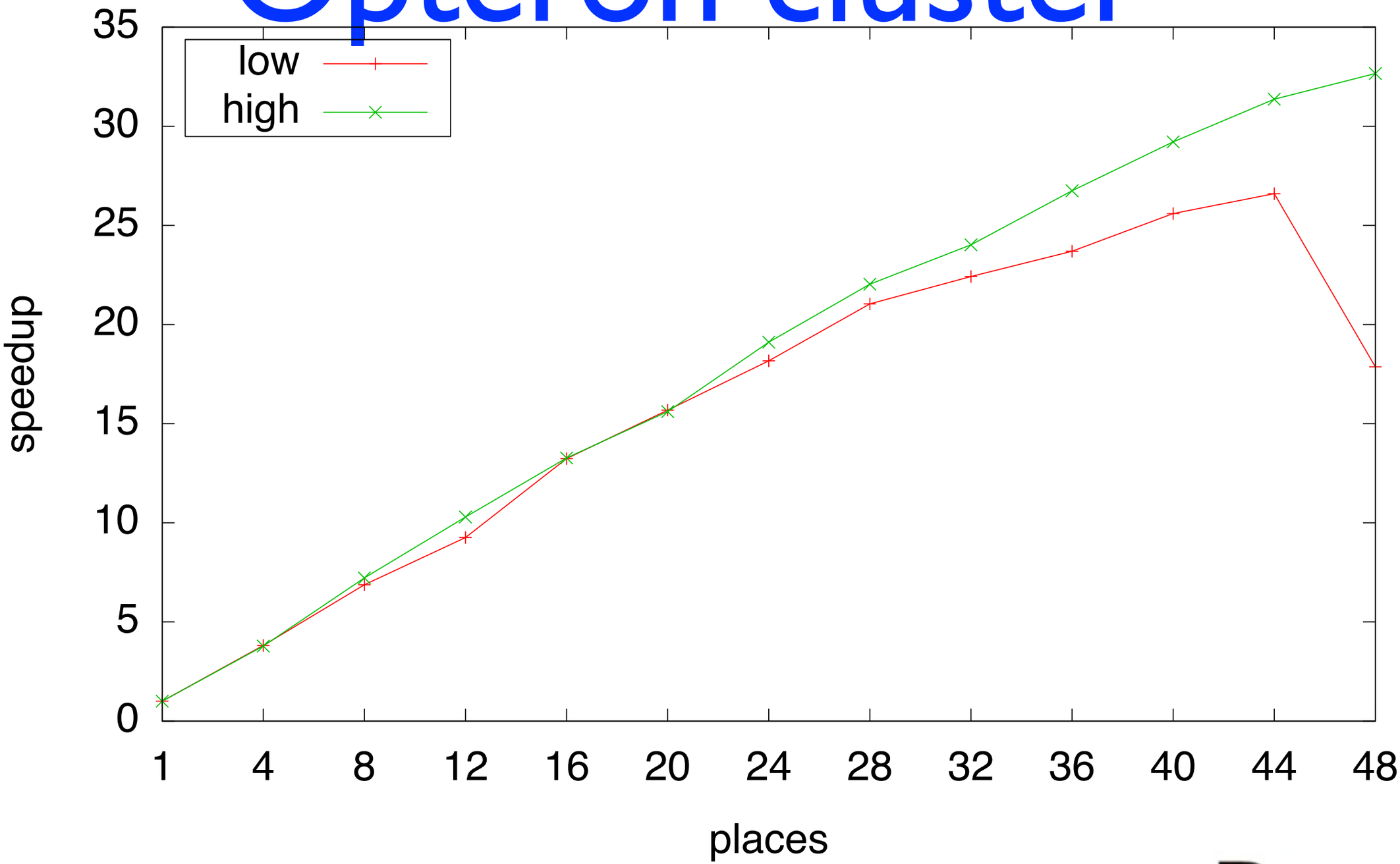
HF with GC

(not in paper)



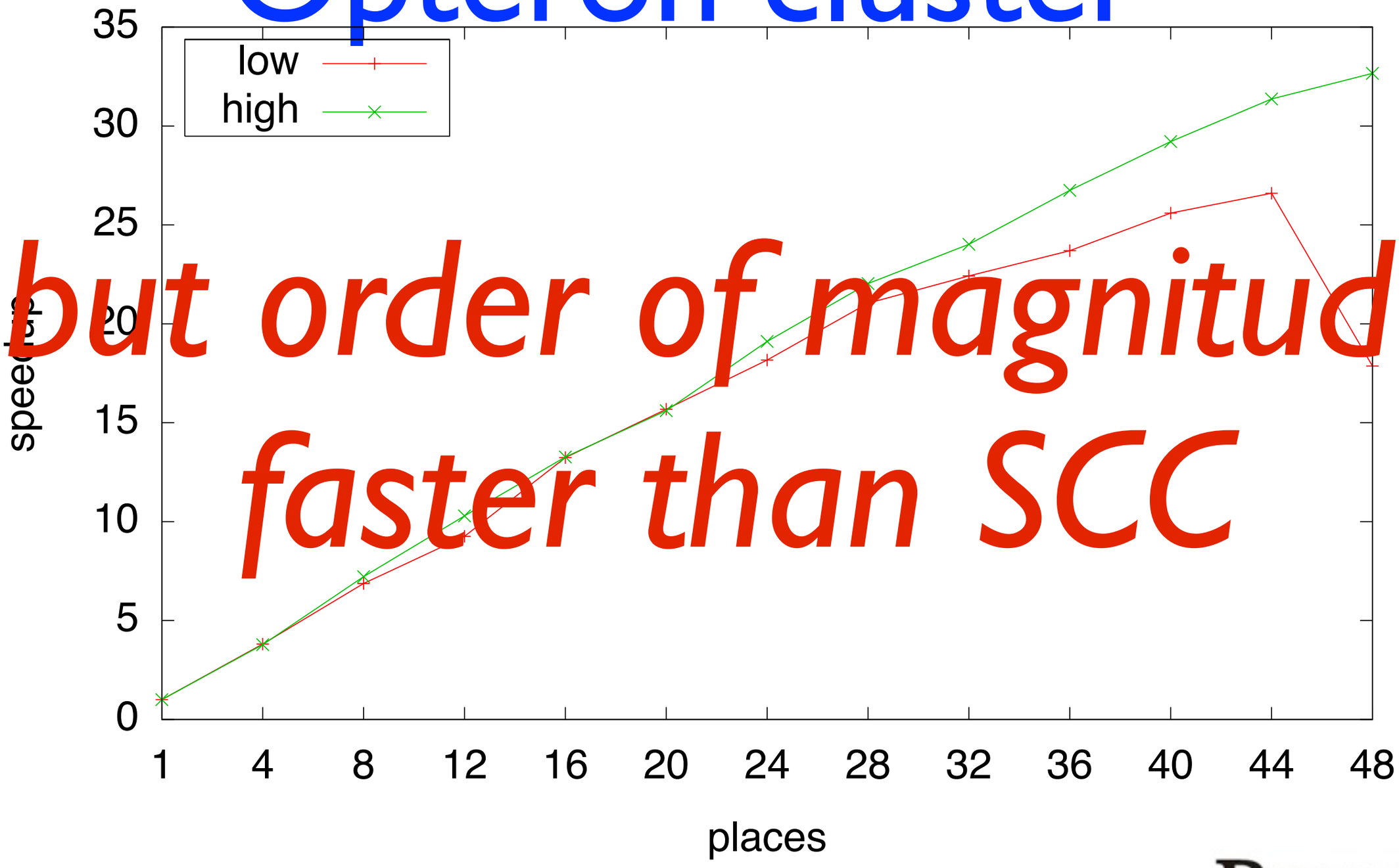
PURDUE UNIVERSITY

BC with MPI on 3x8x2 Opteron cluster



PURDUE UNIVERSITY

BC with MPI on 3x8x2 Opteron cluster



but order of magnitude faster than SCC

PURDUE UNIVERSITY

Conclusions

- SCC memory constraints are limiting:
  - tradeoff work vs memory to get scaling
  - running with GC allows larger workloads (and better performance!)
- But, X10 + SCC is a nice match
- X10 benchmarks (can) show good scaling
  - order of magnitude slower than cluster

PURDUE UNIVERSITY