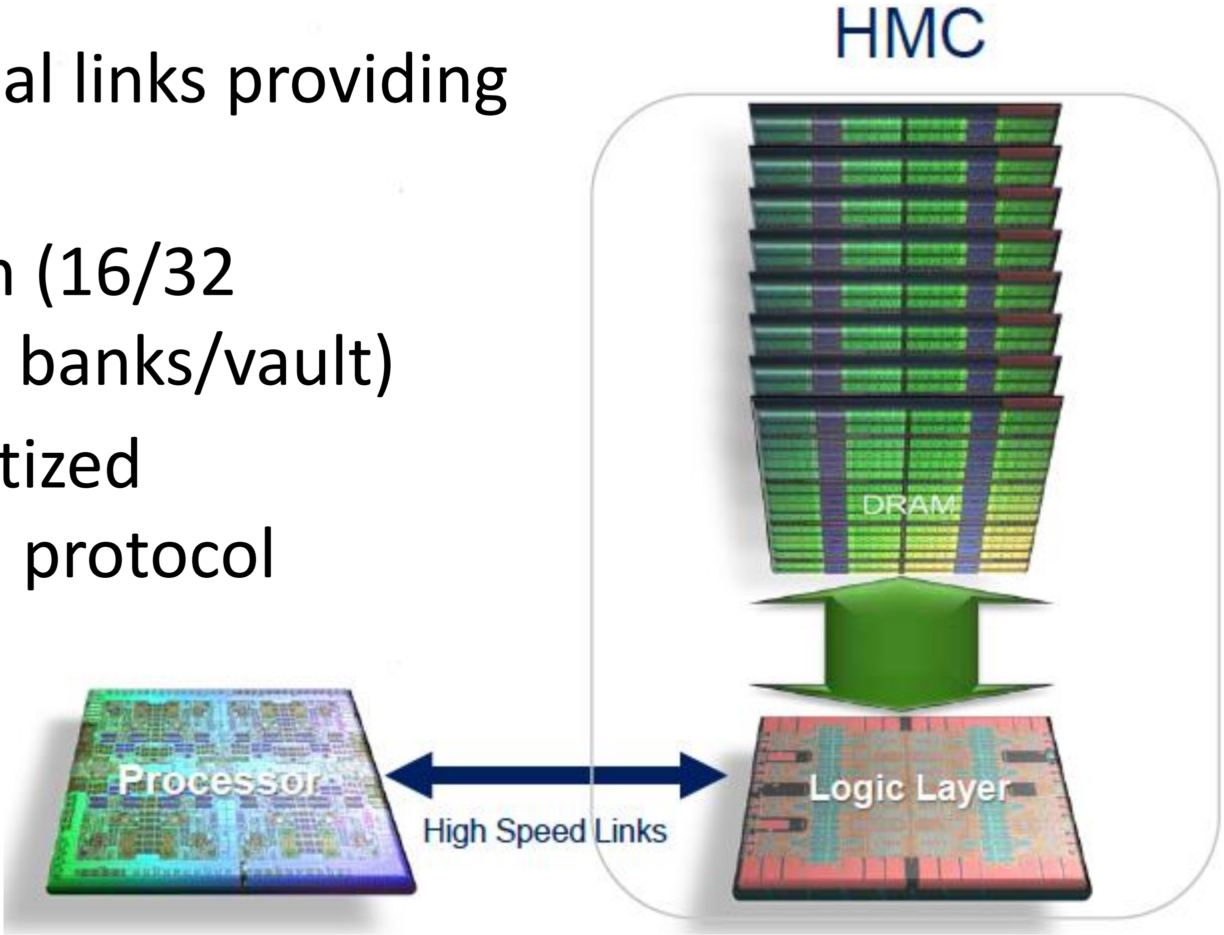
Low Power Hybrid Memory Cube with Link On/Off Management

2014. 7. 10

Junwhan Ahn, Sungjoo Yoo*, Kiyoung Choi
Seoul National Univ. & POSTECH*

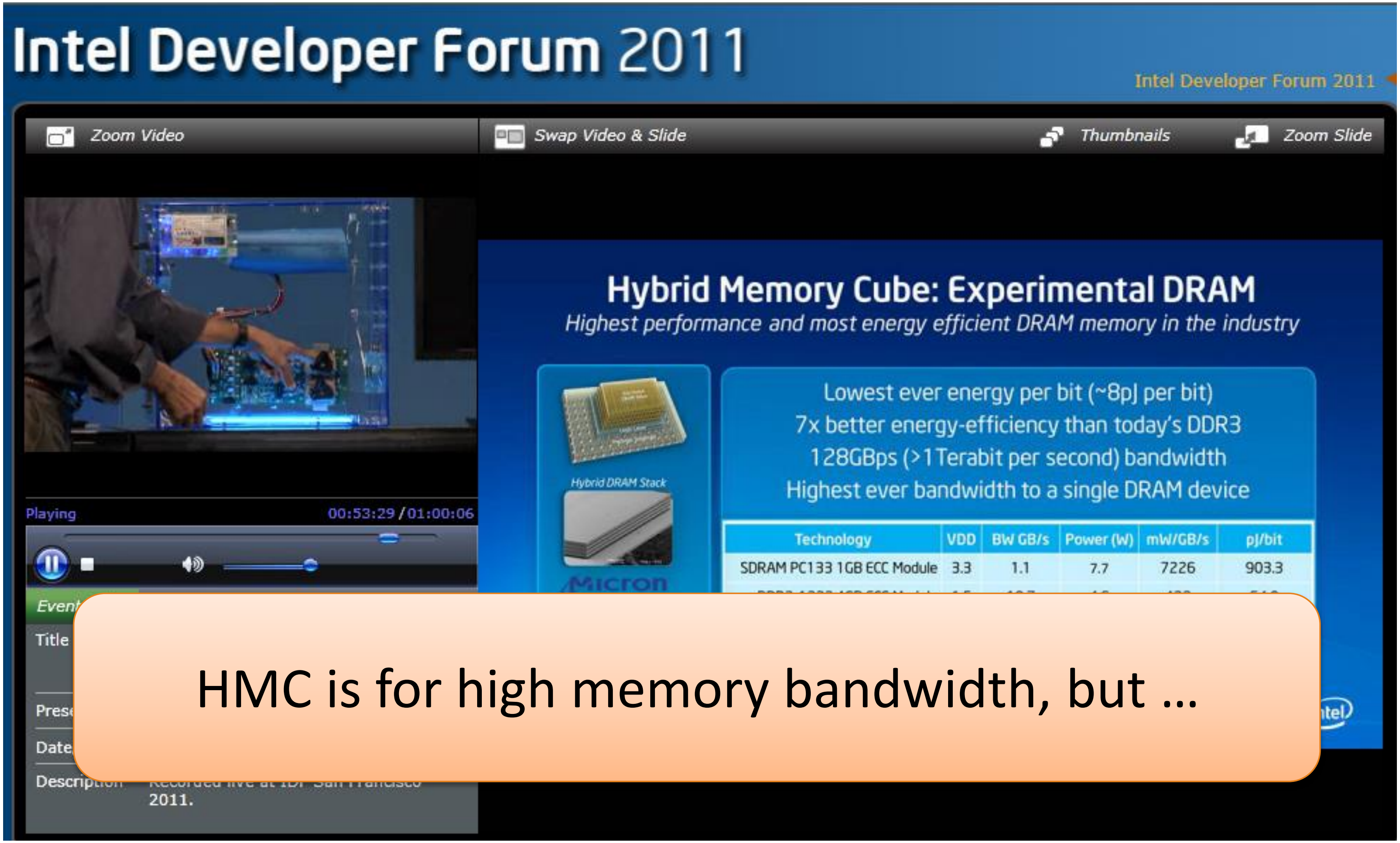
Hybrid Memory Cube (HMC)

- 3D-stacked DRAM with an integrated logic die
- High-speed serial links providing up to 320GB/s
- High parallelism (16/32 *vaults*/cube, 16 banks/vault)
- Abstract, packetized communication protocol



[Micron]

Hybrid Memory Cube



Recorded live at IDF San Francisco 2011.

Hybrid Memory Cube: Experimental DRAM

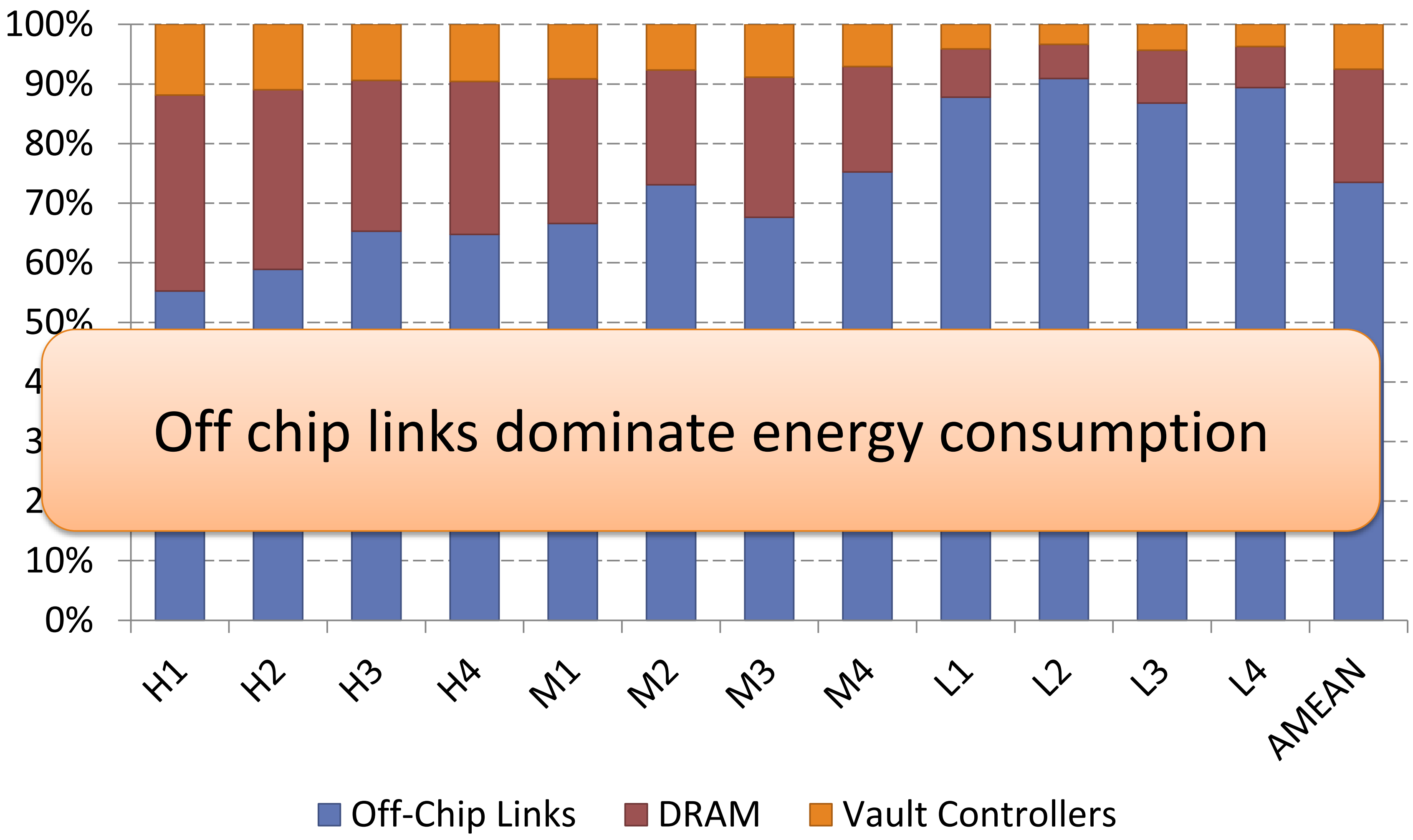
Highest performance and most energy efficient DRAM memory in the industry

Lowest ever energy per bit (~8pJ per bit)
7x better energy-efficiency than today's DDR3
128GBps (>1Terabit per second) bandwidth
Highest ever bandwidth to a single DRAM device

Technology	VDD	BW GB/s	Power (W)	mW/GB/s	pJ/bit
SDRAM PC133 1GB ECC Module	3.3	1.1	7.7	7226	903.3

HMC is for high memory bandwidth, but …

Energy Breakdown

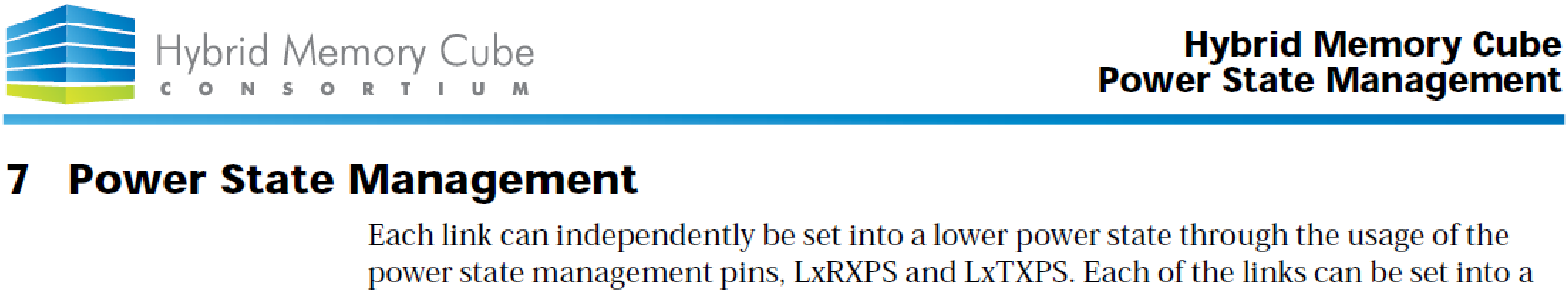


Off chip links dominate energy consumption

Motivation

- Problem: off-chip links of HMC consumes significant static power even when they are not used

- HMC provides power state management for links
  - Sleep mode: SerDes of each link is turned off
  - Down mode: SerDes & PLL are turned off



Hybrid Memory Cube CONSORTIUM

Hybrid Memory Cube
Power State Management

7 Power State Management

Each link can independently be set into a lower power state through the usage of the power state management pins, LxRXPS and LxTXPS. Each of the links can be set into a

Motivation

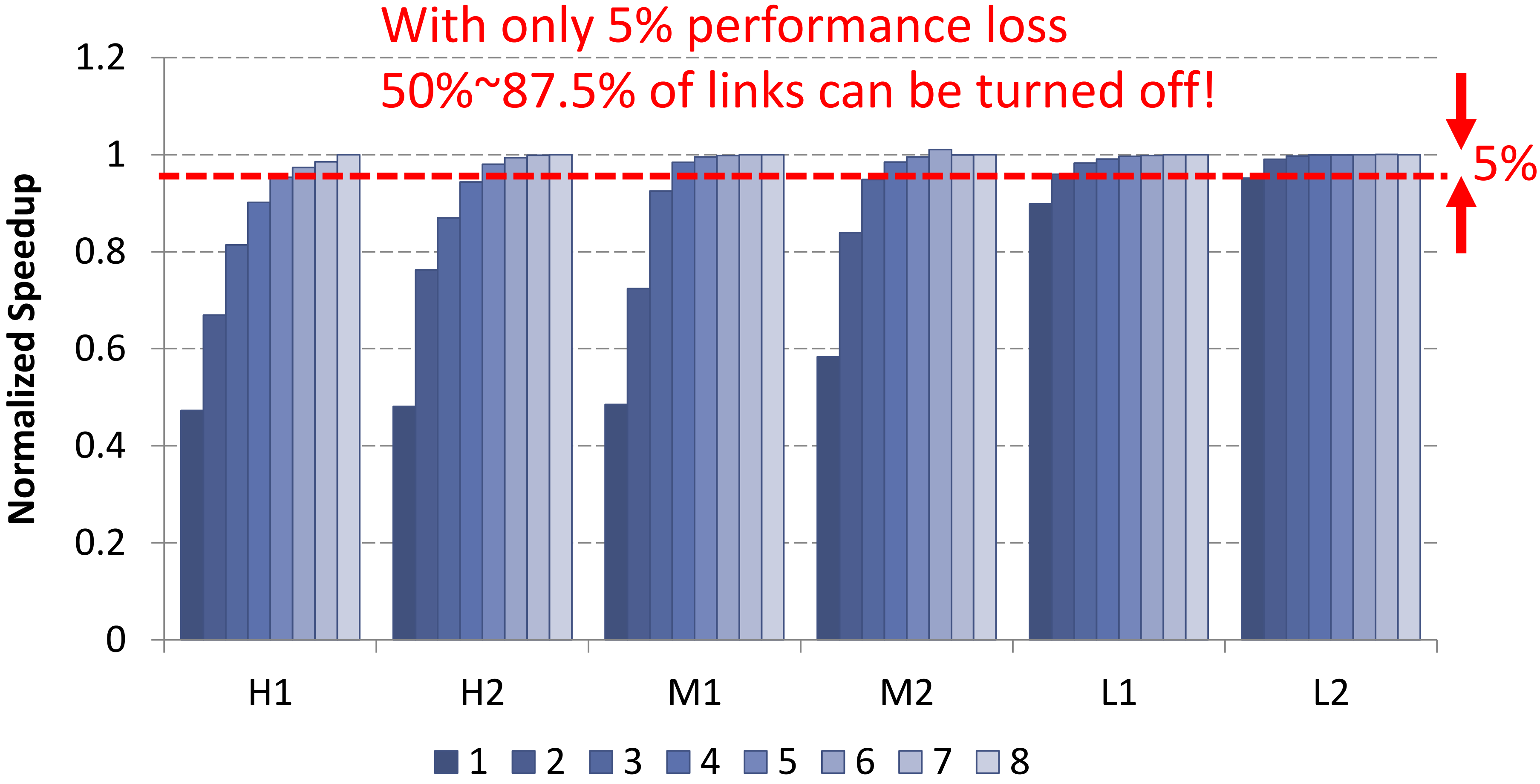
- HMC provides power state management for links
  - Sleep mode: SerDes of each link is turned off
  - Down mode: SerDes & PLL are turned off

- Problem: very long sleep/wakeup latencies
  - **650ns** to enter sleep mode (2,040 cycles at 3GHz)
  - **150us** to enter down mode (450,000 cycles at 3GHz)

Impact of long sleep/wakeup latencies on system performance should be minimized

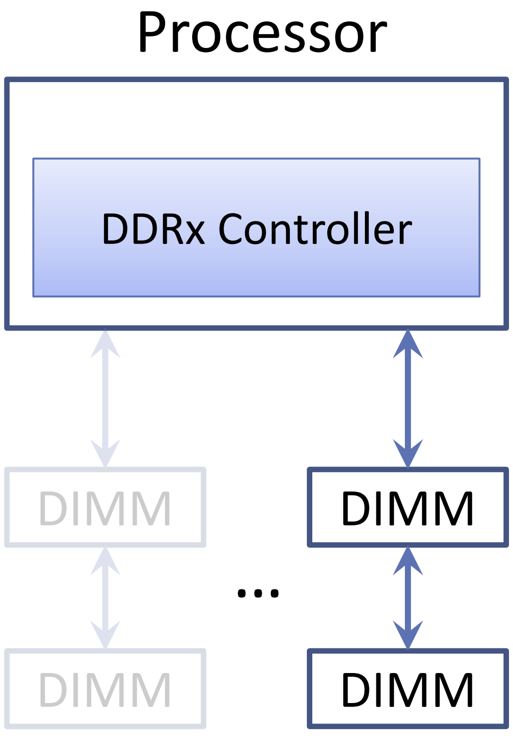
Motivation I: Performance vs. # Links

- Bandwidth demand varies across applications

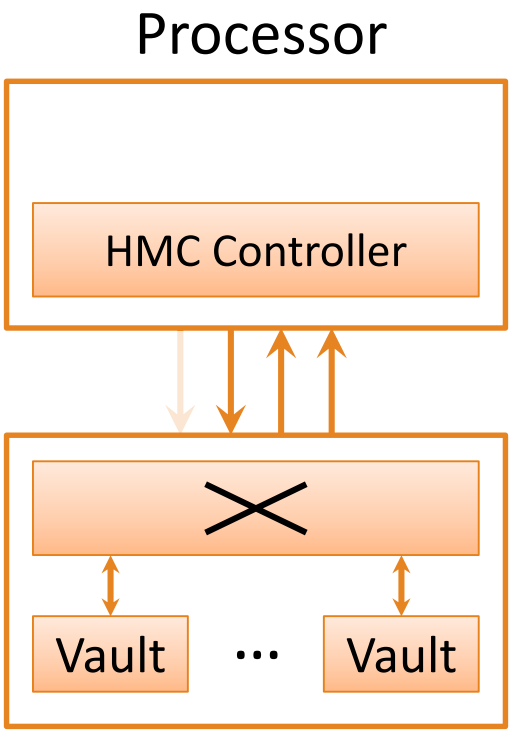


Opportunity to Turn Off Links

- Multi-channel DIMM cannot turn off links, i.e., channels
- HMC has an internal crossbar which enables link turn-off



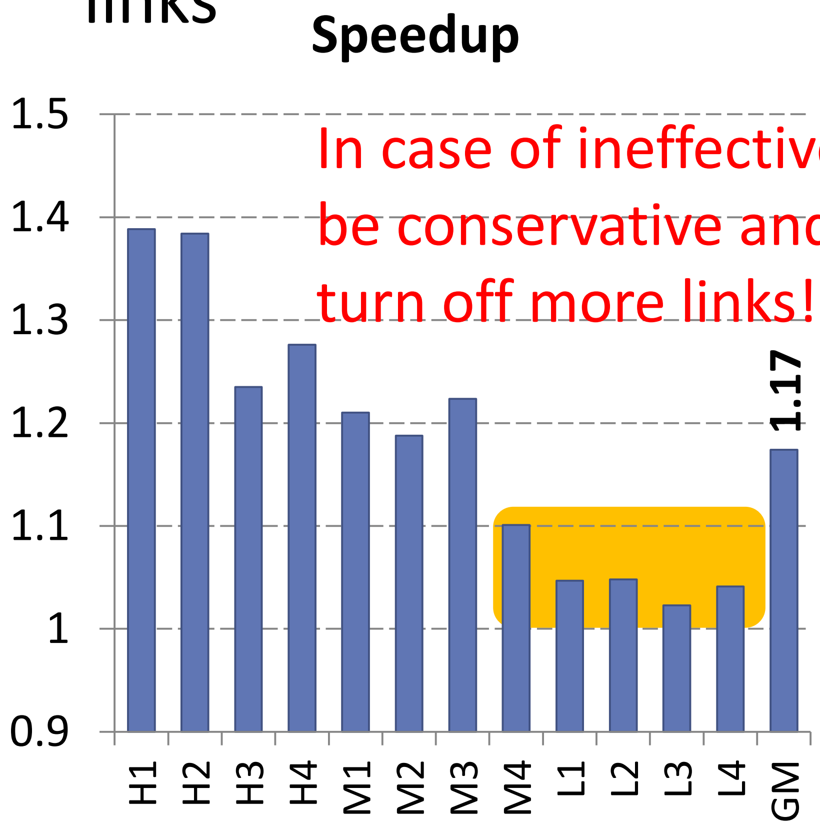
DDRx SDRAM

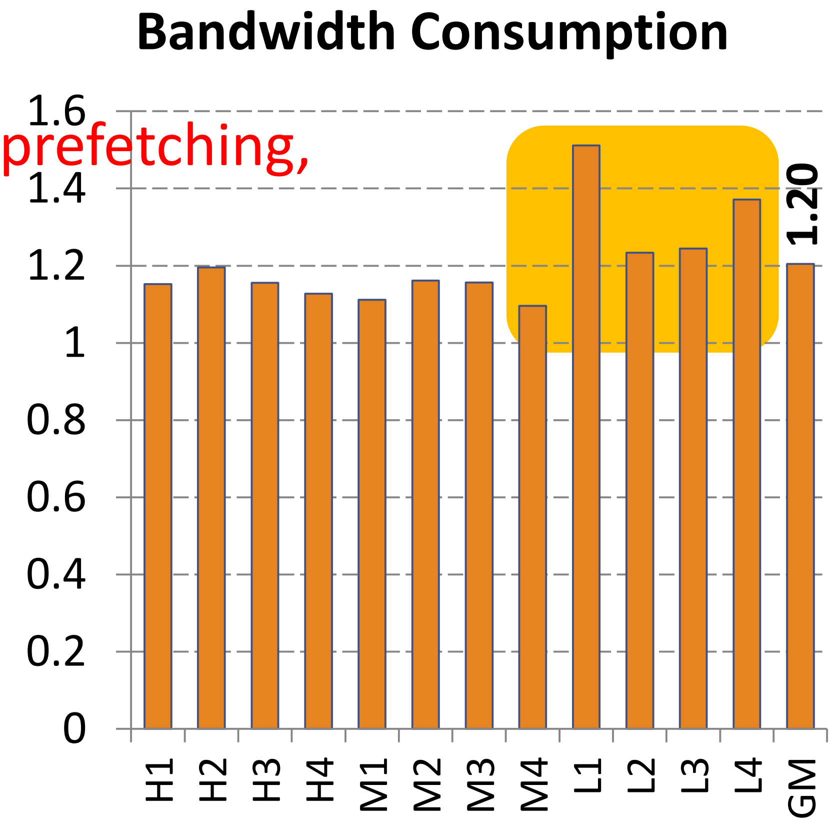


HMC

Motivation II: Prefetching vs. Link Turn-Off

- Prefetching complicates link on/off control
- In-effective prefetching prevents us from turning off links





In case of ineffective prefetching, be conservative and turn off more links!

Solution Overview

- HMC link power management
  - Turn off as many links as possible with a small performance loss

- Two-level prefetching
  - Become conservative when prefetching is not effective, which enables us to turn off more links

HMC Link Power Management

- Objective: find the smallest number of active links n that satisfies the following performance constraint

Longer link delay is allowed
→ Turn off more links

$$\uparrow \frac{\overline{l_n} - \overline{l_N}}{\overline{m}} \leq \alpha \times \frac{1}{u} \downarrow$$

When memory demand is low

  - $\overline{l_n}$: average link delay under n active links ($1 \leq n \leq N$)
  - $\overline{m}$: average memory access latency
  - u: link utilization (= memory demand)

HMC Link Power Management

- Objective: find the smallest number of active links n that satisfies the following performance constraint

$$\frac{\overline{l_n} - \overline{l_N}}{\overline{m}} \leq \alpha \times \frac{1}{u}$$

  - $\overline{l_n}$: average link delay under n links ($1 \leq n \leq N$)
  - $\overline{m}$: average memory access latency
  - u: link utilization

How to estimate average link delay for each link configuration?

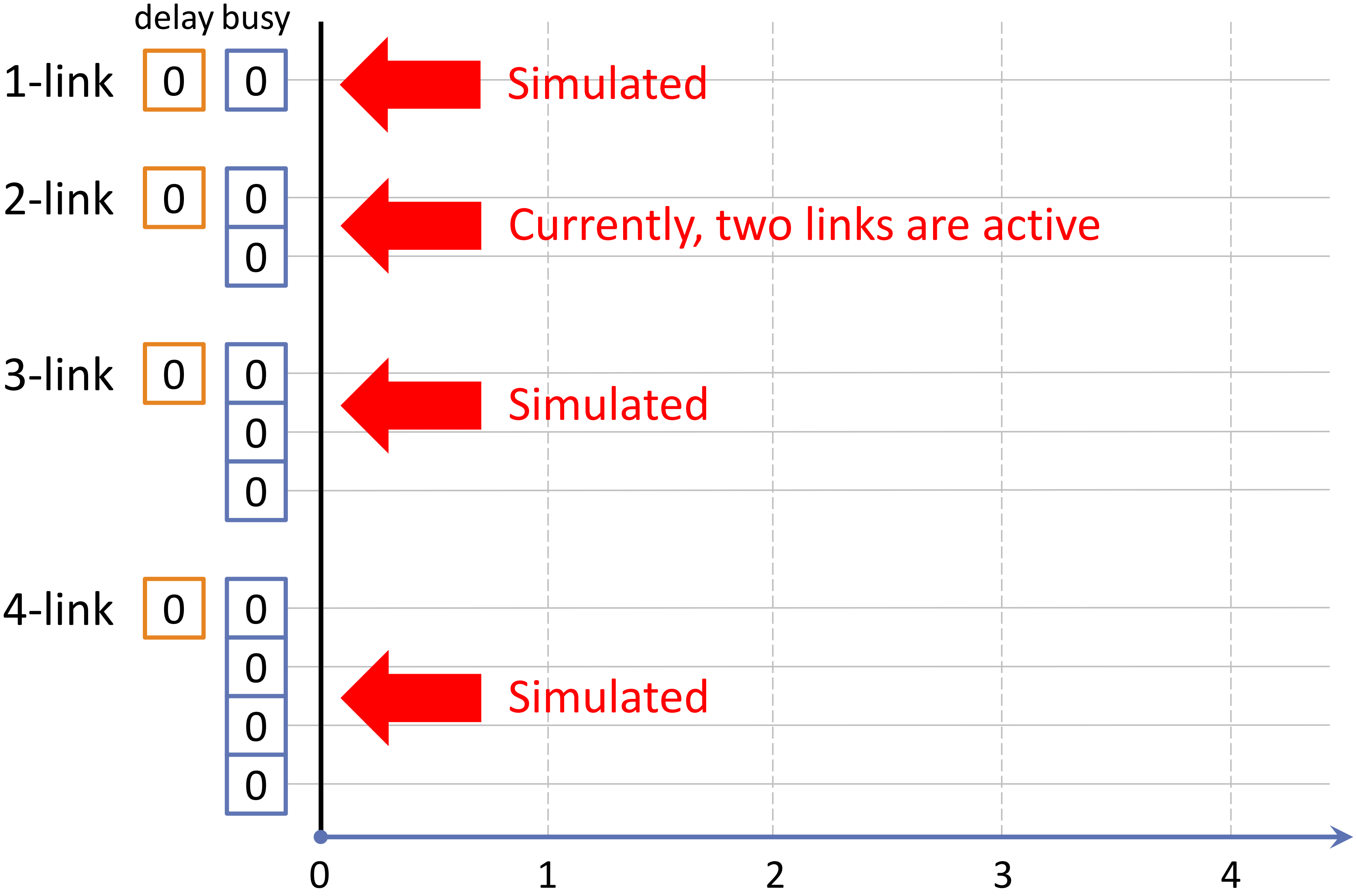
Link Delay Monitor

- Hardware structure that simulates link congestion for all the possible link configurations at the same time

- Two types of counters
  - Delay counter(*n*): total link delay of all requests assuming *n* links
  - Busy counter(*i*, *n*): the number of cycles for the link *i* to become free under *n* links

- Each link configuration (with *n* links) requires n busy counters and one delay counter
  - O(n^2) for busy counters

All The Possible Link Configurations Are Simulated During Runtime



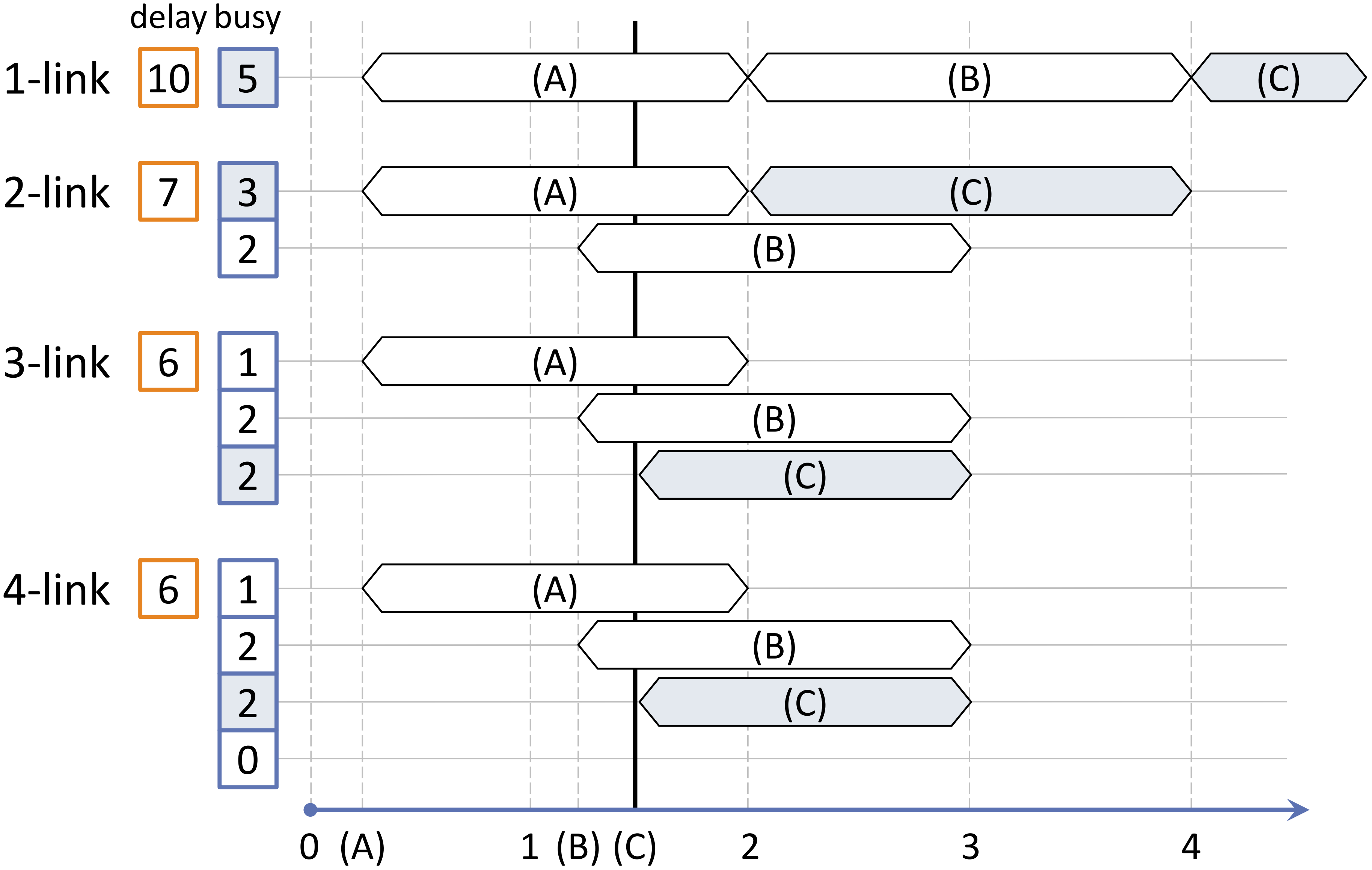
Simulated

Currently, two links are active

Simulated

Simulated

Link Delay Monitor



HMC Link Power Management

- Objective: find the smallest number of active links n that satisfies the following performance constraint

$$\frac{\overline{l_n} - \overline{l_N}}{\overline{m}} \leq \alpha \times \frac{1}{u}$$

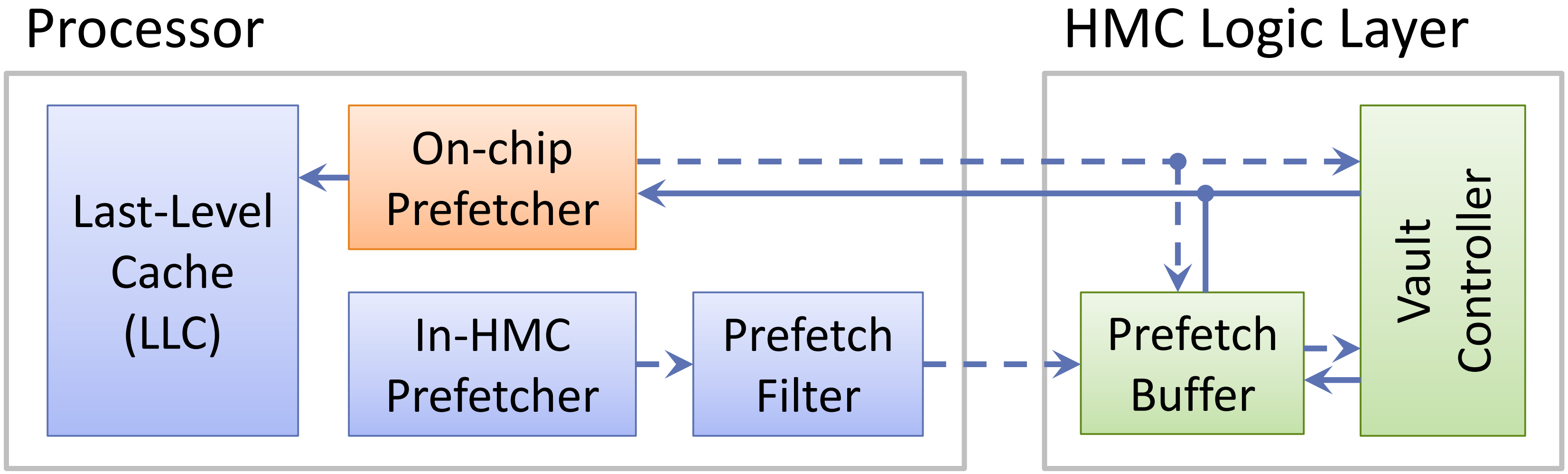
- Periodically adjust the number of active links, *n*
  - Turn on or off only one link at a time
  - Period should be much larger than sleep/wakeup latency

Solution Overview

- HMC link power management
  - Turn off as many links as possible with a small performance loss

- Two-level prefetching
  - Become conservative when prefetching is not effective, which enables us to turn off more links

Two-Level Prefetching



- Level 1: a *conservative* on-chip prefetcher
  - Reduces prefetch traffics thereby enabling us to turn off links
  - However, the conservative prefetcher can incur high miss rate
- Level 2: an *aggressive* in-HMC prefetch buffer
  - Aggressive prefetcher is used to reduce miss penalty by storing prefetched data in the prefetch buffer on logic layer

Evaluation Methodology

- Cycle-accurate x86-64 simulator based on Pin
  - Eight 3GHz, four-issue, out-of-order cores
  - 4MB, 16-way, 64B-block, shared L2 cache
  - Stream prefetcher with different prefetch distance/degree
  - An 8GB HMC with 8 DRAM layers, 32 vaults
  - 8 full-duplex links, 8 10Gb/s lanes per link
- CACTI-3DD & McPAT for HMC modeling
- Workloads: 12 multi-programmed SPEC CPU2006

Comparison

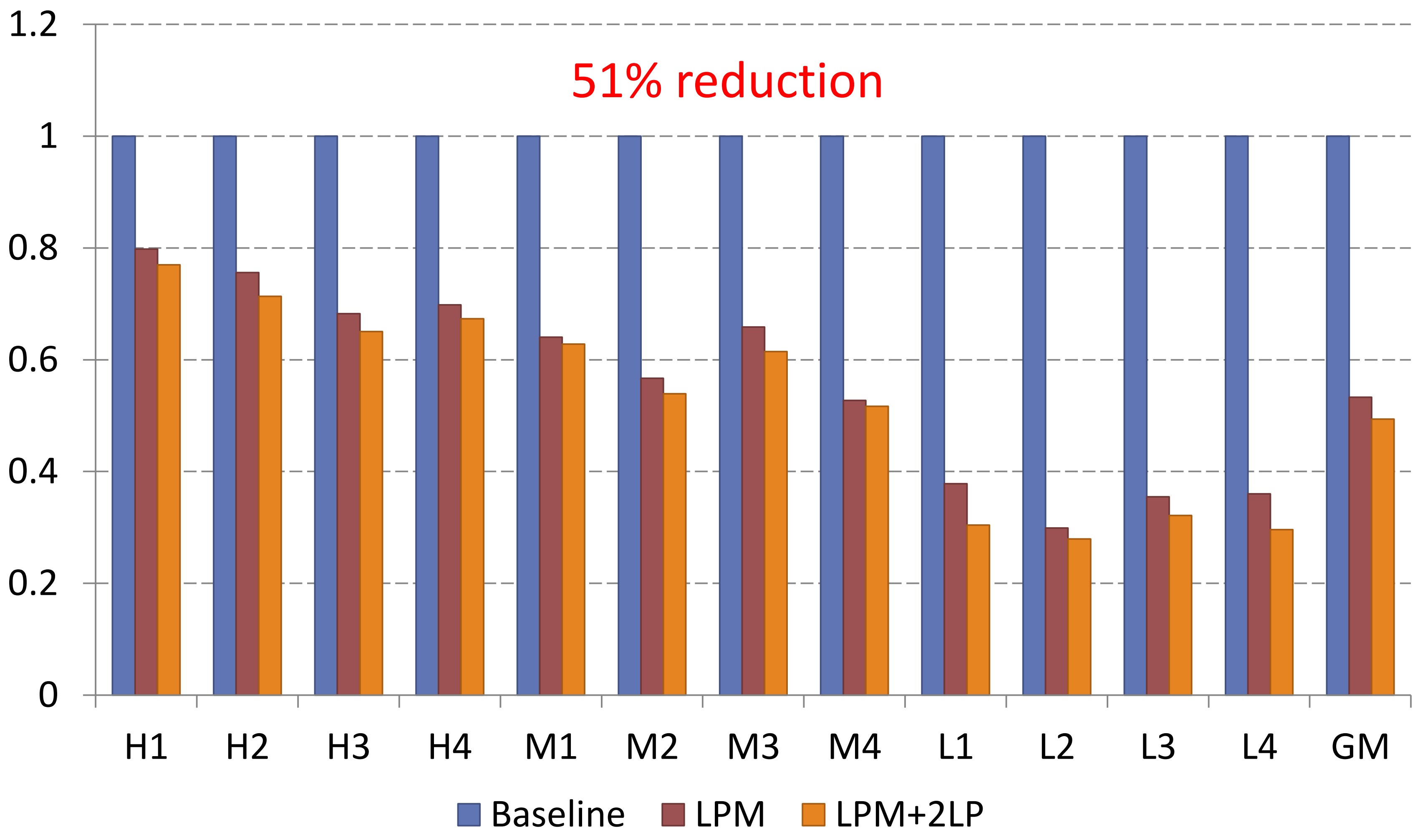
- Static Best: a fixed link configuration for each application

- HDS: Prior work on on/off links of high-degree switches (HDS)

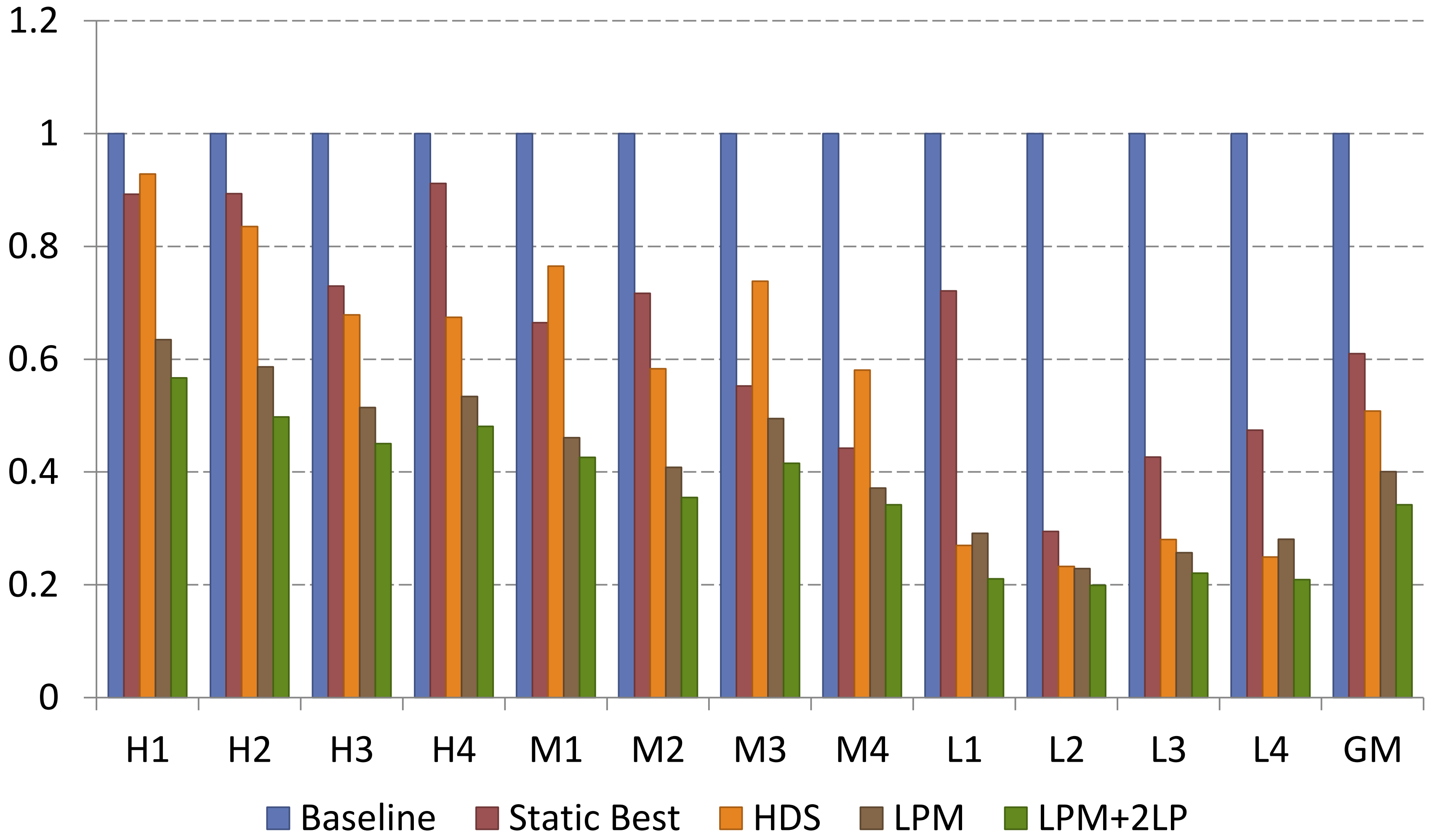
- Proposed link power management (LPM)
  - 100us period, slowdown threshold $\alpha = 0.05$
  - 165 bytes of storage overhead

- Two-level prefetching (2LP)
  - PF distance: 8/64, PF degree: 1/4 (on-chip/in-HMC)
  - 16KB per vault, 16-way, 64B-block in-HMC prefetch buffer

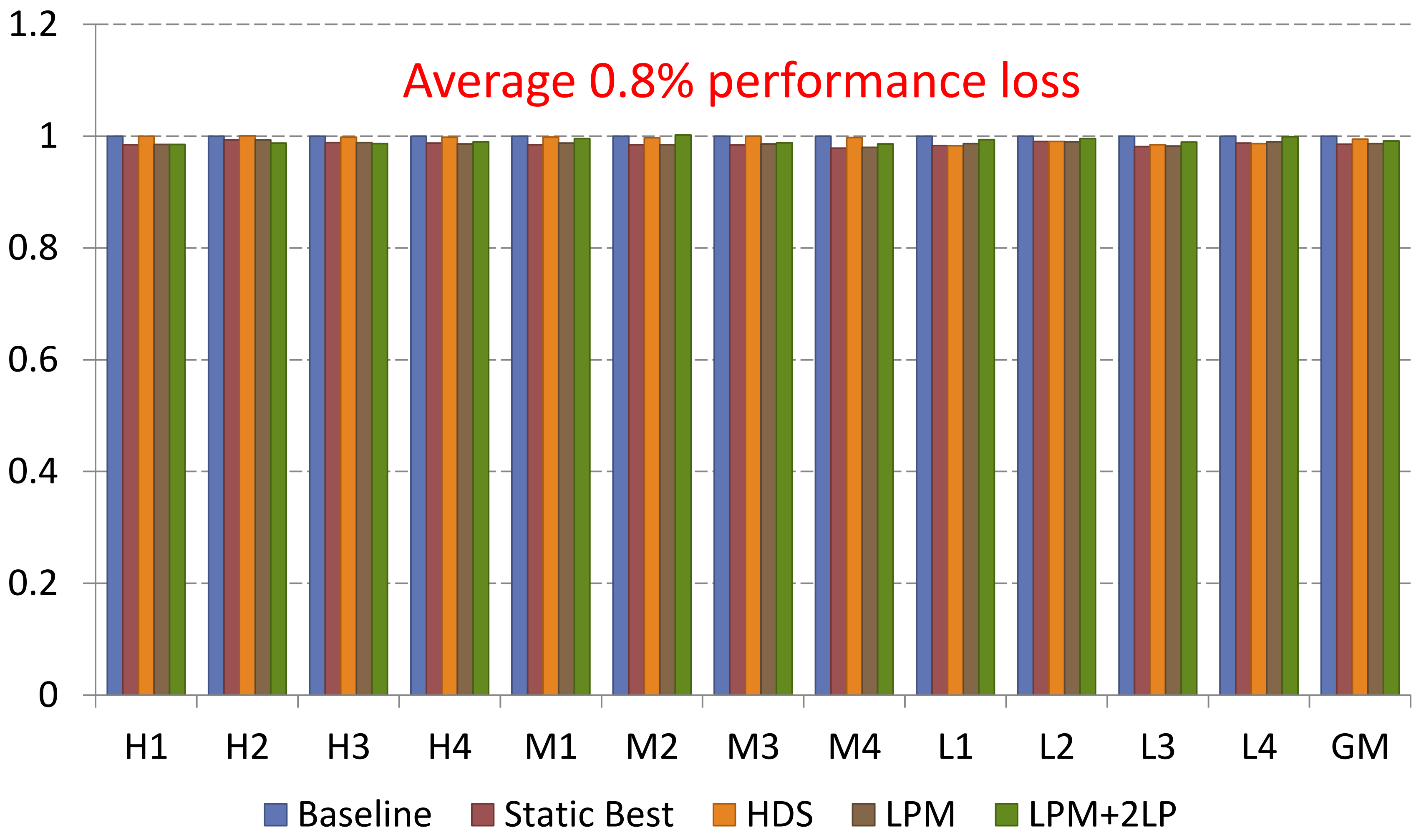
HMC Energy Consumption



Link Energy Consumption



Speedup


Average 0.8% performance loss

Summary

- Hybrid Memory Cube (HMC)
  - Higher bandwidth through power hungry serial links
- Dynamic power management of off-chip links
  - Trade-off between performance and link energy consumption
    - Link delay monitor simulating each possible link configuration
  - Two-level prefetching
    - Conservative prefetcher on CPU chip: prefetch traffic reduction
    - Aggressive prefetch buffer on HMC: LLC miss penalty reduction
- Evaluation
  - 51% reduction in HMC energy with 0.8% performance degradation