High-Level Synthesis of Multithreaded Accelerators for Irregular Applications

Stefano Devecchi, Nicola Saporetti, Marco Minutoli, Vito Giovanni Castellana, Marco Lattuada, Pietro Fezzardi, Antonino Tumeo and Fabrizio Ferrandi

Pacific Northwest
NATIONAL LABORATORY

Summary

Irregular Applications exhibit:

- Unpredictable, fine-grained data accesses
- Pointer or linked list-based data structures (e.g., graphs, unbalanced trees, unstructured grids, sparse matrices), difficult to partition in a balanced way
- Task-level parallelism
- High synchronization intensity
- Memory-bound (exploiting available bandwidth is non-trivial due to high memory reference rates)

Conventional High-Level Synthesis flows address:

- Dense, regular data structures
- Simple memory models
- Instruction-level parallelism
- Compute-bound kernels (Digital Signal Processing-like)
- OpenCL works well for regular, compute-bound workloads

Our contributions:

- Parallel distributed Controller (PC) for complex loops nests
- Hierarchical Memory Interface (HMI), supporting multi-banked/multi-ported memory and atomic memory operations
- Dynamic Task Scheduler (DTS) for unbalanced loop iterations
- **NEW: Support of temporal multithreading (and context switching) on the automatically generated, custom parallel accelerator array**

Previous Architectural Templates

We have developed a set of solutions to address HLS for irregular Applications:

Parallel distributed Controller (PC) allows controlling pools of parallel accelerators (i.e., "hardware tasks") with token passing mechanisms

Hierarchical Memory Interface (HMI) allows supporting multi-ported shared memory with dynamic address resolution and atomic memory operations

Dynamic Task Scheduler (DTS) allows launching a task as soon as an accelerator (kernel) in a pool is available (PC alone supports only block-based fork-joins)

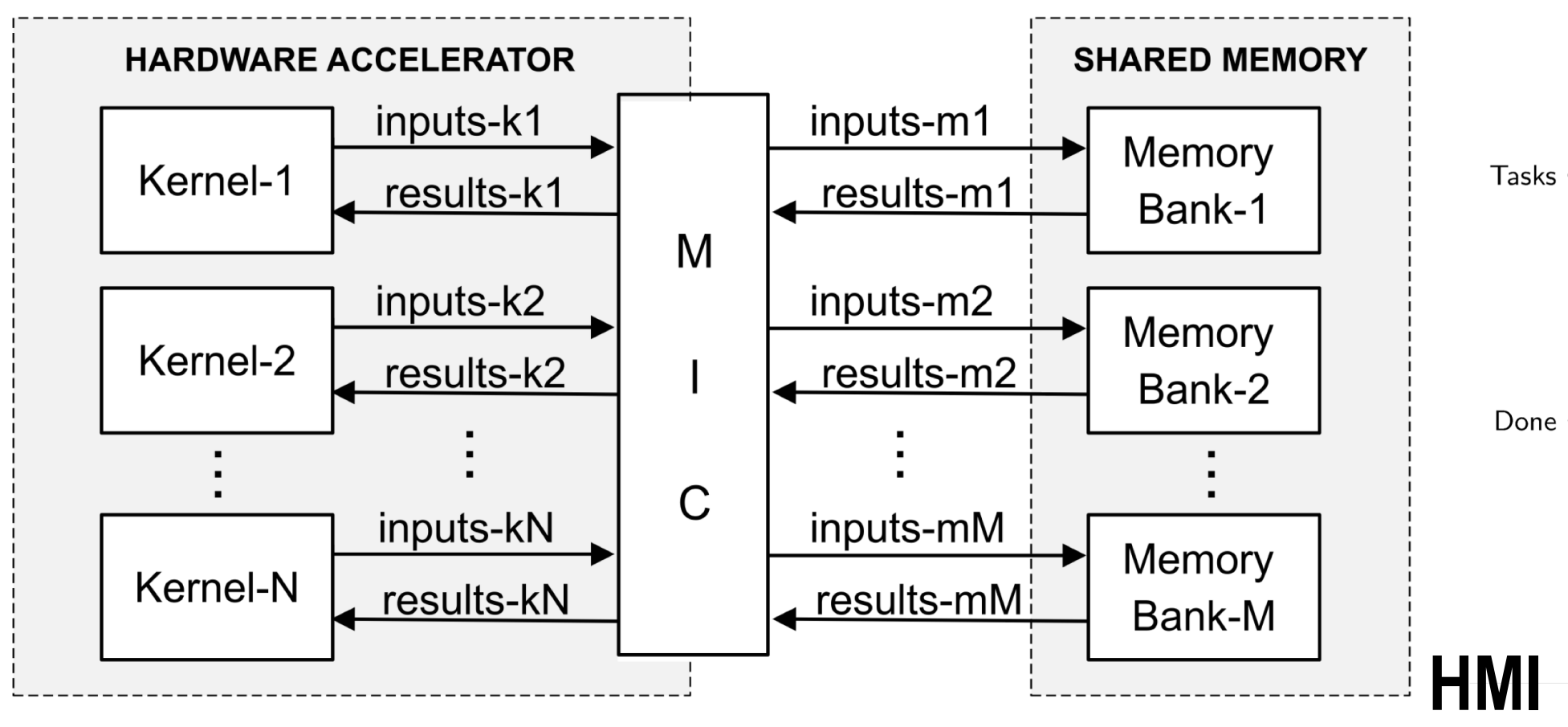
References:

V. G. Castellana, A. Tumeo, F. Ferrandi: An Adaptive Memory Interface Controller for Improving Bandwidth Utilization of Hybrid and Reconfigurable Systems. 2014.

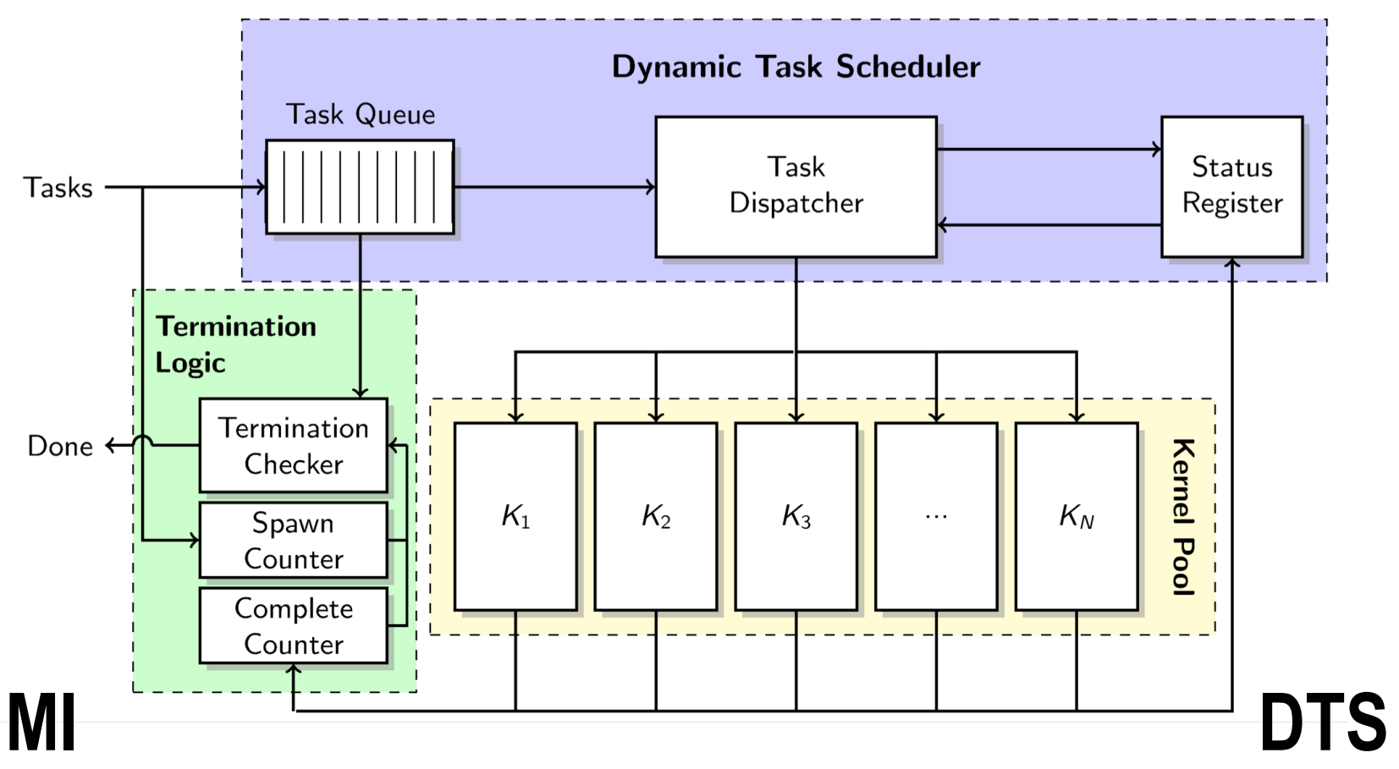
V. G. Castellana, M. Minutoli, A. Morari, A. Tumeo, M. Lattuada, F. Ferrandi: High-Level Synthesis of RDF Queries for Graph Analytics. ICCAD 2015.

M. Minutoli, V. G. Castellana, A. Tumeo: High-Level Synthesis of SPARQL Queries. SC15 poster.

M. Minutoli, V. G. Castellana, A. Tumeo, M. Lattuada, F. Ferrandi: Efficient Synthesis of Graph Methods: A Dynamically Scheduled Architecture. ICCAD 2016.



HMI

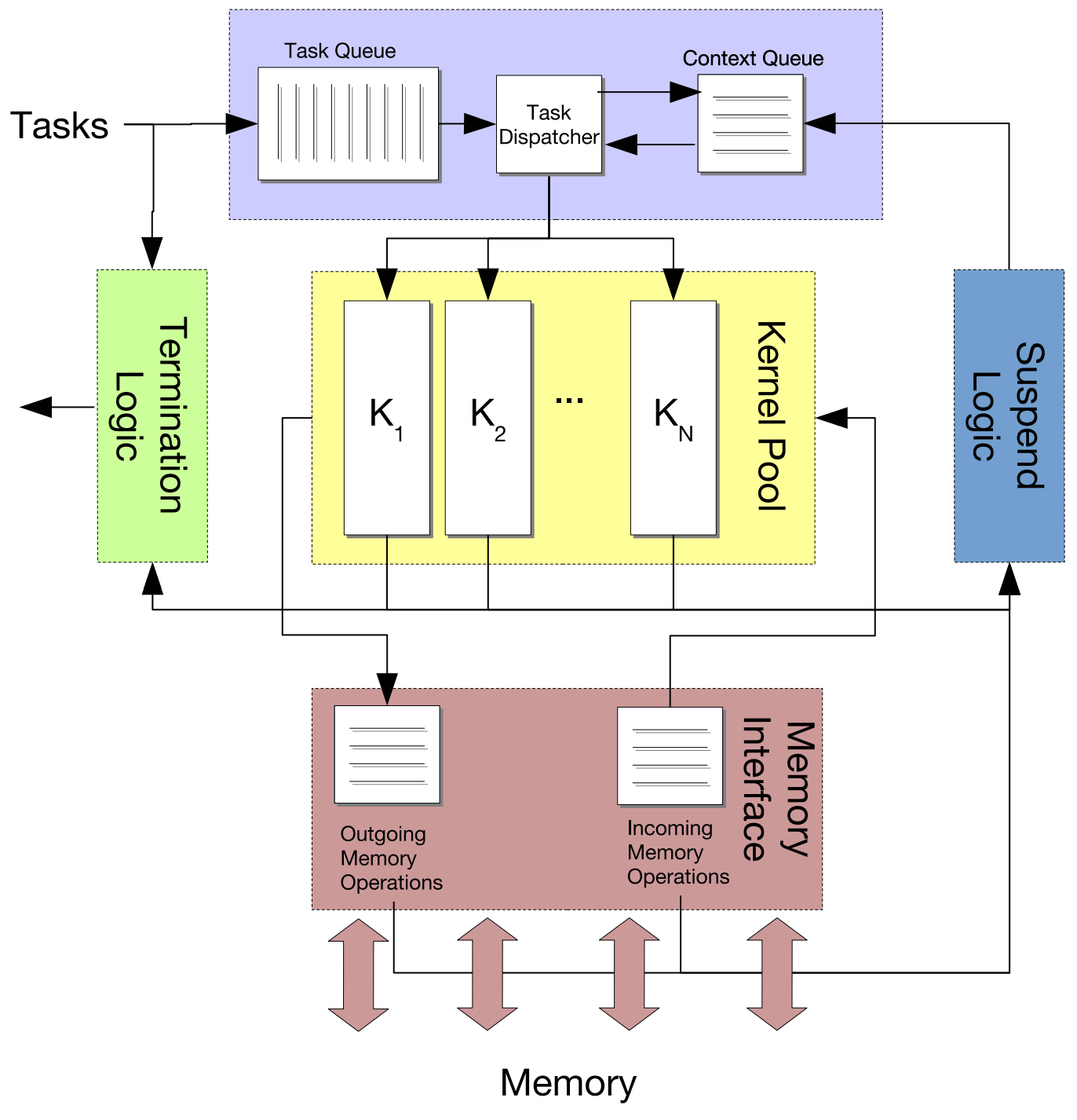


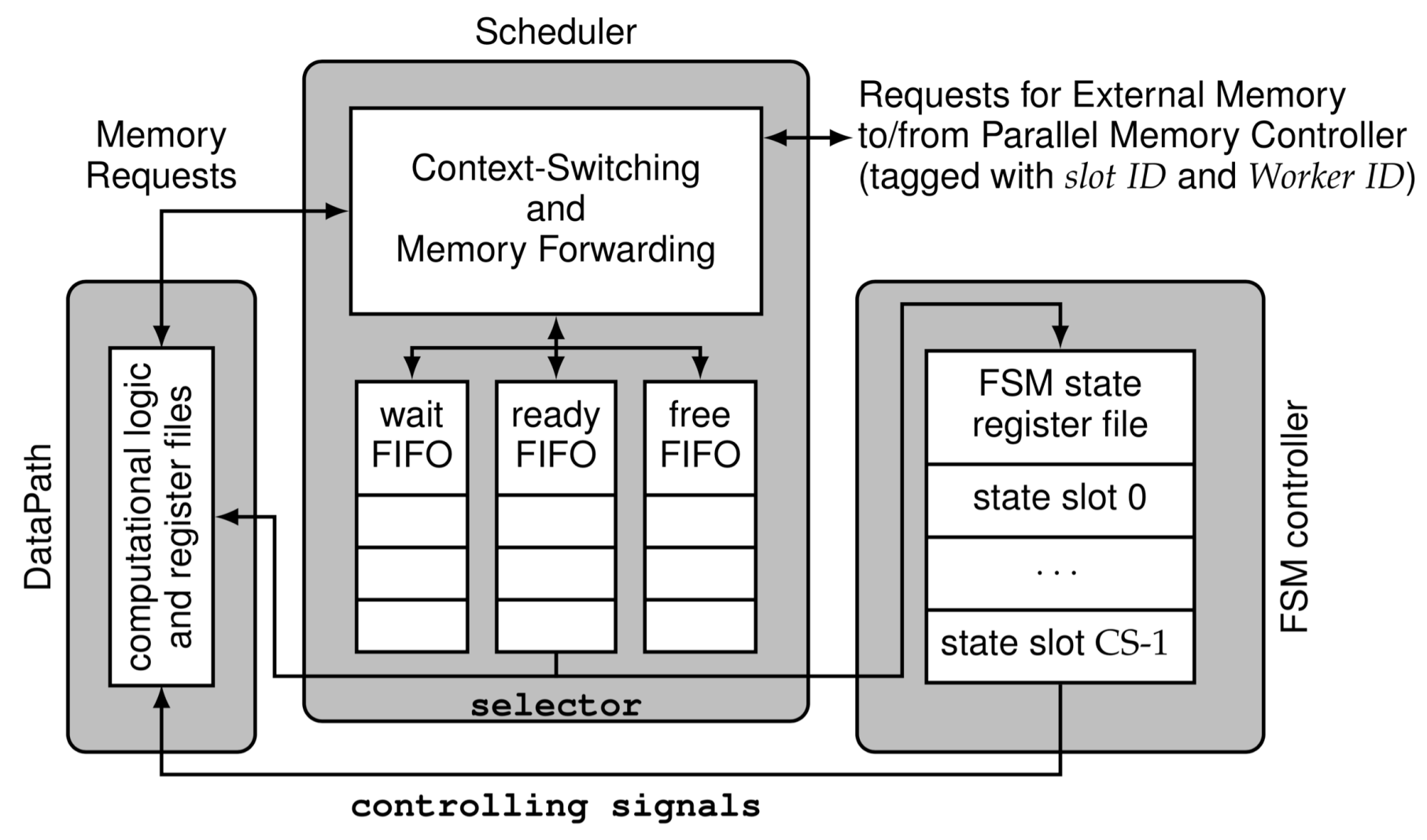
DTS

Temporally Multithreaded Architecture Template and High-Level Synthesis Flow

4:00 PM – Opening Remarks, Pitch Party format
& raffle details

- New architecture template significantly extends the DTS design
- Datapaths replicate registers depending on number of contexts
- Finite State Machines (FSMs) replicate state registers depending on the number of contexts
- Accelerator pool further decoupled from the memory interface with queues for pending memory operations. Memory operations include **atomics**.
- A task is suspended when it emits a **memory operation** (unknown latency)
- Task status stored in Context Queues
- A suspended task becomes ready again when memory operation completes
- Tolerates memory latency while computation continues
- Whole template integrated in an **open-source High-Level Synthesis tool**
- Tool synthesizes starting from **C sources annotated with OpenMP** and explores parameters, such as the number of contexts, accelerators, and memory channels




Scheduler
Memory Requests
Context-Switching and Memory Forwarding
Requests for External Memory to/from Parallel Memory Controller (tagged with slot ID and Worker ID)
DataPath
computational logic and register files
wait FIFO
ready FIFO
free FIFO
selector
FSM state register file
state slot 0
...
state slot CS-1
FSM controller
controlling signals

https://panda.dei.polimi.it

Experimental Evaluation

Synthesis of graph walks (graph pattern matching routines) to solve **Queries 1-7** of the Lehigh University Benchmark (**LUBM**) for the semantic web. Code is a set of nested loops that perform graph walks, look up labels, and count results with atomic memory operations in C. Latest version of Xilinx Vivado, Virtex-7 xc7vx690t.

Plots (only report query Q2), single context, and 32 contexts per accelerator:

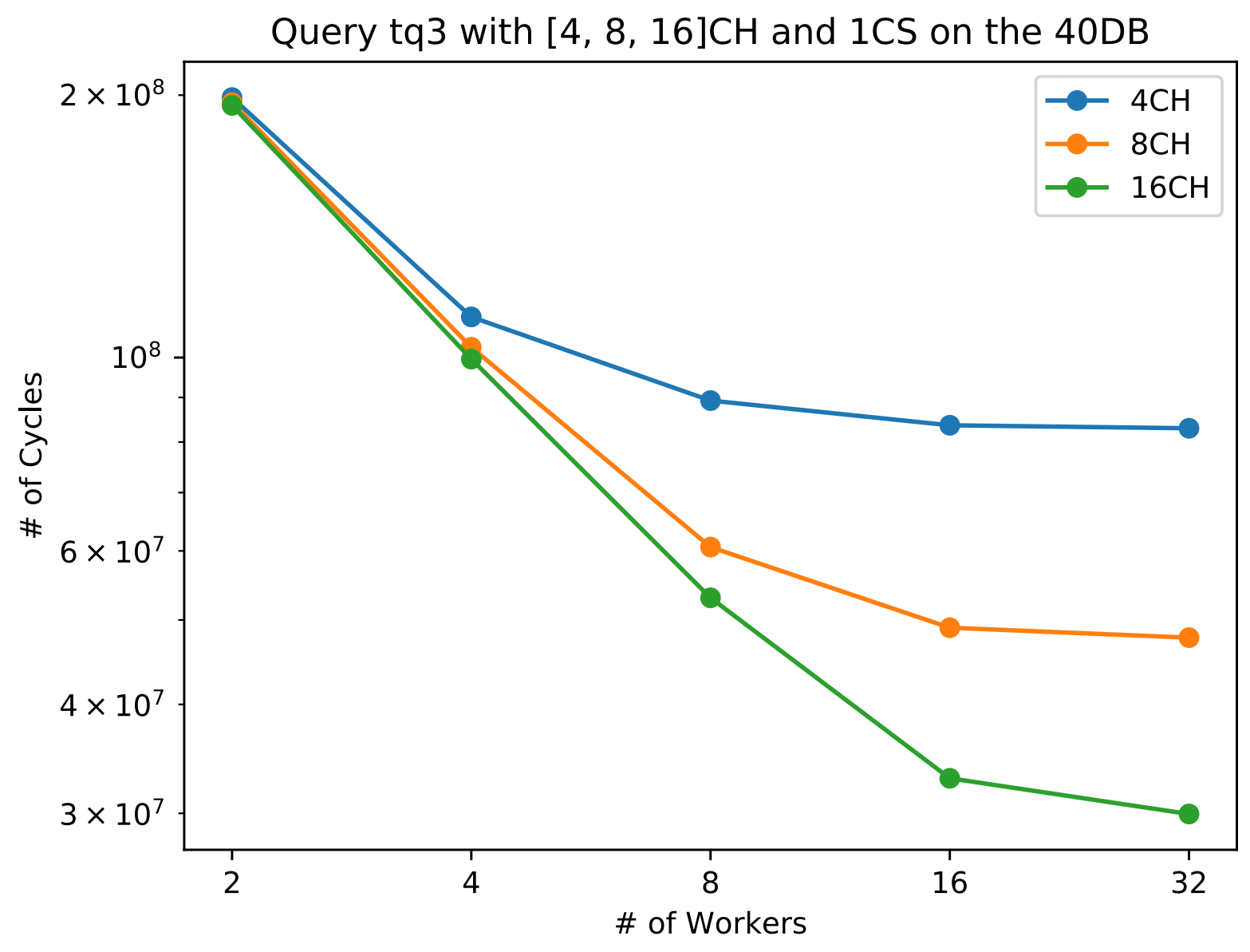
- 2-32 accelerators (workers), 4-16 memory channels (CH)
- Demonstrate problem is memory bound
- Two accelerators and 32 contexts already maximize performance (memory throughput)
- Area for 2 and 32 workers when varying number of contexts (overheads due to muxes)

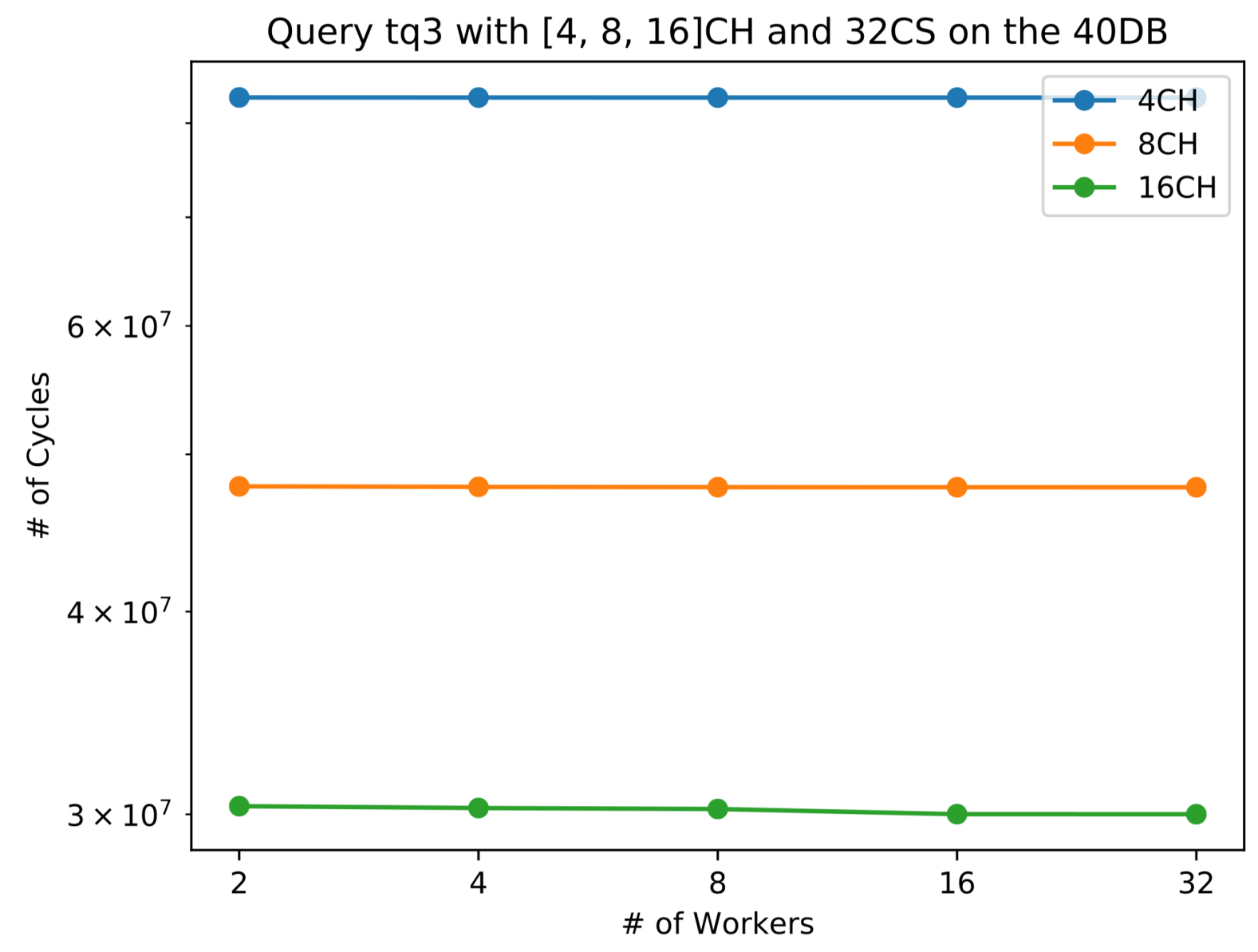
Tables, all queries:

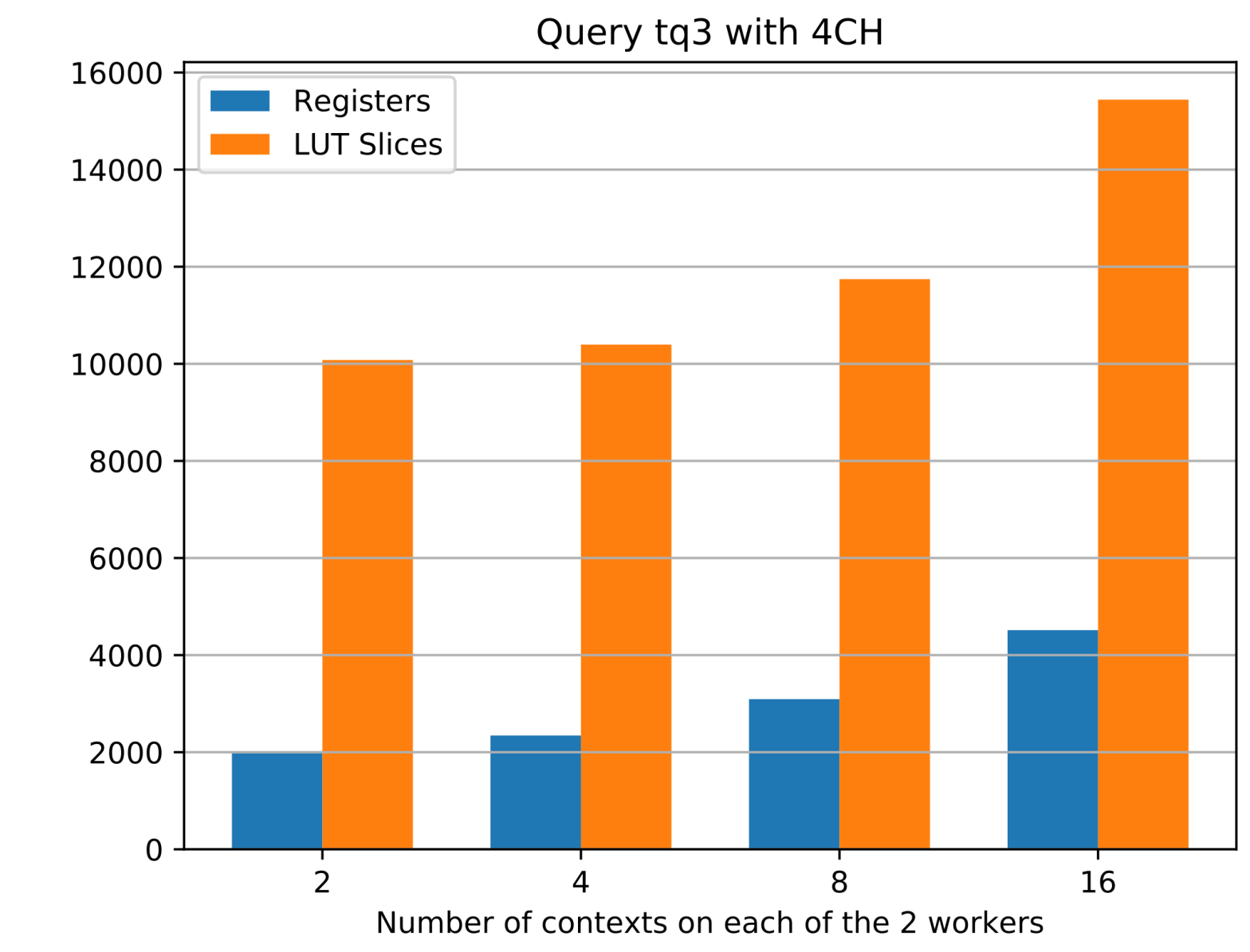
- Parallel Controller (PC) only, Dynamic Scheduler (DT), and multithreading (Svelto)
- Four accelerators (PC and DT) vs. one accelerator with eight contexts (four memory channels)

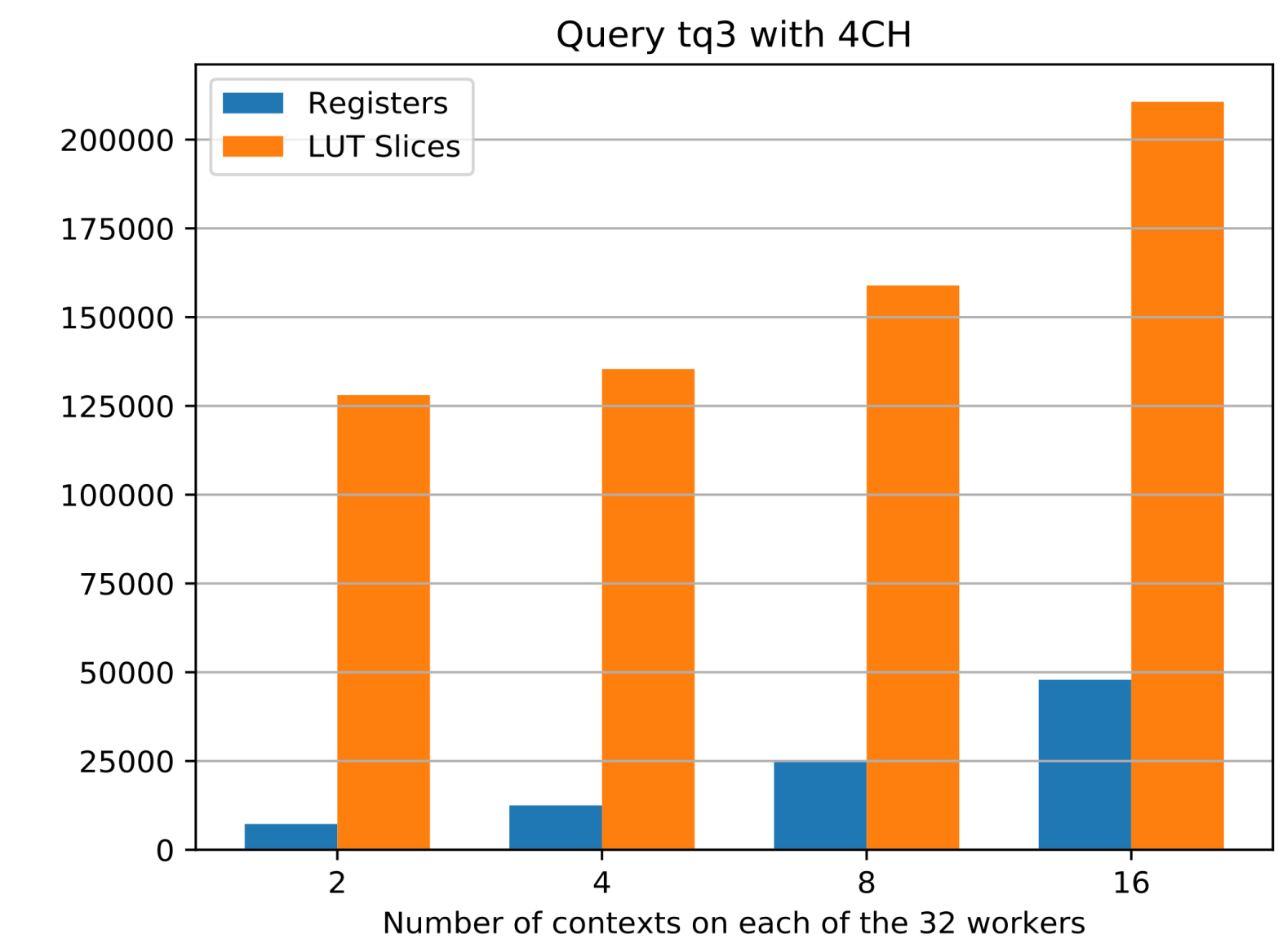
	Parallel Controller		Dynamic Scheduler		Svelto 01W-04CH-08CS	
	LUTs	Slices	LUTs	Slices	LUTs	Slices
Q1	13,469	4,317	10,844	3,503	6,923	2,246
Q2	5,280	1,607	4,636	1,335	4,293	1,408
Q3	13,449	4,308	10,664	3,467	6,841	2,242
Q4	7,806	2,399	6,175	1,918	5,222	1,696
Q5	5,750	1,738	5,330	1,578	4,396	1,424
Q6	10,600	3,426	8,125	2,633	5,811	1,868
Q7	15,002	4,953	11,344	3,747	7,094	2,340

	Parallel Controller # Cycles	Dynamic Scheduler # Cycles	Svelto 01W-04CH-08CS # Cycles
Q1	1,001,581,548	287,527,463	269,153,871
Q2	2,801,694	2,672,295	3,470,665
Q3	98,163,298	95,154,310	84,268,006
Q4	42,279	19,890	18,584
Q5	13,400	8,992	8,514
Q6	629,671	199,749	171,290
Q7	35,511,299	24,430,557	21,500,466









U.S. DEPARTMENT OF
ENERGY

Marco Minutoli, Vito Giovanni Castellana, Antonino Tumeo

Pacific Northwest National Laboratory
P.O. Box 999, MS-IN: J4-30, Richland, WA 99352 USA
{marco.minutoli, vitogiovanni.castellana,
antonino.tumeo}@pnnl.gov

Stefano Devecchi, Nicola Saporetti, Pietro Fezzardi, Marco Lattuada, Fabrizio Ferrandi

DEIB – Politecnico di Milano
P.za Leonardo Da Vinci 32, 20132 Milano Italy
{stefano.devecchi, nicola.saporetti}@mail.polimi.it
{pietro.fezzardi, marco.lattuada, fabrizio.ferrandi}@polimi.it

www.pnnl.gov

PNNL-SA-136670