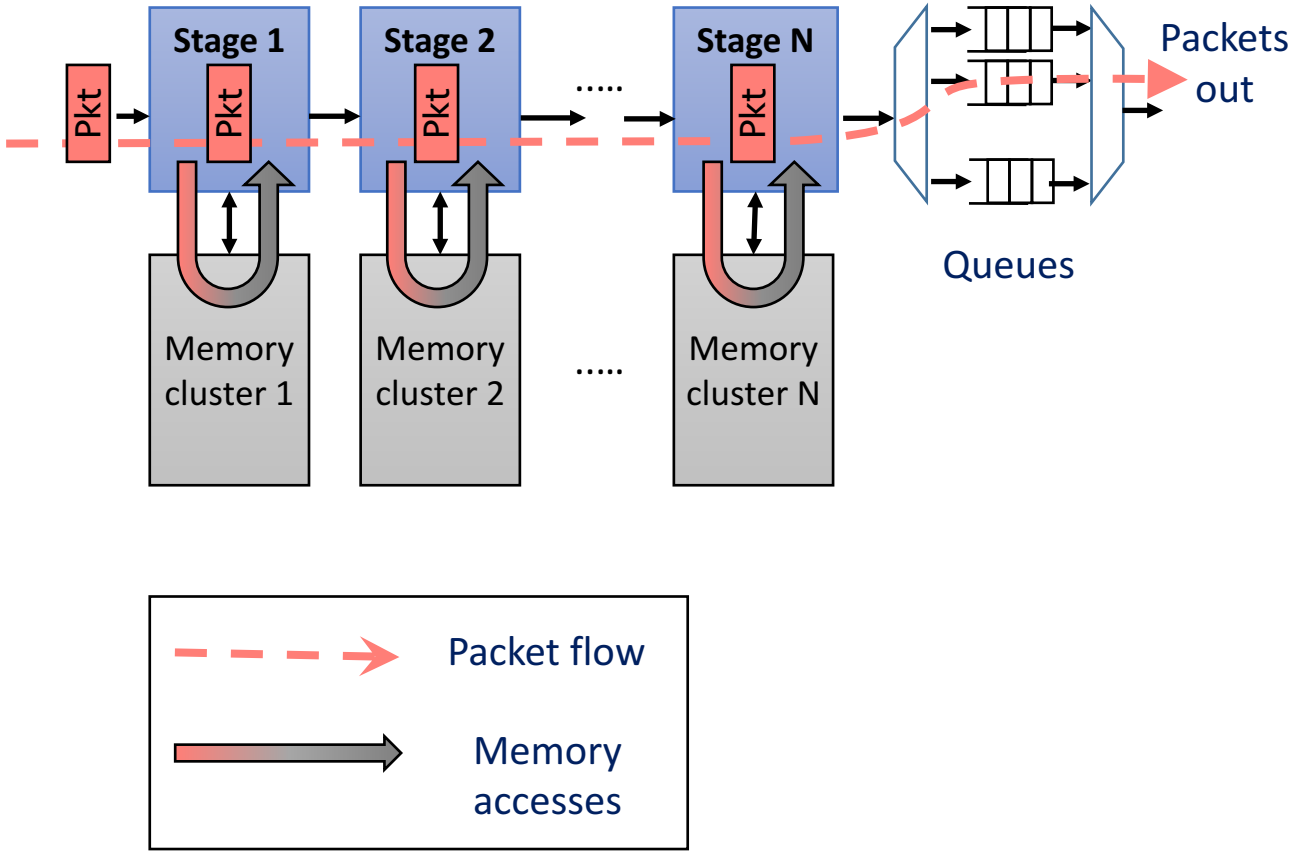
dRMT: Disaggregated Programmable Switching

Sharad Chole, Andrew Fingerhut, Sha Ma (Cisco); Anirudh Sivaraman (MIT); Shay Vargaftik , Alon Berger, Gal Mendelson (Technion); Mohammad Alizadeh (MIT); Shang-Tse Chuang (Cisco); Isaac Keslassy (Technion/VMware); Ariel Orda (Technion); Tom Edsall (Cisco);

RMT Architecture

- ”Forwarding metamorphosis” (SIGCOMM, 2013)
- Pipeline of match-action stages
  - Key Generation
  - Match Lookup
  - Action
- Challenges
  - Fixed order of execution
  - Coupled memory and resources
  - Performance Cliff

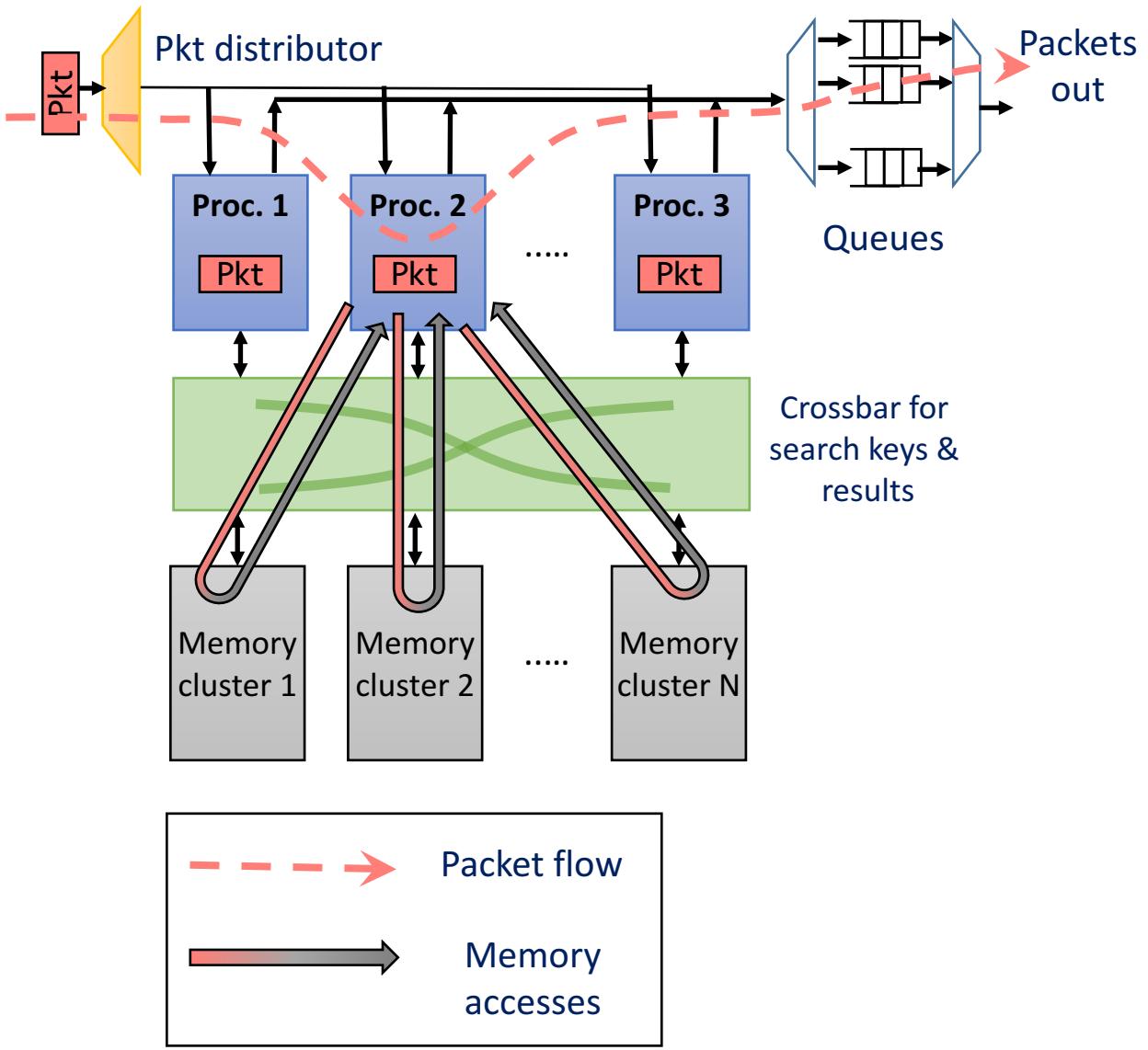


dRMT Architecture

- Disaggregated Resources
  - Memory Clusters
  - Match-action processors
- Benefits
  - Flexibility
  - Higher Resource Utilization
  - Graceful performance degradation

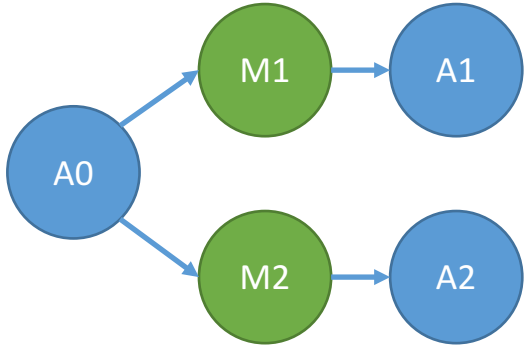
In this talk,

- Deterministic Scheduling
- Hardware Design
- RMT vs dRMT

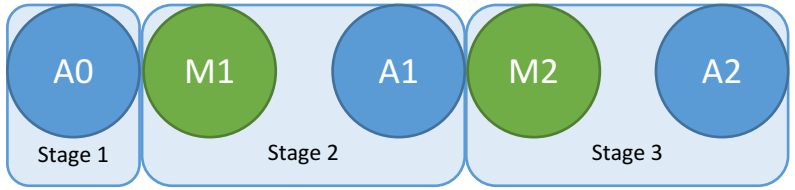


Case for Disaggregation

Memory Disaggregation
e.g. M1 and M2 cannot fit into single match stage



a) Dependency Graph

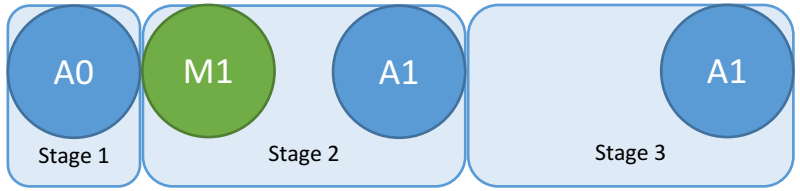


b) RMT Pipeline Schedule

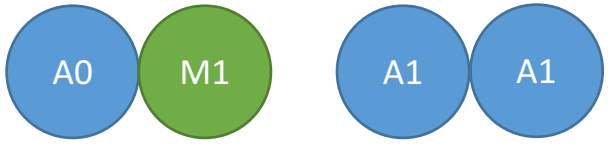
c) dRMT Schedule

Compute Disaggregation
e.g. A1 cannot fit into single action stage

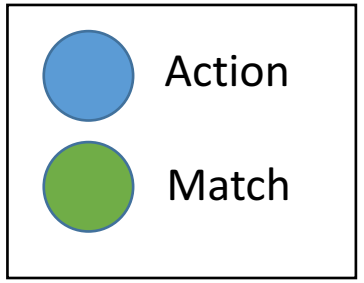
a) Dependency Graph



b) RMT Pipeline Schedule



c) dRMT Schedule



Scheduling

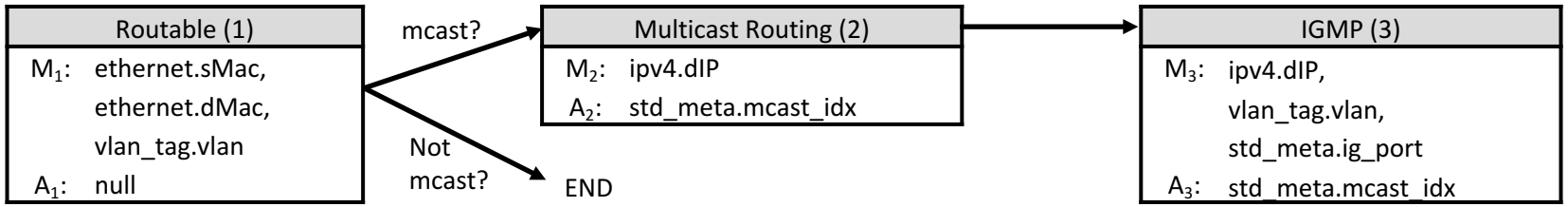
Objective:

- Finding a Fixed Schedule
  - Round-robin across processors
  - Compile time
- Minimize processing Latency
- Maintain required throughput

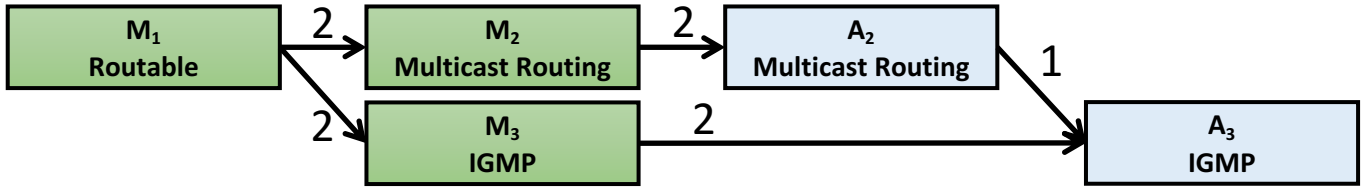
Constraints:

- Program Dependencies
  - Operational Dependency
  - Minimum Latency separation
    - ΔM – Match latency
    - ΔA – Action latency
- Architectural constraint
  - Processors/Threads
  - Key Generation
  - Action ALUs
- **IPC** (Inter Packet Concurrency)
  - Parallel Packets being processed

Fixed Schedule: Example



a) Control Flow Program



Where,
ΔM = 2
ΔA = 1

b) Operation Dependency Graph

Cycle	0	1	2	3	4	5	6	7	8	9
Proc 1	M1		no-op	M2,M3		A2	A3			
Proc 2		M1		no-op	M2,M3		A2	A3		
Proc 1			M1		no-op	M2,M3		A2	A3	
Proc 2				M1		no-op	M2,M3		A2	A3

c) Fixed schedule

Scheduling: ILP Formulation

- Reduced Scheduling
  - Constraint modulo P
- NP-Hard

ILP Formulation:

- Model resource constraints
- Model program dependencies
- Model scheduling constraints

Heuristics:

- Topological Random Sieve
  - Sieve of Eratosthenes
  - Topologically sort the nodes
  - Greedy equivalence classes
  - Randomize sorting
- Sieve Rotator
  - Step 1: ILP on RMT constraints
  - Step 2: Topological Sieve

Performance Simulation

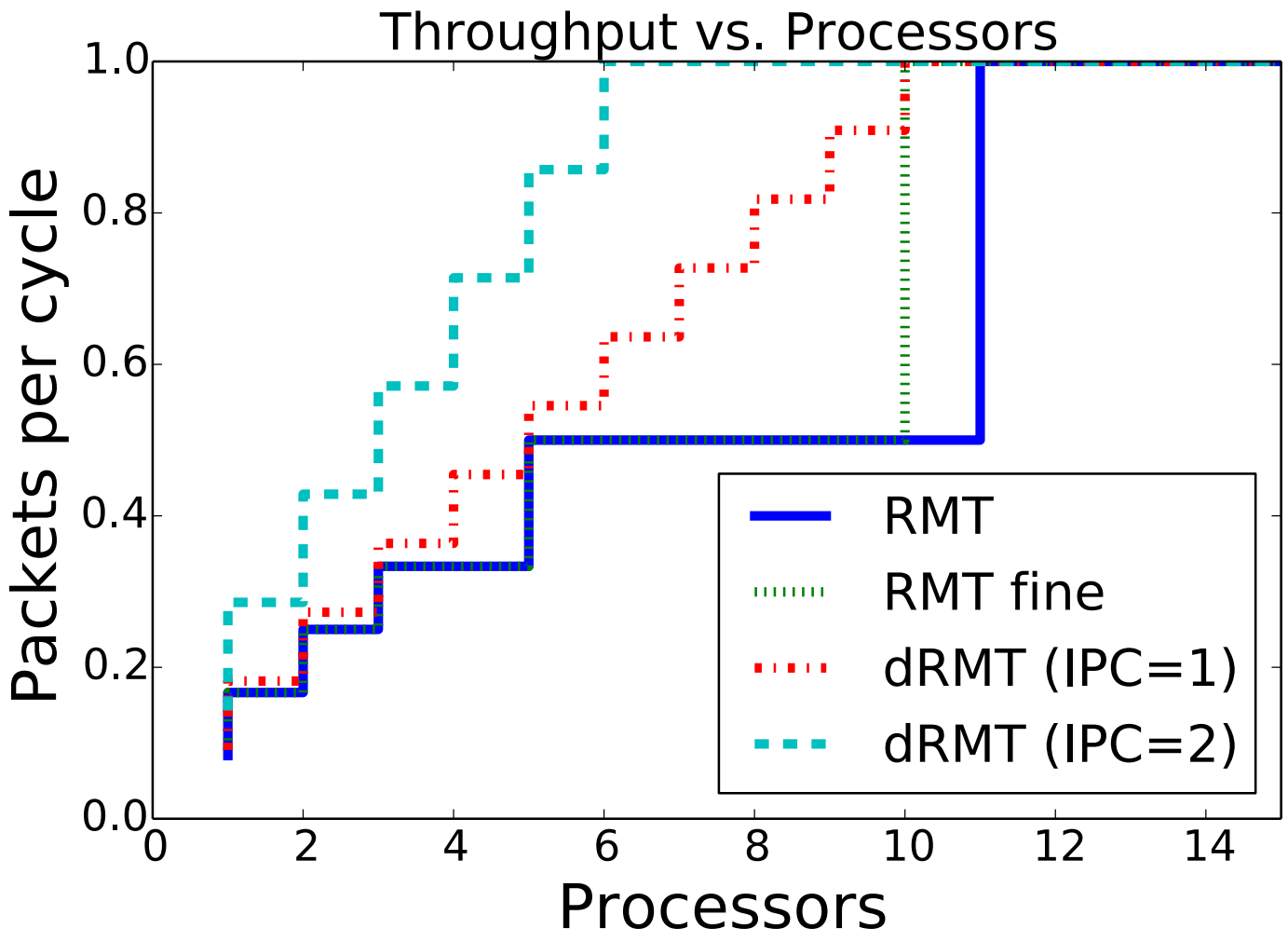
P4 Program	RMT	dRMT (IPC=1)	dRMT(IPC=2)	Lower bound
Ingress (switch.p4)	18	17	15	15
Egress (switch.p4)	12	11	7	7
Switch.p4 combined	22	21	21	21
Anonymized	2x	2x	1x	1x

a) Minimum number of processors to achieve line rate

P4 Program	RMT	dRMT (IPC=1)	dRMT(IPC=2)	Lower bound
Ingress (switch.p4)	360	245	243	243
Egress (switch.p4)	240	217	198	197
Switch.p4 combined	440	243	243	243
Anonymized	2.8x	1.1x	1x	1x

a) Minimum number of threads to achieve line rate

Performance Simulation



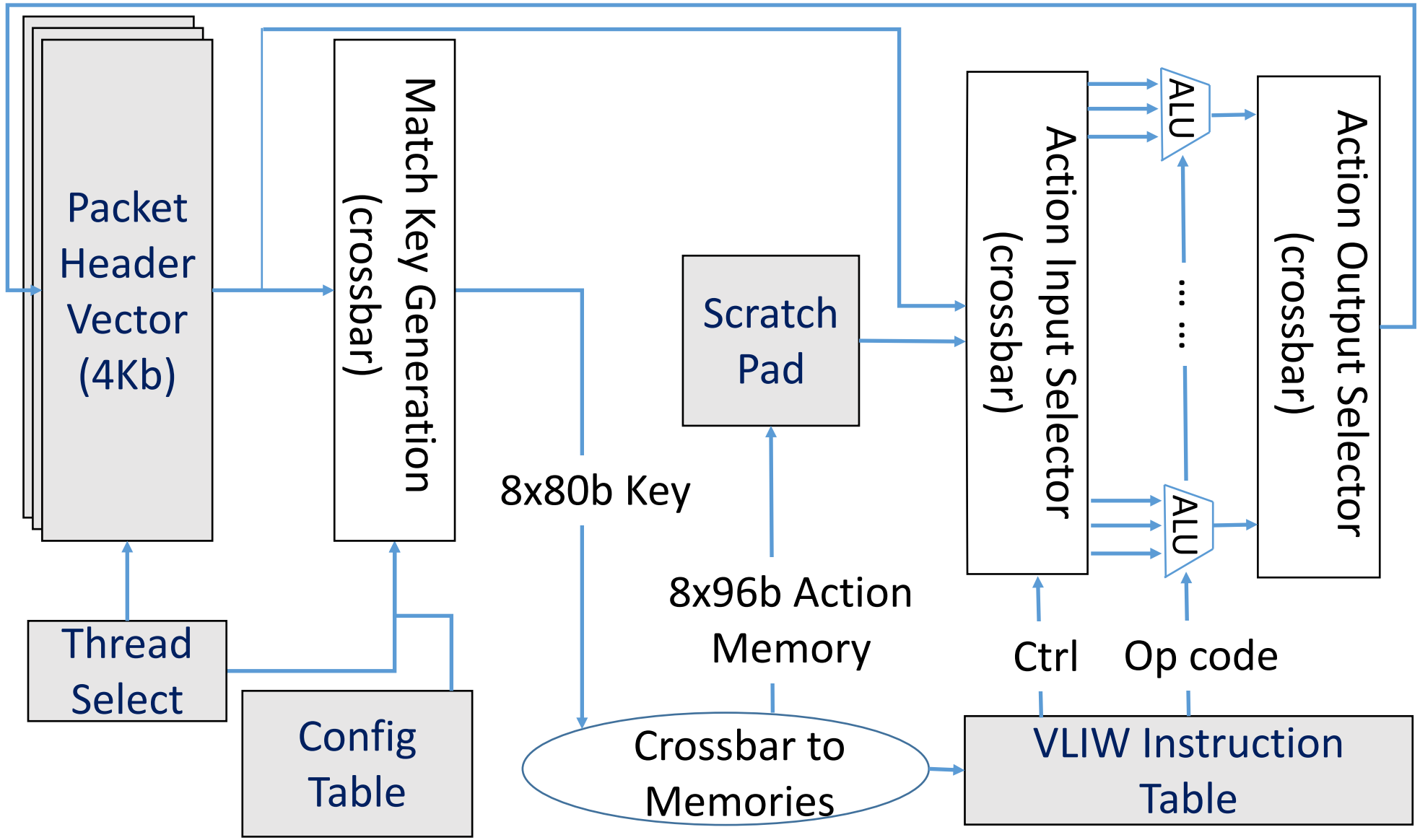
Throughput on switch.p4 egress pipeline

Hardware Implementation

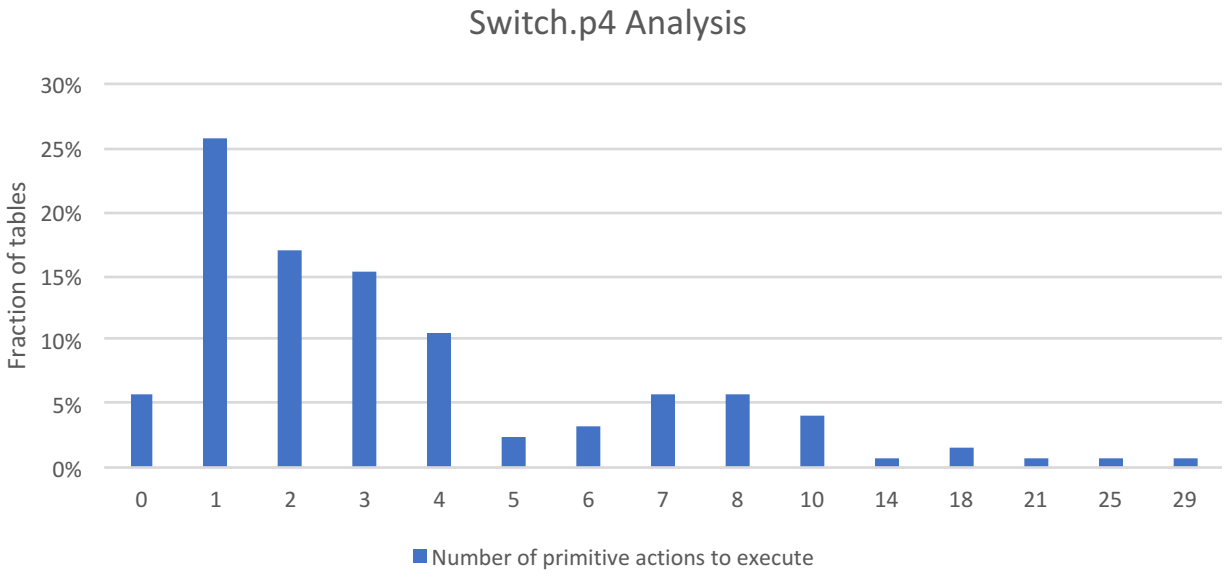
Key Differences:

- Match Action Processor
  - Packet Contexts
  - Scratch pad
- VLIW Instructions
- Crossbar

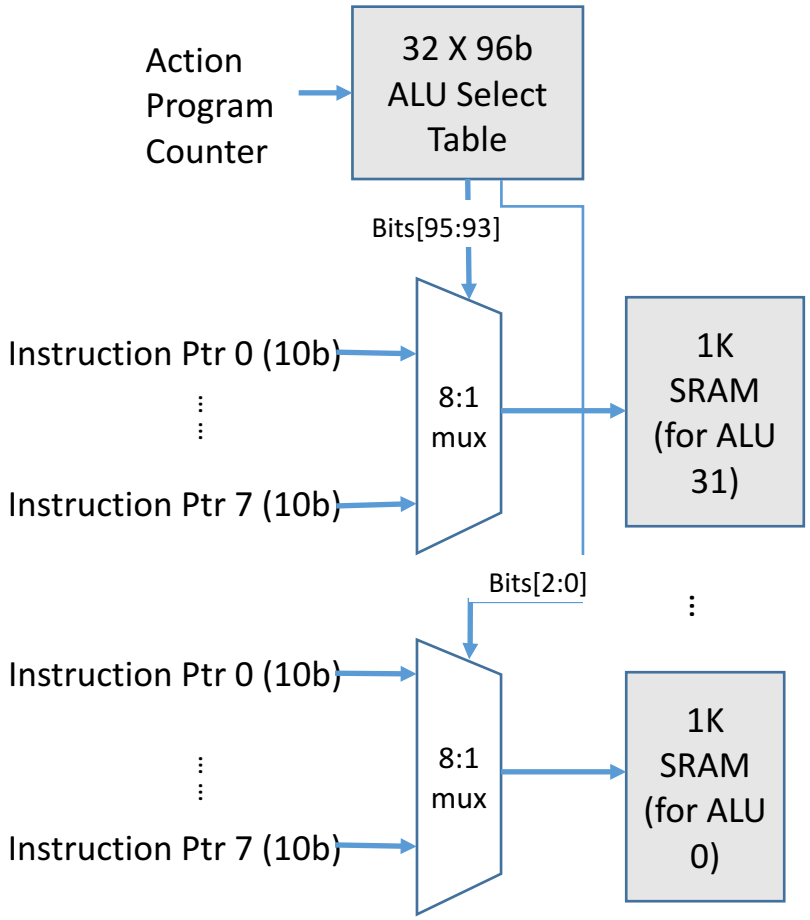
Match-Action Processor



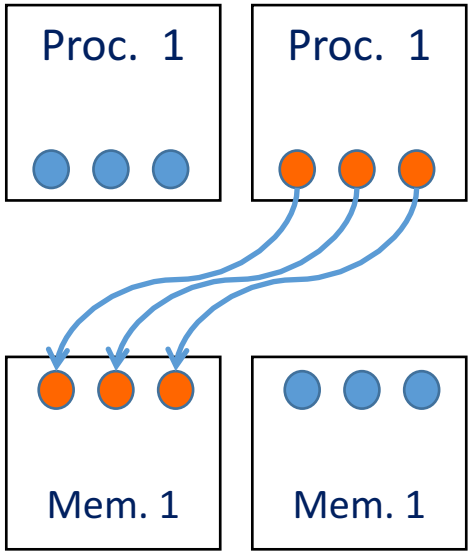
VLIW Instructions



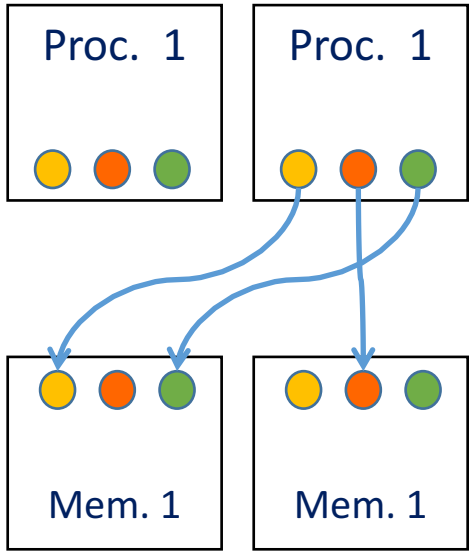
- Over 90% of tables require 8 or fewer primitive actions.
- dRMT implements 32 of 32-bit ALUs
- Instruction memory is implemented as 32 of 1K deep SRAMs



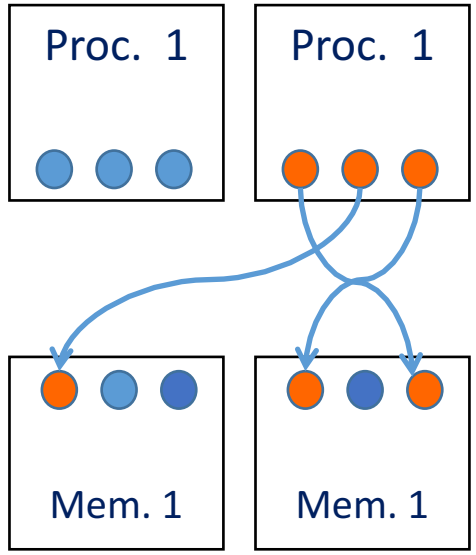
Crossbar choices



Unit Crossbar
(0.561 mm^2)



Segment Crossbar
(0.576 mm^2)



Full Crossbar
(4.464 mm^2)

- The segment crossbar is as powerful as full crossbar if
  1. No tables are split across clusters
  2. Total key-segments per cluster < Max Key Segments per cluster

dRMT uses One-to-Many Segment Crossbar.

Hardware Evaluation

Component	RMT	dRMT(IPC=1)	dRMT(IPC=2)
Key Generation			
Config Register	0.007	0.004	0.005
Key Crossbar	0.098	0.049	0.071
Packet Storage			
Header Vector	0.110	0.326	0.470
Scratch Pad	N/A	0.051	0.051
Actions			
Input Crossbar	0.486	0.171	0.315
ALUs	0.170	0.071	0.071
Output Crossbar	N/A	0.048	0.048
VLIW Table	0.372	0.336	0.336
Total	**1.243**	**1.056**	**1.367**

Estimated area per processor (mm^2)

Hardware Evaluation

Number of Processors	RMT	Crossbar with 32 memory clusters	dRMT(IPC=1) with crossbar	dRMT(IPC=2) with crossbar
16	19.9	0.857	17.7	22.7
24	29.9	1.254	26.6	34.1
32	39.8	1.740	35.5	45.5

a) Area of all processors plus interconnect (mm^2)

Number of Processors	Power for Crossbar with 32 memory clusters
16	0.88 W
24	1.31 W
32	1.75 W

b) Crossbar Power @1.2 GHz 0.9V

Summary

- Disaggregated flexible architecture for high speed programmable switching
- Higher resource utilization
  - Lower latency
  - Lower thread count
- Graceful degradation of throughput
- Independent scaling of processing/memory capacity

Applications:

- Low Latency Switching
- SmartNIC
- Off-chip buffers/non-deterministic memory latencies

Thank you!