Programmable Packet Scheduling at Line Rate

Anirudh Sivaraman, Suvinay Subramanian, Anurag Agrawal, Sharad Chole, Shang-Tse Chuang, Tom Edsall, Mohammad Alizadeh, Sachin Katti, Nick McKeown, Hari Balakrishnan
Programmable scheduling at line rate

• Programmable: Can we express a new scheduling algorithm?

• Line-rate: Highest capacity supported by a communication standard
Programmability at line-rate

- OpenFlow: Match-Action interface, fixed fields, fixed actions
- P4, RMT, FlexPipe, Xpliant: Protocol-independent match-action pipeline.
Why is programmable scheduling hard?

• Plenty of scheduling algorithms
• Yet, no consensus on the right abstractions for scheduling
• In contrast to
  • Parse graphs for parsing
  • Match-Action tables for forwarding
The Push-In First-Out Queue

- Many algorithms determine transmission order at packet arrival
- Relative order of packet transmissions of packets in the queue doesn’t change with future arrivals
- Examples:
  - SJF: Order determined by flow size
  - FCFS: Order determined by arrival time

- Push-in first-out queues (PIFO): packets are pushed into an arbitrary location based on a priority, and dequeued from the head
- First used as a proof construct by Chuang et. al
A programmable scheduler

Classification & Transmission
Order Computation

Ingress Pipeline
Classification & Transmission
Order Computation

Scheduler
Push-In-First-Out (PIFO) Queue
pFabric using PIFO

1. \( f = \text{flow}(p) \)
2. \( p.\text{prio} = f.\text{rem}\_\text{size} \)
Weighted Fair Queuing

1. \( f = \text{flow}(p) \)
2. \( p.\text{start} = T[f].\text{finish} \)
3. \( T[f].\text{finish} = p.\text{start} + p.\text{len} / p.w \)
4. \( p.\text{prio} = p.\text{start} \)

Push-In-First-Out (PIFO) Queue
Traffic Shaping

1. update tokens
2. $p$.send = now + \( \frac{(p\.len - tokens)}{rate} \);
3. $p$.prio = $p$.send

Scheduler

Push-In-First-Out (PIFO) Queue
Composing PIFOs

Hierarchical packet-fair queueing (HPFQ)

PIFO-root (WFQ on A and B)

PIFO-A (WFQ on 1 and 2)

PIFO-B (WFQ on 3 and 4)
The PIFO abstraction

• PIFO: A sorted array that let us insert an entry (packet or PIFO pointer) into a PIFO based on a programmable priority
• Entries are always dequeued from the head
• If an entry is a packet, dequeue and transmit it
• If an entry is a PIFO, dequeue it, and continue recursively
PIFO in hardware

- Meets timing at 1 GHz on a 16 nm node
- 5 % area overhead for 3-level hierarchy
- Challenges wisdom that sorting is hard
Closing thoughts

• Line-rate programmable scheduling is within reach

• Two concrete benefits
  • Program new scheduling algorithms
  • Design and verify a PIFO, not many scheduling algorithms