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

Massachusetts Institute of Technology

BAREFOOT NETWORKS

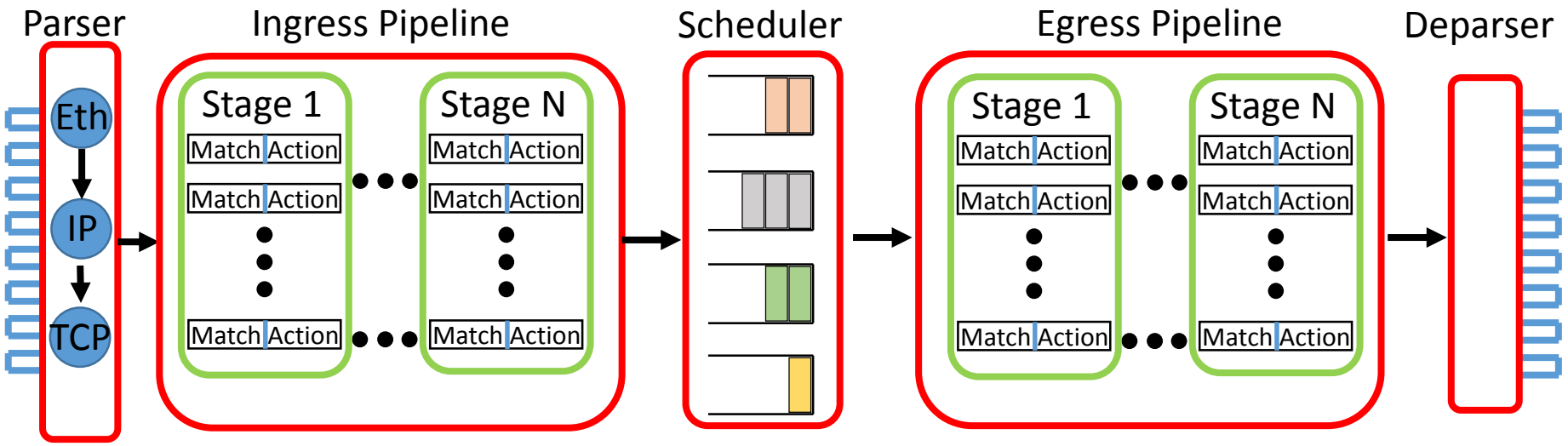
CISCO

Stanford University

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?

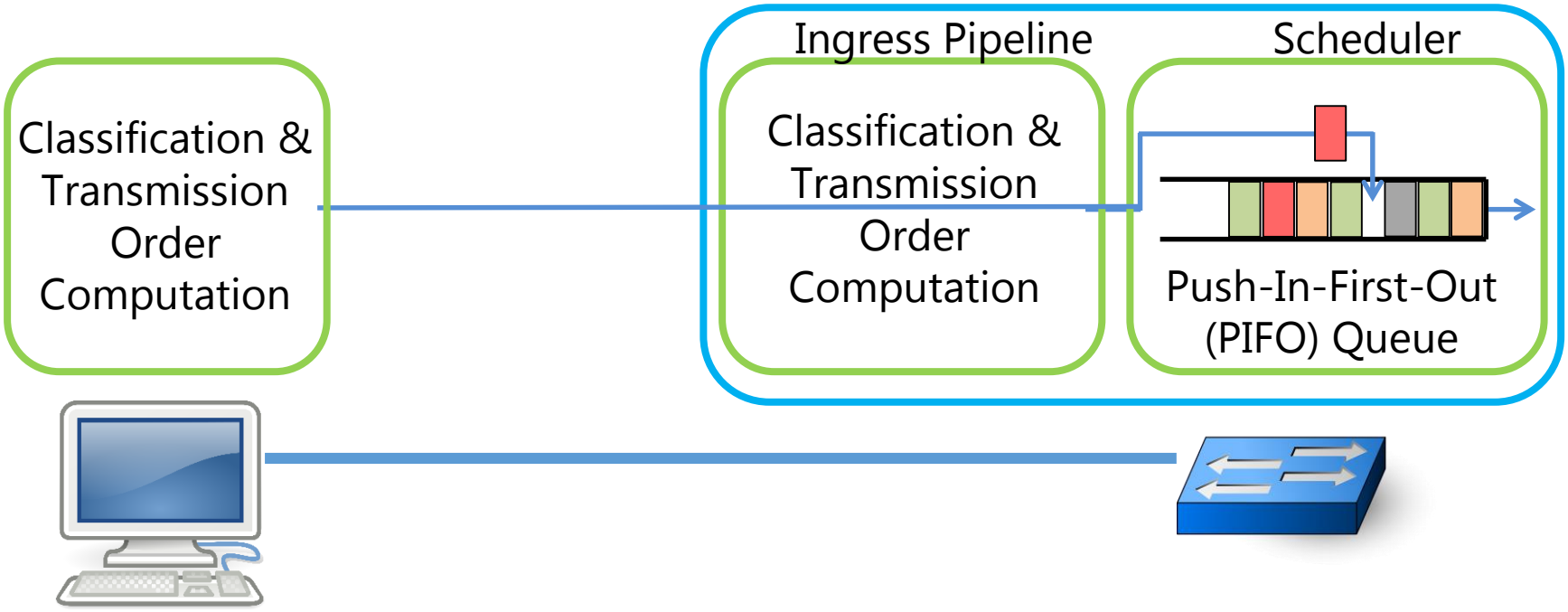
- Plenty of scheduling algorithms
- Yet, no consensus on the right abstractions for scheduling
- In contrast to
  - Parse graphs for parsing
  - Match-Action tables for forwarding

The Push-In First-Out Queue

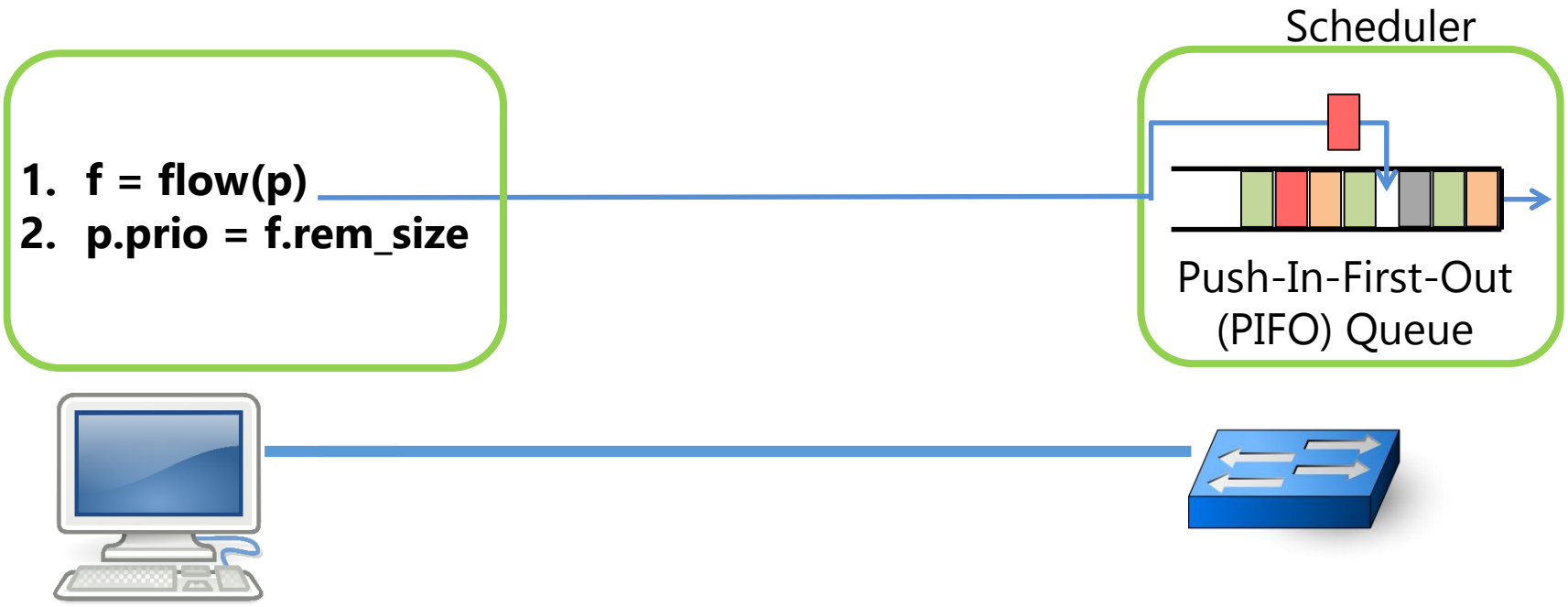
- Many algorithms determine transmission order at packet arrival
- Relative order of packet transmissions of packets in the queue doesn't change with future arrivals
- Examples:
  - SJF: Order determined by flow size
  - FCFS: Order determined by arrival time

- 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

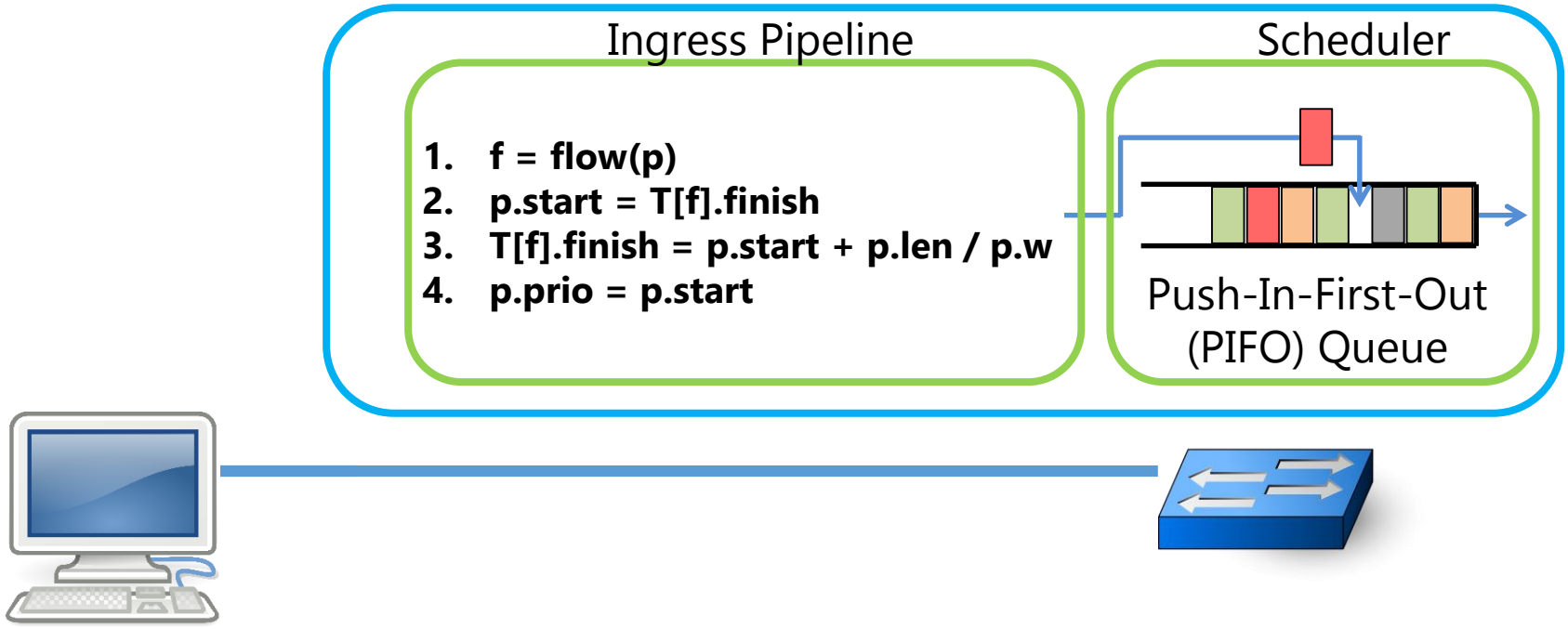
A programmable scheduler



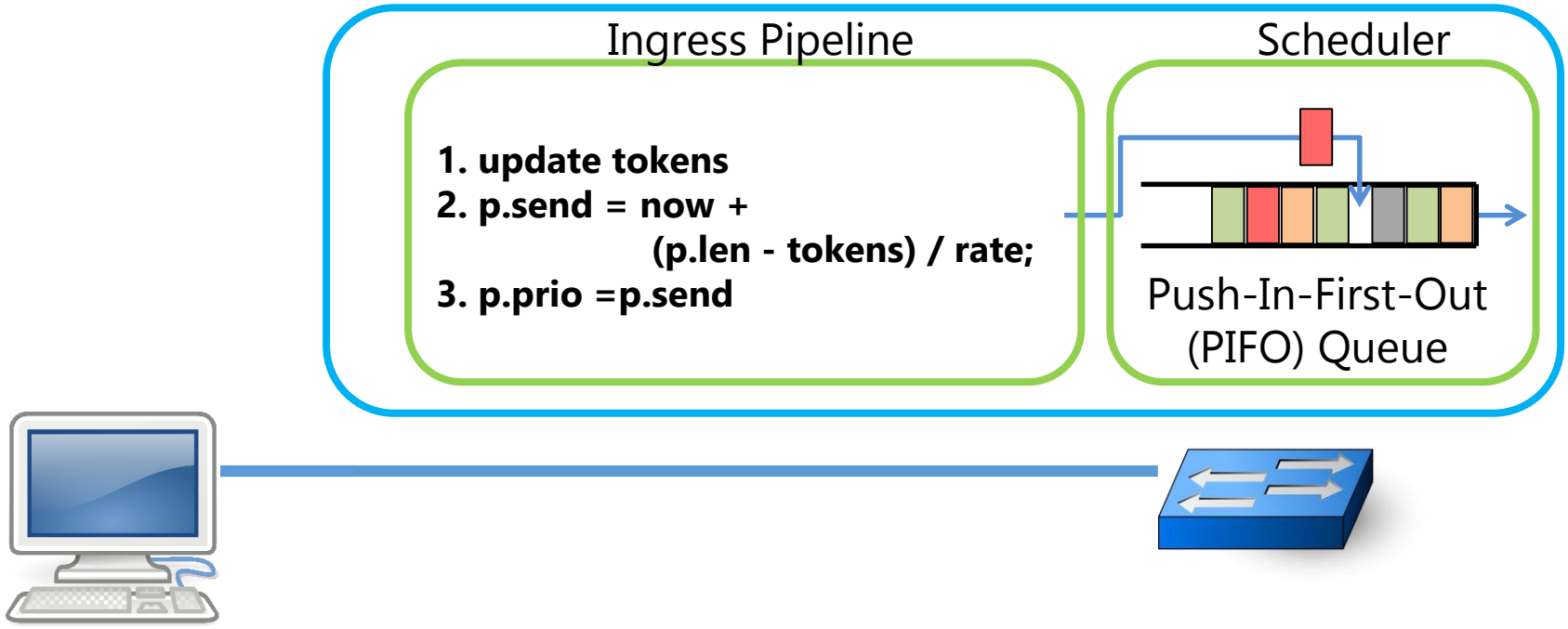
pFabric using PIFO



Weighted Fair Queuing

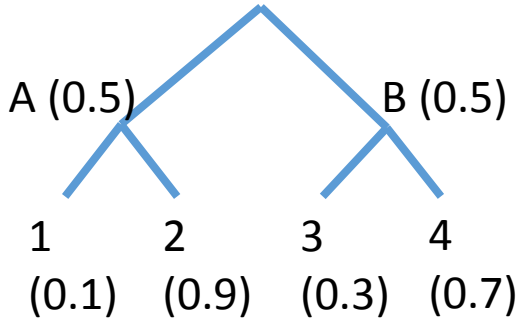


Traffic Shaping

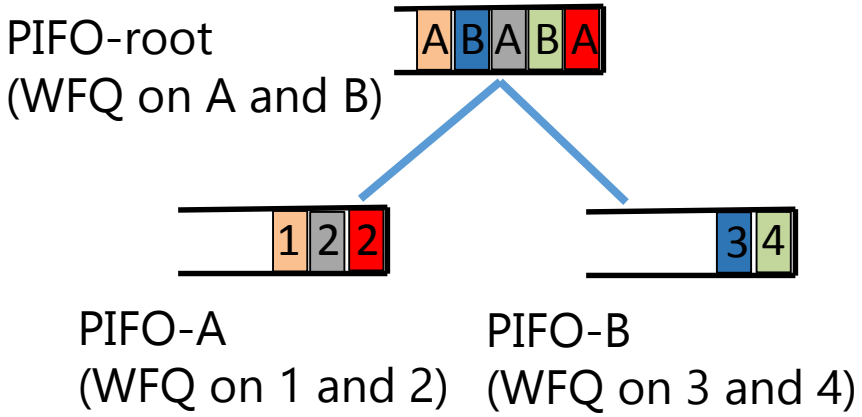


Composing PIFOs

Hierarchical packet-fair queueing (HPFQ)



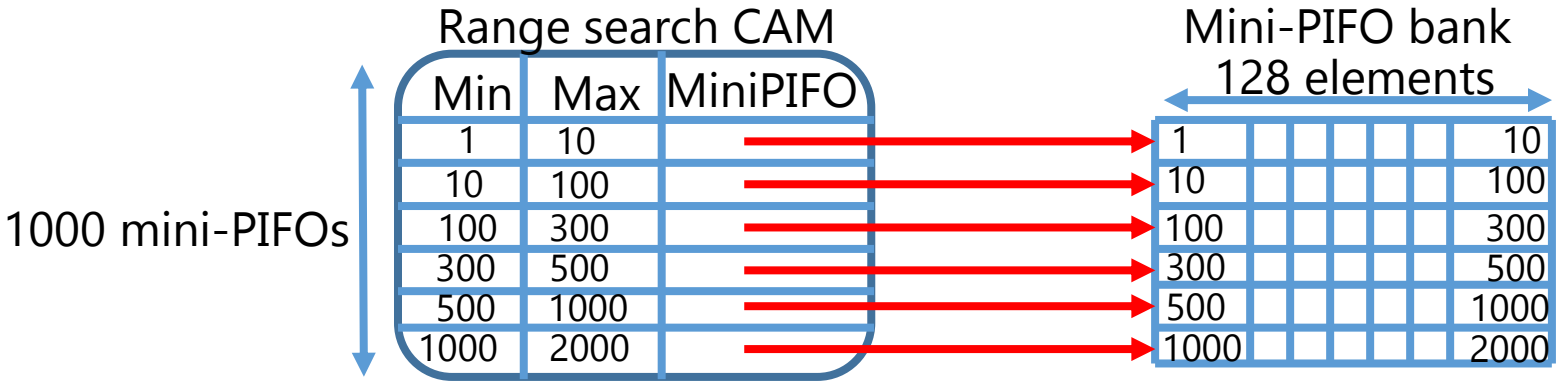
Composing PIFOs



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