

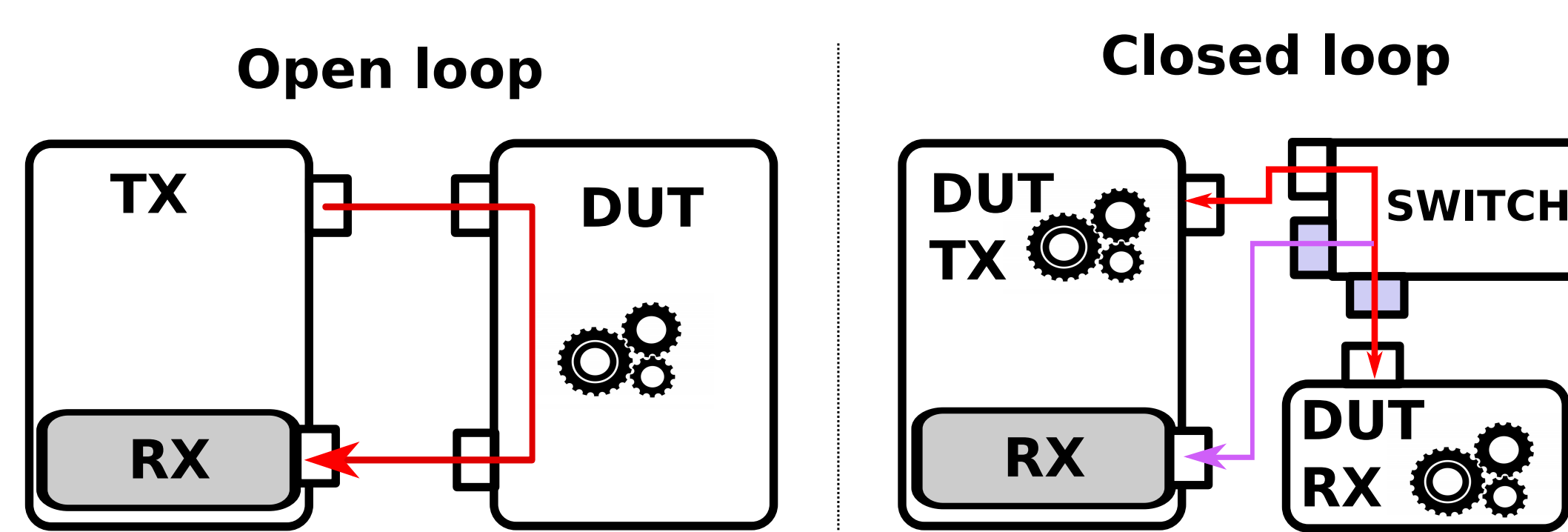
Demo: High-speed per-flow software monitoring with limited resources

Tianzhu Zhang, Leonardo Linguaglossa, Dario Rossi, *Telecom ParisTech*

Massimo Gallo, *Nokia Bell Labs*

Paolo Giaccone, *Politecnico di Torino*

Performance evaluation of Device Under Test (DUT)



Existing works

- Too simple: Speedometer
 - Packet counting, throughput measurement
- Occupy a lot of resources: DPDKStat
 - Advanced features such as DPI
 - Cannot co-locate with TX or DUT
 - Consider worst-case traffic

Motivation

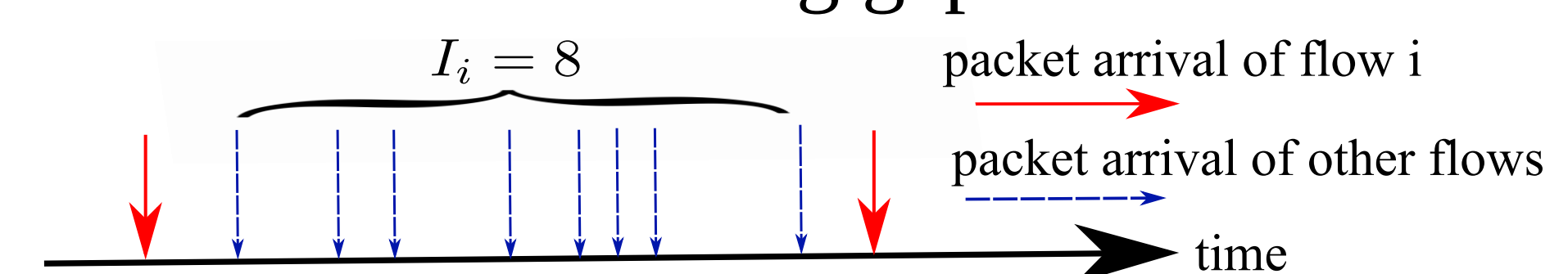
- Perform more complex monitoring tasks
- Occupy minimal resources

Contribution

- Implement FlowMon-DPDK
 - Release it as open source
 - <https://github.com/ztz1989/FlowMon-DPDK>
- Explore the design space
- Test it with worst-case traffic and public traces
- Showcase its usage with two existing prototypes
 - VPP fair drop
 - ClickNF

FlowMon-DPDK essentials

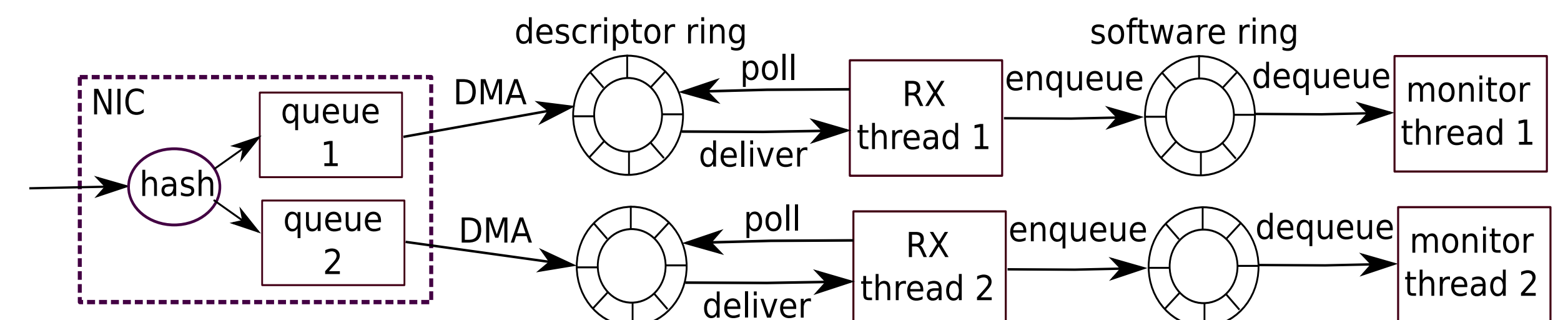
- A light-weighted software packet monitor
- Based on DPDK, written in C
- Support packet-level statistics
 - HW counting (performance baseline)
 - SW counting
- Support flow-level statistics
 - Per-flow packet counting
 - Per-flow interleaving gap



- Percentile with PSquare algorithm

Design space

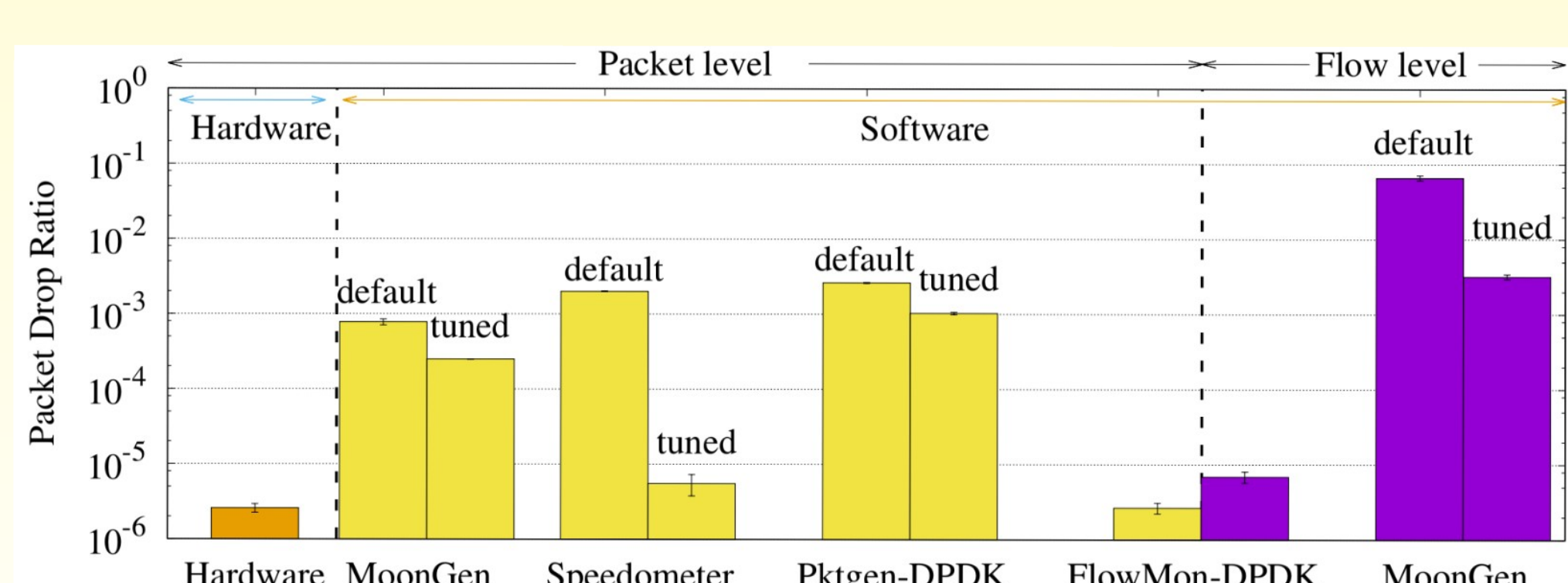
- Pipeline model



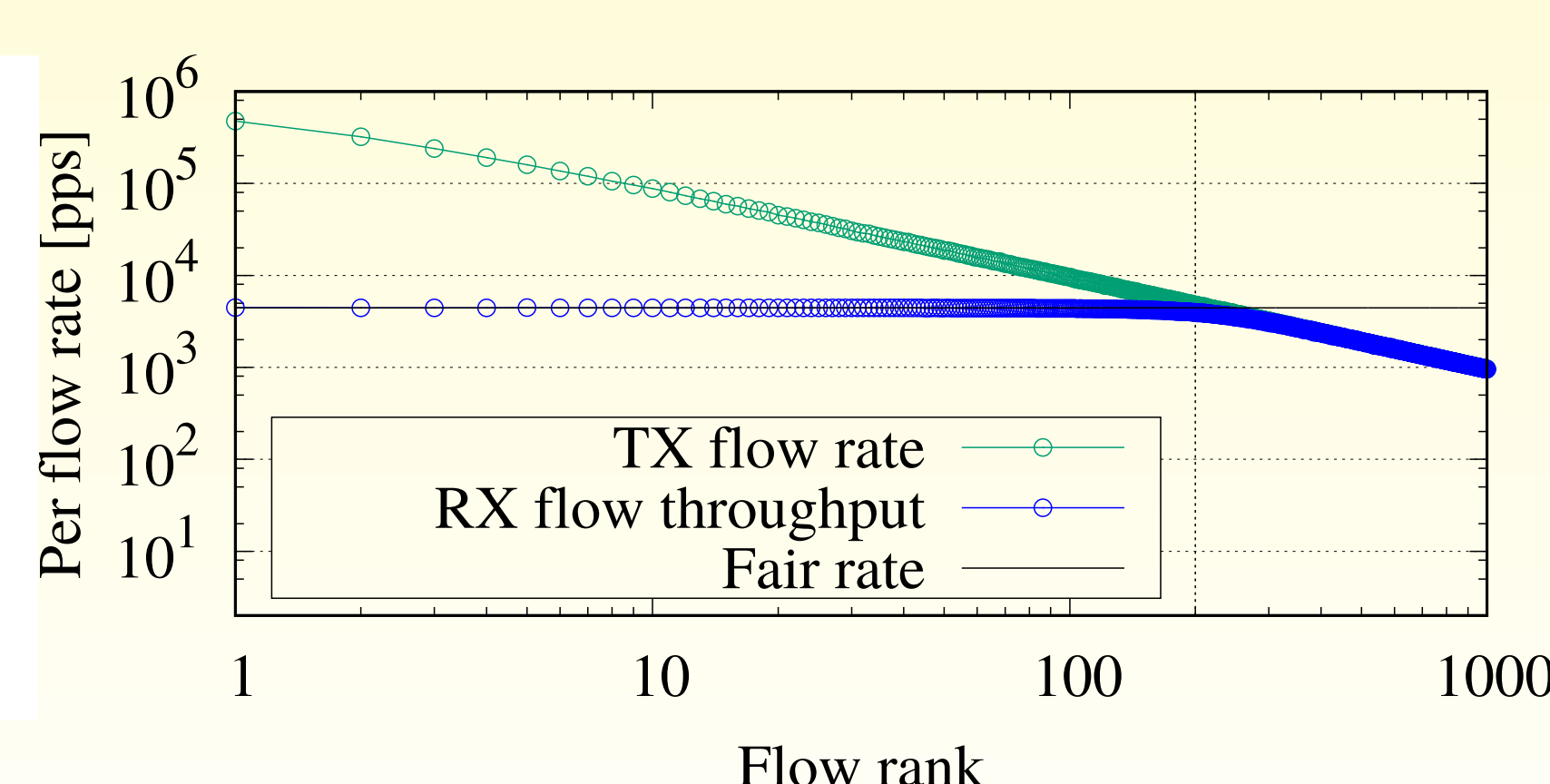
- **POSIX thread** vs. Lthread
- Flow identifier
 - Online hash vs. **RSS hash**
- CPU schedulers
 - **sched_other** vs. sched_deadline
- Software parameters
 - Batch size: 32, 64, 128, **256**, 512, 1024
 - RX queue size: 1024, 2048, **4096**
 - # RSS queues: 1, 2, 3, 4
- Flow table
 - Double hash vs. linked-list vs. combined

Experimental results

Compare with state-of-the-art



Use case 1: VPP fair drop



Use case 2: ClickNF

