Deutsche Telekom Laboratories

Technische Universität Berlin

Revisiting router architectures with Zipf

Steve Uhlig
Deutsche Telekom Laboratories/TU Berlin

Nadi Sarrar, Anja Feldmann
Deutsche Telekom Laboratories/TU Berlin

Rob Sherwood, Xin Huang
Deutsche Telekom Labs USA

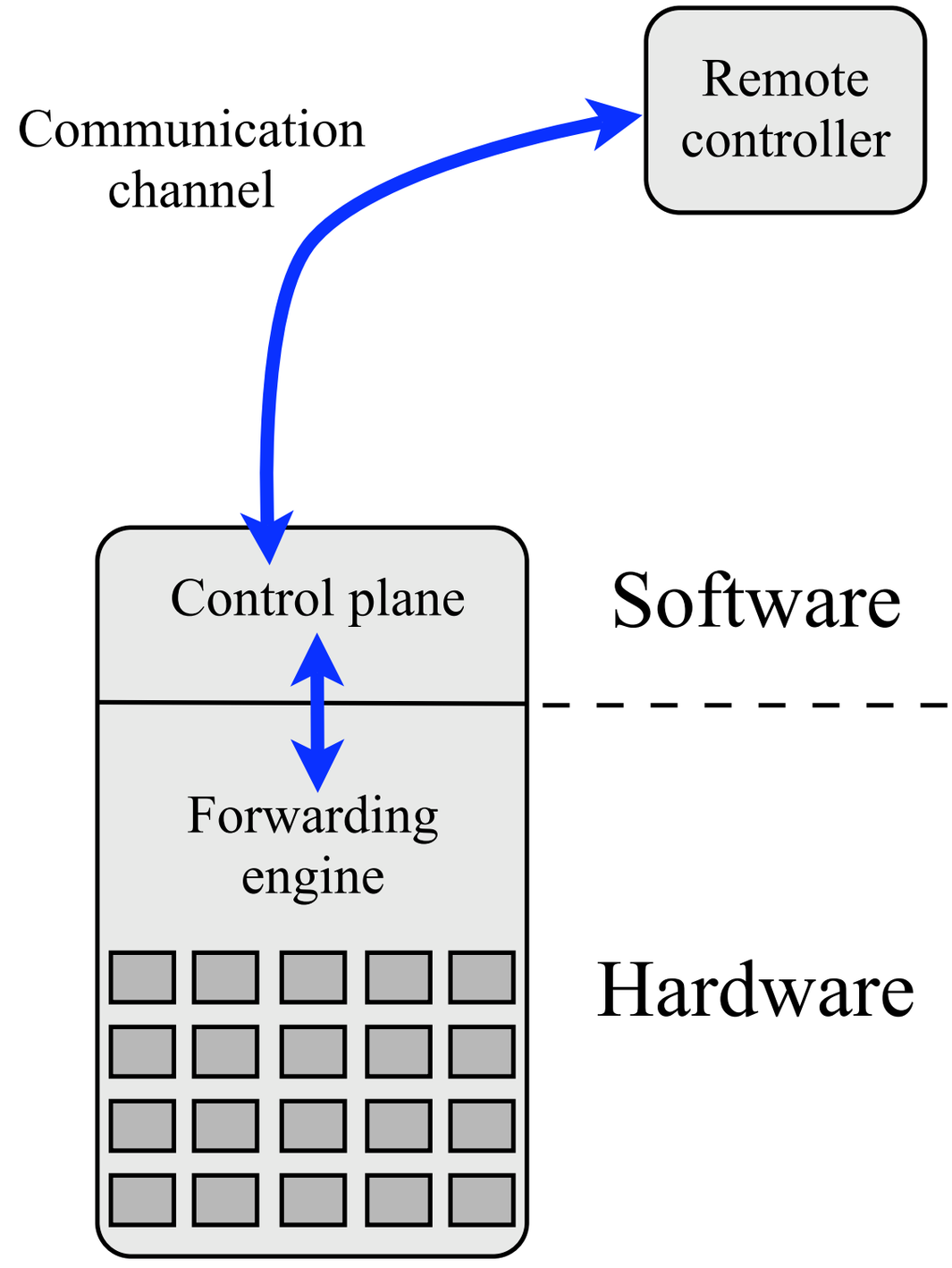
Social Networks and Future Internet, Annecy, France, June 27-28, 2011

Agenda

- **Motivation**
- Revisiting IP routers architecture
- Opportunities and challenges
- Evaluation
- Conclusion

Software-defined networking

- Beyond today's monolithic network equipment
- Separation of control and data plane through software modularity, e.g., Linux
- Do not change existing control plane
- Principles
  - Communication channel between forwarding engine and remote controller
  - Expose network equipment capabilities, e.g., TCAM, QoS

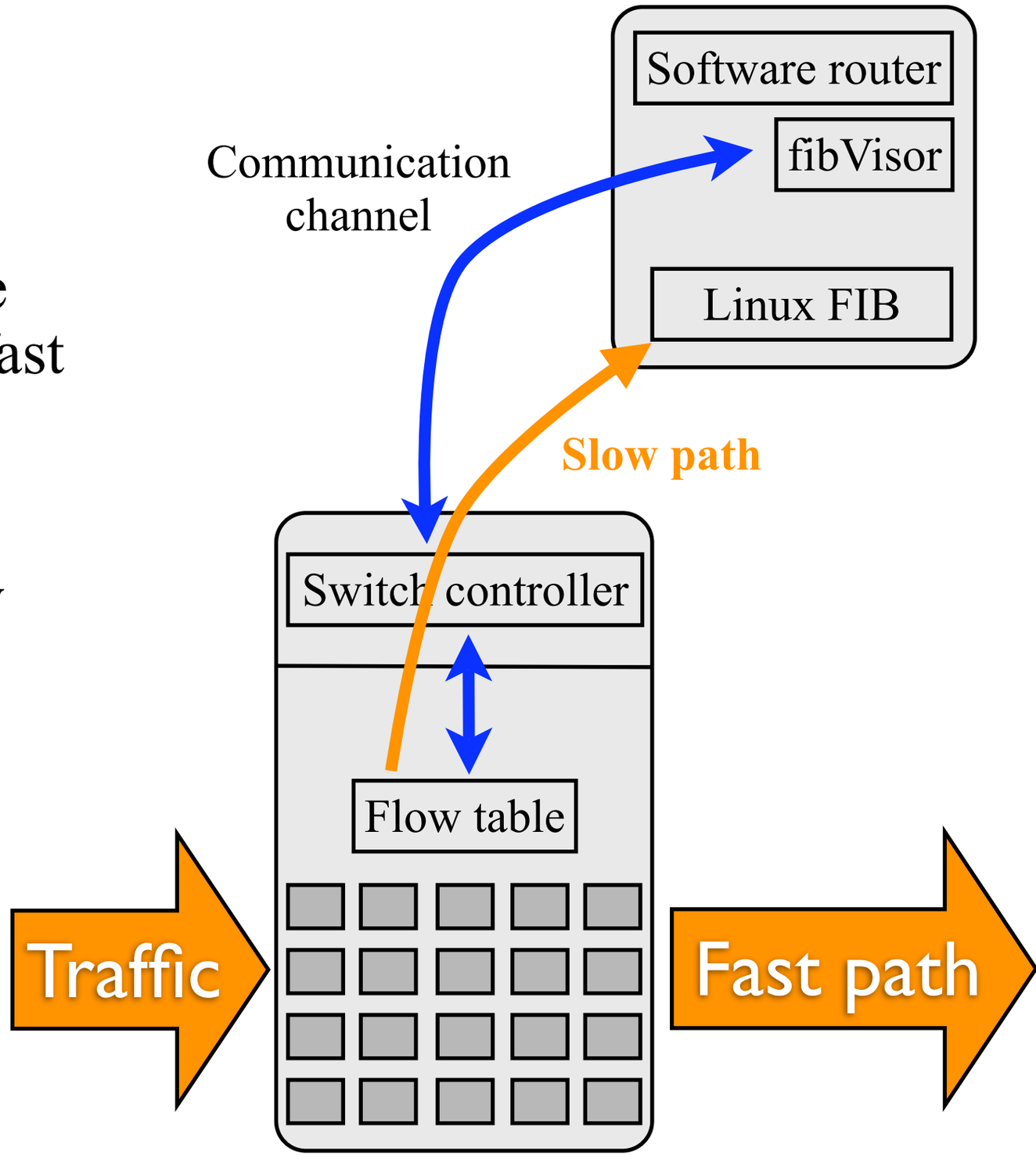


Agenda

- Motivation
- **Revisiting IP routers architecture**
- Opportunities and challenges
- Evaluation
- Conclusion

OpenFlow-based IP router

- System combines software router with fast switching hardware
  - Fast switch handles most of the traffic with a few entries, i.e., fast path
  - Software handles control plane and remaining traffic, i.e., slow path
- Our approach:
  - Take advantage of traffic properties

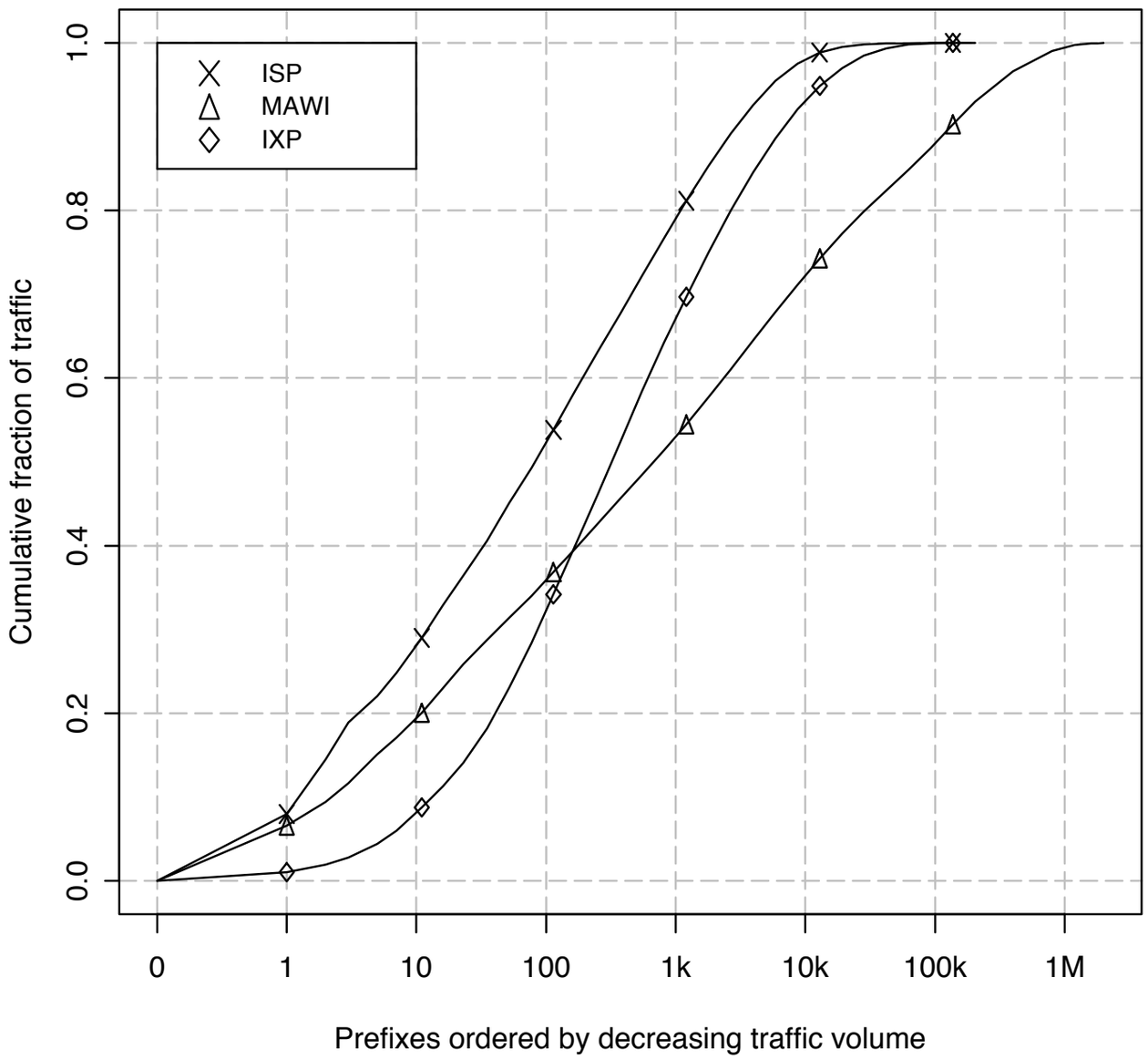


Agenda

- Motivation
- Revisiting IP routers architecture
- **Opportunities and challenges**
- Evaluation
- Conclusion

Zipf in Internet traffic

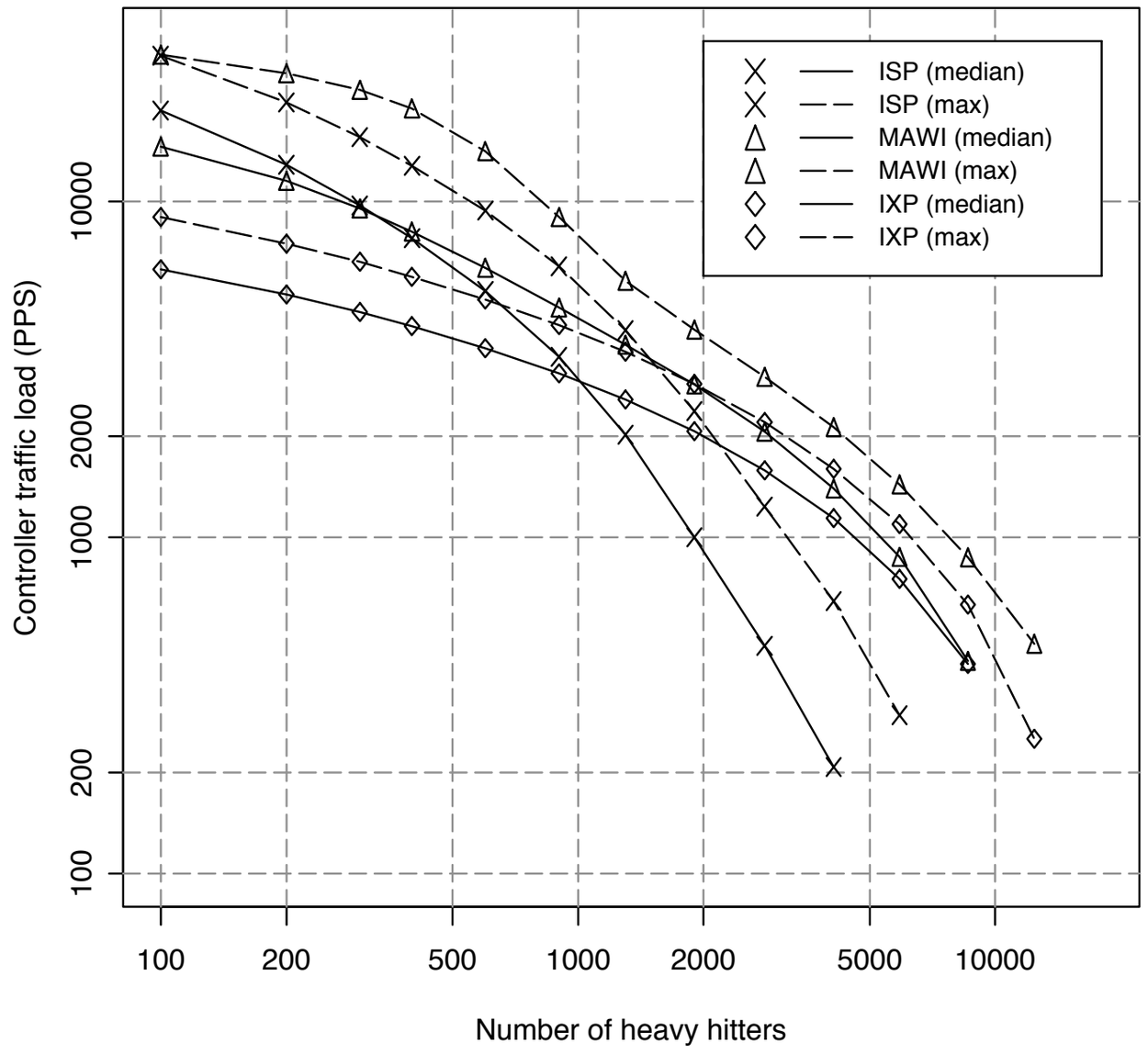
- Data
  - Transcontinental link: 150Mbps backbone link (MAWI), 3.5 days
  - Residential ISP: 1Gbps link, 2 days
  - IXP: > 1Tbps, 4 days
- Observation: Most traffic captured by limited number of prefixes



➡ **Opportunity: In principle, existing switching hardware can do it**

Slow path

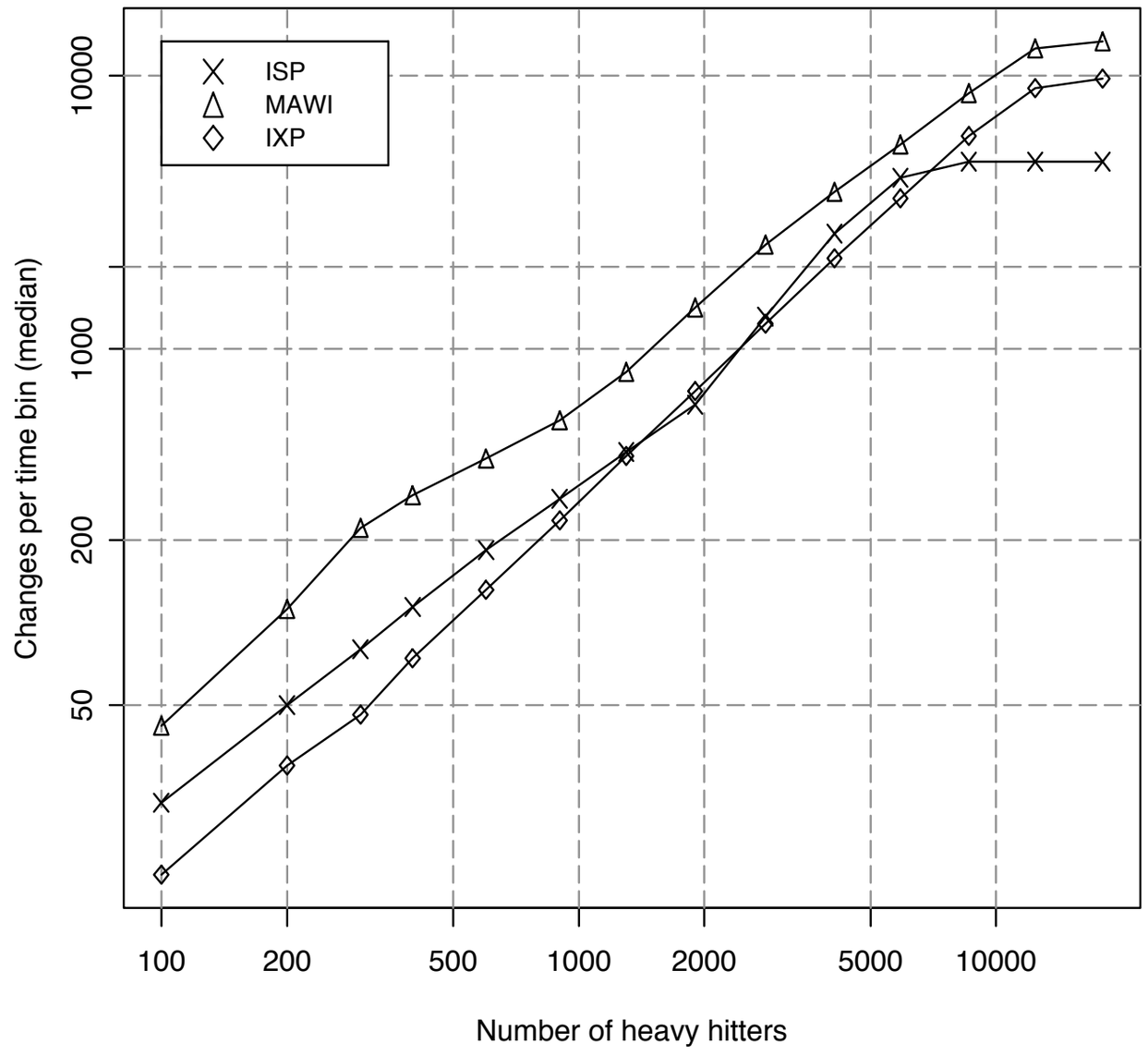
- Assume knowledge of the future traffic
- Slow path rate as a function of number of heavy-hitters
- A few thousand heavy hitters enough to keep slow path rate low
- Limited variations across traces



➡ Opportunity: with a few thousand flows, slow path rate can be kept low

Churn

- Assume knowledge of the future traffic
- What is the best-case churn rate?
- Proportional to number of heavy-hitters



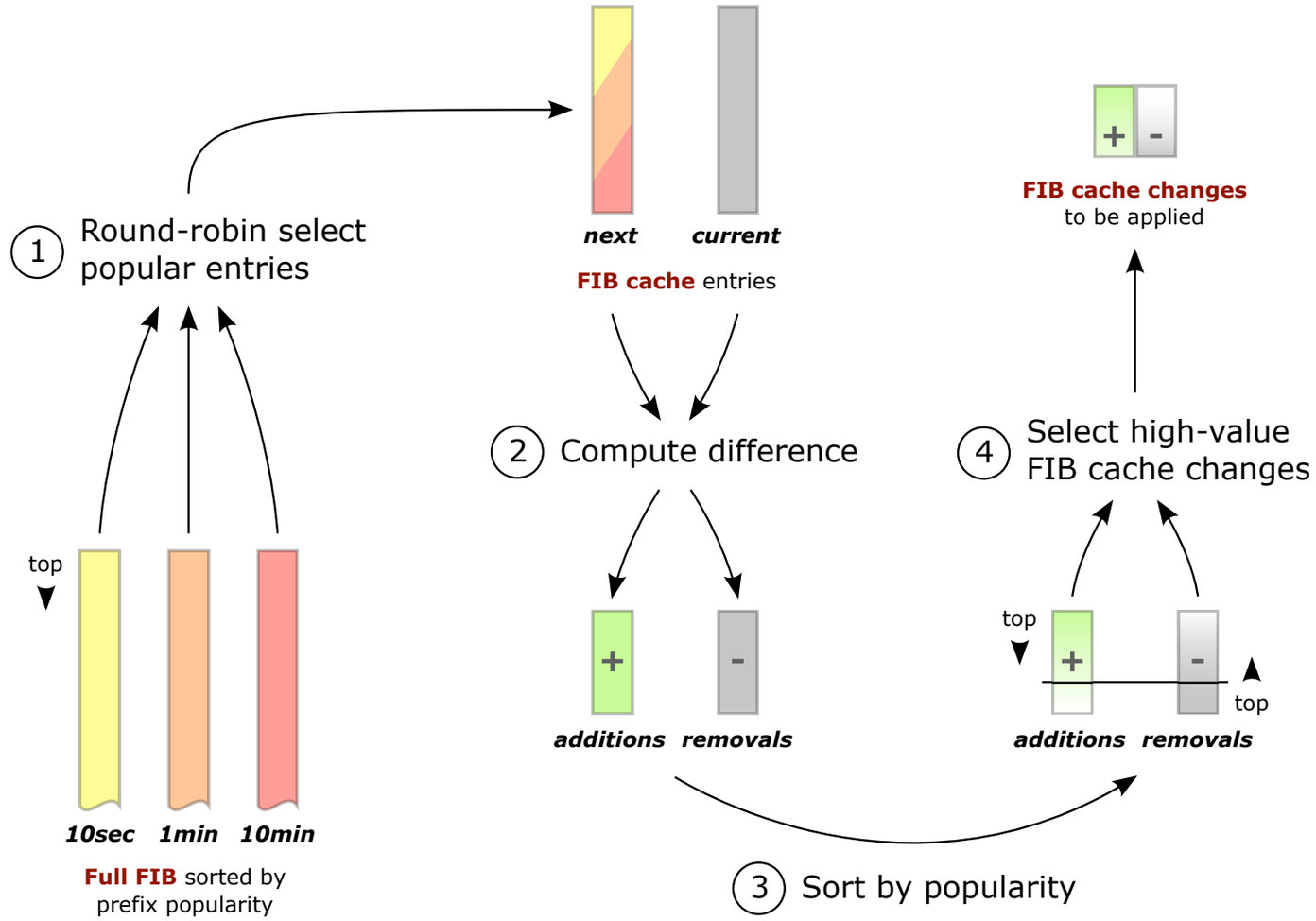
➡ **Challenge: keep churn low while keeping most of the traffic on the fast path**

Agenda

- Motivation
- Revisiting IP routers architecture
- Opportunities and challenges
- **Evaluation**
- Conclusion

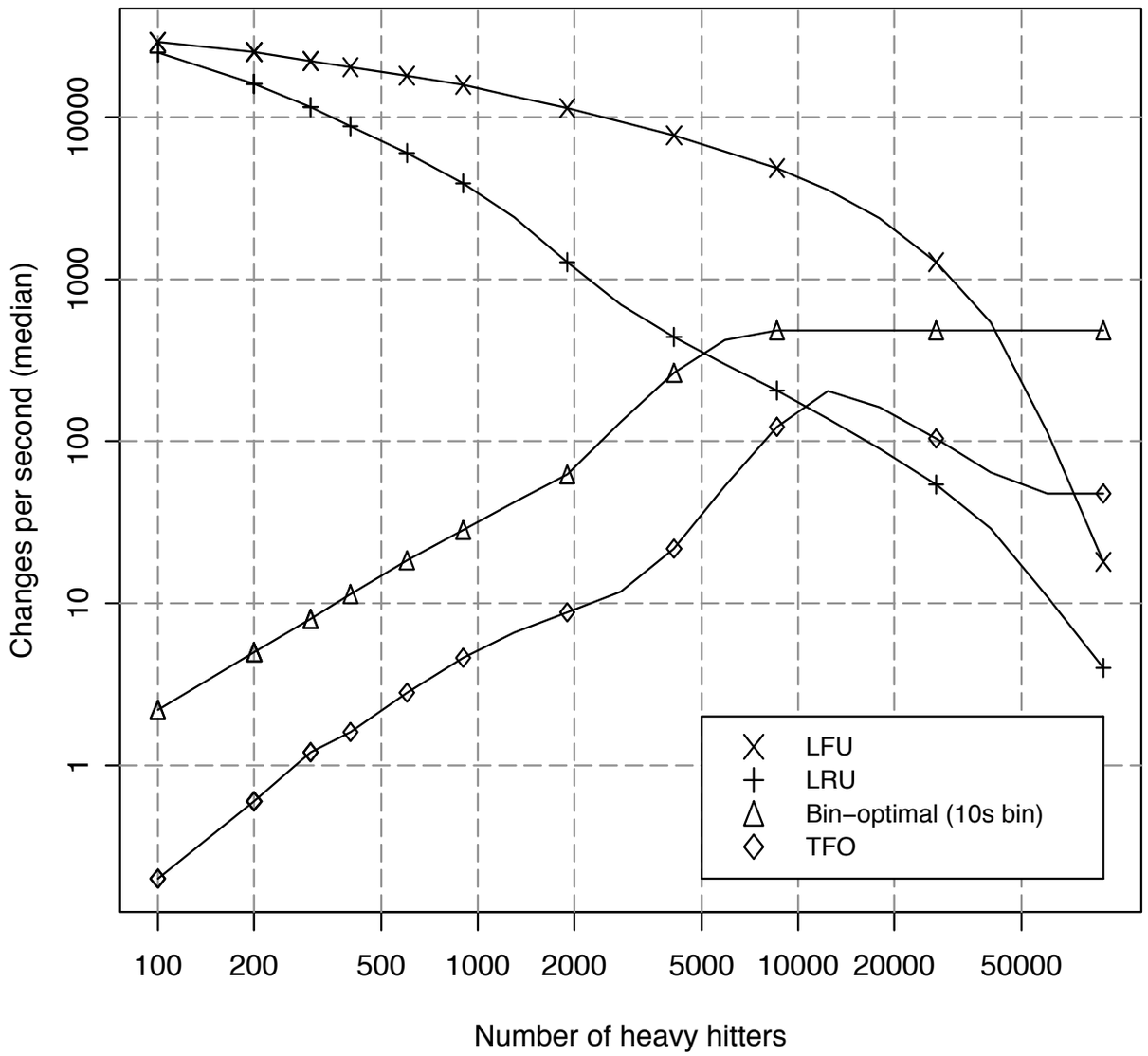
Traffic Offloading

- Tame natural churn of heavy-hitters
- Traffic Offloading (TFO) Algorithm
  - Monitor traffic at multiple time-scales
  - Select heavy-hitters that are expected to lead to low churn
  - Trade-off offloading gain with churn



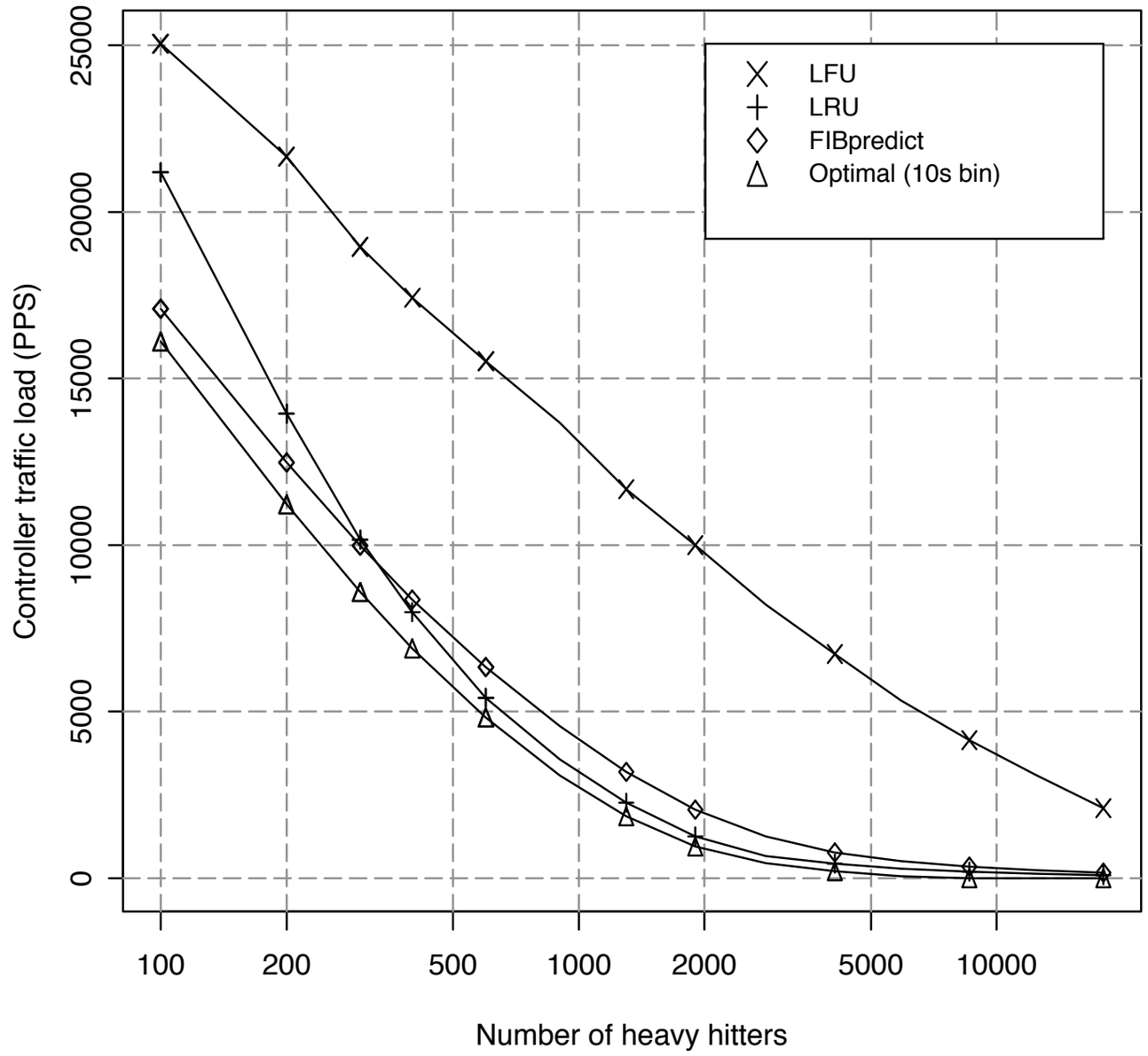
Churn

- Traditional caching
  - Always replace entry upon miss
  - Leads to high churn for low number of heavy-hitters
- TFO keeps churn much lower than bin-optimal and caching
- When number of heavy-hitters high, combination of caching and TFO is ideal



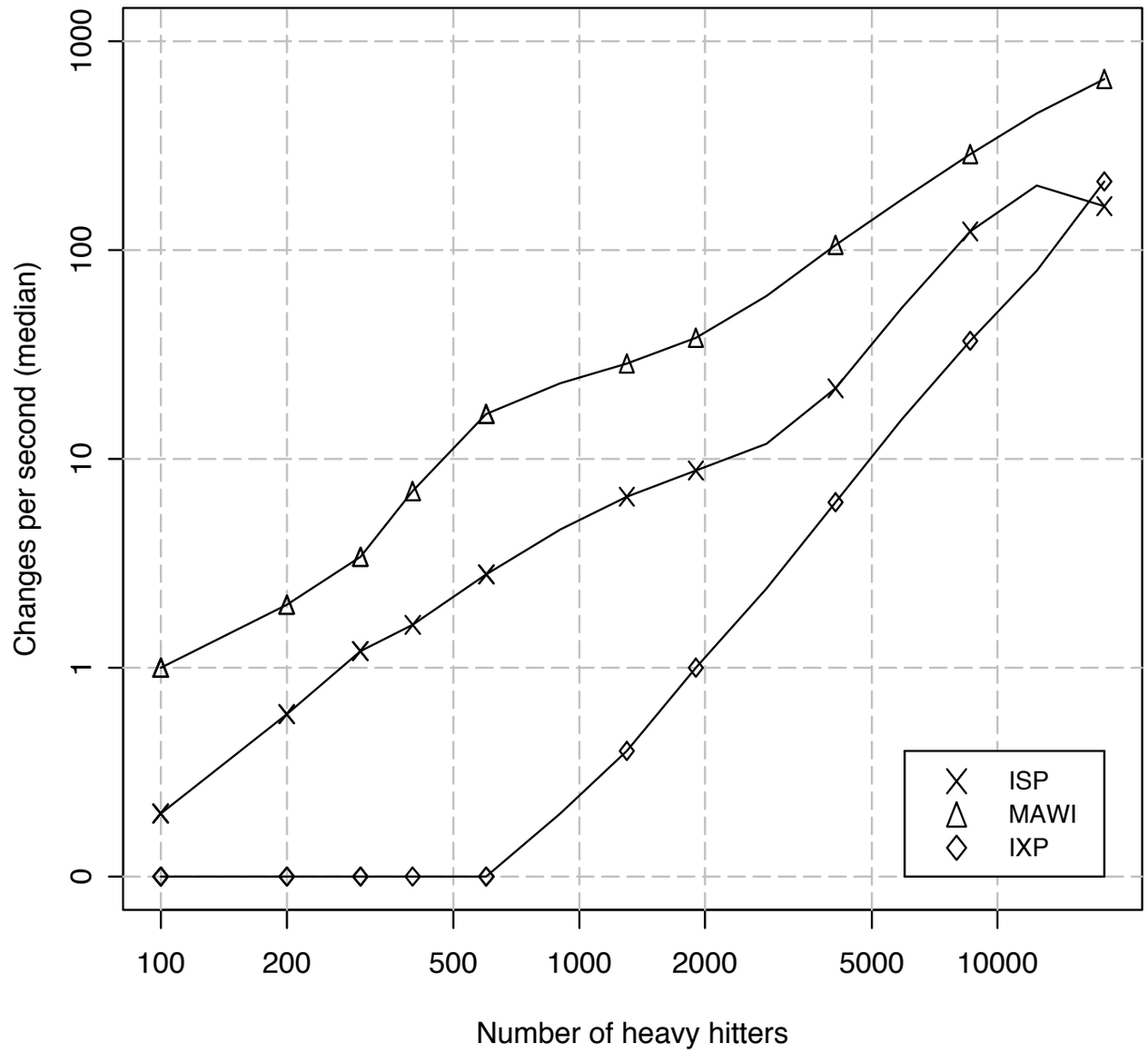
Slow path

- LFU shows importance of heavy-hitters dynamics over short time-scales
- LRU and TFO close to optimal
- Slow path rate low for a few thousand heavy-hitters



TFO: churn

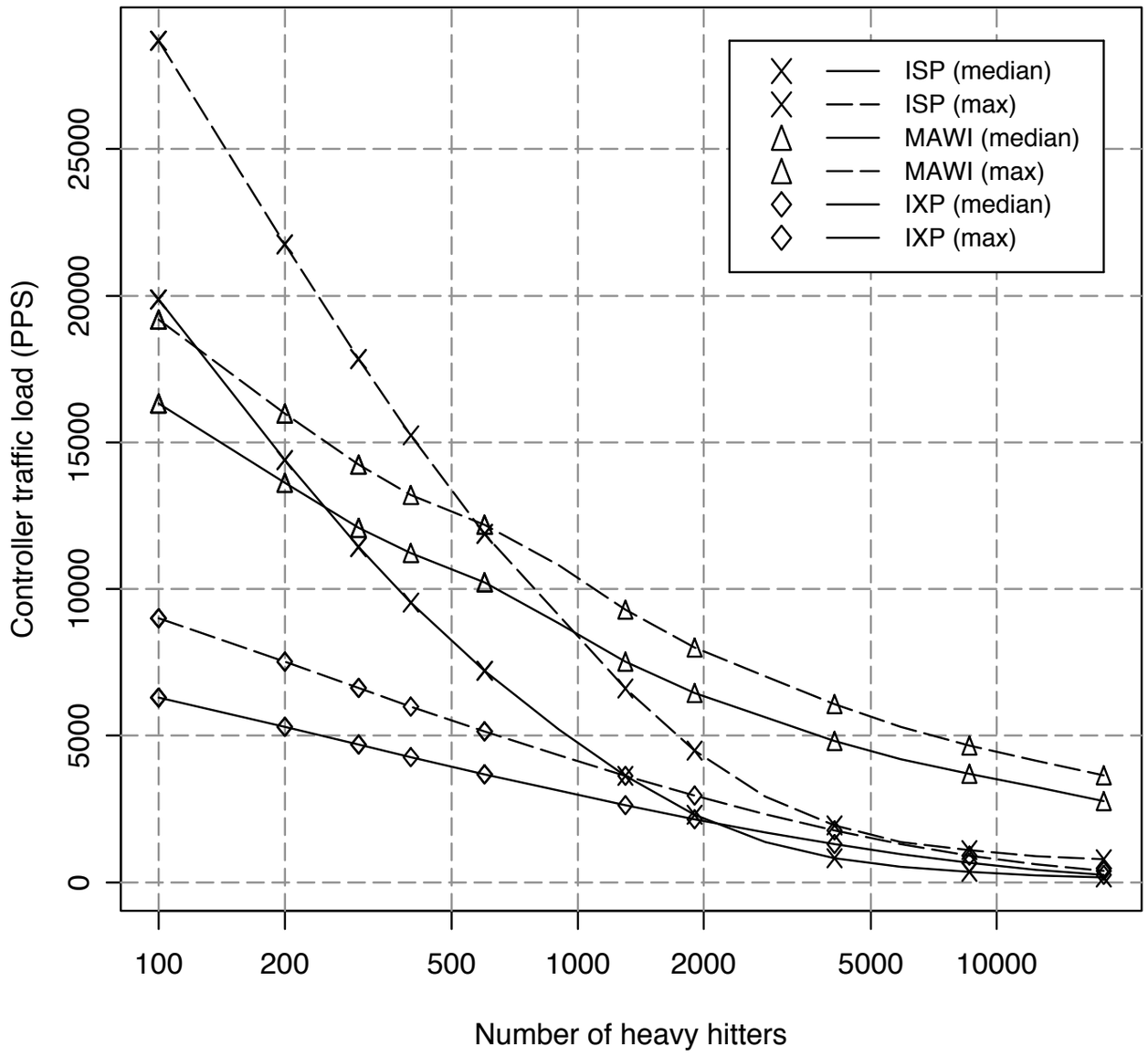
- Churn depends on traffic aggregation
  - IXP: a few changes per second
  - ISP: 10's of changes per second
  - Transcontinental link: up to 100 of changes per second
- TFO tames the churn



➡ **Feasible on today's OpenFlow-enabled switches**

TFO: slow path

- Load can be handled by commodity PC
- Could be done on better embedded switch CPU
- Scaling up
  - Routebricks
  - Packetshader
  - Traditional router
  - …



➡ **Feasible on today's commodity hardware**

Agenda

- Motivation
- Revisiting IP routers architecture
- Opportunities and challenges
- Evaluation
- **Conclusion**

Conclusion

- Revisiting router architecture through SDN
  - Leverage traffic properties (Zipf)
  - Combine open-source routing with fast and cheap switches
- TFO algorithm
  - Beyond traditional caching: carefully select the right heavy-hitters
  - Keep both churn and slow path rates low
- Scale up routers: fast switching hardware + fast software routers