Orchestrating a brighter world NEC

CEDRO: an in-switch elephant flows rescheduling scheme for data-centers

Davide Sanvito◆*, Andrea Marchini+
Ilario Filippini+, Antonio Capone+

◆ NEC Laboratories Europe, Germany
+ DEIB, Politecnico di Milano, Italy

*Work carried out while at *Politecnico di Milano*

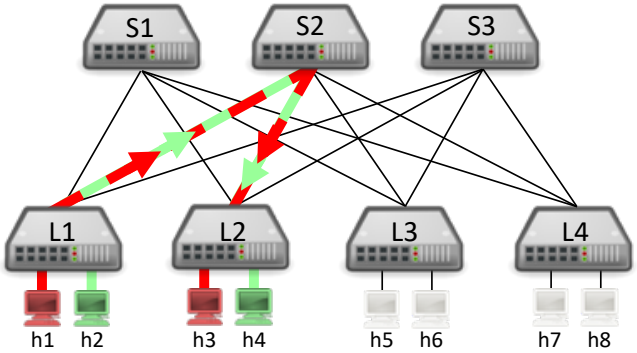
Introduction

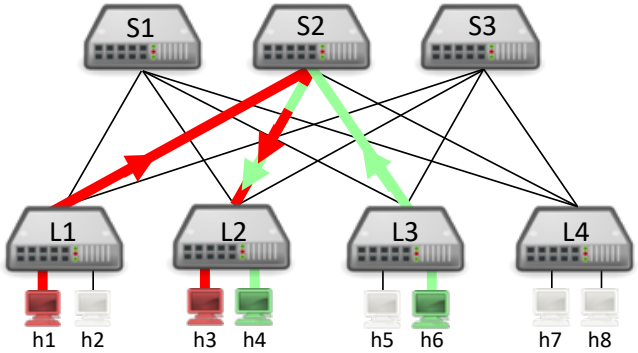
- Data-center networks
- Equal Cost Multi-Path (ECMP)
- Elephant vs mice flows
- Latest advances in programmable network devices
  - Opportunities for network self-adaptation
  - More scalable and prompt reaction compared to CP reactive approaches

CEDRO

Congested **E**lephant **D**etection and **R**e-routing **O**ffloading

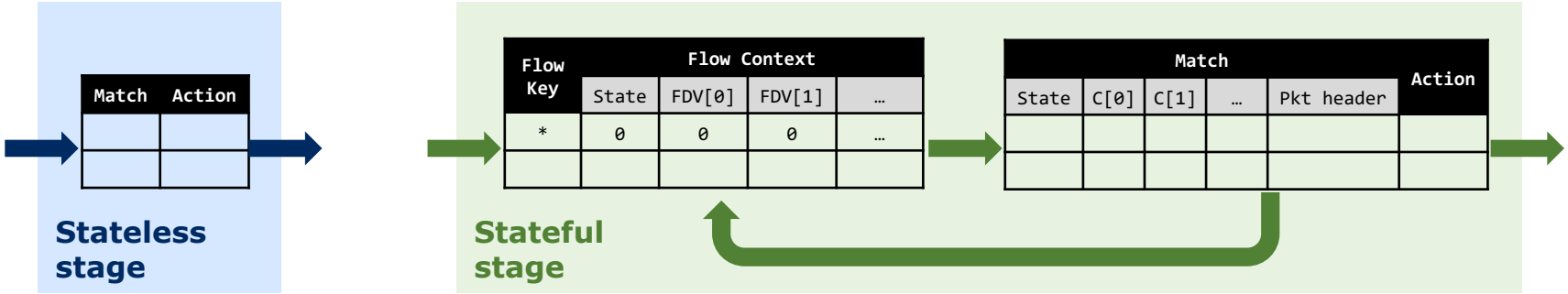
- In-switch mechanism to detect and re-route large flows colliding on a same downstream path
- Based on programmable stateful data planes
- ECMP override
- Local and remote congestion scenarios w/o controller





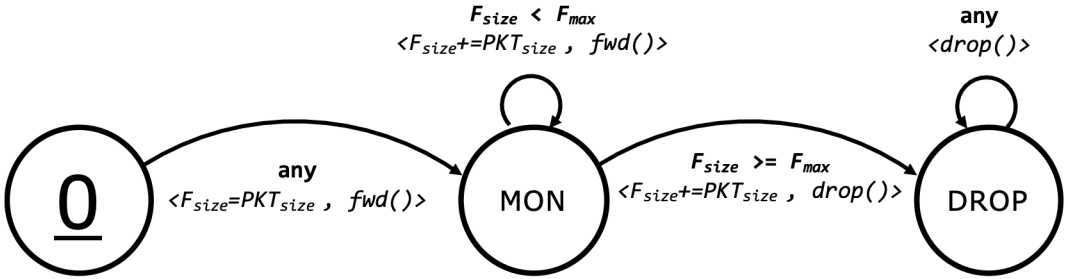
Open Packet Processor (OPP)

- Stateful extension to OpenFlow
- Control function offloaded to the data plane
  - In-network forwarding self-adaptation
- Programming abstraction based on EFSM
  - Flows associated to persistent context (state and data variables)
  - Forwarding based on packet header and state
  - State transition in the DP based on time-/packet-events or conditions



OPP toy example

Per-user (i.e. per-src-IP) tx bytes quota limiter



Flow Key	Flow Context	
	State	F_{size}
*	0	0
10.0.0.1	MON	800
10.0.0.5	DROP	1005

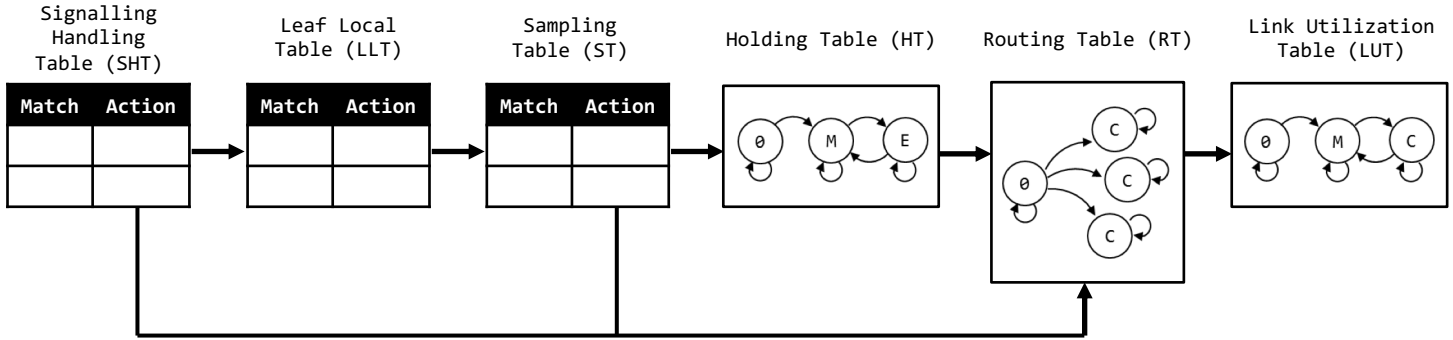
Match			Action
State	$F_{size} >= F_{max}$	Pkt header	
0	*	*	SetState(MON) , $F_{size}=PKT_{size}$, fwd()
MON	0	*	$F_{size}+=PKT_{size}$, fwd()
MON	1	*	SetState(DROP) , $F_{size}+=PKT_{size}$, drop()
DROP	*	*	drop()

CEDRO pipeline overview

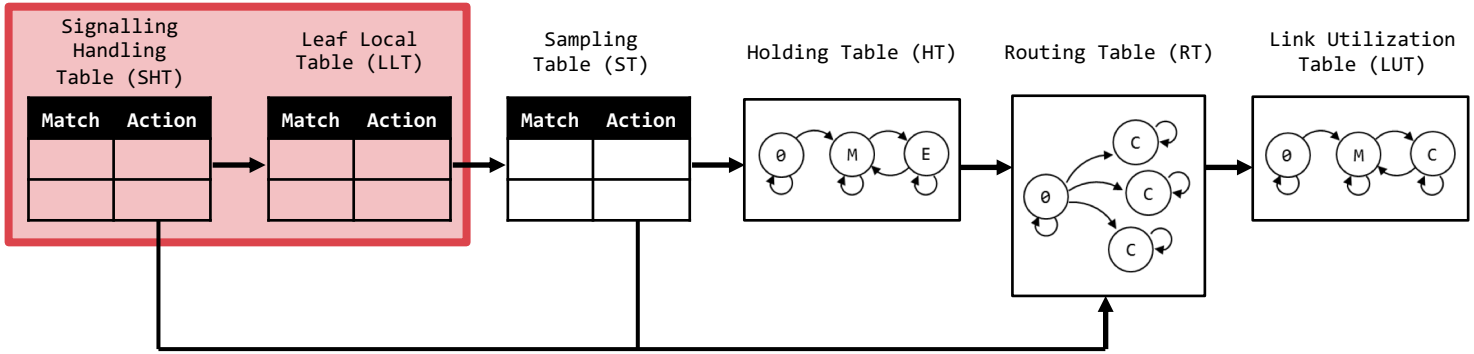
Leaf-spine topology

- Leaf vs spine pipelines

2 stateless stages and 4 stateful stages



CEDRO pipeline: stateless stages



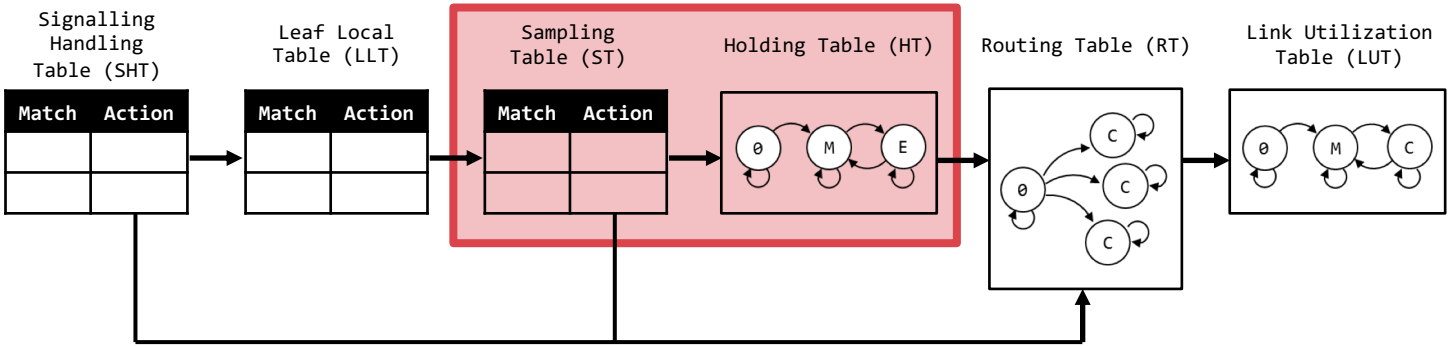
SHT

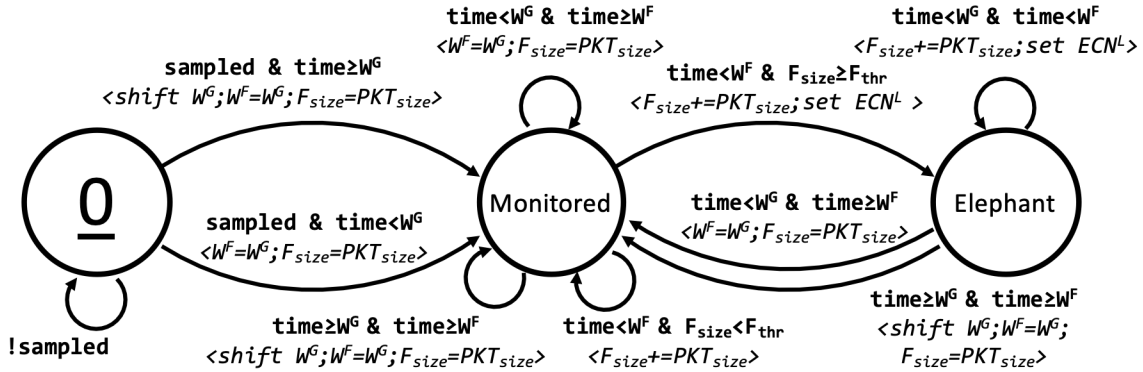
- Handle remote congestion signaling packets

LLT

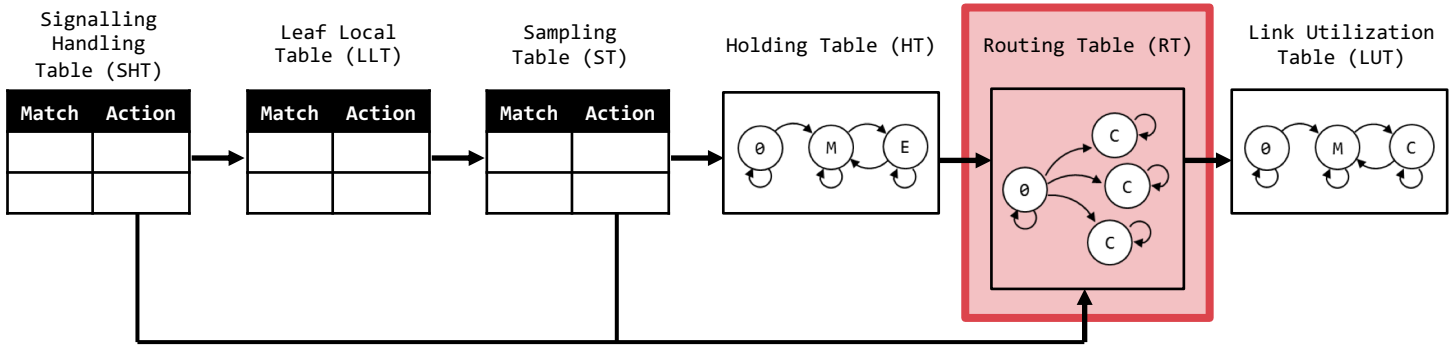
- Handle traffic local to the leaf node

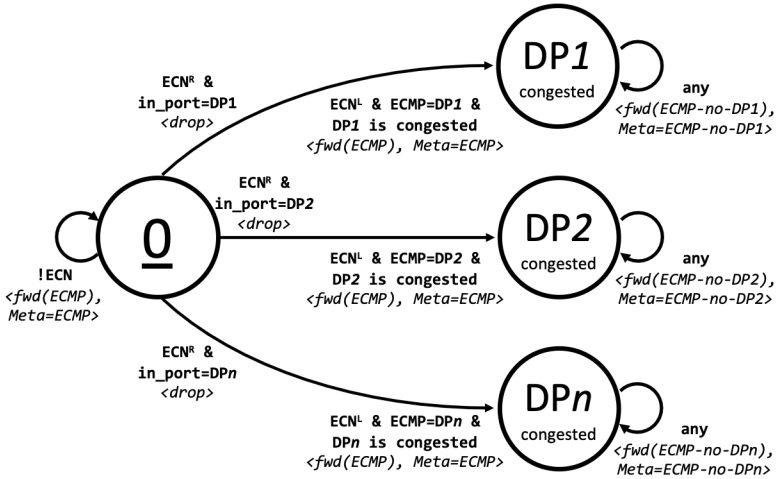
CEDRO pipeline: Sample & Hold

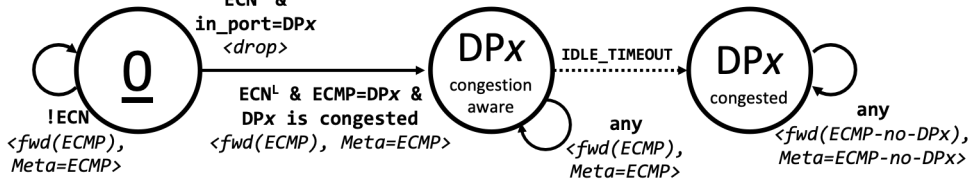




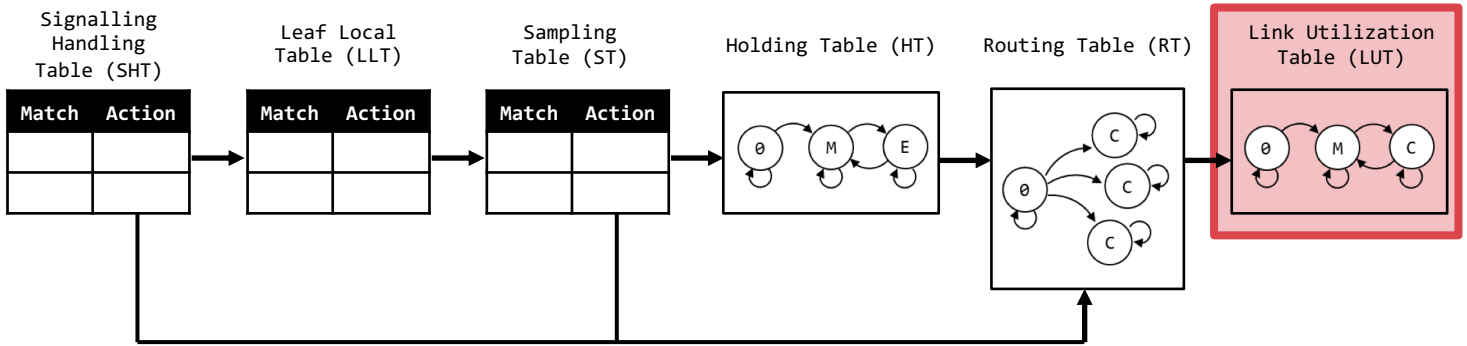
CEDRO pipeline: state-aware ECMP

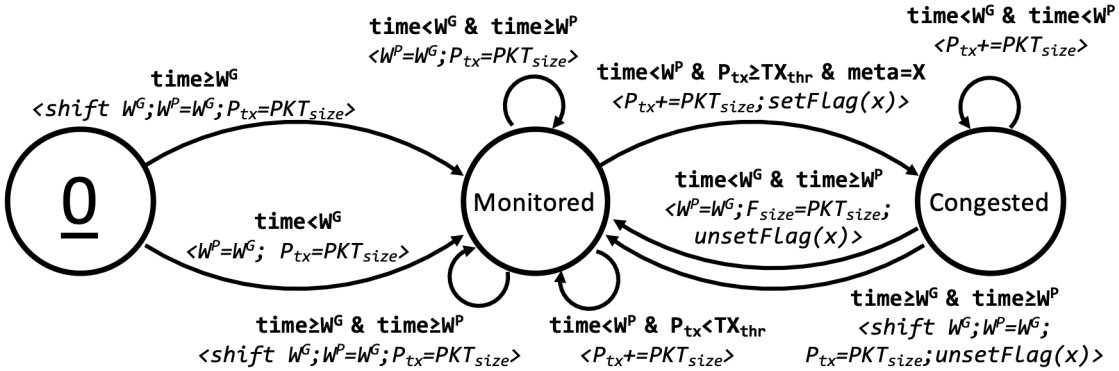






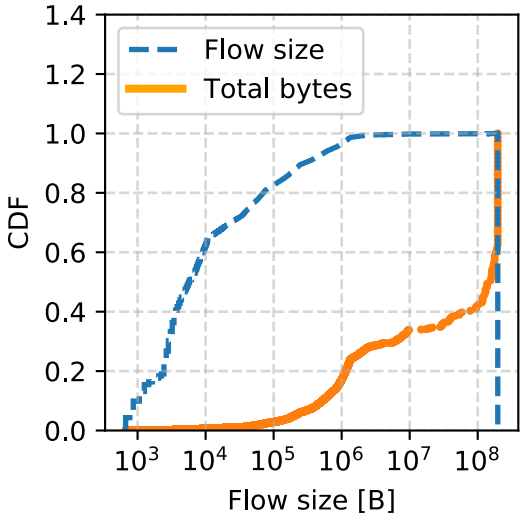
CEDRO pipeline: link utilization

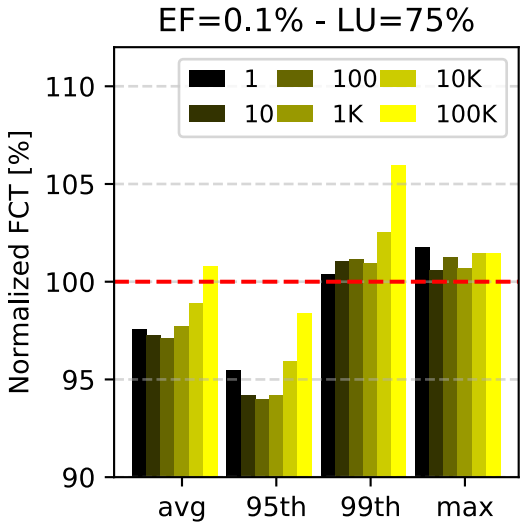




Mininet testbed evaluation

- ofsoftswitch13 (BOFUSS) and Ryu
- Leaf-spine (10 HPR, 10 Leaves, 5 Spines)
- FCT comparison wrt ECMP
  - 13k flows (DCT^{2}Gen)



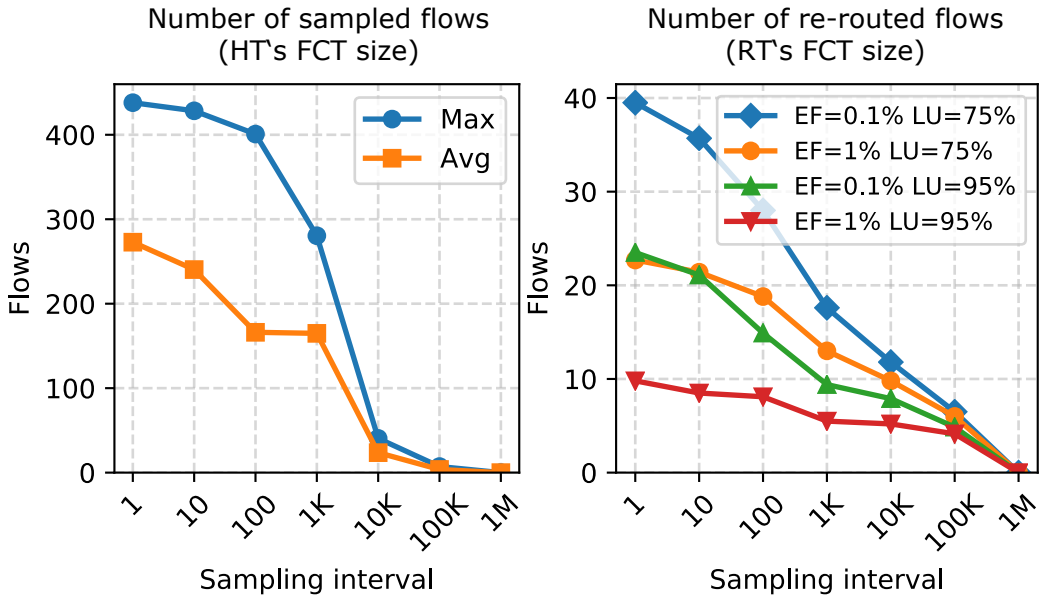


Memory requirements

Analytical analysis

Table Name	Flow Table (FT)	Flow Context Table (FCT)
SHT	2	-
LLT	O(HPR)	-
ST	3	-
HT	10	O(sampled flows)
RT	$O(DP^2)$	O(re-routed elephant flows)
LUT	O(DP)	O(DP)

Experimental analysis



Discussion

- Multiple re-routings
- Failures handling
  - ECMP override: microflow vs macroflow
  - SPIDER: in-network failure detection & recovery scheme
- Deeper multi-rooted topologies

Conclusion and future works

CEDRO

- In-switch scheme to detect and re-route congested large flows
- No external controller involvement
- No end-host cooperation or network stack modifications
- Reduction of avg and 95th FCT

Future works

- Large scale simulation
- Extension to deeper topologies
- Alternative rescheduling strategies based on the capabilities of OPP


Orchestrating a brighter world
NEC