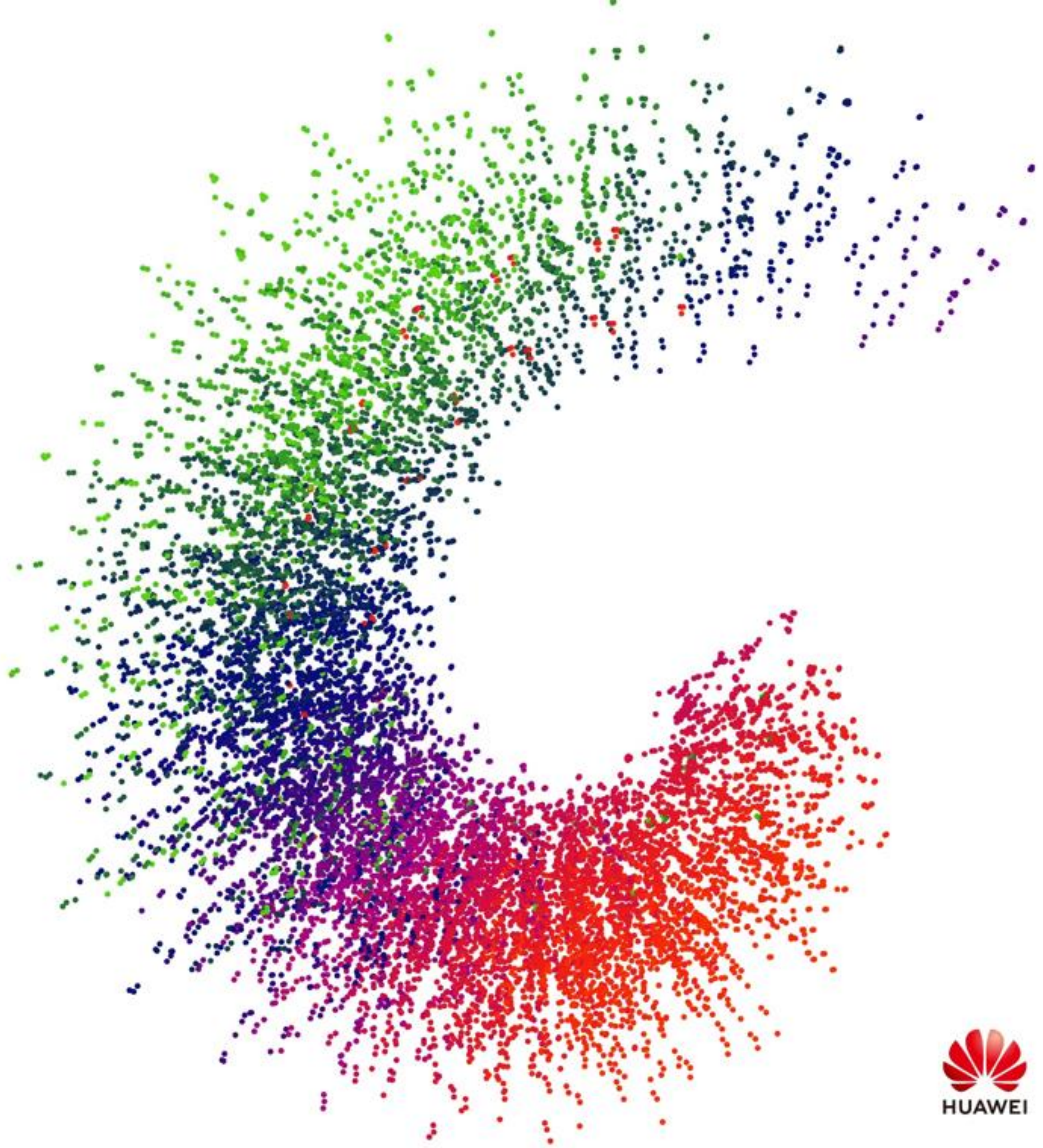
Robust network design and its application to slicing



Julien Lesca

Senior researcher, Network OR, Datacom Dijkstra Lab

HUAWEI

Joint work with:

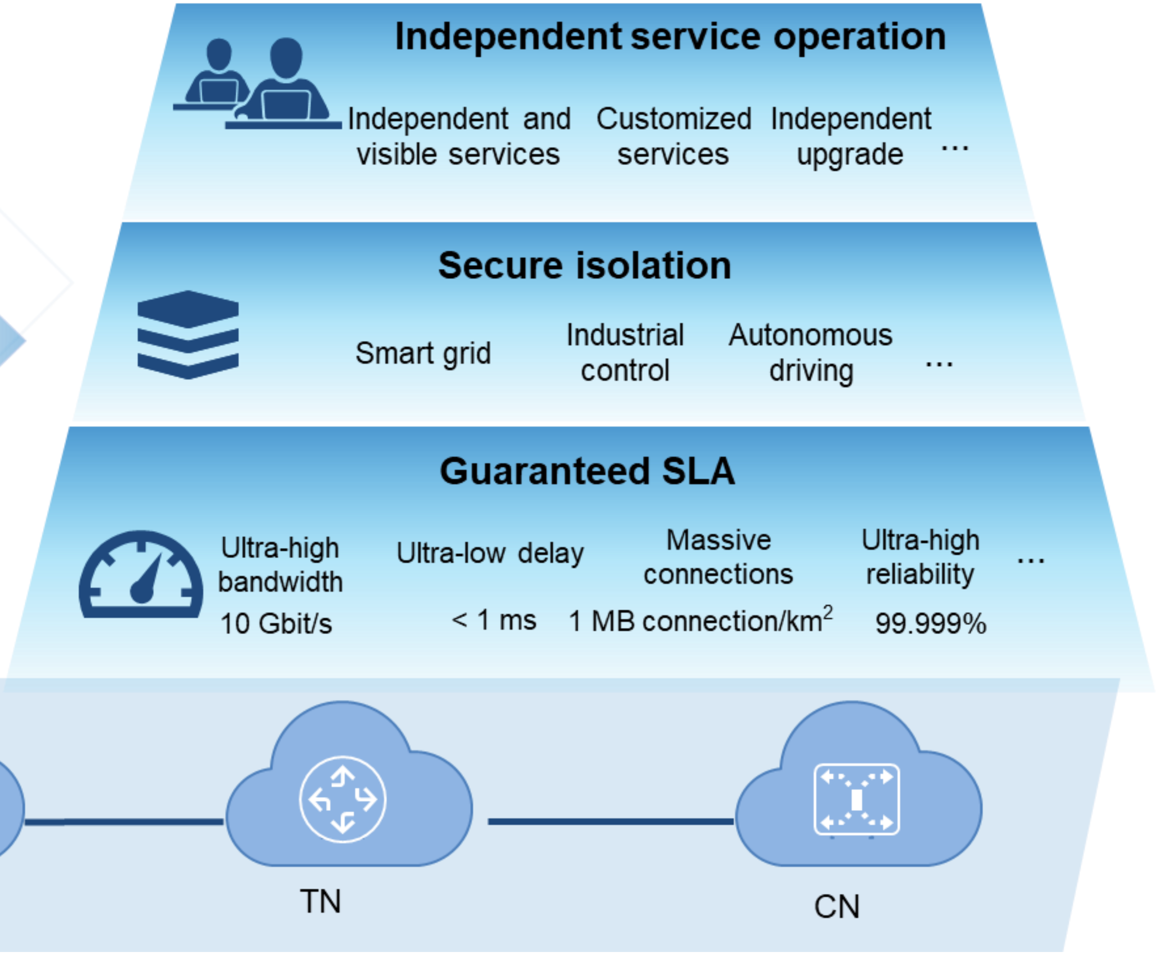
Paolo Medagliani, Jérémie Leguay, Sebastien Martin, CaiShengming, Zengfeng, Yacine Al Najjar, Walid Ben Ameur

Agenda: | Slicing | Robust Design | Algorithm | Results

Network Slicing Motivation: 5G and beyond

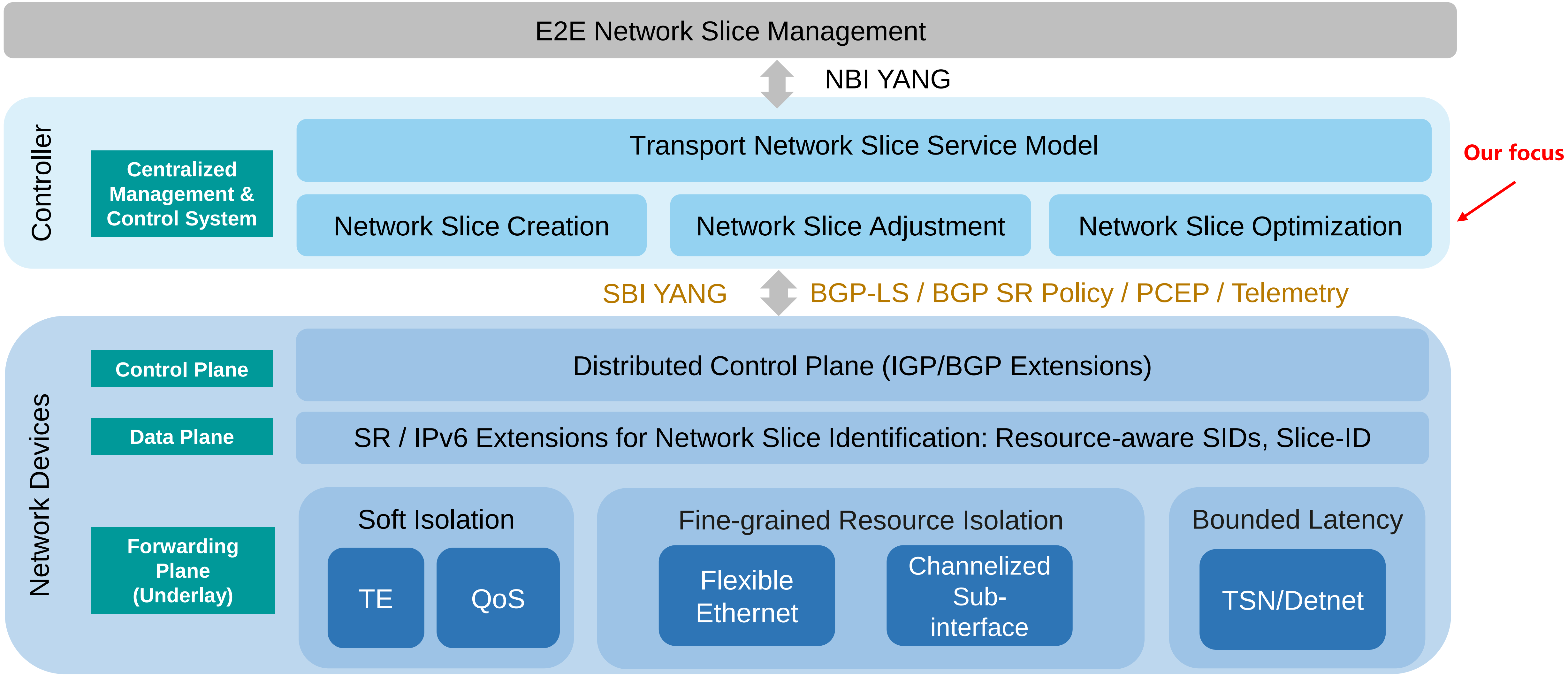
5G vision: enabling industry digitalization

Key requirements for 5G networks: differentiated network services

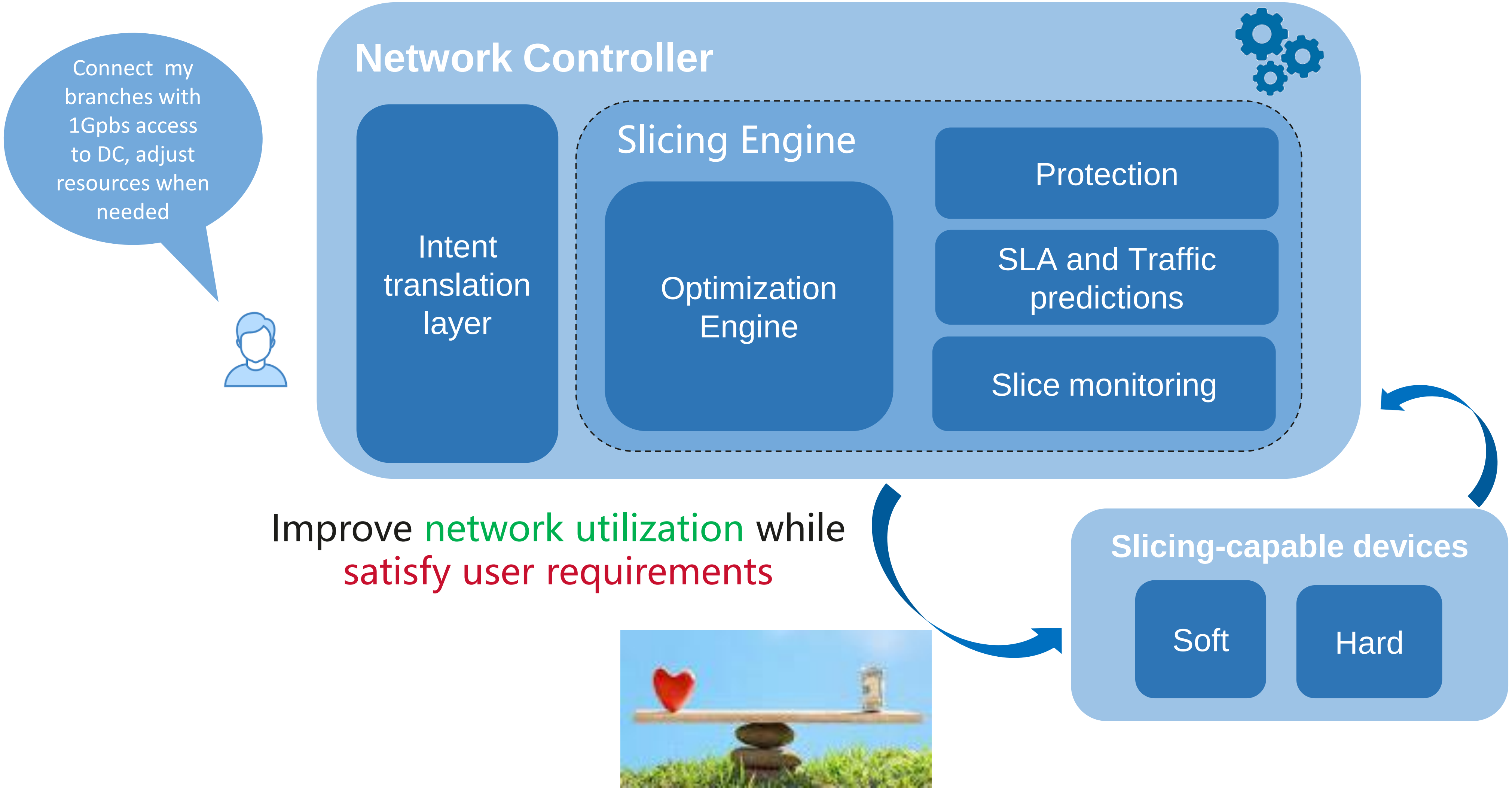


Slices = on-demand and end-to-end isolated networks

Network Slicing Architecture



Towards an Advanced Automation Slicing Engine



Improve network utilization while satisfy user requirements

Simplicity

Simple Inputs
Different Slicing technologies

Reliability

Failure protection
Traffic anomalies

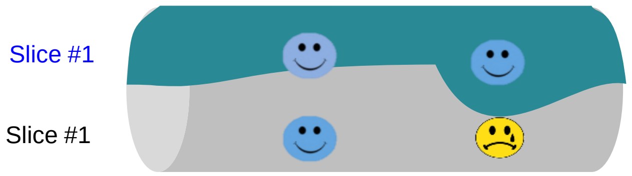
Efficiency

Resource savings

Challenge #1: Allocate resources to meet SLA Requirements

Isolation between slices

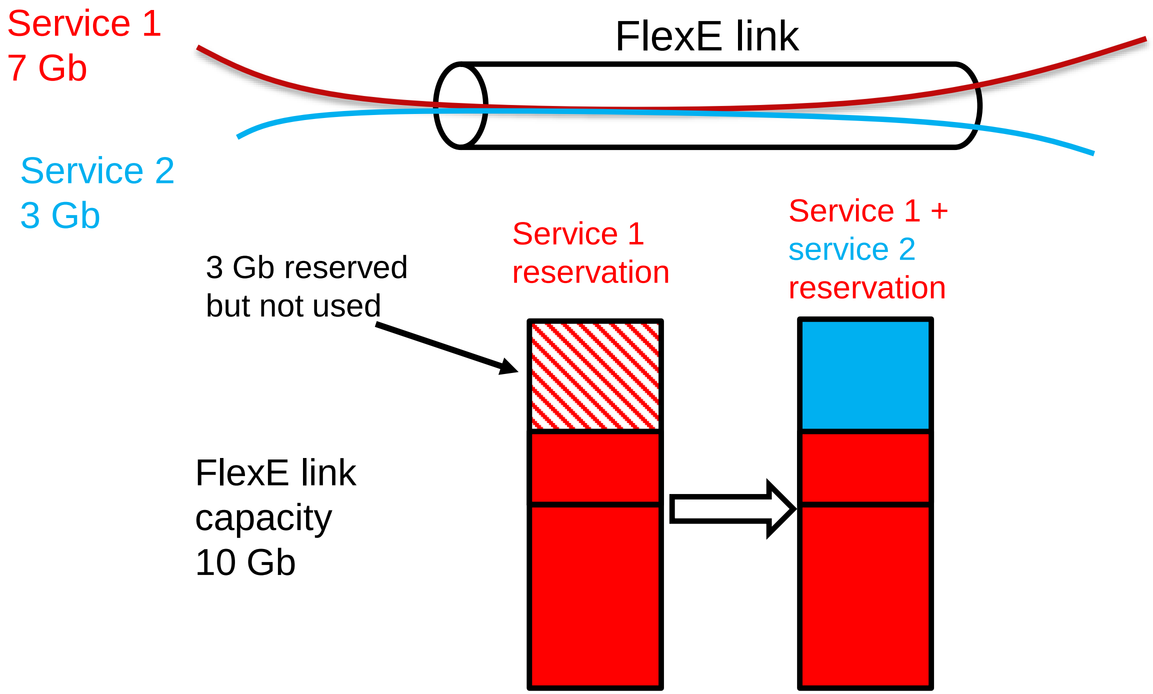
The traffic in one slice must not interfere with the performance of other slices



Slicing technologies: several technologies can be used to ensure isolation inside the data plane (FlexE, channelized sub-interfaces, HQoS)

Bandwidth reservation

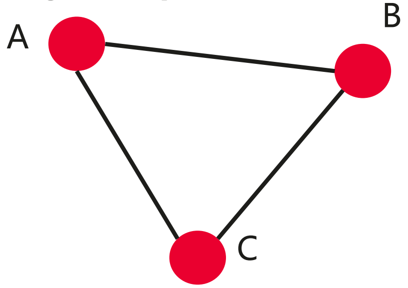
Bandwidth reservations in FlexE: capacity slots are allocated with a granularity of 5GB (the 5 first slots are of 1GB)



Need to solve joint routing and slot allocation problems ⇨ NP-hard problems

Challenge #2: Consider various Traffic Models - "Pipes" / "Hoses"

Running example:



Morning

	A	B	C
A		5	10
B	5		1
C	10	1	

Noon

	A	B	C
A		20	5
B	20		1
C	5	1	

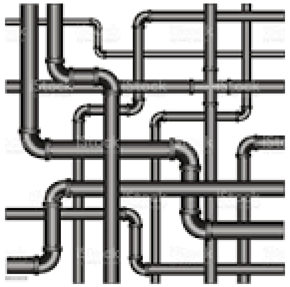
Afternoon

	A	B	C
A		10	1
B	10		10
C	1	10	

Per-Service (« Pipe » model with OD tunnels)

Aggregated « Pipe » model
(Maximum traffic over the day)

	A	B	C
A		20	10
B	20		10
C	10	10	



Per-Access Point ("VPN/Hose" model)

« VPN/Hose » model
(Access bandwidth for sites)

	Hose
A	25
B	21
C	11

Focus of this talk

Benefits:
Sometimes, traffic information is only available per access site.
It is also easier to define by users and it "compresses" inputs.
Users don't want to over-specify.

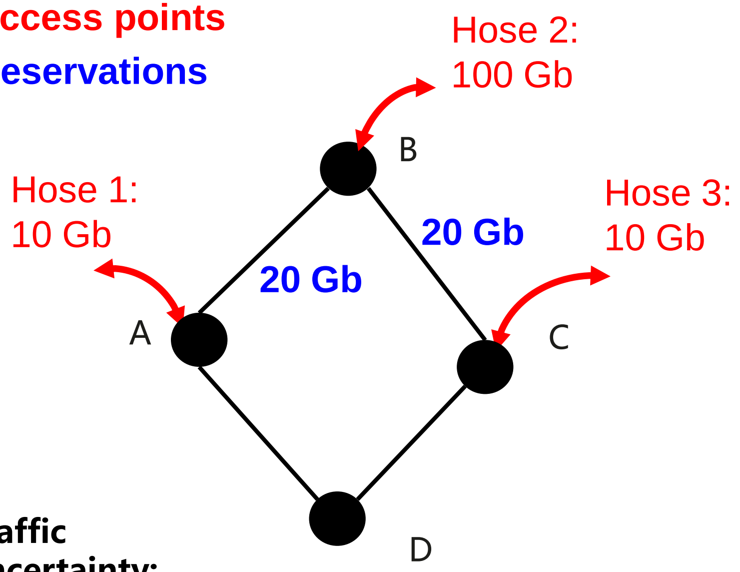
Main challenge: guarantee "non-blocking" slices (ensure that 100% of traffic matrices can be routed)
⇨ we need to solve a robust network design problem
Note: Traditional business VPNs do not have such guarantee today.

Possible specifications of traffic with Hoses

All-to-all directions

Access points

Reservations

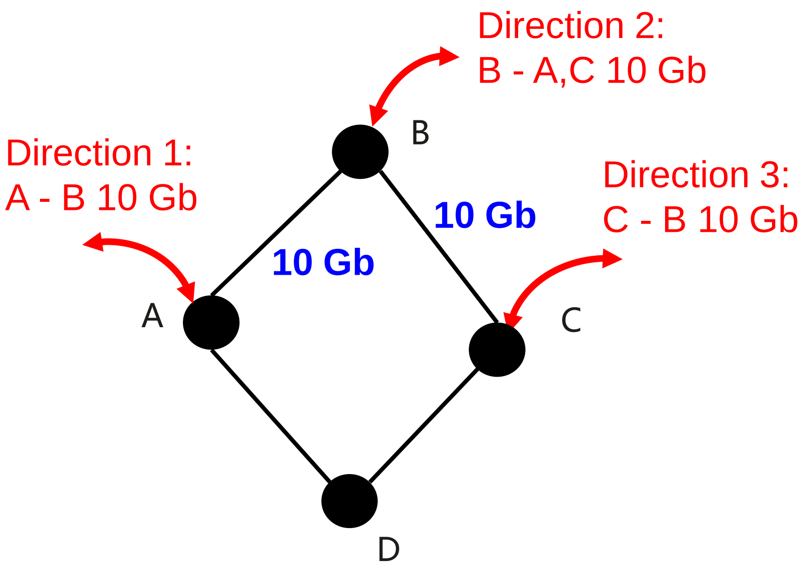


Traffic uncertainty: Demand vector set D is a polytope described by inequalities

$$\begin{cases} d_{(A,B)} + d_{(A,C)} \leq 10\ GB \\ d_{(C,B)} + d_{(C,A)} \leq 10\ GB \\ d_{(B,A)} + d_{(B,C)} \leq 100\ GB \end{cases}$$

Traffic can go in all directions
⇨ costly in terms of reservations

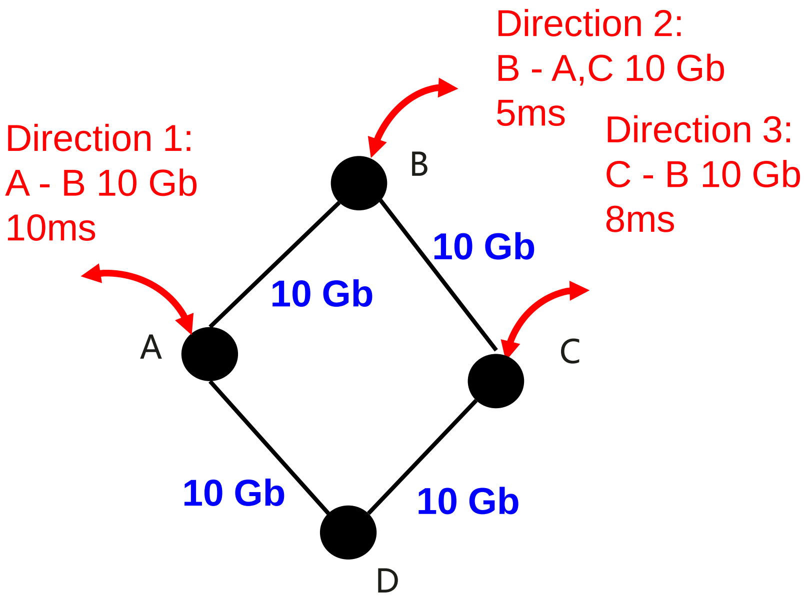
Specific directions



$$d_{(A,B)} \leq 10\ GB$$
$$d_{(C,B)} \leq 10\ GB$$
$$d_{(B,A)} + d_{(B,C)} \leq 100\ GB$$

Some directions may not be necessary
⇨ specify them to better utilize capacity

Directions + QoS



$$d_{(A,B)} \leq 10\ GB$$
$$d_{(C,B)} \leq 10\ GB$$
$$d_{(B,A)} + d_{(B,C)} \leq 100\ GB$$

Applications care about end-to-end QoS
⇨ add routing constraints to directions

Challenge #3: Optimize at large-scale

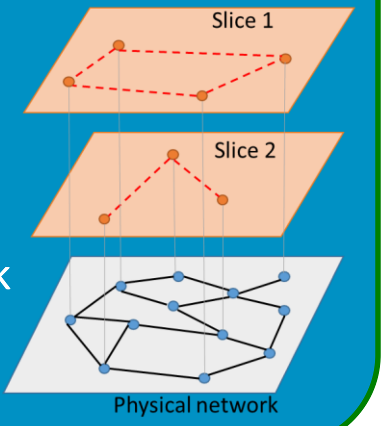
Inputs

- Service requirements:
  - Bandwidth (Hose, Pipes)
  - SLA (Ex: max E2E latency)
  - Protection (Ex: 1-link failures)
- Physical topology
- Current network conditions

Slicing Engine (NCE-IP)

Outputs

- Reservations for sub-interfaces (hard slicing)
- Routing policies in the physical network (optional)



Optimization intents

- Maximize traffic acceptance
- Minimize the reserved capacity
- Load balance link utilization
- …

Optimization challenges

- Handle large-scale networks (50k nodes in IPRAN)
- Get good solutions in a fast manner

Optimization tools

- Use advanced math-heuristics based on combinatorial optimization to solve difficult path computation and resource allocation problems

Over the full life-cycle ⇨ planning, adjustment, optimization

Robust network design:
Minimum cost problem

- Input: cost vector $\lambda \in \mathbb{R}_+^E$
- Decisions: capacities $x \in \mathbb{R}_+^E$ such that there is a routing scheme that can route all demands $d \in \mathcal{D}$ without exceeding the capacities (i.e. with a congestion ≤ 1), we denote the set of feasible capacities:
  - X_{dyn} for a dynamic routing
  - X_{sta} for a static routing
- The minimum costs problems for static and dynamic routing are:
  - $lin_{dyn} = \min_{x \in X_{dyn}} \sum_{e \in E} \lambda_e x_e$
  - $lin_{sta} = \min_{x \in X_{sta}} \sum_{e \in E} \lambda_e x_e$

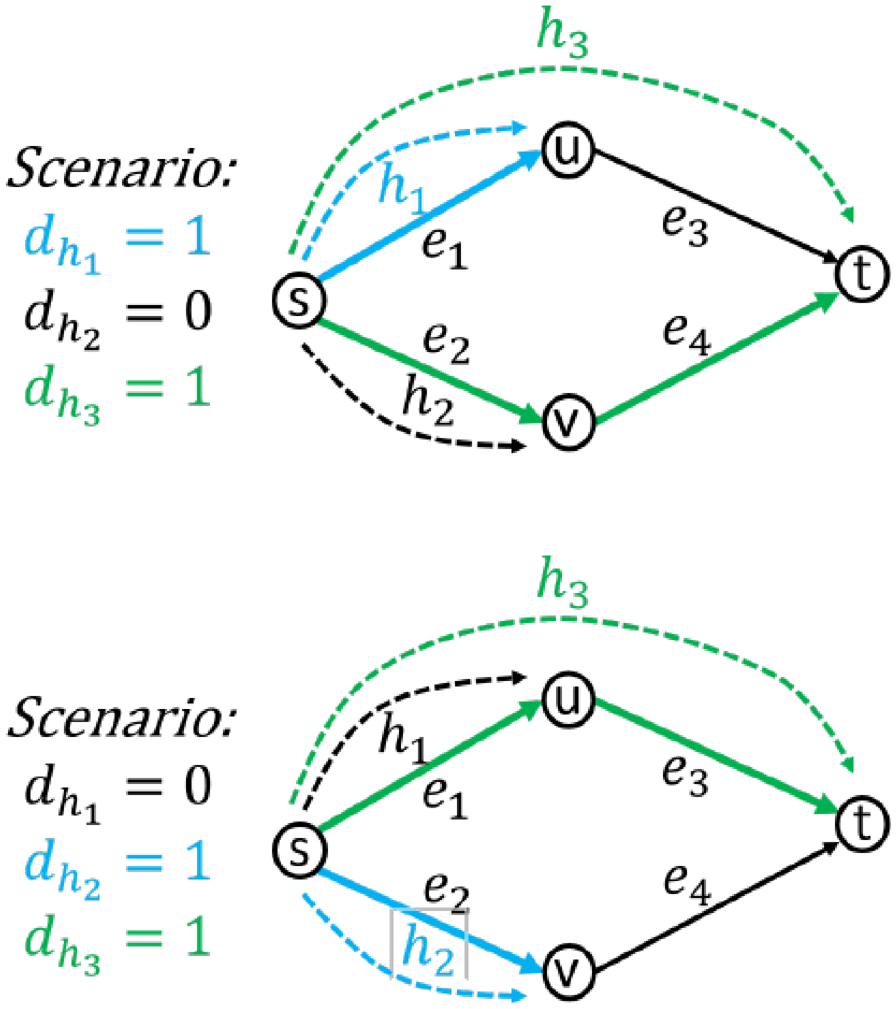
Robust network design variants: Example for Dynamic routing

Minimum cost problem

- Input costs of 1 on all edges: $\lambda_{e_1} = \lambda_{e_2} = 1,\ \lambda_{e_3} = \lambda_{e_4} = 0$
- Input demand polytope $\mathscr{D}$

$$d_{h_1} + d_{h_2} \leq 1$$
$$d_{h_3} \leq 1$$
$$d_h \geq 0, \forall h \in \mathscr{H}$$

- $lin_{dyn}(\mathscr{D}) = 2$



Remark: fractional routing is considered

Robust network design variants: Example for Static routing

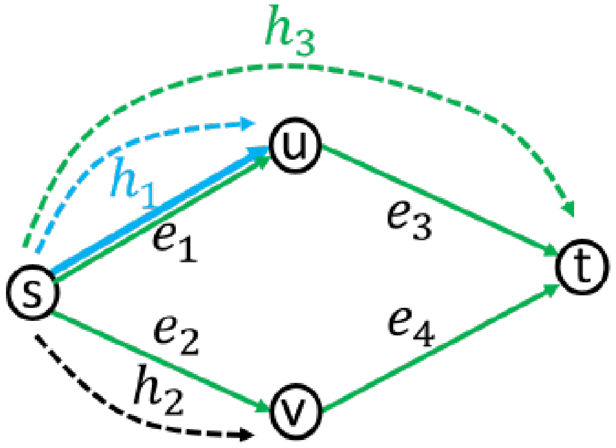
Minimum cost problem

- Input costs of 1 on all edges: $\lambda_{e_1} = \lambda_{e_2} = 1,\ \lambda_{e_3} = \lambda_{e_4} = 0$
- Input demand polytope $\mathscr{D}$

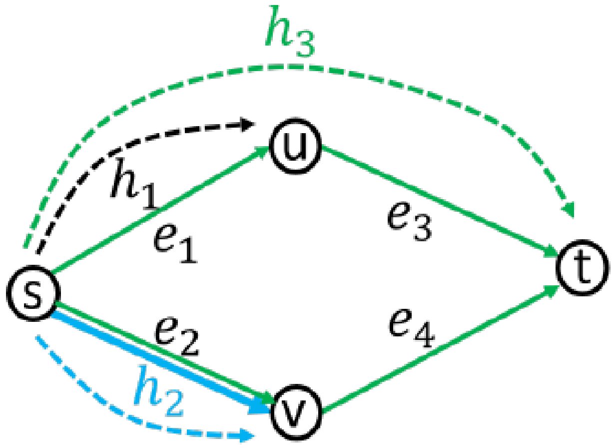
$$d_{h_1} + d_{h_2} \leq 1$$
$$d_{h_3} \leq 1$$
$$d_h \geq 0, \forall h \in \mathscr{H}$$

- The optimal static solution is to route half the demand h_3 on the path (e_1, e_3) and the other half on the path (e_2, e_4).
- $lin_{sta}(\mathscr{D}) = 3$

Scenario:
$d_{h_1} = 1$
$d_{h_2} = 0$
$d_{h_3} = 1$



Scenario:
$d_{h_1} = 0$
$d_{h_2} = 1$
$d_{h_3} = 1$



Remark: fractional routing is considered

HUAWEI

Robust network design: Complexity results

- **Dynamic routing**
  - Splittable case (Fractional routing)
    - CoNP-hard for directed graph (*Hardness of Robust Network Design, Chekuri, Shepherd, Oriolo and Scutellá, 2007*)

- **Static routing**
  - Splittable case (Fractional routing)
    - Polynomial time solvable (*Routing of Uncertain Traffic Demands, Ben-Ameur and Kerivin, 2005*)
  - Unsplittable case (Single path routing)
    - Without capacity: polynomial time solvable (*The VPN Conjecture Is True, Goyal, Olver and Sheperd, 2013*)
    - With capacity: NP-hard (*Provisioning a Virtual Private Network: A Network Design Problem for Multicommodity Flow, Gupta, Kleinberg, Kumar, Rastogi and Yener, 2001*)

Single path routing

Robust slicing model

- Graph $G = (V, E)$
- Access points $Q \subseteq V$
- Ingress/egress bandwidth m_v^{in}, m_v^{out} for each access point v
- Convergence ratio μ_e for each edge $e \in E$
- Bandwidth c_e of edge $e \in E$
- Size configuration s_i^e on edge $e \in E$
- Demands
  - $\mathcal{H}$: pairs of access points able to communicate
  - $\mathcal{D}$: possible demands for $\mathcal{H}$
    - Subset of $\{d \in \mathbb{R}^{\mathcal{H}}: \forall v \in V, \sum_{(v,u)\in\mathcal{H}} d_{(v,u)} \leq$

$$\min \sum_{e\in E} \sum_{i} \lambda^e s_i^e x_i^e$$

$$\sum_{p\in P_h} \varphi_h^p \geq 1, \forall h \in \mathcal{H}$$

Flow constraint

$$\mu_e \sum_{h\in\mathcal{H}} \sum_{p\in P_h : e\in p} d_h \varphi_h^p \leq \sum_i s_i^e x_i^e, \forall e \in E, \forall d \in \mathcal{D}$$

$$d_h^* \times \sum_{p\in P_h : e\in p} \varphi_h^p \leq \sum_i s_i^e x_i^e, \forall e \in E, \forall h \in \mathcal{H}$$

Slot reservation constraint

$$\sum_i s_i^e x_i^e \leq c_e, \forall e \in E$$

Capacity constraints

$$x_{i-1}^e \geq x_i^e, \forall e \in E, \forall i$$

Remark: extended formulation with exponential number of paths and infinite demand constraints

$$d_h^* = \operatorname{argmax}_{d\in\mathcal{D}} d_h$$

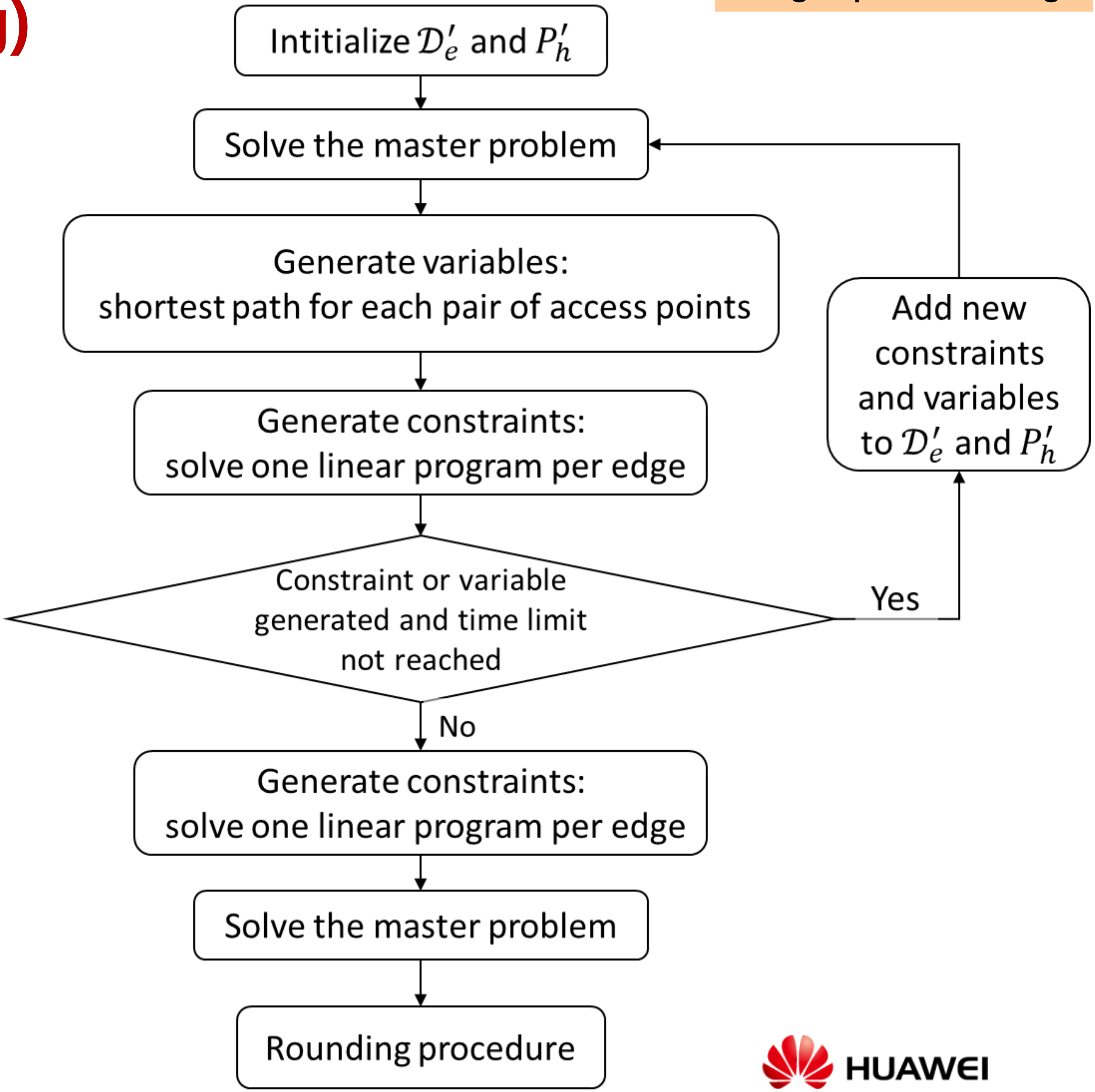
Solving algorithm (static routing)

Single path routing

Master problem

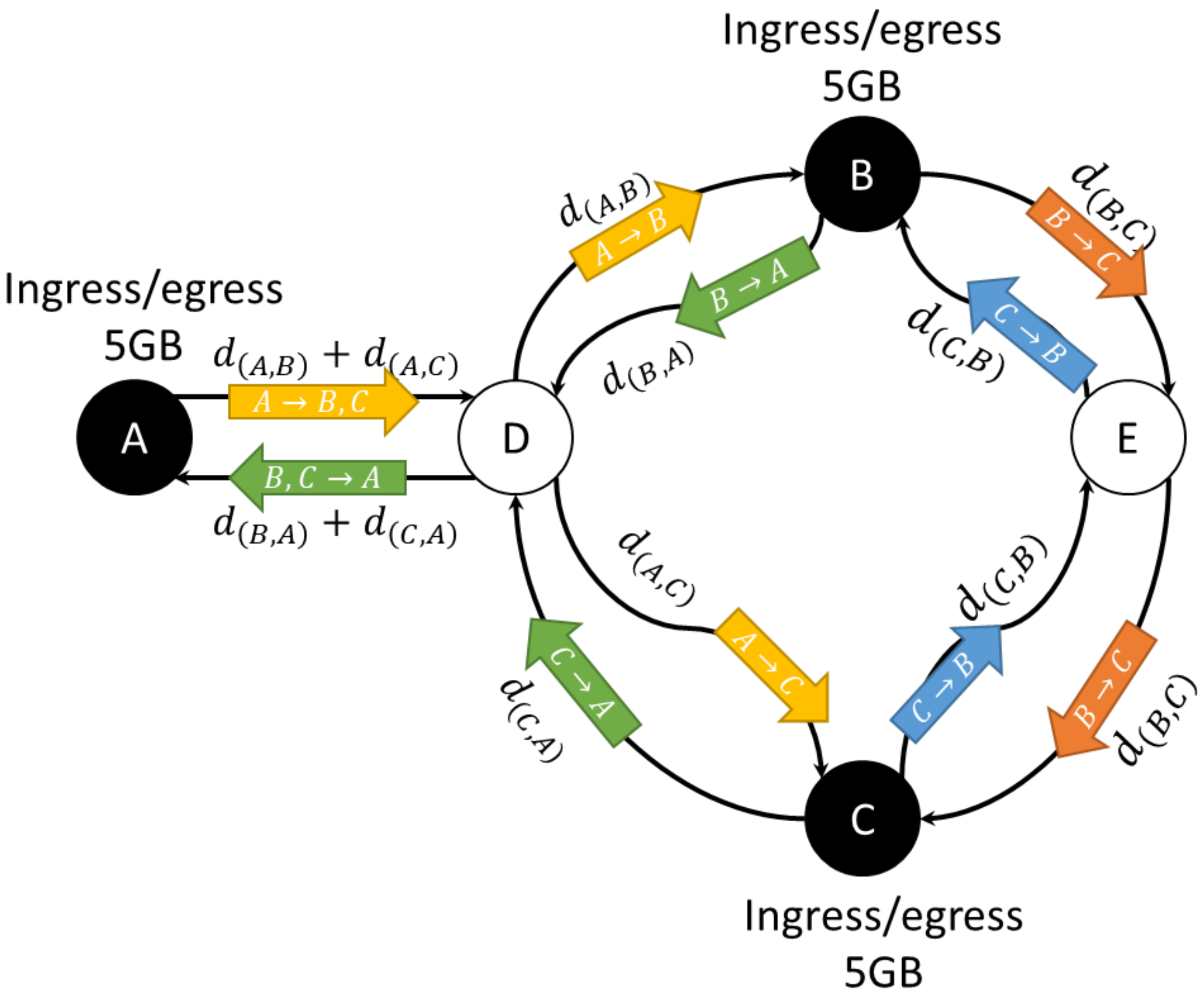
$$\min \sum_{e \in E} \sum_{i} s_i^e x_i^e$$

$$\begin{cases} \sum_{p \in P_h'} \varphi_h^p \geq 1, \forall h \in \mathcal{H} \\ \mu_e \sum_{h \in \mathcal{H}} \sum_{p \in P_h' : e \in p} d_h \varphi_h^p \leq \sum_{i} s_i^e x_i^e, \forall e \in E, \forall d \in \mathcal{D}_e' \\ \sum_{i} s_i^e x_i^e \leq c_e, \forall e \in E \\ x_{i-1}^e \geq x_i^e, \forall e \in E, \forall i \end{cases}$$



HUAWEI

Illustrating exemple for the capacity design



$$\max \sum_{h \in H_e} d_h$$

H_e: set of demands that traverse edge e

$$\begin{cases} \sum_{(q,t)\in\mathcal{H}} d_{(q,t)} \le m_q^{out}, q \in Q \\ \sum_{(s,q)\in\mathcal{H}} d_{(s,q)} \le m_q^{in}, q \in Q \end{cases}$$

Capacity for edge (A, D)

Max $d_{(A,B)} + d_{(A,C)}$

A egress constraint: $d_{(A,B)} + d_{(A,C)} \le 5GB$

...

$= 5GB$

Capacity for edge (B, E)

Max $d_{(B,C)}$

B egress constraint: $d_{(B,A)} + d_{(B,C)} \le 5GB$

C ingress constraint: $d_{(A,C)} + d_{(B,C)} \le 5GB$

...

$= 5GB$

HUAWEI

Results for IPRAN networks

Network topologies

- Small: 90 nodes, 200 links, 30 hoses
- Middle: 5.5 nodes, 11k links, 100 hoses
- Large: 55k nodes, 120k links, 100 hoses (approximatively 10000 demands)

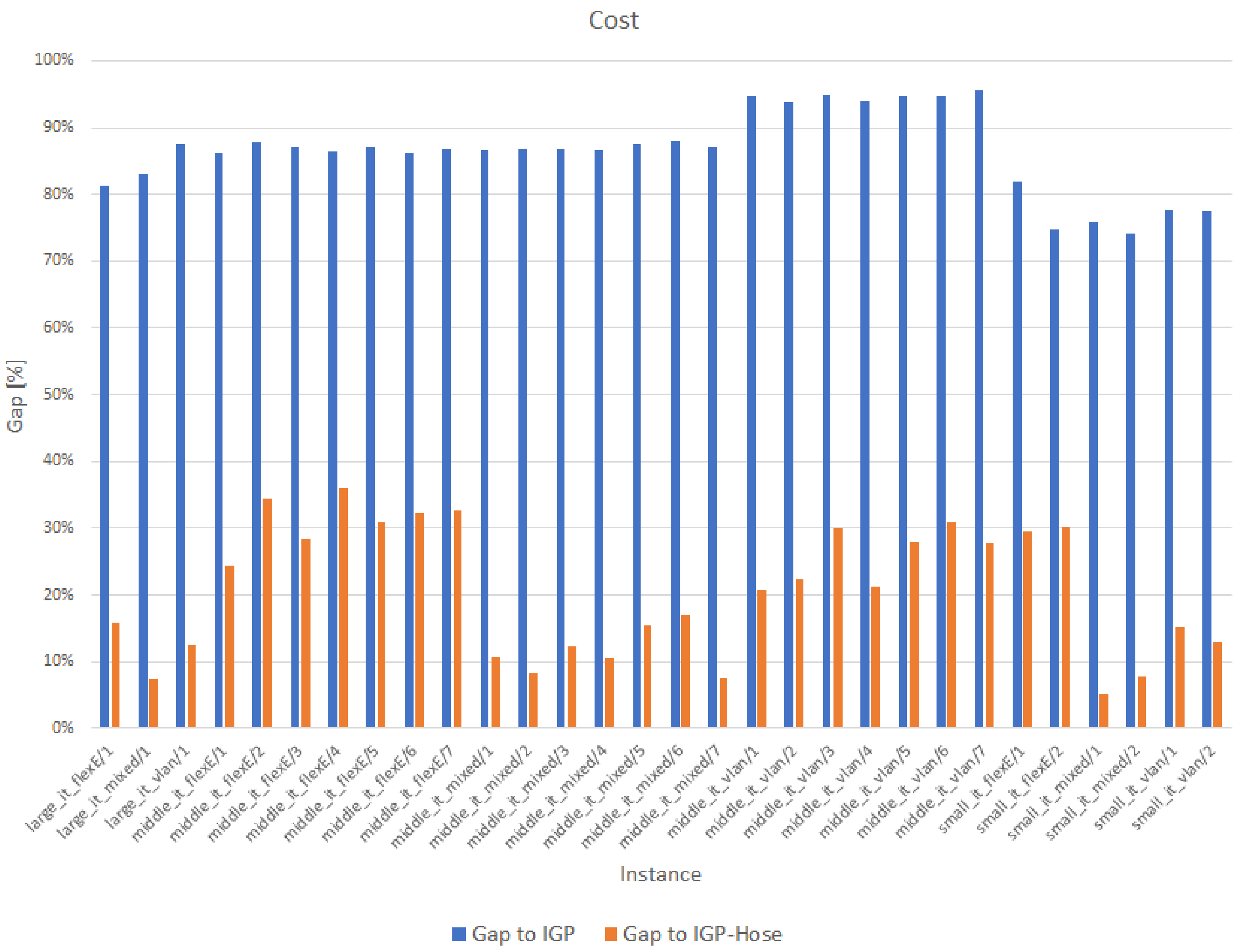
Instances with FlexE, Channelized interfaces and mixed.

Benchmarks

1. IGP shortest paths with worst case traffic in every directions **(IGP)**
2. IGP shortest paths with optimal reservations using the traffic polytope **(IGP-Hose)**

Time limits

- Small: 30s
- Middle & Large: 600s



Main results:
21% average gain over IGP-Hose and 86% over IGP

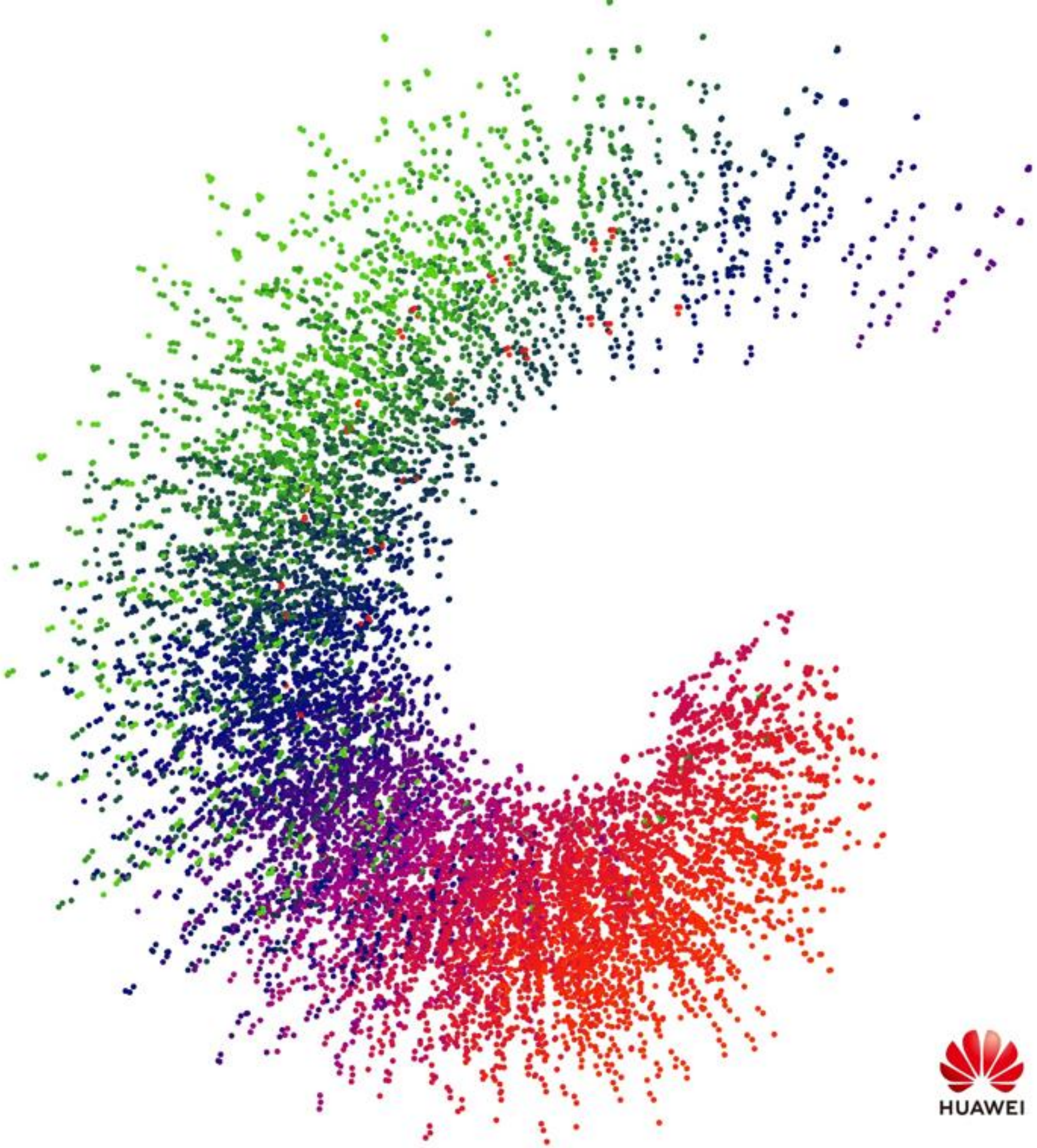
HUAWEI

Conclusions

- **Robust network slicing**
  - Slicing with simplified traffic information (hoses) calls for advanced algorithms to guarantee TMs can be routed
- **Future challenges**
  - **Data-driven approaches**
    - Optimize reservations based on traffic predictions (dimensioning of hoses, identify relevant directions)
  - **Advanced scenarios**
    - Multi-domain network slicing (different technologies & granularities)
    - Hierarchical network slicing

Joint work with:
Paolo Medagliani, Jérémie Leguay, Sebastien Martin,
Caishengming, Zengfeng, Yacine Al Najjar, Walid Ben Ameur

Thanks



Julien Lesca

Senior researcher, Network OR, Datacom Dijkstra Lab

HUAWEI

Column & Constraint generation

Pricing problem

- Variables φ_h^p
- α_h, β_d^e and $\gamma_{d,h}^e$ simplex multipliers of the 1^{st}, 2^{nd} and 3^{rd} constraints
- For any $h \in H$, find path $p^* \in P_h$ which minimizes $\sum_{e \in p^*}[\gamma_h^e + \mu_e \sum_{d \in \mathcal{D}'} \beta_d^e]$
  - Compute shortest path on G where the cost on edge e is $[\gamma_h^e + \mu_e \sum_{d \in \mathcal{D}'} \beta_d^e]$
  - If $\sum_{e \in p^*}[\gamma_h^e + \mu_e \sum_{d \in \mathcal{D}'} \beta_d^e] \leq \alpha_h$ holds then add variable $\varphi_h^{p^*}$

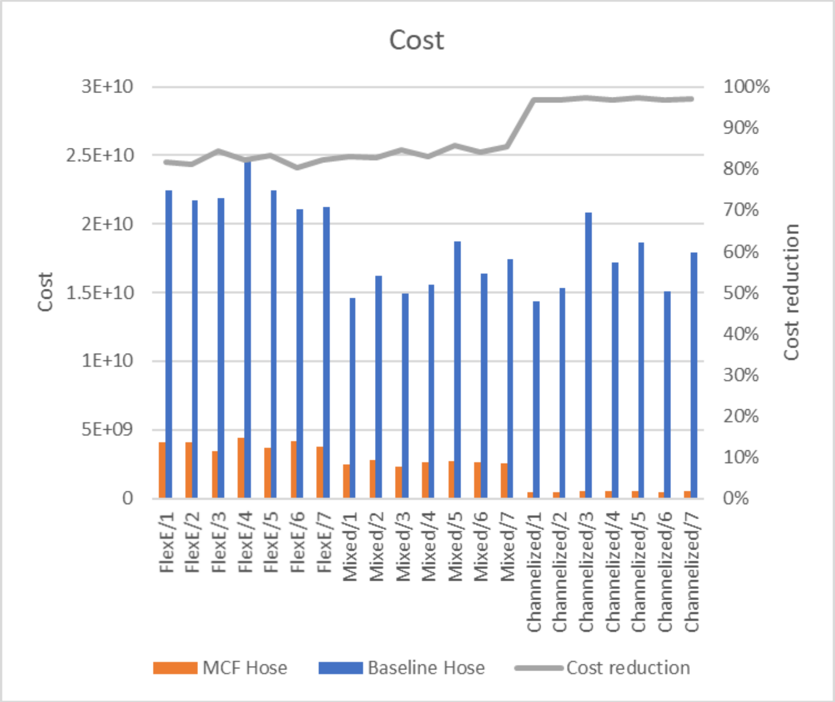
Cutting problem

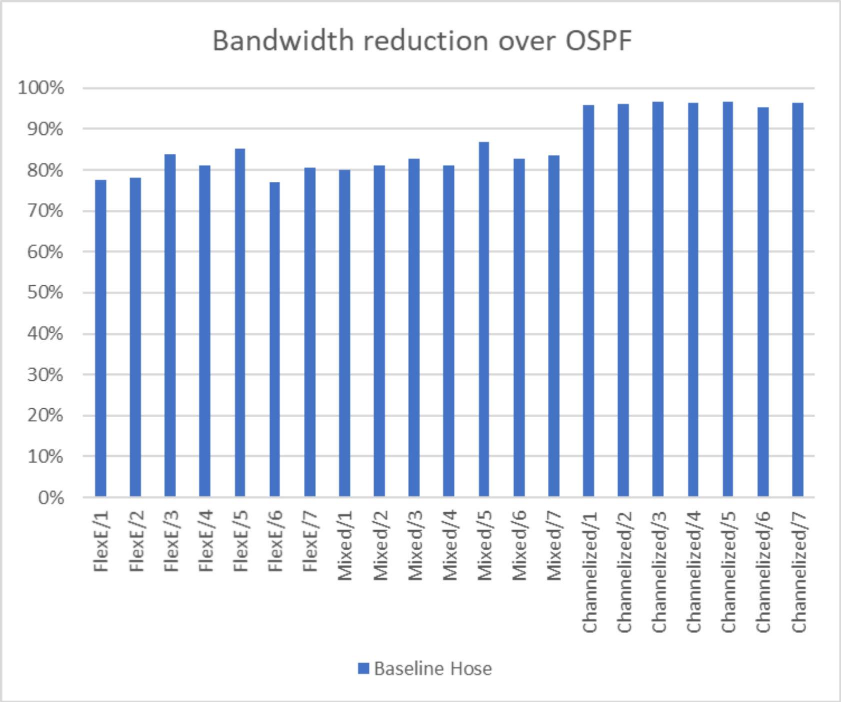
- Constraints $\mu_e \sum_{h \in H} \sum_{p \in P_h : e \in p} d_h \varphi_h^p \leq \sum_i s_i^e x_i^e$
- Solve the following linear programs ($\mathcal{D}$ is a set of linear constraints) $max_{d \in \mathcal{D}} \sum_{h \in H} d_h \sum_{p \in P_h : e \in p} \varphi_h^p$
- If $\mu_e OPT_{LP} \leq \sum_i s_i^e x_i^e$ then add constraint $\mu_e \sum_{h \in H} \sum_{p \in P_h : e \in p} d_h^* \varphi_h^p \leq \sum_i s_i^e x_i^e$

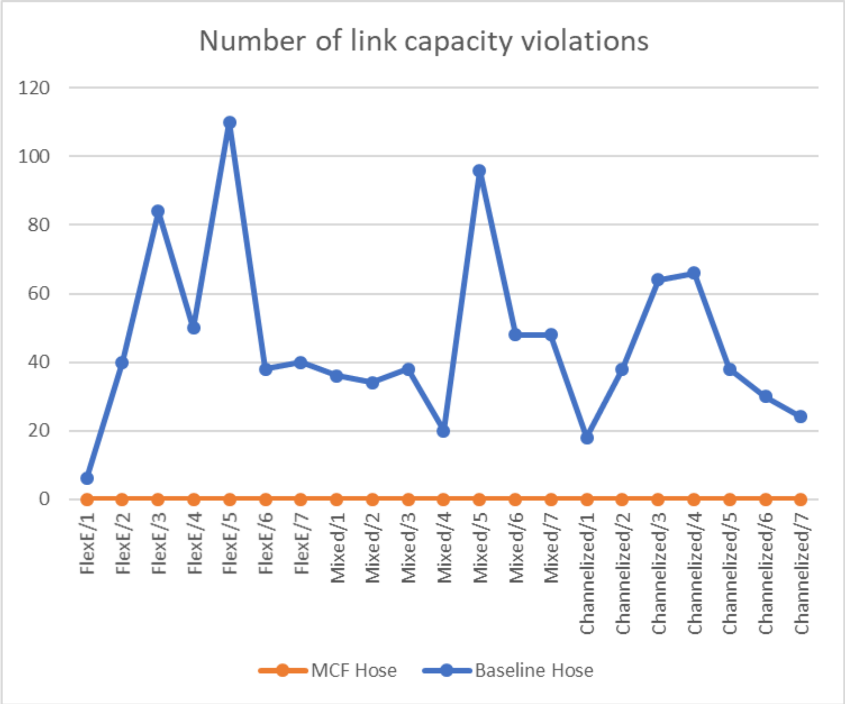
  where $d^* = \text{argm}ax_{d \in \mathcal{D}} \sum_{h \in H} d_h \sum_{p \in P_h : e \in p} \varphi_h^p$

HUAWEI

Results for IPRAN networks







Network topology
5500 nodes
12000 links
100 hoses (approximatively 10000 demands)

Baseline is OSPF routing

Reduction computed as $\frac{OSPF - MCFHose}{OSPF}$

The cost improvement of MCFHose over OSPF is between 80% and 97%

The bandwidth reduction is between 77% and 97%

While MCFHose finds a solution for all the hoses, the baseline presents some capacity violations (between 6 and 110 violated links) as it cannot routing traffic matrices

HUAWEI