DIEEI
Dipartimento di Ingegneria Elettrica, Elettronica e Informatica

TELECOM ITALIA

Università degli Studi di Catania

An Analytical Tool for Performance Evaluation of Software Defined Services

Alfio Lombardo
Vincenzo Riccobene
Giovanni Schembra

DIEEI – University of Catania

Antonio Manzalini

Telecom Italia
Strategy
Future Centre

Presentation outline

- **Paper motivation and reference scenario**
- **Network analytical framework**
  - Model of an NFV node
  - Model of a non-NFV node
  - Model of the whole network
  - Derivation of performance parameters
- **Case study**
- **Conclusions and future work**

Motivation

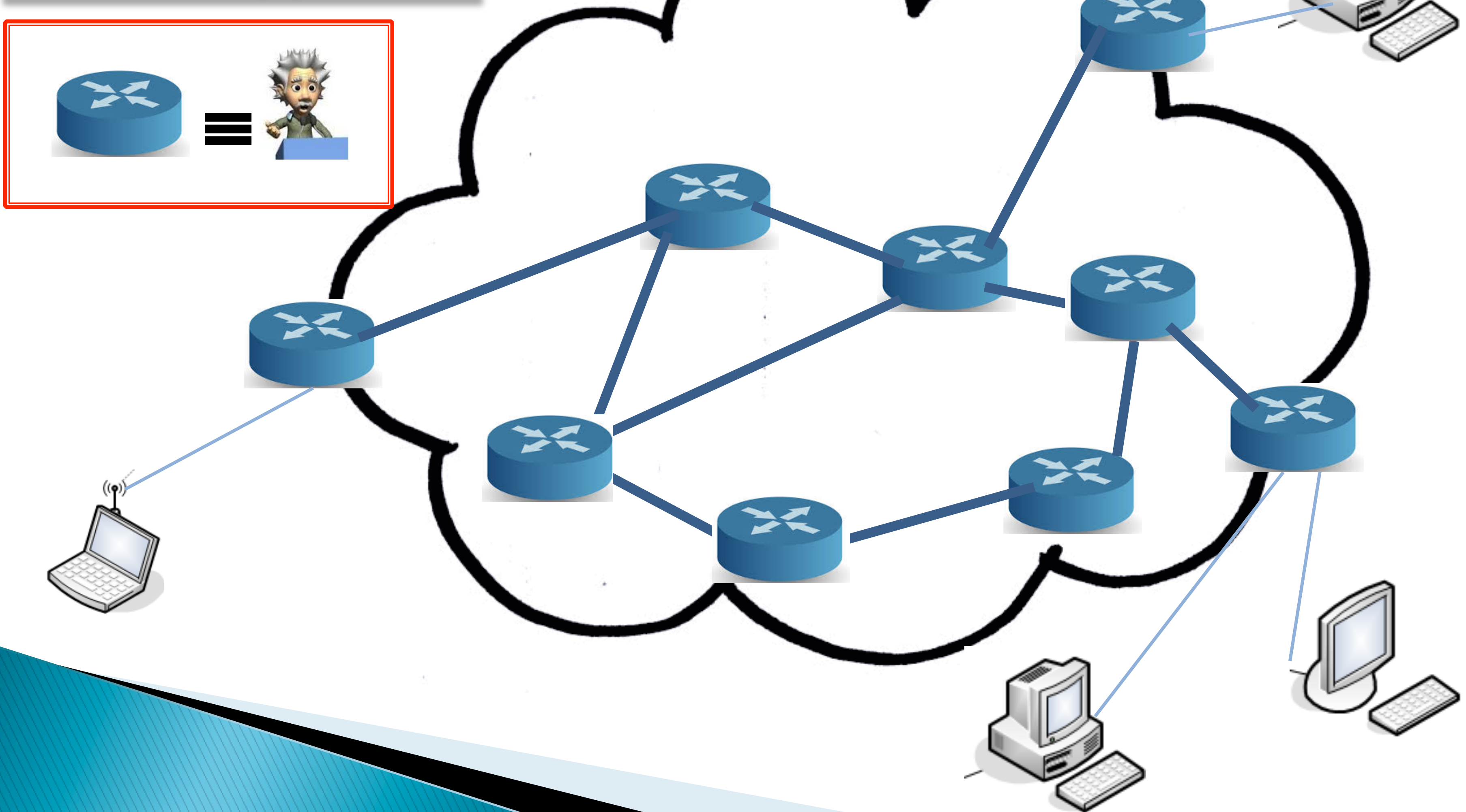
- **Service Providers and Network Operators need:**
  - Flexibility in network deployment and management
  - A flexible and optimal provisioning of network functions and services could reduce equipment costs and allow to postpone network investments
  - New network functionalities, services and policies to increase dynamicity of the market
  - Reducing OPEX and CAPEX

SDN & NFV

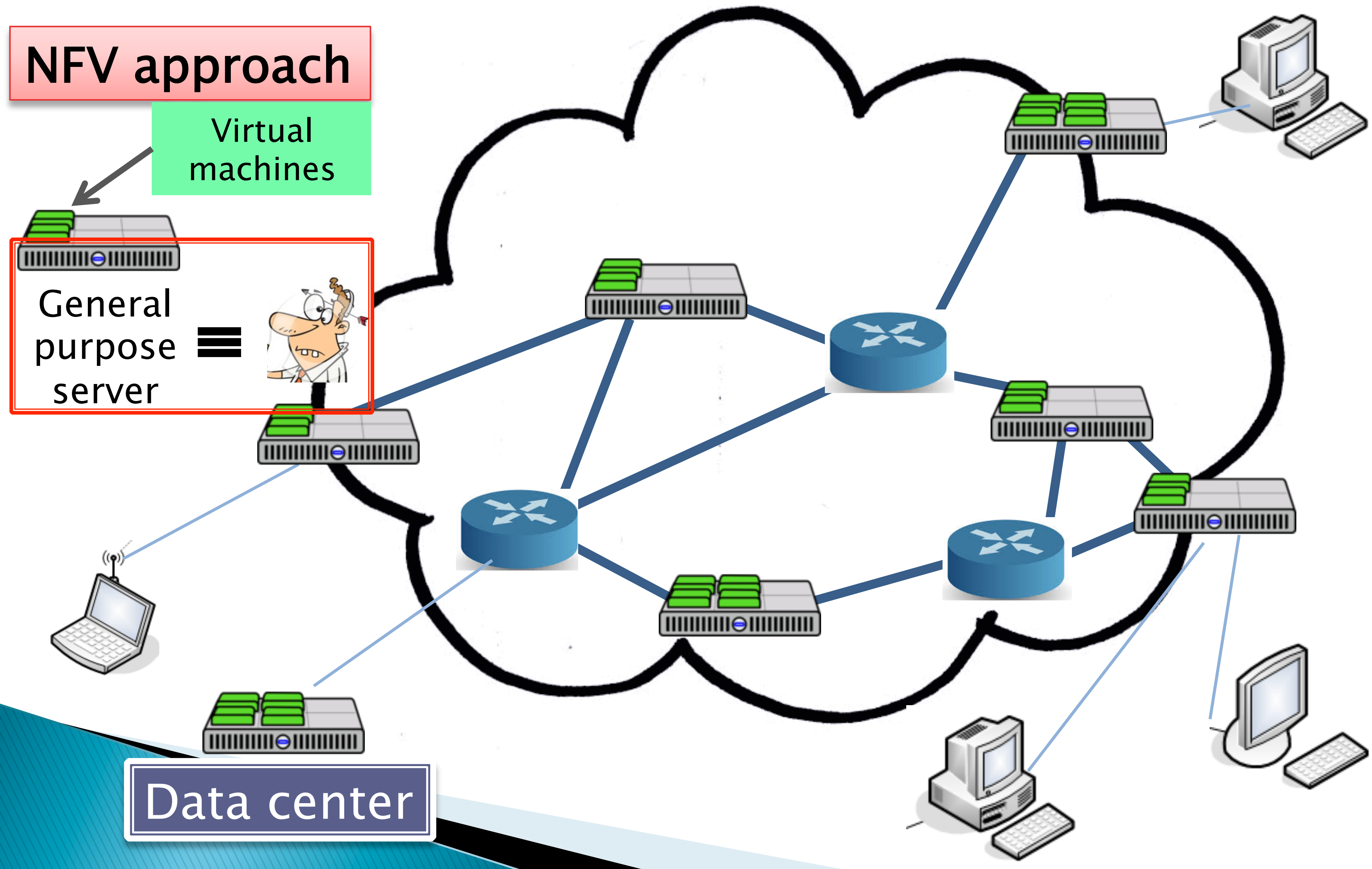
- **SDN: Software Defined Networks**
  - Decoupling the software control plane from the hardware data plane (packets forwarding), and moving its logic to centralized controllers
- **NFV: Network Function Virtualization**
  - Virtualization of some network functions that can run on standard HW, and that can be moved and instantiated in various locations of the network

Network scenario

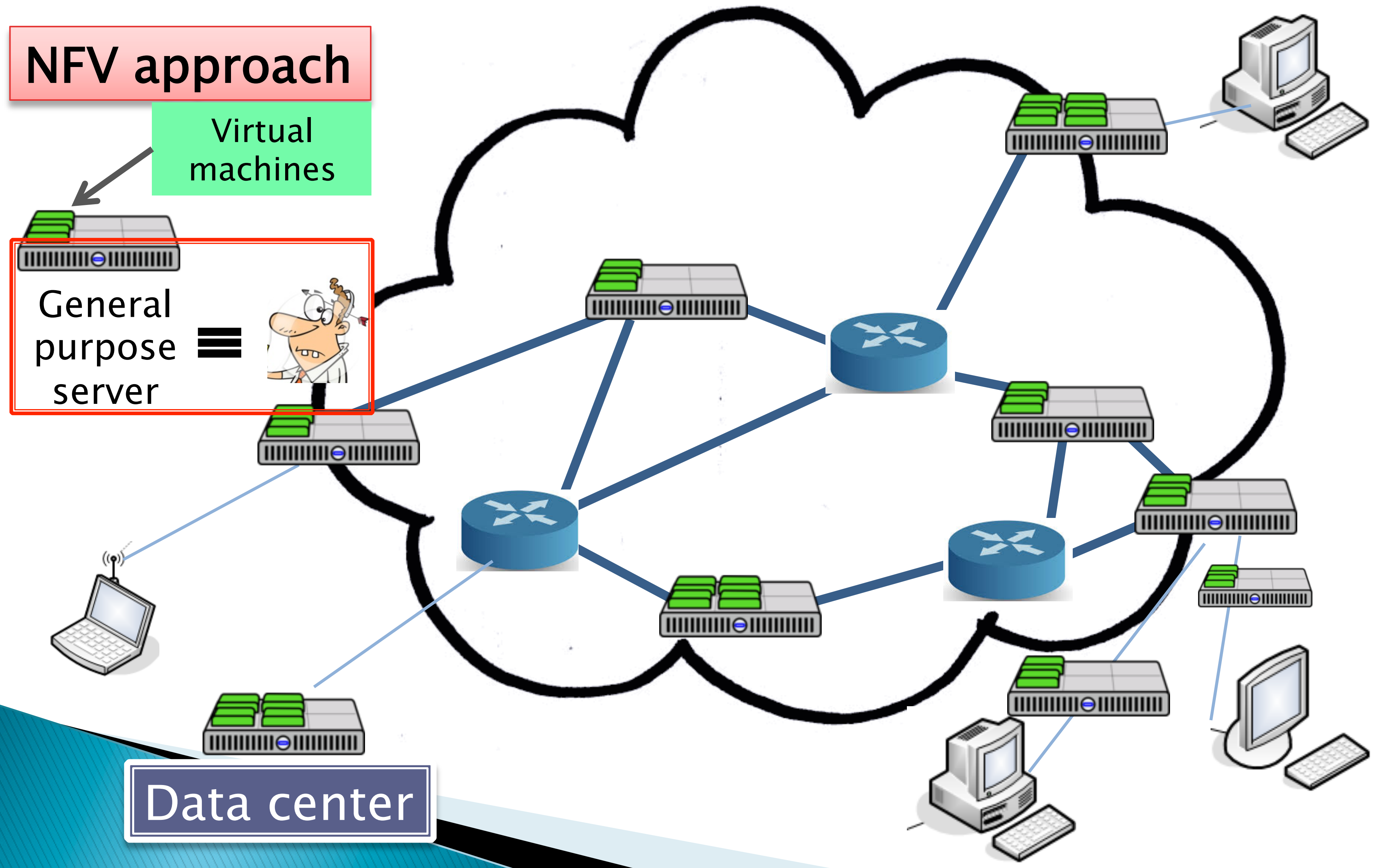
Current approach



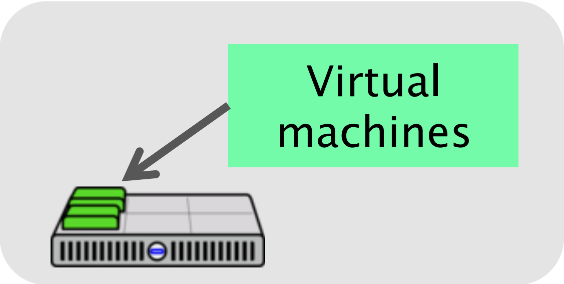
Network scenario



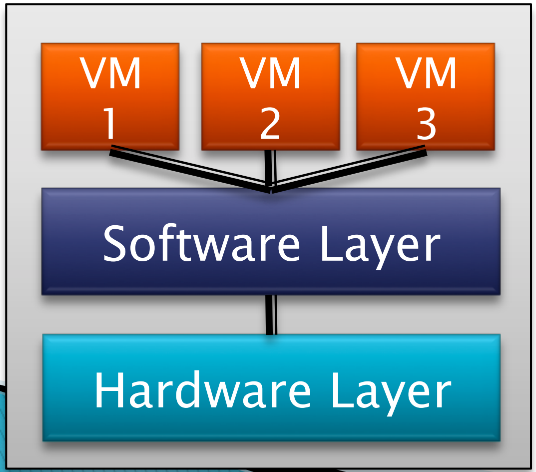
Network scenario



NFV Capabilities



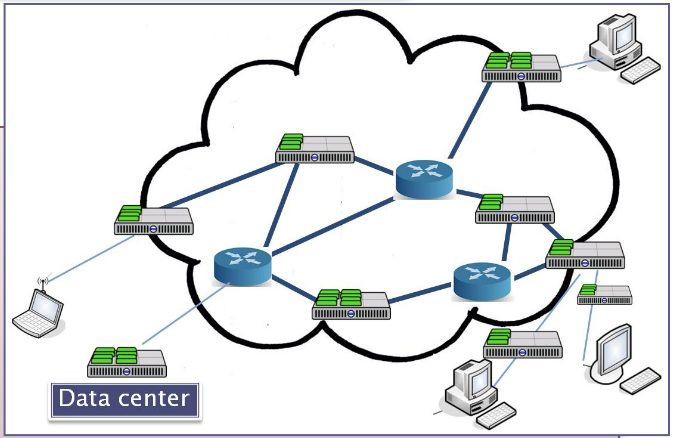
- An "NFV node" is characterized by:
  - A standard hardware architecture (x86 commodity hardware)
  - A virtualization capable software architecture
  - A set of Virtual Machines (VMs) that run Network Functions (e.g. Routers, Firewalls, Load Balancer, ...)



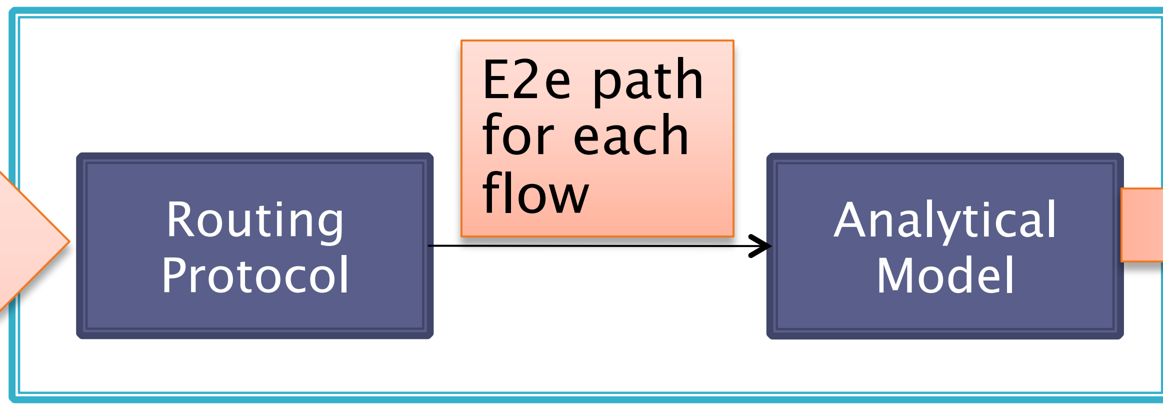
Main goal and paper target

- Analysis of the impact of the Network Function allocation

- An analytical framework for performance evaluation of the network



Analytical framework



- Network topology
- Network Function allocation
- Traffic characterization

- Performance parameters

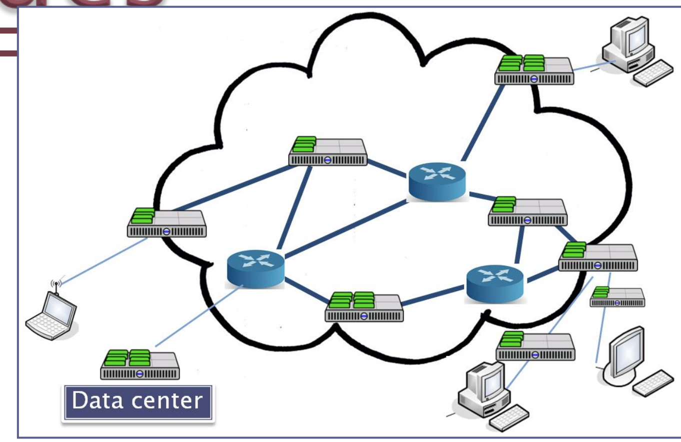
Network topology definition

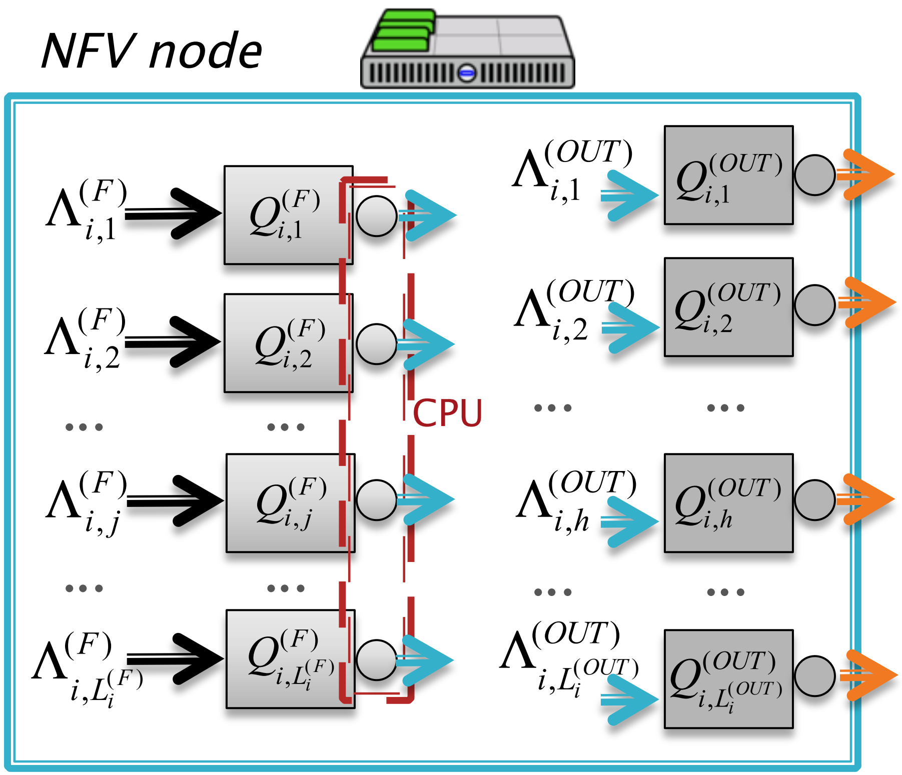
- Let us consider the network represented by a directed graph $G(V, E)$, where:
  - V is a set of vertices
  - E is a set of links among them
- Let F be the set of functions deployed over the network

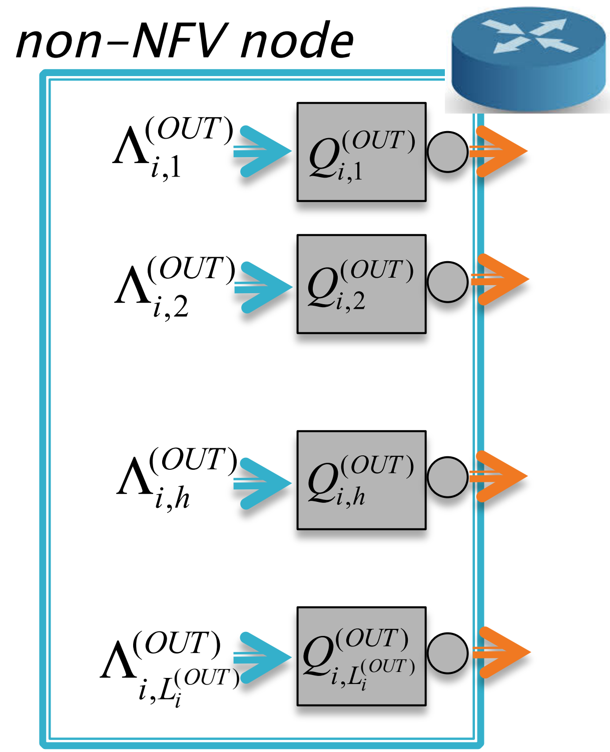
User traffic characterization

- **User traffic is represented by a set *S* of flows, each characterized by the following items:**
  - $\sigma_s \in$ V is the vertex that represents the source of the flow s
  - $\delta_s \in$ V is the vertex that represents the destination of the flow s
  - f_s is the mean bit rate characterizing the flow s
  - $func_s$ is the set of functions required by the flow s

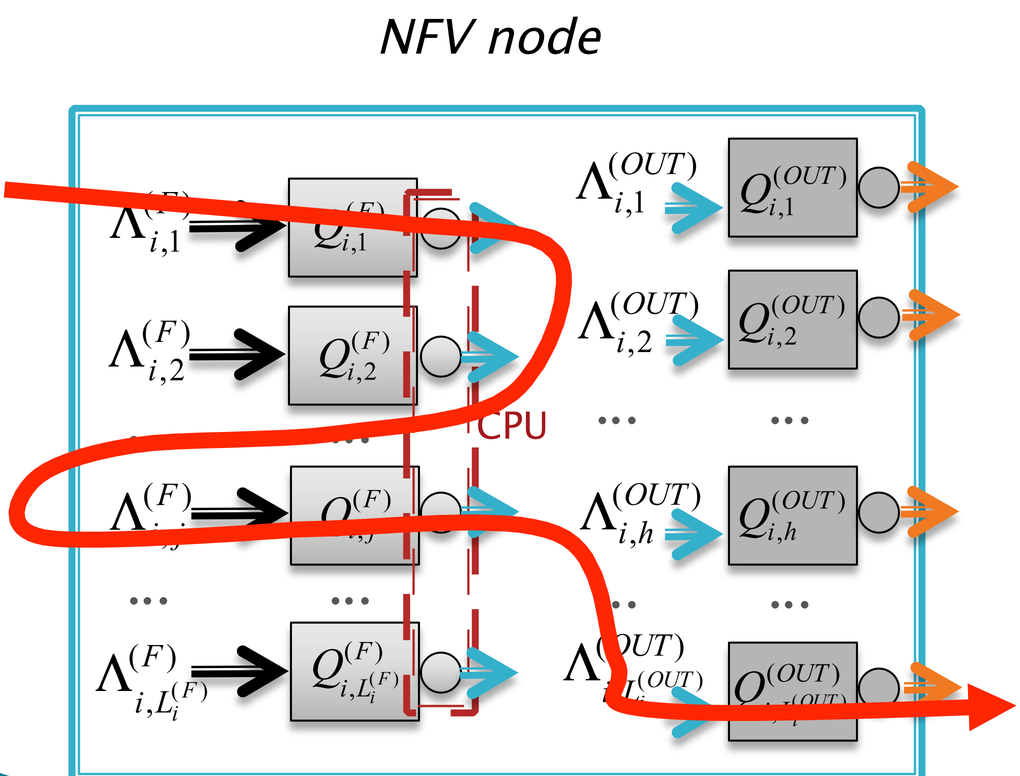
Model of Network nodes







Model of Network nodes

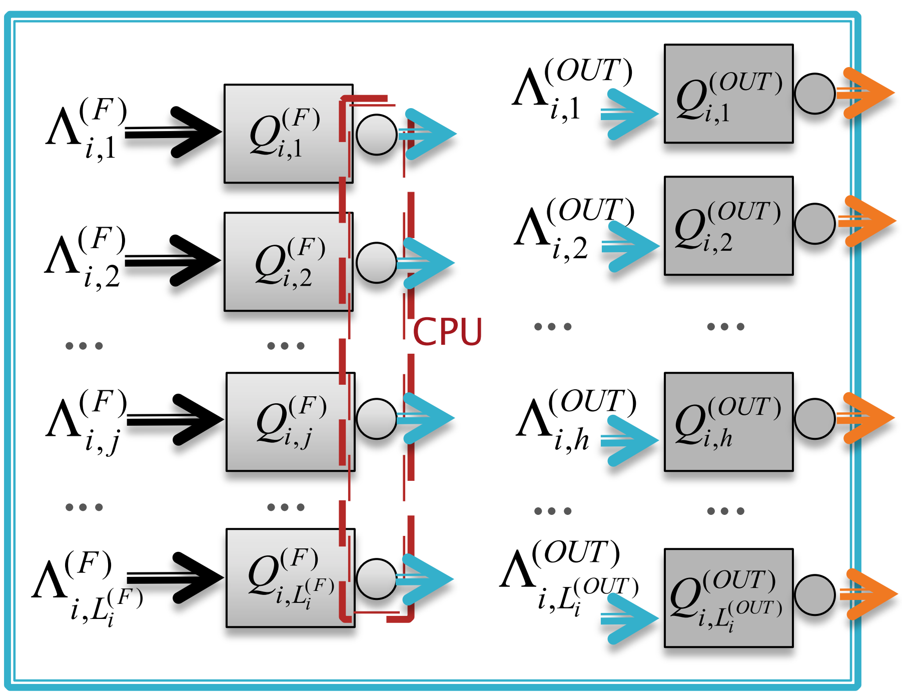


An NFV node can be modeled as a set of queues, that belong to two categories:

- ***Functions Queue*** $Q_{i,j}^{(F)}$
  - They manage the access to the functions
  - Their service rate depends on the CPU processing speed to process the relative function
- ***Output queues*** $Q_{i,h}^{(OUT)}$
  - They manage the packet transmission on the output links
  - Their service rate depends on the output bitrate

Model of Network nodes

NFV node



Function Queues

$$\Lambda_{i,j}^{(F)} = \sum_{\forall k \in \Phi_{i,j}} \lambda_k$$ *Arrival Rate*

$$\mu_{i,j}^{(F)} = p_{i,j} \cdot C_i^{(CPU)}$$ *Service Rate*

Output Queues

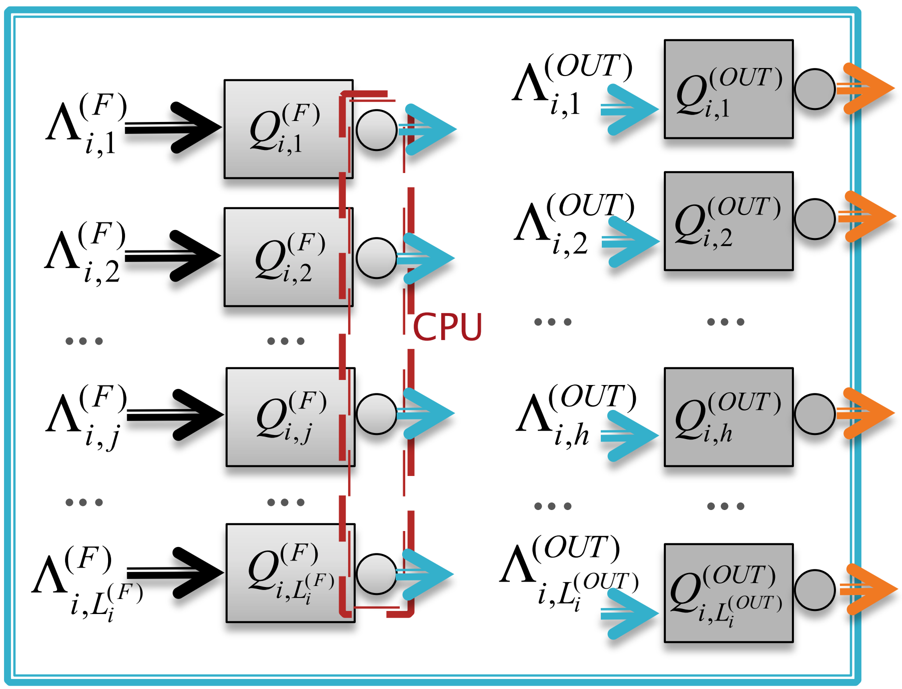
$$\Lambda_{i,h}^{(OUT)} = \sum_{\forall k \in \Psi_{i,h}} \lambda_k$$ *Arrival Rate*

$$\mu_{i,h}^{(OUT)} = C_{i,h}^{(NIC)}$$ *Service Rate*

Model of Network nodes

$\Phi_{i,j}$: set of flows routed through the node i and requiring the function j

NFV node



Function Queues

$$\Lambda_{i,j}^{(F)} = \sum_{\forall k \in \Phi_{i,j}} \lambda_k$$ *Arrival Rate*

$$\mu_{i,j}^{(F)} = p_{i,j} \cdot C_i^{(CPU)}$$ *Service Rate*

Output Queues

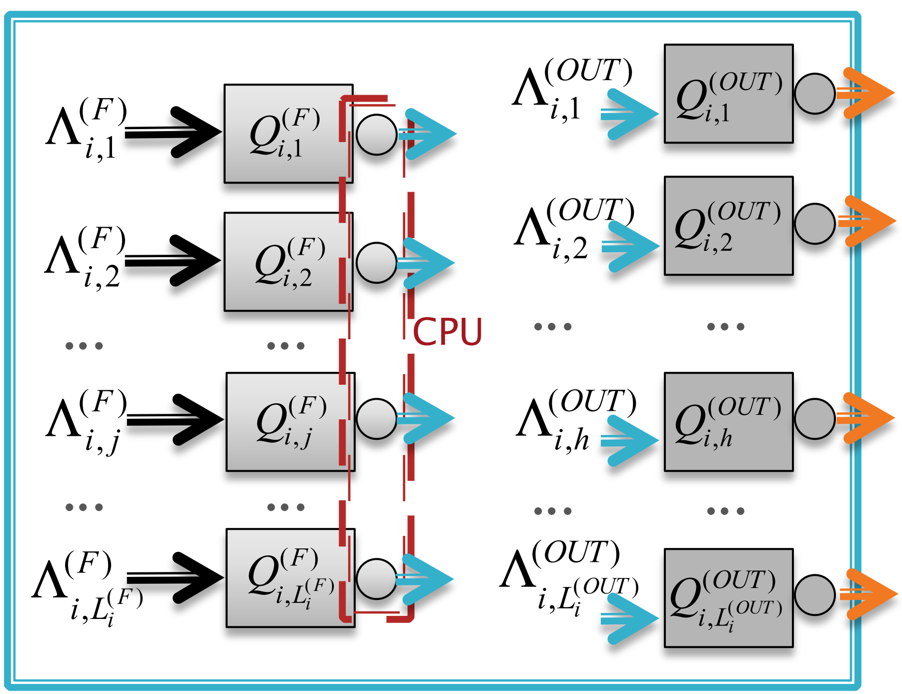
$$\Lambda_{i,h}^{(OUT)} = \sum_{\forall k \in \Psi_{i,h}} \lambda_k$$ *Arrival Rate*

$$\mu_{i,h}^{(OUT)} = C_{i,h}^{(NIC)}$$ *Service Rate*

Model of Network nodes

$p_{i,j}$: the CPU quota of i-th node assigned to VM (function) j

NFV node



Function Queues

$$\Lambda_{i,j}^{(F)} = \sum_{\forall k \in \Phi_{i,j}} \lambda_k$$ *Arrival Rate*

$$\mu_{i,j}^{(F)} = p_{i,j} \cdot C_i^{(CPU)}$$ *Service Rate*

$C_i^{(CPU)}$: the mean packet processing rate of the processor in the *i*-th NFV node

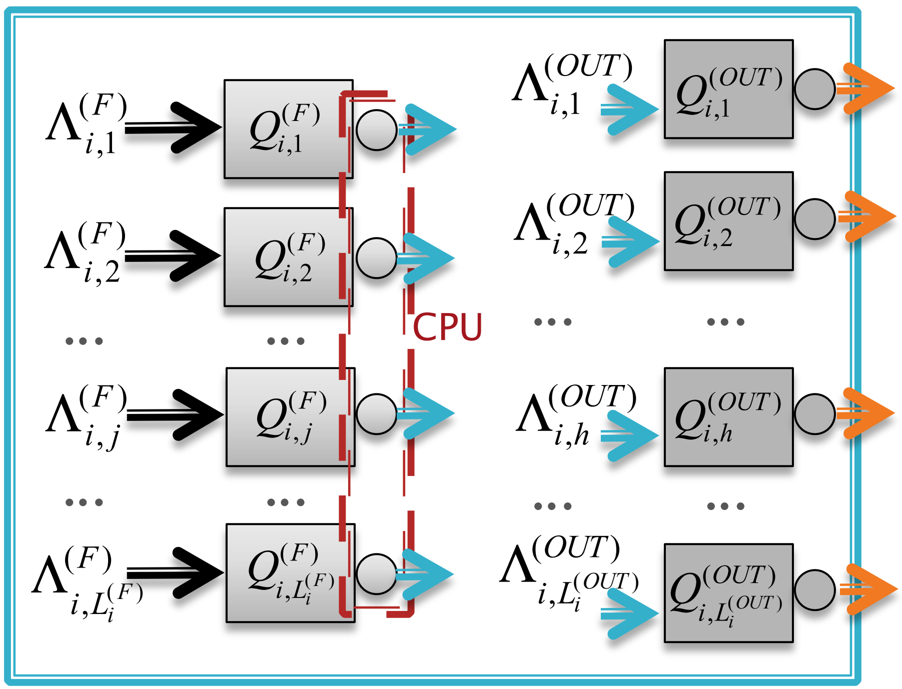
$\forall k \in \Psi_{i,h}$

$$\mu_{i,h}^{(OUT)} = C_{i,h}^{(NIC)}$$ *Service Rate*

Model of Network nodes

$\Psi_{i,h}$: the set of flows crossing the node i and leaving it through the NIC h

NFV node



Function Queues

$$\Lambda_{i,j}^{(F)} = \sum_{\forall k \in \Phi_{i,j}} \lambda_k$$ *Arrival Rate*

$\mu_{i,j}^{(F)} =$

$C_{i,h}^{(NIC)}$: the transmission rate of the h-th output link of the i-th NFV node

Output Queues

$$\Lambda_{i,h}^{(OUT)} = \sum_{\forall k \in \Psi_{i,h}} \lambda_k$$ *Arrival Rate*

$$\mu_{i,h}^{(OUT)} = C_{i,h}^{(NIC)}$$ *Service Rate*

Model of Network nodes

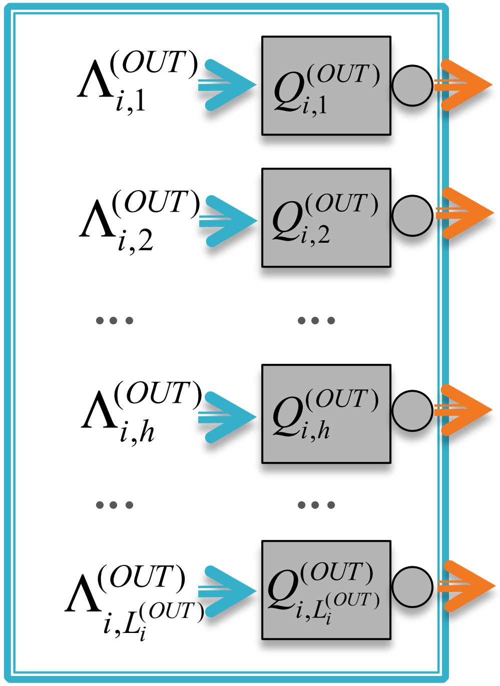
A non-NFV node can be modeled as a set of output queues, one for each output link

Output Queues

$$\Lambda_{i,h}^{(OUT)} = \sum_{\forall k \in \Psi_{i,h}} \lambda_k$$ *Arrival Rate*

$$\mu_{i,h}^{(OUT)} = C_{i,h}^{(NIC)}$$ *Service Rate*

non-NFV node



Markov model

- The whole network can be modeled as a network of queues
- Model definition: an *N*-dimensional continuous-time Markov chain whose state is defined as follows:

$$\boxed{S^{(\Sigma)}(t) = \left(\underline{S}_1(t), \ldots, \underline{S}_N(t)\right)}$$

where $\underline{S}_i(t)$ is equal to:

- $\underline{S}_i(t) = \left(S_{i,1}^{(F)}(t), \ldots, S_{i,L_i^{(F)}}^{(F)}(t), S_{i,1}^{(OUT)}(t), \ldots, S_{i,L_i^{(OUT)}}^{(OUT)}(t)\right)$ (NFV Node)
- $\underline{S}_i(t) = \left(S_{i,1}^{(OUT)}(t), \ldots, S_{i,L_i^{(OUT)}}^{(OUT)}(t)\right)$ (non-NFV Node)

Markov model solution

- **Assumptions:**
  - Exponentially-distributed interarrival times
  - Exponentially-distributed service times in both NF and OUT queues
  - the routing algorithm is able to avoid closed loops

hypotheses of the Jackson theorem

the equilibrium probability distribution of the network has a product-form solution:

$$\underline{\pi}^{(\Sigma)}(t) = \left[\underline{\pi}_1, \ldots, \underline{\pi}_N\right] = \underline{\pi}_1 \cdot \ldots \cdot \underline{\pi}_N$$

$$\underline{\pi}_i = \left(\pi_{i,1}^{(F)} \cdot \ldots \cdot \pi_{i,L_i^{(F)}}^{(F)}\right) \cdot \left(\pi_{i,1}^{(OUT)} \cdot \ldots \cdot \pi_{i,L_i^{(OUT)}}^{(OUT)}\right) \quad \text{if NFV}$$

$$\underline{\pi}_i = \left(\pi_{i,1}^{(OUT)} \cdot \ldots \cdot \pi_{i,L_i^{(OUT)}}^{(OUT)}\right) \quad \text{if non-NFV}$$

Markov model solution

- **Let us indicate:**
  - ***Utilization coefficient*** of the j-th NF queue in the node i

    $$\boxed{\rho_{i,j}^{(F)} = \frac{\Lambda_{i,j}^{(F)}}{\mu_{i,j}^{(F)}}}$$

  - ***Utilization coefficient*** of the h-th OUT queue in the node i

    $$\boxed{\rho_{i,h}^{(OUT)} = \frac{\Lambda_{i,j}^{(OUT)}}{\mu_{i,j}^{(OUT)}}}$$

$$\boxed{\pi_{i,k}^{(F)} \equiv \lim_{t\to\infty} \mathrm{Prob}\left\{S_i^{(F)}(t) = k\right\} = \left[1 - \rho_{i,j}^{(F)}\right]\cdot\left[\rho_{i,j}^{(F)}\right]^k}$$

$$\boxed{\pi_{i,k}^{(OUT)} \equiv \lim_{t\to\infty} \mathrm{Prob}\left\{S_i^{(OUT)}(t) = k\right\} = \left[1 - \rho_{i,j}^{(OUT)}\right]\cdot\left[\rho_{i,j}^{(OUT)}\right]^k}$$

Performance parameters

- **Probability that the VM *j* in the node *i* is not using the CPU quota assigned to it:**

$$P_{i,j}^{(F0)} = 1 - \rho_{i,j}^{(F)} = 1 - \frac{\Lambda_{i,j}^{(F)}}{\mu_{i,j}^{(F)}}$$

- **Mean number of packets in the queueing systems** $Q_{i,j}^{(F)}$ **and** $Q_{i,j}^{(OUT)}$

$$\nu_{i,j}^{(F)} = \frac{\rho_{i,j}^{(F)}}{1 - \rho_{i,j}^{(F)}} \qquad \nu_{i,j}^{(OUT)} = \frac{\rho_{i,j}^{(OUT)}}{1 - \rho_{i,j}^{(OUT)}}$$

- **Mean sojourn time in the queueing system** $Q_{i,j}^{(F)}$ **and** $Q_{i,j}^{(OUT)}$

$$W_{i,j}^{(F)} = \frac{\nu_{i,j}^{(F)}}{\Lambda_{i,j}^{(F)}} \qquad W_{i,j}^{(OUT)} = \frac{\nu_{i,j}^{(OUT)}}{\Lambda_{i,j}^{(OUT)}}$$

Performance parameters

- ***End-to-end delay*** **for each flow**

$$\boxed{W_k^{(e2e)} = \sum_{i=1}^{N} \left[\sum_{j=1}^{L_i^{(F)}} W_{i,j}^{(F)} \cdot I_{i,j}^{(F)}(k) + \sum_{h=1}^{L_i^{(OUT)}} W_{i,h}^{(OUT)} \cdot I_{i,h}^{(OUT)}(k) \right]}$$

where:

$$I_{i,j}^{(F)}(k) = \begin{cases} 1 & \text{if the flow } k \text{ uses the function } j \text{ in the node } i \\ 0 & \text{otherwise} \end{cases}$$

$$I_{i,H}^{(OUT)}(k) = \begin{cases} 1 & \text{if the flow } k \text{ leaves the node } i \text{ through the NIC } h \\ 0 & \text{otherwise} \end{cases}$$

CASE STUDY

Case study

ROUTING ALGORITHM

- **TARGET**
  - finding the end-to-end path for each flow
- **REQUIREMENTS**
  - the first and the last nodes for each flow are the ingress and the egress nodes specified for that flow
  - the path for each flow has to cross nodes implementing the functions requested by that flow

Case study

ROUTING ALGORITHM

SOME NOTATION

- C: ***reference link capacity***
  - defined as the bandwidth of the link with the highest capacity in the network

- **All the link capacities are normalized with respect to C**

Case study

ROUTING ALGORITHM

SOME NOTATION

- I_v^t : **Boolean characterization of the network function distribution**

$$I_v^t = \begin{cases} 1 & \text{if the node } v \text{ implements the Network Function } t \\ 0 & \text{otherwise} \end{cases}$$

Case study

ROUTING ALGORITHM

SOME NOTATION

- I_v^t : **Boolean characterization of the network function distribution**

$$I_v^t = \begin{cases} 1 & \text{if the node } v \text{ implements the Network Function } t \\ 0 & \text{otherwise} \end{cases}$$

- a_s^t : **Boolean characterization of the function requirements for network traffic**

$$\alpha_s^t = \begin{cases} 1 & \text{if the flow } s \text{ requires the Network Function } t \\ 0 & \text{otherwise} \end{cases}$$

Routing Algorithm definition

ROUTING ALGORITHM

- **Routing algorithm output**

$$y_{vw}^{s} = \begin{cases} 1 & \text{if the flow } s \text{ is allocated on the link } v \rightarrow w \\ 0 & \text{otherwise} \end{cases}$$

- **Routing algorithm target**

Minimize

$$\sum_{s=1}^{S} \sum_{v \in V} \sum_{w \in V} y_{vw}^{s} \cdot f_s$$

Sum of loads of all the links in the network

Routing Algorithm definition

ROUTING ALGORITHM

- **Subject to:**

$$0 \le y_{vw}^{s} \le 1$$

Possible values of the variables

$$\sum_{s \in S} y_{vw}^{s} \cdot f_s \le M_{vw} \quad \forall v, w \in V$$

It ensures that no link carries more traffic flow than its capacity

$$\sum_{w \in V} y_{vw}^{s} = \sum_{w \in V} y_{wv}^{s} \quad \forall v \in V \ \text{ and }$$

$$v \ne \{w, \sigma_s, \delta_s\} \quad \forall s \in S$$

Flow-conservation constraint: it ensures that no flow is lost or created except for at the ingress and the destination nodes

Routing Algorithm definition

ROUTING ALGORITHM

- **Subject to:**

$$\sum_{w \in V} y^{s}_{\sigma_s w} = 1 \qquad \forall s \in S$$

$$\sum_{v \in V} y^{s}_{v \delta_s} = 1 \qquad \forall s \in S$$

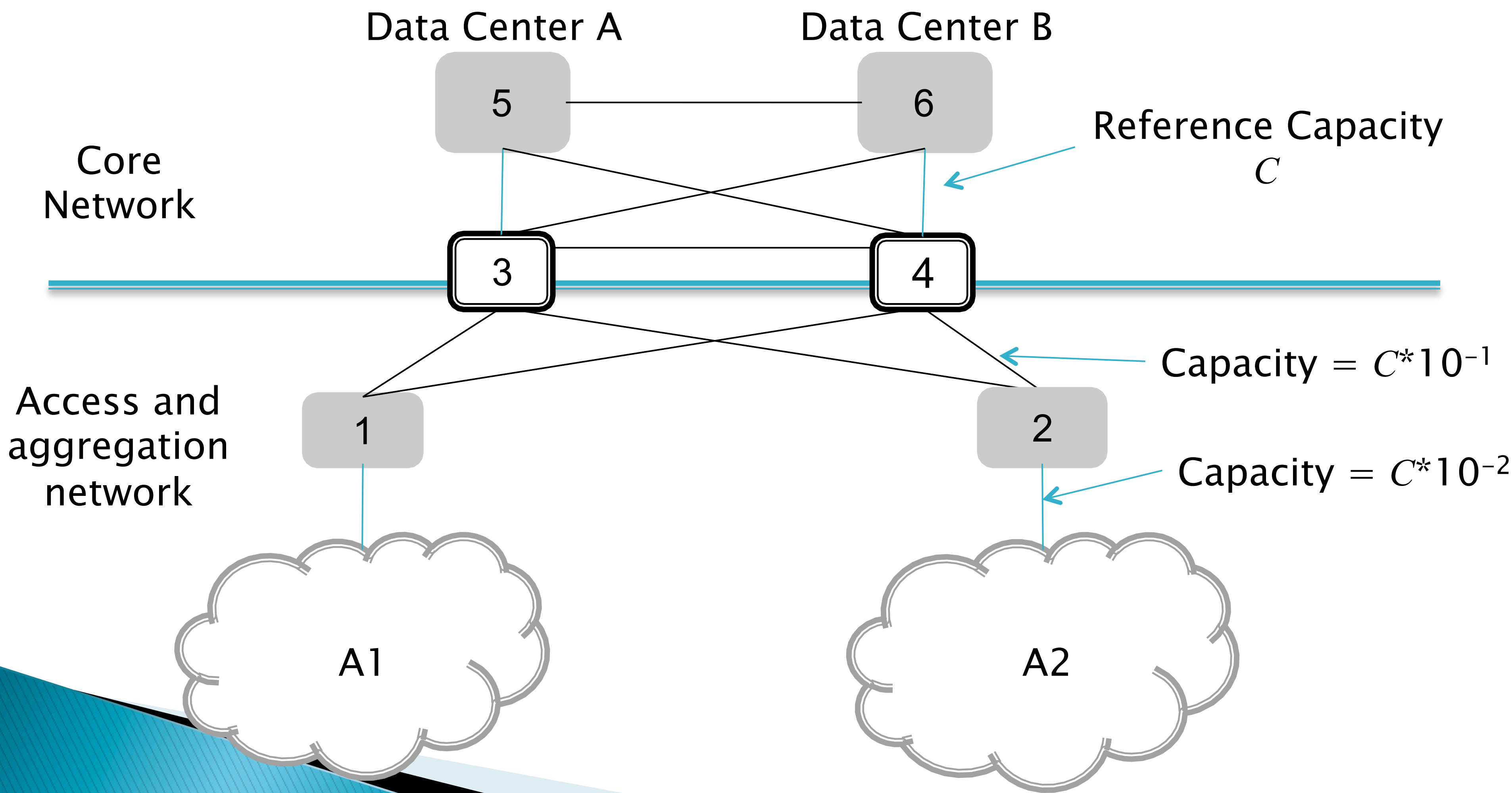
They ensure that the flow *s* enters the network through only one node, and leaves the network from only one node

$$\sum_{v \in V} \sum_{w \in V} y^{s}_{vw} \cdot a^{t}_{s} \cdot I^{t}_{w} \geq 1$$

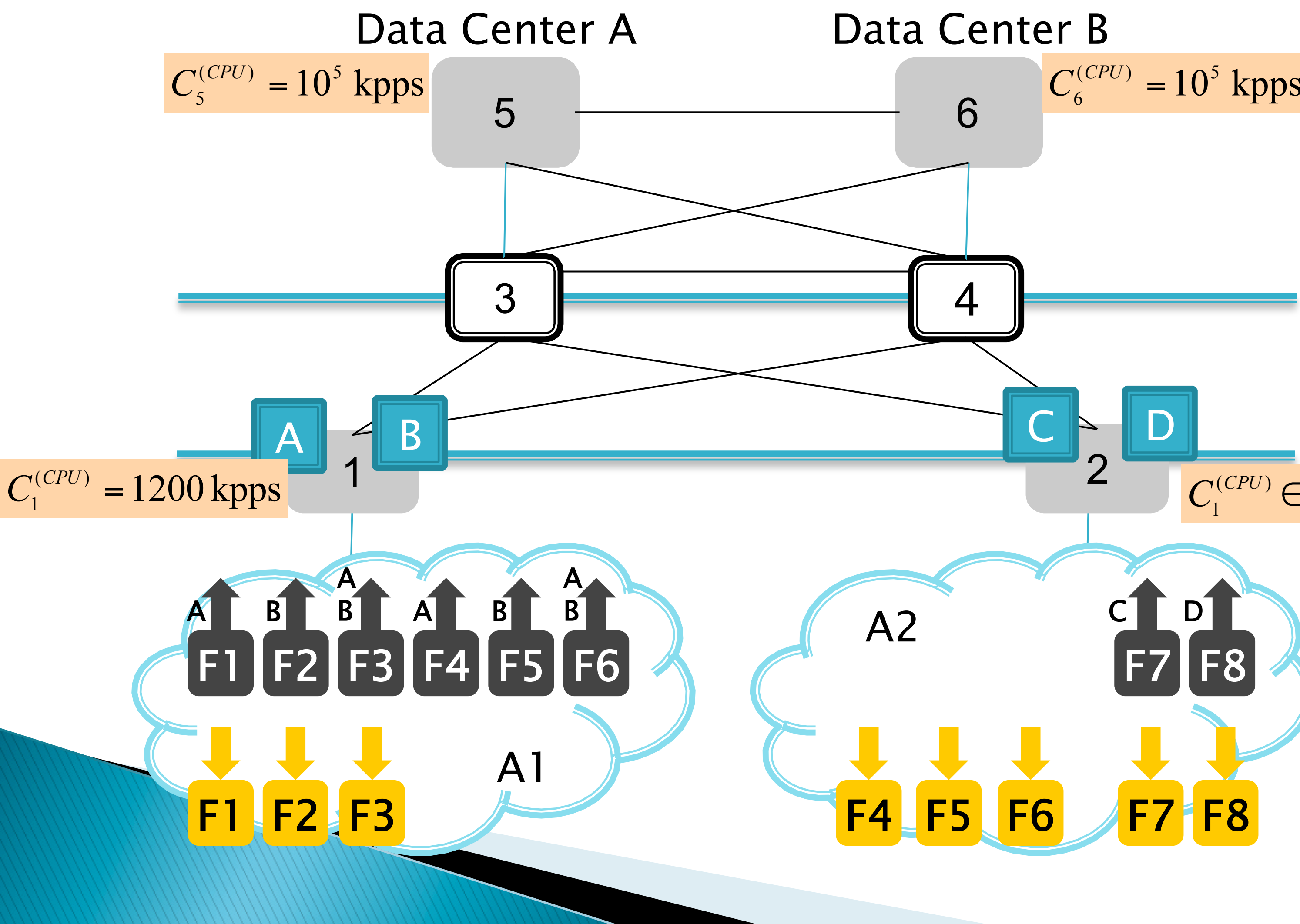
$$\forall s \in S, \quad \forall t \in F$$

It ensures that each traffic flow crosses the nodes which implement the required functions

Case study: network topology



Network topology: Case 1



Eight flows

Fi — Ingress node for the flow i

Fi — Ingress node for the flow i

F1, F2, F3, F4, F5

$f_s = 99.5$ kpps

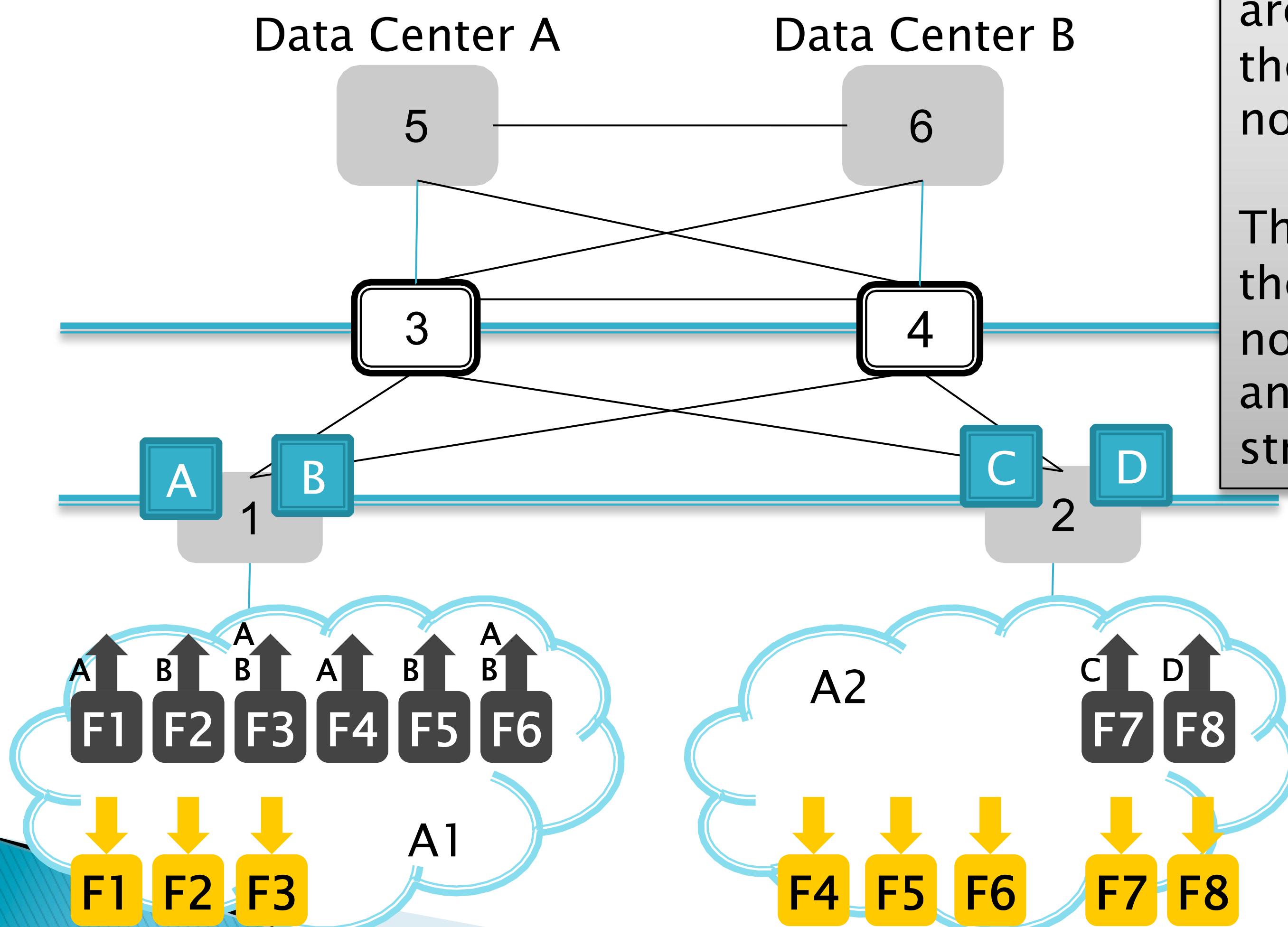
F6

$f_s = 132{,}67$ kpps

F7, F8

$f_s = 666{,}33$ kpps

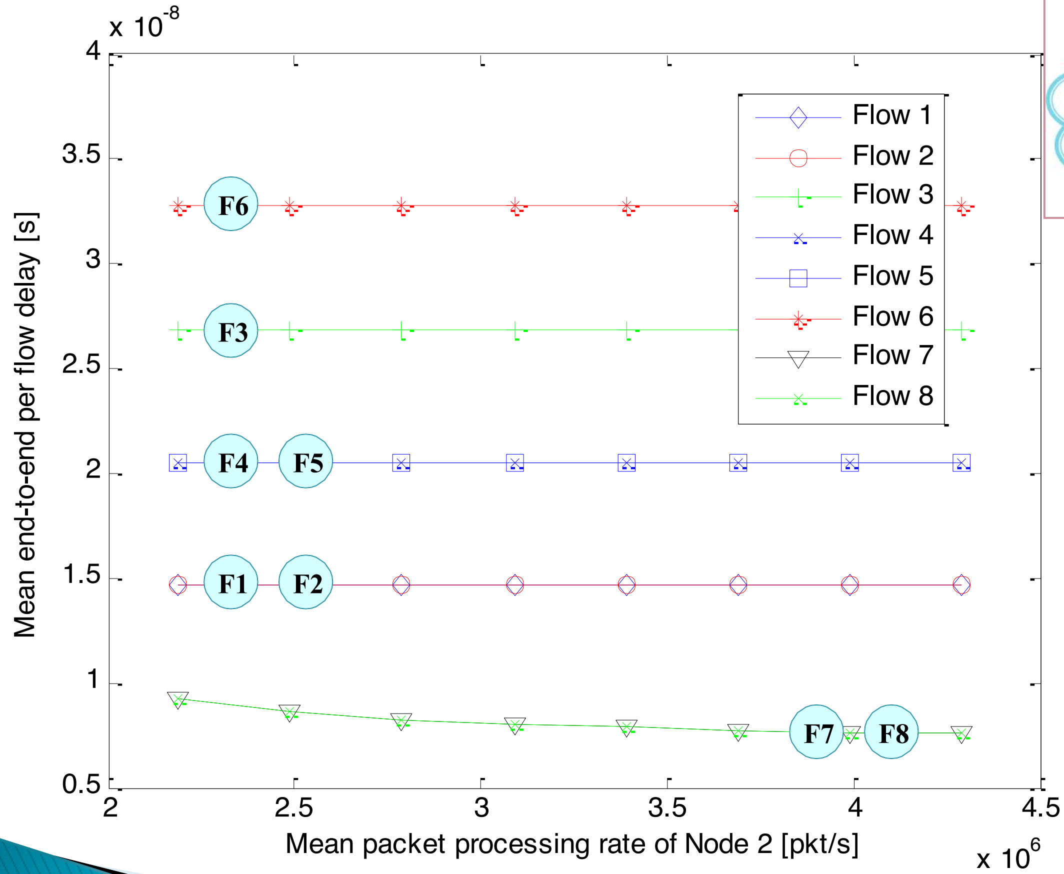
Network topology: Case 1

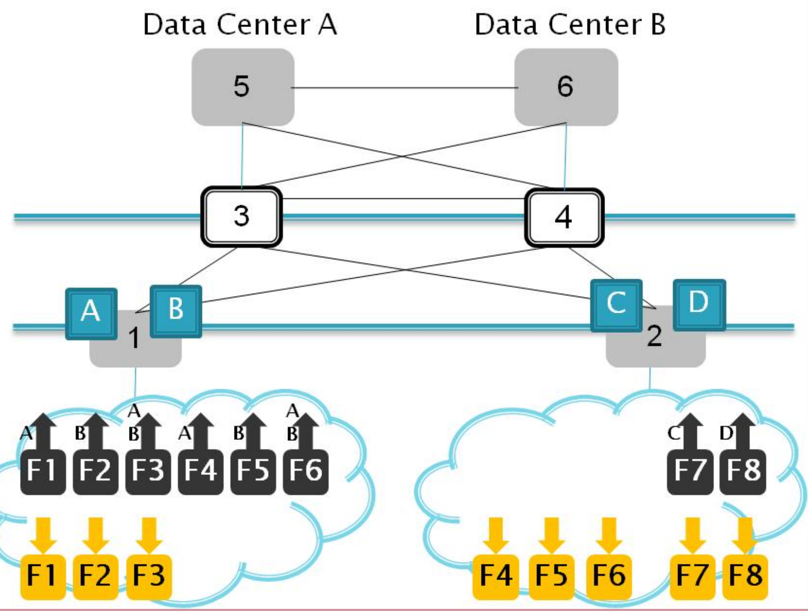


All the functions are allocated on the aggregation nodes.

This case stresses the aggregation nodes processing and does not stress the network.

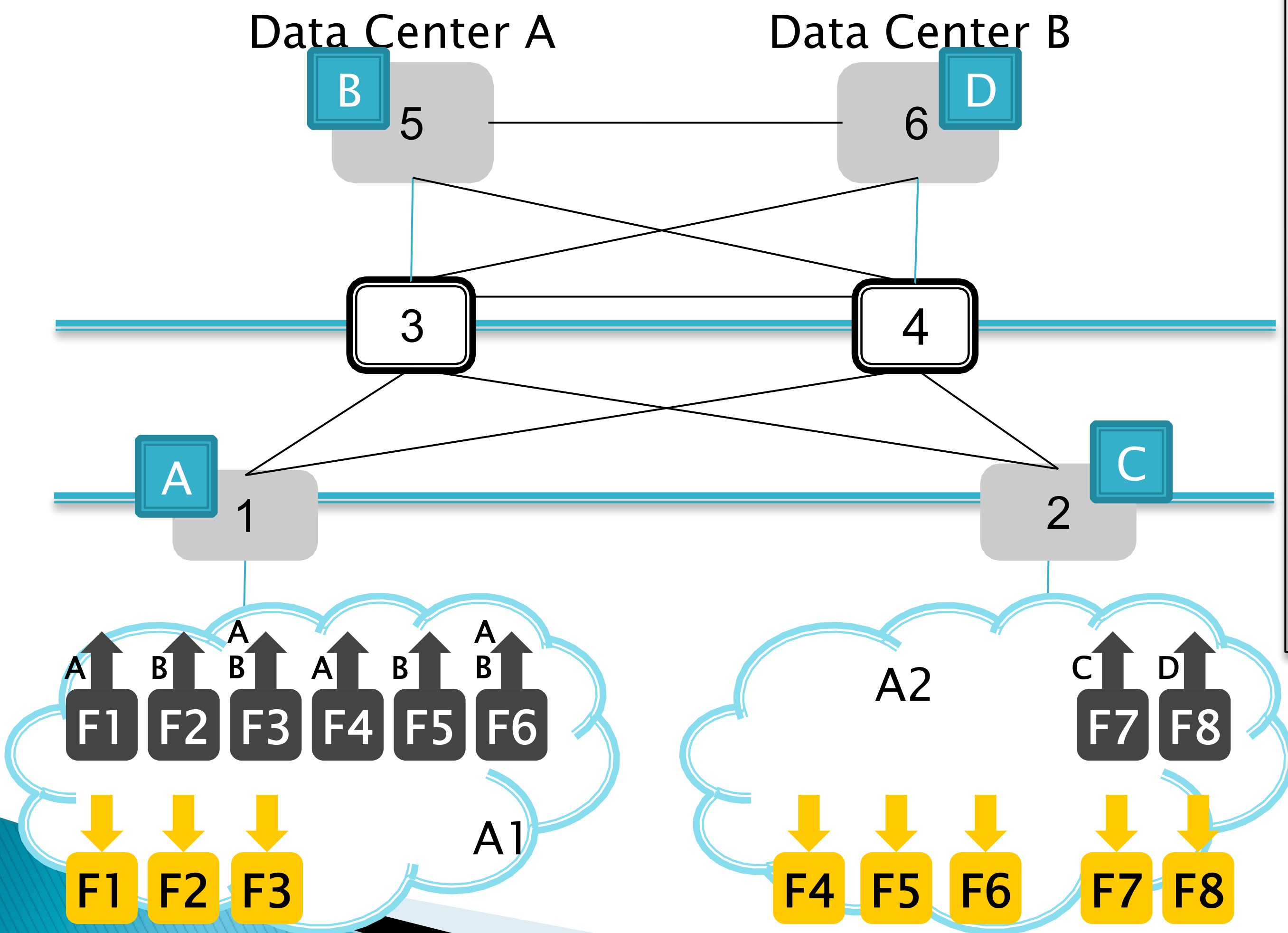
Case 1 results





Only F7 and F8 flows are affected by the Node 2 processing rate because they require functions C and D (that reside on the node 2)

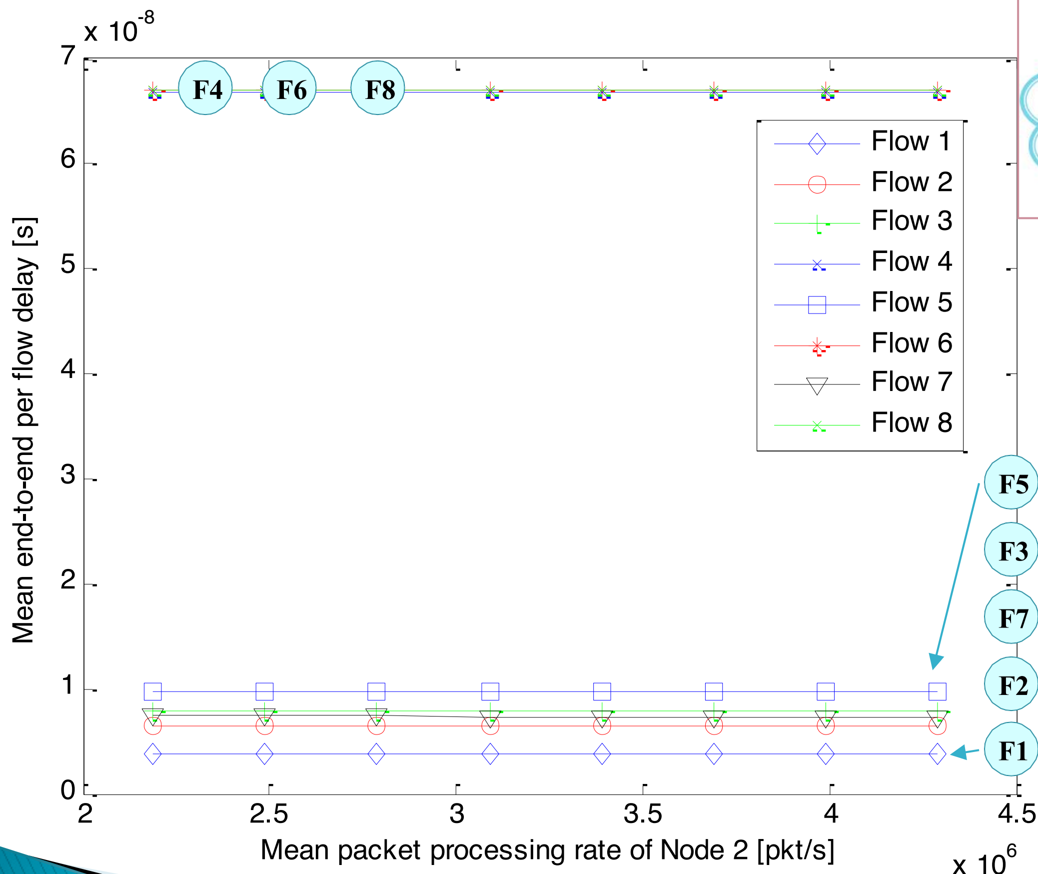
Network topology: case 2

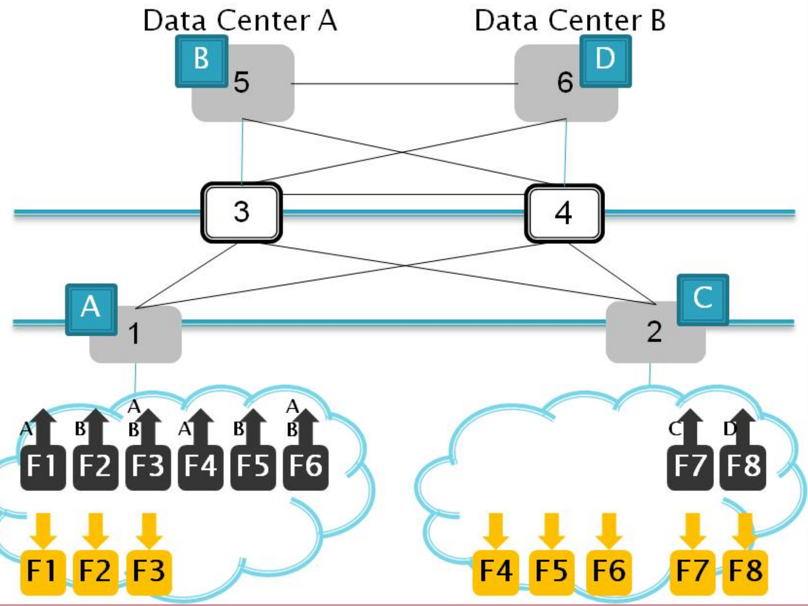


The functions are partially allocated on the aggregation nodes and partially on the Data Centers.

This case stresses both the aggregation nodes processing capacity and the network.

Case 2 results

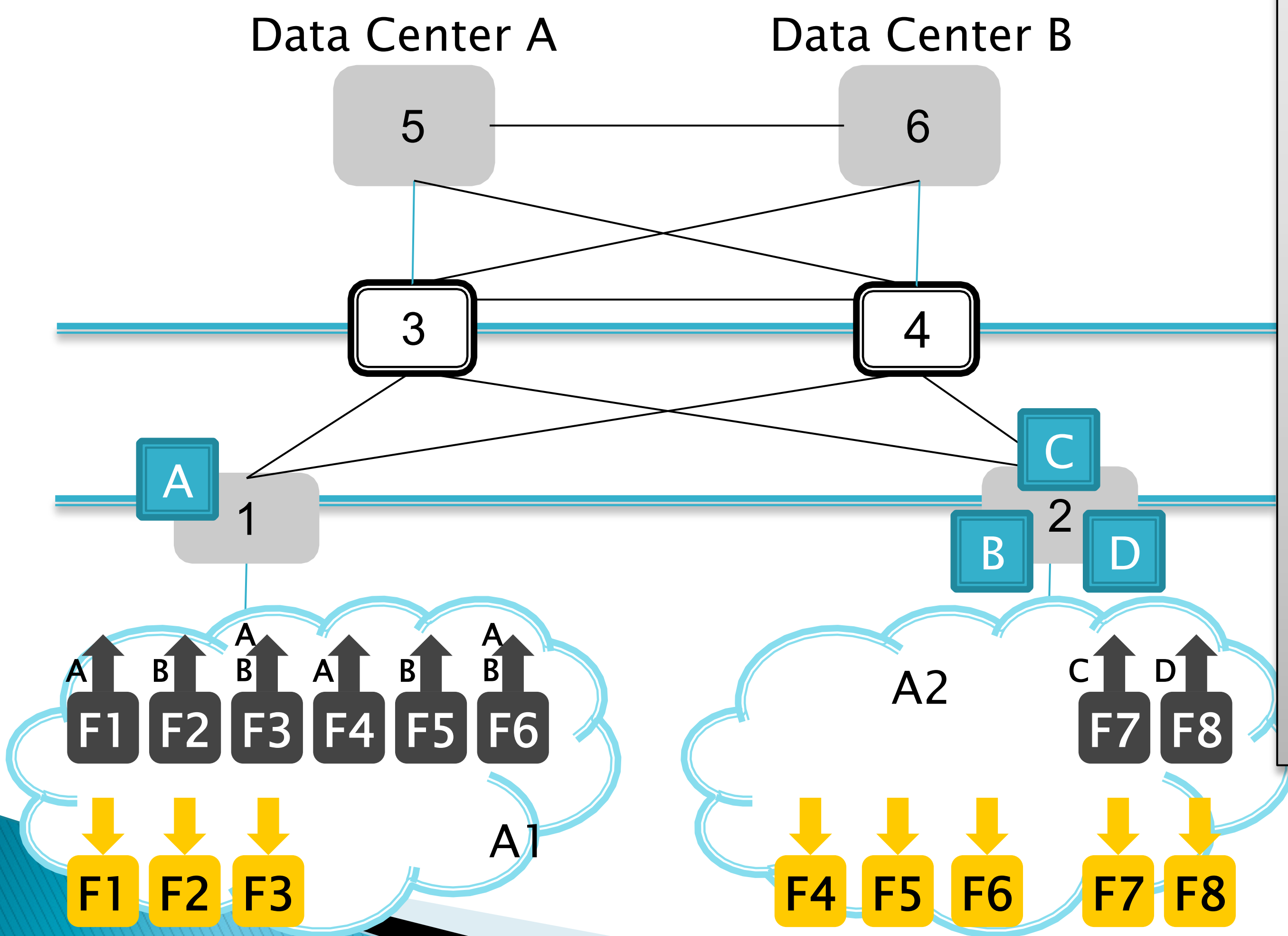




Only F7 is (lightly) influenced by the Node 2 processing rate because it requires the function C

F4, F6 and F8 suffer a higher delay because they have to reach destination in the A2 cloud and, at the same time, need to be processed by the aggregation nodes.

Network topology: Case 3

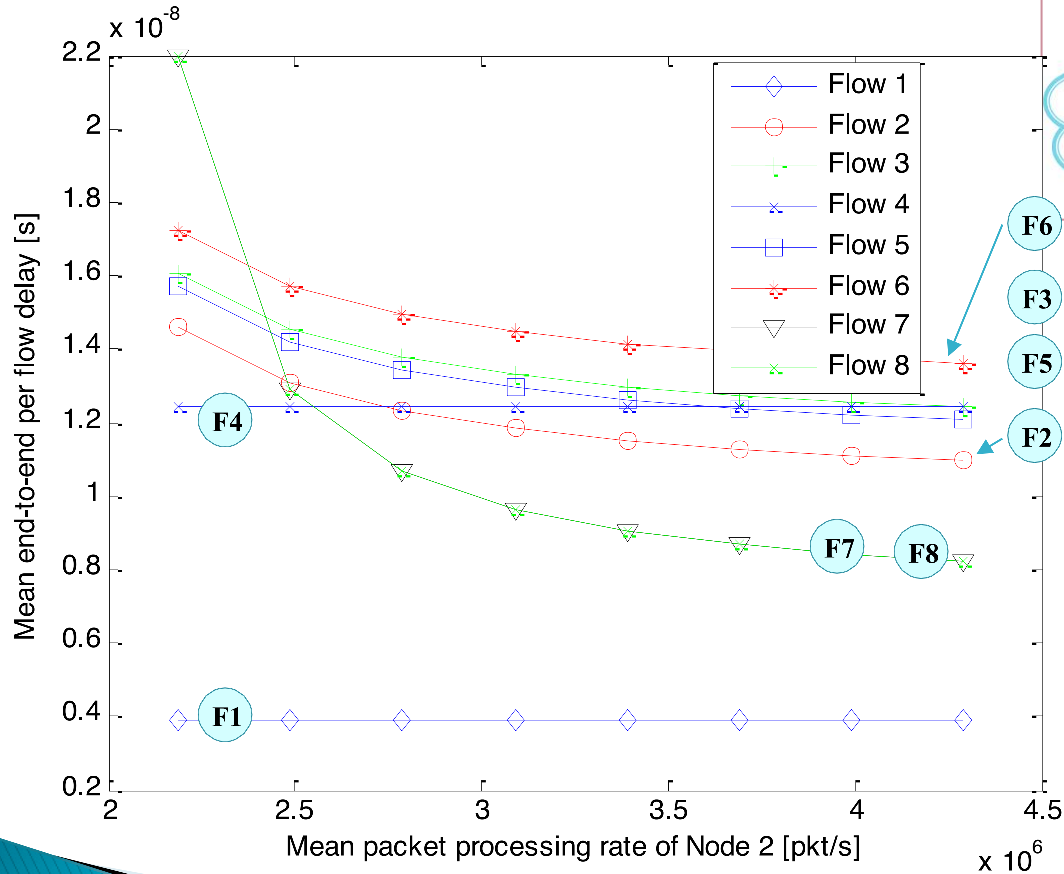


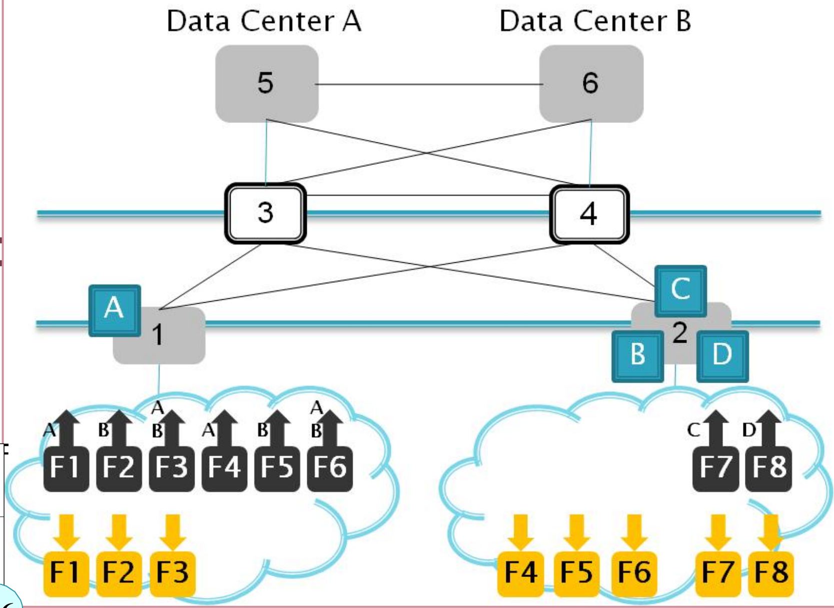
In this case we have stressed:

- Network portion between aggregation nodes and core network
- Processing capacity of Node 2

This case stresses both the aggregation nodes processing capacity and the network.

Case 3 results



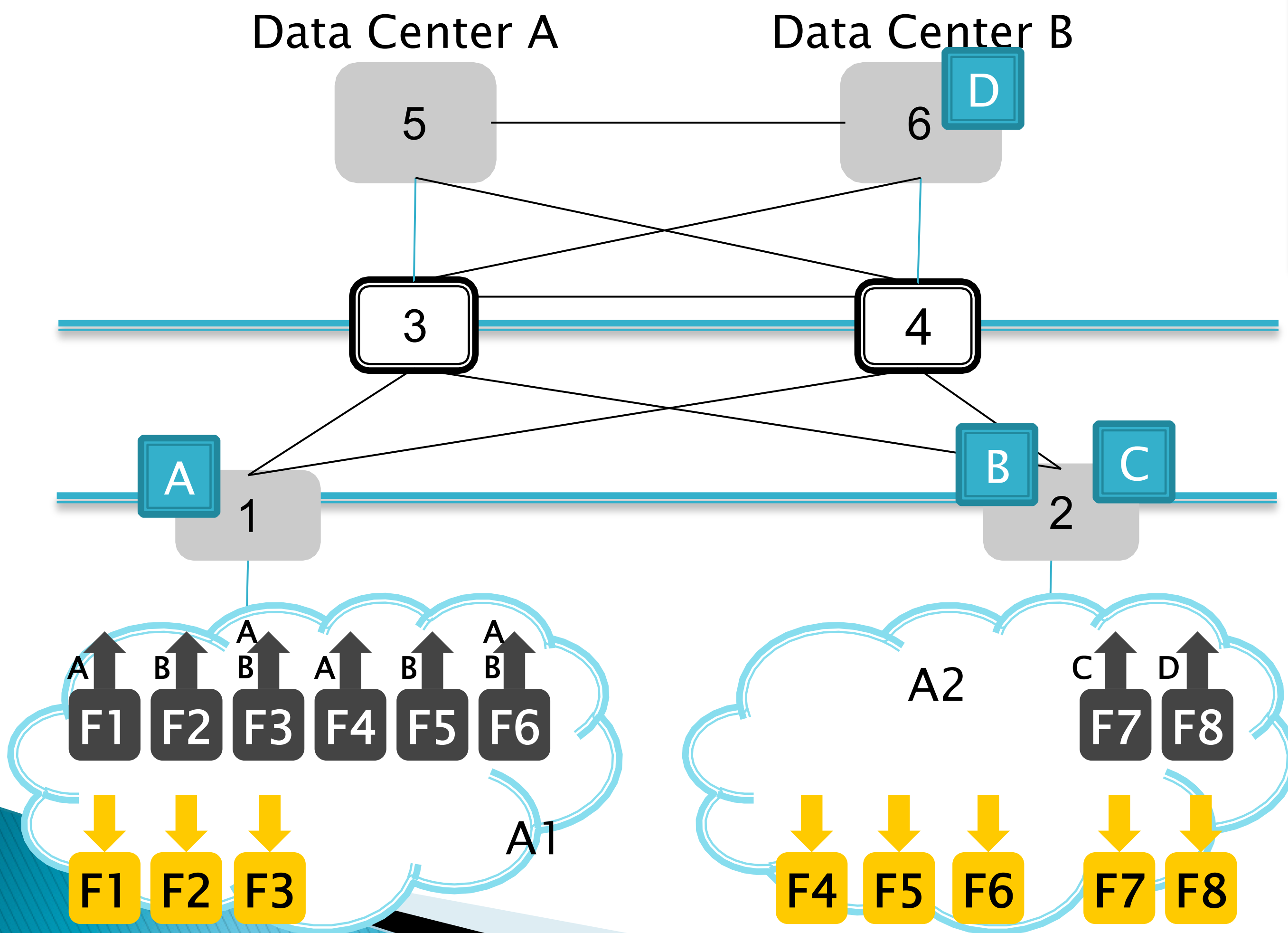


Now [F2, F3, F5, F6] and [F7, F8] flows are influenced by the Node 2 processing rate because:

- [F2, F3, F5, F6] require function B
- [F7, F8] require functions C and D

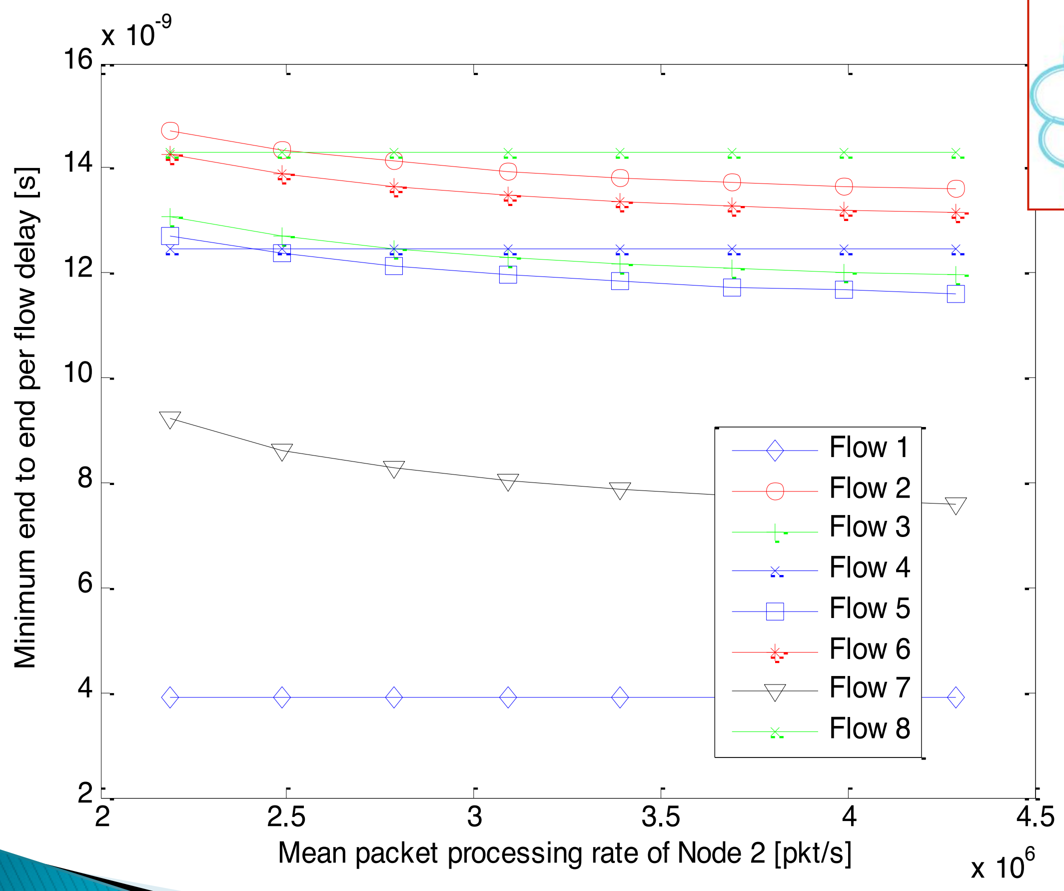
[F7, F8] suffer the same delay

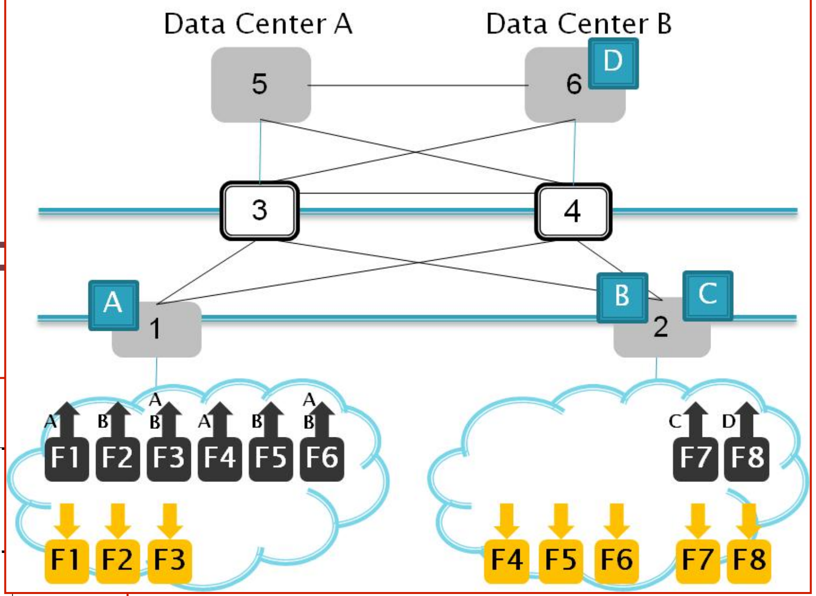
Network topology: case 4



In this case we reduced the processing load of node 2, more stressing the network

Case 4 results



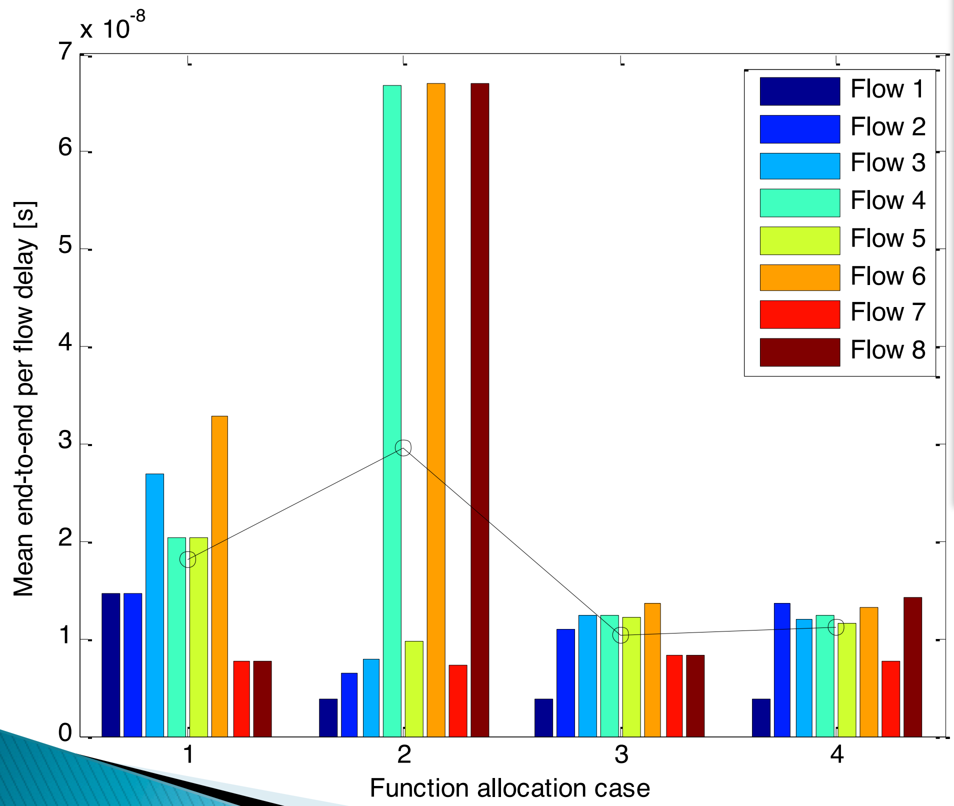


Now all the flows are less influenced by Node 2 processing capacity variation because Node 2 is less overloaded.

Case comparison

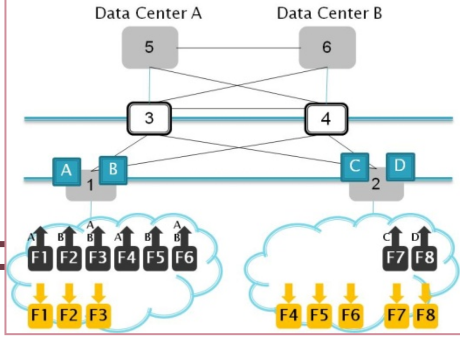
Let us use the model to find the best function allocation

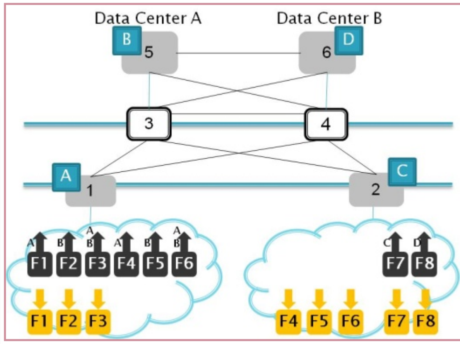
$$C_2^{(CPU)} = 4290\,\text{kpps}$$

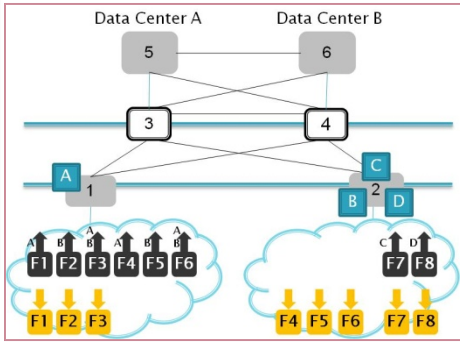


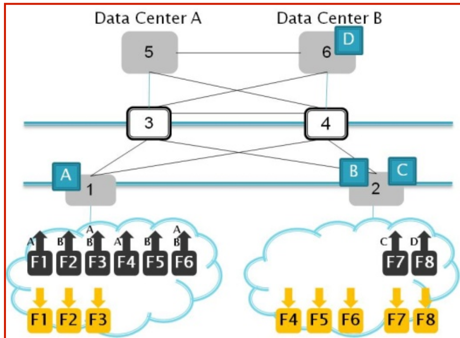
Cases 3 and 4 are the best cases.

The case 2 it the most unfair and present the worst case in terms of mean end-to-end delay









Conclusions

- A telecommunications network with NFV capabilities has been considered
- An analytical framework of the network has been defined
- The model applicability has been demonstrated in a case study

Future work

- **Accurate model of a single NFV node**
  - Markov model of all function queues capturing their correlated behaviors
- **Definition and evaluation of routing algorithms specific for NFV networks**
  - A centralized constrained routing algorithm could optimize the traffic allocation with respect to the function allocation
- **Function allocation policies**

Future work

- Function Migration techniques
- Analytical model of the transient period during function migration
- Definition of green techniques for NFV networks
  - Global approach (e.g. path aggregation and specific function allocation)
  - Local approach (e.g. frequency scaling in node processors)


Thank You


QUESTIONS?