

Reconfiguration de chaînes de fonctions de services dynamiques sans interruption

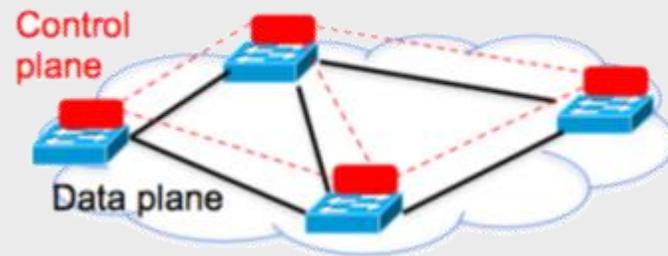


Adrien GAUSSERAN
Équipe COATI
Sophia-Antipolis
adrien.gausseran@inria.fr

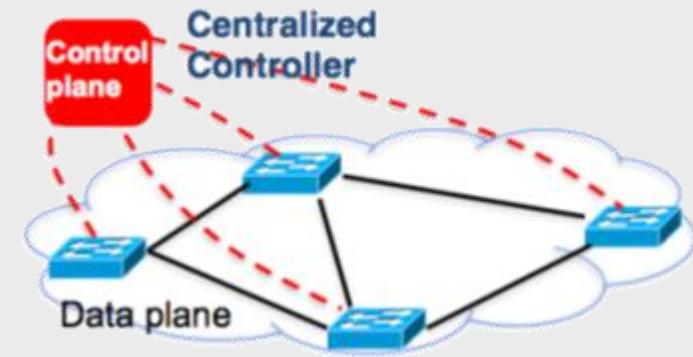
Contexte : Software Defined Networking (SDN)

- Centraliser le contrôle du réseau
- Avantages :
 - Vision et contrôle global du réseau
 - Réseau programmable
 - Routage dynamique

Réseau
Traditionnel :



Réseau
SDN :

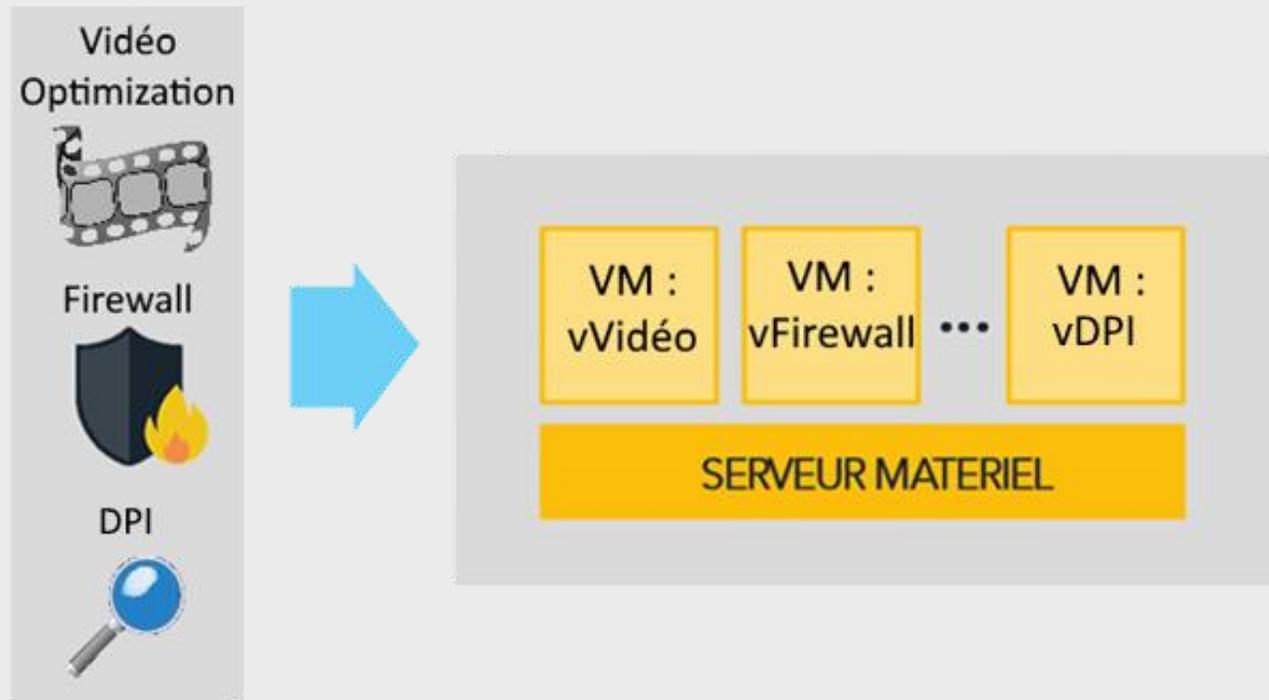


Contexte : Network Function Virtualization (NFV)

➤ Découpler les fonctions réseaux des machines qui les exécutent

➤ Avantages :

- Flexibilité
- Coûts
- Modularité

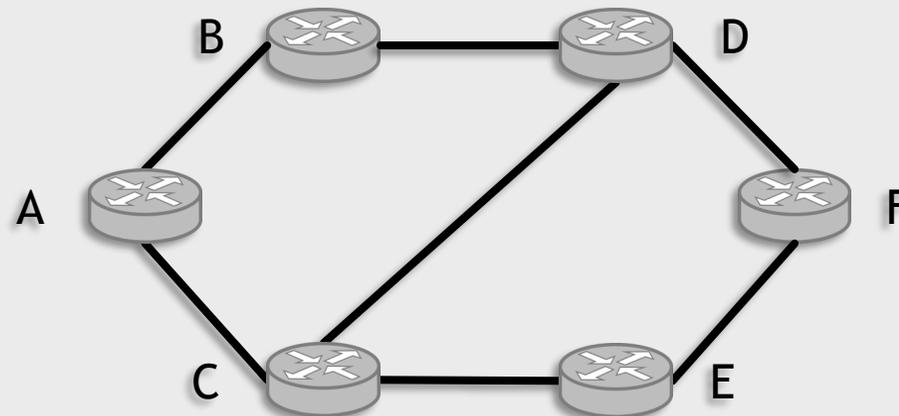


Chaînes de fonctions de services (SFC)

- Chaîne ordonnée de fonctions de réseau à appliquer aux flots sur le réseau.

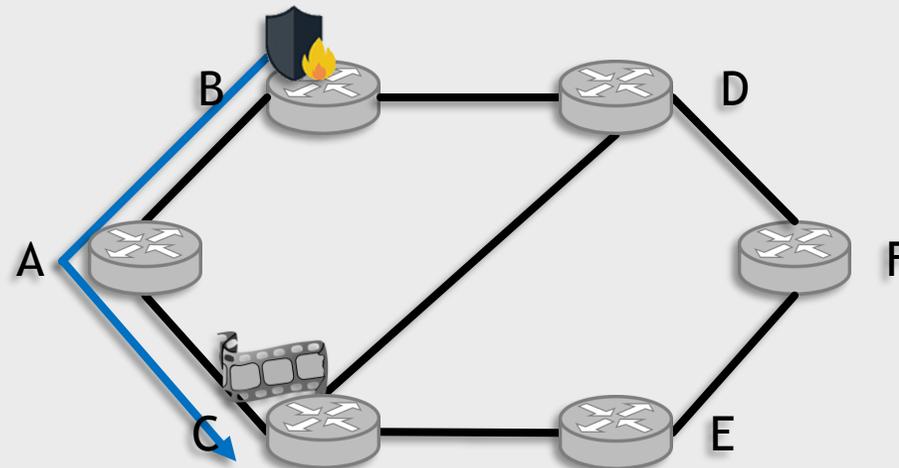
3 flots

B à C		
F à C		
A à F		



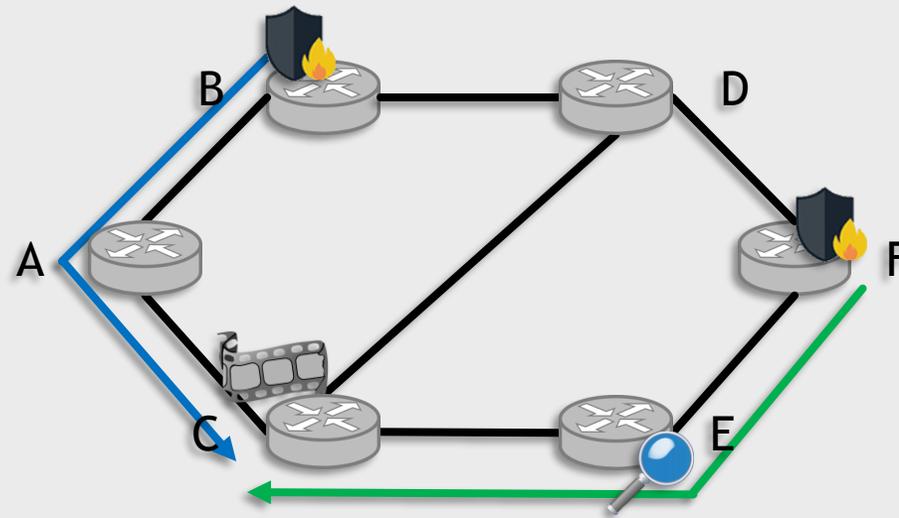
Chaînes de fonctions de services (SFC)

- Chaîne ordonnée de fonctions de réseau à appliquer aux flots sur le réseau.



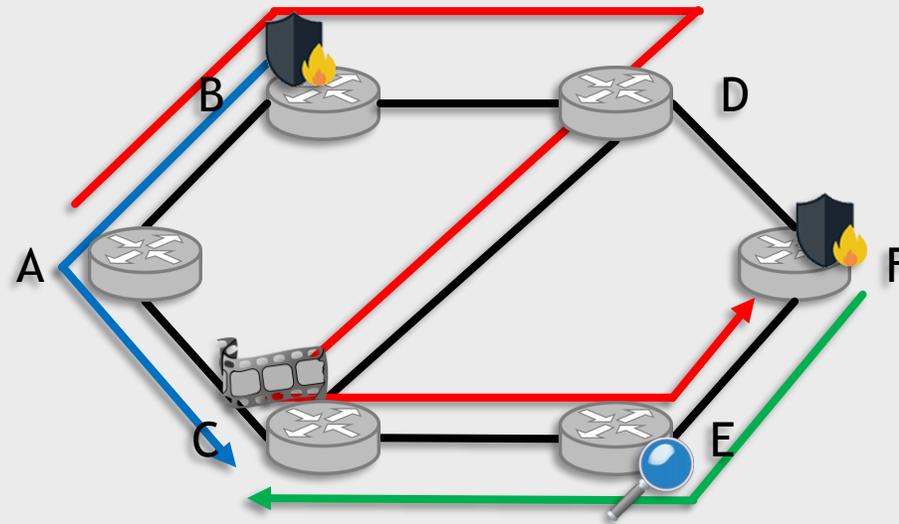
Chaînes de fonctions de services (SFC)

- Chaîne ordonnée de fonctions de réseau à appliquer aux flots sur le réseau.



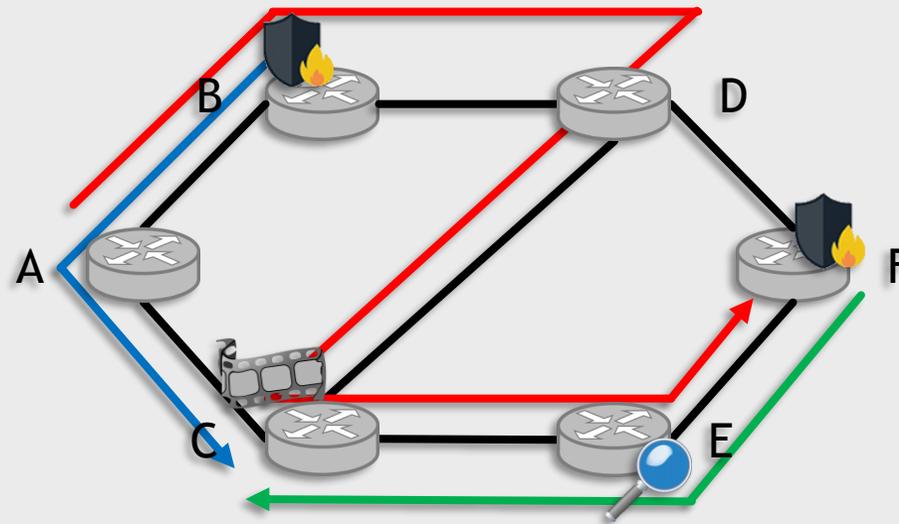
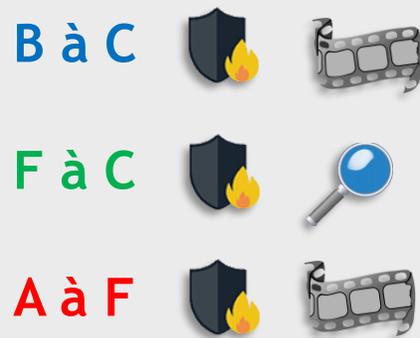
Chaînes de fonctions de services (SFC)

- Chaîne ordonnée de fonctions de réseau à appliquer aux flots sur le réseau.



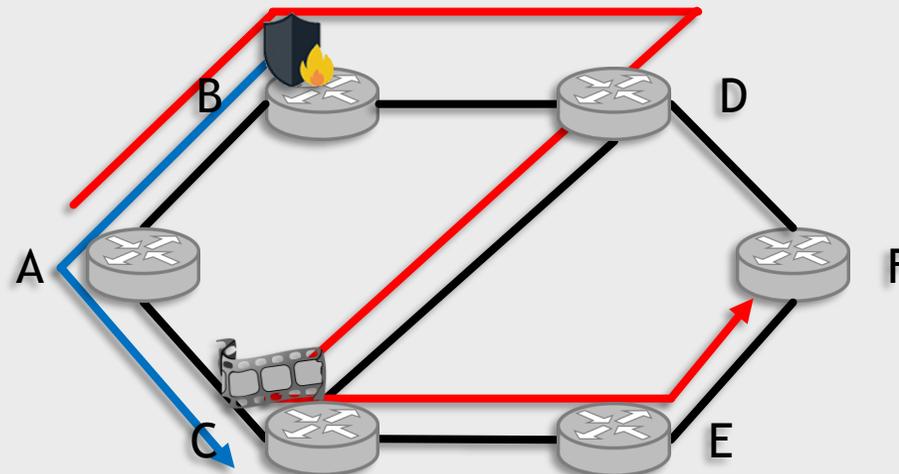
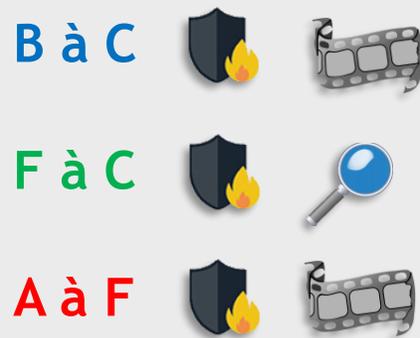
Reconfiguration (make before break)

➤ Allocation dynamique + Désallocation → Routage non optimal



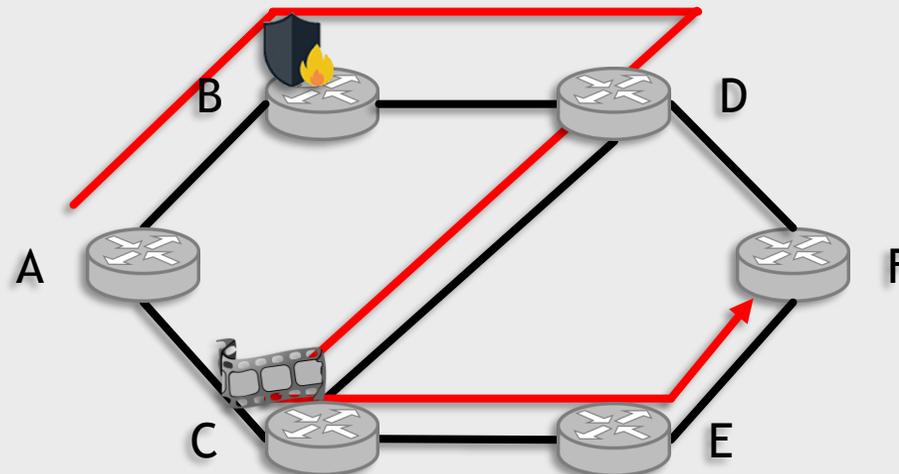
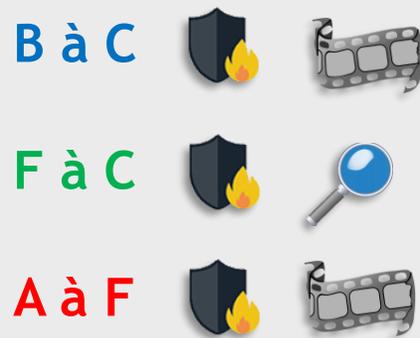
Reconfiguration (make before break)

➤ Allocation dynamique + Désallocation → Routage non optimal



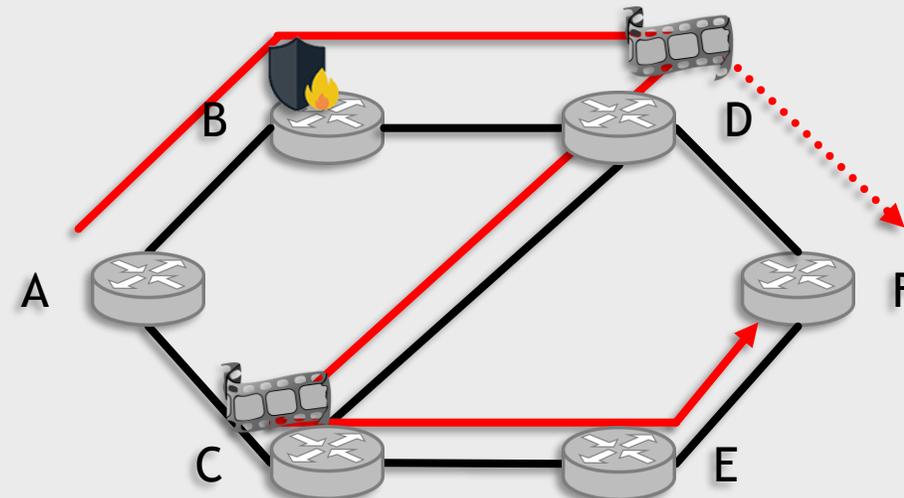
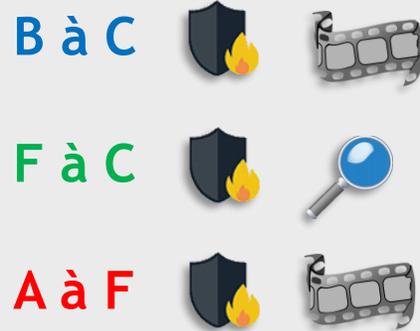
Reconfiguration (make before break)

➤ Allocation dynamique + Désallocation → Routage non optimal



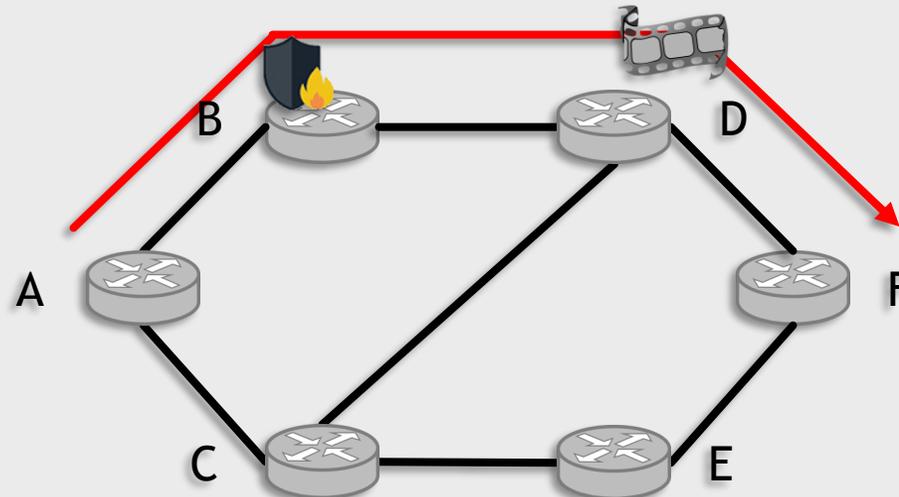
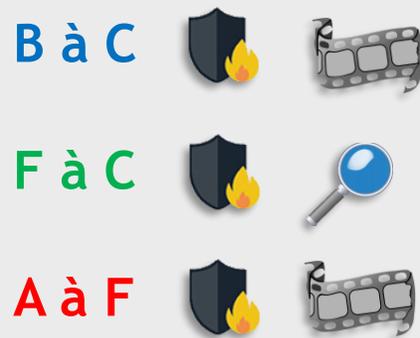
Reconfiguration (make before break)

➤ Allocation dynamique + Désallocation → Routage non optimal



Reconfiguration (make before break)

➤ Allocation dynamique + Désallocation → Routage non optimal



État de l'Art

- Service Function Chaining **sans reconfiguration**
 - [Huin et al. ICC 2017] Optimization of Network Service Chain Provisioning
 - [Apostolos et al. Infocom 2018] Slice-based Column Generation for Network Slicing
- Reconfiguration **sans SFC**
 - [Coudert et al. ONDM 2009] Reconfiguration of the routing in WDM networks with two classes of services
 - [Nga Tran et al. ICC 2013] Reconfiguration of Virtual Network Mapping Considering Service Disruption
 - [Gao et al. Infocom 2018] Virtual Network Reconfiguration with Load Balancing and Migration Cost Consideration
- Reconfiguration **de SFC avec interruptions**
 - [V. Eramo et al. *ACM Transactions on Networking* 2017] An approach for service function chain routing and virtual function network instance migration in network function virtualization architectures
 - [J. Liu et al. TNSM 2017] On dynamic service function chain deployment and readjustment

Problématique

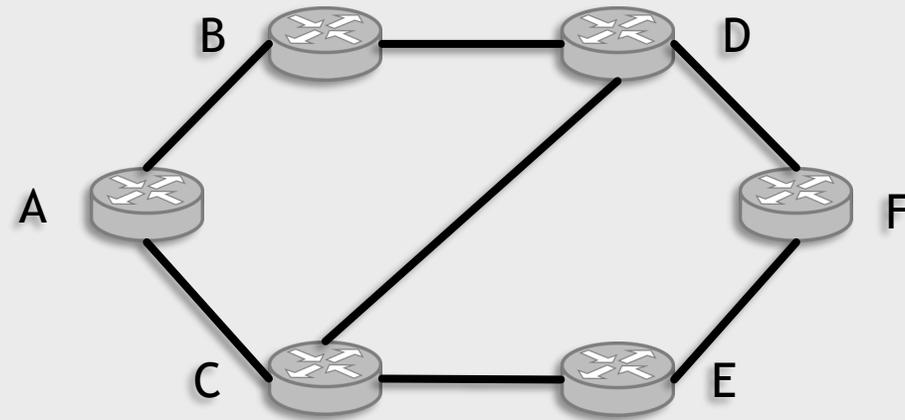
- Allocation dynamique
- Objectif premier :
 - Diminuer les coûts opérationnels
- Objectif secondaire :
 - Augmenter l'acceptation de SFCs

- **Solution** : Reconfigurer sans interrompre le trafic

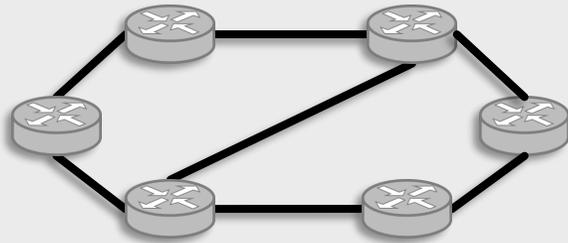
Modélisation de l'Allocation

➤ Problème de routage + allocation → problème de routage

A à F  



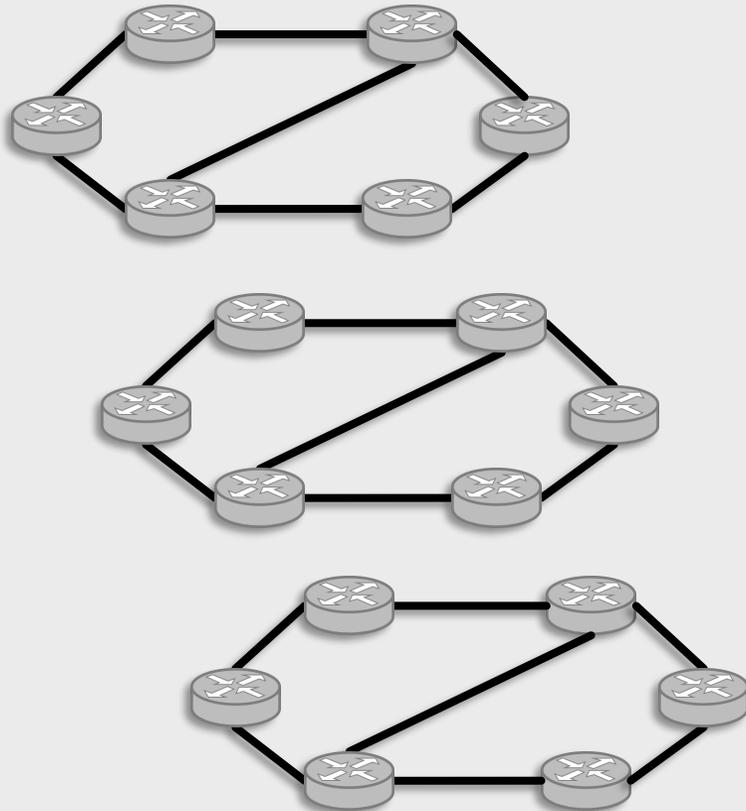
Modélisation de l'Allocation



A à F



Modélisation de l'Allocation

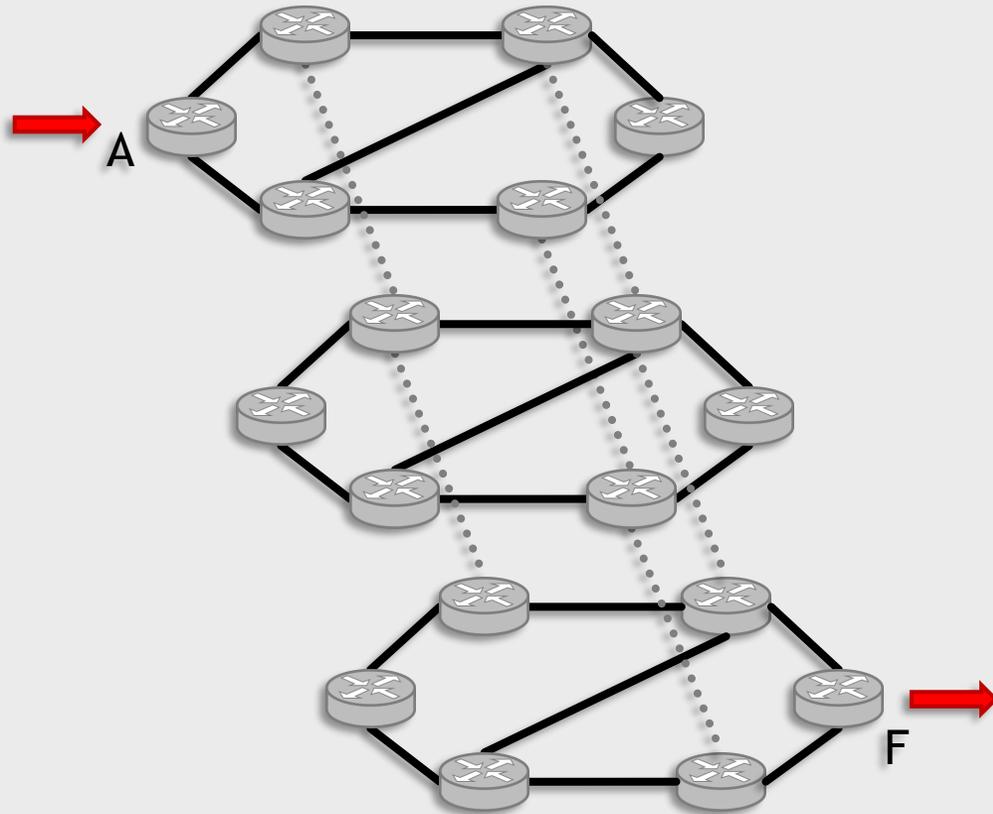


A à F



➤ Partage de la capacité des liens

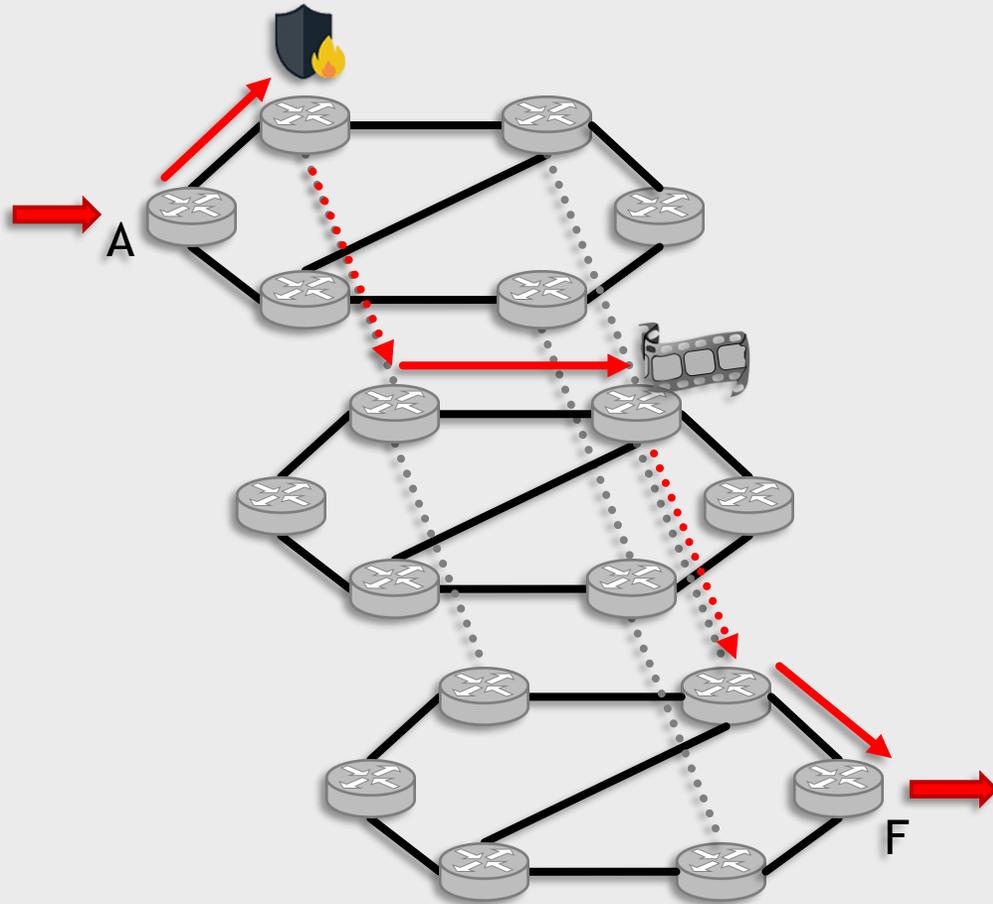
Modélisation de l'Allocation



A à F  

- Partage de la capacité des liens
- Liaison couche à couche par les nœuds
 - Partage de la capacité des nœuds

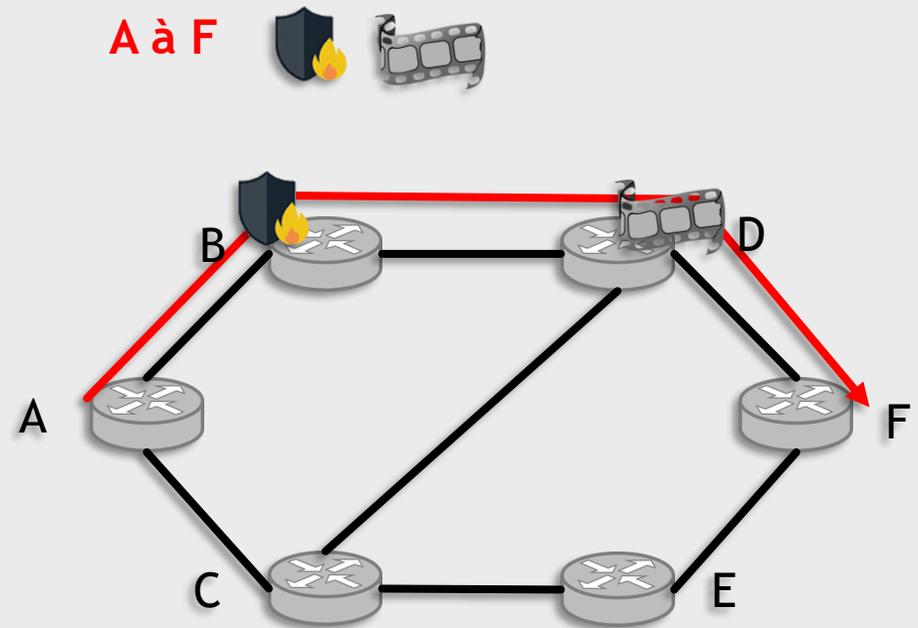
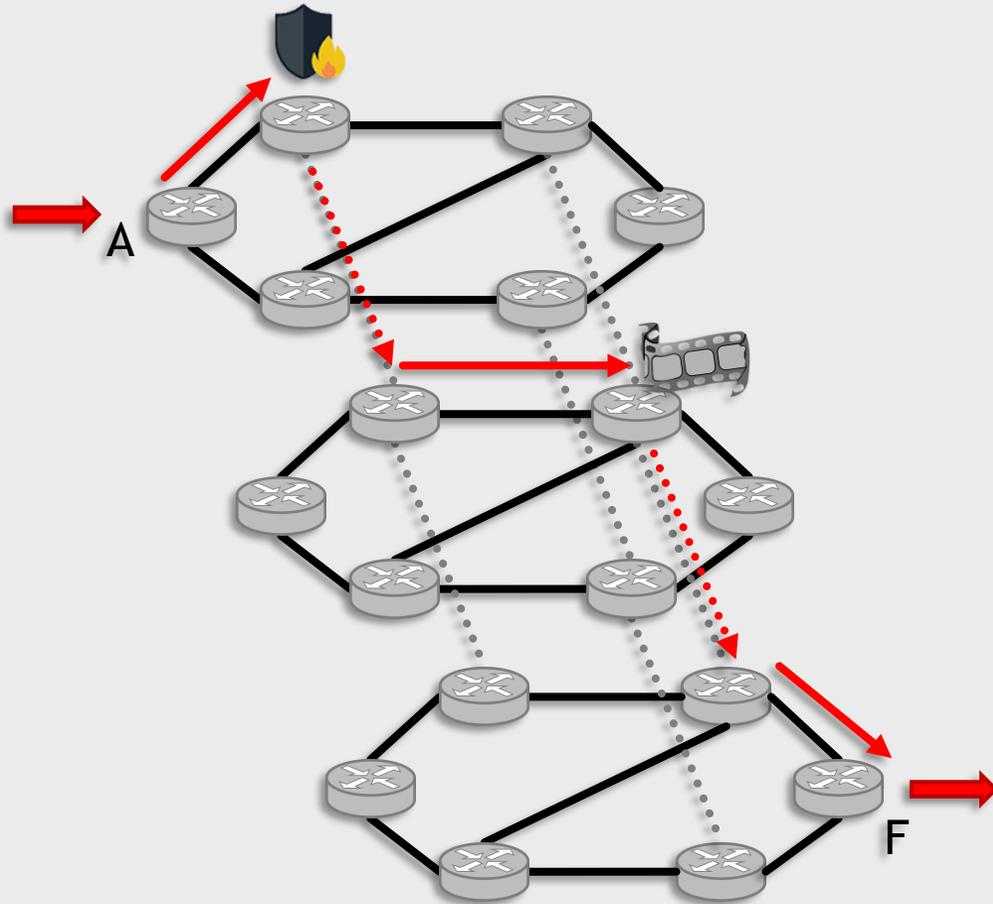
Modélisation de l'Allocation



A à F  

- Partage de la capacité des liens
- Liaison couche à couche par les nœuds
 - Partage de la capacité des nœuds

Modélisation de l'Allocation



ILP : Allocation

➤ Objectif de l'allocation :

$$\min \underbrace{\sum_{d \in D} \sum_{(u,v) \in E} \sum_{i=0}^{|c_d|} bw_d \cdot \varphi_{uv,i}^d}_{\text{red line}} + \beta \cdot \underbrace{\sum_{u \in V} \sum_{f \in F} c_{u,f} \cdot z_{u,f}}_{\text{blue line}}$$

➤ Contraintes de conservation du flot :

➤ Première Couche :

$$\sum_{(u,v) \in \omega^+(u)} \varphi_{uv,0}^d - \sum_{(v,u) \in \omega^-(u)} \varphi_{vu,0}^d + \alpha_{u,0}^d \left\{ \begin{array}{l} = 1 \\ = 0 \end{array} \right. \begin{array}{l} \text{Pour le nœud source} \\ \text{Pour les autres nœuds} \end{array}$$

➤ Dernière Couche

$$\sum_{(u,v) \in \omega^+(v)} \varphi_{uv,|c_d|}^d - \sum_{(v,u) \in \omega^-(v)} \varphi_{vu,|c_d|}^d - \alpha_{u,|c_d|-1}^d \left\{ \begin{array}{l} = -1 \\ = 0 \end{array} \right. \begin{array}{l} \text{Pour le nœud destination} \\ \text{Pour les autres nœuds} \end{array}$$

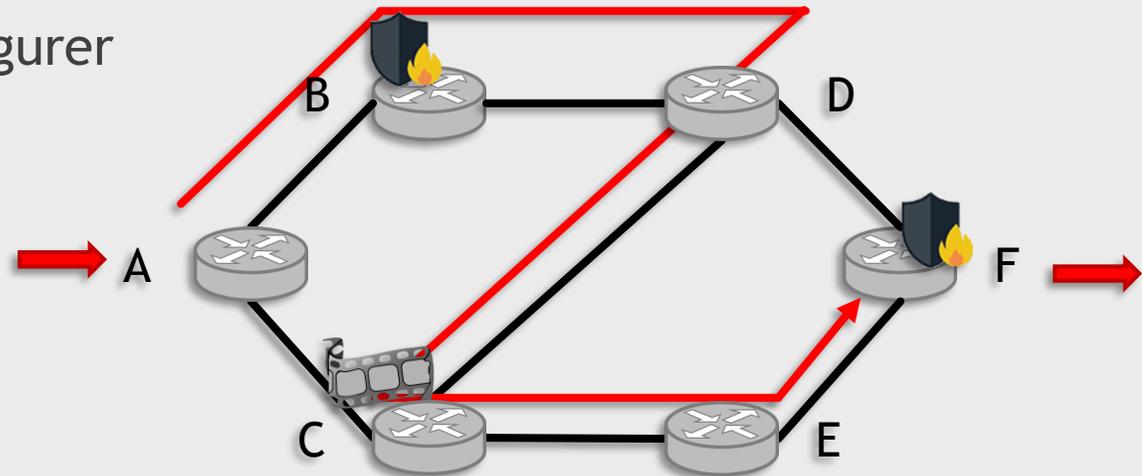
➤ Couches Intermédiaires

$$\sum_{(u,v) \in \omega^+(u)} \varphi_{uv,i}^d - \sum_{(v,u) \in \omega^-(u)} \varphi_{vu,i}^d + \alpha_{u,i-1}^d - \alpha_{u,i-1}^d = 0$$

Modélisation de la Reconfiguration

- Pour calculer une reconfiguration :
 - Allocation initiale
 - Allocation finale
 - Nombre d'étapes pour reconfigurer

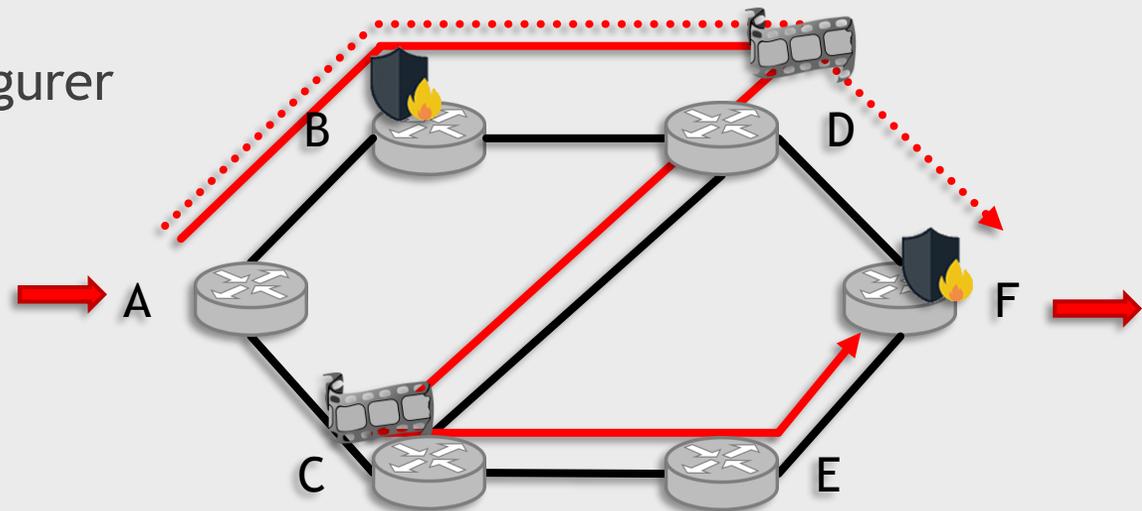
A à F  



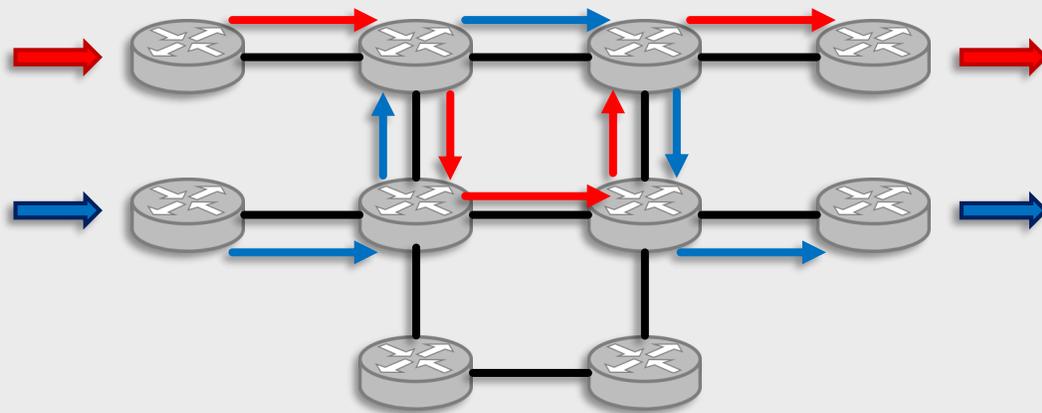
Modélisation de la Reconfiguration

- Pour calculer une reconfiguration :
 - Allocation initiale
 - Allocation finale
 - Nombre d'étapes pour reconfigurer

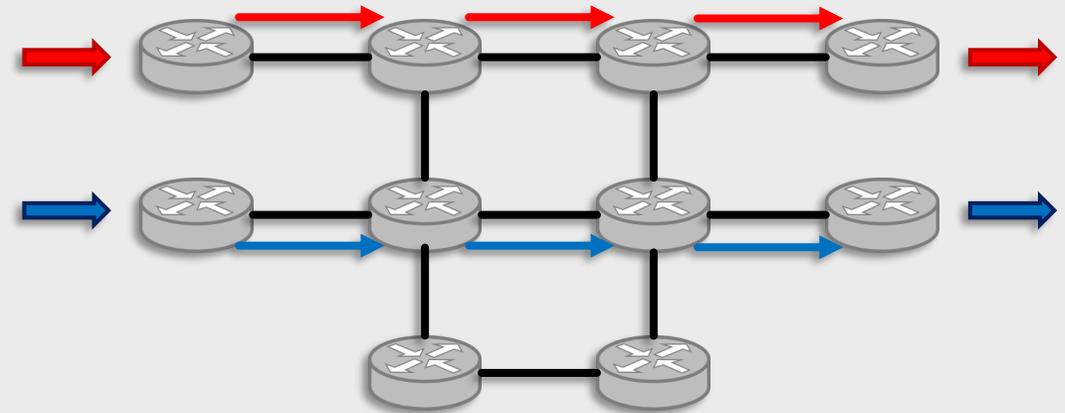
A à F  



Étapes de reconfiguration

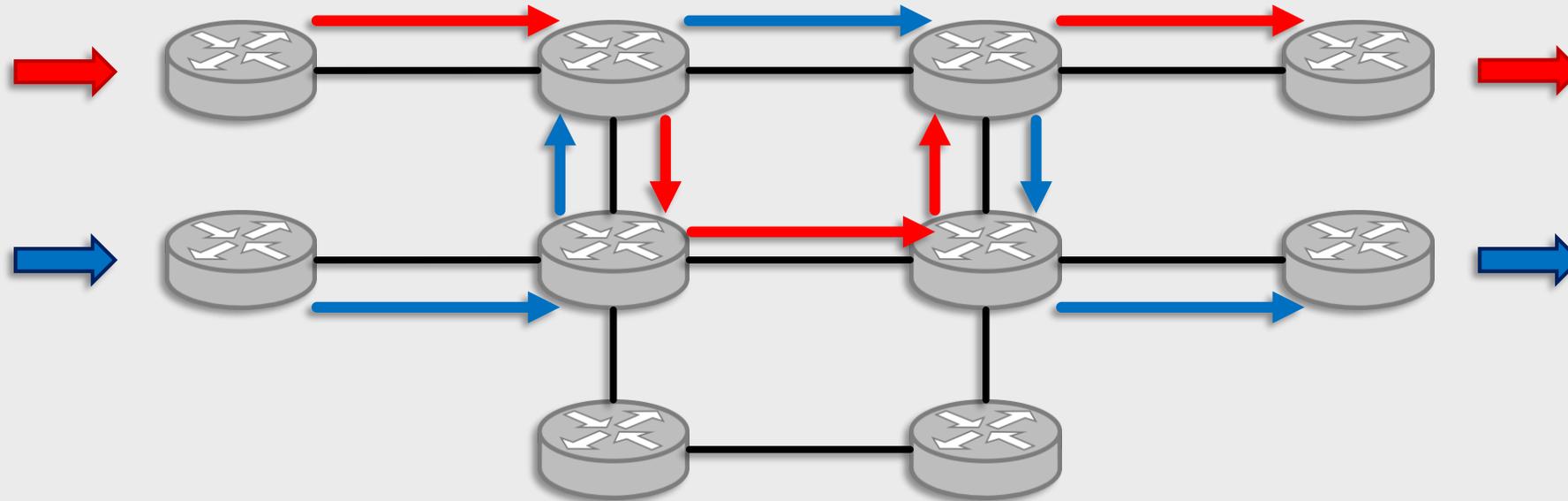


Allocation Initiale
10 liens utilisés



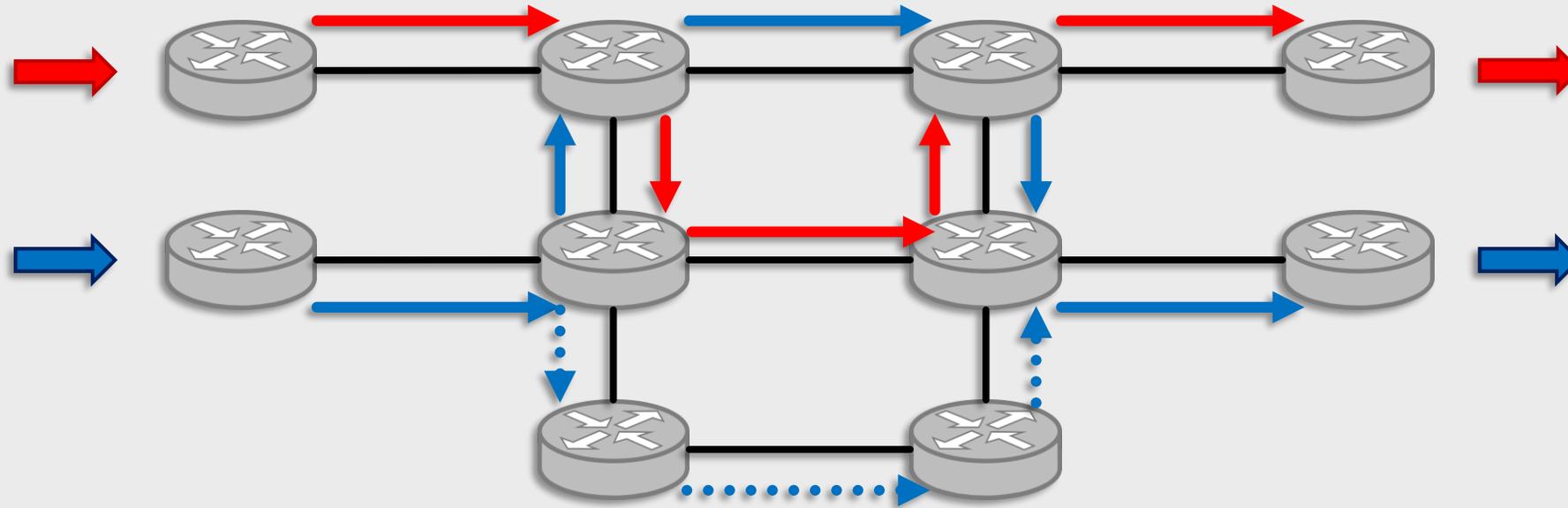
Allocation Finale
6 liens utilisés

Étapes de reconfiguration



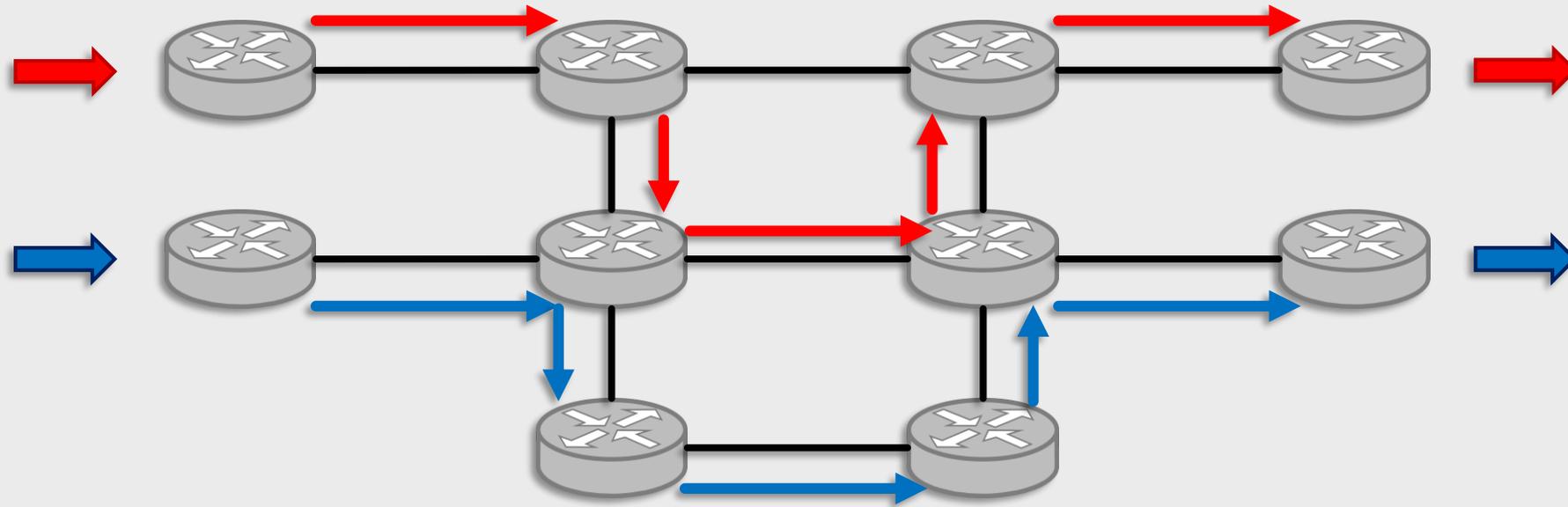
Étape 0 : Étape Initiale

Étapes de reconfiguration



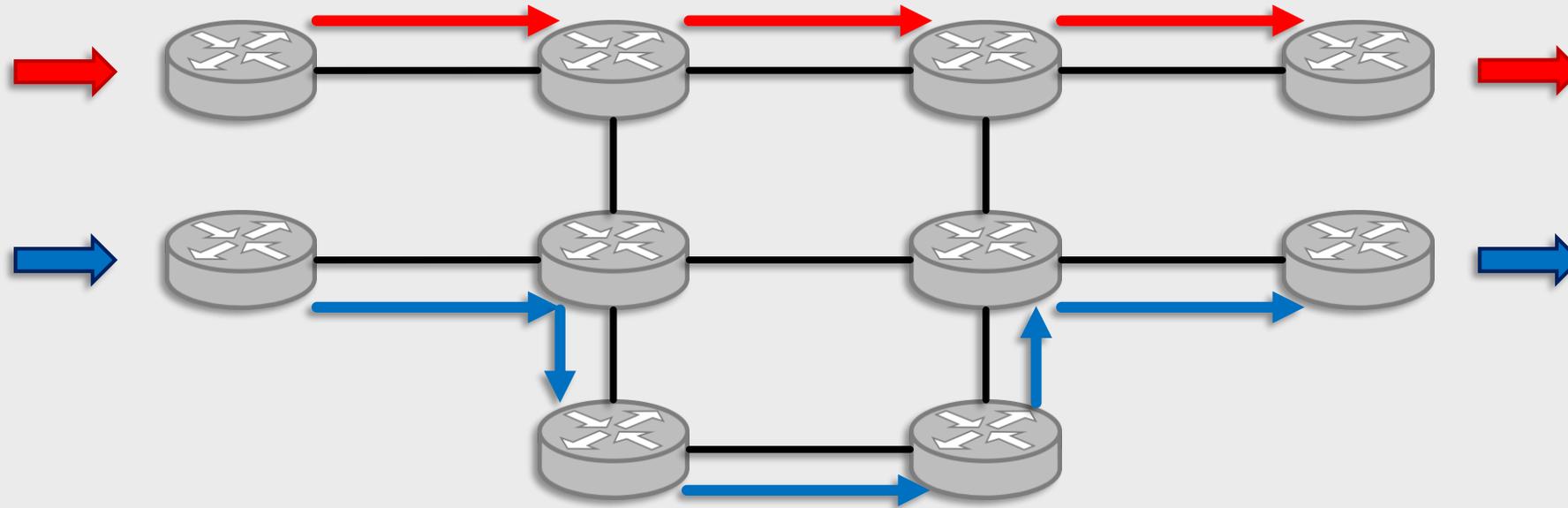
Étape 0 → Étape 1

Étapes de reconfiguration



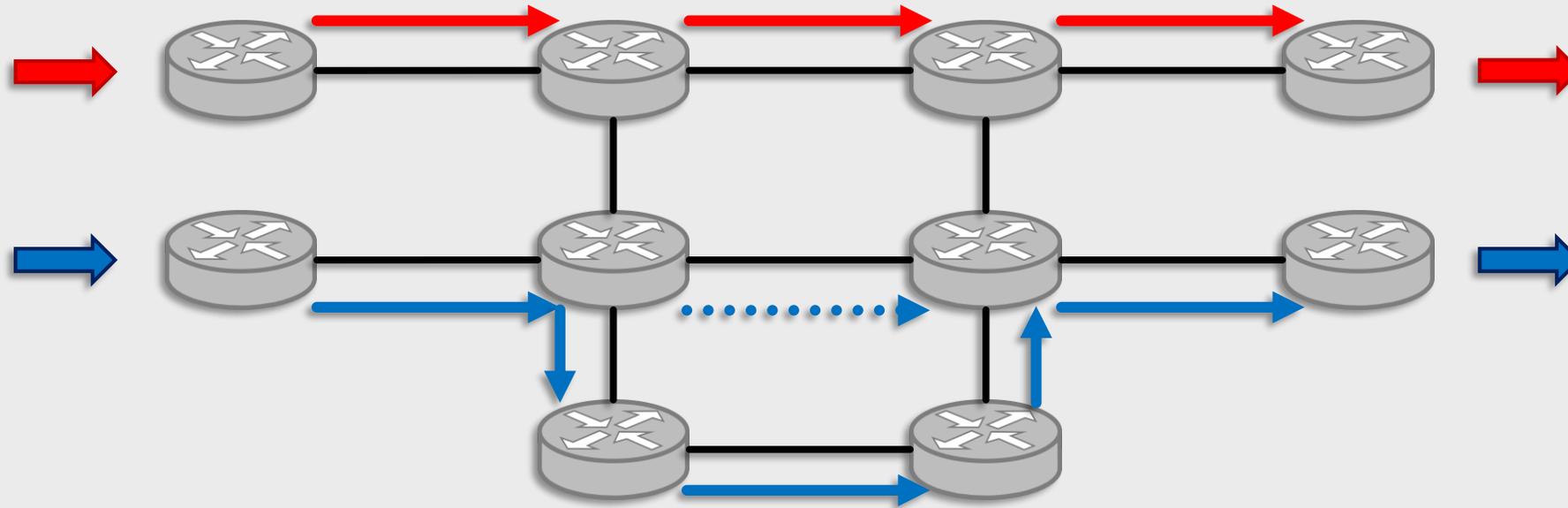
Étape 1

Étapes de reconfiguration



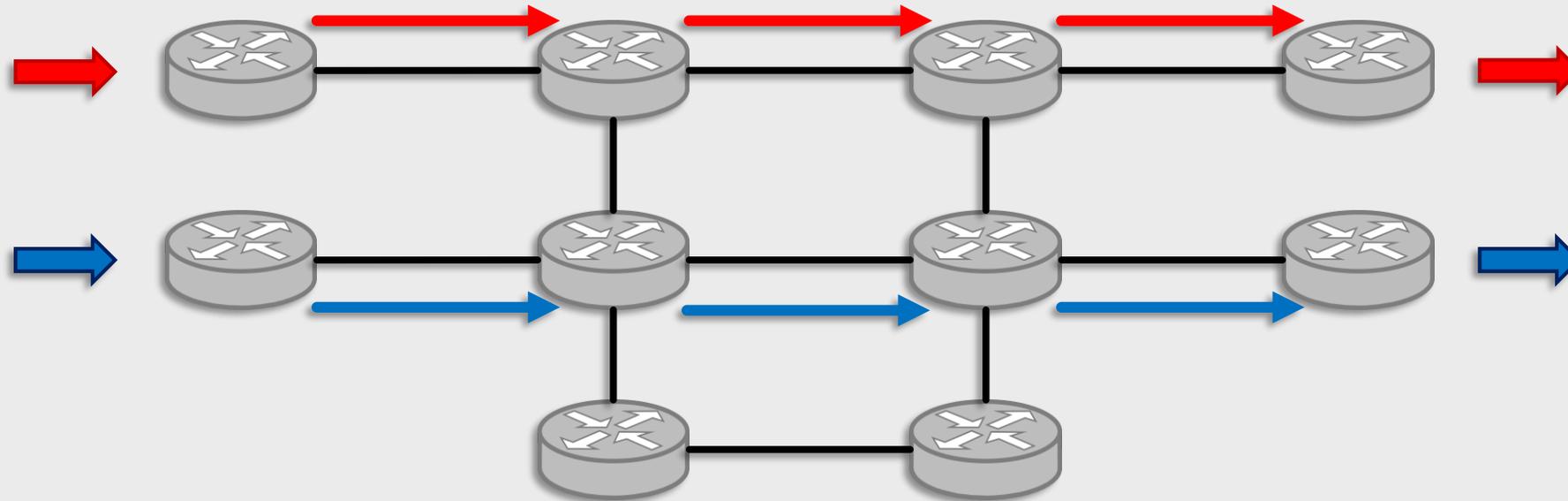
Étape 2

Étapes de reconfiguration



Étape 2 → Étape 3

Étapes de reconfiguration

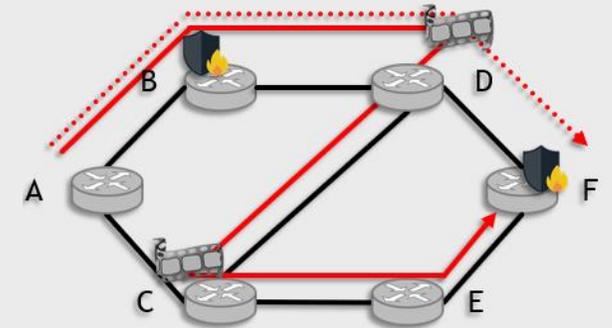


Étape 3 : Étape Finale

ILP : Reconfiguration

- Objectif de la reconfiguration :

$$\min \underbrace{\sum_{d \in D} \sum_{(u,v) \in E} \sum_{i=0}^{|c_d|} bw_d \cdot \varphi_{uv,i}^{d,T}}_{\text{red line}} + \beta \cdot \underbrace{\sum_{u \in V} \sum_{f \in F} c_{u,f} \cdot z_{u,f}^T}_{\text{blue line}}$$



- Contraintes d'intégrité du flow :

$$\varphi_{uv,i}^{d,t} \leq x_{uv,i}^{d,t}$$

$$\varphi_{uv,i}^{d,t-1} \leq x_{uv,i}^{d,t}$$

$$\varphi_{uv,i}^{d,t} + \varphi_{uv,i}^{d,t-1} \geq x_{uv,i}^{d,t}$$

For each Demand $d \in D$, Link $(u,v) \in E$,
Layer $i \in \{0, \dots, |c_d|\}$, time step $t \in \{1, \dots, T\}$.

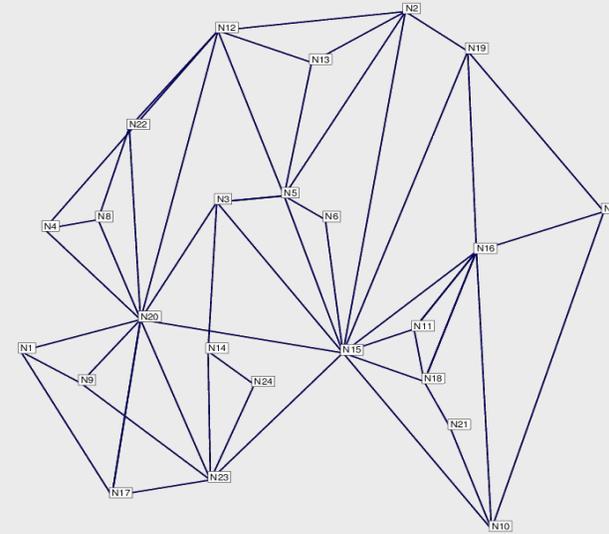
- Contraintes de capacité :

$$\sum_{d \in D} bw_d \sum_{i=0}^{|c_d|} x_{uv,i}^{d,t} \leq C_{uv}.$$

For each Link $(u,v) \in E$,
time step $t \in \{1, \dots, T\}$.

Cadre expérimental

- Réseau ta1 de SNDlib :
 - 25 Nœuds
 - 55 Arêtes
- 5 fonctions virtuelles différentes
- 2 scénarios différents :
 - Faiblement et Fortement congestionné
- Solver CPLEX
- Machine de test:
 - Intel Xeon 8 Threads 3,2Ghz
 - 64 Go de RAM

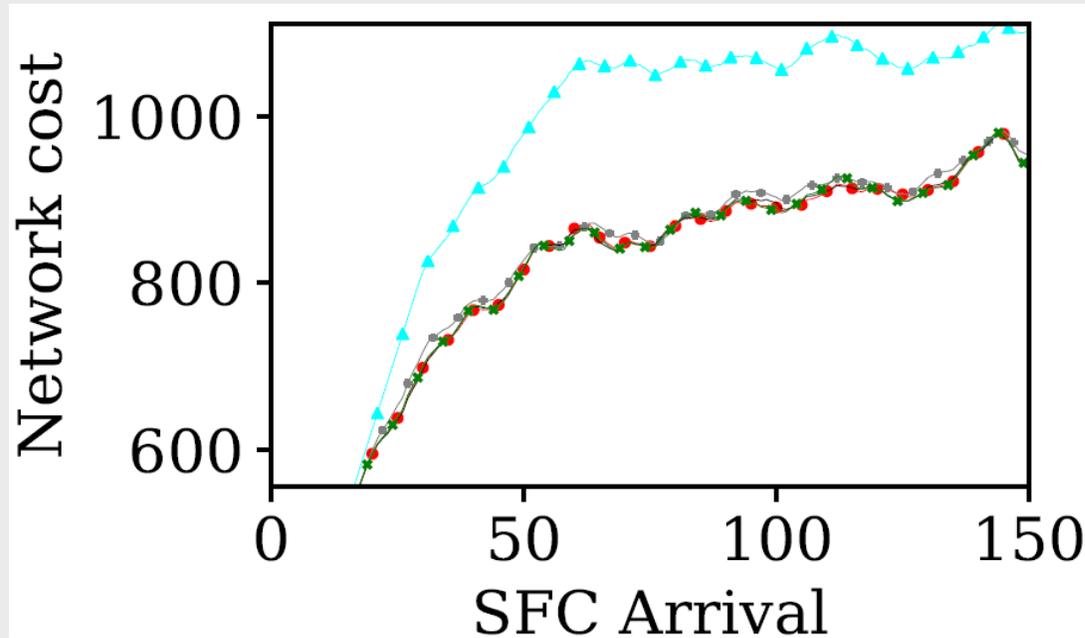


- 3 Algos :
 - No-Reconf
 - Breaking-Bad
 - Break-Free

Scénario faiblement congestionné

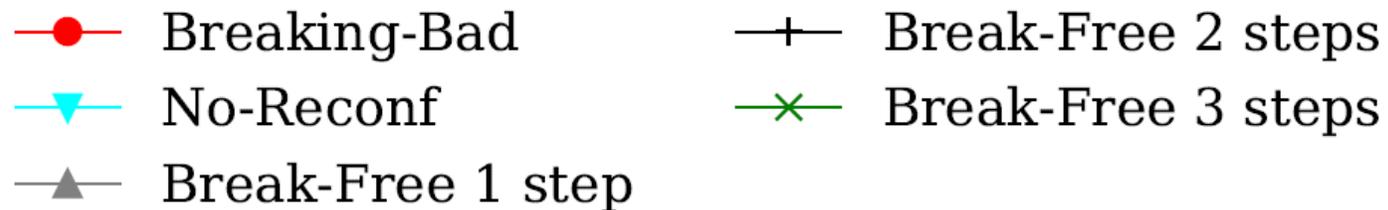
- 100% d'acceptation de SFC
- On cherche à minimiser les coûts opérationnels
- Reconfigurations régulières

Scénario faiblement congestionné

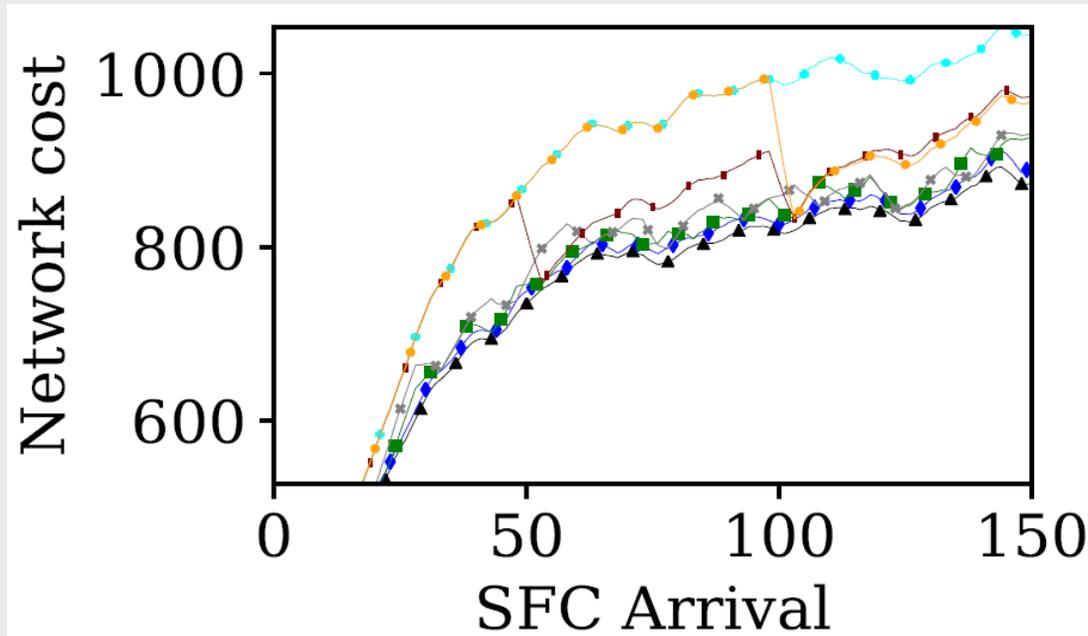


➤ 19% de coût en moins

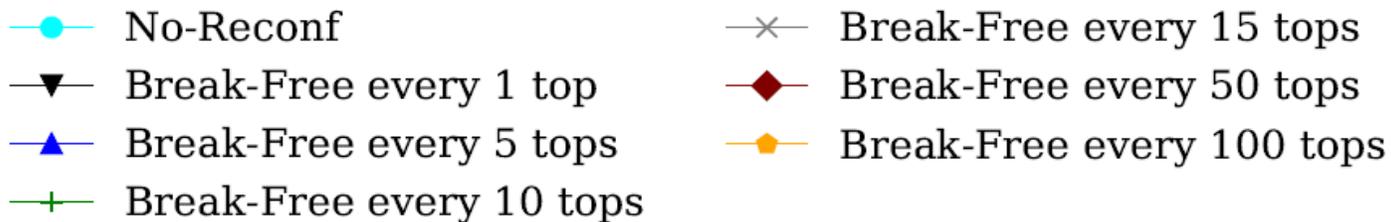
➤ En une étape on est proche de l'optimal



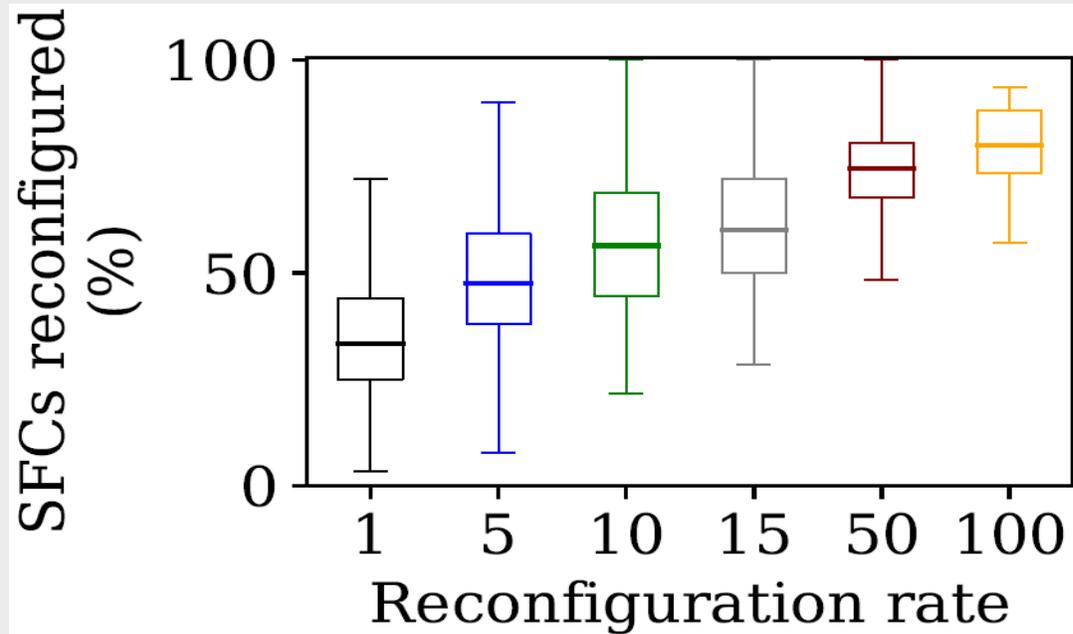
Scénario faiblement congestionné



- Reconfiguration tous les tops ou tout les 5 tops sont équivalentes
- Reconfiguration tous les 10 et 15 tops ne sont que 5% moins bonnes



Scénario faiblement congestionné



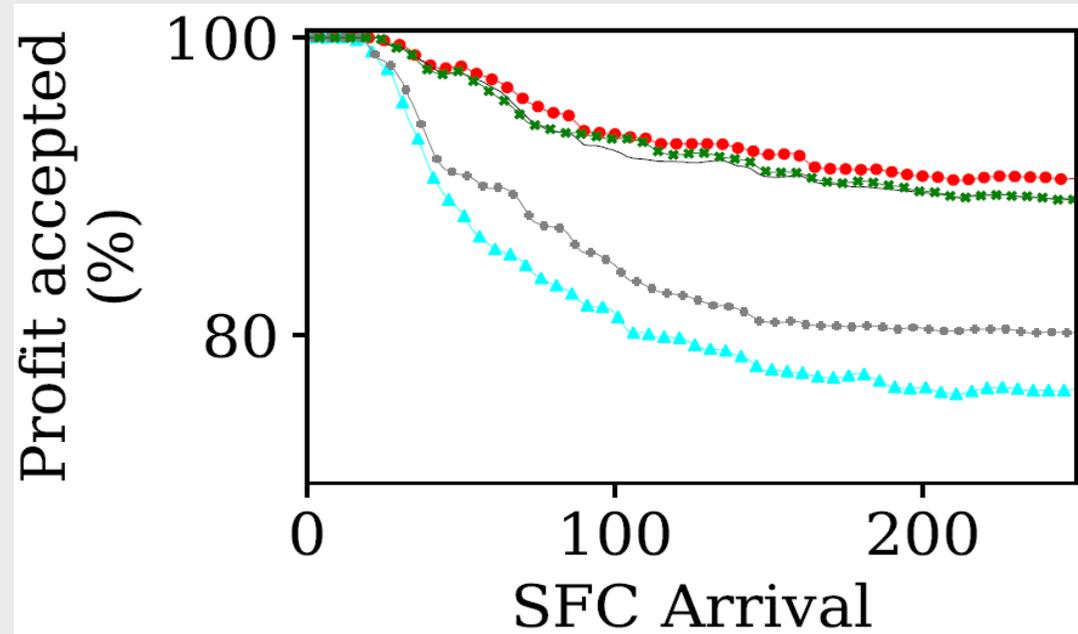
➤ 35% tout les tops à 80% tout les 100 tops

➤ Reconfigurations sont **gratuites** avec Break-Free

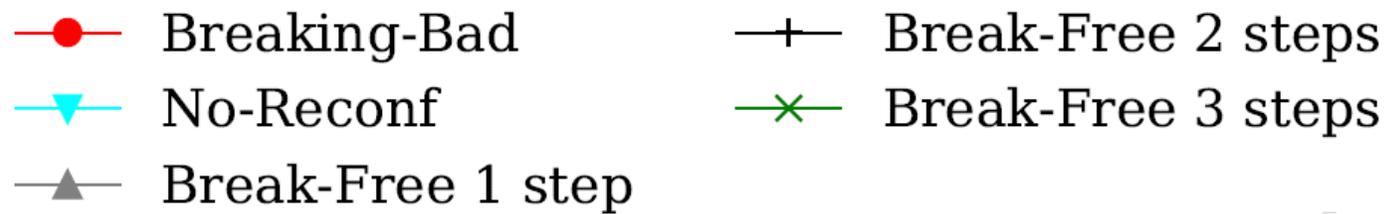
Scénario fortement congestionné

- On cherche à accepter le plus de demandes tout en minimisant les coûts
- Reconfigurations régulières
- Reconfigurations lors de rejets

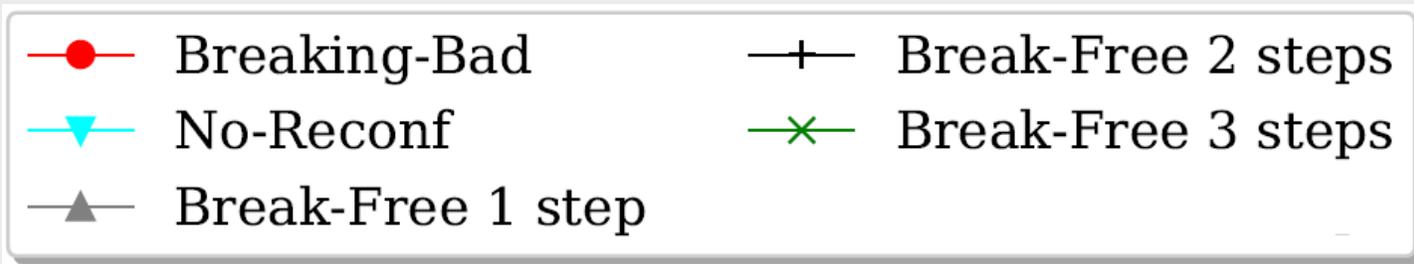
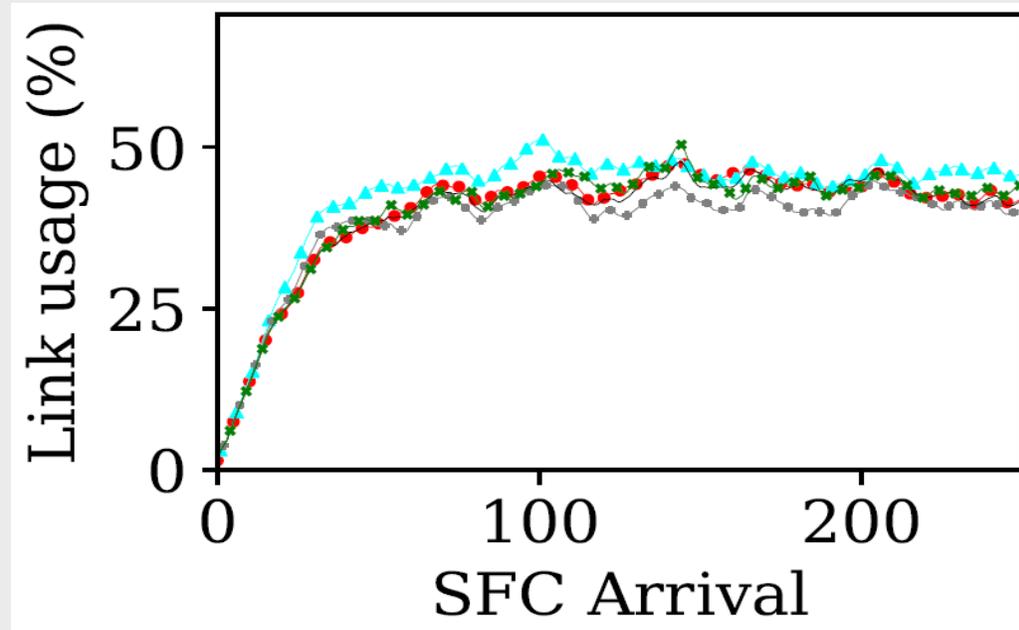
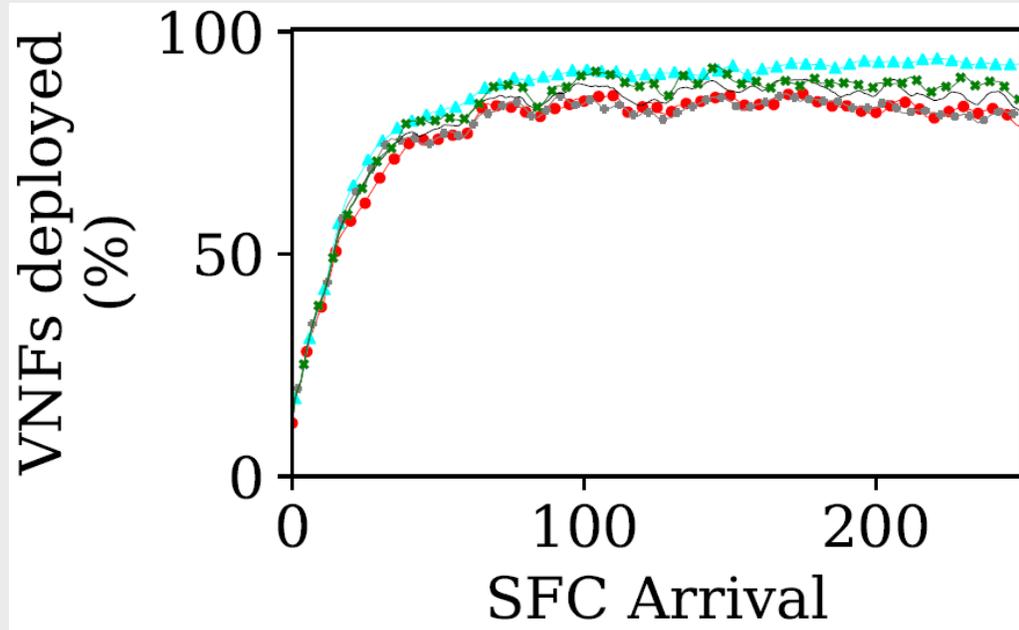
Scénario fortement congestionné



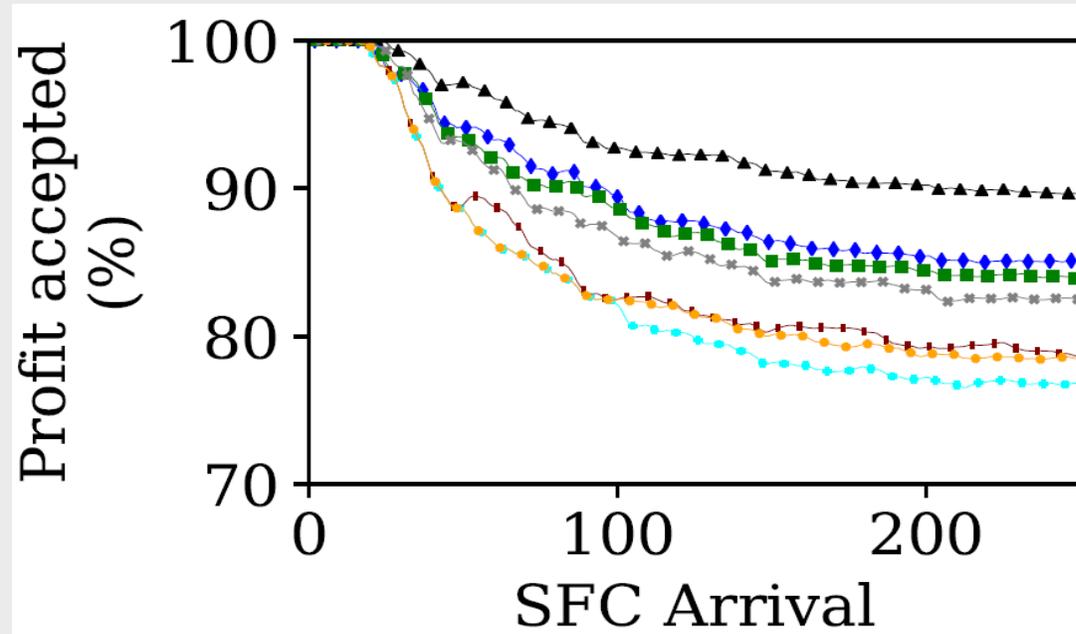
- De 78% à 90% d'acceptation
- Break-Free 2,3 steps et Breaking-Bad équivalents



Scénario fortement congestionné

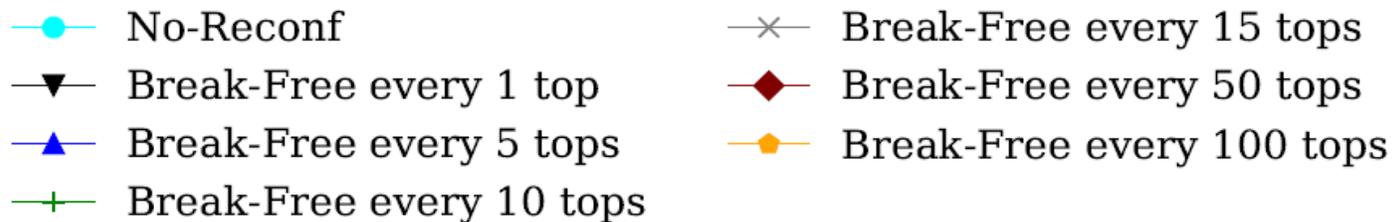


Scénario fortement congestionné

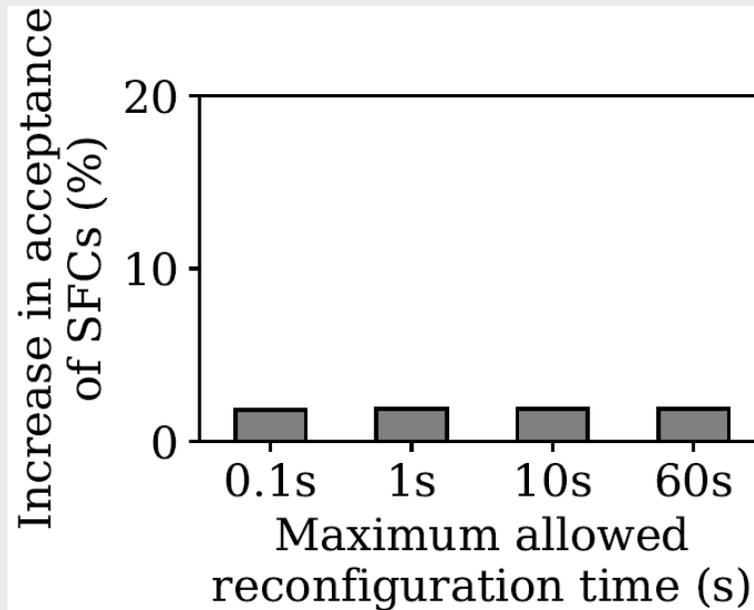


➤ 5% de profit en moins entre 1 et 5 tops

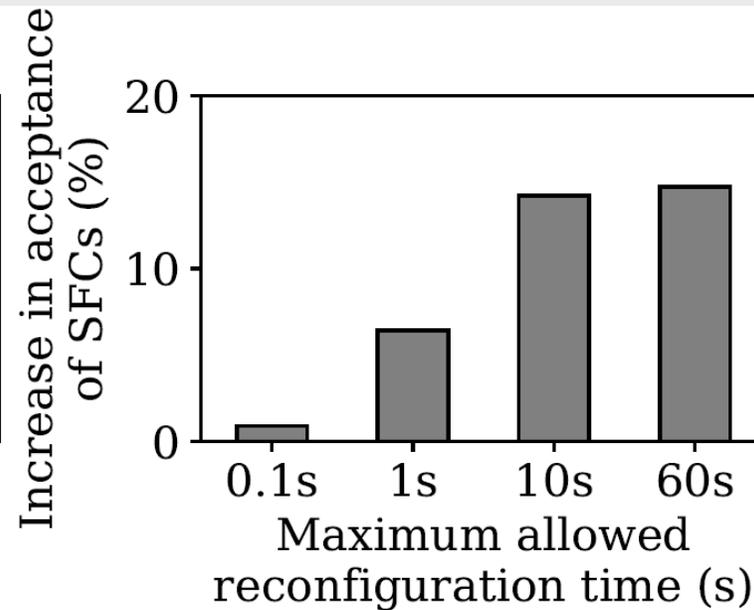
➤ Incidence importante de la fréquence



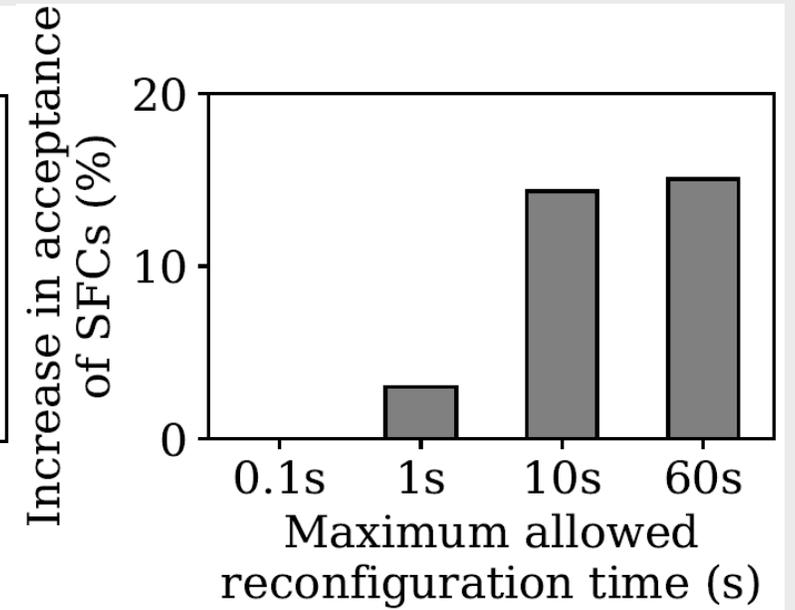
Scénario fortement congestionné



Reconfiguration en une étape



Reconfiguration en deux étapes



Reconfiguration en trois étapes

Conclusion

➤ Aboutissement :

- Un ILP qui permet une allocation dynamique rapide
- Un ILP qui permet :
 - Diminution des coûts opérationnels jusqu'à environ 20%
 - Augmentation de l'acceptation des SFCs jusqu'à environ 20%
- Aucune interruption  Pas de dégradation des services

➤ Prolongement :

- Passage à de la génération de colonnes

Merci de votre attention

➤ Me joindre :

➤ adrien.gausseran@inria.fr

➤ Rapport de recherche :

➤ <https://hal.inria.fr/hal-01963270/file/RR-9241.pdf>