

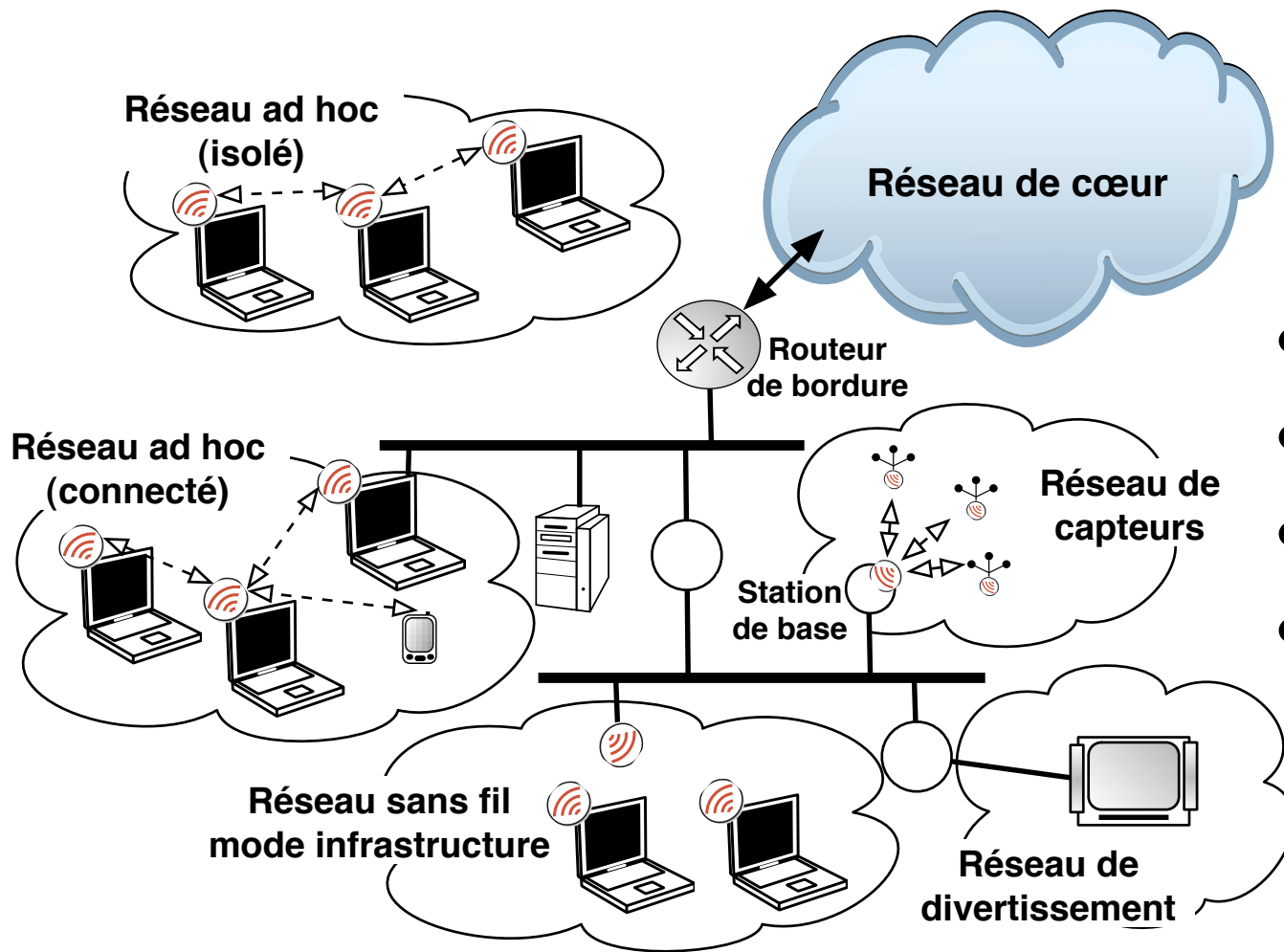
Les réseaux sans fil spontanés pour l'Internet Ambient

Vincent Untz

5 décembre 2007



Réseau spontané de bordure



- Hétérogène
- Sans administration
- Autonome
- Réseau local

Plan

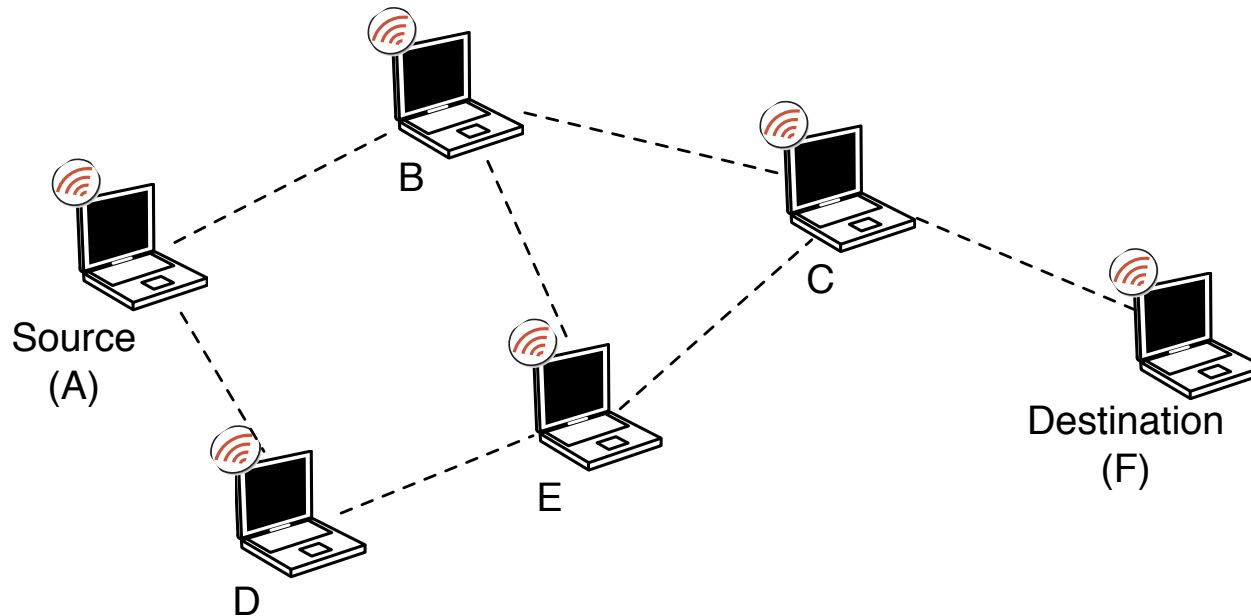
- État de l'art
- Architecture
- Implémentation
- Expérimentations
- Conclusion

Travaux liés

- Réseaux sans fil ad hoc mobiles (MANET)
 - sans infrastructure
 - sauts multiples
 - routage réactif ou proactif
 - fondé sur la philosophie IP
- Nouvelles architectures
 - proches de MANET, mais ancrées dans l'expérimentation

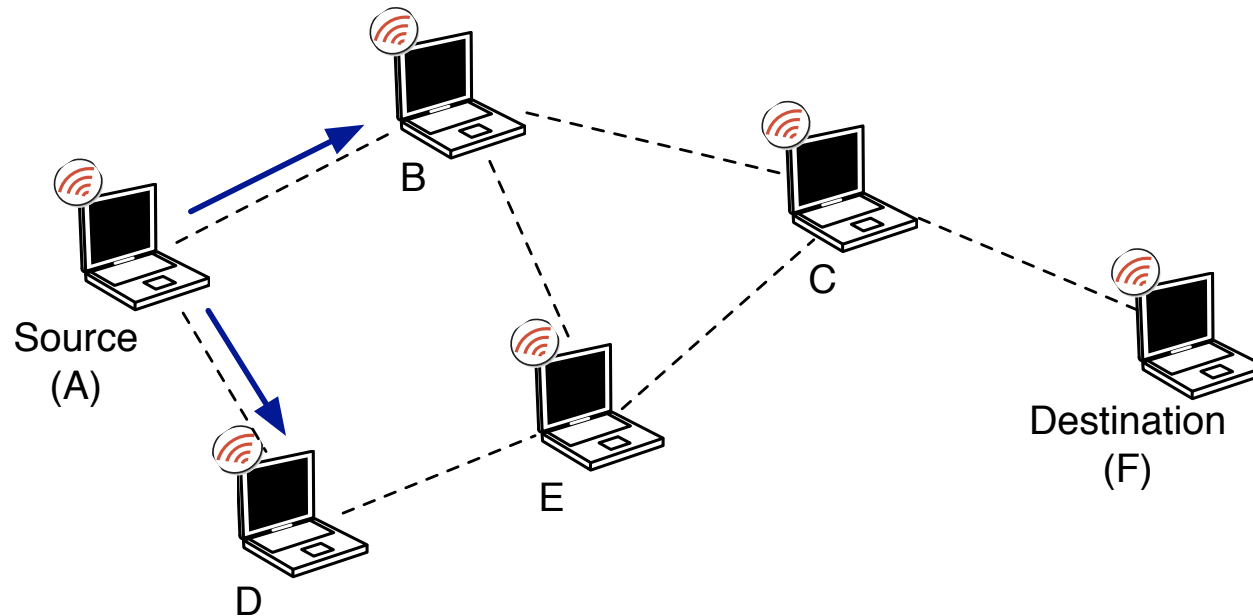
Principe du routage réactif

- Deux messages : Route Request et Route Reply



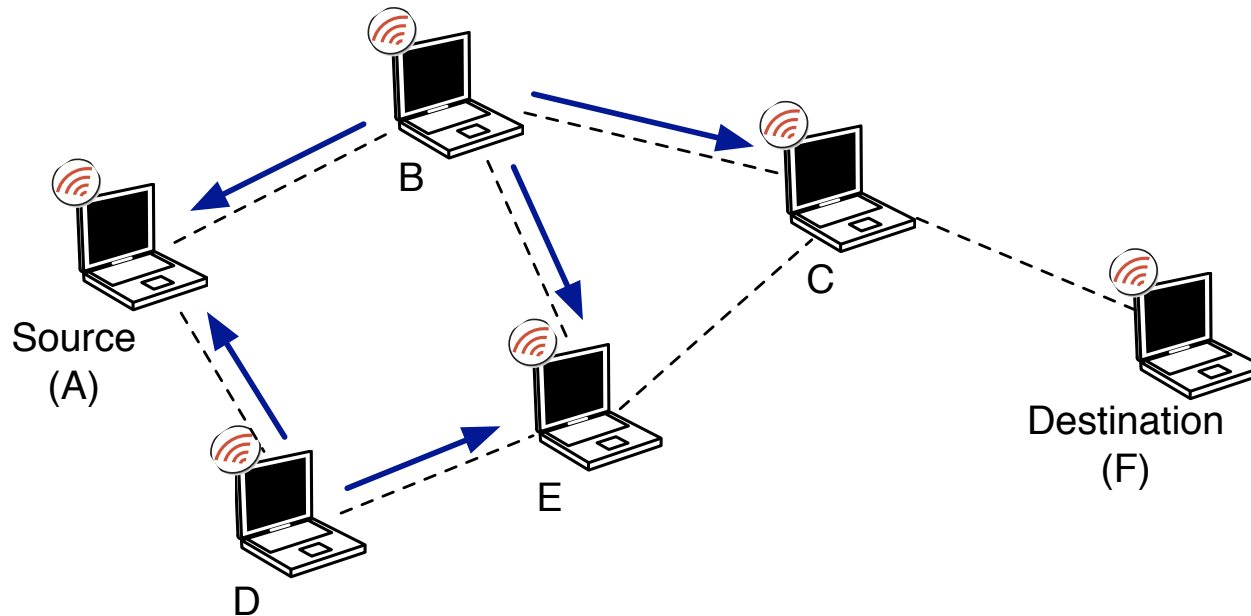
Principe du routage réactif

- Deux messages : **Route Request** et **Route Reply**



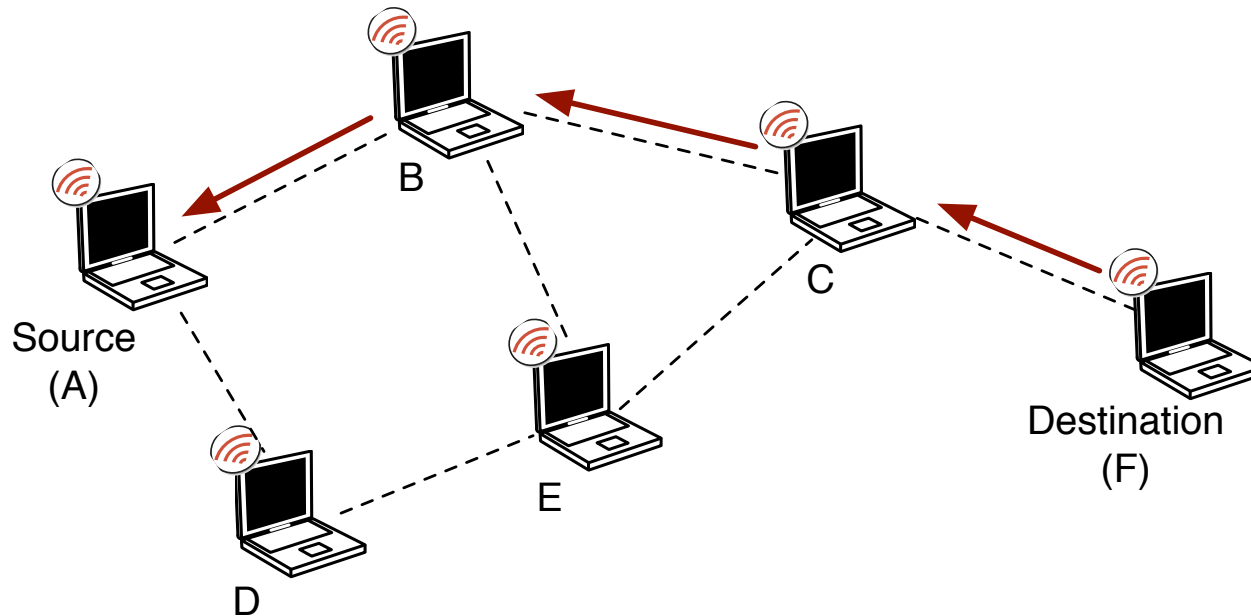
Principe du routage réactif

- Deux messages : **Route Request** et **Route Reply**



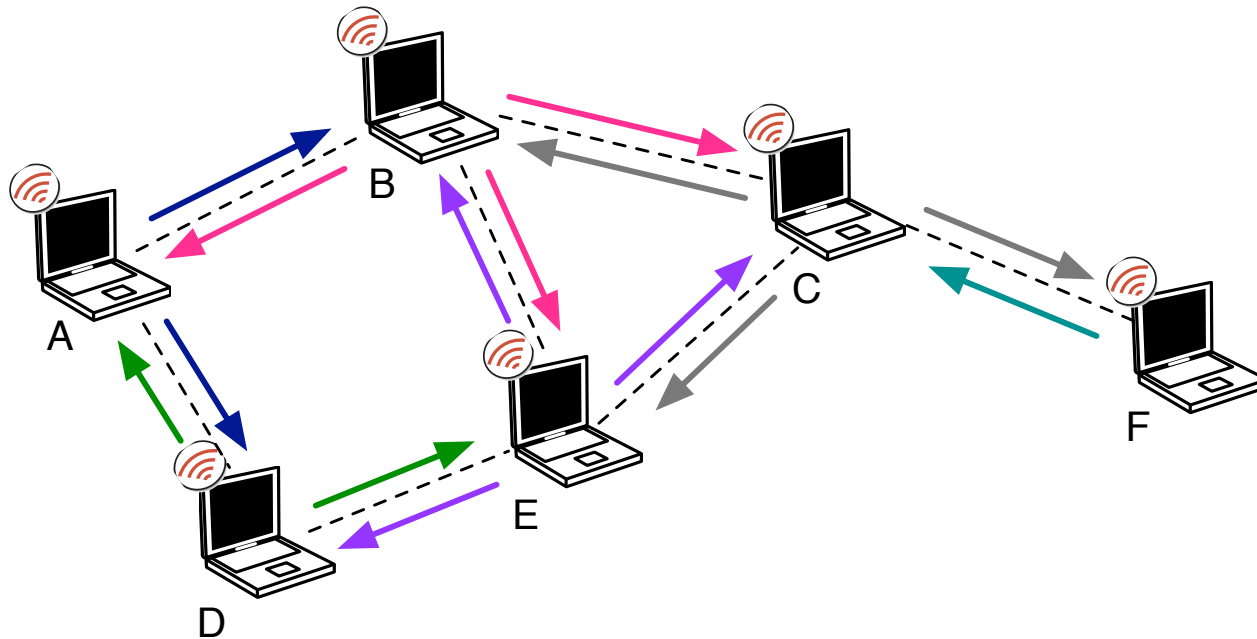
Principe du routage réactif

- Deux messages : Route Request et **Route Reply**



Principe du routage proactif

- Chaque station connaît la topologie du réseau



Modèle TCP/IP

Application (5)

Transport (4)

Réseau (3)

Liaison (2)

Physique (1)

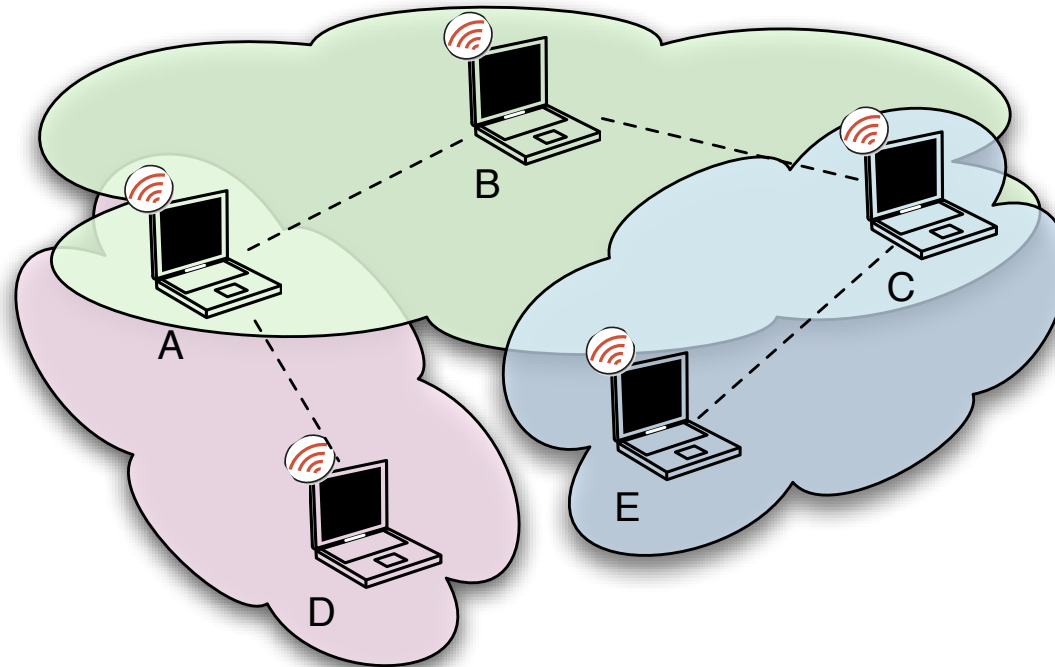
- MANET : couche 3
- Couche 3 est trop « haute »
- Lien entre couches 2 et 3 : ARP

Lunar

- Objectif : effectuer des expérimentations
- Routage réactif similaire à AODV
 - mais fonctionnant réellement comme une extension d'ARP
- Pas de maintenance des routes
- Intégration avec DHCP
- Intégration avec DNS

Ananas

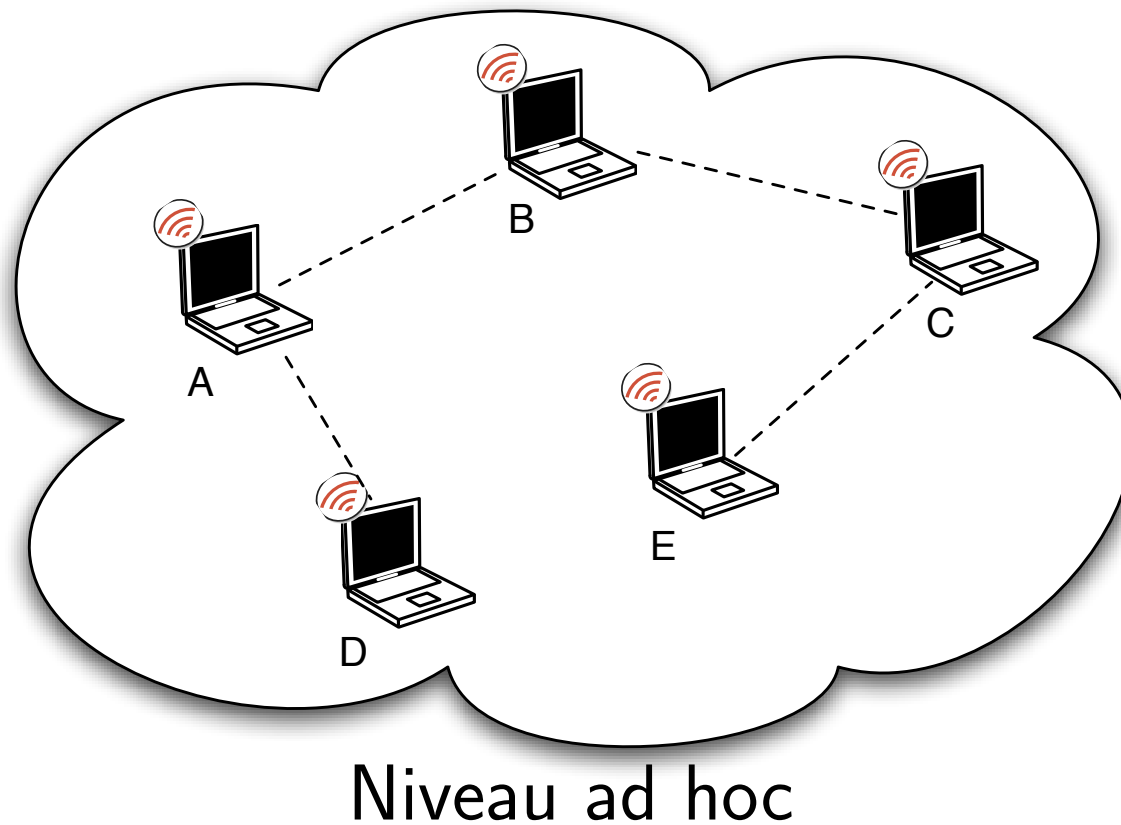
- Objectif : gérer la mobilité



Niveau liaison

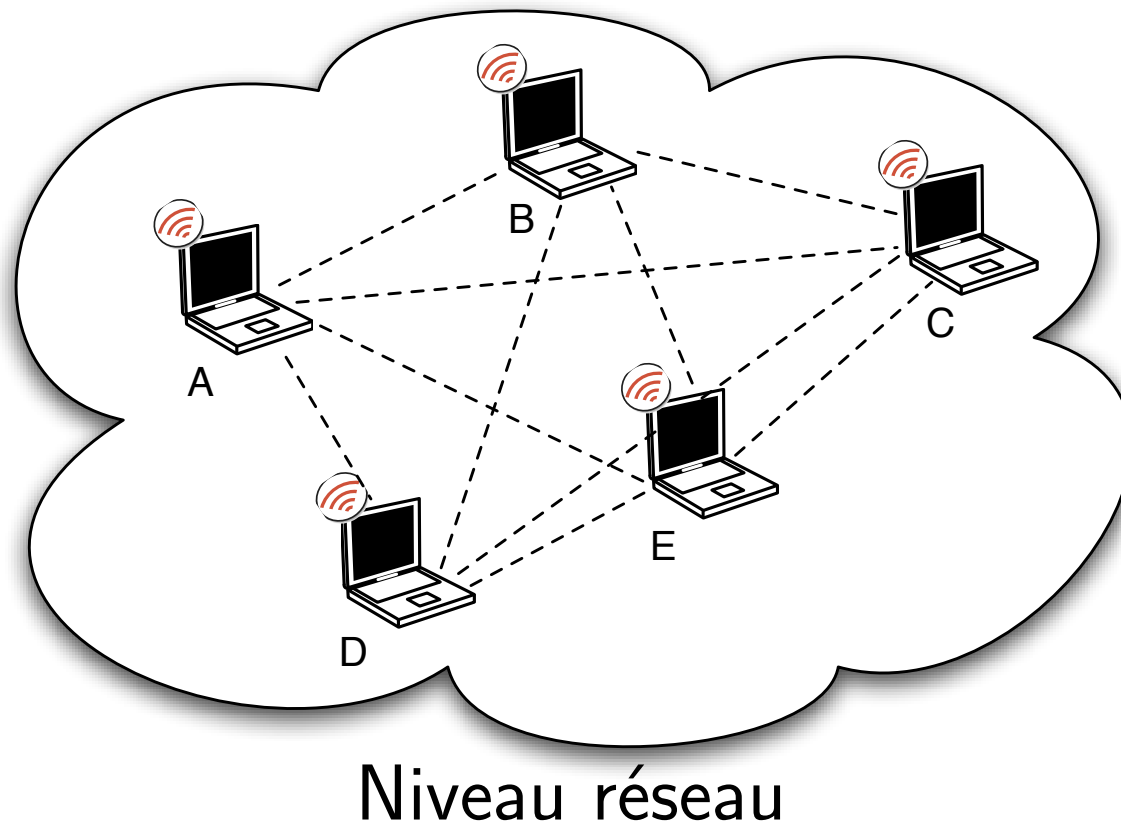
Ananas

- Objectif : gérer la mobilité



Ananas

- Objectif : gérer la mobilité



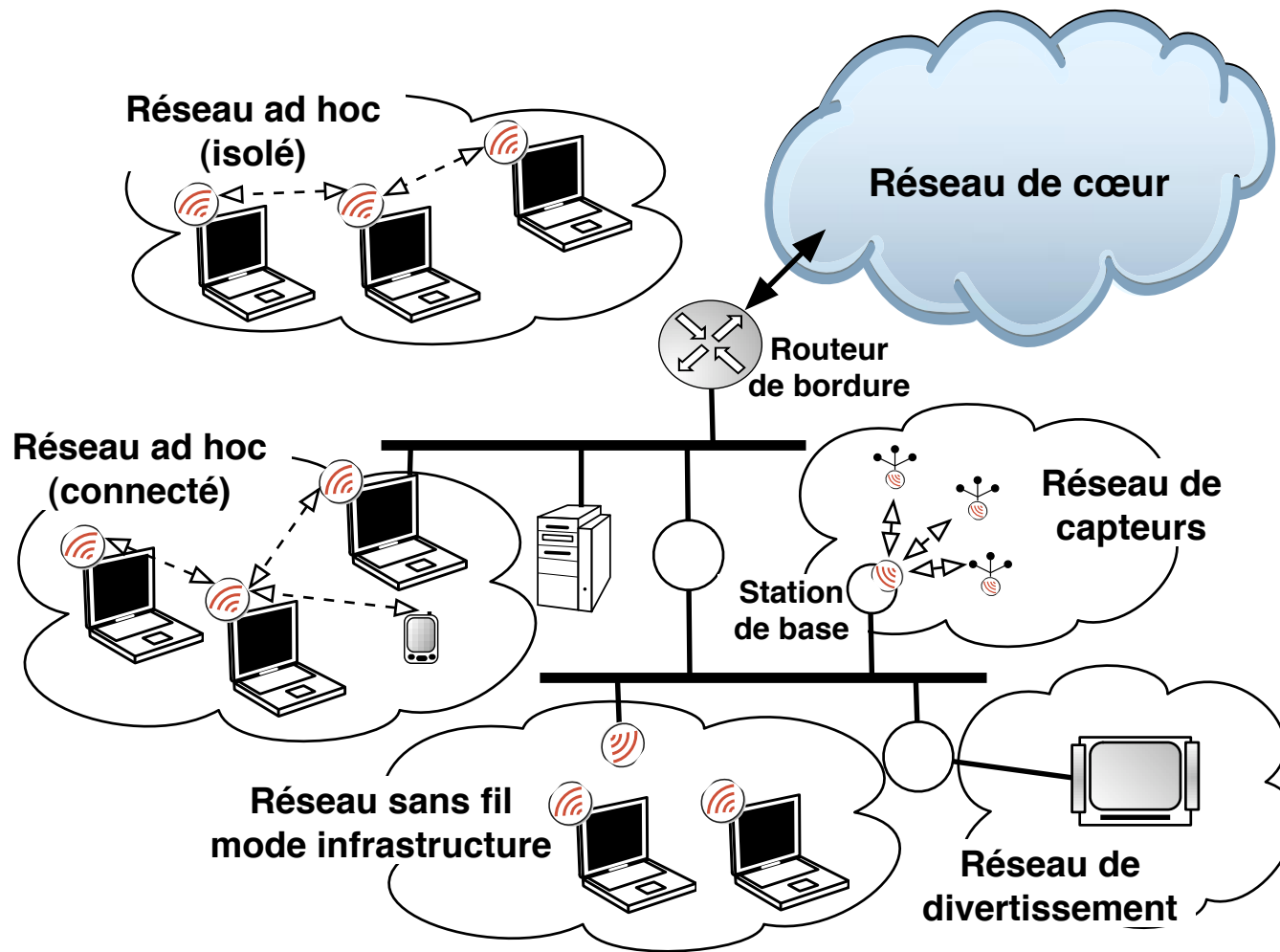
Bilan

- MANET
 - protocoles de routage bien étudiés
 - contexte peu adapté
- Lunar et Ananas
 - vision du réseau comme un réseau local
 - introduction d'une couche 2.5

Plan

- État de l'art
- **Architecture**
- Implémentation
- Expérimentations
- Conclusion

Cahier des charges



- Réactivité
- Auto-organisation
- Robustesse
- Performances
- Compatibilité

Notion de connexion

- Représentation du lien entre deux nœuds de bout-en-bout
 - un nœud intermédiaire n'est pas libre d'effectuer des choix
- Unidirectionnalité
- Par flot

Principaux choix

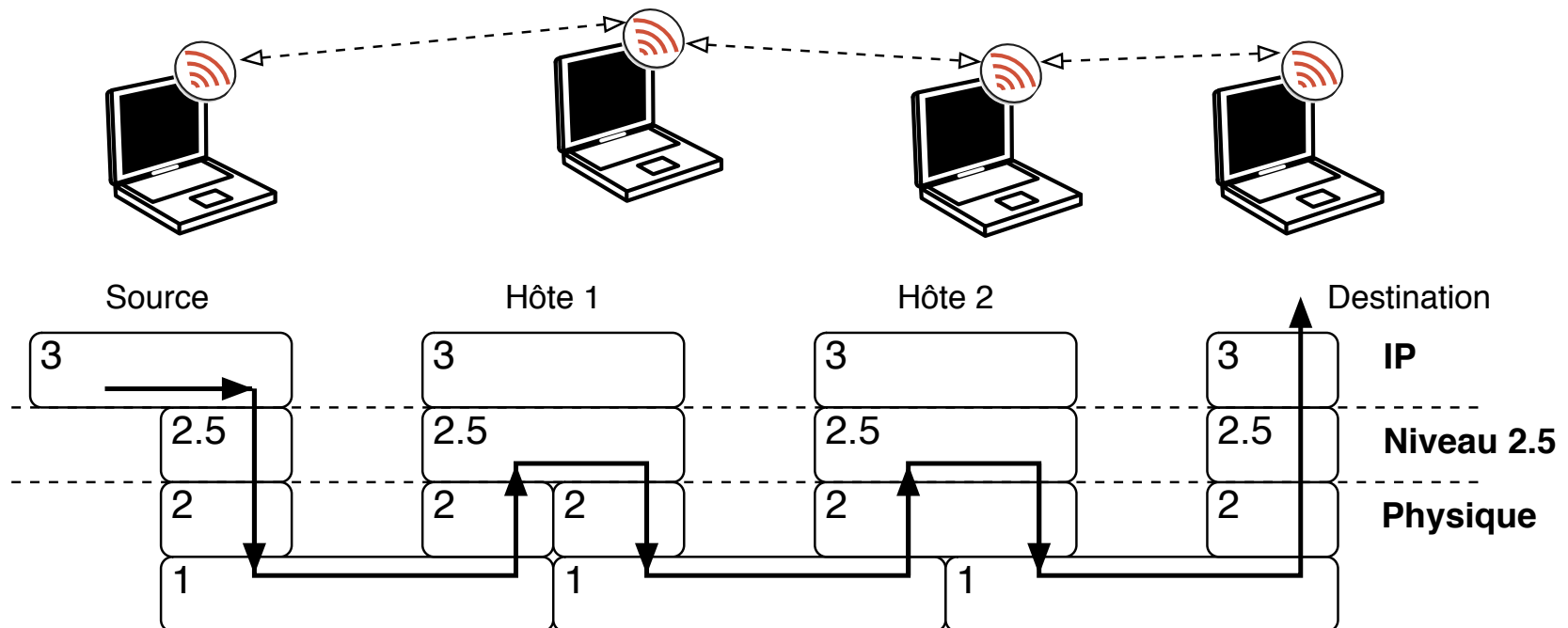
- Réactivité
 - découverte des routes
- Mode connecté
 - une connexion active est une connexion à maintenir
 - chemins multiples pour de multiples connexions
 - chemins de secours

Vers une couche 2.5

- Problème de la virtualisation d'un réseau multi-sauts sous forme de réseau local
- Modèle actuel inadapté
 - couche 2 : n'a pas pour but de faire de l'acheminement multi-sauts
 - couche 3 : dans un réseau local, tous les nœuds sont perçus comme voisins

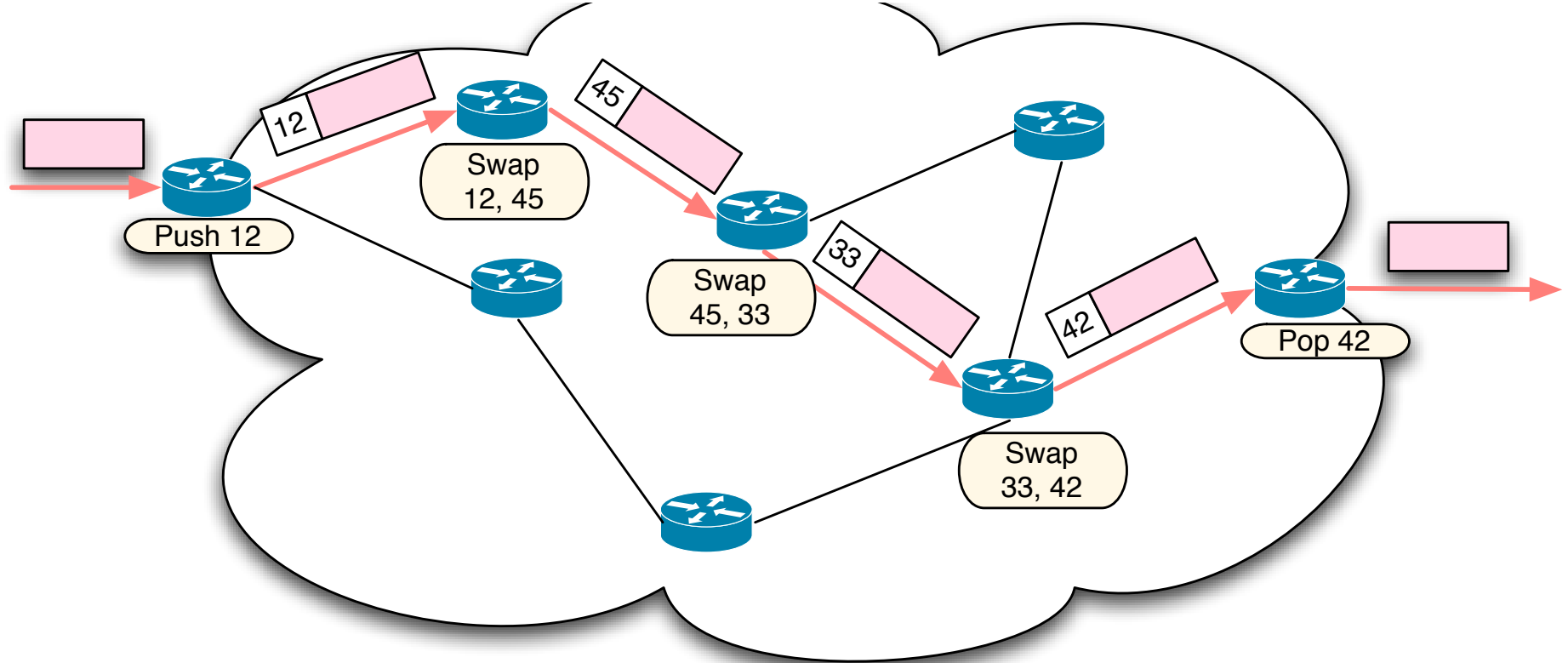
Vers une couche 2.5

- Introduction d'une nouvelle couche
- Encapsulation avec un en-tête



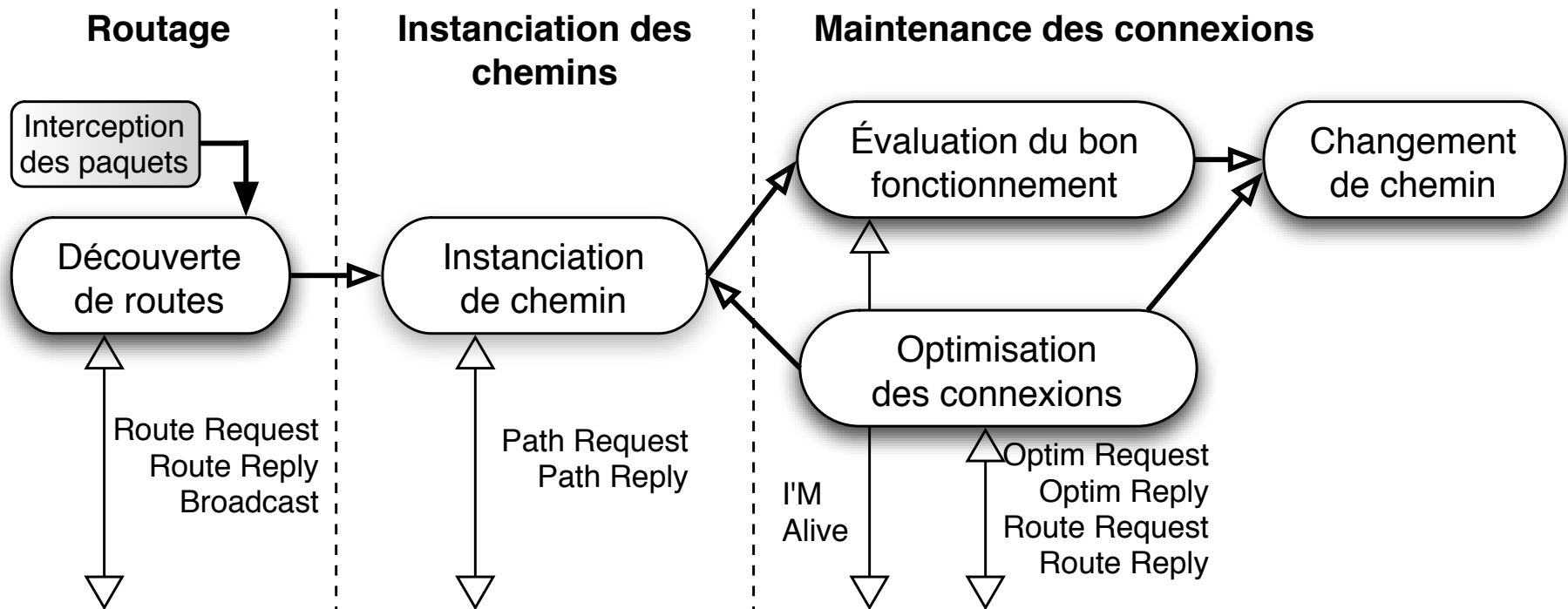
MPLS, une couche 2.5

Cœur de réseau utilisant MPLS



- Chemin MPLS le long d'une route
- Offre de nouvelles possibilités par rapport à IP

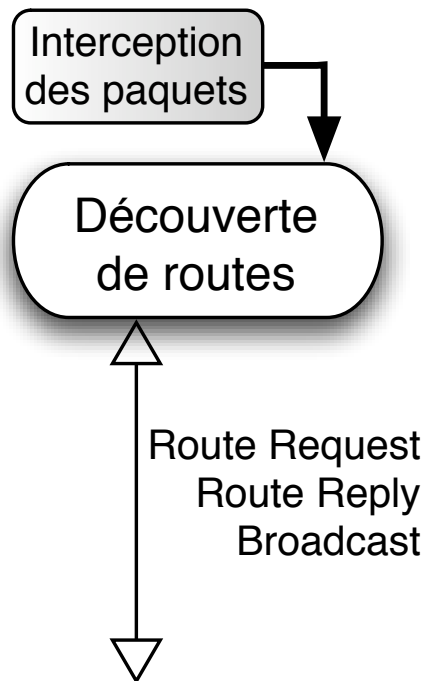
Architecture proposée



Routage

- Objectif :
 - trouver une ou plusieurs routes vers une destination, avec une métrique permettant d'évaluer les résultats
 - si plusieurs routes, sélection de la meilleure et utilisation d'une autre route comme chemin de secours
 - découverte de routes pour chaque connexion
→ chemins multiples

Protocole de routage



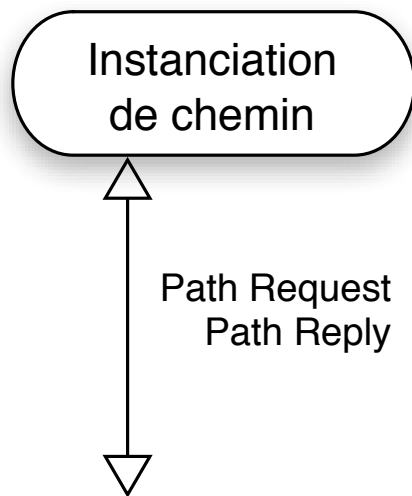
- Paquet intercepté s'il n'y pas de chemin vers la destination
- Tout protocole de routage peut être utilisé
- Définition d'un protocole réactif
- Gestion des paquets de diffusion

Protocole de routage

- Minimisation de l'impact en tant que charge réseau
 - Limitation de la portée des requêtes avec une recherche en anneaux croissants
 - Diffusion à optimiser avec l'utilisation de relais multi-points

Instanciation des chemins

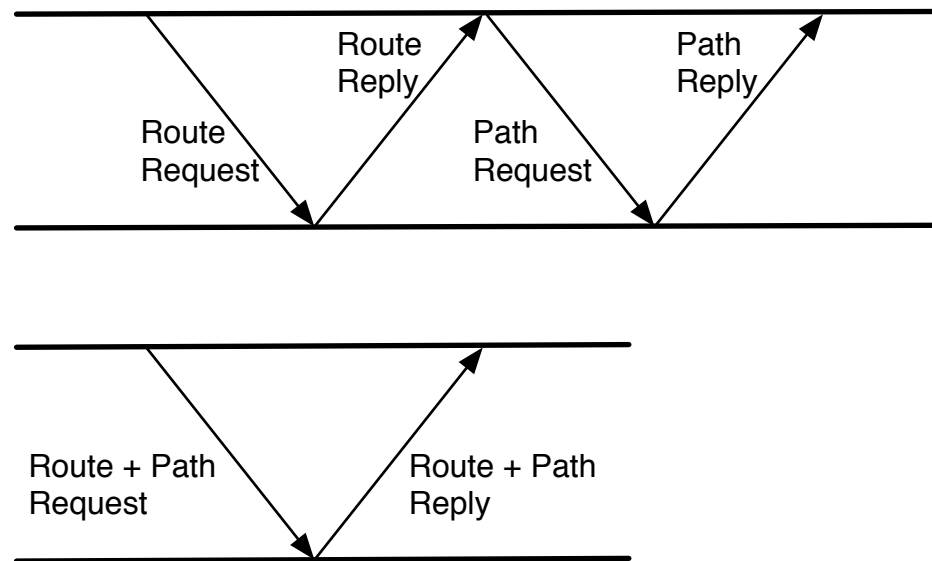
- Objectif : créer le chemin MPLS et les ressources locales sur chaque nœud intermédiaire



- Instanciation réalisée à la demande

Instanciación de caminos

- Couplage possible entre instanciación del camino y la respuesta para la descubierta de rutas

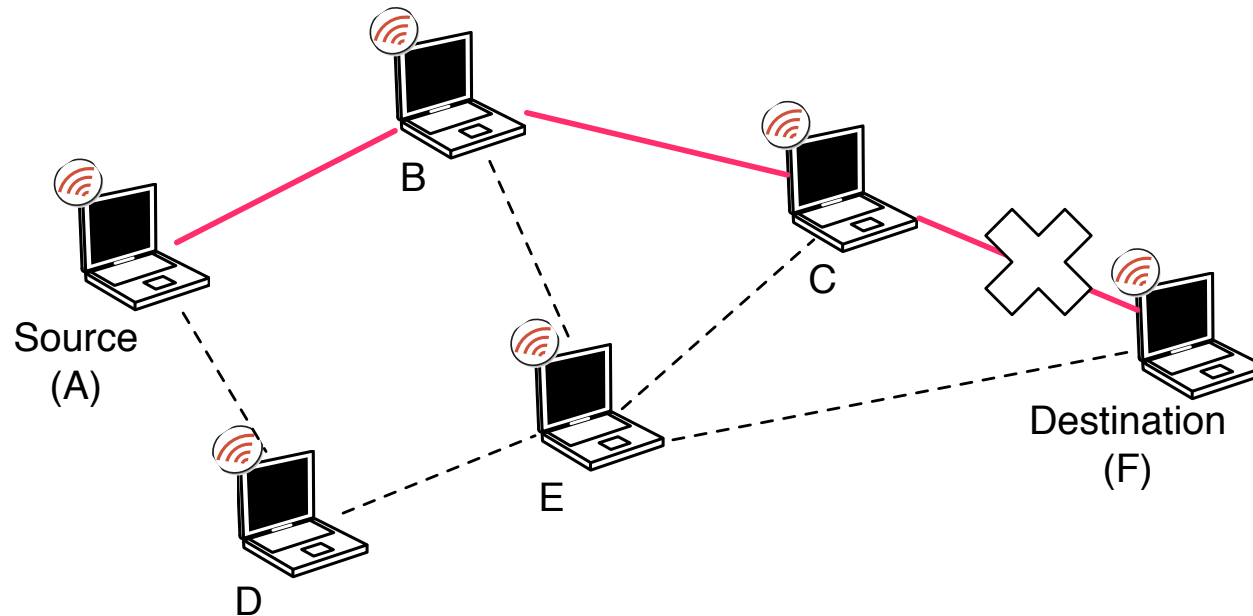


Maintenance des chemins

- Chaque station envoie localement des informations sur les chemins qui la traversent
 - messages l'M Alive
 - statistiques utiles pour le calcul de métriques
- Aucun trafic global causé par ces vérifications
- L'absence de trafic déclenche la destruction d'un chemin

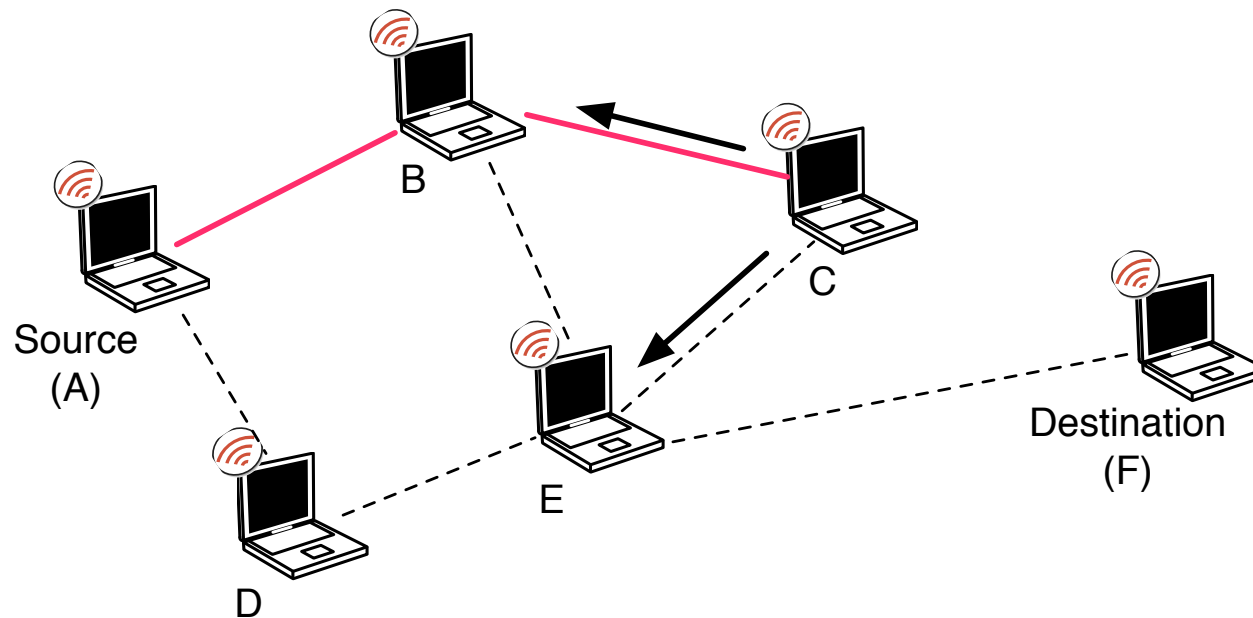
Maintenance des chemins

- La rupture d'un chemin est propagée jusqu'à la source



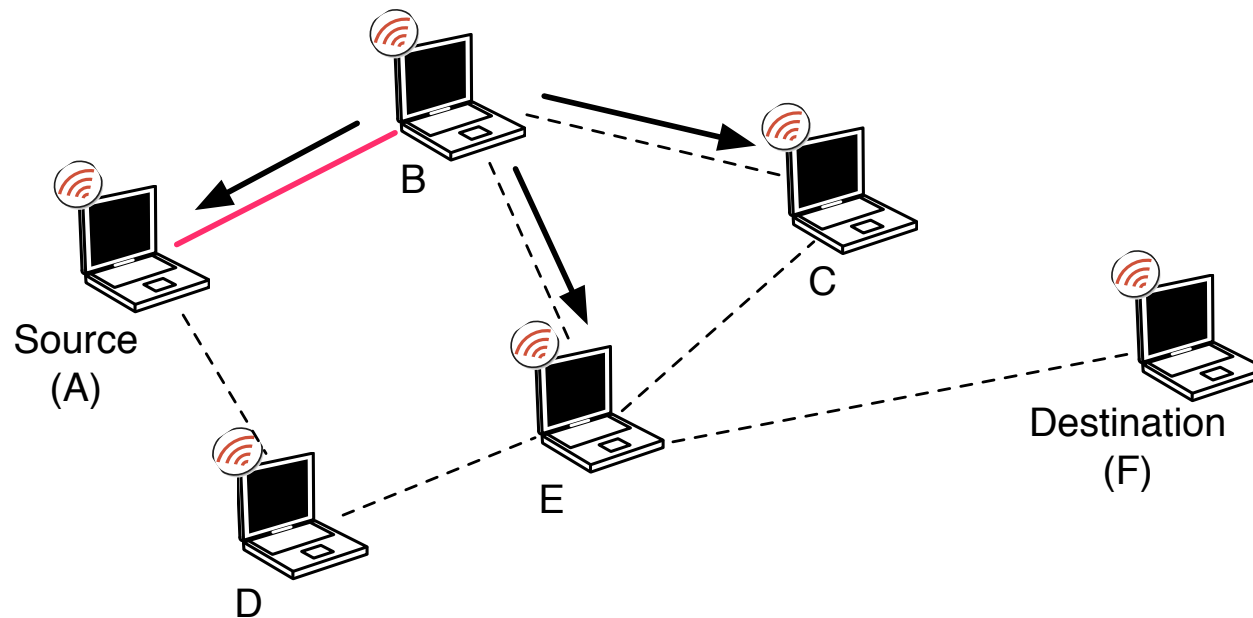
Maintenance des chemins

- La rupture d'un chemin est propagée jusqu'à la source



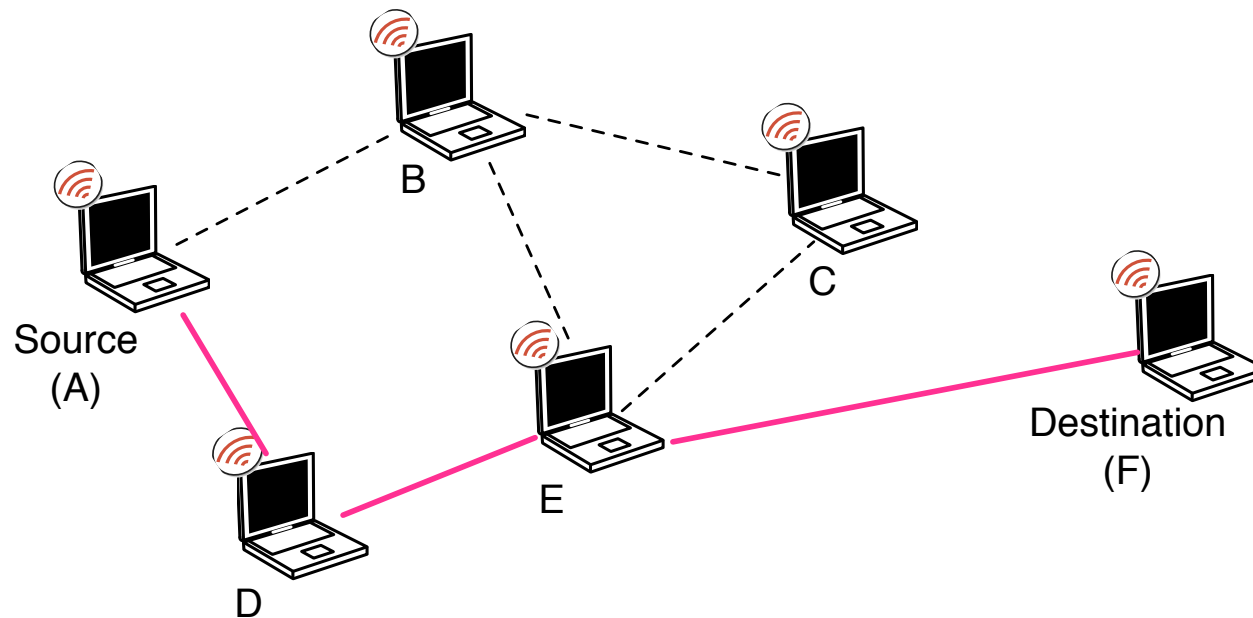
Maintenance des chemins

- La rupture d'un chemin est propagée jusqu'à la source



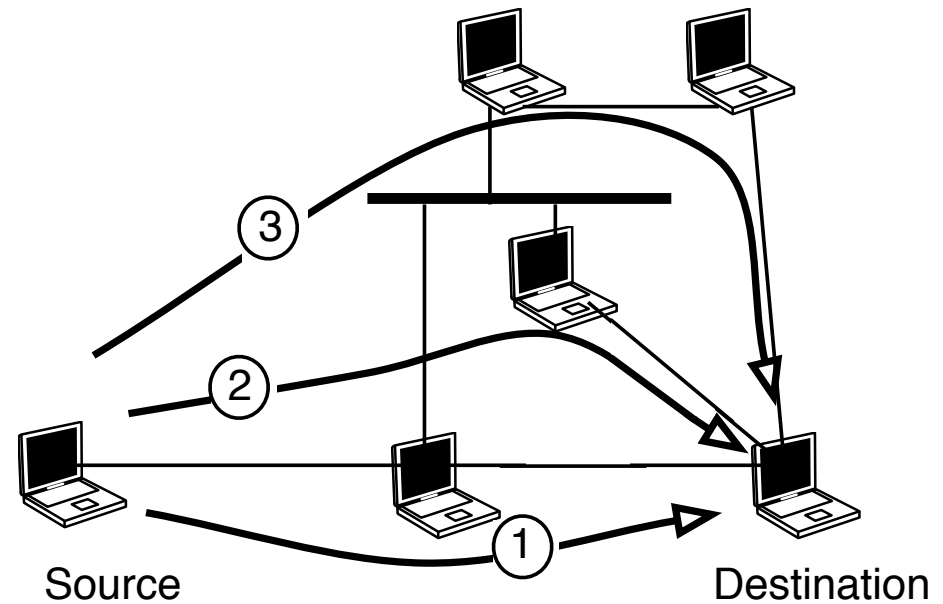
Maintenance des chemins

- La rupture d'un chemin est propagée jusqu'à la source



(Ré)Optimisation

- Recherche de routes alternatives
 - en arrière-plan
 - chemins de secours ou de meilleure qualité
- Changement de chemin est transparent



Plan

- État de l'art
- Architecture
- **Implémentation**
- Expérimentations
- Conclusion

Implémentation : MPLS

- MPLS Linux
 - cible mouvante
 - problèmes de stabilité et dans l'implémentation
- umpls
 - démon en espace utilisateur
 - Linux, FreeBSD
 - ... plus gourmand en ressources !

Implémentation : Lilith

- Implémentation de l'architecture proposée en utilisant MPLS
- Démon en espace utilisateur
- Flexible, mais sans obligation de configuration

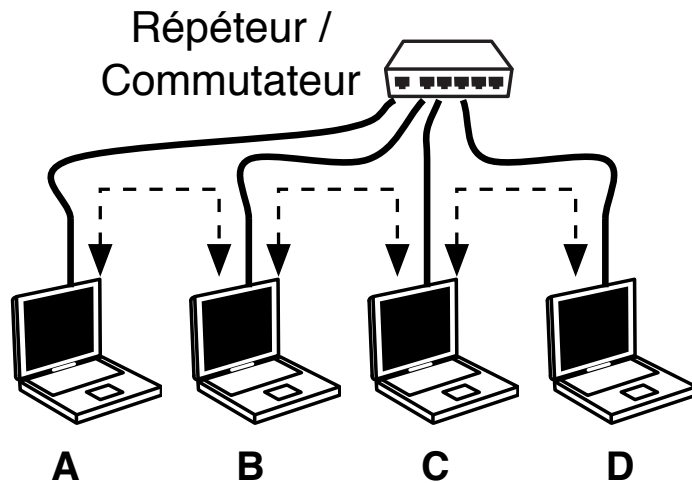
Déploiement et tests

- User Mode Linux
- Intégration avec DHCP
- Difficultés du monde réel
 - peu d'informations réellement disponibles pour une métrique
 - intégration invisible pour les applications
 - problème de la taille maximale des paquets (MTU)

Plan

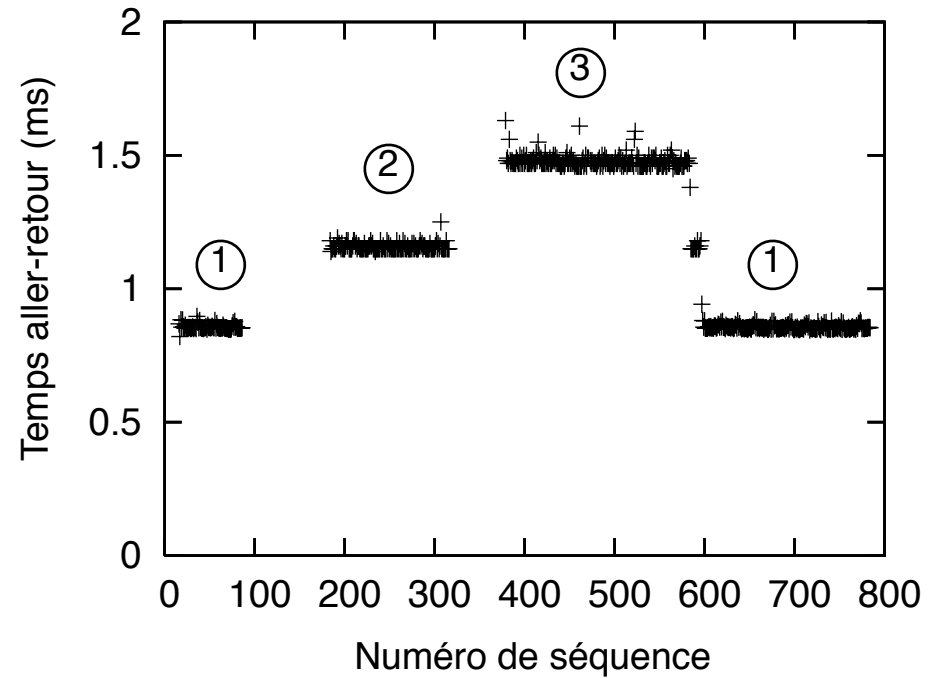
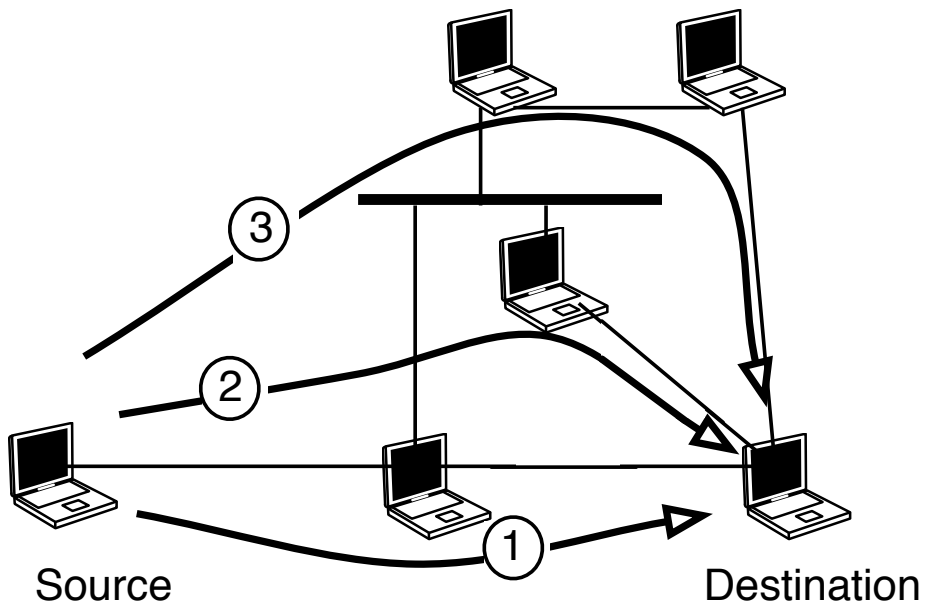
- État de l'art
- Architecture
- Implémentation
- **Expérimentations**
- Conclusion

Mesures de performances

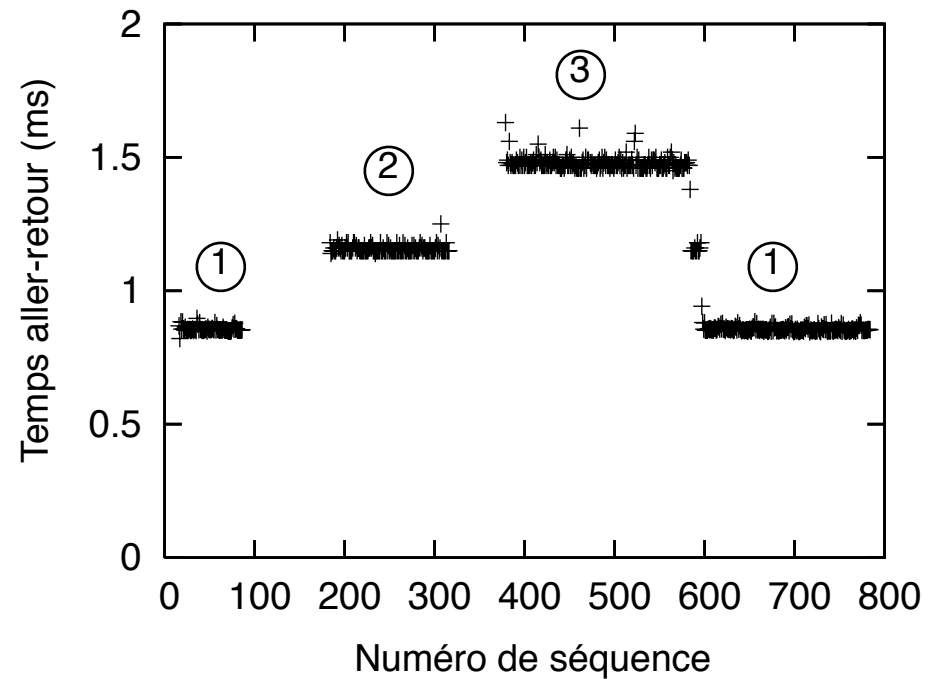
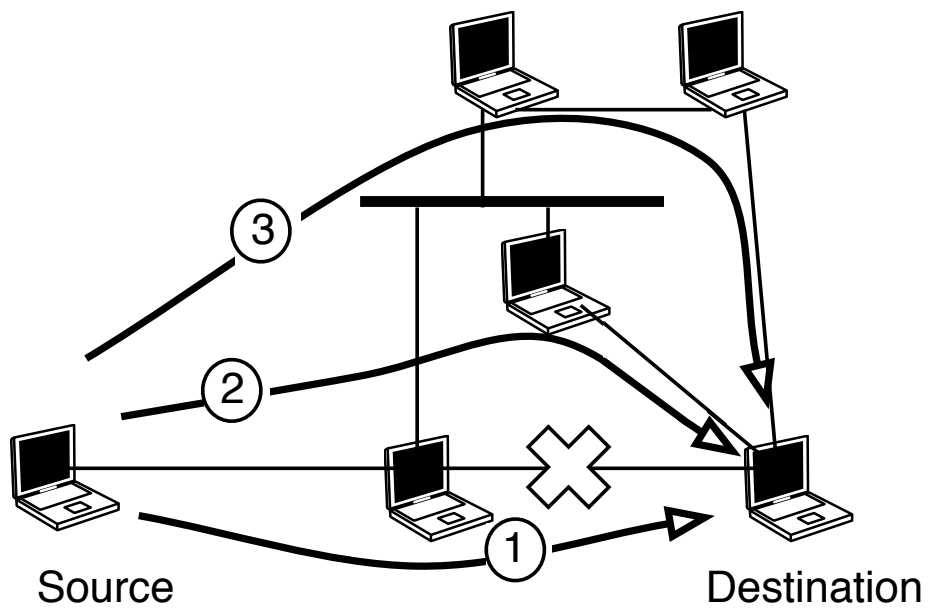


Test	IP	MPLS	Lilith (MPLS)
Temps aller-retour du premier ping (ms) 64 octets	0.9	0.8	6.0
Temps aller-retour moyen (ms) 64 octets	0.3	0.3	0.5
Débit (Mb/s) Connexion par commutateur	91.4	91.1	91.0

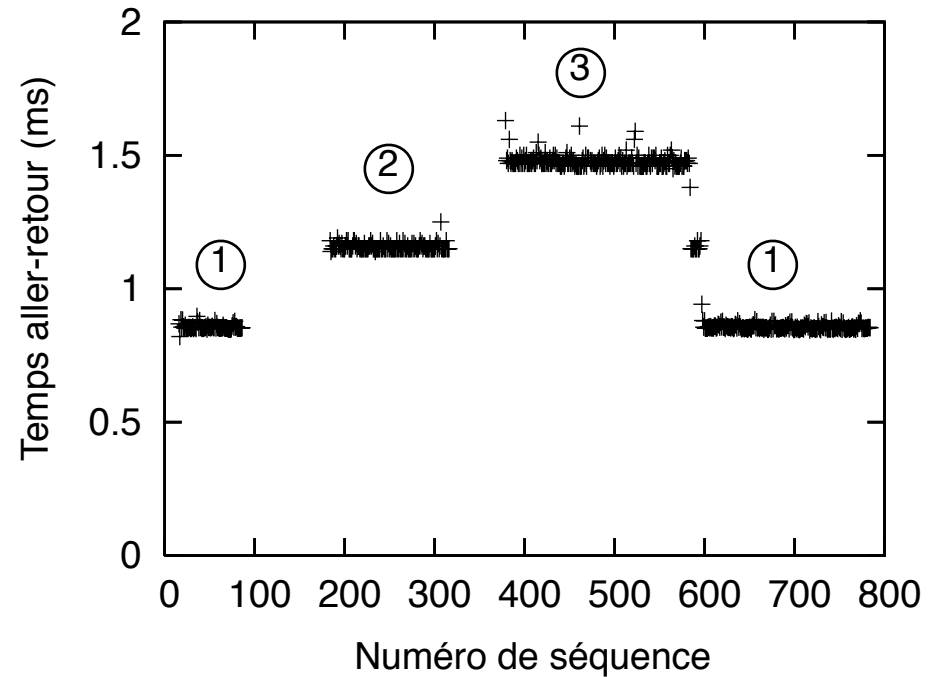
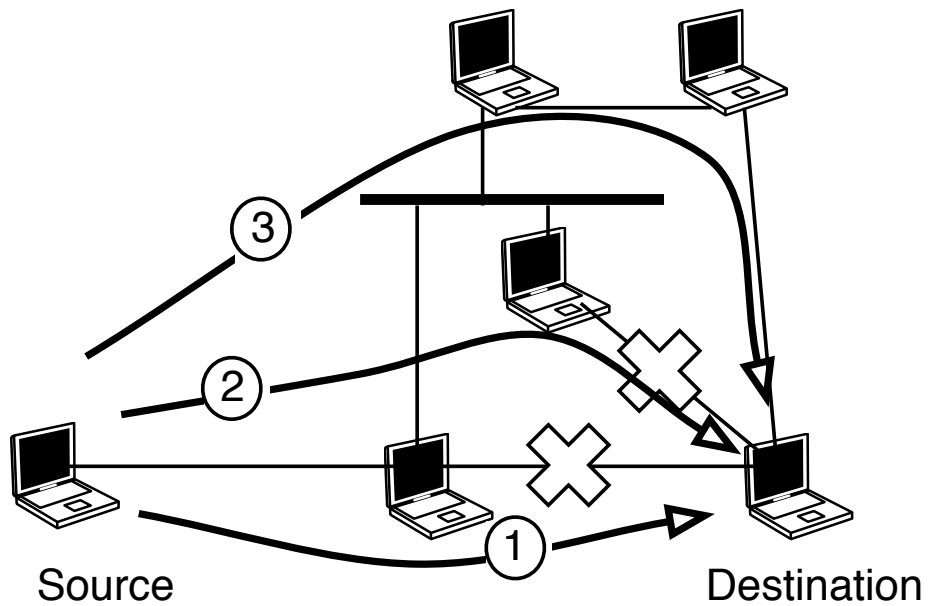
Optimisation et chemin de secours



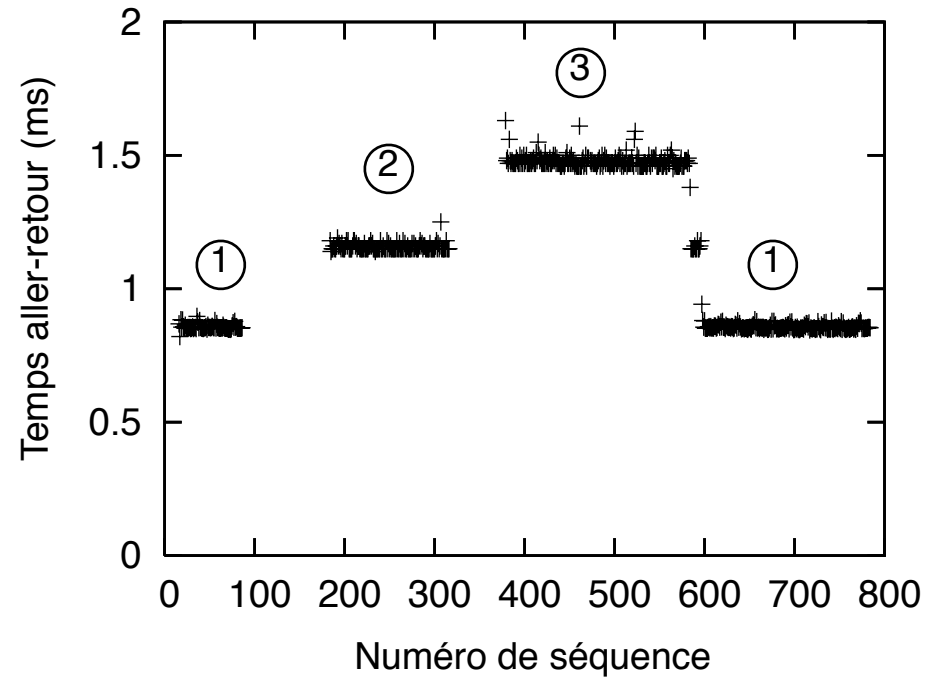
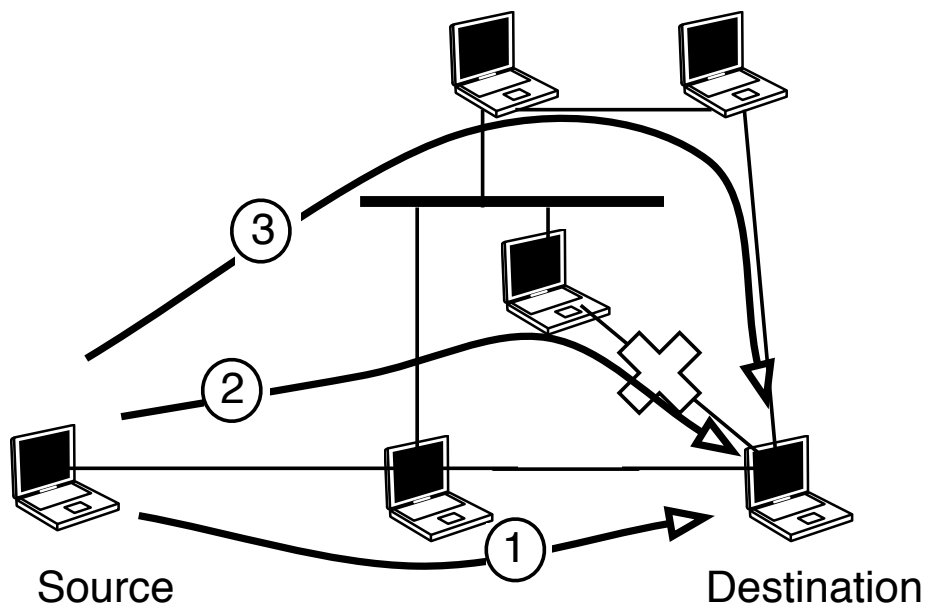
Optimisation et chemin de secours



Optimisation et chemin de secours



Optimisation et chemin de secours



Conclusion

- Contexte des réseaux de bordure spontané
- Proposition d'une nouvelle architecture non-conventionnelle
 - importance des connexions
 - choix de la réactivité
 - utilisation d'une couche 2.5
 - chemins de secours, chemins multiples
- Implémentation complète
- Confrontation avec la réalité

Perspectives

- Meilleure gestion de la diffusion dans l'architecture
- Intégration d'un modèle de sécurité
- Étude sur les métriques
 - Limites des informations disponibles
- Autoconfiguration sur un réseau multi-sauts
 - Zeroconf basé sur IP ?

Merci

vincent.untz@imag.fr