Mobile Ad hoc Networks, MANet ou Réseau ad hoc de TMCs

Ce sont des réseaux :

- fortement **dynamiques** : les terminaux apparaissent, disparaissent, se déplacent :
  un TMC fait partie du réseau s'il est proche du réseau
- où les lignes de transmission **n'existent que lorsqu'un échange a lieu** entre deux TMCs ;
- divisés en deux catégories : **avec infrastructure** et **totalement indépendant** de tout dispositif fixe.

Réseau avec infrastructure

Les équipements mobiles ne communiquent pas directement entre eux :
ils doivent passer par l'intermédiaire d'une sorte de serveur central se comportant comme un «*switch*». Ce switch est appelé point d'accès, «*Access Point*» ou station de base, «*base station*».
Il permet de faire l'intermédiaire entre :

- plusieurs réseaux non-filaires ;
- un réseau non filaire et un réseau fixe.

Avantages

- la solution la plus répandue ;
- un déploiement facile et peu coûteux par rapport à aux installations fixes (plus de câbles, de percages de cloison...) ;
- une utilisation identique aux installations fixes ;
- les utilisateurs peuvent se déplacer tout en restant connecté (dans les limites d'un périmètre autour de la borne et en fonction des obstacles entre celle-ci et l'équipement) ;
- C'est le cas des réseaux dits «*cellulaires*» de la téléphonie mobile.

Inconvénients

- une mobilité plus réduite car dépendante d'une infra structure minimale ;
- de nouvelles possibilités «*d'eavesdropper*», c-à-d d'**écoutes non sollicitées** !

Réseau ad hoc ou sans infrastructure

Il ne requiert **aucune infrastructure** minimale (aucun point fixe dans le réseau).

Il a une **origine militaire** : les infrastructures étant la première cible en cas de conflit !

Il s'est adapté à la vie civile pour ses avantages :

- **organisation rapide** des secours sur les lieux d'une catastrophe naturelle ou humaine ;
- mise en œuvre d'un système d'information entre deux smartphones qui se croisent ;
  Au delà du simple échange de carte de visite ou bien du bluejacking !
- possibilité de définir des communications inter-groupes ;
- réseaux en mouvement tels que l'informatique embarquée (GPS, Smartphone et logiciel de cartographie, musique...) ;
- réseaux déployés en milieu naturel : réseau de capteurs ou en à la maison : IoT, «*Internet of Things*».

Avantage

Il propose la mobilité maximale et un déploiement facile et immédiat.

Inconvénient

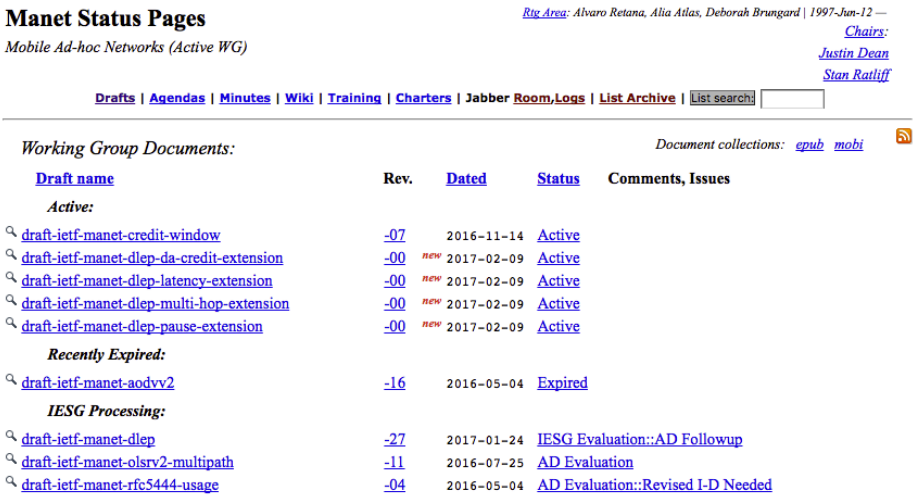
Il est le plus difficile à mettre en œuvre.

Internet Engineering Task Force Working Group MANET

The purpose of this working group is to standardize **IP routing protocol functionality** suitable for wireless routing application within both **static** and **dynamic** topologies.

The fundamental design issues are :

- that the wireless link interfaces have some unique routing interface characteristics ;
- that node topologies within a wireless routing region may experience increased dynamics, due to **motion** or **other factors**.
- `https://tools.ietf.org/wg/manet/`



Manet Status Pages

Mobile Ad-hoc Networks (Active WG)

Rtg Area: Alvaro Retana, Alia Atlas, Deborah Brungard | 1997-Jun-12 —
Chairs:
Justin Dean
Stan Ratliff

Drafts | Agendas | Minutes | Wiki | Training | Charters | Jabber Room,Logs | List Archive | List search:

Working Group Documents:

Document collections: epub mobi

Draft name	Rev.		Dated	Status	Comments, Issues
Active:					
draft-ietf-manet-credit-window	-07		2016-11-14	Active	
draft-ietf-manet-dlep-da-credit-extension	-00	new	2017-02-09	Active	
draft-ietf-manet-dlep-latency-extension	-00	new	2017-02-09	Active	
draft-ietf-manet-dlep-multi-hop-extension	-00	new	2017-02-09	Active	
draft-ietf-manet-dlep-pause-extension	-00	new	2017-02-09	Active	
Recently Expired:					
draft-ietf-manet-aodvv2	-16		2016-05-04	Expired	
IESG Processing:					
draft-ietf-manet-dlep	-27		2017-01-24	IESG Evaluation::AD Followup	
draft-ietf-manet-olsrv2-multipath	-11		2016-07-25	AD Evaluation	
draft-ietf-manet-rfc5444-usage	-04		2016-05-04	AD Evaluation::Revised I-D Needed	

Pourquoi les MANET ou «*Mobile Ad Hoc Networks*»

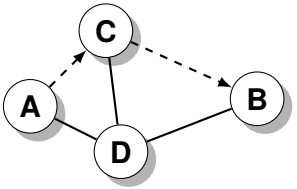
Accéder à des services de communication et de traitement **durant des déplacements**.
Ils se caractérisent par l'utilisation de liaisons sans fil entre le terminale mobile et les autres éléments du réseau.

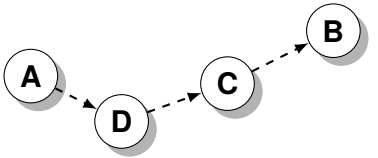
Différents types de réseaux sans fil

- basé sur une infra structure :
  - le **système cellulaire**, GSM, basé sur l'utilisation d'une infrastructure de stations ;
- les réseaux locaux, *LAN*, sans fil, *wireless* :
  - utilisant des liaisons radios (802.11, etc) ou infra-rouge (IrDA) ;
  - extrêmement flexible dans la zone de couverture : possibilité de définir des réseaux ad hoc ;
  - débits inférieurs à ceux des réseaux câblés (de 1 à 54 Mbit/s).
- les **réseaux ad hoc** :
  - indispensables quand une infrastructure est indisponible, impossible à mettre en œuvre ou bien trop chère ;
  - les premières applications sont militaires, pour la mise en place de secours ou bien dans le cas de réseau à la maison ;
  - potentiellement intéressants dans le cas d'un déploiement dans une ville.

Caractéristiques

- l'hôte bouge fréquemment ;
- la topologie du réseau évolue également fréquemment :





- il n'existe **pas d'infrastructure** de cellule ;
- les routes sont définies par des **sauts successifs** au travers de liens sans fil : «*Multi-Hop Wireless links*» ;
- les données doivent être **routées** via des nœuds intermédiaires.

Les inconvénients des autres solutions

L'installation de points d'accès fixes et d'une architecture d'échange entre plusieurs de ces points d'accès (backbone) n'est pas toujours possible :

- cette infrastructure peut ne pas exister dans le cas d'une zone de guerre ou bien avoir été détruite dans le cas d'un désastre naturelle ;
- cette infrastructure ne peut pas être mise en œuvre de manière efficace dans le cas de l'utilisation de liens radio de faible portée, comme dans le cas du Bluetooth qui a une portée de 10m.

Les avantages

Ils ne nécessitent pas d'infrastructure de type backbone.
Ils sont faciles à déployer.
Ils s'auto-configurent automatiquement.
Ils peuvent être employés lorsque l'infrastructure est absente, détruite ou bien impossible à mettre en œuvre.
Ils sont également utilisables dans le cas où l'on veut pouvoir créer un réseau privé temporaire au sein d'une infrastructure existante (dans ce cas là ils sont indépendants l'un de l'autre).

De multiples applications

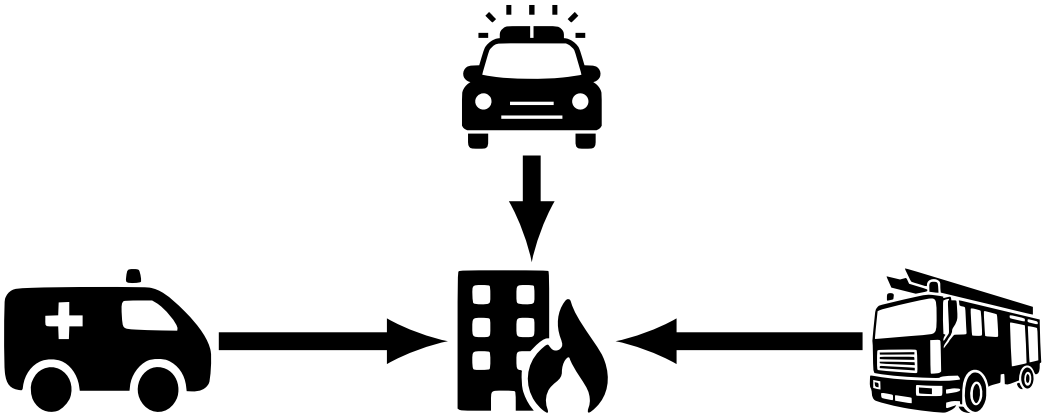
Les PANs, «*Personal Area Network*», composés de téléphone cellulaires, de PCs portables...
Les environnements militaires où ils permettent à différents acteurs de communiquer.
Les environnements civils : salle de réunion, navires ...
Les situations d'urgence : catastrophe naturelle, incendies, opérations de recherche et de secours.
Les environnements urbains pour créer des MAN Metropolitan Area Network.

Les secours arrivent sur place, ils ont besoin :

- ☐ des carnets de santé des blessés ;
- ☐ des plans des bâtiments, avertissement sur la dangerosité de substances présentes sur le lieu ;
- ☐ de la liste d'indices relevés pour déterminer la cause de la situation ;
- ☐ des rapports d'états des différentes équipes présentes pour se coordonner.

Ils exploitent les ressources provenant de :

- des TMCs des personnes présentes ;
- d'éléments fixes présent sur les lieux ;
- d'Internet.



Le partage d'information et l'accès à cette information est critique.

La coopération est nécessaire mais pas toujours désirée !

Un réseau mobile ad hoc est un ensemble de TMC sans fil qui peuvent déplacer librement.

La **topologie** du réseau change :
- de manière imprévisible ;
- rapidement.

Problème de déterminer une route entre un terminal et un autre pour communiquer :
- prendre en compte le **caractère versatile** du réseau pour la recherche d'une route ou sa mise à jour : les terminaux sont mobiles et entrainent la perte ou la création de liens ;
- prendre en compte la **vitesse de mobilité** qui peut entraîner un taux plus importants de destruction/création de liens ;
- avoir de **bonnes performances** suivant de nouveaux critères : **stabilité** d'une route par rapport à la mobilité, **consommation d'énergie** du à l'application de l'algorithme de routage.

Il existe trois groupes d'algorithmes de routage :
- les protocoles **proactifs** : basés sur des méthodes «d'état des liens» et de «vecteur de distance» utilisés dans les réseaux câblés, qui essayent de maintenir une carte du réseau à tout moment ;
- les protocoles **réactifs** : c'est un routage qui se fait à la demande, uniquement lorsqu'il est nécessaire de créer une route entre deux terminaux pour un échange ponctuel. Dans ce cas là, il n'existe pas de connaissance de la carte du réseau ;
- les protocoles **hybrides**.

Sur le temps d'attente pour la découverte d'une route :

- les protocoles **pro-actifs** ont un délai plus court car les routes sont maintenues à tout moment ;
- les protocoles **réactifs** peuvent avoir un délai plus important parce qu'une route entre A et B n'est déterminer que si A veut communiquer avec B.

Sur le surcoût de la découverte et de la maintenance des routes :

- les protocoles **réactifs** ont un surcoût plus petit car les routes ne sont déterminées qu'à la demande ;
- les protocoles **pro-actifs** peuvent avoir un surcoût supérieur parce que les routes sont mises à jour en continue.

Le choix de bons compromis dépend :

- du trafic ;
- de la mobilité des terminaux.

Pro-actifs

- maintiennent toujours les routes ;
- peu ou pas de délai pour déterminer une route ;
- consomme de la bande passante pour maintenir à jour les routes ;
- maintiennent des routes qui peuvent ne jamais servir.

Réactifs

- un surcoût minimal parce que les routes ne sont déterminées qu'à la demande ;
- un délai important lors de la détermination d'une route ;
- emploi des systèmes d'inondations (flooding) pour réaliser une recherche globale ;
- le contrôle des échanges peut être difficile.

But des protocoles de routage

- diminuer le surcoût du au routage ;
- trouver les plus courtes routes ;
- trouver des routes stables malgré la mobilité.

Contraintes

- changement de routes fréquent : les données devront être échangées pour les modifications de routes doivent être plus petits que dans les réseaux traditionnels ;
- les modifications de routes peuvent être associés à la mobilité d'un terminal ;
- les liens ont de petit débits.

Proactifs :

- protocoles de détermination de chemins les plus courts traditionnels et distribués ;
- maintient les routes entre chaque paire de terminaux à tout moment ;
- basés sur des mises à jour périodiques (surcoût en communication) ;
- exemple : DSDV, «*Destination Sequenced Distance Vector*».

Réactifs :

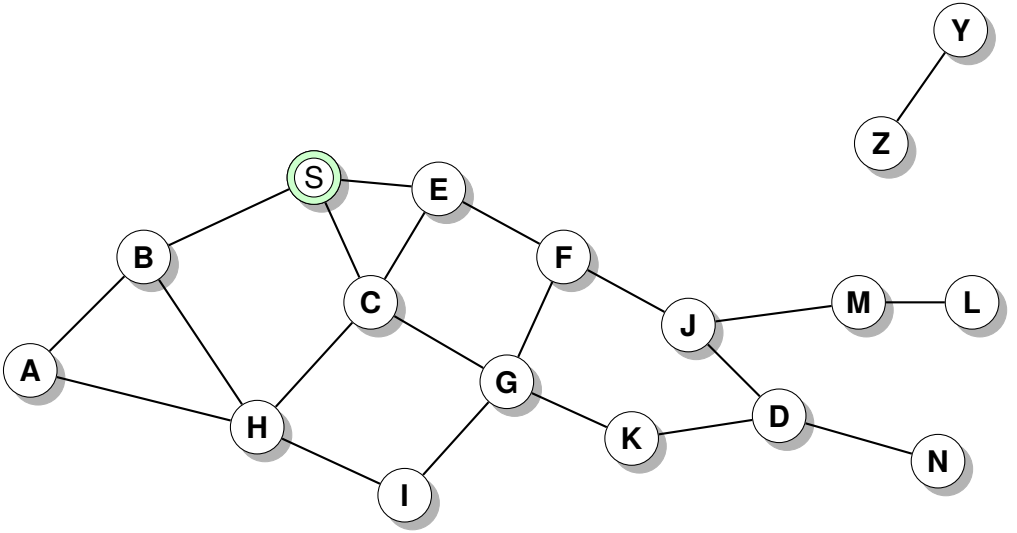
- détermine une route au besoin ;
- c'est la source qui initie la découverte d'une route ;
- exemple : DSR, «*Dynamic Source Routing*».

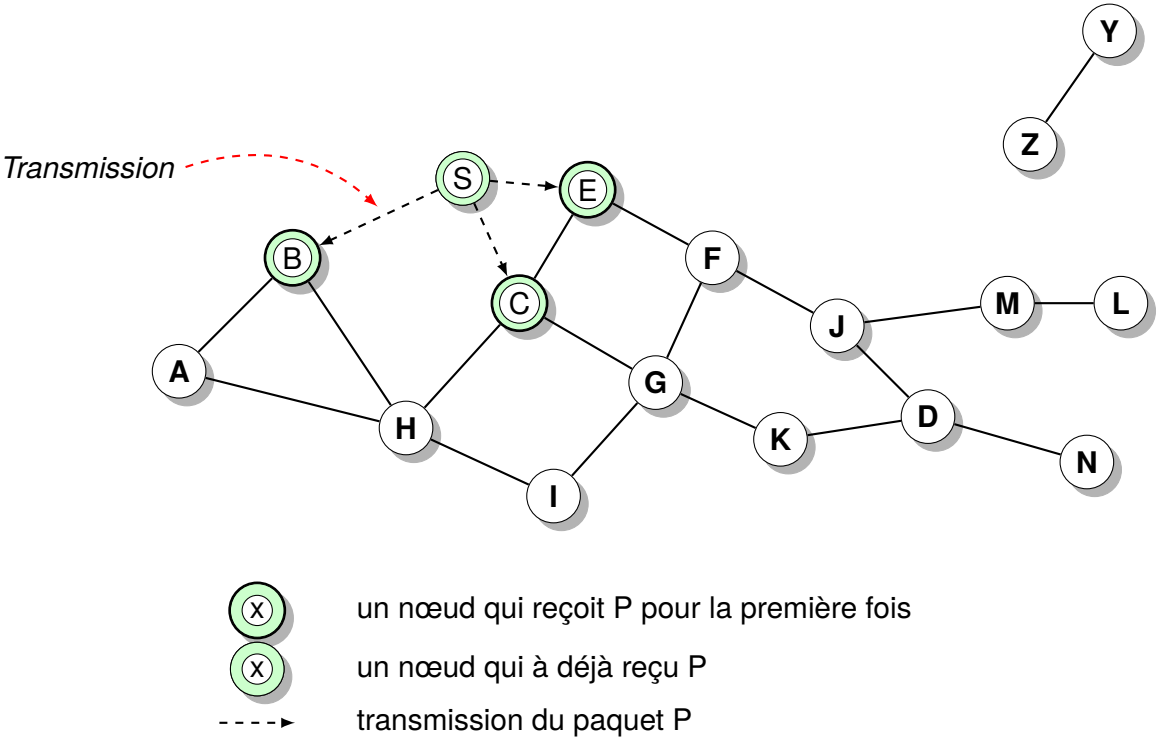
Hybrides :

- adaptatifs : combinaison des deux précédents ;
- exemple : ZRP, «*Zone Routing Protocol*».

L'inondation

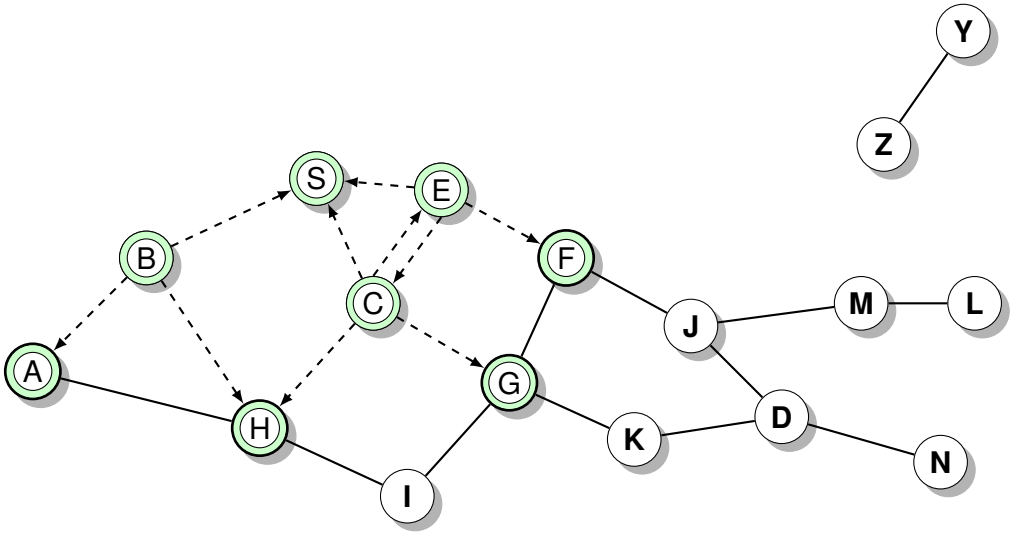
- ▷ la source S envoie son paquet P à tous ses voisins ;
- ▷ chaque nœud qui reçoit P le retransmet à tous ses voisins ;
- ▷ un numéro de séquence est utilisé pour éviter de faire suivre le même paquet plus d'une fois ;
- ▷ le paquet P atteint la destination D, si D est accessible depuis S ;
- ▷ D ne doit pas faire suivre le paquet.

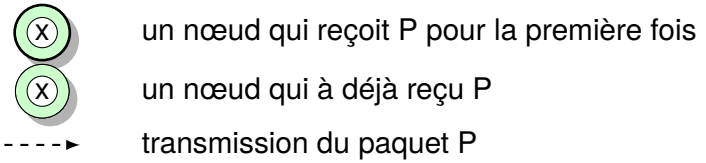



Transmission
S
E
B
C
F
A
H
G
J
M
L
Y
Z
D
K
N
I
X
un nœud qui reçoit P pour la première fois
X
un nœud qui à déjà reçu P
transmission du paquet P

L'inondation

- ▷ à leur tour, les nœuds B, C et E retransmettent le paquet P à l'ensemble de leurs voisins ;
- ▷ le nœud H reçoit le paquet P depuis deux voisins : **collision**.



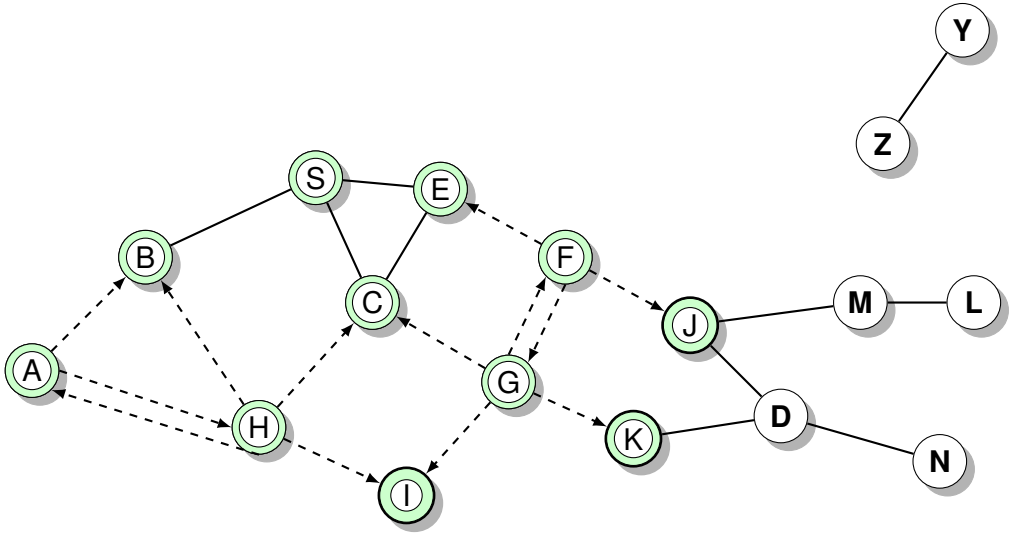


un nœud qui reçoit P pour la première fois

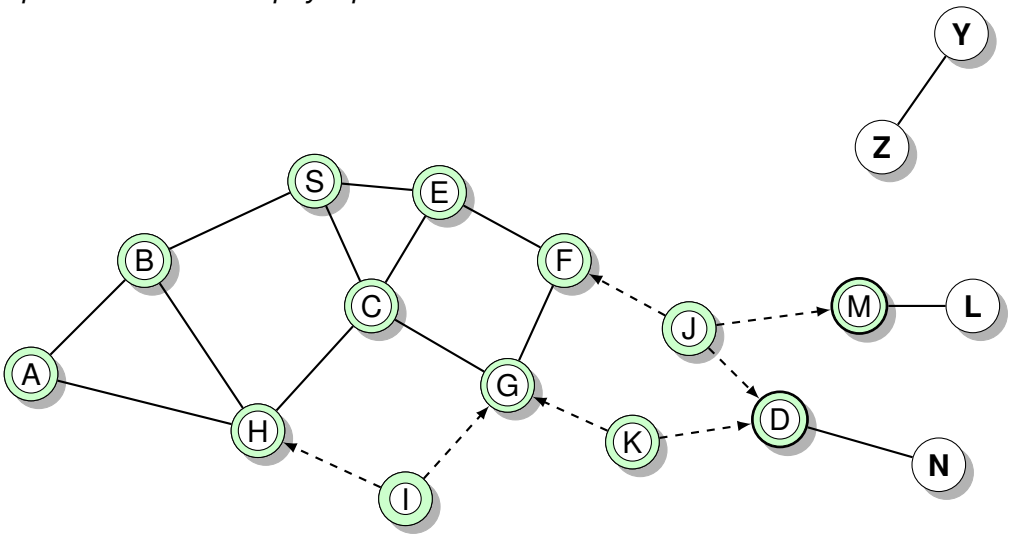
un nœud qui à déjà reçu P

transmission du paquet P

▷ Le noeud C reçoit le paquet P de G et H, met ne le fait pas suivre de nouveau, car il l'a déja fait suivre.



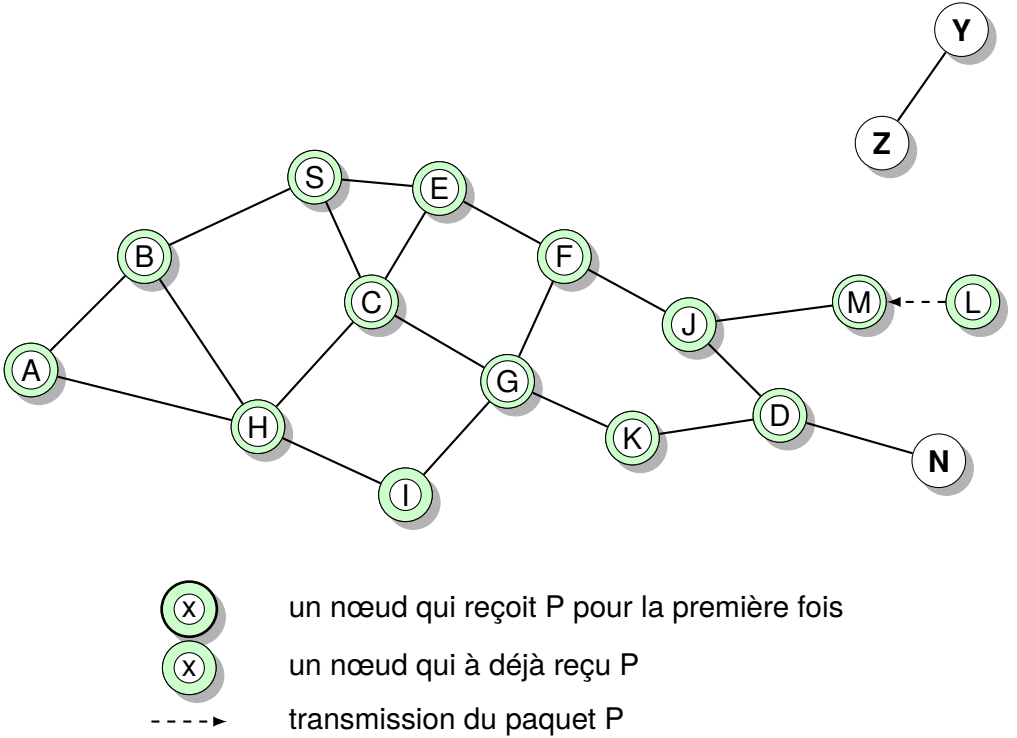
▷ Les noeuds J et K retransmettent tous les deux le paquet P à D.
Comme ils sont invisibles l'un de l'autre, leurs transmissions peuvent entrer en collision et D ne pas recevoir P !
⟹ *Cela dépend de la couche physique de communication*



▷ L'inondation est complète.

Les noeuds inacessibles de S n'ont pas reçu le paquet (Y et Z).

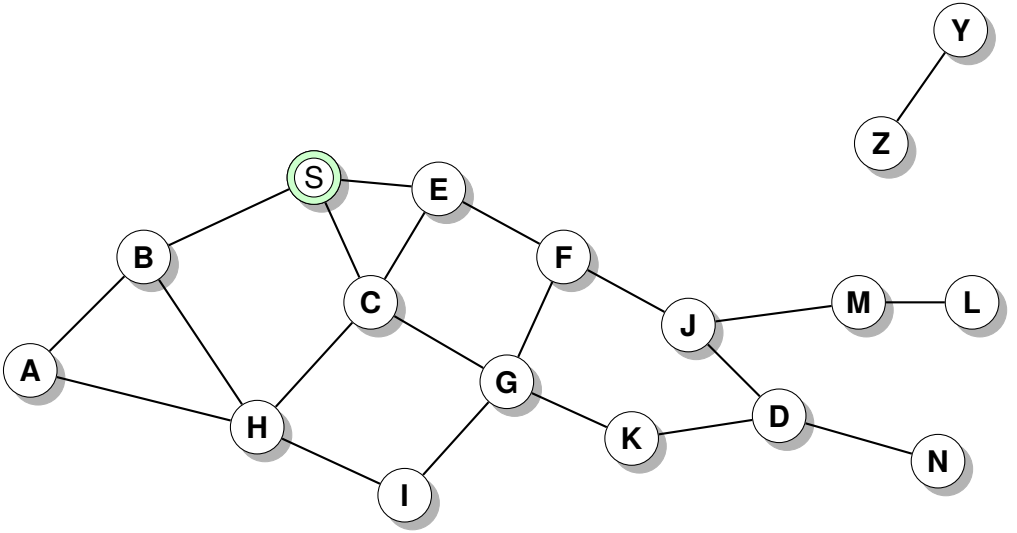
Tous les noeuds accessibles depuis S n'ont pas reçu le paquet (N), car le nœud D ne retransmets pas le paquet P, vu qu'il est la destination recherchée.



DSR : Dynamic Source Routing

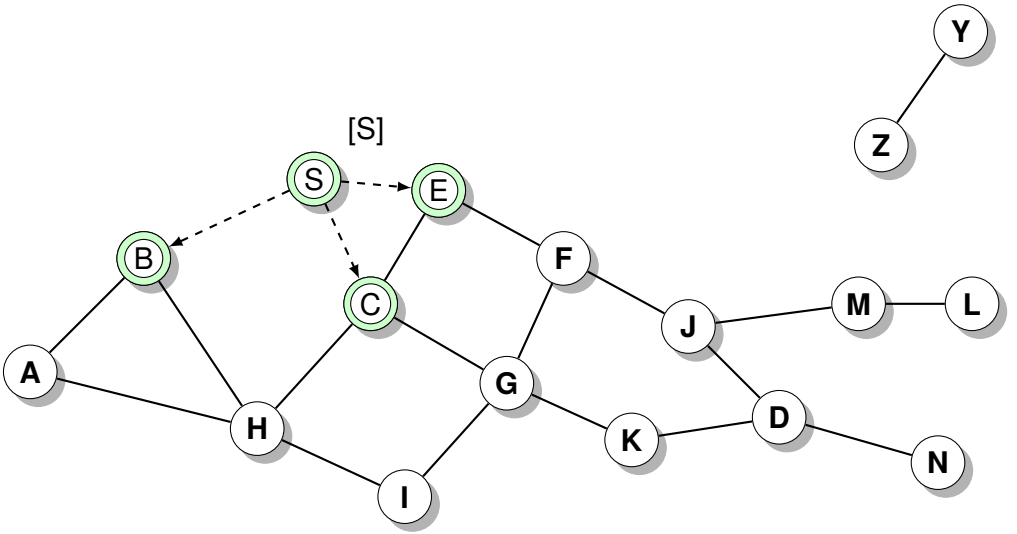
Lorsque le noeud S veut envoyer un paquet au noeud D, mais ne connais pas de route il initie une découverte de route (route discovery) :

- la source S inonde le réseau d'une Requête de route (Route Request RREQ)
- chaque noeud ajoute son identifiant quand il fait suivre ce RREQ



X un nœud qui à reçu un RREQ depuis S pour D

les deux nœuds reliés sont à portée radio l'un de l'autre

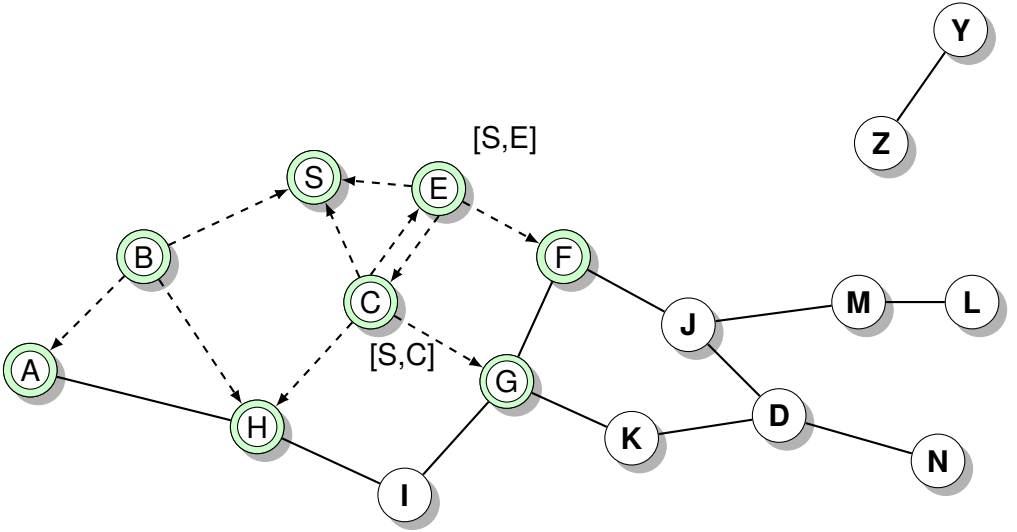


X un nœud qui à reçu le RREQ de S vers D

transmission du paquet RREQ

[...] contenu du RREQ

Le nœud H reçoit de deux voisins : collision possible

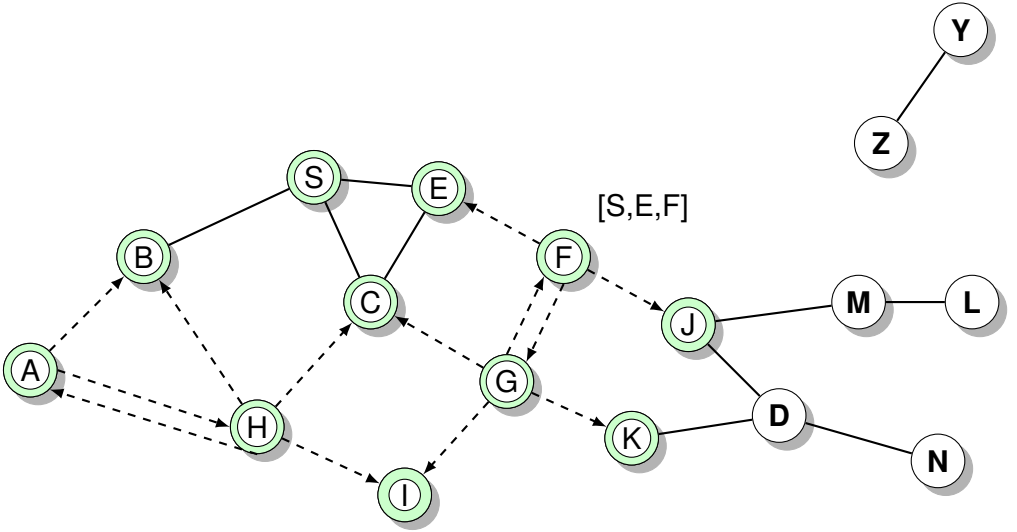


X un nœud qui à reçu le RREQ de S vers D

transmission du paquet RREQ

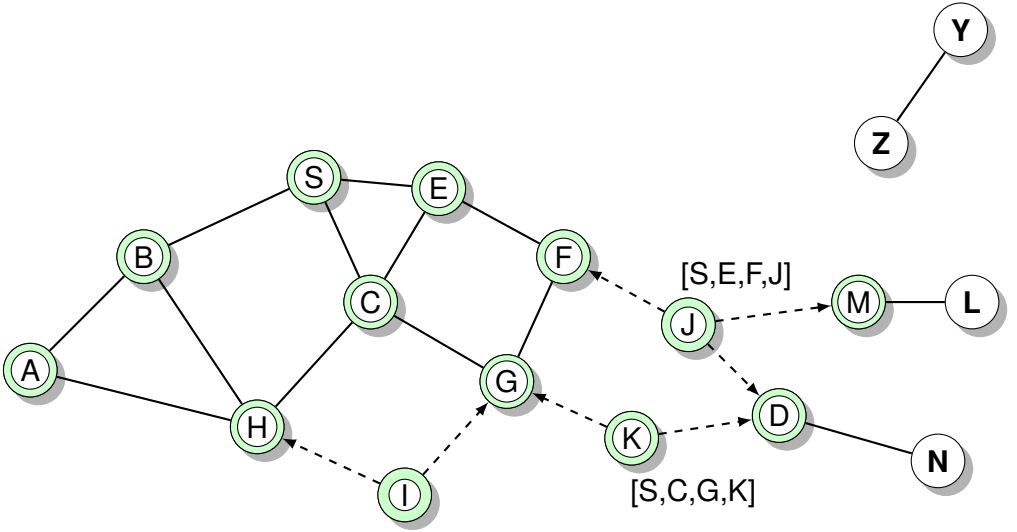
[...] contenu du RREQ

Le noeud C reçoit le RREQ de G et H mais ne le fait pas de nouveau suivre.



DSR : Dynamic Source Routing

Le noeud C reçoit le RREQ de G et H mais ne le fait pas de nouveau suivre.

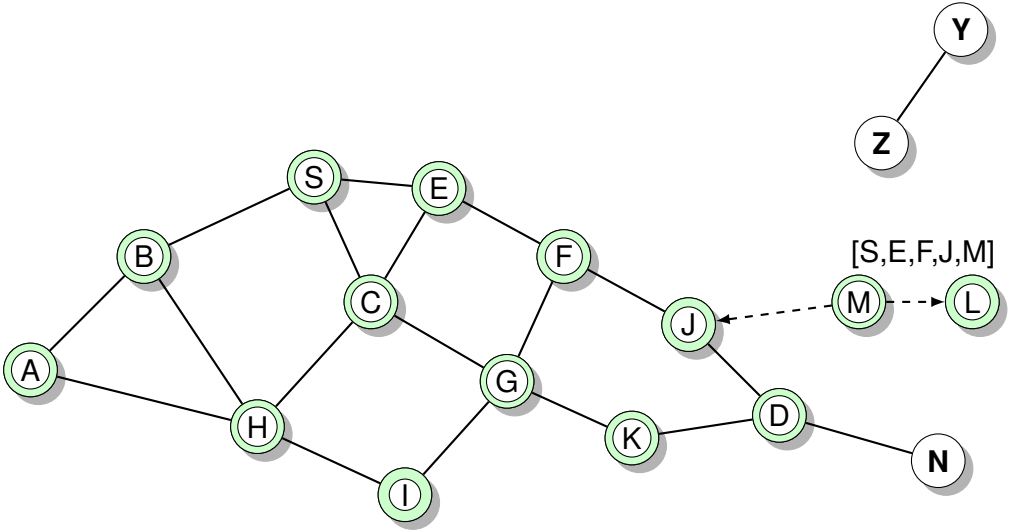


(x) un nœud qui à reçu le RREQ de S vers D

transmission du paquet RREQ

[...] contenu du RREQ

Le noeud D ne retransmet pas le RREQ car D est la destination.

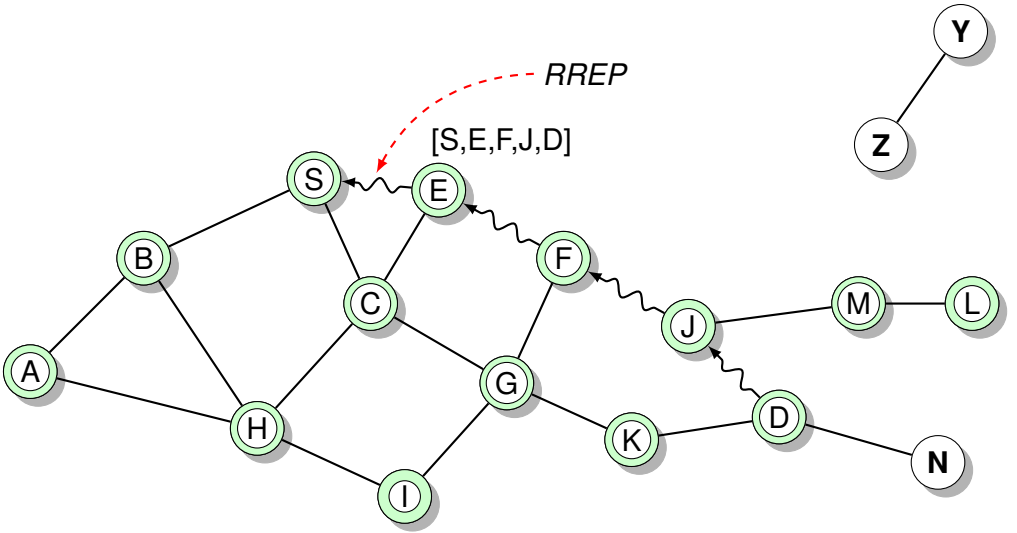


x un nœud qui à reçu le RREQ de S vers D

transmission du paquet RREQ

[...] contenu du RREQ

La destination D en recevant le RREQ renvoie un RREP, Route Reply.
Ce RREP est renvoyé à l'aide du chemin déterminé en renversant l'ordre du RREQ.
Le RREP contient le chemin qui a permis d'atteindre D.



X un nœud qui à reçu le RREQ de S vers D

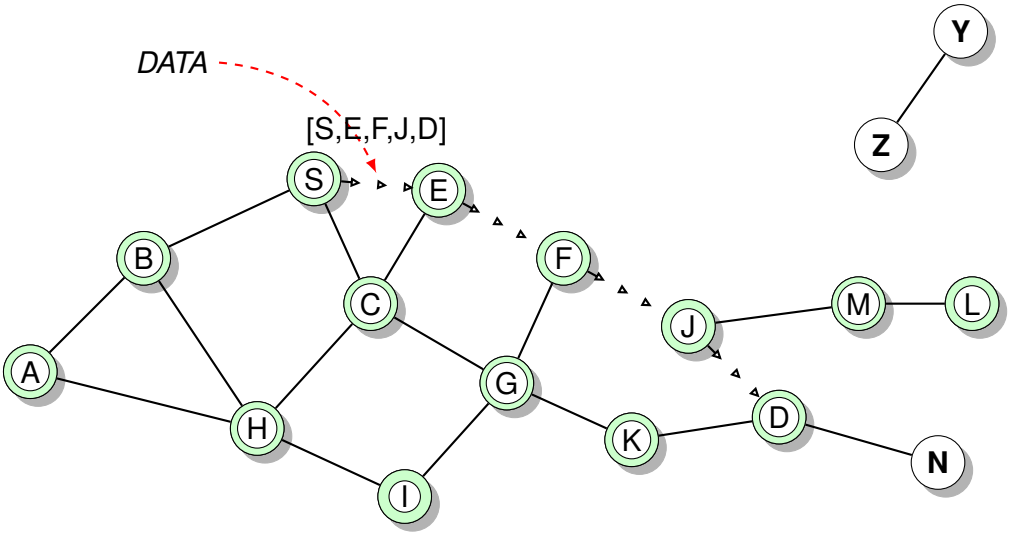
transmission du paquet RREP

[...] contenu du RREP

L'acheminement de données dans le protocole DSR

Chaque paquet de données contient le chemin complet d'acheminement.

À chaque passage dans un noeud ce chemin est diminué.



(X)	un nœud qui à reçu le RREQ de S vers D
→▸▸▸	transmission du paquet DATA
[...]	chemin emprunté par les données ou DATA

Mise en cache des routes apprises

Chaque noeud peut mémoriser une nouvelle route qu'il vient d'apprendre même s'il ne l'a pas initiée.
Lorsque le noeud S trouve la route [S, E, F, J, D] pour aller au noeud D, le noeud S a aussi appris la route [S, E, F] vers le noeud F.
Lorsque le noeud K reçoit un RREQ [S, C, G] à retransmettre, le noeud K apprend l'existence de la route [K, G, C, S] vers le noeud S.
Lorsque le noeud F retransmet le RREP [S, E, F, J, D], le noeud F apprend la route [F, J, D] vers le noeud D.
Lorsque le noeud E retransmet le Data[S, E, F, J, D] il apprend la route [E, F, J, D] vers le noeud D.

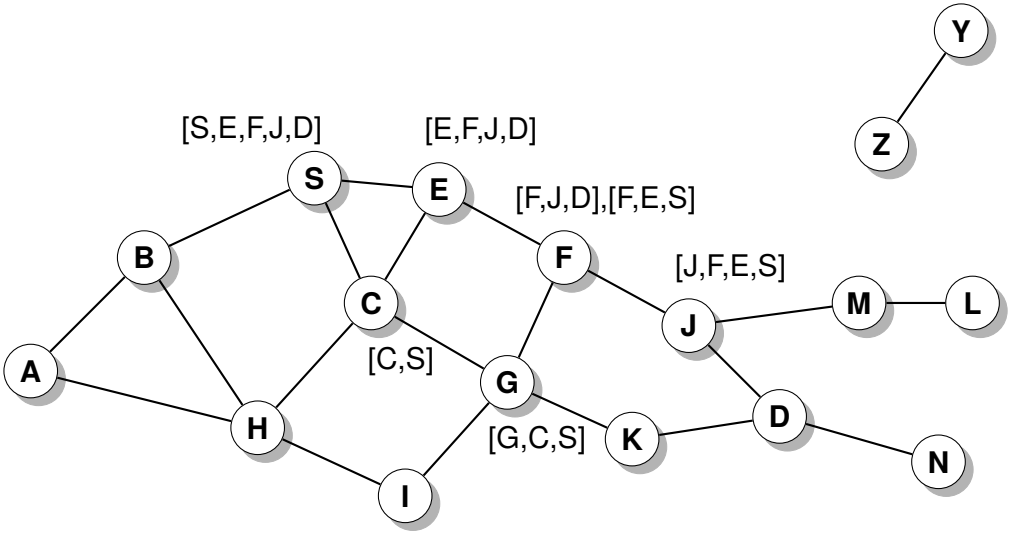
Utilisation du cache de route

Lorsque le noeud S apprend qu'une route vers le noeud D est cassée, il utilise une autre route de son cache s'il en dispose d'une. Dans le cas contraire, il initie la découverte d'une nouvelle route par l'envoi d'un RREQ.
Un noeud X à la réception d'un RREQ pour un noeud D peut envoyer un RREP s'il connait une route vers D.

L'utilisation du cache permet :

- d'accélérer la découverte des routes ;
- de diminuer la propagation des RREQ.

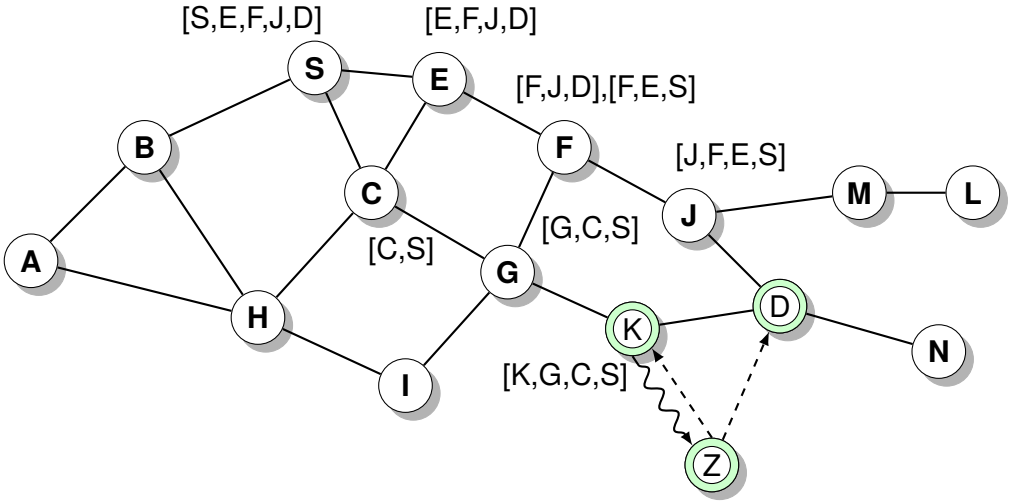
Utilisation du cache de route



[P,Q,R] représente une route mémorisée sur un nœud
(DSR mémorise les routes sous forme d'arbre)

Accélération de la découverte de route

Lorsque le noeud Z envoie un RREQ pour le node C, le noeud K renvoie un RREP [Z, K, G, C] au noeud Z en utilisant son cache.



X — un nœud qui à reçu le RREQ de Z vers C

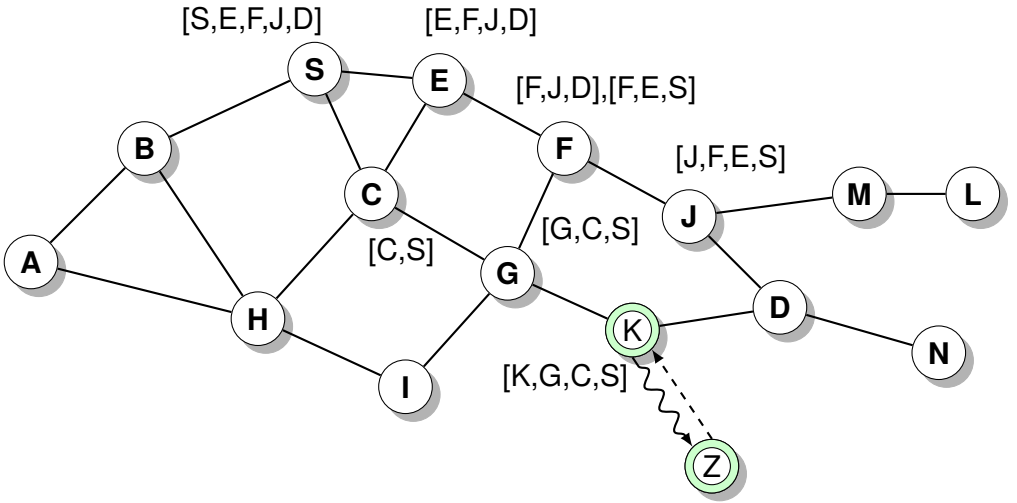
transmission du paquet RREQ

[P,Q,R] représente une route mémorisée sur un nœud

transmission du paquet RREP

Réduction de la propagation des RREQ

Dans ce cas la réponse directe de K évite la propagation de la demande.



X — un nœud qui à reçu le RREQ de Z vers C

transmission du paquet RREQ

[P,Q,R] représente une route mémorisée sur un nœud

transmission du paquet RREP

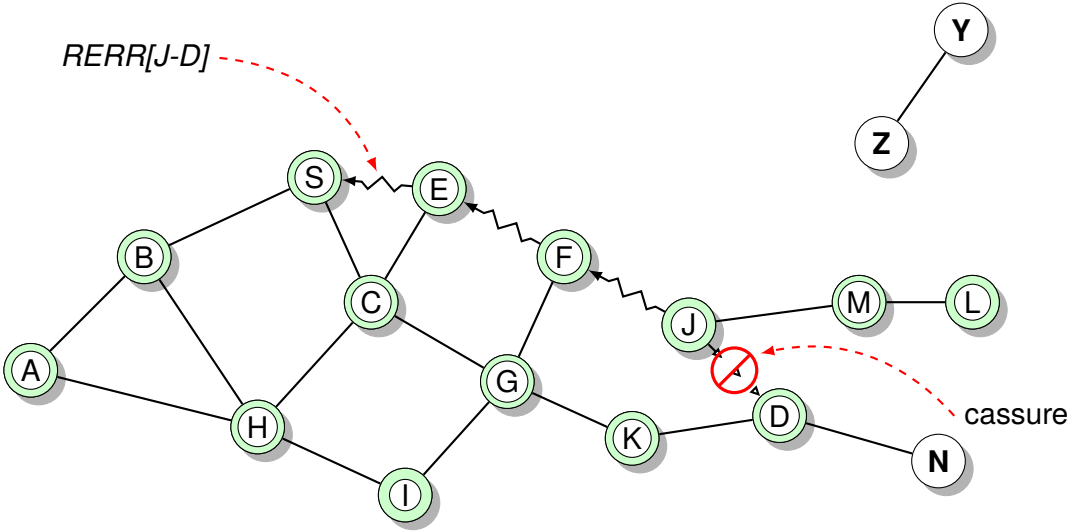
Gestion des erreurs dans DSR

J tente d'envoyer un paquet suivant le chemin [S, E, F, J, D]

J échoue à transmettre à D.

J envoie une erreur de route RERR à destination de S.

Tous les noeuds recevant le paquet RERR mettent à jour leur cache.



X un nœud qui à reçu le RREQ de S vers D

transmission du paquet DATA

transmission du paquet RERR

Avantages

- les routes ne sont entretenues qu'entre des noeuds qui communiquent ;
- l'utilisation de cache pour mémoriser les routes peuvent diminuer le surcoût engendrer par leur découverte ;
- une seule demande de découverte de route peut conduire à l'obtention de différentes propositions des noeuds intermédiaires par rapport à leur cache.

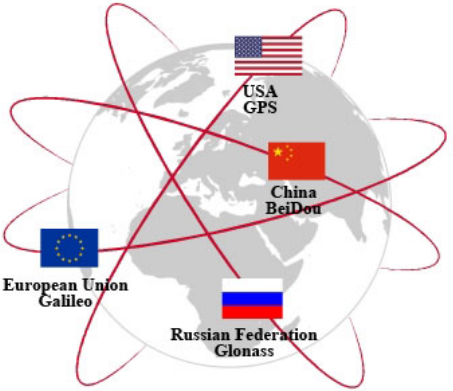
Inconvénients

- l'entête des paquets **augmentent de taille** en fonction de la taille de la route ;
- l'inondation des RREQ peut potentiellement **atteindre tous les noeuds** du réseau ;
- il peut y avoir des **collisions** lors de la transmission d'un paquet par deux voisins simultanément ;
- les caches peuvent être rendu **invalides** par le déplacement d'un noeud ;
- un noeud peut **essayer plusieurs routes** avant de trouver la bonne (utilisation de caches sur des noeuds intermédiaires) ;
- des risques d'engorgement en cas de trop nombreuses réponses obtenues à partir des caches.
  Il est possible de l'éviter si les noeuds écoutent la transmission d'un RREP et n'en transmettent un que s'il dispose d'une réponse plus courte.
- un noeud peut transmettre une **route erronée** en réponse et «*polluer*» les caches de ses voisins .

Comment diminuer la portée de l'inondation ?

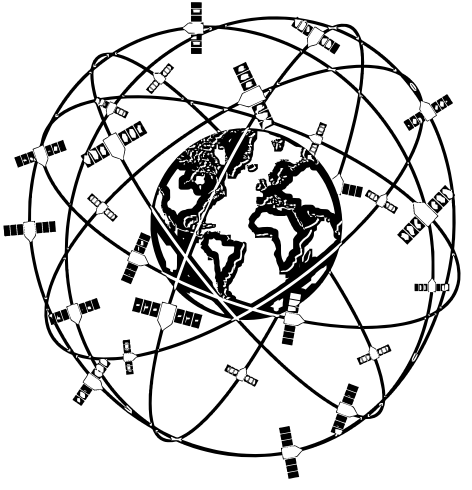
En utilisant une information sur la localisation des terminaux grâce à un dispositif de type GPS :

- le protocole LAR, «*Location Aided Routing*».



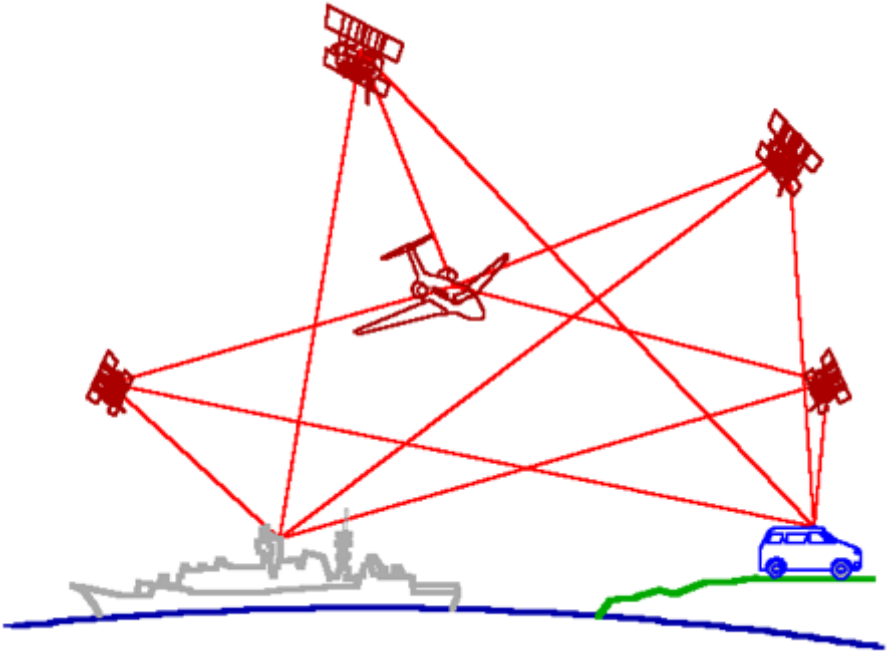
- **GPS** : Département de la défense des États-Unis d'Amérique ;
- **Glonass** : Département de la défense de la Fédération de Russie ;
- **Beidou** : Ministère de la défense de Chine ;
- **Galileo** : Système de navigation européen $\Longrightarrow$ 2020 ;
- **IRNSS** : «*Indian Regional Navigation System*» ;
- **QZSS** : «*Japanese regional navigation system*».

- **Global Positioning System** :
  - permet de déteminer une position et son élévation ;
  - fournit l'heure par une horloge atomique ;
- gérer par l'armée américaine : valeurs dégradées pour un usage commercial ;
- 28 satellites ;
- un minimum de 5 satellites visible à tout moment ;
- le récepteur mesure la distance qui le sépare d'un satellite.



Par triangulation on détermine la position du récepteur.

Les mesures de distance doivent être **extrêmement précises** : on peut ajouter au système des points fixes de position connues : navire, vehicule, etc. pour améliorer la précision de ces mesures.



Exploiter la localisation du terminal pour limiter la propagation de l'inondation

Définir des zones de localisation désirée, «*Expected Zone*», pour un terminal en tenant compte :

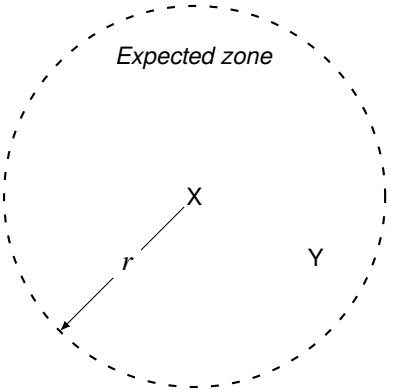
- de la localisation précédente ;
- de la vitesse du terminal.

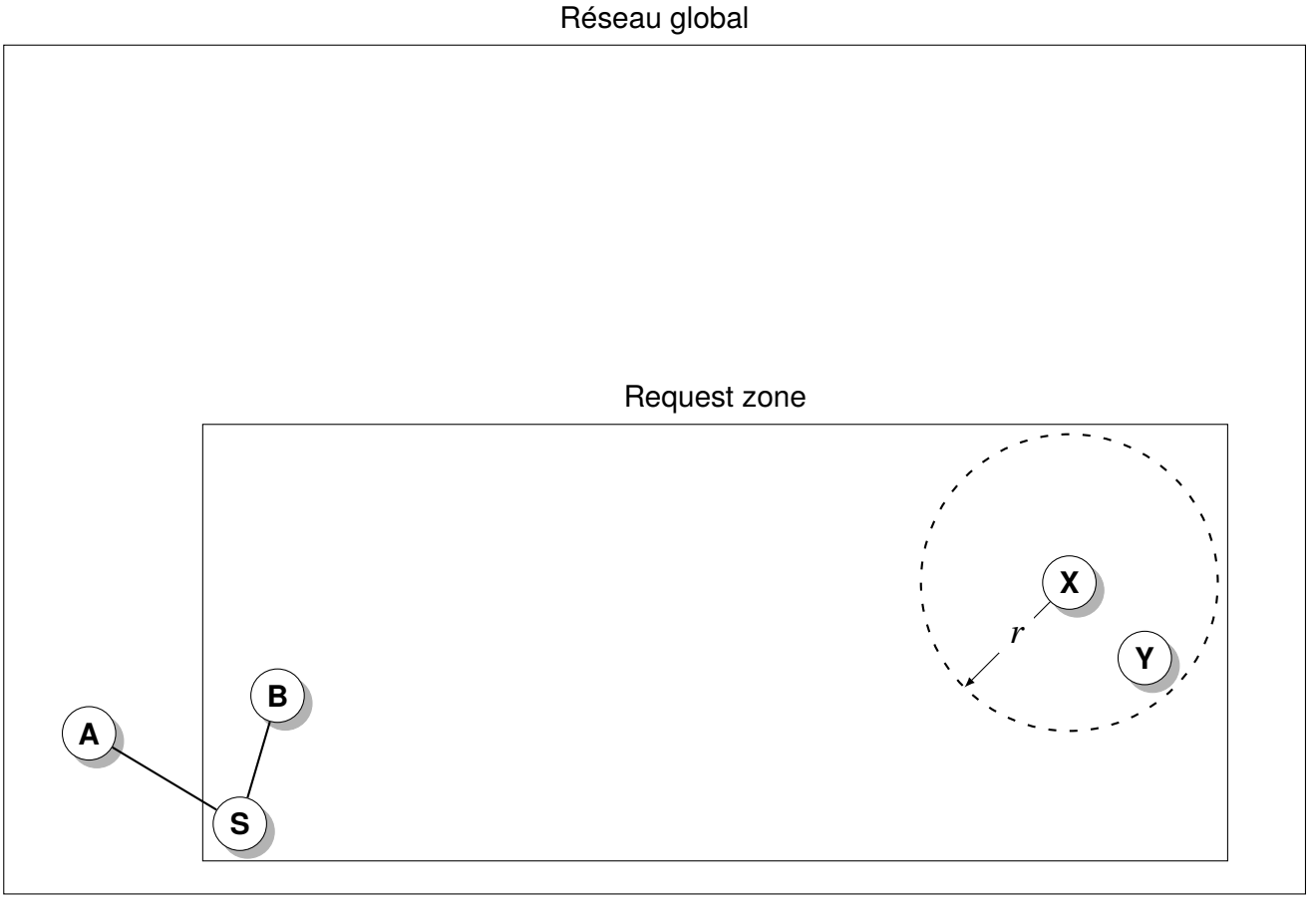
Lors de l'envoi d'une demande de route RREQ est limité à une zone de requête, «*Request Zone*», qui contient à la fois :

- la zone désirée ;
- l'expéditeur de la requête.

Détermination de la position courante d'un nœud

X = dernière position connue à t_0
Y = nouvelle position connue de D au temps courant t_1, inconnue de S
r = (t_1 - t_0) * estimation de la vitesse de D




Réseau global
Request zone
X
r
Y
B
A
S

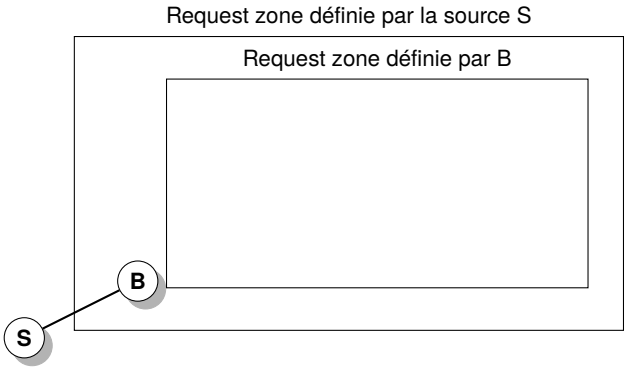
Fonctionnement

Seuls les noeuds à l'intérieur de la zone retransmettent la requête RREQ (sur l'exemple ni A ni B).
La zone de requête est spécifiée dans la demande de route.
Chaque noeud doit connaitre sa position physique pour déterminer s'il fait ou non partie de la zone de requête.
Si la requête échoue en utilisant la plus petite zone de requêe, la source déclenche de nouvelles requêtes avec une zone de requête élargie après un certain délai.
Dans ce cas là, la zone peut couvrir tout le réseau.
Le fonctionnement reste ensuite similaire à DSR.

Fonctionnement adaptatif

Chaque noeud peut modifier la zone de requête incluse dans la requête à retransmettre.
La zone modifiée peut être :

- plus récente et mieux adaptée ;
- plus petite.



Autre possibilité

il est possible d'utiliser une zone de requête implicite, c-à-d de retransmettre uniquement vers les noeuds dont la direction est proche de celle du noeud recherché.
Il est alors nécessaire que chaque noeud dispose de la position des autres noeuds de voisinage.
Il est possible :

- d'ajouter à tout message la position du terminal ;
- de diffuser la position du terminal dans le réseau.

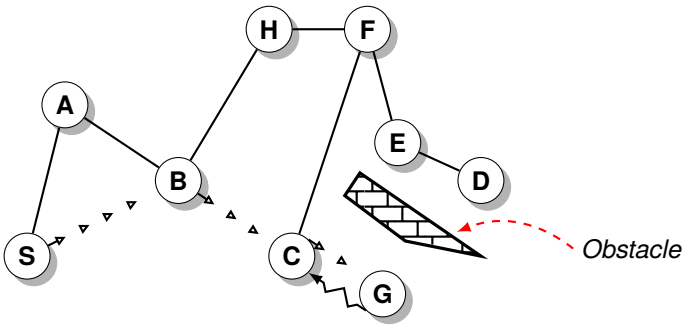
Avantages du LAR

Réduction de :

- la propagation de l'inondation ;
- du surcoût du à la découverte d'une route.

Inconvénients

Les noeuds doivent disposer de leur localisation physique.
Il ne prend pas en compte l'existence d'obstacles pour la transmission :



AODV Ad Hoc On-Demand Distance Vector Routing

Les limitations de DSR

Les routes vers la source sont incluses dans les entêtes des paquets.

Ces entêtes peuvent dégrader les performances (en particulier quand les données sont petites).

Les propositions d'AODV

les tables de routage sont maintenues dans les noeuds : les paquets n'ont plus à contenir les routes.

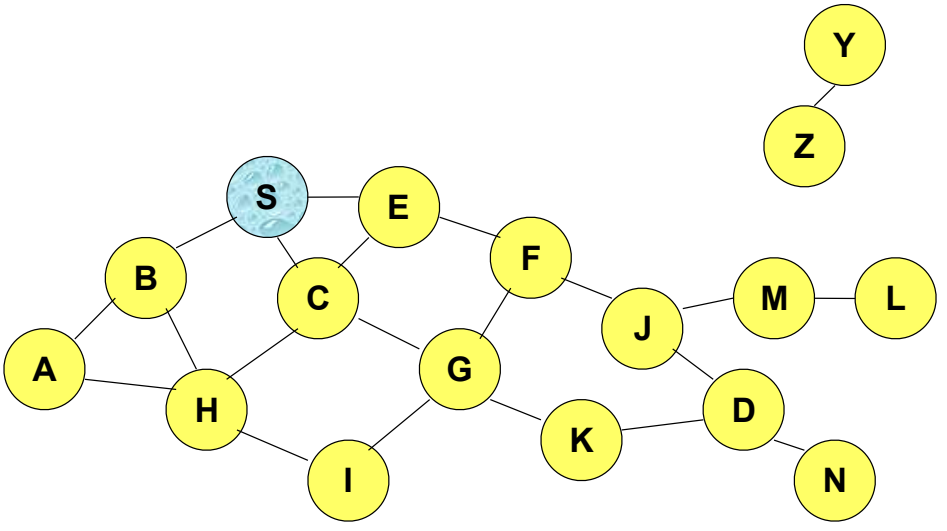
Les routes ne sont entretenues seulement entre deux noeuds qui communiquent.

Les requêtes de routes sont propager de la même manière que dans DSR.

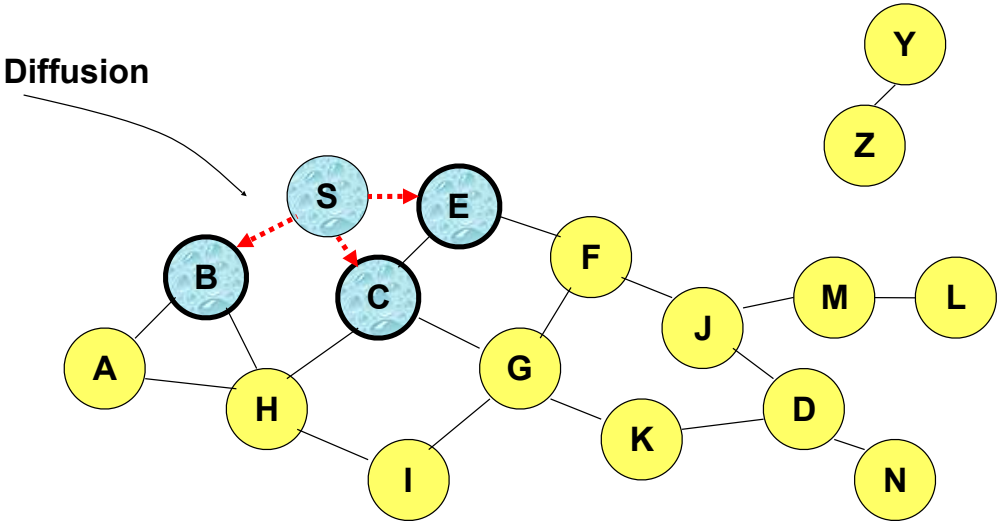
Quand un noeud rediffuse un RREQ, il mémorise un chemin inverse vers la source.

Lorsque la destination est atteinte par un RREQ, elle renvoie un RREP.
Cette réponse est routée suivant le chemin inverse mémorisé lors du passage du Route Request.

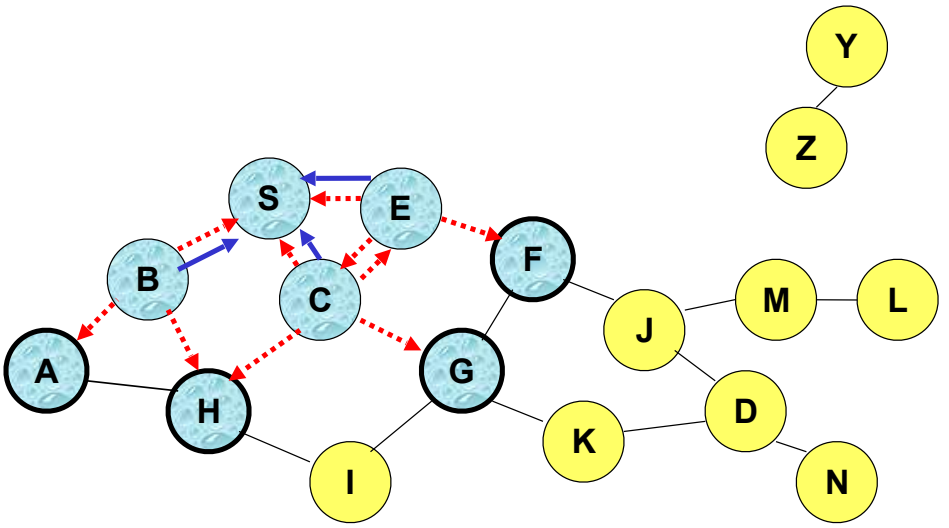
AODV



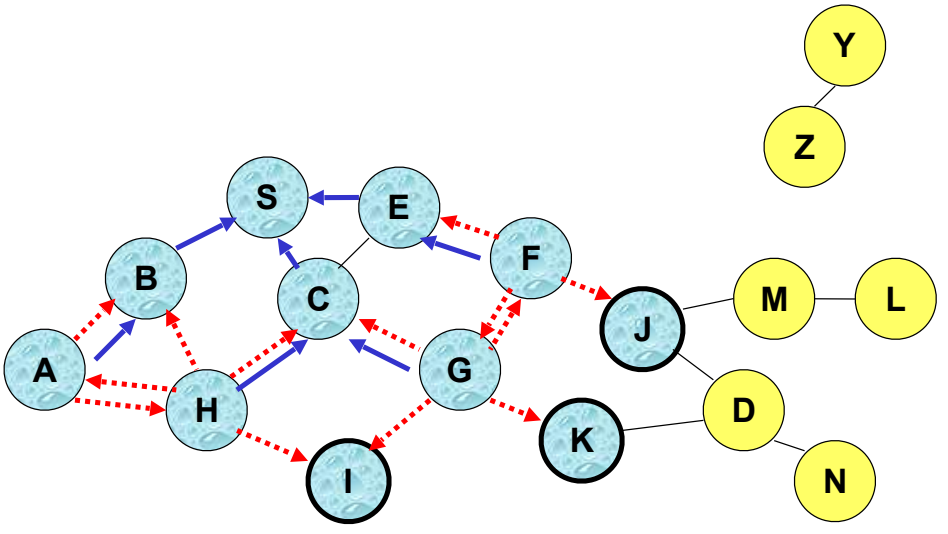
Represente un noeud qui a reçu un RREQ de S pour D


Diffusion
S
E
B
C
F
Y
Z
M
L
J
A
G
H
D
K
I
N
Represente une transmission du RREQ

AODV

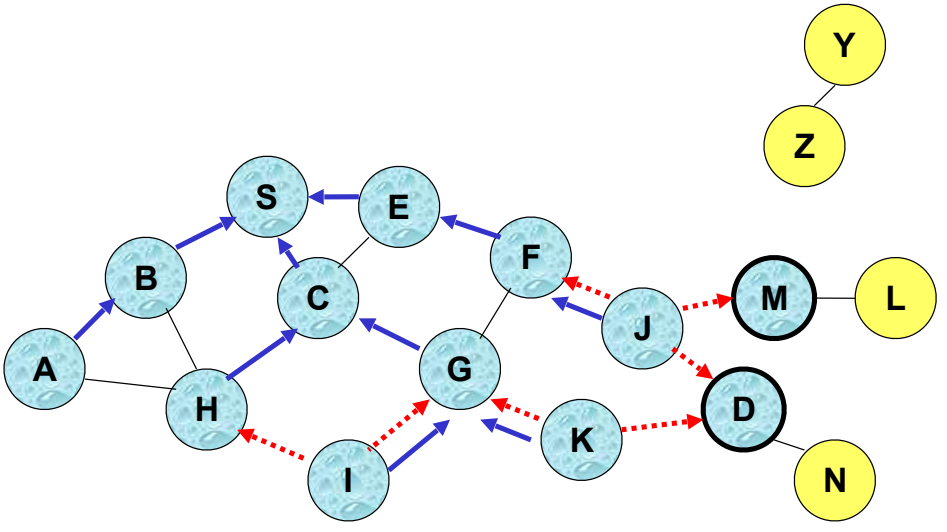

Y
Z
S
E
F
B
C
M
L
J
A
G
H
D
K
I
N
Représente les liens du chemin inverse

AODV

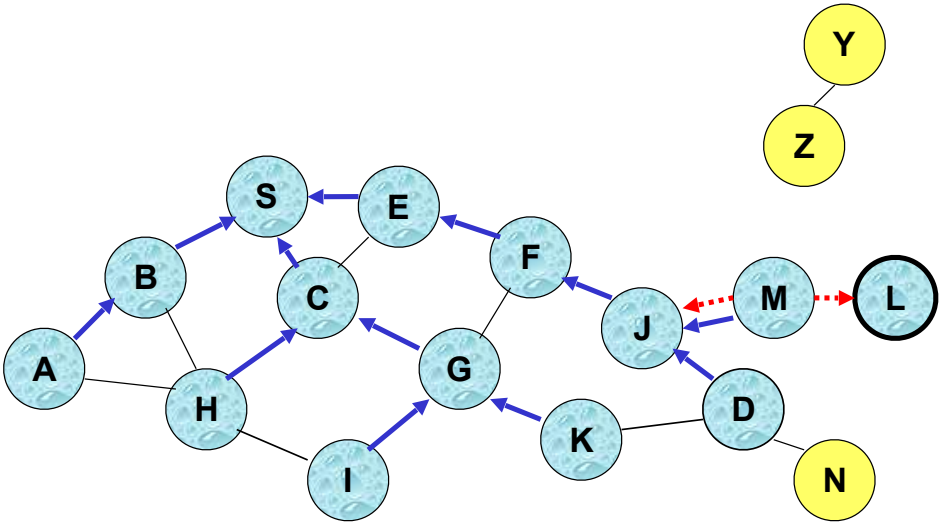


C reçoit une RREQ depuis G et H mais ne le retransmet pas car il l'a déjà fait.

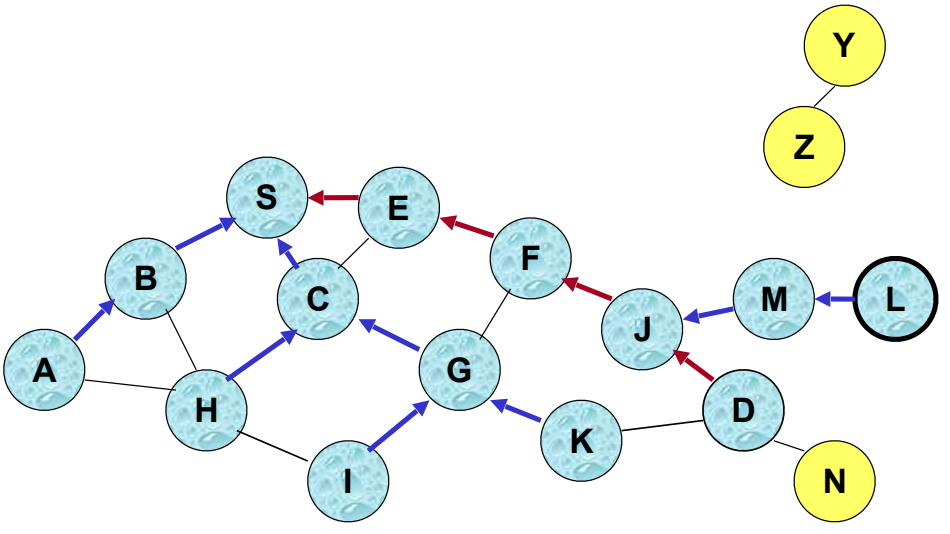
AODV



AODV



AODV


Y
Z
S
E
F
B
C
M
L
J
A
G
H
D
K
I
N
Représente les liens empruntés par RREP

AODV

RREP

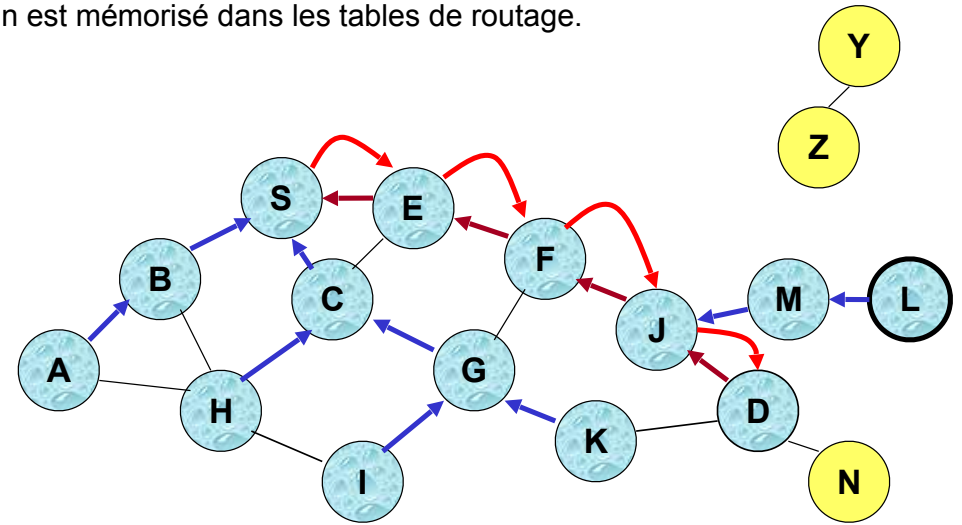
un noeud intermédiaire peut également envoyer un RREP s'il connait un chemin vers D plus récent que celui de S.

Pour déterminer qu'une route est plus récente qu'une autre, un compteur associé à la destination est ajouté au chemin.

Une nouvelle RREQ envoyée par S pour une destination dispose d'une valeur incrémentée du compteur, si un noeud intermédiaire dispose d'une route vers la destination mais avec une valeur inférieure ne renvoie pas de RREP.

AODV

Mise en place d'un « forward path » ou chemin d'acheminement
Ce chemin est mémorisé dans les tables de routage.

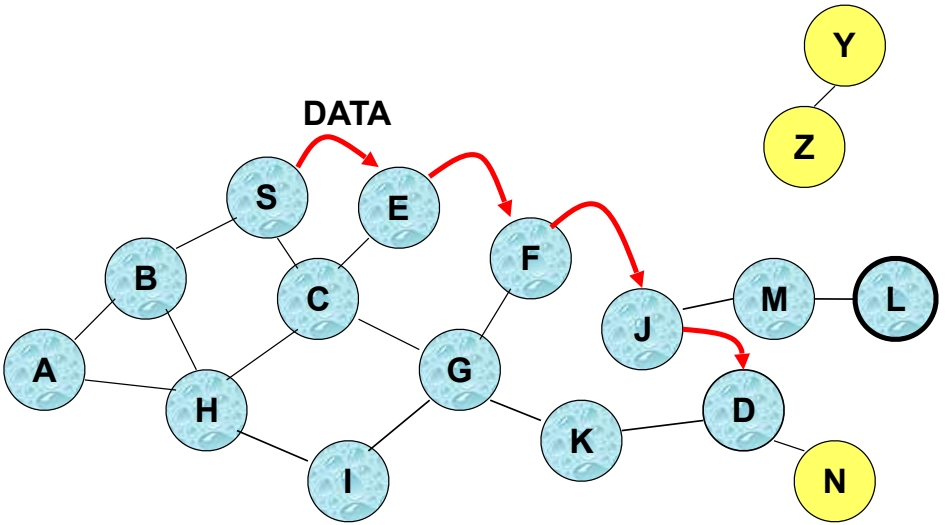


les liens de « forwarding » sont mis en place lors du
parcours sur le chemin inverse

Représente un lien sur le chemin de « forwarding »

AODV

L'envoi de données
les tables de routage sont employées pour l'effectuer.
Les routes ne sont plus incluses dans les entêtes de paquet.



AODV

Gestion des tables de routage

Une entrée de la table de routage qui sert à mémoriser un chemin inverse est purgée après un certain délai (ce temps doit être suffisant pour permettre à la RREP de revenir).

Une entrée de la table de routage qui sert à mémoriser un chemin d'acheminement (forward path) est purgé s'il n'est pas utilisé pendant un certain délai même s'il reste valide (active_route_timeout).

Un voisin d'un noeud est considéré comme actif pour une entrée dans la table de routage, si le voisin a envoyé un paquet utilisant cette entrée pendant le délai de validité de l'entrée (active_route_timeout).

Gestion des liens cassés

Lorsqu'un lien utilisé par la table de routage d'un lien casse, alors tous les voisins actifs sont informés.

L'échec d'un lien est propagé par l'utilisation de messages Route Error (RERR).
Cette propagation modifie les compteurs de destinations :

- lorsqu'un noeud X est incapable de transmettre un paquet P (de S vers D) sur le lien (X, Y) il génère un RERR ;
- le noeud X incrémente le compteur de destination associé au chemin vers D qu'il mémorise ;
- la valeur du compteur incrémentée N est incluse dans le message RERR ;
- Lorsque le noeud S reçoit le message RERR, il initie une nouvelle découverte de route (RREQ) en utilisant une valeur de compteur au moins aussi grande que N ;
- lorsque D reçoit la requête avec un compteur de destination de N, il positionne sa valeur de compteur à N, à moins qu'il ne possède déja une valeur supérieure.

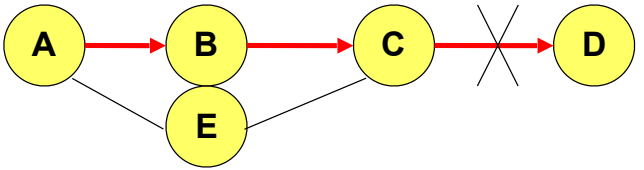
Pour détecter qu'un lien est cassé on échange périodiquement des messages entre les noeuds.

AODV

Utilisation du compteur de destination

Pour éviter d'utiliser d'anciennes ou des routes qui ne fonctionnent plus

Pour éviter de créer des boucles :



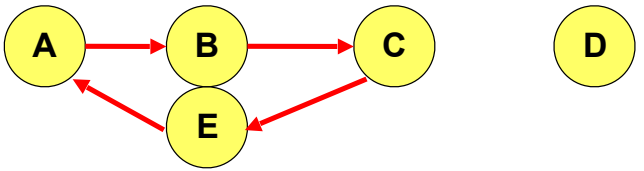
A ne sait pas que le lien entre C et D est cassé parce que le message RERR envoyé par C s'est perdu

C essaye de découvrir une route vers D :

A reçoit un RREQ par exemple par le chemin C E A

A va répondre qu'il connait un chemin vers D en passant par B

On vient de créer une boucle C, E, A, B, C !



AODV

En résumé

les routes n'ont plus besoin d'être mises dans les entêtes des paquets.
Les noeuds maintiennent des tables de routage contenant des entrées uniquement pour les routes actives.
Dans chaque noeud, une unique route est mémorisée pour un noeud destination (contrairement à DSR).

Les routes non utilisées sont purgées même si la topologie ne change pas.

Autres protocoles

De nombreux protocoles existent pour limiter l'inondation :
Power Aware Routing :

- on associe un poids à chaque lien. Ce poids correspond à la quantité d'énergie consommée pour transmettre un paquet sur ce lien ; il tient compte également de l'énergie restante dans les TMCs du lien ;
- on modifie DSR pour tenir compte des ces poids et on préfère les routes avec une somme de poids la plus petite.

Associativity-Based Routing

- seul les liens qui sont restés stables pendant un certain temps sont utilisés ;
- les noeuds incrémentent un compteur d'associativité avec ses voisins on testant périodiquement leur présence.

Signal Stability Based Adaptive Routing

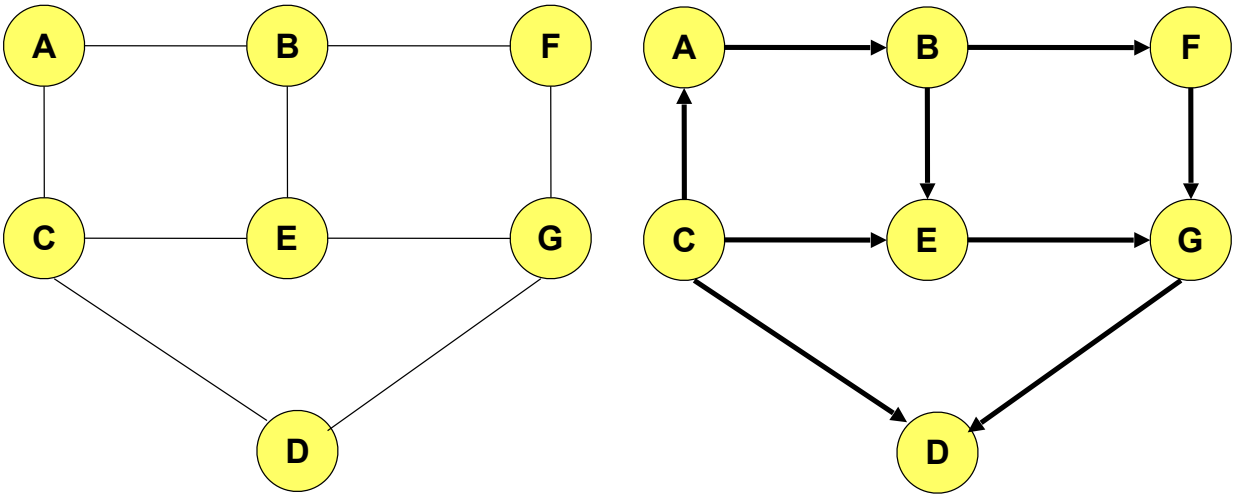
- un noeud X rediffuse un RREQ reçu de Y seulement si le lien (X, Y) possède une forte stabilité du signal de transmission ;
- la stabilité du signal est déterminée en moyennant les modifications de ce signal lors d'échange de paquet précédemment reçus.

Link Reversal Algorithm

Fonctionnement

on définit une orientation sur chaque noeud.

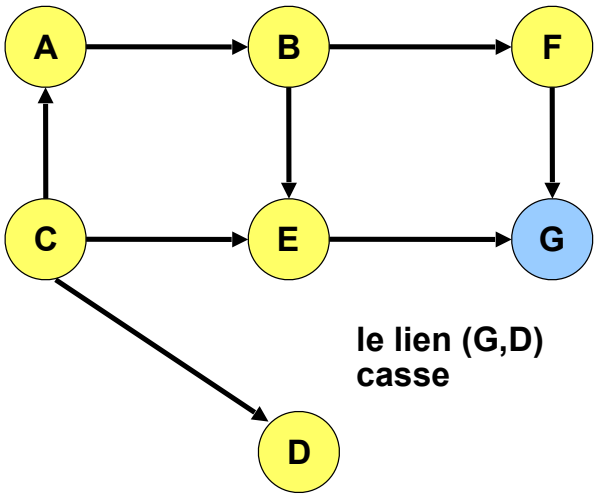
On veut construire des graphes orientés sans cycle (Directed Acyclic Graph) pour chaque destination où la destination est le seul noeud dont on ne peut plus sortir.



Link Reversal Algorithm

Chaque noeud différent de la destination qui ne possède pas de liens en sortie renverse tous les liens qu'il possède en entrée.

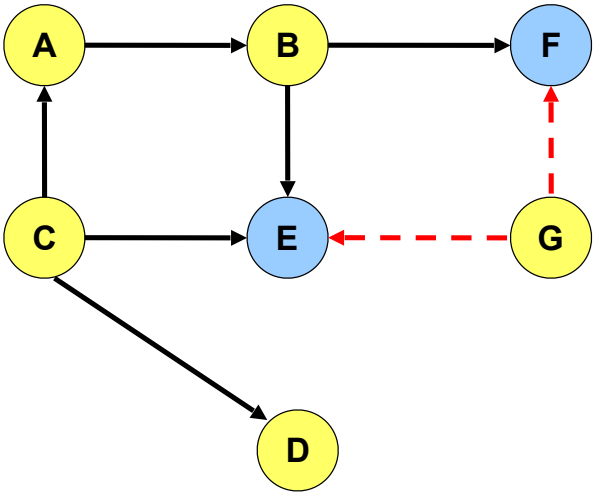
G n'a plus de lien en sortie :



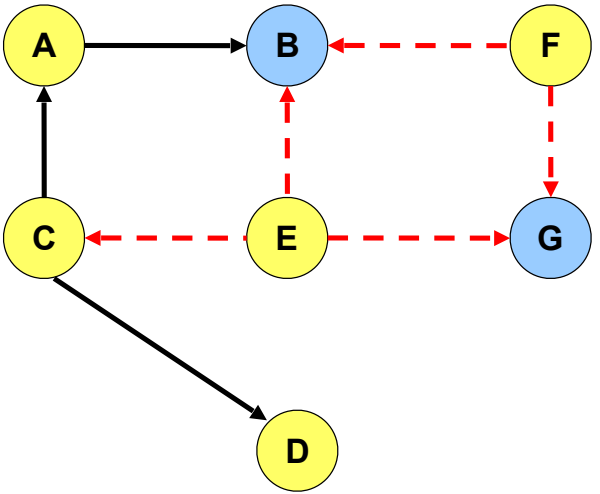
LRA

Le noeud G renverse ses liens

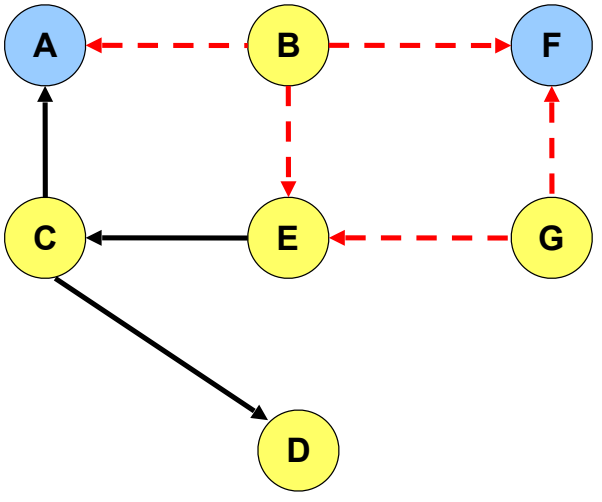
Les noeuds F et E ne possèdent plus de liens en sortie.



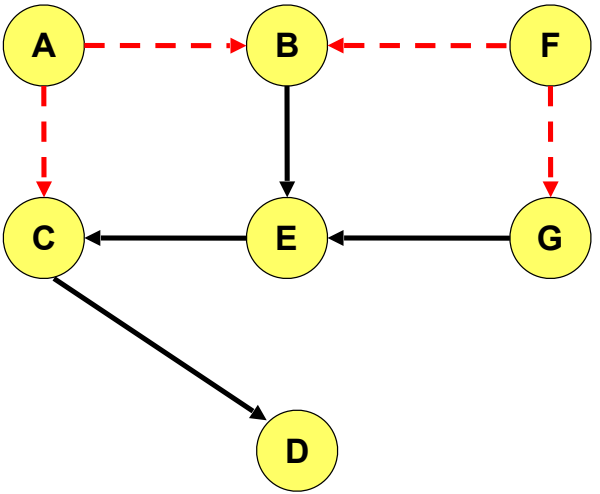
E et F renversent leur liens.
Les noeuds B et G ne possèdent plus de liens en sortie…



au tour de A et F



De nouveau tous les noeuds autre que la destination possède un lien en sortie.
Le DAG a été reconstruit.



LRA

Création

Lorsqu'un paquet est envoyé vers la destination, le DAG de cette destination est construit. La construction initiale correspond à l'utilisation de l'inondation.

Avantages

Il y a une limitation des modifications des tables de routage des noeuds proche d'un lien cassé ;

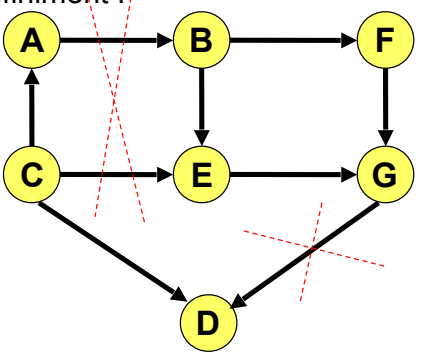
Chaque noeud peut avoir plusieurs routes vers une même destination.

Inconvénients

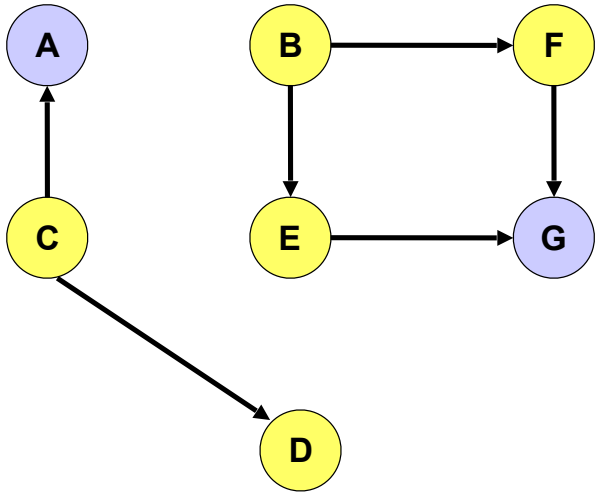
Il est nécessaire de disposer d'un mécanisme permettant de détecter qu'un lien est cassé.

Si le réseau est partitionné, la méthode peut continuer indéfiniment !

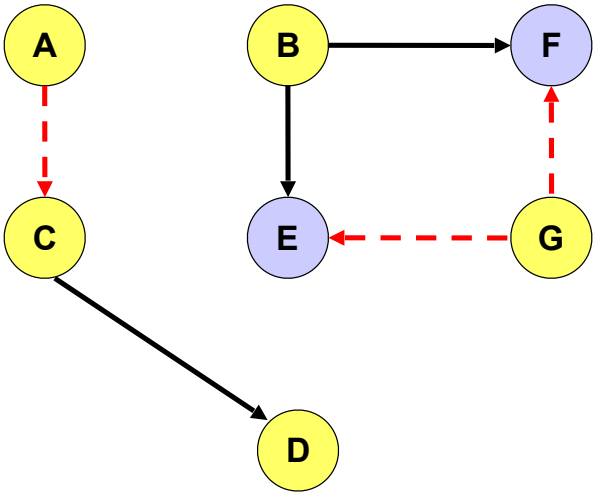
Exemple : le DAG correspond à la destination D



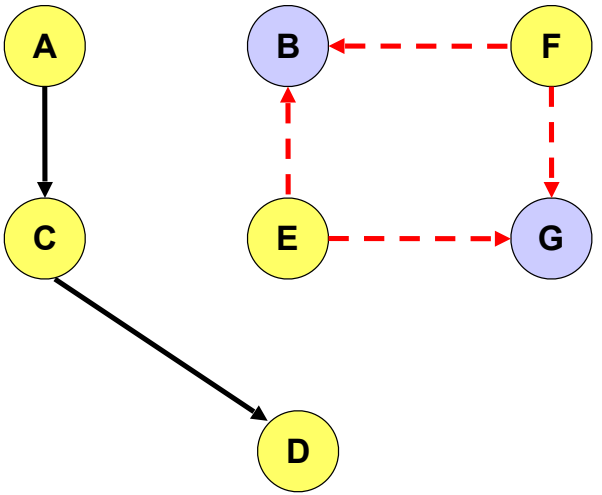
A et G n'ont plus de liens sortants.



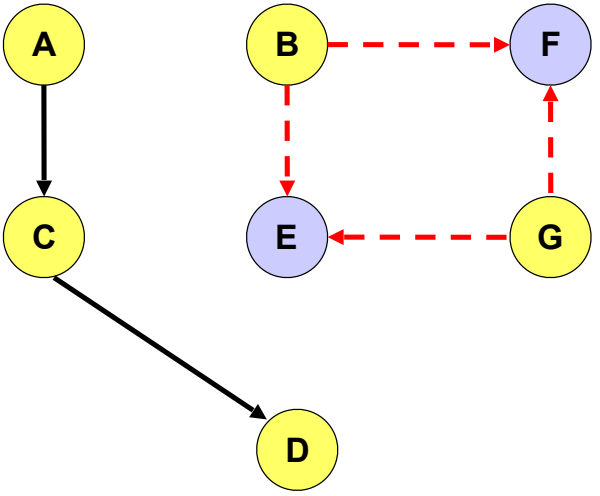
E et F remettent à jour leurs liens…



B et G n'ont plus de liens sortants



E et F n'ont plus de liens sortants…la méthode continue !
Si la partition est déconnectée de la destination alors il y a une boucle.



LRA

Il est possible de **diminuer** la quantité de liens à renverser en utilisant une **méthode partielle**.

Full reversal method

L'algorithme impose qu'un nœud **ne disposant pas de liens en sortie** (à part la destination) **inverse** tous ses liens entrants.

Partial reversal method

- si tous les voisins ont déjà renversé leurs liens, alors le nœud renverse tous ses liens entrants ;
- un noeud ne renverse ses liens entrants **que depuis ses voisins** qui n'ont pas déjà eux-mêmes renversé leur liens précédemment ;
- « *précédemment* » dans le noeud X signifie « *depuis le dernier renversement de lien effectué par le noeud X* ».

Temporally-Ordered Routing Algorithm

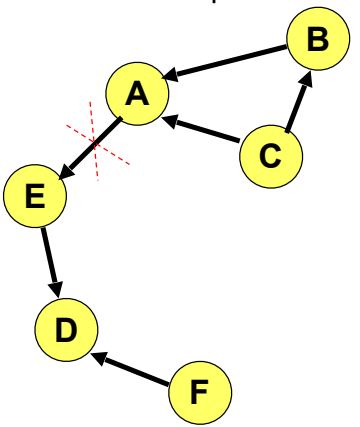
L'obtention de route optimale n'est pas importante : l''utilisation de route plus longue est possible.

Sur chaque noeud l'algorithme TORA est exécuté pour chaque destination, il calcule la hauteur du noeud par rapport à la destination.
Cette hauteur correspond au nombre de saut pour atteindre la destination.
La découverte d'une route est obtenue en utilisant des requêtes et des mises à jour.

Le principe de cet algorithme est de reprendre le Link Reversal Algorithm en version partielle et de l'améliorer en lui permettant de détecter les patitionnements.

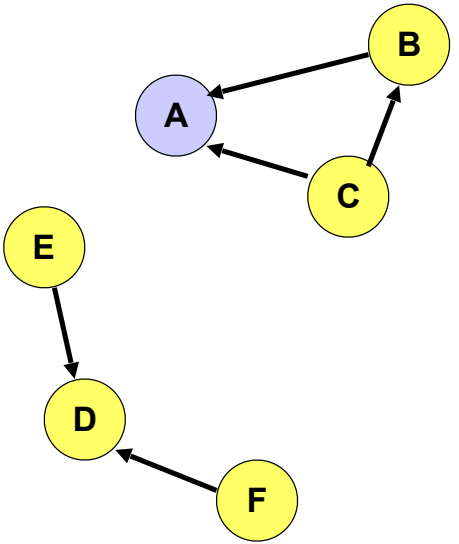
Lorsqu'une partition est détectée, tous les noeuds de la partition sont informés et la méthode d'inversion de lien est stoppée.

Exemple : un DAG pour la destination D



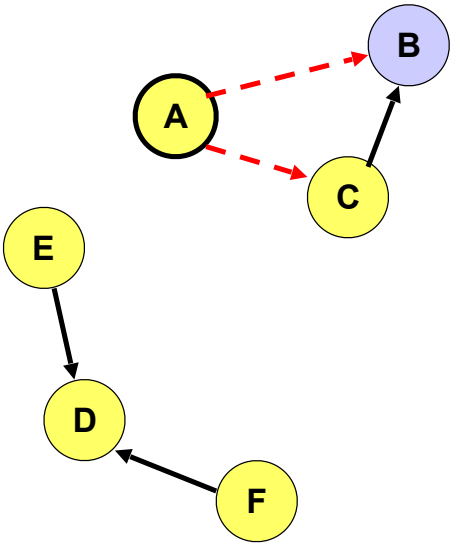
TORA

Le nœud A ne dispose plus de liens en sortie, et en particulier vers la destination D.
Il va “découvrir” qu’il est dans une partition.



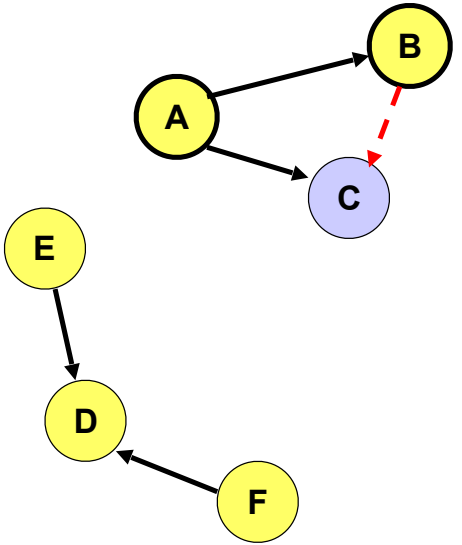
TORA

Le nœud B ne dispose plus de liens en sortie



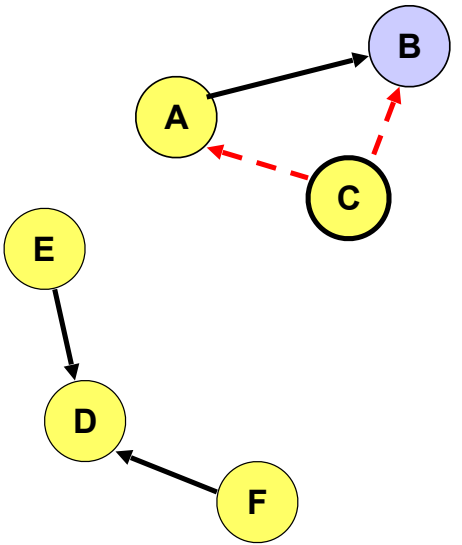
TORA

Le noeud B n'a plus de lien en sortie, et tous ses voisins ont déjà inversé leurs liens précédemment.



TORA

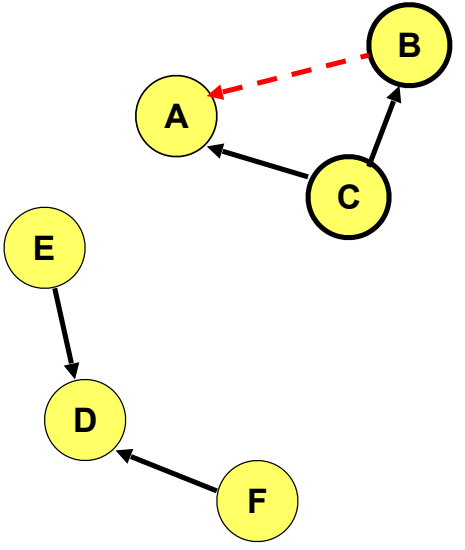
Les noeuds A et B reçoivent la réflexion depuis C.
B n'a plus de lien en sortie.



TORA

Le noeud B propage la réflexion vers le noeud A.

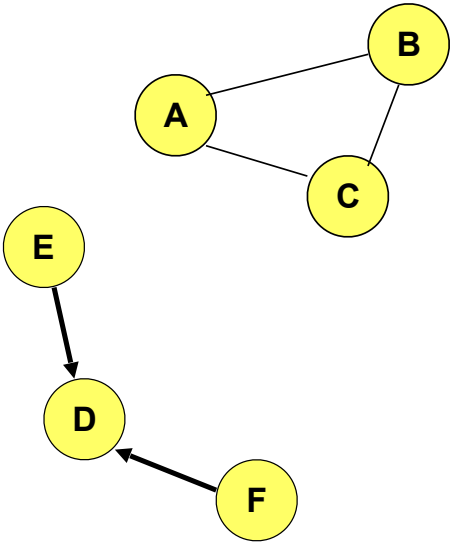
Le noeud A reçoit la réflexion depuis tous ces voisins.
A détermine qu'il est dans une **partition** qui l'isole de D.



Seul A peut le déterminer car il servait de lien vers D.

TORA

Le noeud A envoie un message CLR, clear, pour purger toutes les directions mises sur les liens dans cette partition



Algorithmes de routage proactifs

Chaque noeud **diffuse par inondation** le statut de ses liens.
Chaque noeud **rediffuse l'état des liens** reçu de ses voisins.
Chaque noeud **garde** une **trace** de **l'état des liens** reçus depuis les autres noeuds.
Chaque noeud utilise les informations précédentes pour déterminer le prochain saut (next hop) pour atteindre une destination.

DSDV Destination-Sequenced Distance-Vector
Chaque noeud maintient une table de routage qui contient :
- pour chaque saut (hop), le coût de chaque destination ;
- un numéro de séquence qui est **créé par la destination elle-même ;**
- ce numéro de séquence est utilisé pour éviter la formation de boucle.

Chaque noeud de manière périodique transmets sa table de routage a ses voisins :
- chaque noeud incrémente et ajoute son propre numéro de séquence lorsqu'il transmet sa table de routage ;
- ce numéro de séquence est attaché aux entrée dans la table de routage créées pour ce noeud.

Chaque route est étiquetée avec un numéro de séquence : les routes possédant un plus grand numéro de séquence sont préférées.

Lorsqu'un noeud découvre qu'une route n'est plus bonne, il incrémente son numéro de séquence et la diffuse avec une métrique infinie (coût pour atteindre cette destination).

Une destination diffuse son nouveau numéro de séquence.

Exemple

Le noeud X reçoit depuis Y une information concernant une route vers Z

Soient S(X) le numéro de séquence de destination pour le noeud Z tels qu'il est stocké sur X et S(Y) tel qu'il a été envoyé par Y.

- Si S(X) > S(Y), alors X ne tient pas compte de l'information reçue par Y ;
- Si s(X) = S(Y) et que le coût de passer par Y est inférieur que celui associé à la route connue par X, alors X mémorise Y comme le prochain saut (next hop) vers Z ;
- Si S(X) < S(Y) alors X désigne Y comme prochain saut vers Z et S(X) est mis à jour égal à S(Y).

Optimised Link State Routing OLSR

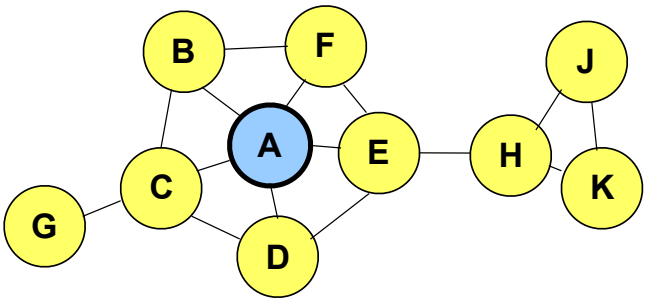
Le surcoût de l'inondation de l'état d'un lien est réduit en limitant le nombre de noeuds nécessaires à la propagation.

Une diffusion depuis un noeud X est seulement propagée par ses relais multi points.

Les relais multi-point d'un noeud X sont ses voisins tels que :
chaque voisin de X à une distance de deux en saut (two hop) et un voisin en un saut d'au moins un des relais multi-point de X.

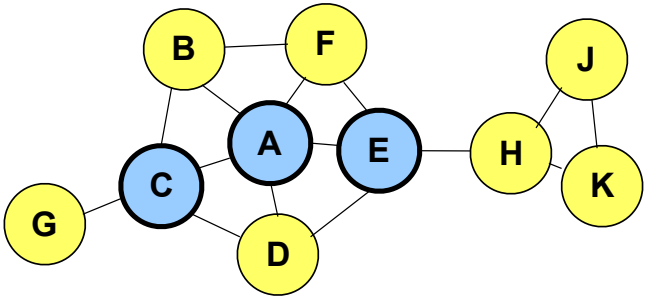
- chaque noeud transmets la liste de ses voisins périodiquement ;
- tous les noeuds peuvent connaitre leurs voisins en deux sauts ;
- chaque noeud peut désigner des relais multi-points.

Exemple : les noeuds C et E sont les relais multi-point de A

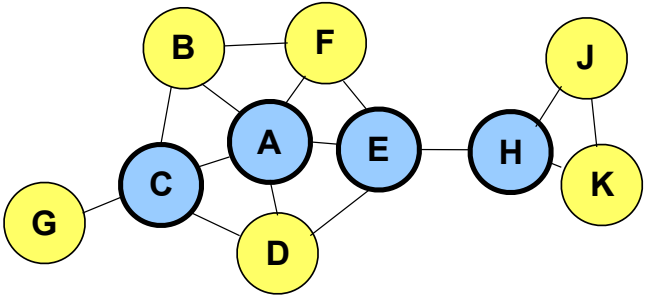


OLSR

Les noeuds C et E diffuse l'information en provenance de A (l'état des liens autour de A).



Les noeuds E et K sont des relais multi-points de H.
Le noeud K retransmet l'information reçue de H
E a déjà retransmis la même information



OLSR

OLSR ne fait une inondation qu'à travers ses relais multipoints.

L'inondation est fait uniquement depuis les relais multipoints.

Les routes utilisées dans OSR inclus uniquement les relais multipoints somme noeuds intermédiaire.

D'autres méthodes : celles hybrides
Zone Routing Protocol (ZRP)

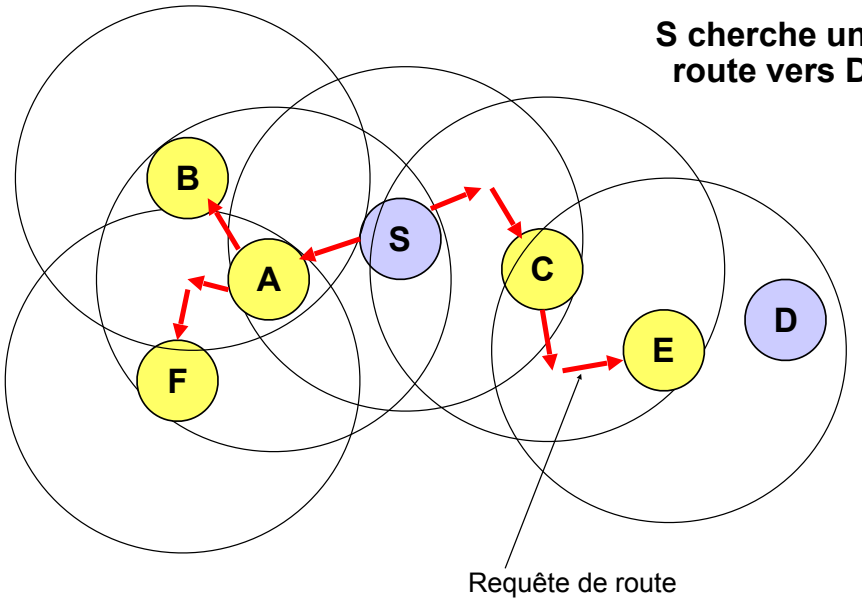
Il combine :
- des protocoles proactifs : ils utilisent des mises à jour de l'état du réseau et entretient des routes même si elles ne sont aps utilisées (pas de trafic) ;
- des protocoles réactifs : ils construisent une route vers la destination que s'il y a des données à transmettre au destinataire de cette route.

Tous les noeuds à une certaine distance de saut au plus de d depuis un noeud X sont inclus dans la zone de routage de X.
Tous les noeuds à la distance de saut exactement de d sont appelés noeuds périphériques de la zone de routage de X.

Routage :
- intra zone : on maintient de manière proactive les routes vers tous les noeuds à l'intérieur de la zone source ;
- inter zone : protocole à la demande (similaire à DSR ou AODV) pour déterminer une route vers l'extérieur de la zone.

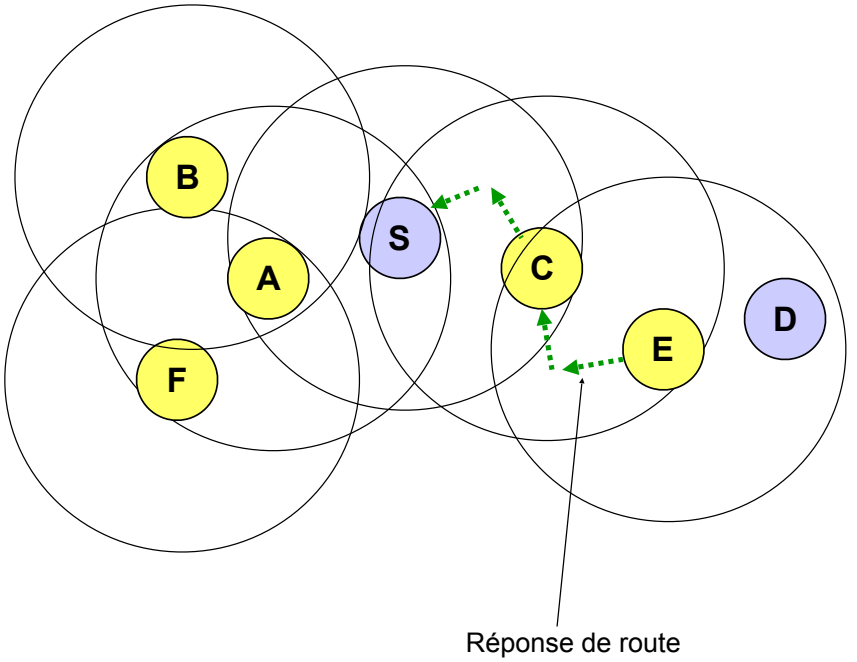
ZRP

La distance de définition de zone est d=2.



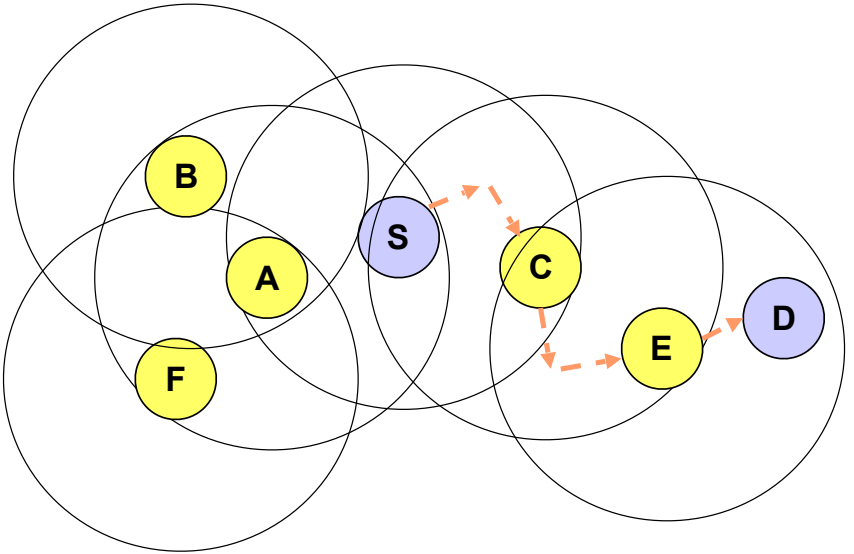
ZRP

E connait une route de E vers D, alors la requête de route doit être transmise de E vers D



ZRP

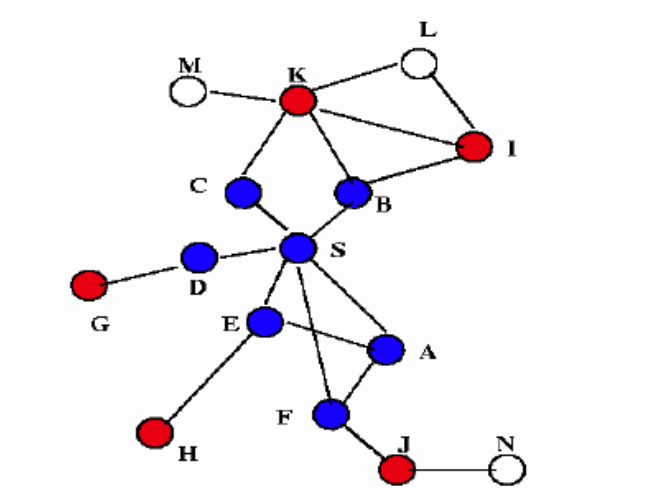
Les données sont ensuite échangées.



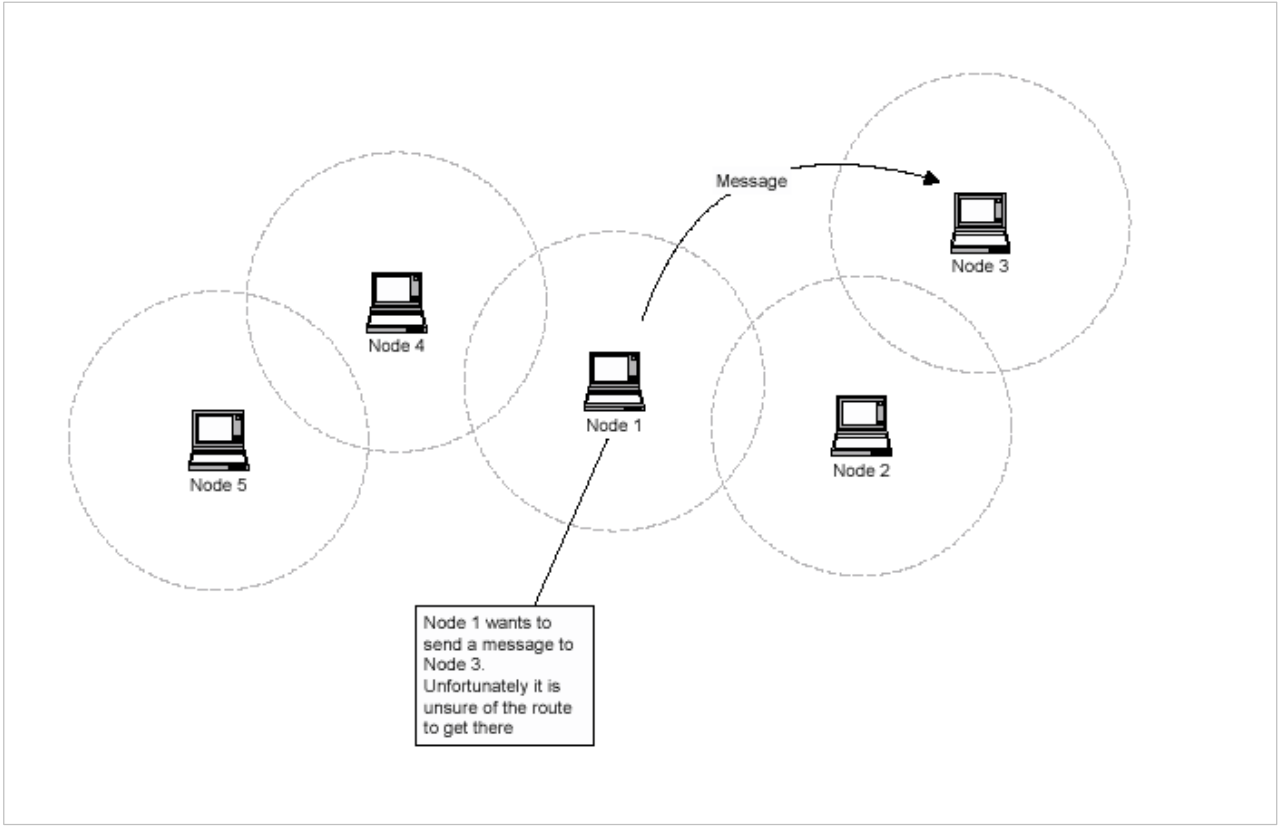
ZRP

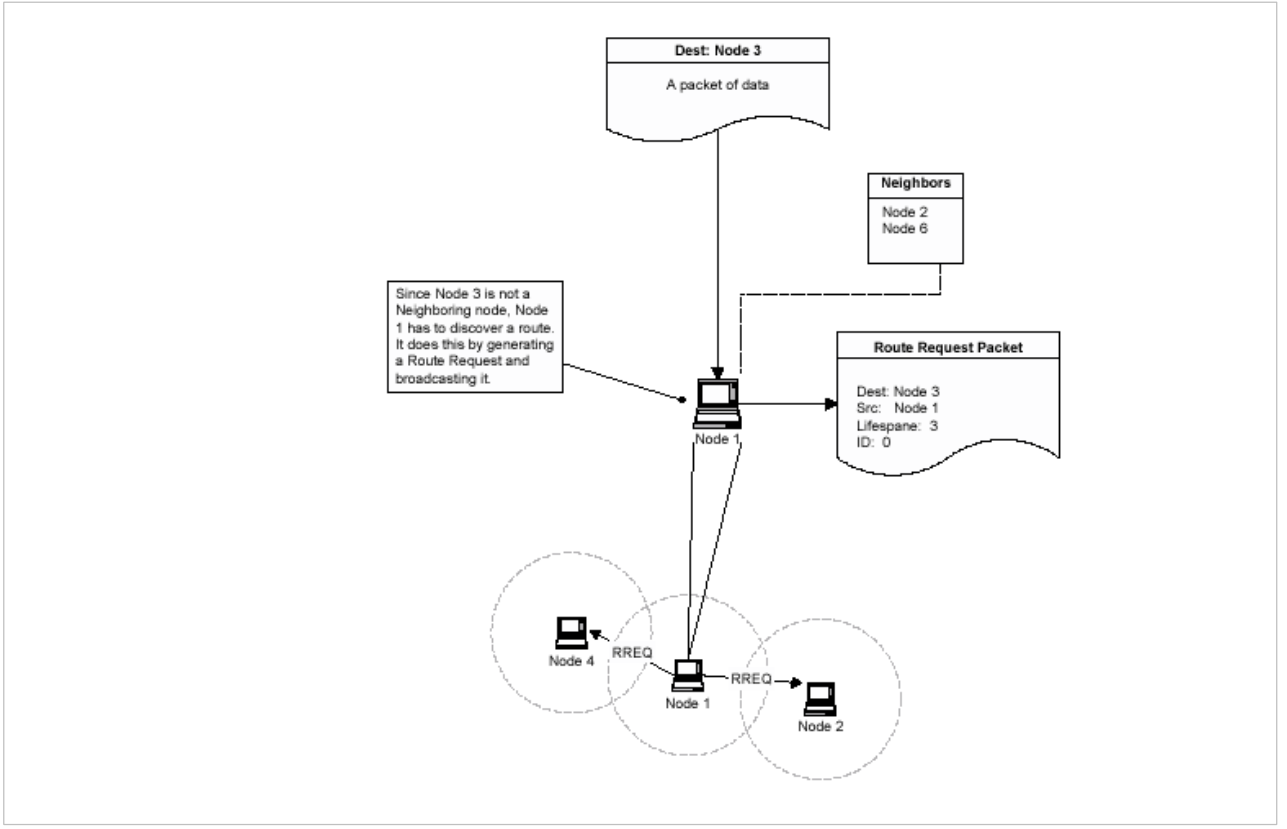
Pour S :

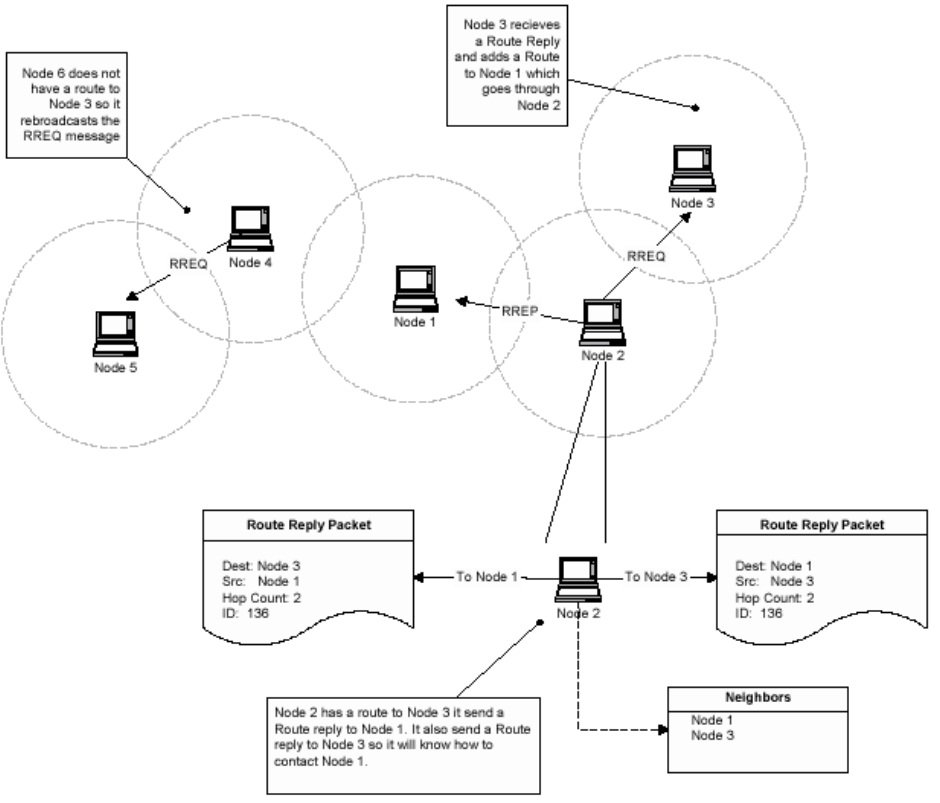
- en bleu : les noeuds internes ;
- en rouge : les noeuds périphériques.

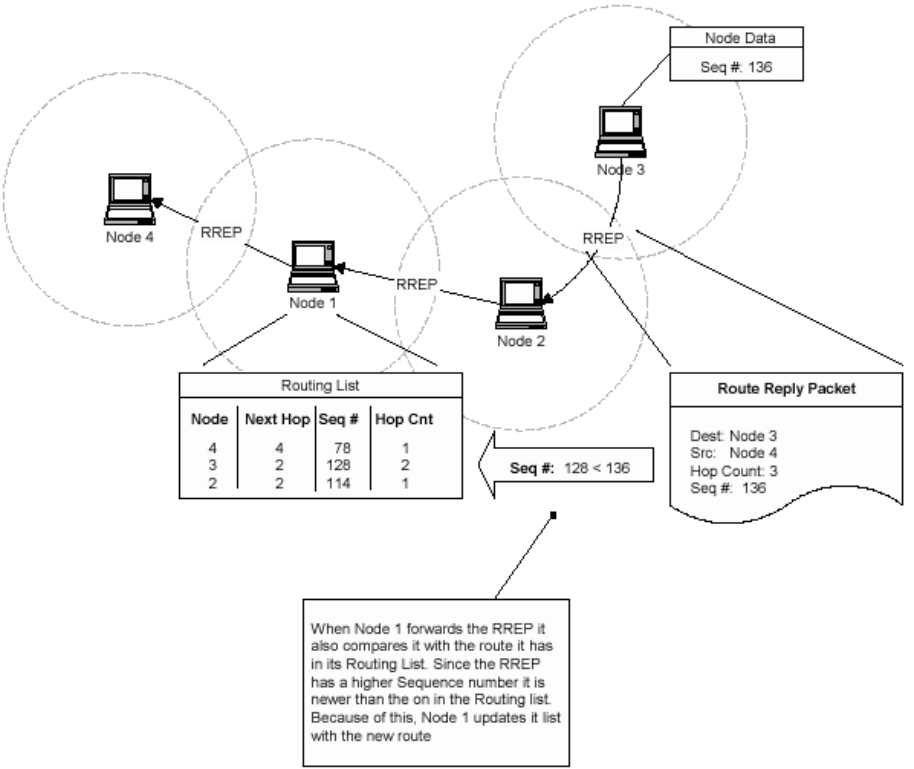


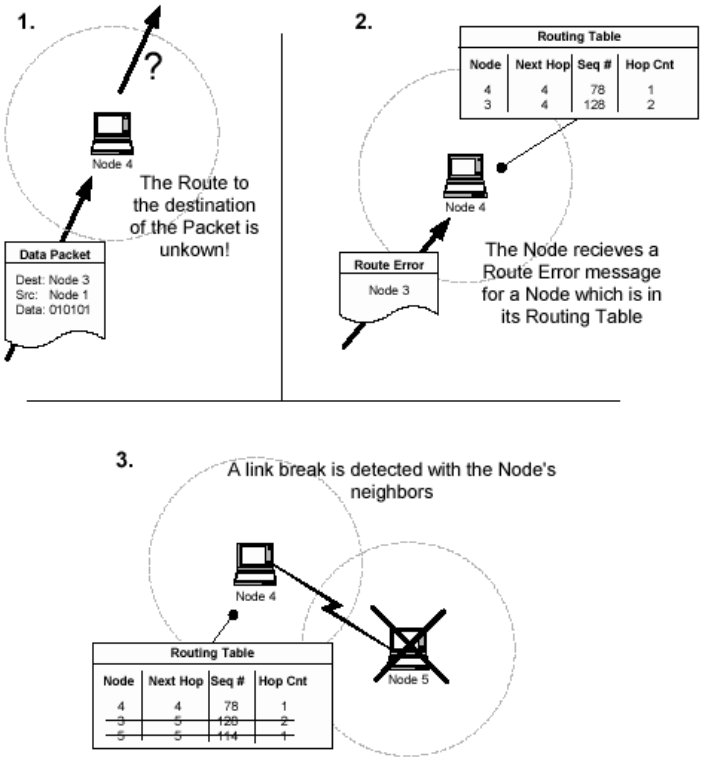
Exemple AODV




Dest: Node 3
A packet of data
Neighbors
Node 2
Node 6
Since Node 3 is not a Neighboring node, Node 1 has to discover a route. It does this by generating a Route Request and broadcasting it.
Route Request Packet
Dest: Node 3
Src: Node 1
Lifespane: 3
ID: 0
Node 1
Node 4
RREQ
Node 1
RREQ
Node 2


Node 6 does not have a route to Node 3 so it rebroadcasts the RREQ message
Node 3 recieves a Route Reply and adds a Route to Node 1 which goes through Node 2
Node 3
RREQ
Node 4
RREQ
Node 1
RREP
Node 2
Node 5
Route Reply Packet
Dest: Node 3
Src: Node 1
Hop Count: 2
ID: 136
To Node 1
To Node 3
Node 2
Route Reply Packet
Dest: Node 1
Src: Node 3
Hop Count: 2
ID: 136
Node 2 has a route to Node 3 it send a Route reply to Node 1. It also send a Route reply to Node 3 so it will know how to contact Node 1.
Neighbors
Node 1
Node 3


Node Data
Seq #. 136
Node 3
Node 4
RREP
RREP
RREP
Node 1
Node 2
Routing List
Node | Next Hop | Seq # | Hop Cnt
4 | 4 | 78 | 1
3 | 2 | 128 | 2
2 | 2 | 114 | 1
Seq #: 128 < 136
Route Reply Packet
Dest: Node 3
Src: Node 4
Hop Count: 3
Seq #: 136
When Node 1 forwards the RREP it also compares it with the route it has in its Routing List. Since the RREP has a higher Sequence number it is newer than the on in the Routing list. Because of this, Node 1 updates it list with the new route


1.
?
Node 4
The Route to the destination of the Packet is unkown!
Data Packet
Dest: Node 3
Src: Node 1
Data: 010101
2.
Routing Table
Node | Next Hop | Seq # | Hop Cnt
4 | 4 | 78 | 1
3 | 4 | 128 | 2
Node 4
Route Error
Node 3
The Node recieves a Route Error message for a Node which is in its Routing Table
3.
A link break is detected with the Node's neighbors
Node 4
Node 5
Routing Table
Node | Next Hop | Seq # | Hop Cnt
4 | 4 | 78 | 1
3 | 5 | 128 | 2
5 | 5 | 114 | 1