

République Algérienne Démocratique et Populaire
Ministère de L'enseignement Supérieur et de la Recherche
Scientifique

Université Abderrahmane Mira de Béjaia
Faculté des Sciences et des Sciences Exactes
Département Informatique



Mémoire de Fin de cycle
En vue de l'obtention du diplôme de Master en informatique

Option : Réseaux et Systèmes Distribués

Thème :

Problème de congestion dans le routage DTN

Réalisé par :

M^{elle} IDIR Sihem

M^{elle} MAKHLOUFI Assia

Devant le jury composé de :

Président : *M^r* ALLEM Khaled

Examineur : *M^r* KABYL Kamal

Examineur : *M^r* SALHI Nadir

Promoteur : *M^r* TOUAZI Djoudi

Promotion 2012

Remerciements

”Tout ce qui commence bien, finit bien ”

Nos sincères remerciements à Dieu le Tout Puissant pour le courage, la force, la volonté, et la santé qu’il nous a données et qui nous ont permit d’aboutir à nos objectifs malgré toutes les difficultés que nous avons rencontrées.

Nous tenons à remercier notre promoteur Monsieur TOUAZI Djoudi de nous avoir fait l’honneur d’assurer l’encadrement de notre travail ainsi que pour sa sympathie, et pour nous avoir donné les moyens et l’assistance nécessaire à la réalisation de ce mémoire de fin d’étude. Son expérience et ses conseils nous été précieux tant pour la compréhension du domaine que pour apprendre à présent les connaissances acquises et les travaux réalisés tout au long de cette période.

Nos vifs remerciements vont à :

Dr SAIDANI Boualam , pour sa sympathie, ses précieux conseils et ses encouragements. Nous tenons à remercier aussi Monsieur KABYLE Kamal pour ses conseils et ses idées enrichissantes.

Messieurs SALHI Nadir , et le président ALLAM Khaled d’avoir accepté de juger notre travail.

Sans oublier l’ensemble du corps professionnel de l’université de Bejaia qui nous a encadrés tout au long de notre cycle universitaire.

Un grand merci à tous nos collègues en Master 2.

Un merci pudique : à nos familles, pour leur soutien qui nous a poussé à chercher au fond de nous la volante de faire toujours beaucoup plus, à nos amis et tous ceux qui ont contribué de près ou de loin à l’aboutissement de ce modeste travail.

Dédicaces

Je dédie ce modeste travail à :

La mémoire de mes chers grands parents , oncle Hamza et mon frère Khaled que Dieu les accueille dans son vaste paradis.

À mes parents que Dieu les garde et les protège.

Mes chers frères (Omar, Walid, Terek, Mamin, Yanis, Ayoub et Md Khaled) et mes sœurs (Mouna avec son marie Hamou et son petit fils Anas, Kenza, Asmaa, Soumia et Sara).

À tous mes oncles (Lakhder, Azzouz et surtout Soufian) et leur épouses et mes tantes (Hada, Zakia, Fatiha, Yamina, Karima et ses enfants- Hamza et Aicha-)

*À tous mes cousins et cousines surtout les deux Meriem, kahina et kawther.
Tous mes amis.*

À tous les enseignants et les étudiants du département informatique en particulier, ceux de ma promotion.

Sihem IDIR.

Dédicaces

Je dédier ce modeste travail à la seule personne qui a toujours été à mes cotés pour me soutenir, me conseiller et me concilier : à mère " Mon âme" ,

À mon père " Mon trésor",

À mon cher frère ELYES,

À ma chère sœur LINDA,

que je leurs souhaite tout le sucée et le bonheur de monde entier dans leurs vies.

À la mémoire de ma grande mère que Dieu l'accueille dans son vaste paradis,

À mes tantes et mes oncles ,

À mes cousines et cousins,

À toute mes aimées, et tous mes collègues ,

À toute personne qui est heureuse pour moi aujourd'hui.

Assia MAKHLOUFI.

Résumé

Avec l'avènement des nouvelles technologies de l'information et de la communication, les réseaux basés sur la technologie TCP/IP de l'internet ont pris beaucoup d'ampleur dans le monde. Ceci a vu naître différents réseaux à base de différentes technologies, parmi ces réseaux il y a les réseaux tolérant aux délais ou DTN (Delay Tolerant Network). Ce type de réseau est caractérisé par une longue période de déconnexion, de grand retard de communication, et une communication opportunistes sur des liaisons intermittentes. Ce qui provoque le problème de congestion lors de la communication.

Les protocoles de routage dans les réseaux tolérant aux délais ne cherchent pas à trouver le chemin le plus court tel que c'est le cas dans les protocoles classiques, mais ils jouent plutôt sur l'augmentation du taux de délivrance de données. En outre, les messages dans les réseaux DTN ne sont pas simplement routés à travers le réseau, ils peuvent être stockés temporairement sur certains hôtes, pour être réémis plus tard lorsque les circonstances le permettent, ce qui cause entre autre le problème de congestion. Nous avons proposé dans ce mémoire un algorithme de flot max qui cherche un chemin qui garentit la transmission des paquets de la source à la destination d'une manière à éviter le problème de congestion.

Mots-clés : routage internet, Delay Tolerant Networks, architecture des réseaux informatiques DTN, routage dans les réseaux informatiques, Les protocoles de routage informatiques.

Abstract

With the advent of new information technologies and communication networks based on TCP / IP technology of the Internet have grown dramatically in the world. This has been born different networks based on deferent technology, among these networks there are networks Delay Tolerant or DTN (Delay Tolerant Network). This type of network is characterized by a long period of disconnection, large communication delay, and an opportunistic communication over intermittent connections. What causes the problem of congestion in the communication.

Routing protocols in Delay Tolerant Networks does not seek to find the shortest path as is the case in conventional protocols, but they are rather on increasing the rate of delivery of data. In addition, messages in DTN networks are not simply routed through the network, they can be stored temporarily on certain hosts, to be reissued later when circumstances permit, that among other causes of the problem congestion. Nous propose in this paper a max flow algorithm that seeks a path that garenté packet transmission from source to destination in a manière avoided the problem of congestion.

Keywords : Internet routing, Delay Tolerant Networks, Architecture DTN networks, routing in computer networks, computer routing protocols.

Liste des abréviations

- AODV** : Ad hoc On Demand Distance Vector
- CLA** : Convergence Layer Adapter
- DTN** : Delay Tolerant Network
- DSDV** : Destination-Sequenced Distance Vector
- DSR** : Dynamic Source Routing
- DTCP** : Disconnected Transitive Communication Protocol
- ED** : Earliest Delivery
- EDLQ** : Earliest Delivery with Local Queuing
- EDAQ** : Earliest Delivery with All Queues
- ICMP** : Internet Control Message Protocol
- IP** : Internet Protocol
- LTP** : Licklider Transmission Protocol
- MED** : Minimum Expected Delay
- MEED** : Minimum Estimated Expected Delay
- OLSR** : Optimized Link State Routing
- RAPID** : Ressource Allocation Protocol for International DTN routing
- RDV** : Rendez-vous
- TCP** : Transport Control Protocol
- TCPCL** : TCP Convergence Layer
- UDP** : User Datagram Protocol

Table des matières

Introduction générale	1
1 Introduction sur les réseaux DTN	2
1.1 Introduction	2
1.2 Définition d'un DTN	2
1.3 Caractéristiques des environnements cibles d'un réseau DTN	3
1.3.1 Connectivités	3
1.3.2 Délais de propagation	4
1.3.3 Débits des données	5
1.3.4 Taux d'erreur	5
1.4 Architecture des réseaux DTN	5
1.4.1 Les entités de communication	5
1.4.2 Le Fonctionnement des DTN	9
1.5 Conclusion	20
2 Routage dans les réseaux DTN	21
2.1 Introduction	21
2.2 Types de routage	21
2.2.1 Routage statique	21
2.2.2 Routage dynamique	22
2.3 Concepts de base du routage DTN	22
2.3.1 Routage proactif	22
2.3.2 Routage réactif	22
2.3.3 Routage source	23
2.3.4 Routage par saut	23
2.3.5 Routage hiérarchique	23
2.4 Le scénario de routage dans un DTN	24
2.5 Propriétés des stratégies	24
2.5.1 Réplication	24
2.5.2 Connaissance	25
2.6 Familles des stratégies	25
2.6.1 Stratégies Inondations	25
2.6.2 Stratégies d'expédition	28
2.7 Quelques protocoles de routage pour les réseaux DTN	29
2.7.1 Disconnected Transitive Communication Protocol(DTCP)	29
2.7.2 Ressource Allocation Protocol for International DTN routing (RAPID)	30
2.7.3 Protocole MaxProp	30
2.8 Conclusion	30

3	Problématique et proposition	32
3.1	Introduction	32
3.2	Problème de congestion	32
3.2.1	Les solutions existantes	33
3.3	Définitions et propriétés :	33
3.4	Problème du flot maximum dans un réseau DTN	35
3.5	Quelques algorithmes de flot max	35
3.5.1	Algorithme de Ford et Fulkerson	35
3.5.2	L'algorithme d'Edmonds et Karp	38
3.5.3	L'algorithme de Floyd-Warshall	39
3.5.4	Table de comparaison	41
3.6	Choix d'algorithme	41
3.7	La solution proposée	42
3.8	Conclusion	46
	Bibliographie	48

Table des figures

1.1	Architecture d'un réseau DTN	3
1.2	Mise en évidence de la connectivité (bout en bout) d'un système constitué de plusieurs tronçons	4
1.3	L'appartenance des nœuds aux différentes régions	6
1.4	Les différents rôles d'un nœud DTN	6
1.5	Identification des régions dans un DTN	7
1.6	Adressage d'un nœud dans un DTN	9
1.7	La technique de commutation Store-and-Forward	10
1.8	Emplacement de la couche Bundle dans la pile protocolaire	11
1.9	La couche Bundle présente dans toutes les régions	12
1.10	Structure d'un transmetteur de bundle	12
1.11	Principe du transfert de garde (Custody Transfer)	14
1.12	Protocole non conversationnel	15
1.13	Les options de livraisons offertes par la couche bundle.	17
2.1	Routage avec contact direct	26
2.2	Routage avec deux sauts	26
2.3	Routage avec inondation à base d'un arbre	27
2.4	Routage base sur la localisation	28
3.1	Contraintes de graphe	35
3.2	Exemple de flot max.	44
3.3	Marquage de la source S	44
3.4	Marquage de tous les sommets	45
3.5	La chaîne augmentante	45
3.6	Le flot amélioré	46

Liste des tableaux

1.1	Comparaison entre les réseaux TCP/IP et les réseaux DTN	10
3.1	Tableau comparatif des algorithmes de flot max.	41

Introduction générale

De nos jours la communication et le transfert de données se fait principalement à travers les réseaux internet en parcourant un ensemble de lien homogène avec une bande passante relativement élevée et très bas délais.

En effet ce genre de scénario n'est pas applicable à quelques types de communications, du fait de la longue distance qui sépare la source de la destination, du la non disponibilité d'une infrastructure internet à côté de l'intermittence de la connectivité du réseau, du taux élevé d'erreurs et du manque de chemin bout en bout. Ceci a donné naissance à nouveau type de réseau appelé réseaux tolérant aux délais ou DTN (Delay Tolerant Network) qui recouvrent tous les types de réseaux hétérogènes.

Contrairement à internet, ces réseaux peuvent supporter des délais plus importants et variables, des longues périodes de déconnexion,et un taux d'erreur élevé. Chaque réseau régional répond donc à des contraintes physiques bien particulières. Les réseaux régionaux ne sont pas conçu pour être interconnectés entre eux cependant par l'utilisation de passerelles spécifiques on peut créer un pont de communication entre deux réseaux.[18]

Dans ce rapport nous commençons par introduire le fondement et les caractéristiques des réseaux DTN dans le premier chapitre, en montrant la différence entre le routage TCP/IP et le routage DTN et l'introduction d'une nouvelle couche dite la couche Bundle entre la couche application et les couches de bas niveau, ainsi que le principe du transfert de garde dans un réseau DTN et l'utilisation du mécanisme Store and Forward pour acheminer les messages de nœud à nœud à cause de la déconnexion permanente et du manque de chemin de bout en bout. Dans le deuxième chapitre, nous allons parcourir les types de routage classique et montrer en détails les concepts de base de routage DTN, ainsi que les propriétés des stratégies utilisées dans ce dernier. En outre, nous illustrons quelques protocoles de routage dans les réseaux DTN.

Finalement, le troisième chapitre est consacré pour la proposition, nous l'introduisons par une petite présentation de la problématique, passons à quelques algorithmes de flot max et le choix de la proposition. Nous clôturons ce mémoire par une conclusion générale.

1

Introduction sur les réseaux DTN

1.1 Introduction

Les missions spatiales lointaines, comme par exemple l'exploration de Mars, voient leurs moyens de télécommunications soumis à des contraintes importantes en termes de connectivité, de délai, de débit et de consommation énergétique.

Les protocoles d'un réseau terrestre ne peuvent pas répondre aux conditions de communications et d'échanges dans certains environnements difficiles. C'est pour ça qu'on a eu recours aux "Réseaux Tolérant aux Délais".[\[18\]](#)

Nous présentons dans ce chapitre le plus que peut apporter ce type de réseau et les contraintes pour utiliser ces nouvelles technologies.

1.2 Définition d'un DTN

Un réseau DTN est composé de plusieurs réseaux sujets à des connexions intermittentes. Ces réseaux ne sont pas nécessairement homogènes ni en termes de technologies ni par rapport aux protocoles utilisés. Les DTN sont composés de nœuds appelés "nœuds DTN". Dans la Figure 1.1, nous montrons un exemple d'architecture DTN composée de trois réseaux avec deux nœuds DTN. Les nœuds DTN servent à interconnecter les réseaux qui, en principe, peuvent

avoir différentes piles de protocoles. Ils servent également à surmonter les interruptions de communication entre les réseaux.[18]

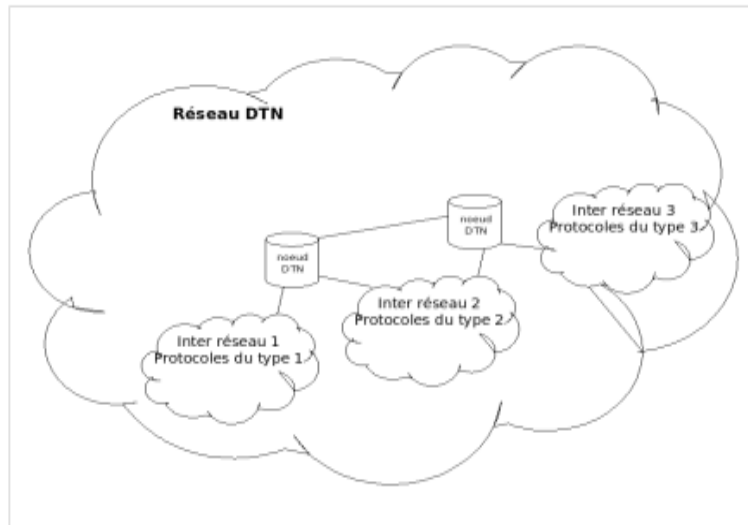


FIGURE 1.1 – Architecture d'un réseau DTN

1.3 Caractéristiques des environnements cibles d'un réseau DTN

Le réseau DTN est basé sur le principe suivant : ” Construire un réseau de réseaux Internet ”. L'idée était de déployer des réseaux Internet standards dans des endroits éloignés, en l'occurrence : les planètes.

Les environnements d'un réseau DTN présentent les caractéristiques suivantes [16] :

1.3.1 Connectivités

Les protocoles utilisés dans les réseaux terrestres reposent sur le principe de connectivité permanente de bout en bout : une connexion est établie pendant une durée particulière de l'émetteur vers le récepteur. La connectivité dans les environnements interplanétaires est au contraire intermittente.

En effet, le caractère (bout en bout) peut être rompu par une configuration orbitale particulière, une planète masquant le trajet habituel, ou encore par des contraintes en termes de temps d'opération des équipements. La figure 1.2 illustre cette problématique.

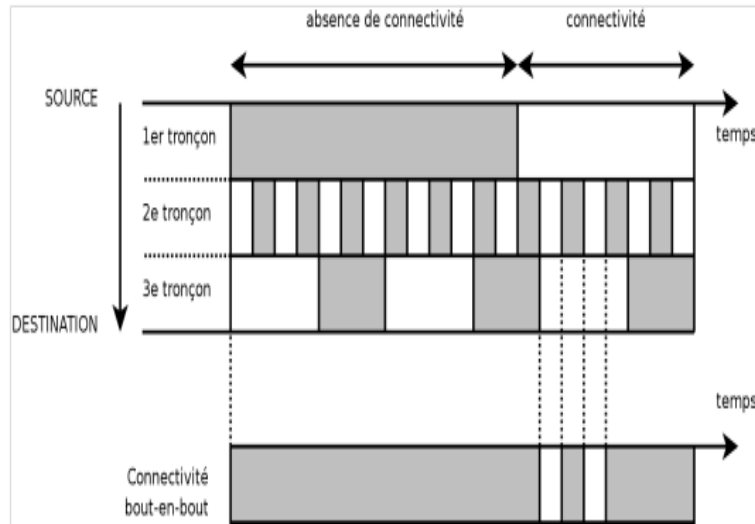


FIGURE 1.2 – Mise en évidence de la connectivité (bout en bout) d'un système constitué de plusieurs tronçons

D'autre part, les conditions requises pour une connexion de bout en bout peuvent également être remplies pour une durée inférieure à la durée nécessaire pour transmettre l'ensemble des données. Les instants de disponibilité de chaque tronçon de la connexion sont déductibles des éphémérides et donc prévisibles des mois voire des années à l'avance.

Enfin, le caractère bidirectionnel de chaque tronçon de la connexion n'est pas assuré au contraire de la plupart des réseaux terrestres. En environnement interplanétaire, des tronçons unidirectionnels sont également mis en œuvre.

1.3.2 Délais de propagation

Le délai de propagation d'un signal dans l'espace dépend de la distance qui sépare l'émetteur et le récepteur, ainsi que de la vitesse de la lumière dans le vide. Compte tenu des distances mises en jeu, les délais attendus sont de l'ordre de plusieurs minutes, bien au-delà des délais terrestres comptés en millisecondes.

En raison des orbites et mouvements propres à l'émetteur comme au récepteur, la distance qui les sépare est variable. Ainsi la distance entre la Terre et Mars varie entre 56 et 400 millions de kilomètres, ce qui implique un délai maximal de 22 minutes.

1.3.3 Débits des données

Les distances de l'ordre de plusieurs millions de kilomètres ont un impact direct sur le rapport signal à bruit, et donc sur les débits qui peuvent être utilisés. Les valeurs typiques s'échelonnent de 8 à 256 kb/s.

1.3.4 Taux d'erreur

Les erreurs de bit sur une liaison exigent la correction (en ajoutant des bits et du traitement) ou la retransmission du paquet complet (donc plus de trafic réseau).

Pour un taux d'erreur donné d'un lien, moins de retransmission sont nécessaires pour une retransmission saut par saut que pour une retransmission de bout en bout.

1.4 Architecture des réseaux DTN

1.4.1 Les entités de communication

Comme schématisé sur la Figure 1.3[23], un réseau DTN est composé d'un ensemble d'entités communicantes appelées : Nœuds. Ces nœuds sont répartis en régions où chacun d'entre eux est uniquement identifié par au moins un tuple contenant le nom de la région et le nom de l'entité. Un nœud lié à plusieurs régions doit avoir au moins un tuple d'identification pour chacune des régions auxquelles il appartient [5].

1.4.1.1 Les nœuds DTN

Un nœud DTN est alors un dispositif pour l'envoi et la réception des messages (appelés aussi : bundles). Il peut jouer le rôle de : source, destination ou de nœud intermédiaire pour la transmission de bundles. Le nom du nœud DTN lui-même, par opposition à une application l'utilisant est défini dans une " région spécifique " à l'aide de l'identifiant de l'entité ou une partie de celui-ci.

Un nœud DTN peut jouer le rôle de [22] :

- **Hôte** : Envoie et/ou reçoit les bundles, mais ne les diffuse pas. Ce qui requiert un stockage persistant durant de longs délais dans lesquels les bundles seront alignés jusqu'à ce que les liens soient disponibles.

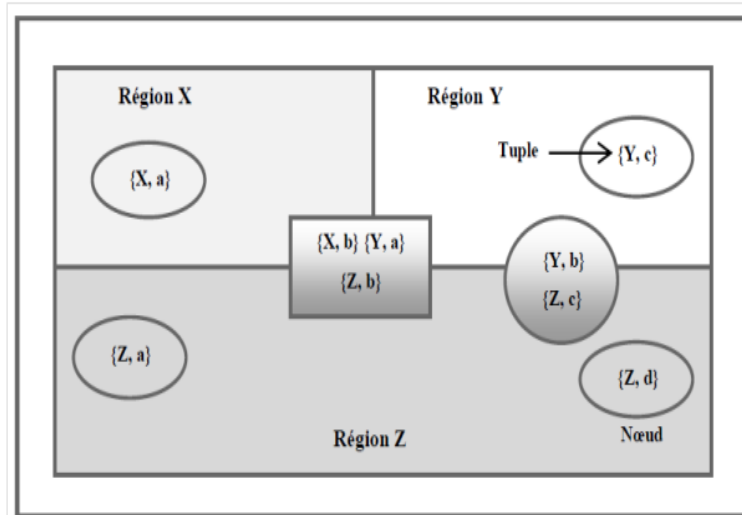


FIGURE 1.3 – L'appartenance des nœuds aux différentes régions

- **Routeur** : Diffuse les bundles au sein d'une seule région DTN et peut optionnellement jouer le rôle hôte.
- **Passerelle** : Diffuse les bundles entre deux ou plusieurs régions DTN et peut optionnellement jouer le rôle d'un hôte. Elle opère sur la couche transport et se base sur la commutation de messages plutôt que sur la commutation de paquets. Cependant, elle fournit l'interopérabilité entre des protocoles spécifiques pour une région et ceux spécifiques pour une autre.

Ces différents rôles peuvent être schématisés comme montré sur la Figure 1.4 ci-dessous [22] :

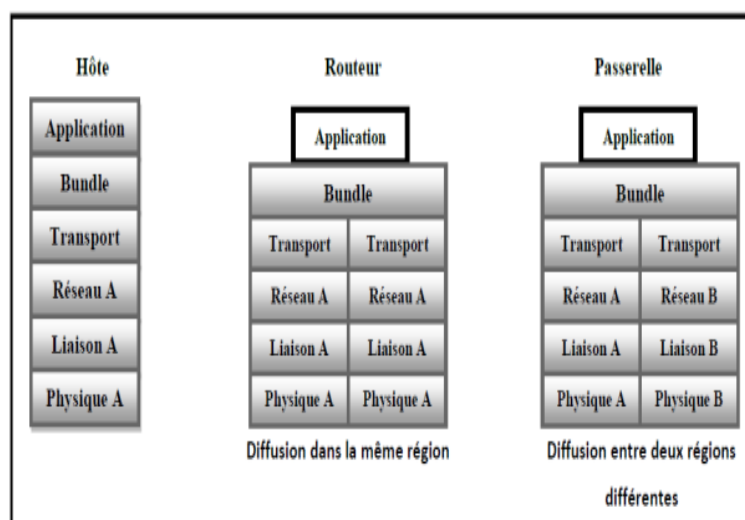


FIGURE 1.4 – Les différents rôles d'un nœud DTN

1.4.1.2 Les régions DTN

L'architecture DTN définit un réseau de plusieurs réseaux où chacun d'eux représente une région dans laquelle les caractéristiques de communication sont homogènes. Une région peut être le réseau Internet du globe terrestre, un réseau tactique militaire, la planète ou même un vaisseau spatial (Figure 1.5). En d'autres termes, une région est une zone qui est influencée par les familles de protocoles, les dynamiques de connexion, les politiques administratives ou de manière générale, les " régions DTN " sont délimitées en se basant sur un critère appelé : les frontières de confiance.

Chaque région DTN a un nom unique et connu, ou que l'on peut connaître parmi toutes les autres régions du DTN. Ainsi, un référentiel pour l'ensemble des noms des régions est nécessaire, qui grâce à lui l'on peut retrouver le nom d'une région à tout moment.

Les bundles DTN (appelés aussi : messages) originaires de régions différentes de celle de destination sont transmis en premier lieu via des entités communicantes appelées : Passerelles, qui connectent la région source à une ou plusieurs autres régions. Le routage à l'extérieur de la région destination n'est fondé que sur le nom de celle-ci et non pas sur le nom complet de la destination elle-même[22].

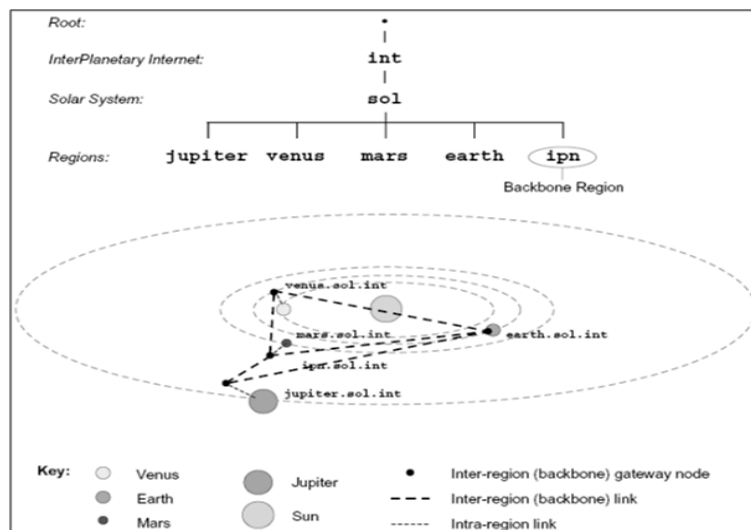


FIGURE 1.5 – Identification des régions dans un DTN

Les régions DTN sont caractérisées par :

- Chaque région doit avoir un espace identifiant partagé par tous les nœuds de la région,

et doit spécifier des conventions de nommage internes afin d'être employées pour l'identification des entités.

- Chaque nœud membre de la région est doté d'un unique identifiant tiré de cet espace identifiant. Notons que pour certains types de régions, un " nœud " peut être composé d'une collection d'éléments calculables et/ou géographiquement distribués. Un seul et unique élément s'applique sur le nœud destiné à recevoir des données provenant des autres nœuds DTN.
- Pour être considéré comme membre de la région, chaque membre potentiel de celle-ci doit être capable d'atteindre les autres membres de la même région, sans passer par d'autres nœuds DTN se trouvant à l'extérieur de celle-ci en utilisant un ou plusieurs protocoles connus au niveau de chaque nœud.
- Un nœud DTN ne doit pas nécessairement être atteint directement. Ceci peut demander une opération de Store and Forward et/ou de transmission par les autres nœuds de la même région[5].

1.4.1.3 Les tuples

Un tuple désigne le nom d'un nœud. Il est composé, comme illustré dans la Figure 1.6, de deux parties [23] :

- L'identificateur de la région (ou nom de région).
- L'identificateur de l'entité (ou nom d'entité).

Le nom de région est nécessaire et suffisant pour router un bundle de données à sa région de destination, mais ne peut pas le délivrer au point de destination spécifique auquel il a été destiné.

Le nom de l'entité est masqué à l'extérieur de la région de définition. Une entité peut être un hôte, un protocole, une application ou une agrégation de tous ceux-là selon la nature de l'adressage et de nommage des structures utilisées dans la région.

Le routage entre les régions n'est basé que sur les IDs des régions qui sont liés à leurs adresses correspondantes dans tout le DTN. Le routage à l'intérieur de la région n'est basé que sur les IDs des entités qui sont liés à leurs adresses correspondantes au sein de la région. Les passerelles appartiennent à deux ou plusieurs régions et déplacent les bundles entre ces régions,

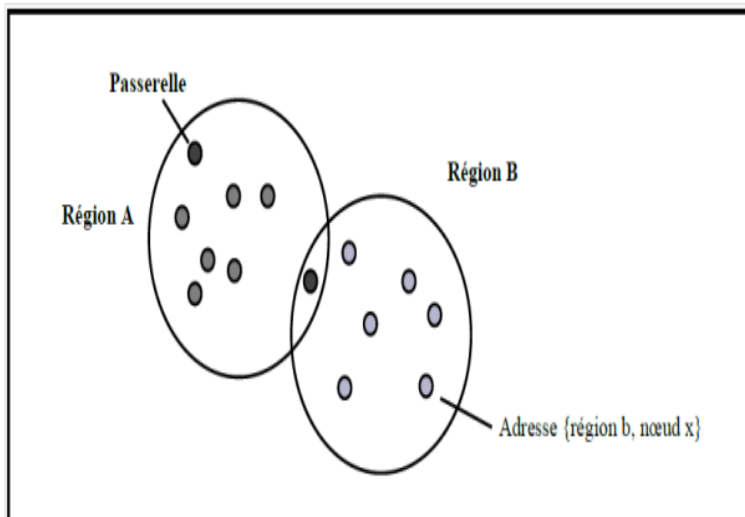


FIGURE 1.6 – Adressage d'un nœud dans un DTN

ainsi ces passerelles possèdent plusieurs IDs région [5].

1.4.2 Le Fonctionnement des DTN

1.4.2.1 Pourquoi pas TCP/IP ?

Le protocole TCP/IP sur lequel se base Internet assure une communication de bout en bout qui garantit la fiabilité de la transmission de données grâce à la combinaison de deux protocoles étroitement liés, TCP et IP. TCP est caractérisé principalement par le fait qu'il [4] :

- Exige une négociation de connexion entre la source et la destination afin de réguler le flux de données.
- Délivre les données reçues dans leur ordre de transmission. Ainsi, si un paquet est perdu, il doit attendre sa retransmission grâce aux délais courts.

Quant à la normalisation du protocole IP, elle a permis aux routeurs d'être interopérables en utilisant l'adressage logique et la technique de commutation de paquets [8]. Ce protocole prend en charge la transmission de données ou de datagrammes entre deux nœuds éloignés. Ces conditions ne sont, malheureusement, pas valides dans les DTNs, d'où la nécessité de penser à d'autres protocoles supportant leurs caractéristiques.

Le tableau suivant fait une comparaison entre les réseaux utilisant le protocole TCP/IP et les réseaux DTN [23] :

TCP/IP	Réseaux DTN
-Support des transmissions à haute vitesse : LAN, FO,...	-Support de transmission sans fil : réseaux satellitaires, ad-hoc
-Faibles délais de propagation (ms) -Faible taux d'erreurs	-Long délais de propagation (min) -Taux d'erreur élevés
-Débits de transmission élevés qq Mbps , dizaine de Gbps.	-Débit de transmission relativement faible qq kbps , dizaine de Mbps.
-Communication bidirectionnelle	-Emission/réception non simultanée pour certaines liaisons
-Connectivité maintenue continuellement de bout en bout	-Liens à connectivité discontinue
-Adressage unique	-Impossible d'utilisé un adressage unique en cas d'interconnexion entre déférentes régions

TABLE 1.1 – Comparaison entre les réseaux TCP/IP et les réseaux DTN

1.4.2.2 La commutation de message Store and Forward

Les réseaux DTN surmontent les problèmes associés à la connectivité intermittente, aux délais longs ou variables, à la vitesse de transmission asymétrique et au taux d'erreur important en utilisant la méthode de transport : **Store and Forward**.

Cette vieille méthode consiste en la transmission de messages (ou morceaux de messages) d'une zone de stockage à une autre, le long d'un chemin qui mène à la destination, La Figure 1.7 ci-dessous, schématise ce processus comme suit : chaque nœud garde le message en sa possession (phase store) jusqu'à ce qu'il rentre en contact avec un autre pour le lui transmettre (phase Forward)[11].

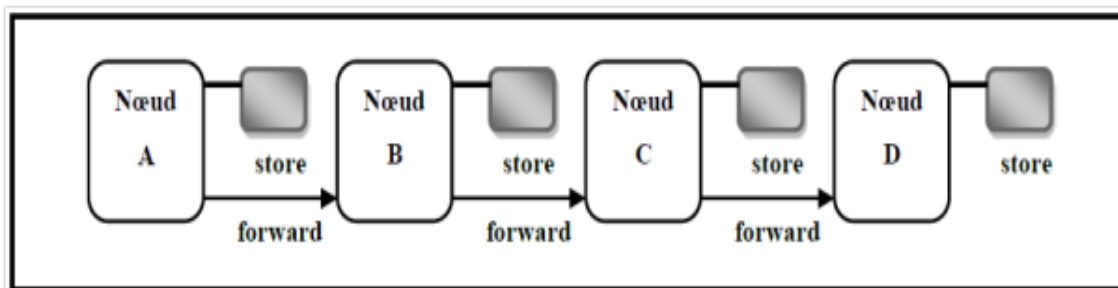


FIGURE 1.7 – La technique de commutation Store-and-Forward

La zone de stockage (comme un disque dur) peut garder un message indéfiniment. On parle de stockage persistant (contrairement à un stockage dit à court terme) fourni par les zones mémoires. Les routeurs Internet utilisent les zones mémoire pour stocker les paquets entrants pour quelques millisecondes en attendant de le commuter vers le prochain nœud, et ce en consultant leur tables de routage. Les routeurs DTN, eux, ont besoin d'un stockage persistant

au niveau de leurs files d'attente pour l'une ou les raisons suivantes :

1. Le lien de communication vers le nœud suivant peut être indisponible pour une longue période.
2. Un nœud dans une paire communicante peut envoyer ou recevoir des données beaucoup plus rapidement ou plus sûrement que les autres nœuds.
3. Un message une fois envoyé peut avoir besoin d'être retransmis si une erreur se produit.

1.4.2.3 Le protocole bundle

- **La couche bundle** : L'architecture DTN met en œuvre la méthode Store and Forward pour la commutation des bundles par l'ajout d'une nouvelle couche protocolaire appelée : La couche Bundle.

La couche bundle se situe au-dessus de la couche transport et relie les spécificités des couches inférieures des régions de sorte que les applications puissent communiquer à travers de multiples régions (Figure 1.8). Cette couche permet de diffuser les bundles (appelés aussi messages) en entier entre les nœuds[22].

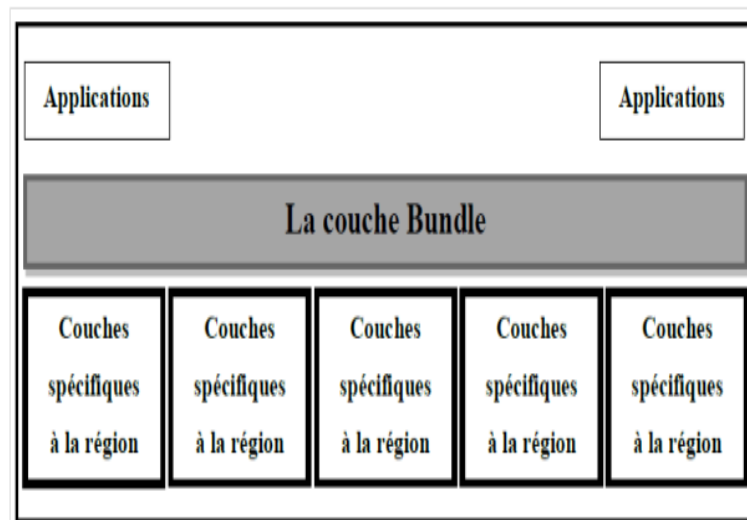


FIGURE 1.8 – Emplacement de la couche Bundle dans la pile protocolaire

La couche bundle est utilisée par tous les réseaux (ou les régions) qui constituent le DTN, tandis que les couches inférieures (de la couche transport à la couche physique) sont choisies selon l'environnement de chaque région (Figure 1.9)[22].

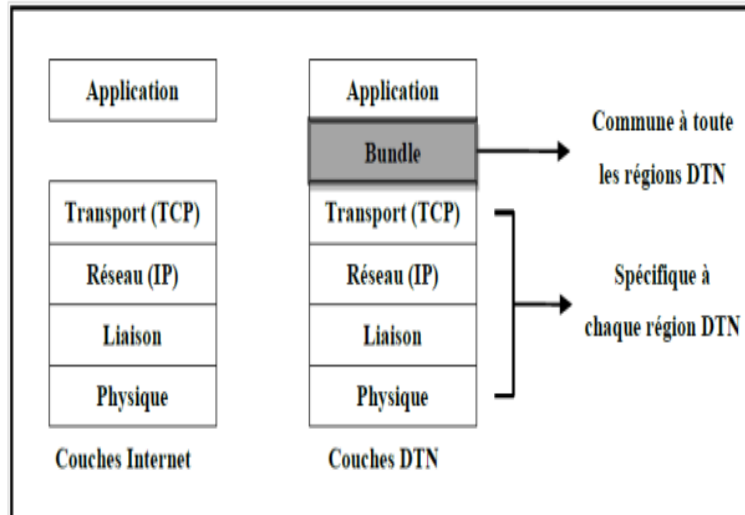


FIGURE 1.9 – La couche Bundle présente dans toutes les régions

- **La couche de convergence** : La couche bundle assure l'interopérabilité entre les couches inférieures des différentes régions hétérogènes à interconnecter grâce à une sous couche appelée : Couche de convergence (CLA).

La présence d'une telle sous couche offre au " Protocole bundle " la particularité de pouvoir fonctionner dès à présent au-dessus de plusieurs protocoles. En effet, si le protocole du niveau transport est TCP, la couche Bundle va utiliser les services de la couche TCPCL . Et pareillement pour le protocole de transport UDP .

La Figure 1.10 ci-dessous, montre la structure d'un transmetteur de bundles, avec la position des sous couches de convergences de différents protocoles réseaux, par rapport aux autres couches de la pile protocolaire[8].

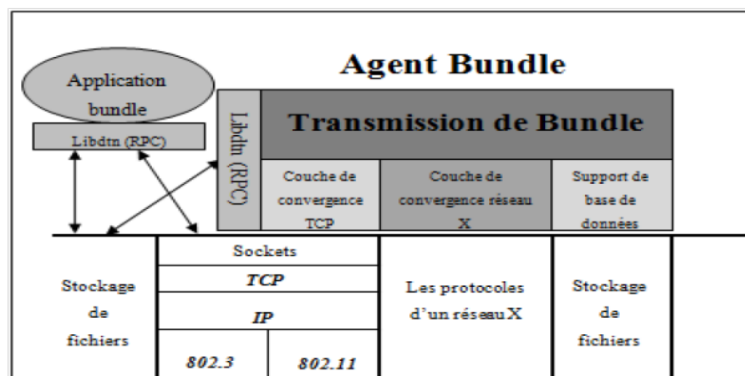


FIGURE 1.10 – Structure d'un transmetteur de bundle .

D'autres couches de convergence sont prévues d'être développées dans le futur, afin de fonctionner au-dessus des réseaux de capteurs, des réseaux à très longues distances ou encore des réseaux destinés aux environnements " tactiques ". Ainsi la très attendue couche

LTP se placerait comme remplaçant de TCP, dédiée aux communications interplanétaires. Elle serait considérée comme protocole de la couche Transport, jouant le rôle que joue le protocole bundle au niveau de la couche Application.

Cette variété permet d'utiliser la couche protocolaire sous-jacente la mieux adaptée aux caractéristiques de chacun des environnements de transmission, et surtout d'utiliser la couche bundle au-dessus de différents réseaux hétérogènes.

- **L'architecture DTN est basée sur la commutation de messages :** Un DTN transmet les bundles de la couche application quel que soit ce qu'ils contiennent comme requête à envoyer. Les bundles d'une application sont envoyés et délivrés dans un mode atomique, même s'ils peuvent être divisés durant la transmission.

Les messages transférés à travers la couche bundle, peuvent également contenir optionnellement un "reply-to-tuple" utilisé à la demande par des opérations de diagnostics spéciaux afin de diriger ceux-ci vers une entité autre que l'émetteur.

L'idée de la commutation de messages fournit un réseau avec des connaissances à priori sur la taille et les exigences de performances des transferts de données demandés. Lorsqu'il y'a une importante quantité de données dans la file d'attente, qui doit être transmise en priorité sur un chemin, l'avantage fourni par la connaissance de ces informations peut être significatif pour la prise de décisions dans la planification de messages à envoyer.

- **Le transfert de garde :** Les réseaux DTN supportent la retransmission nœud à nœud des données perdues ou corrompues au niveau des deux couches "transport" et "bundle". Cependant, vu qu'il n'y a pas une seule couche "transport" qui fonctionne de bout en bout dans les réseaux DTN, alors la fiabilité de bout en bout peut être mise en œuvre sur la couche bundle.

La couche "Bundle" supporte la retransmission nœud à nœud, par le moyen de : Transfert de garde qui consiste en la persistance d'un message au niveau d'un nœud appelé : gardien. Nous pouvons voir à travers la Figure 1.11 que les transferts sont disposés entre les couches bundle des nœuds successifs à la requête initiale de l'application source, appliquant le processus suivant : Lorsque le gardien de la couche bundle courante envoie un paquet au nœud suivant, il demande un transfert de garde et déclenche un temporisateur de retransmission de l'acquittement. Si la couche Bundle du nœud suivant accepte la garde, elle retourne un acquittement à l'expéditeur. Si aucun acquittement n'est retourné avant que le temps fixé ne soit expiré, l'expéditeur retransmet le paquet. La valeur assignée au temporisateur de l'acquittement est soit distribuée aux différents nœuds avec une information de routage, soit calculée localement à la base d'une expérience de transmission passée sur un nœud particulier.

Le gardien doit sauvegarder le bundle jusqu'à [22] :

1. L'obtention de l'acceptation de la garde par un autre nœud ;
2. L'expiration de la durée de vie fixée au bundle qui est prévue beaucoup plus longue que la valeur fixée au temporisateur d'acquittement. Cependant, ce temps d'acquittement devrait être assez long afin de permettre une transmission fiable.

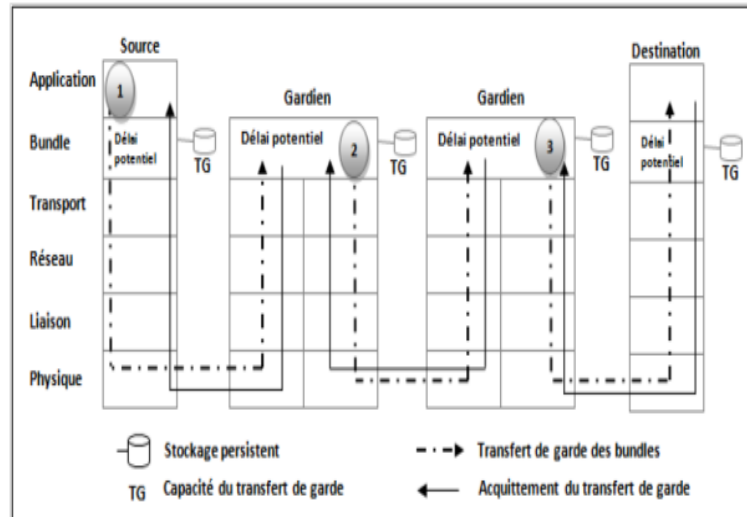


FIGURE 1.11 – Principe du transfert de garde (Custody Transfer)

Le transfert de garde permet à la source de déléguer la responsabilité de retransmission et de récupération de ses ressources relativement liées à la retransmission peu de temps après l'envoi du bundle (de l'ordre d'un aller-retour pour le premier saut du bundle). Chaque nœud du réseau DTN doit être en mesure d'être le gardien, car le service de garde est assuré.

- **Un protocole non conversationnel** : Sur des liens connectés de manière intermittente avec de longs délais, les protocoles conversationnels tels que TCP/IP qui comportent beaucoup de messages de négociation de bout en bout peuvent s'avérer impraticables ou échouer complètement à cause de la quantité de temps prise. Pour cette raison, dans les réseaux DTN, les couches bundles communiquent entre elles en utilisant de simples sessions avec un minimum d'aller-retour. N'importe quelle information du nœud de réception est optionnelle selon la classe de service choisie [22]. Tel qu'illustré dans la Figure 1.12, les protocoles des couches inférieures qui supportent la couche bundle peuvent être conversationnels comme TCP, mais dans les liens connectés d'une manière intermittente avec de longs délais, des protocoles non conversationnels ou peu conversationnels sont implémentés (parfois même au niveau des couches inférieures) [22].

- **Les bundles et l'encapsulation de bundles** : Les bundles sont composés de trois

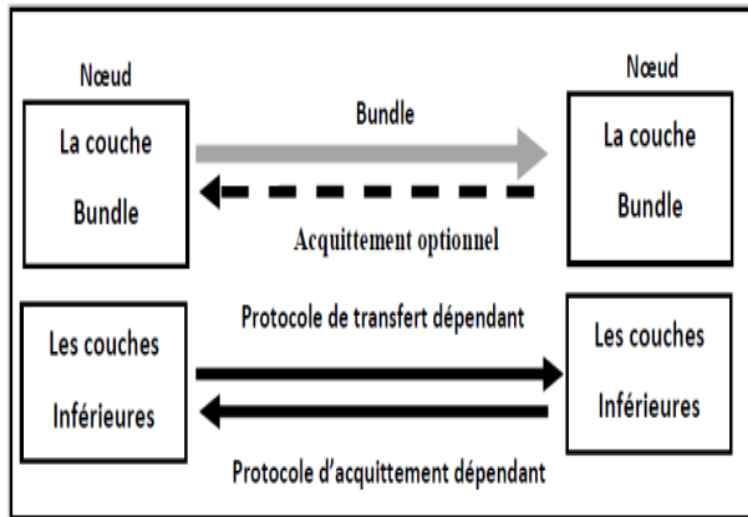


FIGURE 1.12 – Protocole non conversationnel

parties [22] :

1. Des données des utilisateurs de l'application " source ".
2. Des informations de contrôle, fournies par l'application " source " pour l'application " destination " décrivant comment traiter, stocker, se débarrasser et manipuler les données utilisateurs.
3. Un entête de bundle inséré par la couche bundle.

Comme les applications de données utilisateurs, les bundles peuvent être de tailles arbitraires. Ils prolongent la hiérarchie de l'encapsulation de données exécutée par le protocole utilisé dans Internet.

En effet, la couche bundle peut découper les paquets (ou les messages) en " fragments ", exactement comme la couche IP. Puis ces derniers sont réassemblés au niveau du nœud destinataire.

Il existe deux formes de fragmentation/assemblage pour les bundles [5] :

- Tout routeur DTN peut, de manière proactive, choisir de diviser le bloc de données en multiples blocs qui s'auto-identifient et transmet chaque bloc comme étant un bundle. Dans ce cas, les " destinations finales " sont responsables de l'assemblage des petits blocs qu'elles reçoivent en un seul bloc (l'original). Cette forme de fragmentation est analogue à la fragmentation IP.
- Le routeur de bundles peut, de manière réactive, choisir de fragmenter le bundle à la réception. Cette situation se présente lorsqu'une partie du bundle a été délivrée au saut suivant, et continu alors à envoyer de façon optimale le reste des portions du bundle original si des contacts ultérieurs seraient disponibles.

La fragmentation réactive est spécialement conçue pour traiter les cas dans lesquels le routeur est confronté à une transmission de bundles, mais qui n'a aucun contact qui fournit un volume de transfert de données suffisant.

1.4.2.4 Options de livraison

Dans les DTN, les applications peuvent avoir besoin d'utiliser ce que l'on appelle " Options de livraison " [5]. Celles-ci sont paramétrables à l'entête du bundle , nous présentons ci-dessous le rôle de chaque option, ainsi qu'un schéma (Figure 1.13) qui résume leur fonctionnement :

- **Transfert de garde** : La couche bundle transmet le paquet de données en utilisant les protocoles de transport les plus fiables (lorsqu'ils sont disponibles), et la responsabilité du point de livraison fiable (c'est-à-dire le buffer de transmission) permettra d'avancer d'un gardien à un autre jusqu'à ce que le bundle atteigne le point de destination. La couche bundle dépend de la couche transport du réseau qu'elle exploite afin de lui fournir le principal moyen pour un transfert fiable vers la couche suivante. Cependant, lorsqu'une livraison de garde est demandée, la couche bundle fournit d'une manière supplémentaire un timeout d'environ 500 ms, un mécanisme de retransmission, et un mécanisme d'acquiescement saut par saut qui l'accompagnent. Lorsqu'une couche bundle n'a pas besoin de livraison de garde, le timeout de cette couche et son mécanisme de retransmission ne sont pas employés, et les succès de la livraison des bundles à travers la couche bundle dépendent uniquement de la couche transport[13].
- **Retour-réception** : Le bundle de " retour-réception " est délivré par la couche bundle du récepteur lorsque le bundle est consommé par " l'application de destination ". Ce récépissé est fourni à l'entité spécifiée par le tuple source ou par une source désignée (réponse au champ) qui se trouve sur les différents hôtes.
- **Notification sur transmission du bundle** : Envoyée par le routeur du bundle lorsque le dernier fragment du bundle est transmis. L'indication est envoyée à la source de l'objet bundle.
- **Notification sur un transfert de garde** : Similaire à l'option de notification de transmission, mais envoyée lorsqu'un transfert de garde s'est achevé avec succès.
- **Livraison sécurisée** : Indique que l'application a fourni l'authentification du matériel avec le message envoyé. Afin d'opérer dans des circonstances générales, les applications doivent se préparer pour l'alimentation de l'authentification et la livraison sécurisée de la demande. La politique locale détermine si certains bundles peuvent être envoyés sans l'option de

sécurité, et les régions autres que la région d'origine peuvent exiger la sécurité même si la région d'origine ne l'est pas.

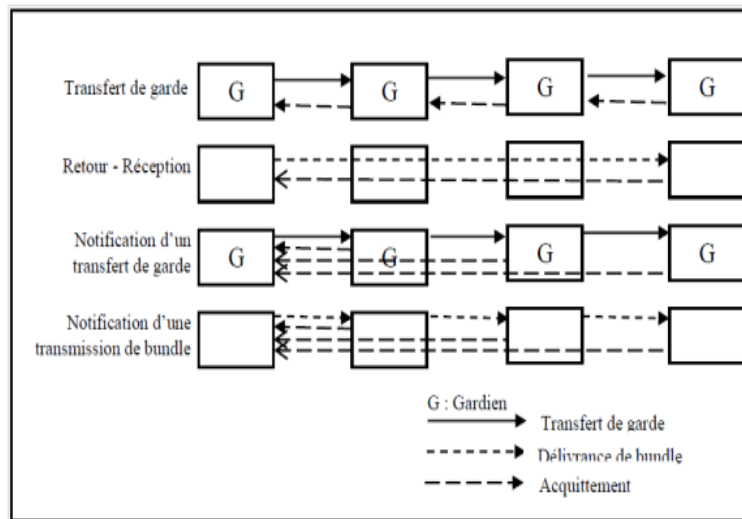


FIGURE 1.13 – Les options de livraisons offertes par la couche bundle.

1.4.2.5 Temps de synchronisation

L'architecture DTN dépend du temps de synchronisation (supporté par l'extérieur des protocoles locaux à la région) pour deux buts primaires :

- Le routage avec des contacts planifiés ou prédits et calcul du " temps de vie " du bundle.
- Routage basé sur les temps et dépendant des coordinations de partage des ressources (telles que les antennes directionnelles), ce qui oblige le temps de synchronisation à atteindre le RDV du contact.

Les calculs de temps de vie sont réalisés par l'inclusion d'une source d'estampillage et un temps explicite du domaine de vie (en unités de temps après le moment précisé dans la source d'estampillage). Sa seule utilisation est de débarrasser des données du réseau de sorte que les exigences de synchronisation posées ne soient pas strictes. Cette approche permet à la source d'estampillage d'être utilisée pour divers buts comme source particulière. Les nœuds DTN doivent veiller à ce que les estampilles de bundles qu'ils envoient ne diminuent jamais.

Les applications spécifient un temps d'expiration (la durée de vie est exprimée en secondes) pour les bundles qu'elles envoient. S'il n'est pas fourni, ou si la valeur fournie par l'utilisateur est plus grande que la politique locale permise, la couche bundle va alors en fournir une autre et c'est celle-ci qui sera considérée comme actuelle " durée de vie ", elle est ajoutée au temps

où le bundle a été soumis par l'application afin de déterminer le temps auquel le bundle sera débarrassé du réseau. Les valeurs appropriées dépendent du réseau et des données, et pourraient varier considérablement en théorie (de quelques millisecondes à plusieurs semaines). [8]

1.4.2.6 La congestion et le contrôle de congestion

Les performances d'un réseau se dégradent lorsqu'une congestion se produit. Celle-ci est causée par une lourde charge de trafic qui engendre une perte de données, due à l'épuisement des tampons au niveau des routeurs, ou aux longs délais dans la transmission de données.

Le problème de contrôle de congestion a été profondément traité dans l'Internet, mais les solutions proposées ne s'appliquent pas dans les environnements intermittents tels que, les réseaux tolérants aux délais. Car une connectivité continue et de bout en bout n'est pas garantie, et la latence est élevée. Ainsi, le contrôle de congestion se déroule localement au niveau de chaque routeur, qui lui, de manière autonome prend la décision d'accepter ou pas le bundle en se basant sur des informations locales. Du coup, grâce au contrôle de congestion, les routeurs sont protégés de l'épuisement des ressources sans qu'il y ait un taux élevé de perte de données [3].

- **Qu'est-ce que la congestion dans les DTN ?** : Dans les réseaux Internet, nous parlons de " congestion " lorsque nous constatons une croissance rapide et instantanée dans l'occupation de l'espace de stockage. Contrairement à cela, dans les DTN, ce qui signifie " congestion " est la croissance constante et continue dans l'occupation de l'espace mémoire [3]. Ceci se manifeste par le manque d'espace de stockage libre sur les nœuds intermédiaires, en raison d'une utilisation élevée d'une route donnée [15].
- **Le contrôle de congestion** : Le contrôle de congestion est le moyen de garantir que l'ensemble des taux de trafic que toutes les sources injectent dans le réseau, ne dépasse pas le taux maximum global que le réseau peut délivrer aux différentes destinations dans le temps. Ceci permet de minimiser la perte de données dans le réseau, en raison des limitations des espaces buffers dans les routeurs. Afin de limiter le taux de trafic au niveau des sources, un contrôle de flux est imposé [5].
- **Le contrôle de flux** : Le contrôle de flux garantit le fait que le taux moyen des données transmises par les nœuds sources ne soit pas supérieur au taux moyen des données que les nœuds récepteurs ont prévu de recevoir [5].
- **Le contrôle de flux et le contrôle de congestion dans l'Internet** : Le contrôle de flux dans l'Internet est géré de bout en bout. TCP détecte au niveau de la destination la croissance de l'occupation de son espace buffer. Il répond en réduisant le taux d'accusés

de réception qui seront détectés par TCP à la source, ce qui le pousse à réduire le taux de transmissions [19]. Ceci impose un contrôle de flux dans l'application source, qui soulage indirectement le taux de croissance excessif dans l'occupation des espaces buffers au niveau de la destination.

Quant au contrôle de congestion, lui est géré au niveau des routeurs, ce qui signifie que le routeur transmet des données plus lentement que le taux de leurs arrivées. Pour cela, deux solutions sont disponibles [3] :

a) **Explicite** : Le routeur envoie un paquet ICMP à la source, qui se traduit par un taux de transmission TCP réduit.

b) **Implicite** : Le routeur rejette les datagrammes, ce qui entraîne l'absence d'acquittement TCP et le force à la source de détecter un taux d'accusé de réception réduit. Du coup, ceci provoque à nouveau un contrôle de flux au niveau de l'application source.

- **Le contrôle de flux et le contrôle de congestion dans les DTN** : Dans les réseaux DTN, le contrôle de flux est généralement exercé par le protocole bundle. Ce dernier impose un contrôle de flux à l'application source afin de limiter son taux de transmission à celui du protocole bundle lui-même. Pour la gestion de la congestion dans les routeurs DTN, elle peut être traitée directement par un contrôle de flux au niveau de l'application source, car il ya une garantie sur la présence d'une connectivité de bout en bout dans l'une des routes.

L'une des solutions existante est son traitement en passant par les fonctions de transfert de garde offertes par le protocole bundle afin d'assurer un contrôle de flux indirectement. Ceci se fait selon la procédure suivante [3] :

1. Lorsque le routeur n'a plus d'espace de stockage, il rejette les bundles qui continuent à arriver ;
2. Les notifications sur un transfert de garde ne seront donc pas envoyées pour les bundles rejetés ;
3. L'absence de la notification provoque une congestion au niveau du gardien (un routeur en amont), causant éventuellement un rejet de bundle aussi ;
4. Cette propagation de rejet des bundles atteint finalement les nœuds sources, ce qui déclenche un contrôle du taux de transmission au niveau du protocole bundle et donc un contrôle de flux.

Les décisions de contrôle de flux dans les DTN s'effectuent au niveau de la couche bundle

elle-même, en se basant sur les informations sur les ressources disponibles dans le nœud du bundle. Lorsque les ressources de stockage deviennent insuffisantes, la couche bundle ne dispose que d'un certain degré de liberté dans la gestion de la situation, elle peut [5] :

1. Ecarter les bundles qui ont expiré, ce qui est une activité DTN qui doit s'exécuter dans n'importe quel cas ;
2. Faire en sorte que la garde du bundle soit cessée dans le cas où les nœuds étaient disposés à le faire ;
3. S'en servir des ressources de stockage disponibles dans le réseau ;
4. Ecarter les bundles qui n'ont pas expiré, mais pour lesquels la garde n'a pas été acceptée .

1.5 Conclusion

Dans ce premier chapitre, nous avons introduit les notions de DTN, montré le contexte de leur apparition et expliqué leur fonctionnement où nous avons évoqué le concept de Store and Forward et le transfert de garde. Aussi, nous avons détaillé l'architecture de ce type de réseaux qui est caractérisée par la présence d'une couche supplémentaire appelée : couche Bundle, au-dessus de la couche transport.

Dans tout type de réseau la transmission des paquets a une grande valeur, ce que nécessite un routage dont il existe plusieurs types. Dans le prochain chapitre nous allons énumérer ainsi que spécifier et détailler le routage dans les réseaux DTN.

2

Routage dans les réseaux DTN

2.1 Introduction

Généralement, le routage est une méthode d'acheminement des informations à la bonne destination à travers un réseau de connexion donné.

Les protocoles de routage dans les réseaux DTN ne cherchent pas à trouver le chemin le plus court tel que c'est le cas dans les protocoles classiques, mais ils jouent plutôt sur l'augmentation du taux de délivrance des données. En outre, les messages dans les réseaux DTN ne sont pas simplement routés à travers le réseau, ils peuvent être stockés temporairement sur certains hôtes, pour être réémis plus tard lorsque les circonstances le permettent.[\[19\]](#)

Nous présentons brièvement dans ce chapitre les types de routage existant, ainsi que les concepts de routage dans les réseaux DTN dont nous allons détailler la présentation.

2.2 Types de routage

2.2.1 Routage statique

Pour lequel la collecte des informations n'est pas nécessaire. Les systèmes à routage statique ne nécessitent pas des informations sur l'état du réseau car les routes ne sont pas recalculées ni changées une fois le système est mis en marche. Quand il y a un changement dans l'état du

réseau (agrandissement du réseau, ajout d'équipements, modification du trafic), le système de routage peut être modifié. Ces changements sont faits, normalement, par intervention directe de l'opérateur dans les équipements. Ce type de routage est normalement conçu lors de l'étape de dimensionnement des réseaux. [19]

2.2.2 Routage dynamique

Dans ce type de routage la route est calculée à la demande pour chaque nouvelle destination. Dans les réseaux DTN, le routage dynamique ne calcule les routes que pour les nœuds actuellement connectés. Parce que les liens ne sont pas toujours disponibles, les messages envoyés entre les routeurs auront du mal à circuler et l'algorithme ne pourra pas converger. Le choix des chemins s'adapte plus ou moins rapidement à des pannes réseaux et machines. L'adaptation à la charge des routeurs est très délicate et non implémentée [19].

2.3 Concepts de base du routage DTN

2.3.1 Routage proactif

Le routage proactif, est le type de routage adopté par la plupart des protocoles Internet standards, et quelques protocoles Ad Hoc tels que DSDV et OLSR . Son principe consiste dans le fait que les routes soient calculées automatiquement, et indépendamment du trafic arrivant. Dans les réseaux DTN, les protocoles répondant à ce type de routage sont capables de calculer les routes de toute la topologie d'un sous-réseau connecté. Néanmoins, ils échouent lorsqu'ils sont appelés à déterminer un chemin vers un nœud qui n'est plus accessible. Malgré cet inconvénient, les protocoles proactifs peuvent fournir des éléments utiles aux algorithmes de routage DTN en leur désignant l'ensemble des nœuds accessibles pour le choix du prochain saut.[2]

2.3.2 Routage réactif

Le routage réactif, est utilisé par certains protocoles de réseaux Ad Hoc tels que AODV et DSR. Les routes, dans ce cas, sont découvertes à la demande lorsque le trafic doit être délivré à une destination inconnue. Pour les réseaux DTN, comme avec les protocoles proactifs, ces protocoles fonctionnent seulement sur un sous-réseau connecté de la topologie globale. Cependant, ils échouent pour de multiples raisons par rapport aux protocoles proactifs, en particulier, lorsqu'une route n'est pas réussie par manque de réponse. Alors que les protocoles proactifs peuvent constater l'échec plus rapidement en s'apercevant que la destination demandée n'est

pas accessible à un instant donné.

Dans les réseaux DTN, les routes peuvent varier avec le temps lorsqu'il s'agit de chemins prévisibles, et peuvent être pré-calculées en utilisant des connaissances sur les futures topologies. L'emploi d'une approche proactive aurait alors probablement impliqué plusieurs séries de calcul de routes, et des index selon le temps. Ainsi, les besoins communs en ressources seraient extrêmes, sauf dans le cas où le trafic demandé est élevé et un grand pourcentage d'échanges de trafic est possible entre les nœuds du réseau. Sinon une approche réactive serait plus ravissante.[2]

2.3.3 Routage source

Le routage source permet de déterminer le chemin complet que doit suivre le message, depuis le nœud source. Ce chemin est codé dans le paquet du message, et il est déterminé une fois et ne change pas lorsque le message traverse le réseau.

2.3.4 Routage par saut

Dans le routage par saut, le prochain nœud du message est déterminé à chaque saut tout le long du chemin. Cette technique de routage permet au message d'utiliser l'information sur les contacts disponibles et les files d'attente à chaque saut. Ce qui est généralement indisponible à la source. Ainsi le routage par saut peut conduire à des meilleures performances. Cependant, en raison de sa nature locale, il peut conduire à des boucles lorsque les nœuds ont différentes vues de topologie (ex : dues à une information de routage incomplète ou retardée).[2]

2.3.5 Routage hiérarchique

Le routage hiérarchique exige que la source connaisse l'adresse hiérarchique de la destination. Si seul l'ID de la destination est disponible, alors la source peut recourir à utiliser le service de localisation.

Le routage hiérarchique est un routage saut par saut. Avant toute opération de routage, chaque nœud du réseau a besoin d'obtenir les informations sur les clusters de tous les niveaux de la topologie. Il effectue sa décision de transmission selon les étapes suivantes [14] :

1. Trouver le plus bas niveau k où la source S et la destination D ont un cluster commun ;

2. Définir une source intermédiaire S_0 et une destination intermédiaire D_0 qui sont des clusters de niveau k de S et D respectivement ;
3. Utiliser l'algorithme optimal de Dijkstra pour trouver le prochain saut n_0 du plus court chemin de S_0 à D_0 en se basant sur l'information de S sur la topologie du niveau k ;
4. Si $k = 0$, S prend la décision de faire une transmission à n_0 . Sinon, retourner à l'étape 3 avec un nouveau $k = k-1$, un nœud D_0 qui sera la passerelle distante de S_0 à n_0 et le nouveau S_0 va être le nœud du niveau k (nouveau k) qui est soit S ou le cluster de S .

2.4 Le scénario de routage dans un DTN

En générale un scénario de routage dans un DTN se déroule en quatre étapes[10] :

- **Attendre une opportunité de transfert** : chaque nœud qui génère un message pour un nœud destinataire détient le message jusqu'à ce qu'il rentre en contact avec un nœud appartenant au chemin source-destination.
- **Echange d'entête de messages** : à la rencontre de deux nœuds, ils échangent les listes des messages qu'ils possèdent.
- **Appliquer l'algorithme de routage** : sélection des messages à envoyer selon l'algorithme implémenté.
- **Echange des contenus des messages** : pour chaque message sélectionné dans un nœud le contenu est transféré à l'autre nœud.

2.5 Propriétés des stratégies

Les stratégies de routage dans un DTN ont été classées selon deux propriétés[20] :

2.5.1 Réplication

Désigne comment la stratégie utilise des copies multiples d'un message, et comment elle choisit de faire ces copies.

Le réseau DTN peut s'appuyer sur des composants qui sont fiable ou imprévisible. Pour compenser cela, de nombreuses stratégies de routage ont été proposées pour faire des copies multiples de chaque message, afin d'augmenter les chances qu'au moins une copie sera livré, ou pour réduire la latence de livraison. L'intuition est que ayant plus de copies du message augmente la probabilité que l'un d'eux trouvera son chemin vers la destination, et diminue le

temps moyen pour être livrés.

2.5.2 Connaissance

Indique comment la stratégie utilise les informations sur l'état du réseau afin de prendre des décisions de routage, et aussi comment elle obtient cette information.

Certaines stratégies de routage nécessitent plus d'informations sur le réseau que d'autres. Ces stratégies utilisent des règles statiques qui sont configurées lorsque la stratégie est conçue, et chaque nœud obéit aux mêmes règles. Cela conduit à des implémentations simples qui nécessitent une configuration minimale et des messages de contrôle. L'inconvénient est que la stratégie ne peut pas s'adapter à différents réseaux ou conditions, de sorte qu'elle ne peut pas faire des décisions optimales.

2.6 Familles des stratégies

Les stratégies de routage dans le réseau DTN sont divisées en deux familles en se basant sur la propriété utilisée afin de trouver la destination.[\[20\]](#) Nous allons d'abord décrire chaque famille en général, puis donner des exemples dans chaque famille.

2.6.1 Stratégies Inondations

Dans cette Stratégies, on remet des copies multiples de chaque message à un ensemble de nœuds, appelés relais. Les relais stockent les messages jusqu'à ce qu'ils se connectent avec la destination. Les premiers travaux dans le domaine de routage dans les DTN entrent dans cette famille et beaucoup entre eux dater avant le terme "délai-tolérant" est devenu populaire.

Traditionnellement, ces stratégies ont été étudiées dans le contexte des réseaux mobiles ad hoc, où la mobilité aléatoire a une bonne chance de mettre la source en contact avec la destination. La réplication de message est ensuite utilisée pour augmenter la probabilité que le message soit livré. Les protocoles de base dans cette famille n'ont pas besoin de toute les information du réseau, mais les régimes les plus avancés utilisent une certaine connaissance pour améliorer les performances. [\[20\]](#)

- **Contact direct**

Dans cette stratégie, le nœud source attend jusqu'à ce qu'il rentre en contact avec le nœud destination pour expédier les données.

C'est le cas dégénéré, car l'ensemble des relais ne contient que la destination. C'est une stratégie simple et ne consomme pas de ressources. Cependant, elle ne fonctionne que si la source est en contact avec la destination ce qui est son inconvénient.

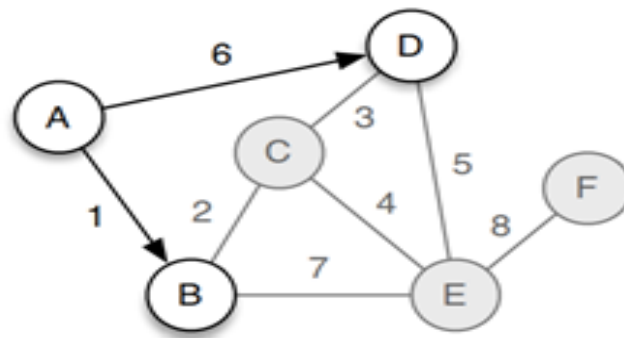


FIGURE 2.1 – Routage avec contact direct

- **Routage avec deux sauts**

Dans cette stratégie, le nœud source peut envoyer le message à tous les nœuds qui lui sont en contact direct. Et tous les nœuds infectés (nœuds relais) par le message ne peuvent envoyer ce dernier qu'au nœud destinataire. Cet algorithme ne consomme pas beaucoup de ressource.

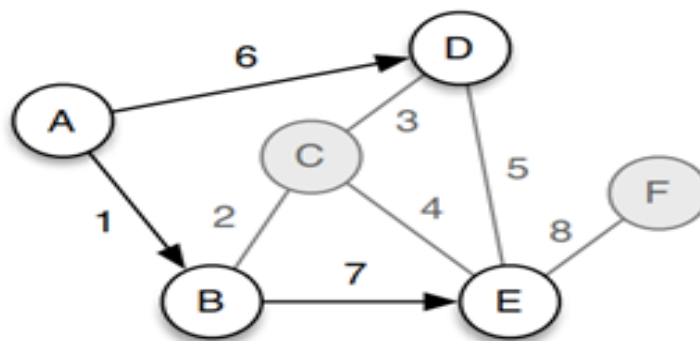


FIGURE 2.2 – Routage avec deux sauts

- **Inondation à base d'un arbre**

Dans cette stratégie, le nœud source génère une copie du message au niveau d'un relais, qui lui génère un autre nombre de copies au niveau des relais qui le suivent, d'où le nom "Tree-based", car l'ensemble des relais forme un arbre dont la racine est le nœud source [21].

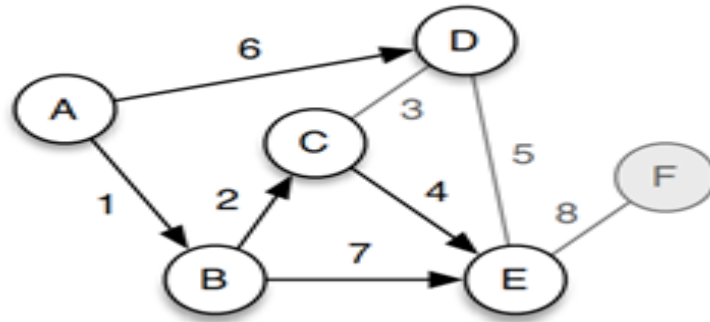


FIGURE 2.3 – Routage avec inondation à base d'un arbre

- **Routage épidémie**

Dans le routage épidémique, tous les nœuds vont éventuellement recevoir tous les messages, car l'algorithme fournit un nombre aléatoire d'échanges de données suffisant. Son principe de base est le suivant :

- Lorsqu'un message est envoyé, il est placé dans le buffer et est étiqueté par un unique ID.
- Quand deux nœuds se connectent, ils s'envoient chacun à l'autre la liste des IDs des messages qu'ils ont dans leurs buffers. Cette liste est appelée "vecteur d'état".

En utilisant ce vecteur d'état, les nœuds s'échangent les messages qu'ils ne possèdent pas, et à la fin de cette opération, tous les nœuds auront les mêmes messages dans leurs buffers.

En d'autres termes, le routage épidémique essaie d'envoyer chaque message sur tous les chemins du réseau. Ce qui fournit une grande quantité de redondances car tous les nœuds reçoivent tous les messages, mais qui rend cette stratégie extrêmement robuste. En outre, il essaie tous les chemins, il livre chaque message dans un temps minimum s'il y'a des ressources suffisantes.

Le routage épidémique est relativement simple. Car il ne nécessite aucune connaissance

sur le réseau. Pour cette raison, il a été proposé pour être utilisé comme repli si aucune autre meilleure méthode n'est disponible. L'inconvénient est qu'une énorme quantité de ressources est consommée et ce, est dû au grand nombre de copies, ce qui nécessite une grande quantité d'espace buffer, de bande passante et d'énergie. [1]

2.6.2 Stratégies d'expédition

Les stratégies de cette famille utilisent la topologie du réseau qui sélectionne le meilleur chemin et le message est alors transmis de nœud en nœud dans cette voie. Certaines de ces approches ont été explorées dans des réseaux câblés et les réseaux sans fil multi-hop.

Cependant, les protocoles conçus pour ces environnements ne fonctionnent pas dans les réseaux DTN, puisque ils supposent que les liens sont généralement connectés. Par définition, les stratégies de cette famille exigent une certaine connaissance de réseau. Ils envoient généralement un seul message le long de meilleur chemin, afin de ne pas utiliser la réplication.[21]

- **Routage basé sur la localisation**

Cette approche de transfert exige le moins d'information sur le réseau pour attribuer des coordonnées à chaque nœud. Une fonction de distance est utilisée pour estimer le coût de la prestation des messages d'un nœud à un autre. Les coordonnées peut avoir un sens physique, tels que les coordonnées GPS. En général, un message est transmis à un potentiel de saut suivant, si ce nœud est plus proche dans l'espace de coordonnées que la garde de courant.

L'avantage de routage basés sur la localisation est qu'elle nécessite très peu d'informations sur le réseau, éliminant ainsi le besoin pour les tables de routage et de réduire le surdébit de contrôle. En effet, ces techniques sont applicables à DTN.

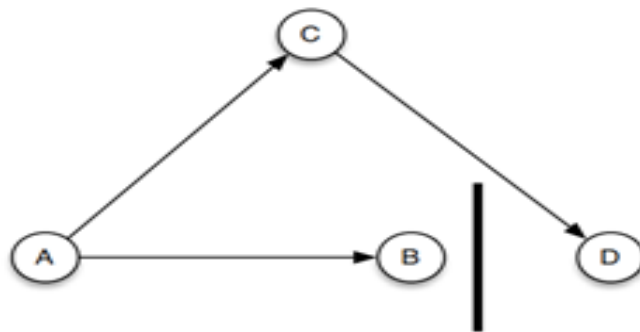


FIGURE 2.4 – Routage base sur la localisation

- **Dégradé de routage**

Une autre approche consiste à attribuer un poids à chaque nœud qui représente son aptitude à délivrer des messages à une destination donnée. Lorsque le dépositaire d'un message d'un autre contact avec un nœud qui a une métrique mieux pour la destination du message, le message sera transmis à travers ce lien. Cette approche est appelée gradient de routage parce que le message suit un gradient d'amélioration des valeurs de fonction d'utilité vers la destination.

Cela exige des connaissances plus que le réseau basé sur la localisation de routage pour deux raisons. Tout d'abord, chaque nœud doit stocker une métrique pour toutes les destinations possibles. Deuxièmement, des informations suffisantes doivent être propagées à travers le réseau pour permettre à chaque nœud de calculer sa métrique pour toutes les destinations. La métrique pourrait être basé sur de nombreux paramètres, tels que le temps du dernier contact entre le nœud et la destination, le reste d'énergie de la batterie, ou la mobilité. Un des défauts de gradient de routage est qu'il peut d'abord prendre un certain temps pour trouver un bon gardien, car il peut prendre un certain temps pour que les valeurs de la fonction d'utilité à propager, ou parce que les valeurs métriques dans la région autour de gardien initial sont tous également pauvres[21].

- **Métrique de lien**

Stratégie de routage qui utilise des métriques de liaisons des protocoles de routage traditionnels. Elle construit une topologie en graphe, qui attribue des pondérations à chaque lien et enfin exécute l'algorithme de plus court chemin pour trouver les meilleurs chemins. Cela nécessite plus d'informations sur le réseau dont chaque nœud doit disposer de suffisamment de connaissances pour exécuter un algorithme de routage. Les poids des liens sont affectés pour essayer de fournir un service optimal aux points de terminaison, basée sur une métrique de performance : la plus grande bande passante, la plus faible latence, et le ratio le plus élevé de livraison. En réseaux tolérant aux délais, le paramètre le plus important est le rapport de livraison, puisque le réseau doit être en mesure de fournir des données de manière fiable.

Une métrique secondaire est la latence de livraison. Ainsi, le défi est de déterminer un système d'attribution des métriques de liaison qui maximise le taux de livraison et minimise le temps de latence de livraison[21].

2.7 Quelques protocoles de routage pour les réseaux DTN

2.7.1 Disconnected Transitive Communication Protocol(DTCP)

Ce protocole repose sur le critère n'est pris en compte pour le choix des nœuds relais, le protocole DTCP introduit la notion d'utilité afin de juger les nœuds pertinents pour relayer les messages. Le calcul de cette probabilité repose sur certaines informations comme la liste des nœuds récemment rencontrés, la liste des nœuds fréquemment rencontrés et l'intervalle de redécouverte. [17]

2.7.2 Ressource Allocation Protocol for International DTN routing (RAPID)

La plupart des protocoles de routage qui ont été proposés avant RAPID ont relaxé une des contraintes suivantes :

- Capacité de stockage des nœuds DTN.
- Bande passante des contacts qui s'établissent entre les nœuds DTN.

Par contre, RAPID propose une stratégie de routage qui peut tenir compte des deux contraintes en même temps, dont il propose de dériver une utilité par message à partir d'une métrique de routage donnée afin de prendre par la suite des décisions tel que : quel message à supprimer en cas de congestion et quel message doit être répliqué en premier en cas d'insuffisance de la bande passante[17].

2.7.3 Protocole MaxProp

MaxProp est un protocole de routage connu pour les réseaux DTN et qui se base sur plusieurs mécanismes afin d'optimiser les deux métriques de routage, le taux et le délai de livraison des messages en présence de contrainte de bande passante et de stockage. Un manque de bande passante se traduit par l'établissement d'un contact de très courte durée entre deux nœuds DTN, un contact qui ne permet pas de terminer tous les échanges prévus entre les deux nœuds. Afin de remédier à ce problème, MaxPop définit l'ordre selon lequel les messages seront transmis, cela en se référant aux priorités associées aux différents messages, sachant que la priorité de chaque message correspond au coût associé à sa destination. Le message qui a la plus haute priorité sera transmis en premier et en cas de congestion de l'unité de stockage du nœud DTN, le message qui a la plus petite priorité sera le premier à être supprimé. Chaque nœud MaxPop maintient une liste de nœuds classés selon les coûts qui leur sont associés.[12]

2.8 Conclusion

Les réseaux DTN sont une nouveauté très prometteuse dans la recherche dans les réseaux, qui offrent l'espoir de rapprocher les gens et des dispositifs qui jusque-là étaient soit incapables de communiquer, ou pourrait le faire qu'à un coût élevé.

Tout d'abord, pour atteindre un taux de livraison élevé avec des ressources à faibles consommation, des techniques hybrides qui s'appuient à la fois sur la topologie posée sur la connaissance et la réplication sera nécessaire.

Cela a été implicitement noté par plusieurs chercheurs dans le domaine, mais le défi consiste à déterminer le bon équilibre entre la redondance et la consommation des ressources, et de trouver des solutions gérables pour l'utilisation de la topologie du réseau. Deuxièmement, dans les cas où le volume des messages est faible, simple épidémie de routage fonctionne extrêmement bien.

3

Problématique et proposition

3.1 Introduction

Au cours des deux dernières décennies, le contrôle de congestion dans les réseaux informatiques a émergé comme un défi technologique et scientifique majeur. En particulier, et même pour des configurations très simples, le comportement dynamique des réseaux est à ce jour encore assez mal compris[3].

Dans ce chapitre, nous allons traiter quelques approches de flot max dans un réseau, en effet, nous allons les comparer, ainsi que, choisir une parmi ces dernières pour la développer d'une manière à assurer la résolution de problème de congestion dans les réseaux DTN.

3.2 Problème de congestion

La congestion d'un réseau informatique est la condition dans laquelle une augmentation du trafic (flux) provoque un ralentissement global de celui-ci, les trames entrantes dans les buffers de commutateurs sont rejetées dans ce cas.

La congestion est liée à la politique du multiplexage établie sur le réseau considéré. En effet, on doit explorer le réseau pour trouver un bon chemin qui garantit la transmission du message de la source jusqu'à la destination.

Le problème de flot maximum consiste à trouver un flot réalisable depuis une source unique et vers un puits unique dans un graphe de flot qui soit maximum. Quelque fois le problème répond simplement à la question de trouver la valeur de ce flot. Le problème du flot maximum peut être vu comme un cas particulier de plusieurs autres problèmes de flots dans les réseaux. [7]

3.2.1 Les solutions existantes

Dans un nœud DTN, le problème d'allocation de la bande passante est considéré local, alors ce qui reste à résoudre est le problème d'allocation des buffers, car les paquets bundle pour lesquels la garde a été acceptée ne peuvent être jetés sans risque lorsque la mémoire est saturée. Les seuls mécanismes disponibles pour remédier à cela sont soit transmettre le message pour un autre gardien, soit supprimer le message si son temps d'expiration spécifié pour son application est atteint. D'autres messages (sans garde) peuvent être supprimés [9].

Pour alléger le problème au niveau de lien saturé, trois stratégies ont été utilisées :

1. La réservation d'un stockage permanent pour un transfert de message non gardés.
2. Store and forward en mémoire.
3. Le routage en mémoire "cut Through" (aux liens disponible seulement).

3.3 Définitions et propriétés :

Avant de commencer le traitement de problème on doit d'abord le structurer [6] :

- **Réseau** : Graphe orienté $G = (X, A)$, deux sommets particuliers s , et t . s n'a pas d'arc entrant et t n'a pas d'arc sortant.

Pour chaque arc a , un entier $c(a)$ appelé sa capacité.

Le graphe partiel obtenu à partir du graphe d'origine mais dont ont été retiré tous les arcs dont la capacité résiduelle est nulle, est appelé réseau résiduel.

- **Flot dans un réseau** : Un flot est une application F de A dans \mathbb{N} telle que :

1. Pour chaque arc a $F(a) < c(a)$.
2. Pour chaque sommet autre que s et t la somme des flots entrants est égale à celle des flots sortants.
3. La valeur du flot est égale à la somme des flots sortants de s .

Un flot saturé est un flot tel que sur tout chemin de s à t il existe un arc ayant un flot égal à sa capacité.

- **Coupes** : Pour un réseau $G = (X, A, s, t)$, une coupe est donnée par une partition de X en deux sous-ensembles disjoints $X = Y \cup Z$ tels que $s \in Y$ et $t \in Z$. La capacité de la coupe est égale à la somme des capacités des arcs qui ont une origine dans Y et une extrémité dans Z .

Pour un flot F et une coupe Y, Z , on note :

$$\mathcal{F}(Y, Z) = \sum_{\text{or}(u) \in Y, \text{ext}(u) \in Z} \mathcal{F}(u)$$

De plus on introduit :

$$\Delta(Y, Z) = \mathcal{F}(Y, Z) - \mathcal{F}(Z, Y)$$

Avec cette notation la valeur du flot à maximiser est :

$$V_{\mathcal{F}} = \mathcal{F}(\{s\}, X \setminus \{s\})$$

- **Capacité des arcs** : Un réseau de transport est un réseau où à chaque arc $j \in A$ est associé une capacité $C_j \geq 0$ (et éventuellement un coût d_j) (Exemple de capacité : tonnage ou débit maximum). C'est la limite supérieure du flux admissible sur j . Un flot est admissible seulement si : $F_j \leq C_j \forall j=1, \dots, m \Leftrightarrow F \leq C$.

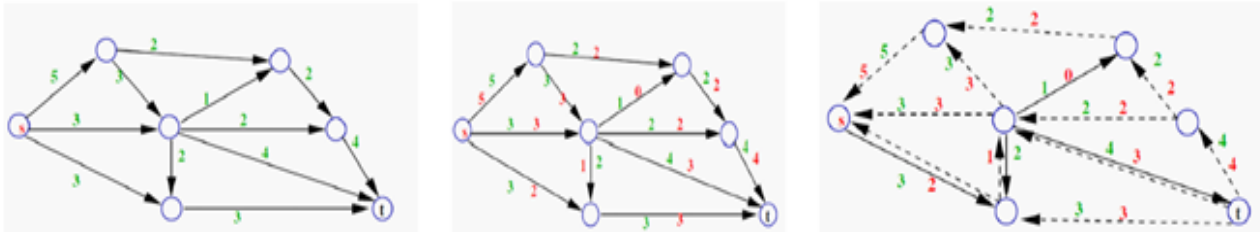
Pour un arc j donné, représente la quantité de flot pouvant encore passer par cet arc qu'on appelle aussi la capacité résiduelle tel que : capacité résiduelle de j = capacité j - flot j .

- **Graphe des augmentations** : On associe à un réseau de transport $G = (X, s, t, A, c)$ et à un flot F sur G un graphe G_F donné par :

1. les sommets de G_F sont ceux de G .
2. Pour tout arc $a = (x, y)$ de G on construit un arc entre x et y dans G_F si $F(a) < c(a)$.

3. Pour tout arc $a = (x, y)$ de G on construit un arc entre y et x dans G_F si $F(a) > 0$.

Exemples :



Un graphe et ses capacités

Un flot sur ce graphe

Le graphe des augmentations

FIGURE 3.1 – Contraintes de graphe

3.4 Problème du flot maximum dans un réseau DTN

Le problème du flot maximum dans un réseau, Il s'agit d'un problème d'optimisation classique dans le domaine de la recherche opérationnelle.

Ce problème d'optimisation peut être représenté par un graphe comportant une entrée (à gauche) et une sortie (à droite). Le flot représente la circulation de l'entrée vers la sortie d'où l'utilisation d'un algorithme dans les problèmes de réseaux. Les applications sont multiples : problèmes informatiques, routiers, etc. Il s'applique également à tous les autres problèmes de transferts comme les importations/exportations, les flux migratoires, démographiques mais aussi sur les flux plus abstraits tels que les transferts financiers.

Pour les données de très grande taille, il existe plusieurs algorithmes plus performants pour résoudre le même problème connu sous le nom de problème de flot maximum[6].

3.5 Quelques algorithmes de flot max

3.5.1 Algorithme de Ford et Fulkerson

L'algorithme de Ford-Fulkerson, du nom de ses auteurs L.R Ford et D.R Fulkerson, consiste en une procédure itérative qui permet de déterminer un flot (ou flux) de valeur maximale à partir d'un flot constaté. Il s'agit donc d'un problème d'optimisation classique dans le domaine

de la recherche opérationnelle. A chaque itération, la solution courante est un flot qui satisfait les contraintes de capacité (c'est donc un flot réalisable) et l'algorithme essaie d'augmenter la valeur de ce flot. [6]

- **Calcul du flot maximum** : Il suffit de considérer le cas d'une source et un puits basé sur l'idée d'une chaîne augmentante : une suite de changements possibles $\pm\delta$ dans une chaîne d'arcs dont le résultat est de transférer quantité δ de la source à un autre sommet.
- **La coupe minimum** : Quand l'algorithme ne trouve plus de chaîne augmentante au puits, il a trouvé un ensemble E de sommets autour de la source (ceux pour lesquels il existe une chaîne augmentante) tel que tous les arcs partant de E sont saturés (et tous ceux entrant en E ont flux nul) ; dans le flot maximum trouvé le flux total sur ces arcs partant de E est le même que le flot total trouvé ; donc l'algorithme a trouvé une coupe avec cette valeur.

- Algorithme :

```

Début
(1) Déterminer un flot compatible F (par exemple  $F(a) = 0$  pour tout a)
(2) Construire un réseau  $R^*(F)$  comme suit :
    -  $R^*(F)$  a exactement les mêmes sommets que R;
Pour tout arc  $a=(x,y)$  dans R faire :
    Si  $c(a) > F(a)$  alors
        Créer dans  $R^*(F)$  un arc  $(x,y)$  de capacité  $c^*(x,y) = c(a) - F(a)$ 
    Finsi ;
    Si  $F(a) > 0$  alors
        Créer dans  $R^*(F)$  un arc  $(y,x)$  de capacité  $c^*(y,x) = F(a)$ .
    Finsi ;
Finpour ;
(3) Si l n'existe pas de chemin de s à t dans  $R^*(F)$  alors
    STOP : le flot f est maximum
Sinon,
    Soit P un tel chemin et soit  $\varepsilon = \min_{(x,y) \in P} c^*(x,y)$ 
Finsi ;
Pour tout arc  $(x,y)$  dans P faire
    Si l'arc  $(x,y)$  existe dans R alors
        - Augmenter le flot de  $\varepsilon$  unités sur  $(x,y)$  ;
    Finsi ;
    si l'arc  $(y,x)$  existe dans R alors
        - Diminuer le flot de  $\varepsilon$  unités sur  $(y,x)$  ;
    Finsi ;
    Retourner à (2).
Finpour ;
Fin,

```

A l'étape (3), lorsqu'il n'existe plus de chemin de s à t dans $R^*(F)$, on peut déterminer

l'ensemble W des sommets x pour lesquels il existe un chemin de s à x . On a donc $s \in W$ et $t \notin W$. La coupe $(W, V-W)$ est de capacité minimale. [6]

3.5.2 L'algorithme d'Edmonds et Karp

L'algorithme de Edmonds-Karp construit un flot maximal en partant d'un flot nul $\forall a \in A, f(a)=0$ et en augmentant son débit le long de chemin $s_i=s_1, s_2, \dots, s_n, s_{n-1}=t$ de s à t acceptables, c'est à dire tels que $\forall i, (s_i, s_{i+1}) \in A$ et que le débit maximum $\max_i c(s_i, s_{i+1})$ soit strictement positif.

La particularité de l'algorithme d'Edmonds-Karp est de toujours choisir un chemin acceptable dont la longueur (en nombre d'arêtes) est minimale. Dès qu'un nouveau chemin L acceptable de longueur minimale est trouvé, son débit est ajouté à f , et parallèlement, les capacités c de G sont mises à jour. La découverte d'un tel chemin se fait par un parcours en largeur qui utilise une file Q des nœuds à traiter et un tableau $p : s \rightarrow s \cup -1$ où $p(u)$ vaut -1 si u n'a pas encore été traité, et l'avant dernier sommet dans un chemin de longueur minimale de s à u sinon. [6]

Edmonds-Karp()

Début

```

 $\forall (u, v) \in A, f(u, v) = 0$ 
  Répéter
    fin=parcours ()
    si fin=1 alors
      retourner f,
    finsi,
    m =  $\infty$ 
    u = t,
    tant que u  $\neq$  s faire
      m = min(m, c(p[u], u)) ;
      u = p[u] ;
    fin tant que ;
    u = t
    tant que u  $\neq$  s faire
      f(p[u], u) = f(p[u], u) + m
      c(p[u], u) = c(p[u], u) - m
      c(u, p[u]) = c(u, p[u]) + m
      u = p[u]
    fin tant que
  fin répéter
fin.

```



```

Parcours
Début
   $\forall u \in S, p[u] = -1$ 
   $Q = [s]$ 
  tant que  $Q \neq \emptyset$  faire
     $u = \text{dépiler } Q$ 
    pour chaque  $v$  tel que  $(u, v) \in A, c(u, v) > 0$  et  $p[v] = -1$  faire
       $p[v] = u,$ 
      Si  $v = t,$  alors
        retourner 1 ;
      fin si ;
      ajouter  $v$  _a la fin de  $Q$ 
    fin pour
  fin tant que
  retourner 0 ;
Fin.

```

3.5.3 L'algorithme de Floyd-Warshall

L'algorithme de Floyd-Warshall est un algorithme pour déterminer tous les plus courts chemins dans un graphe orienté et value, en temps cubique.

L'algorithme de Floyd-Warshall prend en entrée un graphe orienté et value (V, E) , sous la forme d'une matrice d'adjacence donnant le poids d'un arc lorsqu'il existe et la valeur ∞ sinon. Le poids d'un chemin entre deux sommets est la somme des poids sur les arcs constituant ce chemin. Les arcs du graphe peuvent avoir des poids négatifs, mais le graphe ne doit pas posséder de cycle de poids strictement négatif. L'algorithme calcule, pour chaque paire de sommets, le poids minimal parmi tous les chemins entre ces deux sommets.

Soit $V = 1, 2, 3, 4, \dots, n$ l'ensemble des sommets de G et soient i et j deux sommets de V . On considère un chemin p entre i et j de poids minimal dont les sommets intermédiaires sont dans $1, 2, 3, \dots, k$. L'algorithme de Floyd-Warshall est basé sur l'observation suivante :

- soit p n'emprunte pas le sommet k .
- soit p emprunte exactement une fois le sommet k (car les cycles sont de poids positifs ou nuls) et p est donc la concaténation de deux chemins, entre i et k et k et j respectivement, dont les sommets intermédiaires sont dans $1, 2, 3, \dots, k-1$.

Notons W^k la matrice telle que W_{ij}^k est le poids minimal d'un chemin de i à j n'empruntant que des sommets intermédiaires dans $1, 2, 3, \dots, k$, s'il en existe un, et ∞ sinon W^0 est la

matrice définissant G .

L'observation ci-dessus se traduit par l'égalité , pour tous i, j et k dans $1, 2, 3, 4, \dots, n$, en supposant les opérations \min et $+$ convenablement adaptées au cas d'un opérande ∞ . Dès lors, on peut calculer successivement les matrices pour $k=1, 2, 3, 4, \dots, n$. Le calcul peut même être effectué en place, dans une unique matrice .

Le pseudo-code suivant effectue ce calcul[7] :

Procédure FloydWarshall (G : matrice $n \times n$)

$W := G$

pour $k:=1$ à n

 pour $i:=1$ à n

 pour $j:=1$ à n

$W_{ij} := \min(W_{ij}, W_{ik} + W_{kj})$

 fin pour;

 fin pour;

fin pour;

3.5.4 Table de comparaison

Critères	Etapas	Flot de départ	Stratégie	Complexité
Algorithmes				
Ford-Fulkerson	-Le marquage -Le changement de flux(déterminer un chemin augmentant).	Nul	Inondation	$O(nm^2)$ Dépend des capacités sur les arcs, le nombre de sommets et d'arcs.
Edmonds, et Karp	-Déterminer un chemin augmentant -Mettre à jour la capacité du chemin acceptable et mettre à jour le débit.	Nul	Recherche en arrière en largeur d'abord	$O(m^2n)$ (pour un graphe de n sommets et m arêtes)
Floyd-Warshall	-Emprunter les sommets de i à j . -Récupérer le poids minimal de chemin -Mettre à jour la matrice	Poids d'arc ou nul	Inondation	$O(n^3)$ (pour un graphe de n sommets)

TABLE 3.1 – Tableau comparatif des algorithmes de flot max.

3.6 Choix d'algorithme

Et comme proposition pour notre projet nous prenons l'algorithme de Ford Fulkerson comme approche que nous allons exploiter. Notre choix sur cette approche est dû aux raisons suivantes :

- La simplicité en résolution.
- C'est la source de la majorité des autres approches.
- La complexité qui est linéaire $O(n M^2)$.

Lorsque nous prenons un exemple de réseau muni de capacité pour chaque arc (chaque lien entre deux nœuds), pour une bonne circulation des paquets (messages) de la source vers le puits, nous ajoutons un autre critère pour éviter le problème de congestion au niveau d'un lien : le critère c'est le flot d'un arc.

A l'exploitation de l'algorithme lorsque on trouve un chemin conflictuel on cherche un autre chemin qui garantit la transmission des messages jusqu'à la destination (chaîne augmentante) et test si le lien est disponible ou pas au départ dans chaque étape de l'exécution dont le flot d'un arc ne dépasse pas sa capacité .

3.7 La solution proposée

L_{xy} : est un arc entre deux sommets.

Si ce lien est disponible (connecté) on met 1 sinon on met 0 .

s : c'est la source de réseau R (graphe),

t : c'est le puits de réseau R .

F : est un flot initial réalisable, initialisation à 0, $f(x,y)$ un flot pour chaque arc (x,y) .

c : pour chaque arc (x,y) appartient à un réseau R est associé une capacité $c(x,y)$.

a : l'arc (x,y) .

P : le chemin entre s et t ; ε : minimum des capacités pour chaque arc de P.

Fonction état-lien (x,y)

Début

L_{xy} : booléen

Si lien connecté alors

$L_{xy}=1$;

Sinon

$L_{xy}=0$;

Finsi

Return L_{xy} ;

Fin ;

Algorithme :

```

Début
(1) Déterminer un flot compatible F (par exemple  $F(a) = 0$  pour tout a)
(2) Construire un réseau  $R^*(F)$  comme suit :
    -  $R^*(F)$  a exactement les mêmes sommets que R;
    L = état-lien(x,y);
Si (L=1) alors
    Pour tout arc a=(x,y) dans R faire :
        Si  $c(a) > F(a)$  alors
            Créer dans  $R^*(F)$  un arc (x,y) de capacité  $c^*(x,y) = c(a) - F(a)$ 
        Finsi ;
        Si  $F(a) > 0$  alors
            Créer dans  $R^*(F)$  un arc (y,x) de capacité  $c^*(y,x) = F(a)$ .
        Finsi ;
    Finpour ;
Finsi ;
(3) Si l n'existe pas de chemin de s à t dans  $R^*(F)$  alors
    STOP : le flot f est maximum
Sinon,
    Soit P un tel chemin et soit  $\varepsilon = \min_{(x,y) \in P} c^*(x,y)$ 
Finsi ;
Pour tout arc (x,y) dans P faire
    Si l'arc (x,y) existe dans R alors
        - Augmenter le flot de  $\varepsilon$  unités sur (x,y);
    Finsi ;
    si l'arc (y,x) existe dans R alors
        - Diminuer le flot de  $\varepsilon$  unités sur (y,x);
    Finsi ;
    Retourner à (2).
Finpour ;
Fin,

```

S : la source.

P : le puit.

$[c, f]$:

c : la capacité pour chaque arc.

f : le flot de pour chaque arc.

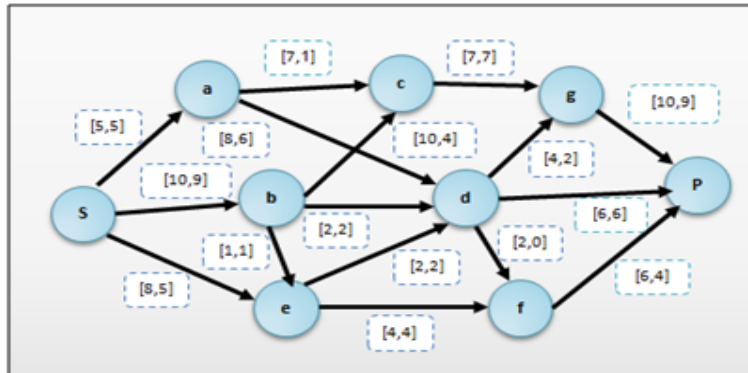


FIGURE 3.2 – Exemple de flot max.

-On marque la source par (+)

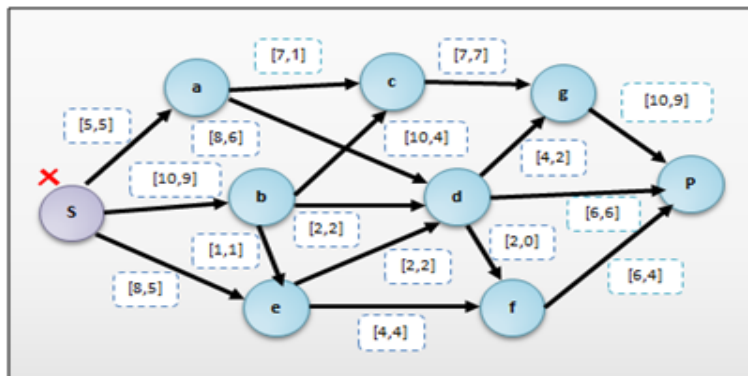


FIGURE 3.3 – Marquage de la source S

- $Y = \{S\}$, $C^- = C^+ = \emptyset$

(x,y) : x marqué et y non marqué et $f(x,y) < c(x,y) \rightarrow$ marquer y par $(+x)$

- $Y = \{S,b\}$, $C^+ = \{(S,b)\}$, $\delta_b = \min \{d_S, c(x,y) - f(x,y)\} = \{\infty, 10-9=1\}=1$

- $Y = \{S, b, e\}$, $C^+ = \{(S,b);(S,e)\}$, $\delta_e = \min \{ \delta_e, c(x,y) - f(x,y)\} = \{\infty, 8-5\}=3$

- $Y = \{S,b,e,c\}$, $C^+ = \{(S,b), (S,e), (b,c)\}$, $\delta_c = \min \{1, 8-6\}=1$

- $Y = \{S,b,e,c,a\}$, $C^+ = \{(S,b), (S,e), (b,c)\}$, $C^- = \{(a,c)\}$, $\delta_c = \min \{ \delta_c=1, 7-1=6\}=1$

- $Y = \{S,b,e,c,a,d\}$, $C^+ = \{(S,b), (S,e), (b,c), (a,d)\}$, $C^- = \{(a,c)\}$, $\delta_d = \min \{1, 10-4\}=1$

- $Y = \{S, b, e, c, a, d, f\}$, $C^+ = \{(S, b), (S, c), (b, c), (a, d), (d, f)\}$, $C^- = \{(a, c)\}$, $\delta_f = \min\{\delta_d = 1, 2-0\} = 1$
- $Y = \{S, b, e, c, a, d, f, g\}$, $C^+ = \{S, b, e, c, a, d, f, g\}$, $C^+ = \{(S, b), (S, e), (b, c), (a, d), (d, f), (d, g)\}$, $C^- = \{(c, a)\}$, $\delta_g = \min\{1, 4-2\} = 1$
- $Y = \{S, b, e, c, a, d, f, g, P\}$, $C^+ = \{(S, b), (S, e), (b, c), (a, d), (d, g), (d, f), (g, P)\}$, $C^- = \{(a, c)\}$, $\delta_P = \min\{1, 1\} = 1$
- Jusqu'à Y contient tous les sommets marqués

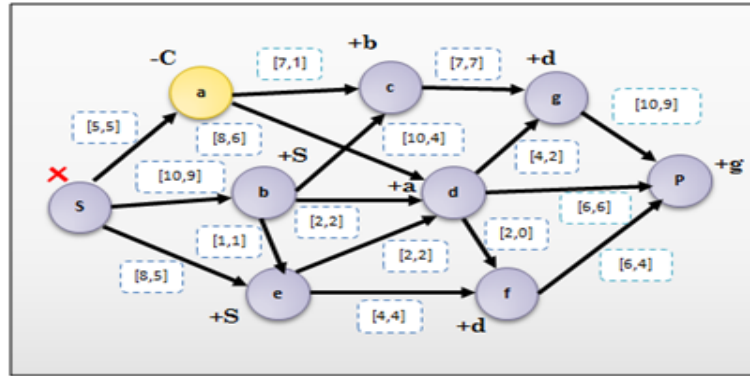


FIGURE 3.4 – Marquage de tous les sommets

Le chemin de S vers P correspond tous les arcs $\{(S, b), (b, c), (c, a), (a, d), (d, g), (g, P)\}$.

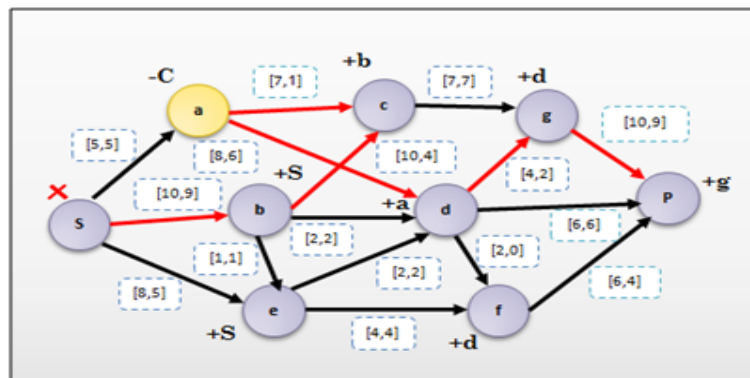


FIGURE 3.5 – La chaîne augmentante

On a trouvé une chaîne augmentante qu'on peut reconstruire puisque chaque sommet connaît son prédécesseur grâce au marquage.

La chaîne augmentante est $:\{(S, b), (b, c), (c, a), (a, d), (d, g), (g, P)\}$, $\varepsilon = 1$

On peut ajouter 1 ou retrancher 1 dans arc de chemin pour améliorer le flot par 1.

Le flot courant est maximum alors $V_{max} = 20$.

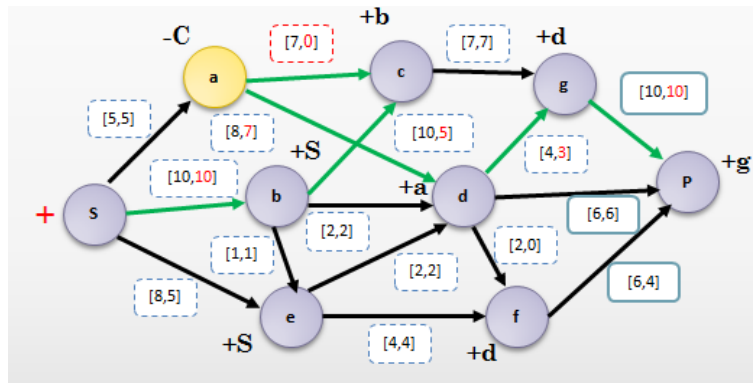


FIGURE 3.6 – Le flot amélioré

3.8 Conclusion

Pour obtenir un service de transmission optimale dans un réseau DTN, d'autres fonctions doivent être ajoutées ou étudiées, telles que le comportement vis-à-vis d'autres flux ou l'équité entre flots (à l'intérieur du protocole). Ceci passe également par une étude de mécanismes plus fins de transfert saut-par-saut, et évaluer également le bénéfice en cas de déconnexions.

En conclusion, fournir un service de transmission fiable pour le futur réseau DTN représente un défi important. Nous avons illustré dans ce chapitre quelques algorithmes pour la recherche de flot max dont nous avons choisi un et nous l'avons modifier d'une manière à s'adapté avec le domaine étudiier qui est le réseau DTN.

Conclusion générale

Fournir une communication fiable dans les futurs réseaux en particulier les réseaux DTN est un véritable challenge. En effet, un réseau DTN est un environnement hostile qui apporte plusieurs défis, dus à des caractéristiques et spécificités propre à ce genre de technologie. En outre, les délais variables, les longues périodes de déconnexion induisent le problème de congestion qu'il faut prendre en considération lors de la conception d'un réseau DTN.

Dans ce mémoire nous nous sommes intéressés au problème de congestion dans le routage au niveau des nœuds dans un réseau DTN, dont la première partie nous avons présentés quelques généralités sur les réseaux DTN telles que l'architecture de recouvrement basée sur l'ajout de la couche Bundle qui se base sur le principe de stockage et de transmission (Store and Forward) pour construire un système bien approprié et englobant les différents réseaux hétérogènes.

En effet, pour notre proposition nous avons illustré dans ce mémoire quelques algorithmes pour la recherche de flot max dont nous avons choisi l'algorithme de Ford-Fulkerson nous l'avons modifier d'une manière à s'adapté avec le domaine étudiier qui est le réseau DTN.

Nous proposons alors comme perspectives d'amélioration :

- Implémenter notre proposition.
- Analyser et évaluer notre approche de routage.
- Pouvoir situer notre stratégie entre toutes celles qui existent.
- Elargir le domaine de la proposition.

Bibliographie

- [1] A. Benmessaoud. Classification des protocoles de routage dans les réseaux tolérants aux délais (dtn). *ESI*, 2009.
- [2] J. Broch, D A. Maltz, D B. Johnson, Y C. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. *In ACM Mobicom*, 1998.
- [3] S. Burleigh, E. Jennings, and J. Schoolcraft. Autonomous congestion control in delay-tolerant networks. *American Institute of Aeronautics and astronautics*, 2006.
- [4] S. Burleigh, A. Hooke, L. Torgerson, and K. Fall. Delay tolerant networking an approach to interplanetary internet. *IEEE Communications Magazine*, 41(6) :128–136, 2003.
- [5] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss. Delay-tolerant network architecture. 2003.
- [6] R. Cori. Conception et analyse d’algorithmes. *Université Bordeaux, France*, 2005.
- [7] T H. Cormen, C E. Leiserson, R L. Rivest, and C. Stein. Introduction to algorithms. *Cambridge, MIT Press and McGraw-Hill*, 2001.
- [8] K. Fall. A delay tolerant network architecture for challenged internets. *Intel Research Berkley*, 2003.
- [9] K. Fall, W. Hong, and S. Madden. Custody transfer for reliable delivery in delay tolerant networks. 2007.
- [10] B. Gallagher, D. Jensen, and B N. Levine. Explaining routing performance in disruption tolerant networks. 2006.
- [11] J P. Gelas. Réseaux interplanétaires (ipn) et réseaux tolérant aux délais (dtn). *Laboratoire de l’Informatique du Parallélisme*, 2006.
- [12] D. Jensen, B N. Levine, J. Burgess, and B. Gallagher. Maxprop : Routing for véhicule-based distription-tolérant network. *IEEE INFOCOM*, 2006.
- [13] A. Krifa. An optimal joint scheduling and drop policy for delay tolerant networks. *PLANETE-INRIA*, 2008.
- [14] C. L and J. Wu. Scalable routing in delay tolerant networks. *MobiHoc’07*, 2007.

- [15] P. Mundur, S. Lee, and M. Seligman. Routing in delay tolerant networks using storage domains. *UMD Tech Report UMIACS-TR-2007-01*, 2006.
- [16] S. Paillard. Architectures réseaux pour communications interplanétaire. *TELECOM PARIS*, 2006.
- [17] M. Samir. Architecture d'hybridation des réseaux de capteurs avec les réseaux satelistes. *Mémoire de Mastère, Université de Béjaia*, 2009.
- [18] H C. Sanchez. Routage store and forward dans les constellations de satellites. *TELECOM ParisTech*, 2008.
- [19] M. Steenstrup. Routing in communications networks. *Prentice Hall International (UK) Ltd., Hertfordshire, UK, UK*, 1995.
- [20] P A S. Ward and E P C. Jones. Routing strategies for delay tolerant network. *Proceedings of ACM SIGCOMM*, 2006.
- [21] P A S. Ward, E P C. Jones, and L. Li. Practical routing in delay-tolerant network. 2005.
- [22] F. Warthman. Delay tolerant network (dtn). *A tutorial. Forest Warthman. Warthman Associates, Version 1.1*, 2003.
- [23] D. Wick. Delay tolerant networking. *RVS Group*, 2007.