

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université A. MIRA - Béjaïa -
Faculté des Sciences Exactes
Département d'informatique

Mémoire de Master

*En vue de l'obtention du diplôme de Master en Informatique
Spécialité : Administration et sécurité des réseaux informatique*

Thème

Le routage intermittent dans les réseaux Tolérant aux délais (DTN)

Présenté par :

M^{elle} HADID Ourdia

M^{elle} BEN ABDALLAH Fadila

Devant le jury composé de :

Président M^r SALHI Nadi

Examineur M^r BAADACHE Abderrahmane

Encadreur M^r TOUAZI Djoudi

2012-2013

Remerciements

Nous tenons à remercier tout d'abord le bon dieu qui nous a donnée du courage et de la patience pour aller jusqu'au bout de ce modeste travail.

*Nous tenons à remercier notre encadreur Mr Touazi, enseignant au niveau du département informatique de l'université de Bejaia pour nous
Avoir fait honneur de diriger notre travail.*

*Nous tenons aussi à remercier les membres de jury, Monsieur S.A.L.H.F
Nadir pour nous avoir fait l'honneur de présider notre travail.*

*Enfin, nous remercions vivement, Mademoiselle B.A.A.D.A.C.H.E
Abderrahmane pour avoir accepté d'examiner notre Travail.*

Dédicaces

Je dédie ce travail à mes chers grands parents

A mes chers parents

A mon frère et sa fiancée

A mes chères sœurs

A toute mes amies

Fadila

Dédicaces

À la mémoire de mon père

À ma mère

À mes frères

Ourdia

TABLE DES MATIERES

Table des figures	i
Liste des tableaux.....	ii
Liste des algorithmes.....	iii
Chapitre 1	Erreur ! Signet non défini.
Introduction générale.....	4
Généralité sur internet	6
Introduction.....	6
Les caractéristiques de l'Internet.....	6
Les limites d'Internet	6
Fiabilité du protocole TCP.....	7
Protocole de routage	7
Le modèle TCP/IP	8
Présentation de TCP/IP	8
Caractéristiques de TCP/IP	8
L'établissement de la connexion	9
Conclusion	11
Rappel sur les graphes	11
Introduction.....	11
Graphe non orienté	12
Graphe orienté.....	12
Graphe pondéré	13
Les algorithmes de cheminement	13
Algorithme de Dijkstra	13
Algorithme de Bellman-Ford	14
Comparaison entre Bellman-Ford et Dijkstra.....	14
Conclusion	15
Le routage dans l'Internet.....	15
Introduction.....	15
Les concepts de base.....	15
Définition du routeur	15
Définition de routage	16
Définition de la table de routage.....	16
Le contenu de la table de routage	16
Type de routage	17
Les algorithmes de routage dynamique	17

Les algorithmes à vecteur de distance (Vector Distance)	17
Les algorithmes de routage à état de liens (Link-State)	18
Les protocoles de routage	19
Le protocole RIP (Routing Information Protocol).....	19
Le protocole OSPF (Open Shortest Path First)	20
Le protocole EGP (Exterior Gateway Protocol).....	21
Les limites d'EGP.....	21
Le protocole BGP (Border Gateway Protocol).....	21
Conclusion	22
Chapitre 2	23
Introduction.....	23
Définition.....	23
Caractéristique des réseaux DTNs	23
Connectivité intermittente.....	24
Délais longs et variables	24
Vitesse de transmission asymétrique.....	24
Taux d'erreur important	24
Les aspects fondamentaux des réseaux DTN.....	25
Protocole non conversationnel	25
La technique store and forward.....	26
Architecture des réseaux DTNs	27
La couche bundle.....	27
Les messages bundle	28
Les entités de communication.....	29
Les nœuds	29
Les régions DTN	30
Nommage et adressage dans un réseau DTN.....	31
Différence entre le cheminement Internet et DTN	32
Fiabilité dans les réseaux DTNs.....	33
Le transfert de garde.....	33
Issue de la couche transport dans un DTN.....	34
Fiabilité saut par saut	34
L'accusé actif	34
L'accusé passif	35
L'accusé à passerelle réseau.....	35
Sécurité dans les réseaux DTNs	35

La congestion et le contrôle de congestion	36
Conclusion	38
Chapitre 3	40
Introduction.....	40
Les étapes de routage dans un DTN	40
Quelques définitions.....	40
Le temps de transfère de message	41
Les ressources du réseau.....	41
Concept de base de routage.....	42
Routage proactif	42
Routage réactif	42
Routage source :	43
Routage par saut :	43
Routage hiérarchique :	43
Routage par la réplication :	44
Routage par la connaissance :	45
Classification des protocoles de routage DTN:.....	45
L'inondation:.....	46
L'expédition:	46
Les oracles de connaissance:.....	46
Oracle de l'état des contacts:.....	47
Oracle de contact:	47
Oracle de file d'attente:.....	47
Oracle de demande de trafic:.....	47
Les connaissances partielles:.....	48
Les connaissances complètes:	48
La classification de Jain & Al:	48
Les algorithmes de routage	49
Algorithme avec des couts invariables:	49
Algorithme Minimum Expected Delay(MED).....	49
Algorithme avec des couts variables	50
Algorithme de livraison le plus rapide (Earliest Delivery Algorithm).....	50
Algorithme Earliest Delivery with Local Querying(EDLQ).....	51
Algorithme Earliest Delivery with All Queues(EDAQ)	52
Conclusion	53
Chapitre 4	54

Introduction.....	54
Le modèle DTN	54
Algorithme	55
Principe de l'algorithme.....	55
Les Pseudos algorithmes.....	57
Exemple de l'application	59
Conclusion	66
Résumé	Erreur ! Signet non défini.
Conclusion générale	68
Perspectives.....	69
Liste des abréviations	70
Références bibliographiques	72

TABLE DES FIGURES

Figure 1 : Établissement de la connexion.....	9
Figure 2 : Graphe non orienté.....	10
Figure 3 : Graphe orienté.....	11
Figure 4 : Principe du routeur.....	14
Figure 5 : Protocole non conversationnel.....	25
Figure 6 : La technique Store-and-Forward.....	25
Figure 7 : La couche bundle.....	27
Figure 8 : Les différents rôles d'un nœud DTN	28
Figure 9 : Transfert de garde.....	34
Figure10 : Diagramme résumant la classification de Jain et Al.....	50
Figure 11 : Les instants d'arrivées d'un message.....	57
Figure 12 : Arrivée de message avant le commencement de connexion.....	58
Figure 13 : Arrivée de message après le commencement de connexion.....	59
Figure 14 : Exemple de réseau de communication.....	64
Figure 15 : Réseau en cas d'existence de panne.....	70

LISTE DES TABLEAUX

Tableau 1: Matrice état des liens.....	68
Tableau 2 : Matrice état des liens en cas de panne.....	71

LISTE DES ALGORITHMES

Algorithme 1 : Arrivée de message avant le commencement de la connexion.....	60
Algorithme 2 : Arrivée de message après le temps commencement de première.....	60

INTRODUCTION GENERALE

Le réseau Internet a fait un grand succès en matière d'interconnexion des appareils de communication, grâce à l'utilisation d'un ensemble de protocoles homogènes appelés TCP/IP.

Des centaines de milliers de dispositifs qui composent le réseau Internet utilisent ces protocoles pour le routage de données et la garantie de la fiabilité des échanges de messages. Cette qualité de transmission provient du fait que TCP/IP offre plusieurs avantages. En effet, TCP est un protocole qui se base sur des chemins de bout en bout caractérisés par un bas débit de transmission et des délais très courts, au plus bas taux d'erreur grâce au mécanisme de retransmission en cas de perte de données et de reprise après erreur.

Les communications autres qu'Internet (mobile, satellite) sont réalisées sur des liaisons indépendantes. Chacune ayant ses propres caractéristiques, ont des délais de transmissions qui s'allongent, lorsque les nœuds sont placés de façon très éparse. De manière générale, elles ne sont pas compatibles avec Internet, entre elle non plus.

Pour ces raisons que le DTNRG a pensé à une solution qui supporterait de tels environnements de communication et qui prendrait en charge la variation des délais quelles que soient leurs valeurs. Pour cela, il a proposé une nouvelle architecture de réseaux appelée les réseaux tolérants aux délais (ou DTN : Delay Tolerant Network).

Le concept DTN été initialement développé pour des réseaux formés d'éléments spatiaux. La principale difficulté abordée par ce concept réside dans les longs délais et les déconnexions inhérentes. Dans un tel contexte, les déconnexions sont le plus souvent liées au mouvement des planètes et aux phénomènes de masquage induits. La mécanique spatiale qui régit ces mouvements étant parfaitement prévisible, les déconnexions dues à ces phénomènes le sont également. Les lois de la mécanique spatiale permettent aussi d'aller plus loin dans ce raisonnement et de calculer de manière précise les instants et durées de connexion entre satellite et relais terrestres en prenant en compte différents paramètres dont l'élévation minimale par exemple. La fonction de routage dans un réseau consiste à trouver une route à suivre pour aller d'une source à une destination en passant par des éléments intermédiaires du réseau. Tout l'intérêt et la complexité de cet aspect

consiste à trouver une route qui sera capable d'optimiser une ou plusieurs métriques particulières. Comme par exemple, le nombre de sauts, le délai d'acheminement ou le temps moyen d'attente dans les nœuds intermédiaires. Le calcul des routes dans un réseau est un élément fondamental pour le bon fonctionnement global du réseau ainsi que ses performances.

L'étude du routage dans un réseau a toujours été d'une grande importance, car il permet de définir le chemin formé par les nœuds successifs par où peuvent transiter les données avant d'arriver à destination. Une telle étude dans un environnement intermittent qui, n'est régi par aucune règle, l'est encore plus, car l'objectif dans ce cas, n'est plus de trouver le chemin qui minimise certaines métriques (ex : le plus court chemin), mais il s'agit de repérer celui qui maximise la probabilité de délivrer un message en prenant en considération l'état du réseau à tout moment. Pour cela, et dans le cadre de notre travail, nous allons présenter le routage intermittent dans les réseaux DTN.

Afin de bien expliquer toutes les notions de ce projet, nous avons divisé notre rapport en quatre chapitres.

Le premier chapitre a été consacré à la présentation générale du réseau internet et leur mode de fonctionnement.

Dans le second chapitre, nous présentons les réseaux tolérants aux délais et leur caractéristique. Nous abordons les aspects fondamentaux des réseaux DTN, la technique de store-and-forward, l'architecture des réseaux DTN, le nommage et adressage dans un réseau DTN et la sécurité dans les réseaux DTNs.

Dans l'avant dernier chapitre, nous nous focalisons sur le routage dans les réseaux DTN. Nous commençons par les étapes de routage dans un DTN, et les concepts de base de routage dans les réseaux tolérants aux délais pour finir avec une présentation des différentes classifications des protocoles de routage DTN et les algorithmes de routage dans ce type de réseau.

Finalement, nous concluons avec un chapitre sur la simulation de la communication dans un réseau informatique sous forme un graphe où les nœuds sont ses sommets et les arcs les liens qu'il existe entre eux. Les liens supportant le trafic sont intermittents, ils se déconnectent et se reconnectent périodiquement.

CHAPITRE 1**GENERALITE SUR INTERNET****INTRODUCTION**

L'Internet a été l'un des plus grands succès dans l'interconnexion des équipements d'échange d'informations à travers le globe terrestre. C'est un réseau simple, ouvert, flexible, où le seul service offert aux utilisateurs est le routage au mieux des paquets. Pour faciliter l'interconnexion des réseaux, la création d'applications et l'ajout de services à l'Internet, sont indispensables. La famille des protocoles TCP/IP permet aux différentes machines de se communiquer entre elles sur le réseau internet.

LES CARACTERISTIQUES DE L'INTERNET

L'Internet repose sur un ensemble de caractéristiques, qui dans le passé, convenaient à un très grand nombre d'applications. Ces caractéristiques incluent :

L'existence d'un chemin de bout en bout, de la source à la destination.

La disponibilité de la bande passante.

Les réseaux doivent rester le plus simple possible et que l'intelligence doit se trouver dans les machines des utilisateurs qui effectuent les opérations classiques de type recouvrement des erreurs.

Adaptation de débit.

Localisation des ressources.

LES LIMITES D'INTERNET

Les applications reposant sur la suite des protocoles Internet ne fonctionnent pas correctement lorsque les données emploient un chemin sur lequel la connexion entre certains nœuds est intermittente ou lorsque le délai de propagation du signal devient trop important. La principale cause est la fiabilité de son protocole de couche transport, mais

aussi, le mode de fonctionnement du protocole de routage d'Internet qui n'est pas toujours satisfaisant[.1]

FIABILITE DU PROTOCOLE TCP

Le protocole de transport le plus utilisé offert par la suite de protocole Internet est TCP. Ce dernier intervient au dessus de la couche IP. TCP utilise des accusés de réception mais ce protocole n'est pas adapté aux transmissions qui emploient des chemins caractérisés par des liens intermittents. TCP est un protocole dit conversationnel. En fait, afin d'établir une communication TCP, l'émetteur et le récepteur doivent négocier le flot de communication.

L'établissement d'une connexion TCP entraîne au moins un temps d'aller-retour avant que les données de l'application ne puissent être transmises. En plus TCP ne délivre les données à l'application réceptrice que dans l'ordre de transmission, par conséquent les données erronées nécessitant d'être retransmises vont retarder la livraison jusqu'à ce que ces dernières soient correctement transmises, au moins un temps d'aller-retour.

L'application ne peut éviter ce blocage à moins de supporter le coût de l'ouverture d'une connexion parallèle pour transmettre ses paquets. Pare encore, le débit de transmission de TCP diminue avec l'augmentation du temps d'aller-retour.

Employer TCP sur un chemin de bout en bout constitué de plusieurs liaisons dont certaines longues ou ayant des connexions intermittentes présente encore plus de difficultés, puisque TCP effectue des retransmissions de bout en bout, lesquelles retardent la libération de l'espace tampon [1].

PROTOCOLE DE ROUTAGE

Le système de routage d'Internet donne la possibilité aux routeurs de choisir le meilleur chemin par lequel ils vont transmettre les données. Au plus haut niveau, cette tâche est réalisée par BGP. BGP est un protocole de routage entre systèmes autonomes (SA). Les SAs, utilisent différents protocoles de routage, tel que OSPF qui employé pour faire

suivre les paquets. Ce protocole sélectionne le chemin à suivre par les données sur un réseau dont la topologie peut changer de façon imprévisible lorsque plusieurs chemins sont disponibles à un instant donné. Pour cette raison, les routeurs voisins doivent, à intervalles de temps réguliers, s'échanger des informations de routage.

Certains ont un délai d'expiration. Passé ce délai, et si un de ces voisins ne lui a pas envoyé des paquets de mise à jour, il suppose que la connexion avec ce voisin là est coupée.

BGP s'appuie sur TCP, et de ce fait, ses performances dans un environnement caractérisé par de longs délais sont limitées.

Le problème plus sérieux est dû au partitionnement transitoire des réseaux, dans lesquels les longs délais d'acheminement sont dus à des connectivités intermittentes, ou encore, au fait que les liens soient créés et supprimés sans cesse [1].

LE MODELE TCP/IP

PRESENTATION DE TCP/IP

TCP (Transmission Control Protocol) est l'un des principaux protocoles de la couche transport du modèle TCP/IP. Il permet, au niveau des applications, de gérer les données en provenance (ou à destination) de la couche inférieure du modèle (c'est-à-dire le protocole IP). Lorsque les données sont fournies au protocole IP, celui-ci les encapsule dans des datagrammes IP, en fixant le champ protocole à 6. Pour savoir que le protocole en amont est TCP. TCP/IP, devenu standard de fait, est actuellement la famille de protocoles réseaux qui gère le routage le plus répandu sur les systèmes UNIX et Windows, et surtout, c'est le protocole sur lequel l'Internet est basé.

CARACTERISTIQUES DE TCP/IP

Le succès de TCP/IP vient de ses caractéristiques intéressantes :

Ce protocole est indépendant du support physique du réseau. Cela permet à TCP/IP d'être véhiculé par des supports et des technologies différentes comme une ligne série ou un câble coaxial Ethernet.

Le mode d'adressage est commun à tous les utilisateurs de TCP/IP, quelle que soit la plate-forme qui l'utilise. Si l'unicité de l'adresse est respectée, les communications aboutissent même si les hôtes se trouvent chacun, sur l'autre bout du globe terrestre. Les protocoles de hauts niveaux sont standardisés, ce qui permet les développements largement répandus et interopérables sur tous types de machines.

Il existe une route de bout en bout, entre la source et la destination, pour toute la durée de la communication.

Le lien entre la source et la destination est bidirectionnel.

Le délai aller-retour (Round Trip Time ou RTT) est faible : inférieur à la seconde et il est souvent de l'ordre de plusieurs dizaines de millisecondes.

La conversation est un bon moyen de corriger les erreurs de transmissions, comme par exemple le mécanisme d'acquittement dans TCP.

Les pertes de bout en bout sont relativement faibles.

Tous les routeurs et les stations connaissent le protocole IP.

Des mécanismes de sécurité (intégrité, authentification), à l'émission et à la réception, sont suffisants, les nœuds intermédiaires ne sont pas impliqués.

La commutation de paquets est le meilleur moyen d'obtenir interopérabilité et performance.

Emprunter une seule route entre la source et la destination est suffisant en termes de fiabilité, car les mécanismes de répétition (ARQ) se chargent de la fiabilisation.

L'ETABLISSEMENT DE LA CONNEXION

Le processus de communication, qui se fait grâce à une émission de données et d'un accusé de réception, est basé sur un numéro d'ordre (appelé généralement numéro de séquence), il faut que les machines émettrices et réceptrices (client et serveur) connaissent le numéro d'ordre initial de l'autre machine.

L'établissement de la connexion entre deux applications se fait toujours comme indiqué dans la figure 1 :

- Les ports TCP doivent être ouverts

- L'application sur le serveur est passive, c'est-à-dire que l'application est à l'écoute, en attente d'une connexion
- L'application sur le client fait une requête de connexion sur le serveur dont l'application est en ouverture passive. L'application du client est dite en ouverture active.

Les deux machines doivent donc synchroniser leurs séquences grâce à un mécanisme communément appelé Three Ways Handshake (poignée de main en trois temps), que l'on retrouve aussi lors de la clôture de session.

Ce dialogue permet d'initier la communication, il se déroule en trois temps, comme sa dénomination l'indique :

- Dans un premier temps la machine émettrice (le client) transmet un segment dont le drapeau SYN est à 1 (pour signaler qu'il s'agit d'un segment de synchronisation), avec un numéro d'ordre N, qui s'appelle numéro d'ordre initial du client.
- Dans un second temps la machine réceptrice (le serveur) reçoit le segment initial provenant du client, puis lui envoie un accusé de réception, c'est-à-dire un segment dont le drapeau ACK est à 1 et le drapeau SYN est à 1 (car il s'agit là encore d'une synchronisation). Ce segment contient le numéro d'ordre de cette machine (du serveur) qui est le numéro d'ordre initial du client. Le champ le plus important de ce segment est le champ accusé de réception qui contient le numéro d'ordre initial du client, incrémenté de 1.
- Enfin, le client transmet au serveur un accusé de réception, c'est-à-dire un segment dont le drapeau ACK est à 1, dont le drapeau SYN est à zéro (il ne s'agit plus d'un segment de synchronisation). Son numéro d'ordre est incrémenté et le numéro d'accusé de réception représente le numéro d'ordre initial du serveur incrémenté de 1.

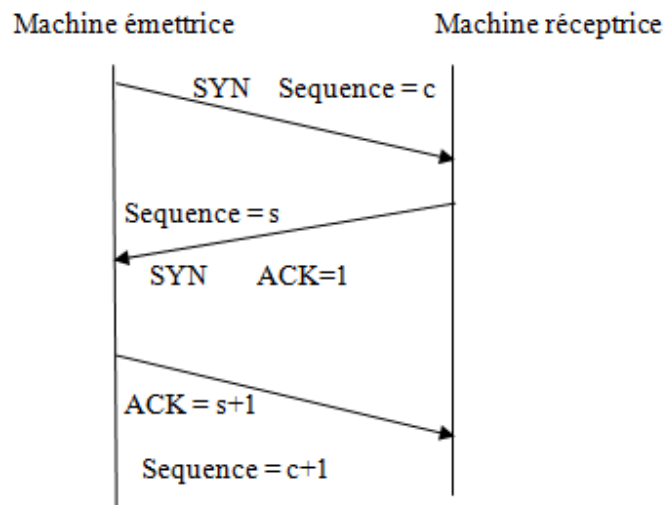


Figure 1: Etablissement de la connexion

CONCLUSION

L'Internet c'est un outil de communication efficace qui a créé de nouvelles formes de lien social. Il est le lieu où l'on pouvait trouver un grand nombre d'informations, mais plusieurs applications basées sur l'Internet ne fonctionnent malheureusement pas bien lorsque la fiabilité des interconnexions entre différents nœuds (routeurs) n'est pas garantie.

RAPPEL SUR LES GRAPHERS

INTRODUCTION

L'utilisation des graphes pour les réseaux commence, nécessairement, par une phase de modélisation dont l'importance est à rappeler. Avant de voir différents types de réseaux et comment les graphes interviennent dans leur modélisation, il est important de rappeler qu'un graphe est un objet, orienté ou non, qui représente une relation binaire entre des éléments d'un ensemble.

GRAPHE NON ORIENTE

Un graphe non orienté est une relation G symétrique sur un ensemble fini S . De la même manière que pour les graphes orientés, on le note $G = (S, A)$ où S est l'ensemble des sommets et / ou A est l'ensemble des arêtes.

Dans le cas d'une relation symétrique, on représente son graphe en remplaçant les doubles arcs par un simple trait.

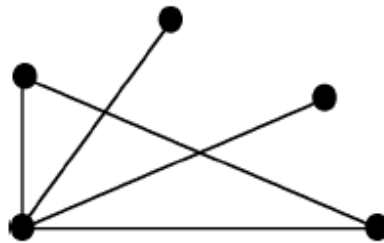


Figure 2: Graphe non orienté

GRAPHE ORIENTE

Un graphe orienté $G=(S, A)$ est la donnée d'un ensemble fini, non vide, de sommets S tel que $S = \{s_1, s_2, \dots\}$ et d'un ensemble de flèches (arcs) $A \subset S^2$, tel que $A = \{a_1, a_2, \dots\}$.

Si $a = (s_1, s_2) \in A$, on dit que a est la flèche (l'arc) de s_1 à s_2 . On note $a = s_1 \rightarrow s_2$. Si $s_1 = s_2$ on dit que a est la boucle en s . Les sommets s_1 et s_2 sont adjacents si $(s_1, s_2) \in A$.

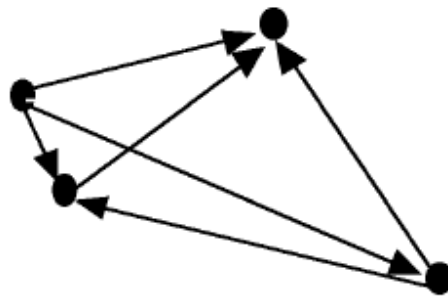


Figure 3: Graphe orienté

GRAPHE PONDERE

On dit qu'un graphe est pondéré si on a affecté à chaque arête un nombre positif (quel que soit sa signification).

- Ce nombre positif est alors appelé poids de l'arête.

LES ALGORITHMES DE CHEMINEMENT

Les algorithmes de calcul du plus courts chemins les plus utilisés par les protocoles de routage sont ceux de Bellman-Ford et de Dijkstra . Les protocoles de routage utilisant l'algorithme de Bellman-Ford sont dits de type Distance-Vector. Les protocoles de routage utilisant l'algorithme de Dijkstra sont dits de type Link-State.

ALGORITHME DE DIJKSTRA

À partir d'un graphe, l'algorithme de Dijkstra utilise le parcours en largeur d'abord pour trouver les chemins les plus courts entre une source et toutes les destinations du graphe. Puisqu'un graphe est composé de nœuds et d'arêtes, l'algorithme de Dijkstra peut trouver les chemins les plus courts liant un nœud quelconque à tous les autres nœuds du graphe en une seule exécution.

Pour illustrer l'intérêt de cet algorithme, on peut prendre par exemple le réseau routier d'une région : chaque sommet est une ville et chaque arc est une route dont le poids est le kilométrage. L'algorithme de Dijkstra permet alors de trouver le plus court chemin d'une ville à une autre. Cet algorithme est très largement utilisé, par exemple pour des applications de routage de données, de pathfinding dans des jeux vidéo et bien entendu de calcul d'itinéraires. L'application de l'algorithme de Dijkstra exige un graphe dont les caractéristiques qui sont :

Orienté ou non orienté.

Pondéré ou non pondéré.

Valeur de la pondération non négative (c'est à dire, ≥ 0).

Il doit exister au moins un chemin partant du nœud de départ vers tous les autres nœuds du graphe. Évidemment, cette condition ne s'applique pas pour le nœud de départ.

Il est donc important de s'assurer que les valeurs inscrites sur les arêtes des graphes soient non négatives.

ALGORITHME DE BELLMAN-FORD

L'Algorithme de Bellman-Ford est un algorithme de programmation dynamique qui permet de trouver les plus courts chemins, depuis un sommet source donné, dans un graphe orienté pondéré. Contrairement à l'algorithme de Dijkstra, qui ne peut être utilisé que lorsque tous les arcs ont des poids positifs ou nuls, l'algorithme de Bellman-Ford autorise la présence de certains arcs de poids négatif et permet de détecter l'existence d'un circuit absorbant, c'est-à-dire de poids total négatif, accessible depuis le sommet source.

Dans les réseaux informatiques, l'algorithme de Bellman-Ford est utilisé pour déterminer le cheminement des messages, à travers le protocole de routage RIP.

COMPARAISON ENTRE BELLMAN-FORD ET DIJKSTRA

Plusieurs critères doivent être pris en compte pour déterminer, entre Bellman-Ford et Dijkstra, lequel est le plus adapté aux protocoles de routage :

La fiabilité

La consommation de ressources sur les routeurs

La vitesse de convergence

L'algorithme de Dijkstra ne s'applique que dans le cas d'un graphe pour lequel le poids des arcs est non négatif; ce qui constitue la majorité des graphes (cartes routières, réseaux divers...). Pour les graphes dont le poids des arcs peut être négatif, on doit avoir recours à l'algorithme de Bellman-Ford.

L'algorithme de Bellman-Ford est moins rapide que l'algorithme de Dijkstra. Aussi, dans la majorité des cas (graphes dont le poids des arcs est non négatif), on utilisera Dijkstra.

CONCLUSION

Les algorithmes de recherche de plus court chemins sont très utilisés en informatique pour optimiser le routage entre les nœuds du réseau quelque soit sa nature. Chaque algorithme à ses points forts et ses points faibles et le choix de l'utilisation d'un algorithme dépend de ce qu'on veut optimiser comme performance.

LE ROUTAGE DANS L'INTERNET

INTRODUCTION

Le routage devient indispensable pour interconnecter un réseau au reste de l'internet, pour construire un grand réseau ou gérer des flux à l'intérieur du réseau. La difficulté de configuration d'un réseau routé n'est pas uniquement liée à l'administration du routeur proprement dit. Un plan d'adressage doit être conçu et géré par l'administrateur du réseau.

LES CONCEPTS DE BASE

DEFINITION DU ROUTEUR

Le routeur est un matériel d'interconnexion qui a accès à toutes les informations des couches 1, 2 et 3, notamment aux adresses logiques qui sont indépendantes de toute méthode d'accès et de toute topologie physique. Le rôle du routeur est également d'optimiser les transmissions entre réseaux ou sous-réseaux en déterminant la route la plus rapide pour acheminer les paquets à destination ainsi que le format le plus approprié en fractionnant éventuellement le paquet source [2].

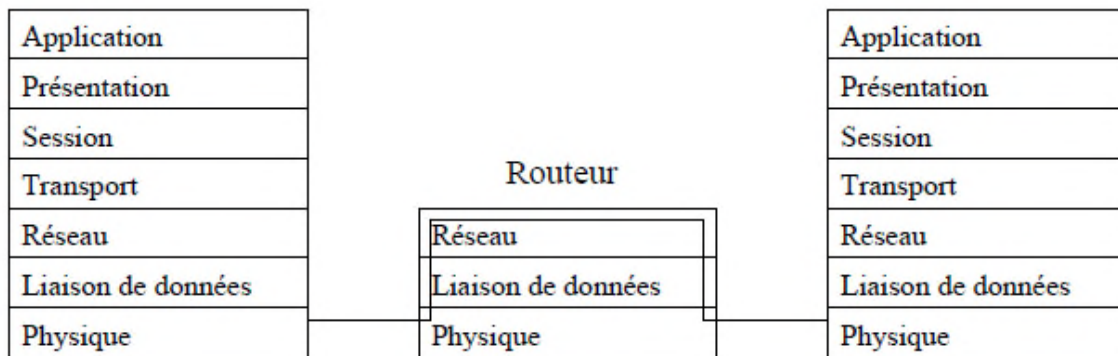


Figure 4:Principe du routeur

DEFINITION DE ROUTAGE

Le routage est l'acheminement des informations à la bonne destination à travers un réseau à partir de son adresse IP qui se base sur des algorithmes propres au protocole de routage, et qui prend en considération plusieurs facteurs importants, comme la durée moyenne de la transmission, la charge du réseau, la longueur totale du message.

DEFINITION DE LA TABLE DE ROUTAGE

La table de routage est une structure de données utilisée par un routeur ou un ordinateur qui comporte l'ensemble des adresses réseaux connues.

LE CONTENU DE LA TABLE DE ROUTAGE

La table de routage contient :

Les coûts liés à l'envoi des données.

Les adresses du routeur lui-même.

Les adresses des sous-réseaux auxquels le routeur est directement connecté.

Les routes statiques.

Les routes dynamiques, apprises par des protocoles de routage dynamique comme BGP, OSPF, etc.

Une route par défaut.

TYPE DE ROUTAGE

LE ROUTAGE STATIQUE

Dans ce cas, la table de routage est initialisée manuellement par l'administrateur. Les chemins possibles sont configurés une fois pour toute et ne sont changés que si on modifie la configuration du routeur. Les routeurs intermédiaires ne prennent aucune décision de routage. Ce type de routage est simple peut être utilisé pour un réseau local avec une connexion externe [2].

LE ROUTAGE DYNAMIQUE

La table de routage est mise à jour périodiquement à l'aide des protocoles spécifiques. Les routeurs envoient régulièrement la liste des réseaux ou des sous-réseaux que l'on peut atteindre par eux. Ceci permet aux autres routeurs de mettre à jour leurs tables de routage et d'évaluer dynamiquement la meilleure route vers chaque réseau [2].

LES ALGORITHMES DE ROUTAGE DYNAMIQUE

Il existe deux types d'algorithmes de routage dynamique.

LES ALGORITHMES A VECTEUR DE DISTANCE (VECTOR DISTANCE)

Les informations échangées permettent pour chaque routeur de retenir la plus courte distance (le plus petit nombre de sauts) pour atteindre une destination.

Ils sont basés sur l'algorithme de Belman-Ford :

Un routeur diffuse régulièrement à ses voisins les routes qu'il connaît.

Une route est composée d'une adresse destination, d'une adresse de routeur et d'une métrique indiquant le nombre de saut nécessaires (la distance) pour atteindre la destination.

Un routeur qui reçoit ces informations compare les routes reçues avec ses propres routes connues et met à jour sa table de routage :

Si une route reçue comprend un plus court chemin (nombre de sauts plus inférieur).

Si une route reçue est connue.

Ce type d'algorithme que l'on retrouve dans le protocole RIP à l'avantage de la simplicité pour des réseaux limités mais présentes plusieurs inconvénients parmi lesquels :

La taille des informations de routage est proportionnelle au nombre de routeurs interconnectés.

La métrique de distance est difficilement utilisable sur des réseaux étendus car elle présente une grand lenteur de convergence (beaucoup d'échange sont nécessaires avant d'obtenir des valeurs de distance optimisées et sa tables).

Des bouclages peuvent exister, éventuellement à l'infini.

Il ne peut y avoir de chemins multiples [3].

LES ALGORITHMES DE ROUTAGE A ETAT DE LIENS (LINK-STATE)

Ils sont basés sur la transmission d'une carte complète des liens possibles entre les routeurs. Ceux-ci doivent ensuite calculer, localement, les meilleures routes pour une destination. Ils sont basés sur la technique du plus court chemin SPF (Shortest Path First).

Les routeurs maintiennent une carte complète du réseau et calculent les meilleurs chemins localement en utilisant cette topologie.

Les routeurs ne communiquent pas la liste de toutes les destinations connues (contrairement aux algorithmes Vector Distance).

Un routeur basé sur l'algorithme SPF teste périodiquement l'état des liens qui la relie à ses voisins, puis diffuse périodiquement ses états (Link-State) à tous les autres routeurs du domaine.

Les messages diffusés ne spécifient pas des routes, mais simplement l'état (up, down) entre deux routeurs.

Lorsqu'un message parvient à un routeur, celui-ci met à jour la carte de lien et calcule localement, pour chaque lien modifié, la nouvelle route selon l'algorithme de Dijkstra (Shortest Path Algorithm) qui détermine le plus court chemin pour toutes les destinations à partir d'une même source.

Ce type d'algorithme utilise le protocole OSPF et présente un certain nombre d'avantages :

La convergence est rapide et sans boucle.

Les chemins multiples sont possibles.

Les métriques ne sont pas limitées à la distance.

Chaque routeur calcule ses routes indépendamment des autres.

Les messages diffusés sont inchangés d'un routeur à l'autre et permettent un contrôle aisé en cas de dysfonctionnement.

Les messages ne concernent que les liens directs entre routeurs et ne sont donc pas proportionnels au nombre de réseaux dans le domaine [3].

LES PROTOCOLES DE ROUTAGE

Pour l'Internet, qui est constitué d'interconnexion de réseaux, une organisation hiérarchique doit être établie au niveau du routage :

Le routage à l'intérieur de systèmes autonomes qui correspondent à un domaine de routage lié à un découpage de l'internet et sous la responsabilité d'une autorité unique (un SA est identifié par un numéro unique attribué par le NIC).

Le routage d'interconnexion entre les SAs.

Ces deux niveaux de routage font appel à des protocoles spécifiques :

Les protocoles de routage interne IGP (Interior Gateway Protocol) telles que RIP et OSPF qui concerne les routeurs internes.

Les protocoles de routage externe comme EGP (Exterior Gateway Protocol) ou BGP (Border Gateway Protocol) utilisés par les routeurs externes ou les routeurs de bord (Border Router).

LE PROTOCOLE RIP (ROUTING INFORMATION PROTOCOL)

Le protocole RIP est un protocole de routage à vecteur de distance qui utilise le nombre de sauts comme métrique pour déterminer la direction et la distance vers n'importe quelle liaison de l'inter réseau. S'il existe plusieurs chemins vers une destination, le protocole RIP sélectionne celui qui comporte le moins de sauts. Toutefois, le nombre de sauts étant la seule métrique de routage utilisée par ce protocole, il ne sélectionne pas toujours le

chemin le plus rapide. En outre, le protocole RIP ne peut acheminer un paquet au-delà de 15 sauts. La version 1 du protocole RIP (RIPv1) n'incluant pas les informations de masque de sous-réseau dans les mises à jour de routage, tous les équipements du réseau doivent nécessairement utiliser le même masque de sous-réseau. On parle dans ce cas de routage par classes.

La version 2 (RIPv2) fournit un routage par préfixe et envoie les informations de masque de sous-réseau dans ses mises à jour de routage. On parle ici de routage sans classe. Avec les protocoles de routage sans classe, les sous-réseaux d'un même réseau peuvent comporter des masques différents. Cette technique fait référence à l'utilisation de masques de sous-réseau de longueur variable [4].

LE PROTOCOLE OSPF (OPEN SHORTEST PATH FIRST)

Il fait partie d'une deuxième génération de protocole de routage. Il est beaucoup plus complexe que RIP, mais ses performances sont supérieures. Le protocole OSPF utilise une base de données distribuée, qui garde en mémoire l'état des liaisons. Ces informations forment une description de la topologie du réseau et de l'état des nœuds qui permet de définir l'algorithme de routage par un calcul des chemins les plus courts.

Cinq types des liaisons ont été définis :

Les liaisons à partir d'un routeur.

Les liaisons de réseau de transit.

Les récapitulations de réseau IP qui partent des routeurs interzones.

Les récapitulations de routeurs externes qui arrivent sur des routeurs interzones.

Les liaisons externes.

L'algorithme OSPF est mis en œuvre à partir des bases de données. Il permet à partir d'un nœud, de calculer le chemin le plus court avec les contraintes indiquées dans les contenus associés à chaque liaison. Les routeurs OSPF communiquent entre eux par l'intermédiaire du protocole OSPF, qui est placé au-dessus de la couche IP.

Le protocole RIP est adapté à la gestion du routage dans de petits réseaux, alors OSPF s'applique à des réseaux beaucoup plus complexes d'interconnexion de sous-réseaux [5].

LE PROTOCOLE EGP (EXTERIOR GATEWAY PROTOCOL)

Deux routeurs, appartenant à des systèmes autonomes distincts, et qui s'échangent des informations de routage, sont appelées voisins externes. Ils sont appelées voisins internes s'ils appartiennent au même système autonome. Le protocole utilisé par les routeurs externes pour propager les informations d'accessibilité vers les autres systèmes autonomes est appelé protocole de routeur externe, EGP. Ce dernier est particulièrement important dans l'internet car les systèmes autonomes l'utilisent pour diffuser les informations d'accessibilité vers le réseau fédérateur.

Ce protocole est composé essentiellement, de trois procédures qui permettent la mise en place de l'échange d'information entre système autonomes. La première procédure concerne la définition d'une passerelle voisine. Le seconde détermine la liaison qui va leur permettre de communiquer. La troisième procédure concerne alors l'échange de paquets entre les deux voisins, sur la liaison entre système autonome [6].

LES LIMITES D'EGP

Les faiblesses EGP sont apparues avec le développement exponentiel d'internet et les besoins d'éviter des routeurs situés dans des zones politiquement sensibles.

EGP a beaucoup des caractéristiques d'un protocole de routage à vecteur de distance. Annoncer une route pour une destination, signifie que les utilisateurs externes vont utiliser cette route. Dans certains cas, par exemple, pour les circuits X.25, la facture mensuelle dépendra directement du volume du trafic; laisser d'avantage d'utilisateurs envoyer plus de données, aura une influence immédiate sur les sommes à payer. Même s'il n'y a pas d'effet direct sur les factures, ajouter du trafic externe n'est pas neutre; cela utilise des ressources de transmission.

LE PROTOCOLE BGP (BORDER GATEWAY PROTOCOL)

Le protocole BGP est un exemple de protocole EGP qui permet l'échange d'informations de routage entre systèmes autonomes, tout en garantissant une sélection de chemins exemple de boucle. Le protocole BGP est le protocole de mises à jour de routage le plus

utilisé par les grandes sociétés et les FAI sur internet. BGP4 est la première version de BGP à prendre en charge le routage inter-domaine sans classes (CIDR) et le regroupement des routes. À la différence des protocoles IGP courants, comme RIP et OSPF, le protocole BGP ne se sert pas de métriques tels que le nombre de sauts, la bande passante ou le délai. Il prend à la place ses décisions de routage, selon des stratégies de réseau (ou règles utilisant divers attributs de chemin BGP) [6].

CONCLUSION

Comme on vient de le voir, l'internet est constitué d'un ensemble de réseaux sur lesquels une multitude de protocoles coopèrent pour transporter des flux d'informations entre les différentes machines qui veulent communiquer, en utilisant les mêmes protocoles de routage tels que RIP, OSPF, EGP, BGP et des protocoles de transport TCP/IP.

CHAPITRE 2**INTRODUCTION**

La technologie Internet a fait un grand succès en matière d'interconnexion des appareils de communication, grâce à l'utilisation d'un ensemble de protocoles homogènes appelés TCP/IP.

Des centaines de milliers de dispositifs qui composent le réseau Internet utilisent ces protocoles, pour le routage de données et la garantie de la fiabilité des échanges de messages. Cette qualité de transmission provient du fait que TCP/IP offre plusieurs avantages. En effet, c'est un protocole qui se base sur des chemins de bout en bout. Afin de faire bénéficier tout le monde de la connexion Internet et leur apporter les outils de l'ère numérique, des projets de connexion des villages isolés ont été lancés. Néanmoins, le manque d'infrastructures dans ces zones a causé des débits de transmission faibles ou nuls dans certains cas. Ce qui empêche de trouver des connexions de bout en bout tout au long d'une communication.

C'est dans ce cadre qu'une nouvelle architecture de réseau a vu le jour, celle des réseaux tolérants aux délais (ou DTN : Delay Tolerant Network).

DEFINITION

Un DTN est un réseau de réseaux régionaux. C'est un overlay au-dessus des réseaux régionaux, incluant l'Internet. Ces réseaux sont sujets à des connexions intermittentes. Un DTN assure donc l'interopérabilité des réseaux :

En s'accommodant des longs délais entre (ou dans) les réseaux régionaux.

Traduit les caractéristiques de communication entre les réseaux régionaux.

CARACTERISTIQUE DES RESEAUX DTNS

Les caractéristiques des réseaux tolérants aux délais répondent bien aux problèmes rencontrés dans les réseaux Internet traditionnels, à savoir :

CONNECTIVITE INTERMITTENTE

S'il n'y a pas de chemin de bout en bout (partitionnement du réseau), un protocole comme TCP/IP ne peut pas fonctionner. TCP retarde la libération des buffers car TCP est un protocole de bout en bout. Plus le temps de transmission est élevé, plus les buffers seraient gourmands en espace mémoire. Ceci impliquerait alors une augmentation de la puissance utile et finalement une augmentation des coûts.

Pour toutes ces raisons, il ne paraît donc pas possible d'utiliser le protocole TCP.

De même, UDP n'est pas satisfaisant. Tout d'abord, UDP n'utilise pas ARQ et par conséquent, l'acquittement des données devra se faire en passant par la couche qui se trouve au-dessus d'UDP, c'est-à-dire la couche application. Cependant, pour les mêmes raisons que TCP, une transmission bout à bout au sein de la couche application ne serait pas adaptée aux communications spatiales.

Ainsi, les protocoles classiques sont inefficaces pour de telles communications d'où la création d'un autre protocole au sein du DTN qui s'appelle bundle protocole.

DELAIS LONGS ET VARIABLES

Dû au problème de propagation et aux temps d'attente dans les files des nœuds intermédiaires, les protocoles Internet et les applications qui comptent sur le retour d'ACK rapide ne pourront fonctionner. De nouveaux type de réseaux apparaissent et ne se conforment pas aux hypothèses de base faites sur l'Internet.

VITESSE DE TRANSMISSION ASYMETRIQUE

Internet supporte une asymétrie modérée (ADSL). Mais dans le cas d'asymétrie importante, cela empêche le bon fonctionnement de protocoles conversationnel.

TAUX D'ERREUR IMPORTANT

Les erreurs de bit sur une liaison exigent des corrections (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 hop by hop que pour une retransmission de bout en bout.

LES ASPECTS FONDAMENTAUX DES RESEAUX DTN

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 DTNs, 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 [7], tel qu'illustré dans la Figure 5, les protocoles des couches inférieures qui supportent la couche bundle peuvent être conversationnels comme TCP, mais dans les liens connectés de manière intermittente avec de longs délais, des protocoles non conversationnels ou peu conversationnels, sont implémentés parfois même au niveau des couches inférieures.

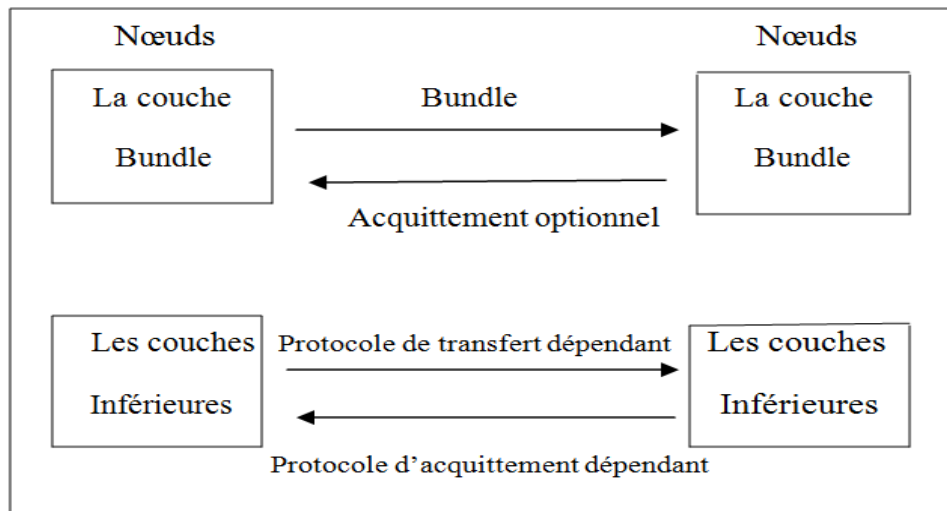


Figure 5: Protocole non conversationnel

LA TECHNIQUE STORE AND FORWARD

Le DTN résout tous les problèmes rencontrés par les protocoles bout à bout en utilisant la méthode *store-and-forward message switching*. Il s'agit de transmettre, le long du chemin voulu, les informations d'une zone de stockage à une autre (Voir figure 6). Cette opération nécessite des zones de stockage, telles des disques durs, permettant de stocker des messages pendant des jours, et non de simples allocations dynamiques de mémoire de quelques millisecondes comme cela aurait pu être le cas avec une autre méthode. De telles capacités de mémoire sont nécessaires car il se peut qu'un lien entre deux nœuds consécutifs ne soit pas disponible pour une longue période. En outre, si une erreur se produit, le message devra être mémorisé dans ces zones afin de pouvoir être retransmis.

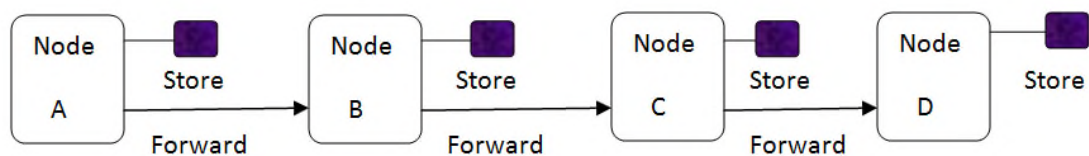


Figure 6: La technique de store-and-forward

ARCHITECTURE DES RESEAUX DTNS

L'architecture des réseaux DTN est basée sur les éléments suivant :

LA COUCHE BUNDLE

Les réseaux DTN mettent en œuvre les techniques store and forward de commutation de données en ajoutant au protocole existant une nouvelle couche protocolaire appelée couche bundle (Bundle Layer).

La couche bundle stocke et transmet tous les bundles (message) entre les nœuds. Elle supporte des transmissions de bout en bout. Les bundles sont automatiquement transportés d'un nœud au suivant, bien qu'elle puisse fragmenter un bundle en plusieurs fragments.

Cette couche permet les retransmissions de nœuds à nœuds en s'appuyant sur la garde de transfert. Un tel transfert est organisé entre les couches bundle des nœuds successifs, à la demande initiale de l'application source. Lorsque le gardien de la couche bundle courante envoie un bundle au nœud suivant, il demande une garde de transfert (Custody Transfer) et arme un timer TTA (Time-To-Acknowledge). Le TTA est le temps au bout duquel le bundle est envoyé à nouveau si le nœud suivant n'a pas retourné un accusé de réception. Par conséquent, le gardien (Bundle Custodian) devra conserver le message jusqu'à ce qu'un nœud accepte la garde (custody) ou le délai d'acheminement TTL (time to live) du message soit expiré.

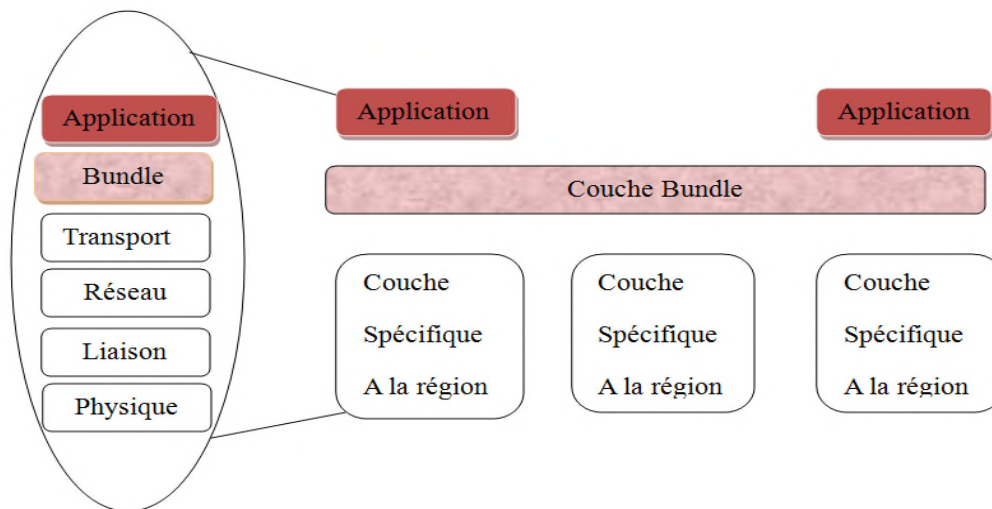


Figure 7: La couche bundle

Un seul protocole est utilisé à travers l'ensemble du réseau appelé bundle protocole qui constitue le DTN. Seuls les protocoles inférieurs sont alors spécifiques à l'environnement de communication de chaque région.

Le bundle protocole se situe entre la couche applicative et les couches spécifiques à chaque région. Dans le cas de TCP/IP, il se situe ainsi comme une couche supérieure à la couche transport.

LES MESSAGES BUNDLE

Un bundle aussi appelé message consiste essentiellement de trois choses :

Les données de l'application source.

Les informations de contrôle fournies par l'application source qui décrivent comment le traiter, le sauvegarder, autrement dit comment récupérer les données utilisateur.

Les en-têtes insérés 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 exécutées par le protocole utilisé dans l'Internet.

En effet, la couche bundle peut découper les paquets (ou les messages) en fragments comme la couche IP qui découpe les datagrammes en fragments (au niveau des routeurs) puis ces derniers sont réassemblés au niveau du nœud destinataire.

Il existe deux formes de fragment/assemblage pour les bundles [8] :

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 destinataires finaux sont responsables de l'assemblage des petits blocs qu'ils reçoivent en un seul bloc (l'original). Cette forme de fragmentation est analogue à la fragmentation IP.

Le routeur de bundle 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 continue 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 bundle mais qui n'a aucun contact qui fournit un volume de transfert de données suffisant.

LES ENTITES DE COMMUNICATION

LES NŒUDS

Un nœud DTN est un dispositif pour l'envoi et la réception de messages (appelés aussi Bundles). Il peut jouer le rôle de la source, de destination ou de nœud intermédiaire pour la transmission des 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 plusieurs rôles de [7] :

Hôte : Un nœud qui émet ou reçoit des 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.

Routeur : Un nœud qui transmet un bundle à un autre nœud dans une même région DTN et peut optionnellement jouer le rôle d'un hôte.

Passerelle : Elle 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 les protocoles spécifiques pour les différentes régions.

Les différents rôles peuvent être schématisés comme le montre la figure suivant :

Hôte	Routeur		Passerelle	
Application	Application		Application	
Bundle	Bundle		Bundle	
Transport	Transport	Transport	Transport	Transport
Réseau A	Réseau A	Réseau A	Réseau A	Réseau B
Liaison A	Liaison A	Liaison A	Liaison A	Liaison B
Physique A	Physique A	Physique A	Physique A	Physique B
	Diffusion dans la même région		Diffusion entre deux régions différentes	

Figure 8: Les différents rôles d'un nœud DTN [7]

LES REGIONS DTN

L'architecture DTN définit un réseau de plusieurs réseaux, dans lesquels chaque réseau 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. En d'autres termes, une région est une zone qui est influencée par les familles de protocoles, les dynamiques de connexion, les politiques administratives. D'une manière générale, les régions DTN sont délimitées en se basant sur un critère appelé les frontières de confiance [7].

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, on peut retrouver le nom d'une région à tout moment.

Les bundles DTN 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.

Les régions DTN sont caractérisées par [9]:

Chaque région doit avoir un espace identifiant partagé par tous les nœuds de la région, et doit spécifier des conventions de nommage internes 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 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 d'une région, le nœud 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* ou de transmission par les autres nœuds de la même région.

NOMMAGE ET ADRESSAGE DANS UN RESEAU DTN

DTN est un réseau de réseaux où chaque réseau représente une région à l'intérieur de laquelle ont lieu des communications homogènes. Il est nécessaire donc de pouvoir identifier les régions. Le tuple désignant le nom d'un nœud est composé de deux parties :

Un identificateur de la région ou le nom de région.

Un 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 peuvent appartenir à deux ou plusieurs régions. Elles déplacent les bundles entre ces régions. Ainsi ces passerelles possèdent plusieurs IDs régions [9].

Exemple : { earth.sol.int, src.someclient.com:1131 }. Dans cet exemple, earth.sol.int représente l'identifiant de la région, et src.someclient.com:1131 celui du nœud, c'est à dire une URL et un port. Ainsi, au niveau du routage inter-région, c'est l'identifiant de la région qui est utilisé alors que pour le routage intra-région c'est celui du nœud, le routeur connaissant le mécanisme d'adressage utilisé dans sa région. Enfin, en faisant abstraction des mécanismes d'adressage spécifique, les IDs permettent d'identifier indifféremment les machines ou les adresses *Anycast*, *Broadcast* ou *Multicast*.

Du fait des contraintes de communications, l'identifiant du nœud n'est pas résolu au niveau de la source (Comme c'est le cas pour les résolutions DNS). Cela pourrait, en effet, entraîner des délais supplémentaires. Cet identifiant est résolu à l'arrivée du message dans la région cible, par la passerelle. Ce type de résolution, appelé *late binding*, offre plusieurs avantages dans le cadre d'un DTN:

- Pas besoin de connaître tous les systèmes d'adressage
- Moins de communications administratives (résolution ou synchronisation des identifiants)

DIFFERENCE ENTRE LE CHEMINEMENT INTERNET ET DTN

Sur l'Internet, les protocoles TCP et IP sont employés dans tout le réseau. TCP fonctionne aux points finaux de l'extrémité du chemin, ou il contrôle et assure une livraison bout à bout fiable des messages. IP fonctionne sur tous les nœuds le long du chemin, où il route les datagrammes des messages. Les routeurs Internet n'exigent pas

une couche transport pour le routage, mais ils mettent en application les deux couches transport et application pour l'entretien et la mise à jour de leur tables de routage. Dans un réseau DTN, les piles de protocole de tous les nœuds incluent au même temps la couche bundle et la couche transport [9]. Les passerelles DTN ont la même pile de couche que les routeurs DTN mais en double, puisque les passerelles peuvent contenir des protocoles de couche inférieures différentes (au-dessous de la couche bundle) de chaque coté de leur double pile. Ceci permet à ces passerelles d'enjamber deux régions qui emploient différents protocoles des couches inférieures.

FIABILITE DANS LES RESEAUX DTNS

LE TRANSFERT DE GARDE

Les réseaux DTNs 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 DTNs, 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 [1] qui, consiste en la persistance d'un message au niveau d'un nœud appelé gardien. Nous pouvons voir à travers la Figure 9 que les transferts sont disposés entre les couches bundle des nœuds successifs à la requête initiale de l'application source, en 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:

L'obtention de l'acceptation de la garde par un autre nœud.

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.

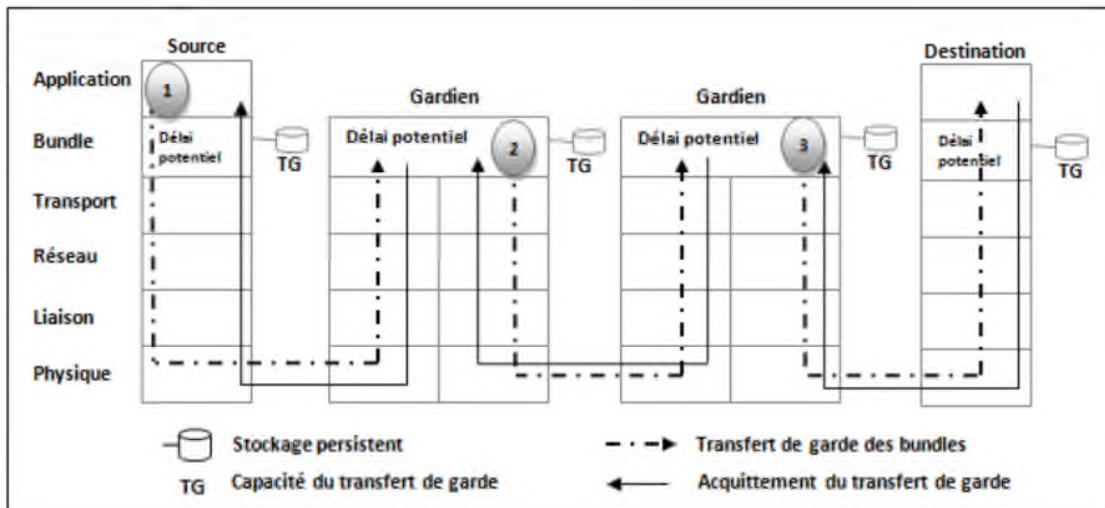


Figure 9: Transfert de garde

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 DTN doit être en mesure d'être le gardien car le service de garde est assuré.

ISSUE DE LA COUCHE TRANSPORT DANS UN DTN

FIABILITE SAUT PAR SAUT

L'accusé est envoyé saut par saut. Si un nœud reçoit l'accusé du nœud auquel il a envoyé le message, il libère ses ressources.

Cette approche n'est pas fiable, car s'il se produit une panne dans le chemin, la destination ignore complètement la source.

L'ACCUSE ACTIF

Dans cette approche, le message M original sera acheminé à la destination normalement (sans accusé). A la réception de M, la destination génère un nouveau message R qu'elle va envoyer à tous les nœuds rencontrés (même ceux qui ne sont pas infectés par M) jusqu'à la source.

Cette solution assure la fiabilité mais elle est trop coûteuse.

L'ACCUSE PASSIF

A la réception de M, la destination envoie un message K (Kill) implicite à la rencontre d'un nœud infecté, de nœud à nœud jusqu'à la source.

L'ACCUSE A PASSERELLE RESEAU

Le principe est d'utiliser le DTN pour l'acheminement du message de la source à la destination et d'utiliser le réseau cellulaire pour le routeur de R. cette solution est très rapide.

SECURITE DANS LES RESEAUX DTNS

Dans les réseaux DTNs, la sécurité repose sur le mécanisme de cryptographie à clé publique, l'ensemble des utilisateurs et les nœuds de commutation ont une paire de clés (privée et publique) et un certificat. L'émetteur peut signer ses bundles avec sa clé privée produisant la signature numérique spécifique du paquet. La signature permet au récepteur d'authentifier l'émetteur et l'intégrité du message en décryptant le bundle avec la clé publique de ce dernier.

En utilisant la méthode de cryptographie à clé publique, les étapes de sécurité sont les suivantes :

La source envoie son bundle avec la signature spécifique du paquet, à un nœud de commutation adjacent. Si ce nœud ne possède pas encore le certificat de l'émetteur, il l'obtient auprès de l'émetteur ou auprès de l'autorité de certification(CA).

Le nœud commutateur qui reçoit le premier bundle de la source, vérifie l'identité de l'utilisateur et ses droits d'accès, en utilisant ses copies des certificats des utilisateurs

voisins et la clé publique de CA. Ensuite, remplace la signature de l'émetteur par sa propre signature et commute le message.

Chaque nœud de commutation suivant, vérifie seulement l'identité du précédent en procédant de la même manière que le premier nœud.

LA CONGESTION ET LE CONTROLE 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 [11].

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 [11].

A. Qu'est-ce que la congestion dans les DTNs ?

Dans les réseaux Internet, nous parlons de congestion lorsque nous constatons une croissance rapide et instantanée dans l'occupation de l'espace de stockage. Par contre, dans les DTNs, la congestion signifie la croissance constante et continue dans l'occupation de l'espace mémoire [11]. 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.

B. 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 [9]. Afin de limiter le taux de trafic au niveau des sources, un contrôle de flux est imposé.

C. Le contrôle de flux

Le contrôle de flux garantit le fait que le taux moyen de données transmises par les nœuds sources ne soit pas supérieur au taux moyen de données que les nœuds récepteurs ont prévu de recevoir [9].

D. 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 [12]. Ceci impose un contrôle de flux dans l'application source, qui réduit indirectement le taux de croissance excessif dans l'occupation des espaces buffers au niveau de la destination.

Le contrôle de congestion 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 [10]:

Explicite

Le routeur envoie un paquet ICMP à la source, qui se traduit par un taux de transmission TCP réduit. Puis, comme expliqué précédemment, ceci impose un contrôle de flux au niveau de l'application source.

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.

E. Le contrôle de flux et le contrôle de congestion dans les DTNs

Dans les réseaux DTNs, 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. Néanmoins, la congestion dans les routeurs DTN ne peut être traitée directement par un contrôle de flux au niveau

de l'application source, car il n'y a aucune garantie sur la présence d'une connectivité de bout en bout dans aucune route. Pour cela, elle doit être traitée 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 [10] :

Lorsque le routeur n'a plus d'espace de stockage, il rejette les bundles qui continuent à arriver.

Les notifications sur un transfert de garde ne seront donc pas envoyées pour les bundles rejetés.

L'absence de la notification provoque une congestion au niveau du gardien, causant éventuellement un rejet de bundle aussi.

Cette propagation de rejet de 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 DTNs 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 [13] :

Ecarter les bundles qui ont expiré, ce qui est une activité DTN qui doit s'exécuter dans n'importe quel cas.

Faire en sorte que la garde du bundle soit cessée dans le cas où les nœuds étaient disposés à le faire.

S'en servir des ressources de stockage disponibles dans le réseau.

Ecarter les bundles qui n'ont pas expiré, mais pour lesquels la garde n'a pas été acceptée.

CONCLUSION

Dans ce chapitre, nous avons présenté un état de l'art sur les réseaux tolérants aux délais. Nous avons introduit la notion de DTN, le contexte de leur apparition et nous avons expliqué leur fonctionnement en présentant la technique 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. Le principe de routage dans ce réseau sera présenté dans le chapitre suivant.

CHAPITRE 3

INTRODUCTION

Le réseau DTN est un environnement de nature déconnecté du fait de ses caractéristiques. Différentes solutions permettent d'assurer un routage correct au sein des réseaux DTN. Dans ce chapitre nous allons présenter les concepts, les étapes de base de routage dans les réseaux tolérant au délai, la classification des protocoles de routage dans ce réseau ainsi que les différents algorithmes de routage utilisés.

LES ETAPES DE ROUTAGE DANS UN DTN

En général, le transfert de message dans un DTN se déroule en quatre étapes : [9]

Attendre une opportunité de transfert : chaque nœud qui génère un message pour un nœud destinataire retient 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 : Le nœud sélectionne les 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.

QUELQUES DEFINITIONS

Nous allons présenter quelques définitions concernant le routage, le temps de transfert de message, les ressources du réseau et les critères d'évaluation des algorithmes de routage.

LE TEMPS DE TRANSFÈRE DE MESSAGE

Le temps total nécessaire pour le transfert de message tel que définit par Kevin et Al [16] est divisé en quatre composantes : le temps d'attente, le temps d'attente dans la file, le temps de transmission et le temps de latence ;

Le temps d'attente : Il correspond au temps d'attente du message entre son arrivée au nœud et le moment de la disponibilité du contact.

Le temps d'attente dans la file : C'est le temps expiré durant le passage des messages les plus prioritaires.

Le temps de transmission : Il correspond au temps nécessaire pour la transmission de tous les bits du message.

La latence : C'est le temps de propagation d'un bit à travers le support de communication.

LES RESSOURCES DU RESEAU

Dans les réseaux DTNs, en plus de la déconnexion qui pose problème au routage, nous sommes en présence de ressources limitées qui peuvent être :

Instant du contact : Il représente la possibilité de prédire la disponibilité des contacts entre des nœuds, qui peut être exacte (contact programmé), approximative ou aléatoire.

Capacité du contact : Il correspond à la quantité de données qui peut être échangée entre deux nœuds durant la disponibilité du contact.

Espace de stockage : Pour remédier au problème de déconnexion, tous les routeurs du réseau doivent être dotés d'un espace de stockage pour pouvoir stocker les messages durant la période de déconnexion.

Capacité de traitement : C'est la capacité des mémoires et des CPU à exécuter des protocoles de routage complexes.

Energie : Un nœud DTN doit avoir de l'énergie pour s'alimenter durant le mouvement.

CONCEPT DE BASE DE ROUTAGE

Parmi les concepts de routage on trouve :

ROUTAGE PROACTIF

Les protocoles de routage proactifs s'efforcent de constamment maintenir les routes disponibles permettant à un nœud d'atteindre l'ensemble des autres nœuds du réseau. Lorsqu'il souhaite transmettre un message, l'émetteur regarde dans sa table de routage pour déterminer par quel nœud voisin il doit faire transiter ce message afin que celui-ci arrive à destination. L'existence de telles tables de routage est possible car chaque nœud envoie périodiquement sa propre table à ses voisins et choisit, pour chaque destination, le (ou les) voisin(s) qui offre(nt) la route la plus intéressante. Un nœud détecte lorsqu'un voisin disparaît ou apparaît et met alors à jour sa table de routage en fonction et grâce à l'échange fréquent de messages dédiés.

Parmi les plus connus, on peut citer :

Optimized Link State Routing Protocol (OLSR): Chaque nœud est capable de connaître son voisinage à deux sauts (chacun de ses voisins lui fournissant son voisinage direct).

Destination-Sequenced Distance-Vector (DSDV) [17] : Les tables de routage utilisées par ce protocole de routage sont construites de proche en proche, par envoi de tables de routage complète entre voisins, en cas de grosse modification de son voisinage.

ROUTAGE REACTIF

Les protocoles de routage réactifs s'efforcent de découvrir les routes nécessaires à l'acheminement d'un message seulement lorsque c'est nécessaire (i.e. lorsqu'une application le demande et qu'il n'existe pas déjà une route précédemment découverte), on parle alors, de routage à la demande.

Lorsqu'il souhaite envoyer un message, le nœud source effectue une demande de route qui consiste à inonder le réseau d'un message particulier. Une fois que ce message est reçu par le nœud destination, l'information comprenant la route qu'il a utilisée est renvoyée à la source qui peut alors transmettre son (ses) message(s).

Parmi les protocoles les plus connus, on peut citer :

Dynamic Source Routing (DSR) [14] : Dans ce protocole, une fois la route trouvée, c'est la source qui indique dans le message le chemin qu'il doit emprunter pour arriver à destination.

Ad hoc On Demand Distance Vector (AODV) [13] : Dans ce protocole, et contrairement à DSR, une fois une route trouvée, la table de routage de chacun des nœuds de la route connaît le prochain nœud par lequel il doit faire transiter le message pour qu'il arrive à destination.

ROUTAGE SOURCE :

Le routage source permet de déterminer le chemin complet que doit suivre le message, depuis le nœud source à destination. 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.

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.

ROUTAGE HIERARCHIQUE :

Ce type de routage facilite la hiérarchisation du réseau en topologie abstraite et peut ne pas générer le plus court chemin.

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 au 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 [15]:

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

Définir une source intermédiaire S_0 et une destination intermédiaire D_0 qui sont des clusters de niveau k de S et D respectivement.

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 .

Si $k = 0$, alors 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 .

ROUTAGE PAR LA REPLICATION :

Dans les réseaux DTNs, l'amélioration des performances pour le routage passe, souvent, par l'ajout de diversité dans les opportunités de communication. De ce fait, un grand nombre de protocoles dans ce domaine utilise la réplication.

Les protocoles basés sur cette stratégie ne tiennent pas en compte le fait que les nœuds contiennent des informations pouvant être utilisées pour déterminer le chemin. La source envoie plusieurs copies d'un même message à un ensemble de nœuds, appelés relais (Relay). Ces derniers vont garder le message jusqu'à ce que la connexion avec le destinataire soit établie.

Il existe plusieurs approches de routage basées sur cette stratégie, nous présentons quelques unes :

- **La stratégie de contact direct (Direct Contact)** : Les algorithmes de routage basés sur cette approche attendent que la source soit en contact avec la destination avant de transférer les données. Cependant, les protocoles qui implémentent cette approche ne fonctionnent que lorsque la source et la destination peuvent entrer en contact.

- **La stratégie de routage par propagation (Epidemic Routing) :** Cette approche, initialement proposée pour la synchronisation des bases de données dupliquées, a été appliquée par Vahdat et Becker pour la transmission des données dans les DTNs.
- **Le routage épidémique :** Chaque nœud peut être le gardien (Custody), ce qui rend la probabilité que le message soit délivré très élevée. Il présente cependant, l'inconvénient de consommer beaucoup de ressources. Plus encore, le message continue à se propager sur le réseau après avoir été livré à l'application destinataire.

ROUTAGE PAR LA CONNAISSANCE :

Quelques stratégies de routage exigent plus d'informations sur le réseau que d'autres. A une extrémité, un nœud peut prendre des décisions avec zéro connaissance à propos du réseau, excepté les contacts qui sont couramment disponibles.

Ces stratégies emploient des règles statiques qui sont configurées initialement, et tous les nœuds vont les obéir. Ceci amène aux simples implémentations qui exigent une configuration minimale et des messages de contrôle, puisque toutes les règles sont codées préalablement.

L'inconvénient est que, cette stratégie ne peut pas être adaptée à tous les réseaux ou à toutes les conditions, ainsi elle ne prend pas toujours des décisions optimales.

A l'autre extrémité, le nœud peut avoir besoin des futures disponibilités de chaque contact dans le réseau. Ainsi, plus l'information est précise, plus la stratégie de routage utilise le réseau et ses ressources de façon très fiable, et ce par l'expédition d'un message via le meilleur chemin.

CLASSIFICATION DES PROTOCOLES DE ROUTAGE DTN:

Les protocoles de routage dans les réseaux DTNs utilisent deux principales propriétés:
La réplication où plusieurs copies du message sont utilisées.

La connaissance qui indique comment obtenir des informations sur l'état du réseau, et comment qu'une stratégie les utilise dans le but de prendre une décision de routage.

Une première classification complète a été proposée par Jain & Al [16]. Celle-ci se base sur ces deux critères où le concept d'oracles de connaissances a été intégré. Les éléments de base de cette classification sont les suivants :

L'INONDATION:

Elle consiste en la délivrance de multiples copies par message pour un ensemble de nœuds appelés relais. Les relais stockent les messages jusqu'à se connecter avec la destination où les paquets des données seront délivrés. Les protocoles utilisant cette stratégie ne doivent avoir aucune information sur le réseau.

L'EXPEDITION:

Elle se base sur une approche plus traditionnelle pour l'acheminement des données dans un DTN. Elle utilise l'information sur la topologie du réseau pour sélectionner le meilleur chemin, sur lequel le message est alors envoyé en allant d'un nœud à l'autre. Cette technique a été explorée dans les protocoles développés pour les réseaux filaires. Néanmoins, puisque ces derniers supposent que les liens sont généralement connectés, alors les protocoles désignés pour ces environnements ne fonctionneront pas pour les réseaux tolérants aux délais. Pour cela, les stratégies de cette famille nécessitent quelques connaissances supplémentaires sur le réseau, puis déterminent le meilleur chemin, et envoient un seul message le long de celui-ci.

LES ORACLES DE CONNAISSANCE:

Le problème de routage DTN a beaucoup de variables d'entrées, par exemple : les caractéristiques de la topologie dynamique et la demande de trafic. La connaissance complète de ces variables facilite le calcul des routes optimales. Cependant, avec des connaissances partielles, la capacité de calcul des routes optimales est entravée, et les performances du routage résultant s'avèrent inférieures.

Afin de comprendre cette interaction fondamentale entre performance et connaissances, un ensemble abstrait d'oracles de connaissances a été créé [15]. Ces oracles sont des éléments utilisés pour encapsuler des connaissances particulières sur le réseau, requises par différents algorithmes. Ainsi, nous pouvons distinguer 4 types d'oracles :

ORACLE DE L'ETAT DES CONTACTS:

Cet oracle prend en charge les statistiques globales des contacts. Il fournit en particulier, le temps d'attente moyen jusqu'au prochain contact. Ainsi, l'oracle de l'état des contacts ne fournit que le temps invariant, ou un résumé des caractéristiques des contacts.

ORACLE DE CONTACT:

Cet oracle prend en charge tous les contacts entre deux nœuds à un temps donné. Ceci est équivalent à la connaissance du multi-graphe DTN dans un temps variant.

L'oracle de l'état des contacts peut être construit à base de l'oracle contact mais l'inverse n'est pas vrai.

ORACLE DE FILE D'ATTENTE:

Cet oracle donne des informations sur le taux d'occupation du buffer instantanément, à n'importe quel nœud, à n'importe quel moment, et peut être utilisé pour les routes autour des nœuds congestionnés.

A la différence avec les autres oracles, l'oracle de la file d'attente est affecté par les nouveaux messages qui arrivent au système, et le choix fait par l'algorithme de routage lui-même.

ORACLE DE DEMANDE DE TRAFIC:

Cet oracle prend en charge toute demande présente ou future du trafic. Il est capable de fournir l'ensemble des messages injectés dans le réseau à tout moment.

LES CONNAISSANCES PARTIELLES:

Les algorithmes de cette catégorie calculent les chemins en utilisant une ou plusieurs de ces informations : l'état des contacts, les contacts et les files d'attente. Chaque message est routé indépendamment de la future requête, car les connaissances sur le trafic ne sont pas utilisées. Ces algorithmes sont tous basés sur l'attribution de coûts aux liens, et le calcul d'une sorte de chemin à coût minimum. Les coûts sont attribués aux liens en se basant sur les informations disponibles, et ce dans le but de donner une idée sur le délai estimé pour un message les empruntant.

Les raisons pour lesquelles seulement les algorithmes basés sur le coût sont considérés dans cette classe sont :

Ils permettent d'utiliser les différentes connaissances d'une façon pratique pour l'utilisation, et identifient dans quelle mesure la connaissance globale est nécessaire.

Ils correspondent aux problèmes classiques de routage basé sur le plus court chemin qui sont bien maîtrisés, et pour lesquels les algorithmes efficaces à base de calculs simples sont connus.

Cependant, ces méthodes simples ne donnent pas plus d'un seul et unique chemin, ce qui est un inconvénient, car dans les DTNs, il est important d'utiliser de multiples chemins afin d'atteindre des performances assez optimales.

LES CONNAISSANCES COMPLETES:

Les algorithmes de cette classe utilisent tous les oracles (contacts, files d'attente, trafic) et représentent la première formulation complète du problème du routage pour avoir un routage optimal minimisant le délai moyen dans le réseau.

LA CLASSIFICATION DE JAIN & AL:

En se basant sur tous les critères présentés, Jain et Al ont proposé une classification des protocoles de routage [16]. Nous pouvons schématiser cette dernière par le diagramme suivant:

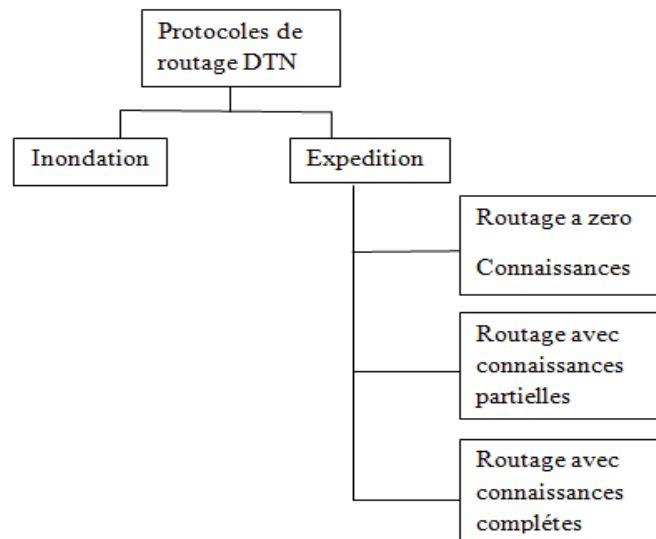


Figure 10:Diagramme résumant la classification de Jain et Al

LES ALGORITHMES DE ROUTAGE

Les algorithmes de routage sont divisé en deux :

ALGORITHME AVEC DES COÛTS INVARIABLES:

ALGORITHME MINIMUM EXPECTED DELAY(MED)

C'est une stratégie qui a été classée parmi celles qui utilisent les oracles de connaissance, elle utilise celui de l'état des contacts. Son principe se définit comme suit :

A chaque lien, est attribué un coût qui est la somme de la moyenne du temps d'attente, délai de propagation et le délai de transmission. L'itinéraire d'un message est indépendant du temps, ainsi une approche proactive peut être utilisée. L'algorithme MED utilise le même chemin pour tous les messages ayant la même paire source-destination. Aucun mécanisme n'est utilisé pour éviter la congestion ou la suppression du message si l'espace de stockage n'est pas disponible [16].

L'algorithme MED est caractérisé par le fait qu'il minimise le temps moyen d'attente, et qu'il ne parvient pas à exploiter les liens supérieurs qui deviennent disponibles après que

l'itinéraire soit calculé. En d'autres termes, si un contact direct vers la destination du message s'avère disponible au moment où le message est en attente d'un prochain saut pré-calculé, alors le contact ne sera pas utilisé.

Néanmoins des améliorations ont été proposées pour cette approche. Parmi elles, celle qui consiste à trouver de multiples chemins disjoints avec des coûts différents, et choisir au hasard parmi eux celui qui pourrait améliorer l'équilibrage de la charge et réduire la congestion [17].

Ou encore, considérer qu'une route pré-calculée pourrait être modifiée si un contact supérieur devient disponible. Ceci en effet crée une forme de routage à source perdue avec un comportement plutôt réactif.

La propriété principale de MED est qu'elle réduit au minimum le temps d'attente moyen. Il n'exploite pas les bords supérieurs qui deviennent disponibles après que l'itinéraire ait été calculé. Dans ce cas-ci, le nouveau contact ne serait pas employé.

ALGORITHME AVEC DES COÛTS VARIABLES

ALGORITHME DE LIVRAISON LE PLUS RAPIDE (EARLIEST DELIVERY ALGORITHM)

Cette approche se base sur l'oracle des contacts, et son principe est le suivant :

Soit Q un paramètre qui permet de distinguer entre la file d'attente locale et la file d'attente globale, initialisé à 0 dans cet algorithme.

Cet algorithme utilise le routage source, et ses itinéraires sont calculés à base de l'algorithme de Dijkstra du plus court chemin. Ainsi, il évite le risque de tomber dans des boucles de routage car l'algorithme de Dijkstra en est dépourvu.

Cet algorithme n'incorpore aucune information sur la file d'attente, et les chemins sont calculés sans tenir compte de la disponibilité de l'espace de stockage aux nœuds intermédiaires. Ce qui peut conduire à une chute lorsque les buffers sont trop pleins.

Il est aussi caractérisé par le fait qu'il est optimal dans ces deux cas:

Si les nœuds dans le chemin sélectionné n'ont pas de messages enfilés, alors il est optimal parce que la valeur de la fonction Q est égale à 0 partout.

Si les capacités des contacts sont largement suffisantes, il est également optimal (même si les valeurs des files d'attente ne sont pas à 0 car une fois un lien est disponible et qu'il est d'une grande capacité, le temps de transmission de toutes les données enfilées (y compris le message lui-même) est négligeable).

Donc la taille de la file n'affecte pas les délais dans ces deux cas.

Les chemins calculés avec cet algorithme ne prennent pas en compte les délais d'attente.

Si un lien prétend être disponible à un certain temps T , l'algorithme suppose qu'il est possible qu'il soit effectivement utilisé.

Cependant, si beaucoup d'autres messages arrivent dans la file, et le contact devient indisponible avant que ces messages ne soient envoyés, alors la situation est désastreuse, car le temps où le message atteint le prochain nœud est très différent du temps prédit lorsque la route a été pré-calculée. Par conséquent, continuer à calculer les routes antérieurement est loin d'être optimal.

ALGORITHME EARLIEST DELIVERY WITH LOCAL QUERYING(EDLQ)

Dans cet algorithme c'est aussi l'oracle des contacts qui est utilisé et le taux d'occupation de la file d'attente locale est pris en compte dans l'estimation des délais des liens.

La fonction Q compte pour l'enfilement au niveau de tous les liens sortants du nœud source, et aide à éviter la congestion au premier saut. Puis, elle ne compte plus pour les files qui seront rencontrées lorsque le message atteint les autres nœuds du chemin.

Cependant, à la différence avec l'algorithme ED, la route est recalculée à chaque saut (routage par saut). Ceci permet au chemin traversé d'être sensible à l'enfilement dans tous ses liens [16].

L'algorithme EDLQ est caractérisé par le fait que :

La fonction de coût dépend du nœud au niveau duquel la route est calculée. Ce qui peut conduire à la formation de boucles et peut être l'oscillation des messages éternellement.

Certaines oscillations peuvent être évitées en employant un vecteur de trace du chemin et effectuer un nouveau calcul avec des routes fixes (Exemple: calcul en utilisant l'algorithme ED) lorsque la boucle est détectée.

De même que l'algorithme ED, les messages peuvent chuter à cause des dépassements des tampons.

ALGORITHME EARLIEST DELIVERY WITH ALL QUEUES(EDAQ)

Cette approche fait appel aux oracles de connaissances où elle utilise ceux des contacts et de files d'attente.

Il utilise l'oracle de la file d'attente pour déterminer instantanément la taille des files d'attente à travers toute la topologie à n'importe quel point dans le temps [16] :

Q est une fonction dont la valeur retournée est :

$Q(e, t, s)$ = donnée enfilée pour le lien e à l'instant t au nœud s.

Les messages sont routés par la source. Les routes ne sont pas recalculées à chaque saut, car lorsque les routes ont été sélectionnées, la fonction Q a déjà pris en compte l'enfilement dans tous les nœuds sans se soucier de la disponibilité de l'espace dans les buffers.

Après calcul de la meilleure route pour le message, la capacité du lien doit être réservée pour le message sur tous les liens (à des temps donnés) le long du chemin. Ces réservations garantissent le fait que les messages aient été déplacés à des délais suffisants pour éviter de perdre des contacts prévus. En outre, ces réservations permettent de faire des prédictions exactes sur les files d'attente dans le réseau.

La mise en œuvre d'une bande passante de réservations est susceptible d'être un défi important pour les réseaux DTNs, où communiquer avec certains nœuds nécessitent des délais importants.

Pour les systèmes où la centralisation est pratique, l'allocation de la bande passante serait grandement simplifiée.

Les propriétés de cette stratégie sont les suivantes :

- EDAQ détermine une route optimale pour un nouveau message étant donné les réserves existantes pour les messages précédents. Ceci parce qu'il estime correctement les délais d'attente fournis par l'oracle de la file d'attente.

- Les réservations en bande passante sont nécessaires afin d'appliquer fidèlement l'oracle de la file d'attente.

CONCLUSION

Dans les réseaux traditionnels, le routage consiste à trouver le meilleur chemin vers la destination en s'appuyant sur une certaine métrique. Dans les réseaux DTN, la notion de chemin, de topologie, est différente puisque les contacts peuvent être intermittents et les nœuds mobiles. On a une topologie variable dans le temps. Les protocoles de routage traditionnels se révèlent inefficaces.

CHAPITRE 4

INTRODUCTION

La simulation de la communication dans un réseau informatique est considérée comme la représentation du réseau par un graphe où les nœuds sont ses sommets et les arcs les liens qu'il existe entre eux. Les liens supportant le trafic sont intermittents, ils se déconnectent et se reconnectent périodiquement.

L'objectif de ce chapitre est de simuler la communication dans un réseau informatique (cas de routage intermittent). L'idée est de représenter le réseau sous forme de graphe et de matrices et d'y apporter les modifications voulues telles que, l'ajout de nœuds, les liens entre eux, la déduction de la matrice à partir d'un graphe donné ainsi que la création d'un graphe à partir d'une matrice saisie par l'utilisateur.

LE MODELE DTN

Le modèle proposé est le suivant :

Nœuds et liens : le réseau est modélisé sous forme d'un graphe $G = (V, E)$ non orienté tel que :

V : ensemble des nœuds du réseau.

E : ensemble des contacts.

Message : le message est un tuple (u, v, t, m) tel que :

u : nœud source.

v : nœud destination.

t : l'instant où le message est éjecté dans le système.

m : la taille du message.

Stockage : dans notre modèle, nous supposons que les nœuds DTN ont un stockage limité. Le nombre de places dans les files d'attente des nœuds est défini.

Routage et acheminement : pour transférer un message à travers un réseau, il est nécessaire de déterminer quel itinéraire il va suivre (fonction de routage), puis à chaque nœud du réseau, d'aiguiller et de transmettre ce message sur une liaison de données convenable (fonction d'acheminement).

Le calcul du routage nécessite de connaître la topologie du réseau et selon le niveau d'optimisation recherché, une estimation du trafic à acheminer et une expression des coûts respectifs des chemins possibles.

Le résultat du calcul des routes est consigné dans les tables de routage.

ALGORITHME

PRINCIPE DE L'ALGORITHME

Chaque paquet d'information arrivant au niveau d'un nœud attend l'opportunité d'un lien disponible et optimal à entreprendre. Le protocole routé IP doit calculer le prochain lien à parcourir en fonction de la destination du paquet indiquée dans son en-tête et du résultat calculé après lecture des informations de routage contenues dans la table de routage.

Dans le routage habituel, la table de routage conventionnelle suffit. Dans notre cas, cas de routage intermittent, nous avons besoin d'autres informations supplémentaires sur l'état des liens (table des liens, table de durée de connectivité des liens, table de durée de rupture des liens, table de durée de parcours d'un lien, table de temps nécessaires de la première connexion des liens par rapport à la référence temps d'origine (repère temps), a_0).

Aussi, un paquet qui arrive au niveau d'un nœud, peut arriver dans un moment où le lien à entreprendre est dans une période ou état de connexion ou de rupture.

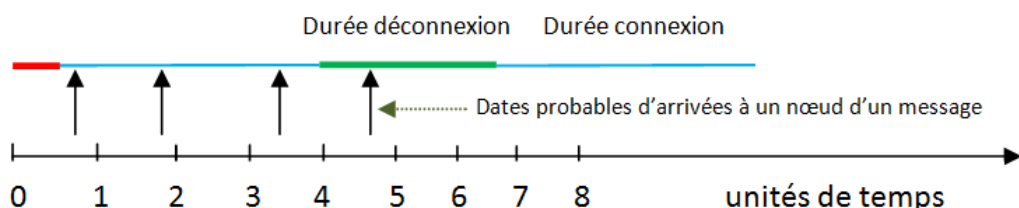


Figure 11: Les instants d'arrivée d'un message

Le paquet arrive au nœud alors que le lien à entreprendre est en période de déconnexion. Dans ce cas, le paquet attend que le lien à entreprendre soit de nouveau reconnecté pour

emprunter ce lien. Donc le temps de traversée d'un lien par un paquet dans ce cas est égal au temps d'intermittence restant pour une nouvelle connexion d'un lien plus le temps nécessaire pour parcourir un lien. (fleche+debut première connexion)

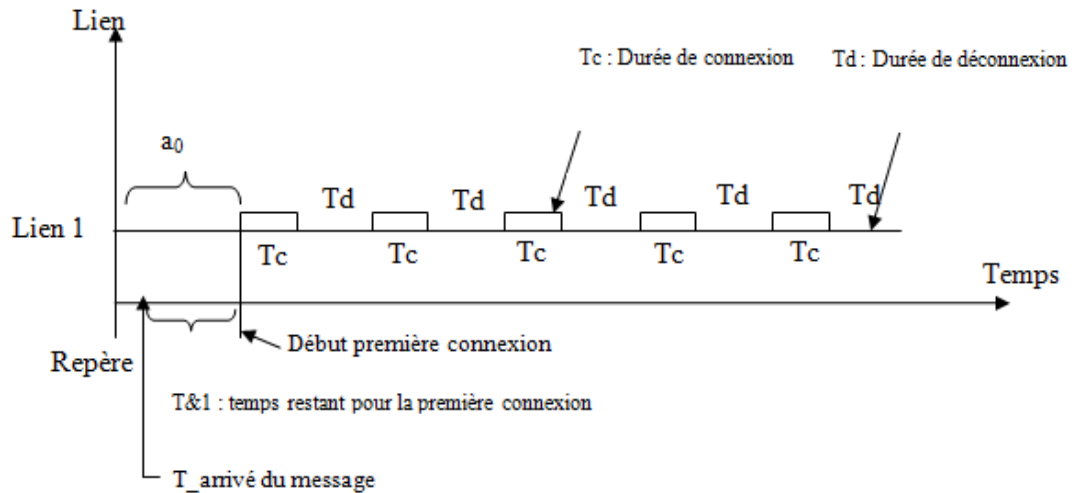


Figure 12: Arrivée du message en période de déconnexion

Le paquet arrive au nœud alors que le lien à entreprendre est en période de connexion, deux cas se présentent : soit le temps restant de connexion est suffisant pour parcourir le lien ou non.

Dans le cas où le temps de parcours restant est insuffisant c'est-à-dire inférieur au temps de parcours, le paquet attend que le temps de connexion restant pour une nouvelle intermittence (ou une déconnexion) du lien en question se termine puis le temps de déconnexion du même lien se termine aussi pour que le message emprunte le lien prévu. Dans ce cas, le temps de traversée d'un lien est égal au temps de connexion restant pour une nouvelle intermittence plus le temps de déconnexion d'un lien plus le temps nécessaire pour parcourir un lien (temps de traversée).

Le temps restant de connexion pour une nouvelle intermittence ou déconnexion du lien est supérieur au temps nécessaire pour parcourir un lien. Dans ce cas, le paquet emprunte directement le lien et le temps de traversée du lien dans ce cas égal au le temps nécessaire pour parcourir le lien.

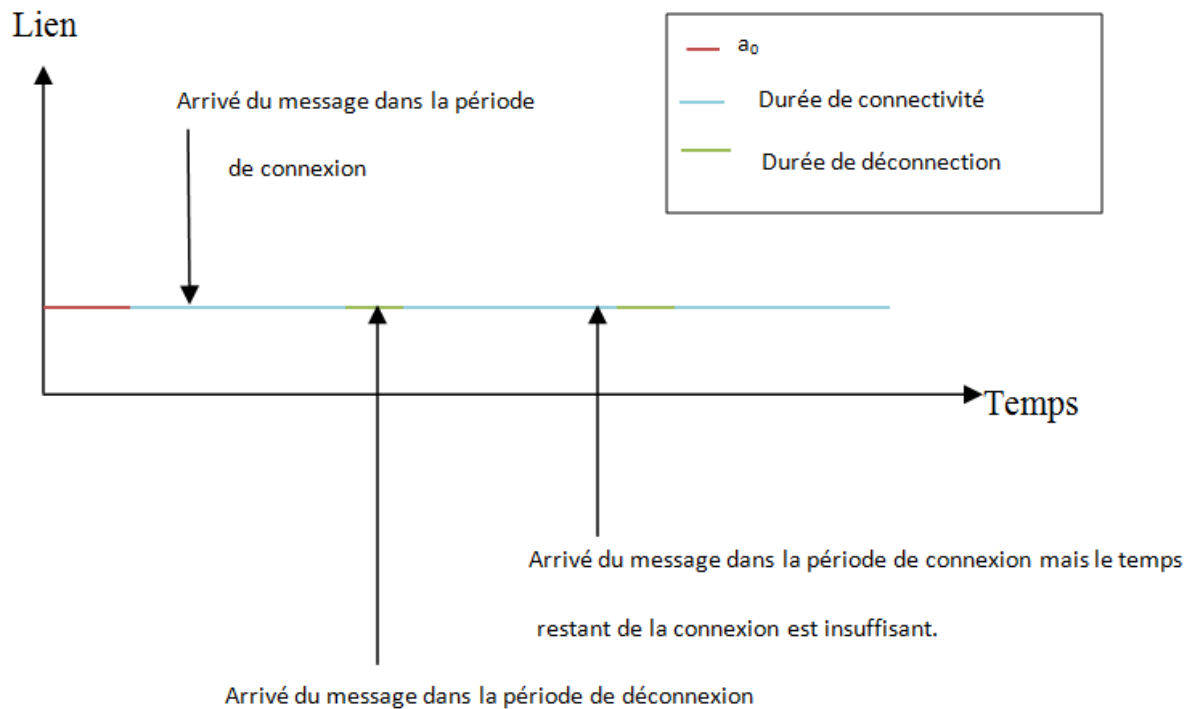


Figure 13: Arrivée de message après de commencement de connexion

LES PSEUDOS ALGORITHMES

Pseudo algorithme 1 : arrivée de message avant de commencement de la connexion.

Variable :

$T(c1c)$: temps commencement premier connexion.

a_0 : temps nécessaire pour la première connexion.

$t_{\text{repère}}$: temps de référence globale.

t_{pi} : temps de parcours du lien L_i .

T_{ti} : temps de traversé du lien L_i .

DEBUT

$T(c1c) = t_{\text{repère}} + a_0 ; S$

 SI ($t_{\text{repère}} \leq T_{\text{ARRIVE DU MESSAGE}} < T(c1c)$) ALORS

$T\&1 = T(c1c) - T_{\text{ARRIVE DU MESSAGE}} ;$

$T_{ti} = T\&1 + t_{pi} ; // \text{TEMPS DE TRAVERSE} = \text{TEMPS RESTANT}$

 POUR LA 1 CONNEXION + TEMPS DU PARCOURS

 FIN SI ;

FIN

Pseudo algorithme 2 : Arrivée de message après de commencement de la connexion.

Variables :

 $t_{\text{repère}}$: temps de référence globale. t_E : temps d'émission du message au nœud N_E (nœud ou le message démarre). a_0 : temps nécessaire pour la première connexion. t_{ci} : temps de connexion d'un lien L_i . t_{di} : temps de déconnexion d'un lien L_i . θ_i : le temps de connexion restant. δ_i : temps de déconnexion restant. $T_{\bar{i}}$: temps de traversée du lien L_i . t_{pi} : temps de parcours du lien L_i . T_G : temps de traversée globale de tous les liens (les segments) du chemin d'un message de la source à la destination. T_R : temps ``reste``.**Hypothèse** : un message démarre d'un nœud S_i à la date t_E . On veut calculer T_G de S_i à S_k .

Début

/* $T_G = \sum_{j=i}^{k-1} T_{t_j}$ */ $T_G=0$ $t = t_E$ Pour $L=1$ à $k-1$ $t_R = [t - (t_{\text{repère}} + a_0)] \text{ MOD } (t_{ci} + t_{di})$ Si $(t_{pi} \leq t_R \leq t_{ci})$ alors /* temps de connexion restant et suffisant pour parcourir un lien. $T_{\bar{i}} = t_{pi}$

Sinon

 Si $((t_R \leq t_{ci}) \text{ et } (t_R \leq t_{pi}))$ alors $\theta_i = (t_{ci} - t_R)$ $T_{\bar{i}} = (\theta_i + t_{di} + t_p)$

Sinon

 Si $(T_{ci} \leq t_R < t_{ci} + t_{di})$ alors $\delta_i = (t_{ci} + t_{di}) - t_R$ $T_{\bar{i}} = (\delta_i + t_p)$

Fin si

 $T_G = T_G + T_{\bar{i}}$ $t = t_E + T_{\bar{i}}$

Fin pour

Fin.

EXEMPLE DE L'APPLICATION

Les différentes matrices utilisées

Matrice à état de liens : c'est une matrice représentant le graphe du réseau. Elle est construite comme suite :

La diagonale prend la valeur ∞ (pas de boucle dans le graphe).

Les autres cases prennent la valeur 1 s'il existe un lien et la valeur 0 sinon.

A Partir de cette matrice on génère les matrices suivantes :

- Matrice représentant le temps nécessaire de la première connexion entre deux sommets donnés.
- Matrice représentant la durée de connectivité périodique des liens.
- Matrice représentant la durée de déconnexion ou de rupture périodique des liens.
- Matrice représentant le temps de parcours des liens.

Ces matrices sont toutes de la même dimension n qui représente le nombre de nœuds de notre réseau.

-Exemple de graphe d'étude à cinq sommets.

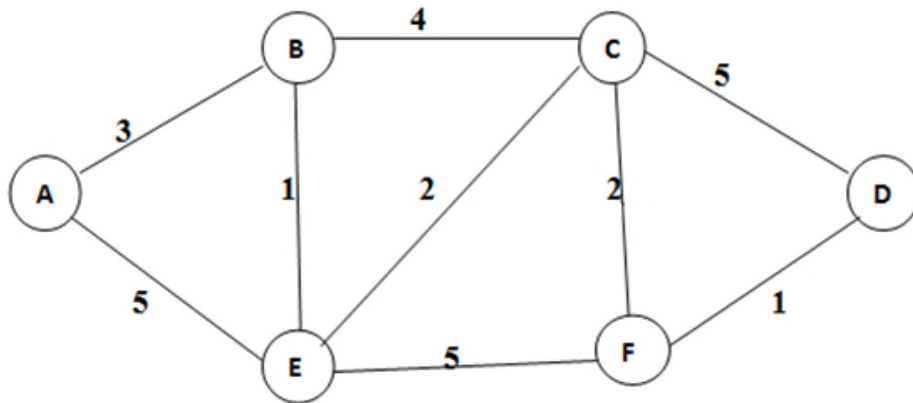


Figure 14: Exemple de réseau de communication

La matrice d'adjacence correspondante :

	A	B	C	D	E	F
A	-	1	0	0	1	0
B	1	-	1	0	1	0
C	0	1	-	1	1	1
D	0	0	1	-	0	1
E	1	1	1	0	-	1
F	0	0	1	1	1	-

Matrice avant connexion : représentant les durées a_0 ou des temps nécessaires de la première connexion des liens par rapport à la référence temps d'origine (repère temps). Prévoir toutes les modifications automatiques qui seront déclenchées à chaque fois qu'une modification (modification du réseau par l'ajout ou suppression d'un nœud ou par l'ajout ou suppression d'un lien) est effectuée sur le réseau (graphe). Prévoir toutes les opérations usuelles sur les temps nécessaires a_0 de la première connexion des liens par rapport à la référence temps d'origine (repère temps).

	A	B	C	D	E	F
A	-	-	-	8	2	-
B	-	-	9	-	8	-
C	-	9	-	6	-	-
D	8	-	6	-	-	1
E	2	8	-	-	-	-
F	-	-	-	1	-	-

Matrice de connexion : représentant les durées de connectivité périodiques des liens et toutes les modifications automatiques qui seront déclenchées à chaque fois qu'une modification (modification du réseau par l'ajout ou suppression d'un nœud ou par l'ajout ou suppression d'un lien) est effectuée sur le réseau (graphe) ainsi toutes les opérations usuelles sur les durées de connectivité des liens.

	A	B	C	D	E	F
A	-	-	-	71	78	-
B	-	-	89	-	68	-
C	-	89	-	71	-	-
D	71	-	71	-	-	23
E	78	68	-	-	-	-
F	-	-	-	23	-	-

Matrice de déconnexion : représentant les durées de déconnexion ou de rupture périodiques des liens et toutes les modifications automatiques qui seront déclenchées à chaque fois qu'une modification (modification du réseau par l'ajout ou suppression d'un nœud ou par l'ajout ou suppression d'un lien) est effectuée sur le réseau (graphe) ainsi Prévoir toutes les opérations usuelles sur les durées de déconnexion ou de rupture périodiques des liens.

	A	B	C	D	E	F
A	-	-	-	6	6	-
B	-	-	5	-	2	-
C	-	5	-	2	-	-
D	6	-	0	-	-	5
E	6	2	-	-	-	-
F	-	-	-	5	-	-

Matrice de temps de parcours : représentant les durées de parcours des liens et toutes les modifications automatiques qui seront déclenchées à chaque fois qu'une modification (modification du réseau par l'ajout ou suppression d'un nœud ou par l'ajout ou suppression d'un lien) est effectuée sur le réseau (graphe) ainsi toutes les opérations usuelles sur les durées de parcours des liens.

	A	B	C	D	E	F
A	-	-	-	1	0	-
B	-	-	3	-	5	-
C	-	3	-	2	-	-
D	1	-	2	-	-	9
E	0	5	-	-	-	-
F	-	-	-	9	-	-

Calcul de plus court chemin :

Les informations nécessaires au calcul du routage étant connues soit à un nœud central soit localement, il convient de calculer le chemin optimal.

A chaque lien est effectuée une métrique (un cout). La distance d'un nœud à la destination finale est la somme des longueurs des liens constituant le chemin.

Plusieurs types d'algorithmes permettent d'effectuer ce calcul (Dijkstra , Belman Ford ...).

On choisi donc une métrique puis un algorithme d'optimisation.

Une métrique de cout prend en compte le cout de fonctionnement d'une liaison pour le calcul du chemin optimale.

-Exemple de calcul le plus court chemin :

On étudie le réseau qu'illustrer dans la figure 14:

L'état initial est le suivant :

Etat (A) = [(B, 3) ; (E, 5)].

Etat (B) = [(A, 3) ; (C, 4) ; (E, 1)].

Etat (C) = [(B, 4) ; (D, 5) ; (E, 2) ; (F, 2)].

Etat (D) = [(C, 5) ; (F, 1)].

Etat (E) = [(A, 5) ; (B, 1) ; (C, 2) ; (F, 5)].

Etat (F) = [(C, 2) ; (D, 1) ; (E, 5)].

Les routeurs diffusent (de manière contrôlée) leurs informations à l'intérieur du SA.

Chacun peut reconstituer la cartographie du système, la matrice état des liens illustrer dans la figure :

Etat	A	B	C	D	E	F
A	0	3	X	X	5	X
B	3	0	4	X	1	X
C	X	4	0	5	2	2
D	X	X	5	0	X	1
E	5	1	2	X	0	5
F	X	X	2	1	5	0

Tableau 1:Matrice état des liens

Dans cette matrice symétrique, les zéros sur la diagonale indiquent un coût nul ; les X repèrent les liaisons inexistantes.

Calculons la table de routage du routeur A:

A connaît deux routes AB de coût 3 et AE de coût 5. Il place la route AE en attente (5 est moins bon que 3) et valide la route AB à partir de laquelle il explore les chemins passant par B.

A	A, B, 3	X	X	A, E, 5	X
---	---------	---	---	---------	---

La route BC est de coût 4, donc AC sera de coût $3 + 4 = 7$. La destination C n'était pas encore connue ; la route est ajoutée dans la table.

La route BE est de coût 1, donc AE sera de coût $3 + 1 = 4$ ce qui est meilleur que la route actuellement connue. La table est mise à jour.

A	A, B, 3	A, C, 7	X	A, E, 4	X
---	---------	---------	---	---------	---

Le routeur A place la route AC en attente (7 est moins bon que 4) et valide maintenant la route AE à partir de laquelle il explore les chemins passant par E.

La route EC est de coût 2, donc AC sera de coût $4 + 2 = 6$ ce qui est meilleur que la route actuellement connue. La table est mise à jour.

La route EF est de coût 5, donc AF sera de coût $4 + 5 = 9$. La destination F n'était pas encore connue; la route est ajoutée dans la table.

A	A, B, 3	A, C, 6	X	A, E, 4	A, F, 9
---	---------	---------	---	---------	---------

Le routeur A place la route AF en attente (9 est moins bon que 6) et valide maintenant la route AC à partir de laquelle il explore les chemins passant par C.

La route CD est de coût 5, donc AD sera de coût $6 + 5 = 11$. La destination D n'était pas encore connue; la route est ajoutée dans la table.

La route CF est de coût 2, donc AF sera de coût $6 + 2 = 8$ ce qui est meilleur que la route actuellement connue. La table est mise à jour.

A	A, B, 3	A, C, 6	A, D, 11	A, E, 4	A, F, 8
---	---------	---------	----------	---------	---------

Le routeur A place la route AD en attente (11 est moins bon que 8) et valide maintenant la route AF.

La route FD est de coût 1, donc AD sera de coût $8 + 1 = 9$. Ce qui est meilleur que la route actuellement connue. La table est mise à jour.

A	A, B, 3	A, C, 6	A, D, 9	A, E, 4	A, F, 8
---	---------	---------	---------	---------	---------

La route AD est validée à son tour.

L'algorithme est terminé.

La table de routage de A est alors la suivante :

Routage (A) = [(A, 0) ; (B, 3) ; (C, 6) ; (D, 9) ; (E, 4) ; (F, 8)].

-Le lien entre C et E tombe en panne. Comme montrer dans la figure suivant :

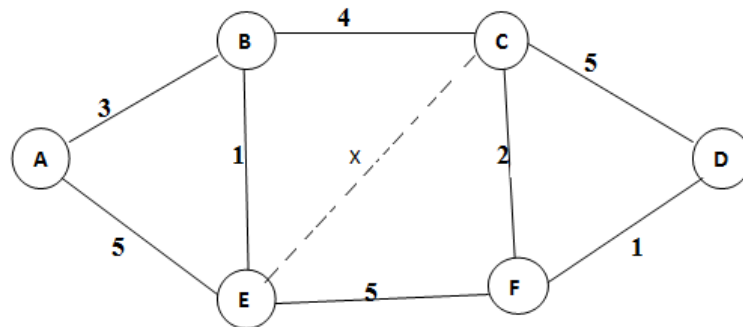


Figure 15: Réseau en cas d'existence de panne

On suppose que la panne existe entre le routeur C et E (le lien C et E est déconnecté).

Les routeurs C et E diffusent l'information dans le système. Le tableau suivant illustre l'état de lien de ce réseau :

Etat	A	B	C	D	E	F
A	0	3	X	X	5	X
B	3	0	4	X	1	X
C	X	4	0	5	X	2
D	X	X	5	0	X	1
E	5	1	X	X	0	5
F	X	X	2	1	5	0

Tableau 2:Matrice état des liens en cas de panne

L'application de l'algorithme par le routeur A est la même que précédemment jusqu'à l'étape qui est représenté dans cette table de routage.

A	A, B, 3	A, C, 7	X	A, E, 4	X
---	---------	---------	---	---------	---

Le routeur A place la route AC en attente (7 est moins bon que 4) et valide maintenant la route AE à partir de laquelle il explore les chemins passant par E.

La route EF est de coût 5, donc AF sera de coût $4 + 5 = 9$. La destination F est ajoutée.

A	A, B, 3	A, C, 7	X	A, E, 4	A, F, 9
---	---------	---------	---	---------	---------

Le routeur A place la route AF en attente et valide la route AC à partir de laquelle il explore les chemins passant par C. On obtient :

A	A, B, 3	A, C, 7	A, D, 12	A, E, 4	A, F, 9
---	---------	---------	----------	---------	---------

Et enfin :

A	A, B, 3	A, C, 7	A, D, 10	A, E, 4	A, F, 9
---	---------	---------	----------	---------	---------

La nouvelle table de routage de A est maintenant :

Routage (A) = [(A, 0) ; (B, 3) ; (C, 7,) ; (D, 10, E) ; (E, 4, B) ; (F, 9, E)].

CONCLUSION

Dans ce travail nous avons réalisé une application pouvant créer une matrice d'adjacence (état de liens) d'un graphe non orienté (réseau) et d'autres matrices (temps de connexion, déconnexion, parcours et matrice a0) à partir de celle ci.

CONCLUSION GENERALE

Nous avons fait connaître à travers ce mémoire les réseaux tolérants aux délais (DTN: Delay Tolérant Network), ces réseaux ont le potentiel de connecter des environnements hétérogènes.

Dans un premier temps, nous avons présenté l'environnement des DTNs, les raisons qui ont mené à leur apparition ainsi que la nouveauté avec laquelle ils arrivent. Nous avons détaillé la couche protocolaire, et expliqué le principe sur lequel est fondé le protocole bundle. Tout cela a été fait en se basant principalement sur des documents fournis par le groupe de recherche sur les DTNs, le DTNRG.

Dans les DTN, une connexion bout en bout peut ne jamais exister. Pour que la communication soit possible, les nœuds intermédiaires prennent la garde des données « Store » et les retransmettent « Forward » quand l'opportunité se présente. Les nœuds et les liens peuvent être inaccessibles pour de longues durées. Le défi dans un tel réseau est de transférer les données sans avoir une connexion de bout en bout.

Le routage est l'un des défis les plus intéressants des DTN. Il exige des techniques pour sélectionner les chemins, programmer les transmissions et gérer le stockage dans les buffers des nœuds.

En plus, nous avons connu le mode de fonctionnement des DTN et tout ce qui s'y réfère, à savoir leurs architectures, les raisons qui ont amenées à leur apparitions, comme expliqué le principe des deux techniques de commutation de messages : « Store And Forward » et « le transfert de garde », le problème de congestion généré par le store and forward et la capacité limitée des buffers ainsi que le mécanisme de sécurité implémentés dans un DTN.

Ensuite nous avons introduit les étapes de routage dans DTN ainsi les concepts de routage selon le routage proactif, réactif, source, par saut et hiérarchique, aussi la classification des protocoles de routage DTN selon l'inondation, l'expédition, les oracles de connaissance. Enfin nous avons détaillé les algorithmes de routage comme MED, ED, EDLQ, EDAQ.

En effet, pour notre proposition nous avons illustré dans ce mémoire un algorithme de recherche de chemin optimale dans un réseau DTN.

PERSPECTIVES

Afin de compléter notre travail, de nombreuses perspectives sont envisageables :

Analyser et évaluer notre approche.

Pouvoir situer notre stratégie entre toutes celles qui existent.

LISTE DES ABREVIATIONS

A**ACK** Acquittement**ADSL** Asymmetric Digital Subscriber Line**AODV** Ad hoc On Demand Distance Vector**ARQ** Automatic Repeat Request est une méthode de correction des erreurs de transmission, qui consiste – lorsqu’une erreur dans un message est détectée à demander à l’émetteur de transmettre le message touché.**AS** Autonomous System**B****BGP** Border Gateway Protocol**C****CA** Certification Authority**CIDR** Classless Inter Domain Routing**CPU** Central Processing Unit**D****DNS** Domain Name System**DSDV** Destination-Sequenced Distance-Vector**DSR** Dynamic Source Routing**DTN** Delay Tolerant Network**DTNRG** Delay Tolerant Network Research Group**E****ED** Earliest Delivery**EDAQ** Earliest Delivery with All Queues**EDLQ** Earliest Delivery with Local Querying**EGP** Exterior Gateway Protocol**F****FAI** Fournisseur d'Accès Internet**I****ICMP** Internet Control Message Protocol**IGP** Interior Gateway Protocol**IP** Internet Protocol**M****MED** Minimum Expected Delay

N

NIC Network Information Center

O

OLSR Optimized Link State Routing

OSPF Open Shortest Path First

R

RIP Routing Information Protocol

RTT Round Trip Time

S

SPF Shortest Path First

SYN Synchronisation

T

TCP Transmission Control Protocol

TTA Time-To-Acknowledge

TTL Time To Live

U

UDP User Datagram Protocol

URL Uniform Resource Locator

REFERENCES BIBLIOGRAPHIQUES

- [1] TSAFACK CHETSA Ghislain Landry. « Émulation de réseaux interplanétaires et validation de protocoles tolérants aux délais »
- [2] PHILIPPE ATELIN, « Réseaux informatiques », Notions fondamentales, Normes, Architectures, Modèles OSI, TCP/IP, Ethernet, Wi-Fi, ...
- [3] STÉPHANE LOHIER ,AURÉLIE QUIDELLEUR « Le Réseau Internet » Des Services aux Infrastructures
- [4] archives.reseaucerta.org/cotecours/routdyn/annexes.htm.
- [5] GUY PUJOLLE , « Les Réseaux »
- [6] depinfo.u-cergy.fr/~dntt/supports/.../cours-B1.routage-dynamique.pdf.
- [7] F. Warthman “ Delay Tolerant Network (DTNs) ” A tutorial. Forest arthman.Warthman Associates (www.warthman.com). Version 1.1. 2003.
- [8] A.Hooke L. Torgerson R. Durst K. Scott. K. Fall V, S. Burleigh and H. Weiss. Delay-Tolerant Network Architecture NY, March-Septembre 2003
- [9] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall and H. Weiss “Delay-Tolerant Network Architecture” March-September 2003
- [10] S. Burleigh, E. Jennings, J. Schoolcraft “Autonomous Congestion Control in Delay-Tolerant Networks”, American Institute of Aeronautics and astronautics, june 2006.
- [11] P. Mundur, S. Lee, M. Seligman “Routing in Delay Tolerant Networks Using Storage Domains” November 2006. UMD Tech Report UMIACS-TR-2007-01 <http://hdl.handle.net/1903/4024>
- [12] M. Choon Chan, R. Ramjee “TCP/IP Performance over 3G Wireless Links with Rate and Delay Variation”, Springer Science + Business Media, Inc. Manufactured in The Netherlands, 2005.
- [13] C. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. ACM SIGCOMM Computer Communication Review, 24(4) :234–244, 1994.
- [14] D. Johnson and D. Maltz. Dynamic source routing in ad hoc wireless networks. Mobile computing, pages 153–181, 1996
- [15] C. L. and J. Wu “Scalable Routing in Delay Tolerant Networks” MobiHoc'07, September 9-14, 2007.
- [16] S. Jain, K. Fall, R. Petra “Routing in a delay tolerant network”, In Proc. ACM SIGCOMM'04, Portland, OR, August 2004.
- [17] M. K. Marina and S. R. Das. On-demand Multipath Distance Vector Routing in Ad Hoc Networks. In IEEE ICNP, Nov. 2001