

# **Rutare dinamica folosind protocoale de tip link-state in retele ad-hoc**

**Studenti: Ene Andrei Bogdan**

**Popescu Andrei Alexandru**

**Grupa: 441A**

**Disciplina: Retele de Calculatoare**

**Profesor coordonator : Stefan Stancescu**

# Cuprins

1. Introducere (Ene Andrei Bogdan)
2. Rutare dinamica (Ene Andrei Bogdan)
  - 2.1 Evolutie
  - 2.2 Caracteristici
  - 2.3 Link State
  - 2.4 Algoritm de functionare
3. OLSR (Popescu Andrei Alexandru)  
Functionarea protocolului
  - 3.1 Detectarea vecinilor
  - 3.2 Mesaje HELLO
  - 3.3 Mesaje TC
  - 3.4 Tabelul de rutare
  - 3.5 Analiza performantei
  - 3.6 Concluzii

# Rutare dinamica folosind protocoale de tip link-state in retele ad-hoc

## **1. Introducere**

O data cu avansul tehnologic a crescut si cererea pentru flexibilitate astfel utilizarea retelelor mobile wireless a crescut foarte mult. Prin aceasta crestere , natural a crescut si dimensiunea retelei (de la 10 pana la 100 de noduri).Pe masura ce retea se intinde pe o arie mai mare decat aria de acoperire a fiecarui nod este necesara o tehnica de rutare pentru ca nodurile care nu se afla in aria de acoperire sa poata comunica intre ele. Astfel trebuie propusa o solutie a problemei , un protocol de rutare. Exista insa problem in creerea unui astfel de protocol , intrucat diferenta dintre retele statice legate prin fire si retele mobile care comunica wireless este foarte mare , retelele wireless fiind mult mai complexe si punanc mulm mai multe probleme. Cele mai mari problem in retelele mobile (ad hoc) sunt :

- latime de banda limitata
- schimbari topologice dese

Astfel telul unui astfel de protocol de rutare ar fi sa aiba cat mai putin traffic de control pentru a utilize putina latime de banda , dar in acelasi timp sa se identifice rapid link-failurile cauzate de modificarea pozitiiei nodului. De asemenea a o alta problema este creerea de bucle de rutare , topologia fiind intr-o continua miscare\modificare. Legaturile unidirectionale reprezinta o alta problema creata de acest gen de retele.

Exista 2 metode de rutare: proactiva si reactiva:

- in metoda reactiva prodocolul de rutare nu preia initiative pentru a gasi o ruta catre destinatie pana cand nu i se cere acest lucru si incearca sa descopere rutele prin flooding

- in metoda proactiva se schimba mesaje de control periodic ; unele mesaje se trimit doar local iar altele se trimit in toata retea , acestea din urma contin informatii despre topologia completa a retelei. Protocoalele proactive furnizeaza rute foarte repede , dar dezavantajul ar fi cresterea traficului de control , implicit un grad de ocupare mai mare al latimii de banda.

Exista si combinatii ale celor 2 metode , acestea tin rute pentru unele destinatii active tot timpul prin eschimbul periodic de mesaje , dar catre alte destinatii acestea descopera rutele doar in momentul in care acestea sunt cerute.

## **2. Rutare Dinamica**

### **2.1 Evolutie**

*Protocoalele de rutare dinamică (adaptivă)* au fost utilizate în rețele de calculatoare de la începutul anilor 1980. Prima versiune a protocolului de rutare dinamică RIP a fost lansat în 1982, dar unele elemente din algoritmi de bază a acestui protocol au fost utilizate pe ARPANET încă din 1969.

Unul dintre cele mai vechi protocoale de rutare a fost Routing Information Protocol (RIP) care a evoluat într-o versiune mai nouă RIPv2. Cu toate acestea , versiunea nouă nu este scalabilă pentru a fi implementată în rețele mari. Pentru satisfacerea acestor cerințe au fost create alte două protocoale avansate de rutare:

- Open Shortest Path First(OSPF)
- Intermediate System-to-Intermediate-System(IS-IS).

Dupa aceste două protocoale, Cisco a dezvoltat Interior Gateway Routing Protocol(IGRP) și Enhanced IGRP(EIGRP) care de asemenea pot fi implementate în rețele de scară largă.

Odata cu aparitia a mai multor dispozitive care au nevoie de o adresa IP,spatial de adrese IPv4 este aproape epuizat. Astfel,a aparut IPv6. Pentru a sprijini comunicarea bazata pe IPv6 , au aparut noi versiuni de protocoale de rutare.

## 2.2 Caracteristici

- ✓ Timpul de convergenta→defineste cat de repede routerele din topologia de retea partajeaza informatia de rutare si ajung la o stare de constinte coerenta despre topologia retelei. Cu cat este mai rapida convergenta , cu atat este mai preferabil protocolul.
- ✓ Scalabilitatea- defineste marimea retelei bazata pe protocolul de rutare implementat. Cu cat este mai mare reteaua , cu atat trebuie sa fie mai scalabil protocolul de rutare.
- ✓ Classless si Classfull – Protocoalele de rutare de tip classless includ masca de retea in mesajele de actualizare. Aceasta caracteristica suporta optiunea Variable Length Subnet Masking(VLSM) si o sumarizare mai optimala a rutelor.Protocoalele de rutare classfull nu includ masca de subretea si nu suporta VLSM,
- ✓ Utilizarea resurselor – include cerintele unui protocol de rutare cum ar fi spatial de memorie,utilizarea procesorului si link-ul de utilizare a latimii de banda.
- ✓ Implementarea si intretinerea descrie nivelul de cunostinte care sunt necesare pentru un administrator de retea pentru a pune in aplicare si a intretine reteaua bazata pe protocolul de rutare utilizat.

## 2.3 Link State

Protocolul de rutare link-state abordeaza altfel aceasta problema:el ia in calcul starea legaturilor,alcatuind o harta a topologiei.

Protocoalele Link State sunt cunoscute si ca Cea Mai Scurta Cale Intai(Shortest Path First) si sunt implementate in jurul algoritmului Dijkstra. Ele sunt mai complicate decat protocoalele Distance Vector. Cu toate acestea,functionalitatea si configurarea acestora nu este complexa.

Protocoalele link-state au urmatoarele caracteristici:

- raspund rapid la schimbarile generate in cadrul retelelor in care sunt utilizate ;

- trimit update-uri "triggered" cand se genereaza schimbari in topologia retelei;

- trimit update-uri periodice,numite si link-state refresh,la interval de timp lungi de obicei 30 de minute.

Structura de date tipice link-state:

- neighbor table este lista ruterelor vecine cunoscute;

- topology table →toate ruterile dintr-o zona au tabela topology table identica

- routing table(lista cu cele mai bune cai catre destinatie)

### 3. Algoritm de functionare

Algoritmul de routare SPF este fundamentul operatiilor OSPF. Când un router SPF este pornit, se inițializează structurile de date ale protocolului de routare și se așteaptă confirmarea din partea protocoalelor de pe nivelele de mai de jos, că interfețele sale funcționează.

După ce un router se asigură că interfețele sale funcționează, folosește protocolul OSPF hello pentru a-și descoperi vecinii. Pachetele hello mai sunt folosite și ca pachete keepalive (se determină dacă mai este funcțională legătura).

În rețele multiacces (multiaccess networks - care suportă mai mult de două routere), protocolul hello alege un router pentru transmisie (designated router) și un router pentru transmisii în cazul excepțiilor (backup designated router). Un designated router este responsabil cu generarea mesajelor update pentru întreaga rețea multiacces. Designated routers permit reducerea traficului în rețea și reducerea mărimii bazelor de date despre topologie.

Când bazele de date link-state ale două routere vecine sunt sincronizate se spune că routerele sunt adiacente (adjacent). În rețelele multiacces routerul designated router determină care routere ar trebui să devină adiacente. Adiacența controlează distribuția de pachete ale protocolului de routare, fiind trimise și primite pachete doar pe baza adiacenței.

Fiecare router trimite periodic mesaje update, pentru a furniza informații despre adiacențele unui router oarecare sau pentru a-i informa pe ceilalți când starea unui router se schimbă. Comparând adiacențele stabilite cu starea legăturilor, routerele căzute pot fi detectate rapid și topologia rețelei poate fi schimbată să reflecte acest lucru. Din bazele de date despre topologie generate de mesajele update, fiecare router calculează un arbore cu rutele cele mai scurte (shortest-path tree) în care el însuși este rădăcină. Apoi, arborele shortest-path devine tabelă de routare.

## Algoritm OSPF

Stabilirea vecinilor- înainte ca un router să inunde cu informații privind starea legăturilor sale celelalte routere, el trebuie să determine dacă există vreun vecin conectat la vreuna din legăturile sale.

Înainte ca două routere să își formeze adiacența, ele trebuie să se pună de acord în privința a trei valori: intervalul de trimitere al pachetelor Hello, intervalul de trimitere al pachetelor Dead și tipul rețelei. În cele mai multe cazuri, pachetele Hello sunt trimise multicast către o adresă rezervată pentru toate routerele OSPF : 224.0.0.5. Utilizarea o adresă multicast, permite unui dispozitiv să ignore pachetul pe interfața dacă aceasta nu este setată să accepte pachete OSPF.

Intervalul Dead este perioada, exprimata in secunde, in care routerul va astepta sa primeasca un nou pachet Hello inainte sa declare vecinul ca fiind „picat”. Ca valoare implicita, intervalul Dead este de 120 de secunde.

Daca intervalul Dead expira inainte ca routerul sa primeasca un pachet Hello, OSPF va elimina acel vecin din baza de date. Routerul va trimite informatia cu starea legaturilor despre vecinul „picat” pe toate interfetele OSPF activate.

#### 4. OLSR (Optimized Link State Routing) protocol

Pentru solutionarea problemei rutarii dinamice in retele ad hoc propunem un protocol proactive de rutare numit OLSR. Datorita naturii sale proactive are avantajul de a avea rutele necesare imediat la cerere. OLSR este o variant optimizata a unui protocol pur link-state pentru retele ad hoc. Pentru inceput a fost redusa dimensiunea pachetelor de control: in loc sa transmita informatiile pentru toate nodurile, se dau informatii decat pentru cateva noduri numite relee selectoare multipunct (multipoint relay selectors). In al doile rand minimizeaza floodingul prin faptul ca foloseste doar acele noduri selectate numite relee multipunct (multipoint relays) pentru a difuza mesajele de control in retea. Doar releele multipunct retransmit mesajele de broadcast. Aceasta tehnica reduce semnificativ numarul de retransmisii in cazul floodingului sau a broadcastului.

In afara de mesajele de control periodice, protocolul nu genereaza alt tip de traffic in cazul unor link-failures.

Protocolul functioneaza in mod individual si nu are nevoie de o entitate centrala. Protocolul nu necesita o conexiune sigura pentru transmiterea mesajelor de control, fiecare nod isi transmite mesajele de control periodic, deci se poate trece peste pierderea unor pachete (lucru foarte normal in transmisa radio).

Protocolul OLSR face o ruteare hop cu hop (hop by hop routing): fiecare nod foloseste cea mai recenta informative din tabela de rutare pentru a ruta un

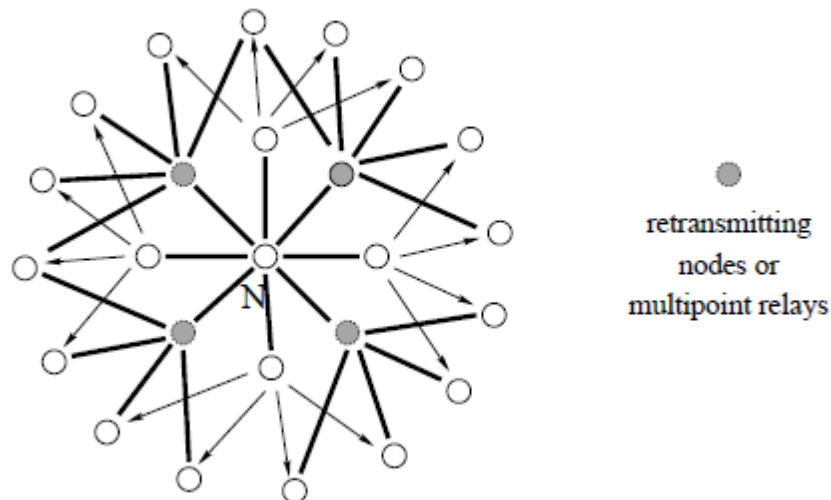


pachet. Astfel în momentul în care un nod își schimbă poziția pachetele sale pot fi transmise la el dacă mutarea s-a produs cu o viteză pe care nodul vecin o poate urmări.

### Releele multipunct

Acestea se folosesc pentru a minimiza traficul floodat prin broadcast în rețea prin reducerea retransmisiei aceluiași informații în aceeași arie. Fiecare nod din rețea își alege câteva noduri din vecinătate, care îi vor retransmite pachetele. Acest set de noduri se numesc relee multipunct. Nodurile vecine care nu sunt relee multipunct primesc și procesează pachetele de control dar nu le retransmit. În acest scop fiecare nod își alege câțiva vecini pe care îi vom numi selectori de relee multipunct (MPR Selectors). Orice mesaj broadcast care vine de la acești selectori MPR ai nodului se presupune că sunt retransmiși de acel nod.

Fiecare nod își alege setul MPR din raza de un hop astfel încât să acopere toate nodurile până la o distanță de 2 hopuri.



Relee multipunct (MPR)

Protocolul OLSR calculeaza cu ajutorul MPR rutele catre toate destinatiile cunoscute, nodurile MPR fiind noduri intermediare in drum. Pentru a fi posibil acest lucru fiecare nod din retea retransmite prin broadcast periodic informatiile despre vecinii distantati la un hop nodurilor sale MPR corespunzatoare.

Functionarea protocolului:

### 1) Detectarea vecinilor

Fiecare nod trebuie sa isi gaseasca vecinii cu care are o legatura directa si bidirectionala. Datorita trnasmisiunii radio propagarea legatura poate deveni unidirectional astfel toate legaturile trebuiesc verificate in ambele directii pentru a putea fi considerate valide.

Pentru aceasta fiecare nod transmite prin broadcast periodic mesaje de HELLO care contin informatii despre vecini si despre starea legaturii (link status). Aceste mesaje sunt primite de catre toate nodurile vecine insa nu vor fi retransmise.

Mesajul HELLO contine :

- O lista a vecinilor cu legaturi bidirectionale valide
- Lista de noduri la care nu se stie daca legatura este bidirectionala

### 2) Mesajele de HELLO permit fiecarui nod sa isi cunoasca vecinii pana la o distant de 2 hopuri. Pe baza acestor mesaje fiecare nod isi alege nodurile MPR. Aceste noduri apoi sunt indicate in mesajele HELLO prin link-statusul MPR (tag). Tot pe baza mesajelor de hello se aleg si selectorii MPR.

Setul MPR este calculat in asa fel incat sa contina un subset de vecini care sa acopere toti ceilalti vecini de pana la 2 hopuri. Setul MPR nu trebuie sa fie cel optim, insa trebuie sa fie destul de mic pentru a fi benefic conceptului MPR.

3) Pentru a forwarda informatiile cu despre nodurile MPR nodurile transmit mesaje de control specific numite mesaje TC. Aceste mesaje sunt transmise in toata retea. Intervalul de transmisie a 2 mesaje TC depinde daca selectorul MPR s-a schimbat sau nu de la transmiterea ultimului mesaj TC.

Fiecare nod al retelei mentine un table topologic care contine informatii din mesajele TC. Un nod inregistreaza informatiile despre nodurile MPR ale celorlalte noduri din retea. Bazandu-se pe aceste informatii se calculeaza tabela de rutare.

4) Tabelul de rutare

Fiecare nod retine tabelul de rutare pentru toata retea pentru a forwarda pachete catre destinatii. Toate nodurile care primesc mesaje TC salveaza o parte din acele informatii pentru a crea perechi interconectate de tipul [ultimul hop, nod] unde nod sunt informatii scoase din mesajele TC. Pentru a utilize doar rutele optime nodurile vor selecta din mesajele TC doar rutele minime de tipul [ultimul hop , nod]. Informatiile din tabelul de rutare arata in felul urmator : adresa de destinatie , adresa urmatorului hop si distant estimate pana la destinatie. Orice ruta doar partial cunoscuta nu va fi introdusa in tabel.

5) Analiza performantei

Ruta optima

Principala problem ape care dorim sa o analizam este daca introducerea seturilor MPR ca subseturi alea listelor vecinilor nu distrug proprietatile de conectivitate ale retelei.

Prin definitie validitatea unui link este confirmata de mesajele de HELLO care sunt retransmise periodic de catre seturile MPR corespunzatoare , neaparand niciun fel de problema doar in cazul in care nodurile mobile se deplaseaza mai repede decat trecerea unui interval de HELLO. Astfel avand un set de cel putin 2 noduri MPR care acopera vecinii de pana la o distanta de 2 hopuri putem spune ca am eliminate problema miscarii nodurilor.

O ultima problema este broadcastul informatiilor despre topologia retelei. Daca setul de noduri MPR coincide cu setul de noduri vecine atunci informatia va ajunge fara probleme in toata retea. In cazul in care drumul minim catre un nod distant este continut de mesajul TC transmis acesta va fi adaugat in tabela de rutare aferenta. Astfel putem concluziona ca aceasta proprietate se conserva in cazul in care seturile de noduri MPR coincide cu subsetul de vecini.

## 6) Concluzii

Pentru retele mobile wireless , performanta unui protocol de rutare este legata de mai multi factori , cum ar fi alegerea tehnologiei fizice , comportamentul layerului de link , etc. Comportamentul general al unui protocol specifica domeniul in care ar putea fi folosit. Protocolul OLSR este un protocol de natura proactiva , astfel acesta favorizeaza acest mod de retelistica unde informatia este necesara din ce in ce mai des, nu exista timpi morti , unde se cer foarte frecvent informatii despre ruta sau se cere calcularea unor rute noi. Acest protocol de asemenea favorizeaza aplicatiile care nu permit o intarziere mare intre transmiterea pachetelor. OLSR este un protocol adaptat pentru retele dense sau retele aglomerate , unde se presupune ca comunicarea are loc frecvent si de asemenea are loc intre un numar mare de noduri.

## **Bibliografie**

1. ETSI STC-RES10 Comitee. Radio Equipment and Systems:High performance Radio Local Area Network Type 1, june 1996
2. J.Garcia-Luna-Aceves and M.Spohn Source-Tree Routing in Wireless Networks, November 1999
3. P.Jacquet and A.Laouiti. Analysis of Mobile Ad-Hoc Network Routing Protocols ,INRIA 2000
4. Philippe Jacquet,Pascale Minet,Paul Muhlethaler, and Nicolas Rivierre. Increasing reliability in cable-free Radio LANs. Wireless personal Communications volume 4, January 1997.
5. David B.Johnson and David A. Maltz.Dynamic Source Routing in Ad Hoc Wireless Networks. Mobile Computing, volume 5.
6. Richard G. Ogier . Efficient Routing Protocols for Packet-Radio Networks Based on Tree Sharing ,November 1999
7. RFC 3626

