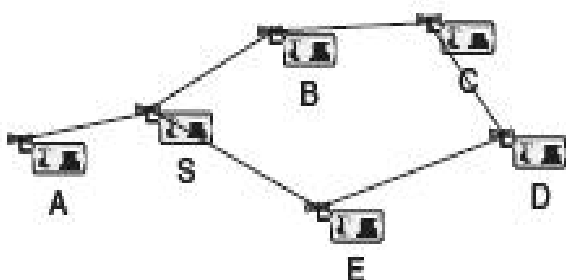


**COMPARATIE INTRE PERFORMANTELE
PROTOCOALELOR DE RUTARE
OLSR, TORA, AODV, DSR**

1 Introducere

Intr-o retea multihop, nodurile intermediare trebuie sa transmita pachete de la sursa la destinatie. Nodurile intermediare trebuie sa decida carui vecin de ordin 1 i se va transmite informatia, daca nodul intermediar nu este destinatia. Se folosesc tabele de rutare care enumera vecinii cei mai semnificativi pentru retransmisia de pachete. Constructia si intretinerea acestor tabele este sarcina protocolului de rutare.

Vom face cateva consideratii de ordin general:



Destination	Next-hop neighbor	Cost
A	A	1
D	A	3
D	B	3
D	E	2
E	E	2

(a) Node S's routing table

Destination	Next-hop neighbor	Cost
A	S	2
D	C	2
D	S	3
E	A	2
E	C	3

(b) Node B's routing table

Exemplu simplu de tabele de rutare intr-o topologie posibila de WN (S->D)

Cea mai simpla cale de transmitere a pachetelor este inundarea retelei prin transmiterea pachetului catre toti vecinii. Cat timp sursa si destinatia sunt in aceeași componenta interconectata a rețelei, pachetul va ajunge cu siguranta la destinatie. Pentru a evita circularea continua a pachetelor si broadcast storm-urile, nodul trebuie sa retransmita pachetul o singura data, necesitand prezenta unor mecanisme de identificare a sursei si a secventei pachetului. Este de asemenea necesara implementarea unei forme de expirare la nivelul pachetului (gen campul TTL la retelele IP), pentru a evita propagarea nenesesara daca destinatia nu este disponibila.

O alternativa la propagarea catre toti vecinii este transmisia pachetului catre un vecin arbitrar. Aceasta metoda, cunoscuta in literatura de specialitate ca "gossiping" rezulta intr-o traversare aleatoare a pachetului prin retea catre destinatie. In mod clar, intarzierile sunt mari in acest caz. De fapt, flooding-ul si gossiping-ul sunt doua modalitati extreme ale spatiului de proiectare. Alternativ, sursa ar putea transmite mai multe copii ale pachetului pe cai aleatoare,

sau fiecare nod intermediar ar inainta mai multe copii unui subset de vecini, utilizand un protocol de control topologic. Ultima modalitate se mai numeste si flooding controlat.

Aceste metode, desi simplu de implementat, produc performante foarte slabe, cauzate de ignorarea topologiei retelei. Din aceasta cauza se prefera utilizarea de protocoale de rutare complexe, a caror sarcina o constituie constructia tabelii de rutare, identificarea vecinilor favorabili pe baza costurilor, si forward-area in consecinta.

Construirea tabelilor de rutare este realizata de algoritmi specifici (in retelele fixe, protocoalele sunt in general distance vector sau link state Bellman-Ford/Dijkstra). Retelele wireless necesita o abordare diferita: protocoale de rutare distribuite, cu overhead mic, autoconfigurabile si rezistente la schimbari dese de topologie.

Rutarea ad-hoc a primit o atentie stiintifica considerabila, si un numar mare de protocoale a fost conceput. Aceste protocoale se pot imparti in protocoale conservatoare table-driven, proactive, care incearca sa mentina o informatie corecta in tabellele de rutare si protocoalele on-demand care construiesc tabela numai cand pachetul este transmis unei destinatii necunoscute. Granita intre cele doua tipuri nu este foarte stricta, si o serie de protocoale hibride au fost implementate.

Protocoale table-driven: DSDV (Destination-Sequenced Distance Vector), CGSR (Clusterhead Gateway Switch Routing) si WRP (Wireless Routing Protocol) .

Protocoale on-demand: DSR (Dynamic Source Routing), AODV (Ad hoc On-demand Distance Vector), TORA (Temporally-Ordered Routing Algorithm), ABR (Associativity-Based Routing) si SSR (Signal Stability Routing) .

O problema comuna protocoalelor ad-hoc este necesitatea flooding-ului de mesaje de control pentru explorarea topologiei retelei si identificarea nodurilor destinatie.

Desigur, protocoalele mentionate mai sus au ca obiectiv principal, in implementare, eficienta energetica si pornesc de la presupunerea ca traficul de date nu are constrangeri legate de livrarea pachetelor. Cresterea interesului in aplicatii care necesita o performanta end-to-end garantata si introducerea de senzori video pun probleme serioase protocoalelor actuale. Transmiterea cu succes a datelor in acest caz necesita un management al retelei atat din punct de vedere al consumului energetic cat si QoS (Quality of Service).

2 Parte teoretica

2.1 Protocolul de rutare OLSR (Optimized Link State Routing)

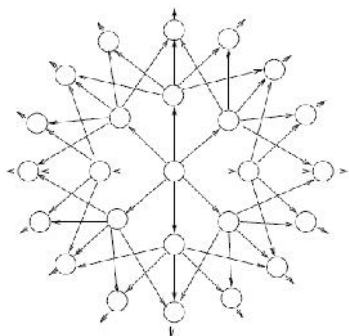
Protocolul OLSR a fost implementat pentru rețele mobile ad-hoc, fiind de tip proactiv (fiecare nod schimbă periodic informații legate de topologia rețelei cu celelalte noduri). Acest protocol folosește rutare hop-cu-hop, fiecare nod utilizând informațiile proprii pentru rutarea pachetelor. OLSR menține stabilitatea algoritmului „link state” și are avantajul că rutele sunt disponibile în orice moment datorită naturii sale proactive.

OLSR include și elemente reactive la schimbarea topologiei rețelei. Această reactivitate poate fi optimizată prin reducerea intervalului maxim de timp pentru transmisia periodică a traficului de control. Deoarece menține continuu rute către toate destinațiile din rețea, protocolul este benefic pentru modele de trafic în care un subset mare de noduri comunică cu un alt subset.

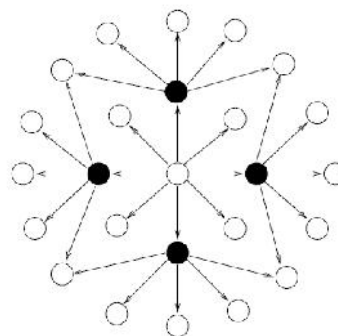
OLSR este proiectat să lucreze într-o manieră complet distribuită. Nu necesită mecanisme de control al erorilor la transmisia mesajelor, deoarece fiecare nod transmite periodic informație de control și poate susține o pierdere rezonabilă a acestor mesaje.

2.1.1 Prezentare generală a funcționării protocolului

În cazul OLSR, fiecare nod selectează un set din nodurile vecine ca „relee multipunct”. Doar nodurile selectate ca relee sunt responsabile cu dirijarea traficului de control. Acestea au rolul de a inunda în întreaga rețea mesajele de control care le recepționează, astfel se reduce și numărul de retransmisii al traficului de control. Releul multipunct este o optimizare a mecanismului clasic de inundare, figurile următoare demonstrând acest lucru:



**Inundarea mesajelor în rețea folosind
algoritmul „link state” clasic**



Inundarea mesajelor în rețea folosind relee multipunct

Nodurile care au fost alese ca relee de un vecin anunță periodic această informație în mesajele lor de control; asadar anunță faptul că au conectivitate directă cu nodurile care le-au ales ca relee. Releele sunt de asemenea folosite pentru calcularea rutelor de la un nod dat către orice destinație din rețea.

Rolul cel mai important al releelor multipunct este sa minimizeze inundarea mesajelor in retea prin reducerea retransmisiilor redundante. Nodurile vecine selectate ca relee retransmit toate mesajele receptionate. Vecinii unui nod care nu sunt considerati relee, receptioneaza si proceseaza mesajele difuzate, inasa nu le retransmit.

Fiecare nod mentine informatii despre setul de vecini selectati ca relee. Aceste informatii sunt obtinute din mesaje periodice HELLO receptionate de la vecini.

2.1.2 Functionarea protocolului OLSR

OLSR a fost implementat cu un set de functionalitati de baza, care sunt necesare pentru ca protocolul sa functioneze, si un set de functii auxiliare, care pot fi aplicabile in scenarii specifice.

Scopul divizarii functionarii OLSR intr-un set de functii de baza si un set de functii auxiliare este acela de a se obtine un protocol simplu si usor de inteles si de a adauga complexitate doar daca functii aditionale mai avansate sunt necesare.

Funcitiile de baza ale includ specificatii universale ale mesajelor de control ale protocolului, transmisia acestora in retea, detectarea link-urilor si a nodurilor vecine, descoperirea topologiei si calcularea rutelor.

Detectarea link-urilor este realizata prin emisia periodica de mesaje HELLO, pe interfetele la care sunt conectate alte noduri. Prin mesajele HELLO nodurile retelei schimba intre ele tabelele de link-uri locale, de vecini si de relee multipunct. Astfel fiecare nod descopera topologia locala a retelei.

In privinta detectiei vecinilor exista doua cazuri. Daca retea are noduri cu o singura interfata, tabela de vecini poate fi dedusa direct din informatiile schimbate pentru detectia link-urilor. Daca retea are noduri cu interfete multiple, sunt necesare informatii aditionale.

Calcularea tabelii de rutare se face pe baza informatiilor „link state” achizitionate de fiecare nod si a configuratiei interfetelor.

Functionalitatea protocolului OLSR poate fi divizata in trei module principale: detectia vecinilor, inundarea „link state” a mesajelor si utilizarea releelor multipunct. Majoritatea traficului de control este bazat pe informatiile pe care le stocheaza fiecare nod care ruleaza protocolul OLSR. Aceste informatii sunt actualizate dinamic pe masura ce fiecare nod receptioneaza mesaje de control.

Intrebari :

- 1) Ce avantaje prezinta protocolul OLSR?
- 2) In ce retele este utila folosirea acestui protocol?
- 3) Care sunt rolurile releelor multipunct?

- 4) Ce noduri sunt responsabile cu dirijarea traficului de control?
- 5) Care sunt functiile de baza ale protocolului OLSR?

2.2 Protocolul TORA (Temporally-Ordered Routing Algorithm)

TORA este un algoritm de rutare, folosit in principal pentru a spori scalabilitatea si reprezinta un protocol de rutare adaptiv. Este utilizat in cadrul retelelor multi-hop care prezinta urmatoarele caracteristici: executie distribuita, rutare multipath, organizare si mentinere reactiva sau proactiva a rutei, minimizarea incarcarii comunicarii.

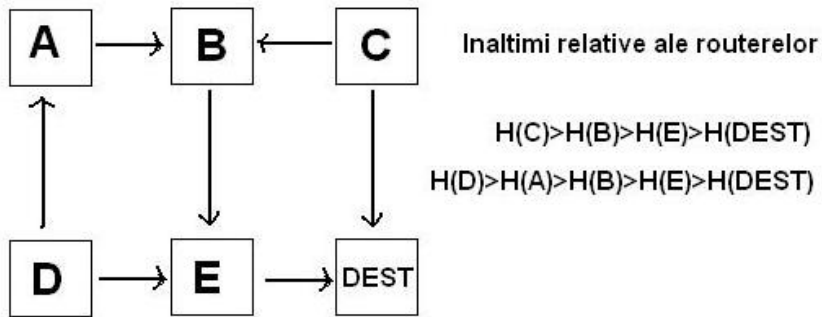
TORA este distribuit, routerele trebuie sa mentina doar informatii despre routerele adiacente. In timpul operatiunii de reactie, sursele initiaza, la cerere, stabilirea rutelor pentru o anumita destinatie. Acest mod de operare poate fi avantajos in retelele dinamice, cu modele de trafic relativ rare, astfel incat nu este necesar sa se mentina in orice moment rute intre fiecare pereche sursa-destinatie. In acelasi timp, destinatiile selectate pot initia operatii proactive. Acest lucru permite ca rutele sa fie mentinute in mod activ pentru destinatiile pentru care rutarea este ceruta in mod constant/frecvent.

TORA este proiectat pentru a minimiza overhead-ul comunicarii asociat cu adaptarea la schimbarile topologice ale retelei.

2.2.1 Prezentare generala a functionarii protocolului

TORA indruma link-urile intre routere sa formeze o structura a rutei care este utilizata pentru a transmite datagrams catre destinatie. Un router atribuie o directie link-ului cu un router vecin. Masura intretinuta de catre un router poate fi gandita conceptual ca fiind "inaltimea" router-ului (de exemplu, link-urile sunt orientate de la routerele inalte la cele mai joase). Semnificatia inaltimei si a link-ului directionat este ca un router poate transmite datagrams doar in sensul curentului. Link-urile de la un router la un router vecin ce are inaltimea necunoscuta nu pot fi folosite pentru a inainta.

In exemplul urmator, C este mai aproape de destinatie decat B in ceea ce priveste numarul hopurilor, dar inaltimea metrica a lui C este mai mare decat cea a lui B:

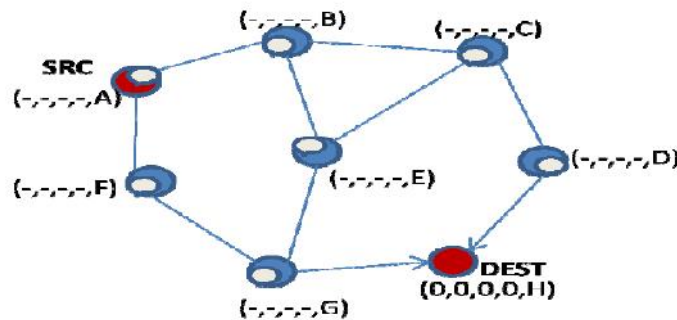


Reprezentare conceptuala a graficului in functie de inaltimile relative ale routerelor din retea

2.2.2 Functionarea protocolului TORA

TORA stabileste rute scalate intre sursa si destinatie si nu foloseste teoria celui "mai scurt drum". TORA construiește rute optimizate prin utilizarea a patru mesaje: de interogare, de actualizare, de reglare si de optimizare. Aceasta operatiune se efectueaza de catre fiecare nod pentru a trimite diversi parametri intre sursa si destinatie. Link-urile construite in TORA sunt denumite "inaltimi". La inceput, inaltimea nodurilor este de forma $(-, -, -, i)$ si cea a destinatiei este setata $(0, 0, 0, 0, dest)$. Inaltimele sunt ajustate ori de cate ori exista o schimbare in topologie.

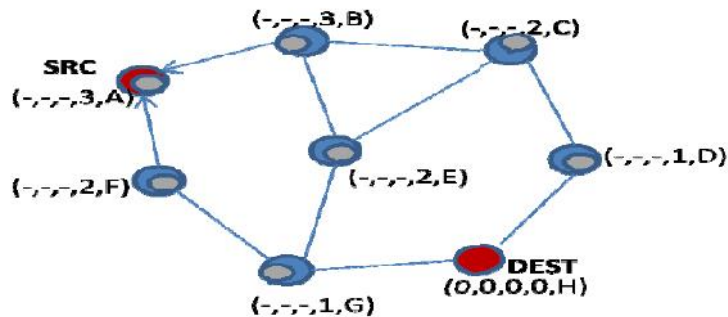
Un nod care are nevoie de o cale spre o destinatie trimite o cerere a rutei. In cazul in care un pachet ajunge la un nod cu informatii despre nodul destinatie, este trimis pe calea inversa un raspuns cunoscut ca actualizare. Mesajul de actualizare stabileste valoarea inaltimii nodurilor invecinate la cea a nodului care trimite actualizarea.



Descoperirea rutei prin intermediul TORA

In figura nodul A este sursa iar nodul H este destinatia. Nodul A emite un mesaj de cerere in intreaga retea. Doar un vecin aflat la un hop de destinatie raspunde cererii. In cazul in care cererea ajunge la un nodul cu informatii referitoare la destinatie, acest nod trimite inapoi o

actualizare. In acest caz, nodul D si nodul G sunt la un hop distanta de destinatie. Prin urmare, acestea vor raspandi actualizarile ca in figura de mai jos:



Descoperirea rutei prin intermediul TORA

In acest tip de algoritm exista erori. Principala eroare este data de faptul ca este, in mare masura, dependent de numarul de noduri activate la instalarea initiala. Defectul este ca raspunsul la cererile de trafic nu este independent. Acesta depinde de rata de modificare a traficului (numarul de noduri) din retea. In cazul in care volumul traficului din retea creste cu o panta pozitiva abrupta, TORA nu ar fi o alegere buna pentru acest tip de retea.

Intrebari :

- 1) Ce avantaje prezinta protocolul TORA?
- 2) Care este principalul dezavantaj?
- 3) Ce se intampla dupa ce un nod trimite un mesaj de cerere a rutei?
- 4) Ce nod va raspunde unui mesaj de cerere?
- 5) In ce retele este utila folosirea algoritmului de rutare TORA si in ce retele nu este utila?

2.3 Protocolul AODV (Ad hoc On-demand Distance Vector)

Algoritmul AODV permite rutarea dinamica, self-starting si multi-hop intre nodurile mobile participante care doresc sa stabileasca si sa mentina o retea ad-hoc, permitandu-le acestora sa obtina rapid rute pentru noile destinatii. AODV permite nodurilor mobile sa raspunda in timp potrivit in cazul distrugerii link-urilor si modificarilor din topologia retelei. Cand se produce distrugerea link-ului, AODV anunta setul de noduri afectat, pentru ca acestea sa fie capabile sa anuleze rutele care folosesc link-ul pierdut.

AODV este un protocol reactiv, asa incat rutele sunt create si mentinute doar atunci cand acestea sunt necesare. In tabelul de rutare sunt stocate informatii referitoare la urmatorul hop pana la locul destinatiei si o secventa de numere, care este primita de la destinatie si care indica "prospetimea" informatiilor primite. De asemenea, informatia despre vecinii activi a fost primita prin intermediul mecanismului descoperirii nodului destinatie. Cand ruta corespunzatoare se intrerupe, vecinii pot fi notificati.

2.3.1 Prezentare generala a functionarii protocolului

Cererile rutei (Route Requests - RREQs), raspunsurile rutei (Route Replies - RREPs) si erorile rutei (Route Errors - RERRs) sunt tipuri de mesaje definite de AODV. Aceste tipuri de mesaje sunt primite prin intermediul UDP, IP. De exemplu, se asteapta ca nodul solicitant sa isi utilizeze adresa IP ca fiind adresa de IP initiatore pentru mesaje.

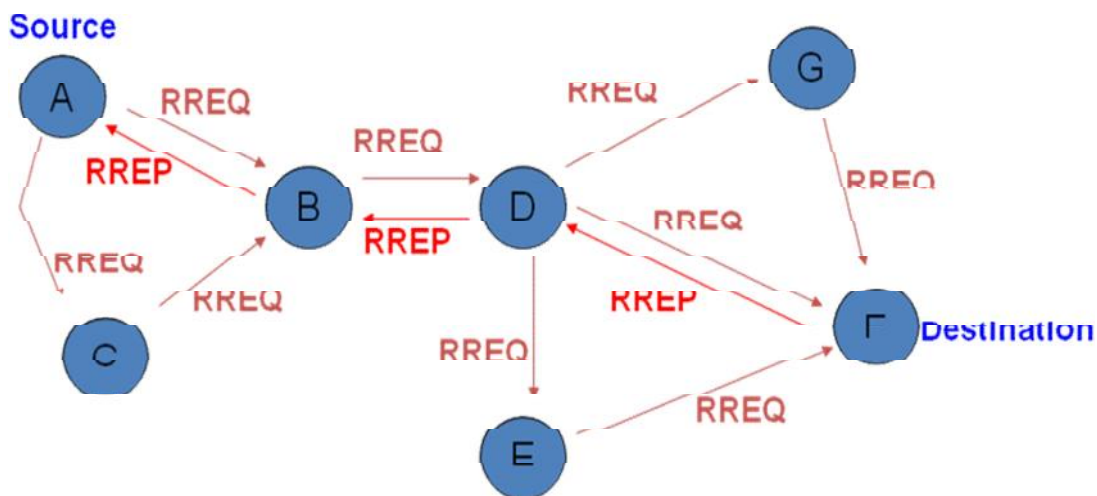
Pentru transmiterea mesajelor, este folosita adresa IP de broadcast (255.255.255.255). Acest lucru inseamna ca astfel de mesaje nu sunt transmise oarecum. Oricum, operatiunile AODV nu necesita ca anumite mesaje (de exemplu, RREQ) sa fie difuzate pe scara larga, in intreaga retea ad-hoc. Atata timp cat punctele finale ale conexiunii au rute valide reciproc, protocolul AODV nu trimite trafic inutil in retea.

In cazul in care este necesara o ruta spre o noua destinatie, nodul transmite un mesaj RREQ pentru a gasi o cale catre destinatie. Ruta poate fi determinata atunci cand RREQ ajunge fie la destinatie, fie la un nod intermediar cu o "stare destul de proaspata" a caili pana la locul de destinatie. Ruta in "starea destul de proaspata" este o cale de intrare valida pentru destinatie al carei numar de ordine este cel putin egal cu cel continut in RREQ. Traseul este valabil prin trimiterea unicast a unui RREP inapoi la originea RREQ. Fiecare nod care primeste cererea parcurge un traseu inapoi spre solicitarea initiala, astfel incat RREP sa poata fi transmis unicast de la destinatie de-a lungul unei cai spre origine sau, de asemenea, de la orice nod intermediar care este capabil sa satisfaca cererea.

Nodurile monitorizeaza status-ul link-ului hopurilor urmatoare din traseele active. In cazul in care se observa o distrugere a link-ului intr-o ruta activa, atunci este folosit un mesaj RERR pentru a anunta alte noduri de pierderea care a survenit.

Mesajele RERR indica acele destinatii care nu mai sunt accesibile prin intermediul link-ului distrus. Pentru a permite acest mecanism de raportare, fiecare nod pastreaza o "lista a precursorilor", care contine adresele IP pentru fiecare dintre vecinii sai care l-ar utiliza ca un

urmator hop spre fiecare destinatie. Informatiile din listele precursorilor sunt cel mai usor dobandite in timpul procesului de generare al mesajelor RREP care, prin definitie, trebuie sa fie trimise la nod intr-o lista a precursorilor. In cazul in care RREP are o lungime al carei prefix este diferit de zero, atunci initiatorul RREQ care a solicitat informatia RREP este inclus printre precursorii rutei.



Functionarea protocolului AODV

Cand se produce ruperea legaturii nodul, poate incerca sa o repare pe plan local, doar in cazul in care destinatia nu se afla mai departe de un anumit numar de hopuri. Pentru a repara legatura, nodul majoreaza numarul de ordine al destinatiei si transmite mesajul RREQ catre nod. TTL-ul din antetul IP-ului trebuie sa fie calculat, astfel incat procesul de reparatie sa nu fie raspandit in intreaga retea.

Nodul asteapta mesajul RREP catre mesajul RREQ pentru o perioada de timp specificata. In cazul in care mesajul RREP nu este primit, atunci starea intrarii din tabelul de rutare este modificata ca nefiind valida.

Daca nodul primeste mesajul RREP, atunci este comparat numarul hopului. In cazul in care acesta este mai mare decat precedentul, atunci RERR, al carui domeniu N a fost setat, este difuzat. Domeniul N din mesajul RERR indica faptul ca nodul a reparat la nivel local legatura si intrarea din tabel nu trebuie sa fie stearsa. Mesajul RREP primit este tratat ca fiind originalul mesaj RREP. Reparatia legaturii inainte ca datele sa fie trimise nodurilor ce nu sunt disponibile reprezinta o reparatie proactiva. Aceasta poate fi ineficienta, din cauza riscului generat de reparatia rutelor care nu mai sunt folosite. In concluzie, reparatia proactiva poate fi folosita pe baza traficului local.

Intrebari :

- 1) Prezentați principiul de functionare al protocolului AODV.

- 2) Care sunt avantajele AODV?
- 3) Ce dezavantaj prezinta?
- 4) Ce fel de protocol este AODV? Explicati.
- 5) Care sunt mesajele folosite in AODV si ce semnifica fiecare?

2.4 Protocolul DSR (Dinamyc Source Routing)

Rutarea dinamica a sursei apartine protocoalelor de rutare on-demand din cadrul retelelor ad-hoc multi-hop wireless ale nodurilor mobile. Rutarea sursei reprezinta o tehnica in care emitatorul unui pachet determina succesiunea completa a nodurilor prin care va fi trimis acest pachet. DSR permite retelei sa se auto-organizeze si auto-configureze, fara a fi nevoie de existenta unei infrastructuri a retelei sau de administrarea acesteia.

Protocol DSR este compus din doua mecanisme principale: “descoperirea rutei” (route discovery) si “intretinerea rutei”(route maintenance), care lucreaza impreuna pentru a permite nodurilor sa descopere si sa mentina rutele catre destinatiile arbitrare din cadrul retelelor ad-hoc.

Descoperirea rutei este mecanismul prin care un nod S care doreste sa trimita un pachet spre un nod destinatie D obtine o sursa de ruta catre D. Descoperirea rutei este folosita doar atunci cand S incearca sa trimita un pachet catre D si nu cunoaste nicio ruta disponibila catre D.

Mentinerea rutei este mecanismul prin care nodul S este capabil sa detecteze, in timp ce foloseste o sursa a rutei, daca topologia retelei s-a schimbat, astfel incat aceasta nu mai poate fi folosita ca traseu catre D din cauza unui link care nu mai functioneaza. Cand intretinerea rutei indica ca sursa unei rute este distrusa, S poate utiliza orice alta ruta pe care o cunoaste pentru a ajunge la D, sau poate apela din nou la descoperirea rutei, pentru a gasi o noua ruta. Calea de intretinere pentru aceasta ruta este folosita doar atunci cand S expediază pachete pentru D.

In cadrul DSR, descoperirea rutei si mentinerea rutei opereaza in intregime “on-demand”. In special, spre deosebire de alte protocoale, DSR nu necesita niciun fel de pachete periodice la orice nivel din cadrul retelei. Modificarile in topologia retelei care nu afecteaza rutele aflate in folosinta sunt ignorate si nu provoaca reactii din partea protocolului.

Protocolul permite existenta mai multor rute catre orice destinatie si permite fiecarui expeditor sa selecteze si sa controleze routerele folosite in rutarea pachetelor sale. Alte avantaje ale protocolului DSR includ rutarea garantata fara bucle, sprijinul pentru utilizarea in retelele care contin link-uri unidirectionale, recuperarea rapida atunci cand se produce o schimbare in retea a rutelor.

Protocolul DSR este conceput in principal pentru retelele mobile ad-hoc de pana la aproximativ doua sute de noduri si este proiectat sa functioneze corect chiar cu rate foarte ridicate de mobilitate.

2.4.1 Functionarea protocolului DSR

Versiunea de baza a DSR utilizeaza in mod expres “rutarea sursei”, in care fiecare pachet de date trimis contine in antetul sau lista ordonata de noduri prin care va trece pachetul. Aceasta utilizare explicita a rutarii sursei permite expeditorului sa selecteze si sa controleze rutele folosite pentru propriile pachete, accepta utilizarea mai multor rute catre orice destinatie (de exemplu, pentru echilibrarea incarcarii) si permite o simpla garantie ca rutele folosite sunt fara bucle. Prin includerea rutei sursa in antetul fiecarui pachet de date, nodurile care transmit oricare din aceste pachete pot memora aceste informatii pentru o utilizare ulterioara.

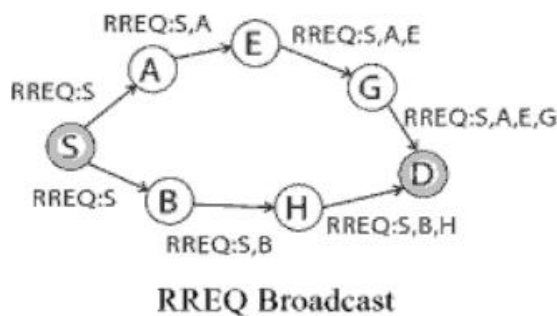
Ca raspuns la o singura descoperire a rutei, un nod poate gasi “cache-ul” mai multor rute catre orice destinatie. Acest suport pentru rute multiple permite ca reactia la modificarile rutarii sa fie mult mai rapida, deoarece un nod cu mai multe rute catre o destinatie poate incerca o alta cale a rutei in cazul in care una care a fost utilizata a esuat. De asemenea, acest “cache” al rutelor multiple impiedica nevoia de a genera o noua ruta ori de cate ori o ruta aflata in folosinta este distrusa. Expeditorul unui pachet controleaza si selecteaza ruta utilizata pentru propriile sale pachete care, impreuna cu suportul pentru rute multiple, permit ca diferite aspecte, ca de exemplu echilibrul sarcinii, sa fie definite. In plus, toate rutele folosite sunt garantate a fi fara bucle, deoarece expeditorul poate evita duplicarea hopului in traseele selectate.

Operatiunile de descoperire si creare a rutei prin DSR sunt proiectate pentru a permite link-urilor unidirectionale si rutelor asimetrice sa fie usor de suportat. In retelele wireless, se poate ca un link sa nu functioneze la fel de bine in ambele sensuri din cauza antenei diferite sau a surselor de interferenta. DSR permite ca astfel de link-uri unidirectionale sa fie utilizate cand este necesar, imbunatatind performanta generala si conectivitatea retelei in sistem.

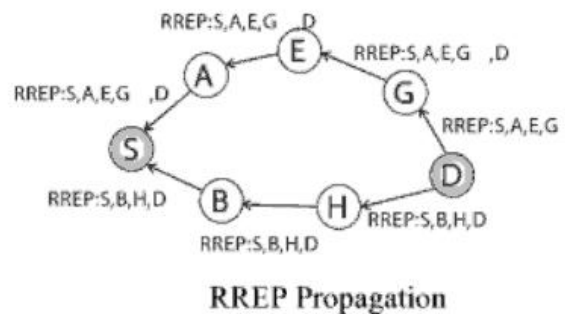
2.4.2 Mecanismul de descoperire a rutei

Atunci cand un nod sursa doreste sa transmita un pachet IP catre un nod destinatie, sursa verifica ruta din “cache” pentru a determina daca exista deja o ruta catre destinatia dorita. Daca in “cache-ul” rutelor este gasita o ruta catre destinatia dorita care nu mai este valabila, atunci pachetul este trimis prin intermediul acestei rute, dar plasand in antetul pachetului ruta sursa: adresele fiecarui nod prin care trece pachetul pana sa ajunga la destinatie (succesiunea completa a hopurilor).

In caz contrar, daca nu exista o astfel de ruta se apeleaza la descoperirea rutei prin emiterea unui pachet de cerere a rutei (RREQ) catre vecini. Fiecare nod intermediar verifica daca este cunoscut vreun traseu catre nodul destinatie. Daca nu exista nicio astfel de ruta, nodul intermediar adauga la inregistrarea rutei pachetului propria sa adresa si trimite pachetul catre vecini. Acest proces inceteaza cand este gasit un nod destinatie sau un nod intermediar care cunoaste o ruta catre destinatie ce nu mai este valabila. Din acest punct, este generat un pachet raspuns al rutei (RREP) pentru a reuni nodurile sursa: in cazul in care se ajunge la destinatie, inregistrarea rutei din pachetul RREQ este transferata pachetului RREP; in caz contrar, daca avem de-a face cu un nod intermediar, "cache-ul" rutei pachetului si inregistrarea rutei pachetului RREQ sunt impreunate si transferate in pachetul RREP. Apoi, pachetul RREP trebuie sa se intoarca la sursa pentru a transmite ruta descoperita.



Procesul de descoperire a rutei



Propagarea a doua mesaje RREP inapoi catre sursa

2.4.3 Mecanismul de mentinere a rutei

Fiecare nod care trimite un pachet este responsabil sa confirme ca datele pot trece peste legatura de la acel nod la hopul urmat. Cat timp o ruta se afla in folosinta, procedura de mentinere a rutei urmareste operatiile rutei si informeaza emitatorul daca apare vreo eroare in timpul rutarii. Acest mecanism determina daca urmatul hop al rutei inca functioneaza si utilizeaza aceasta informatie pentru a corecta „cache-ul” rutei.

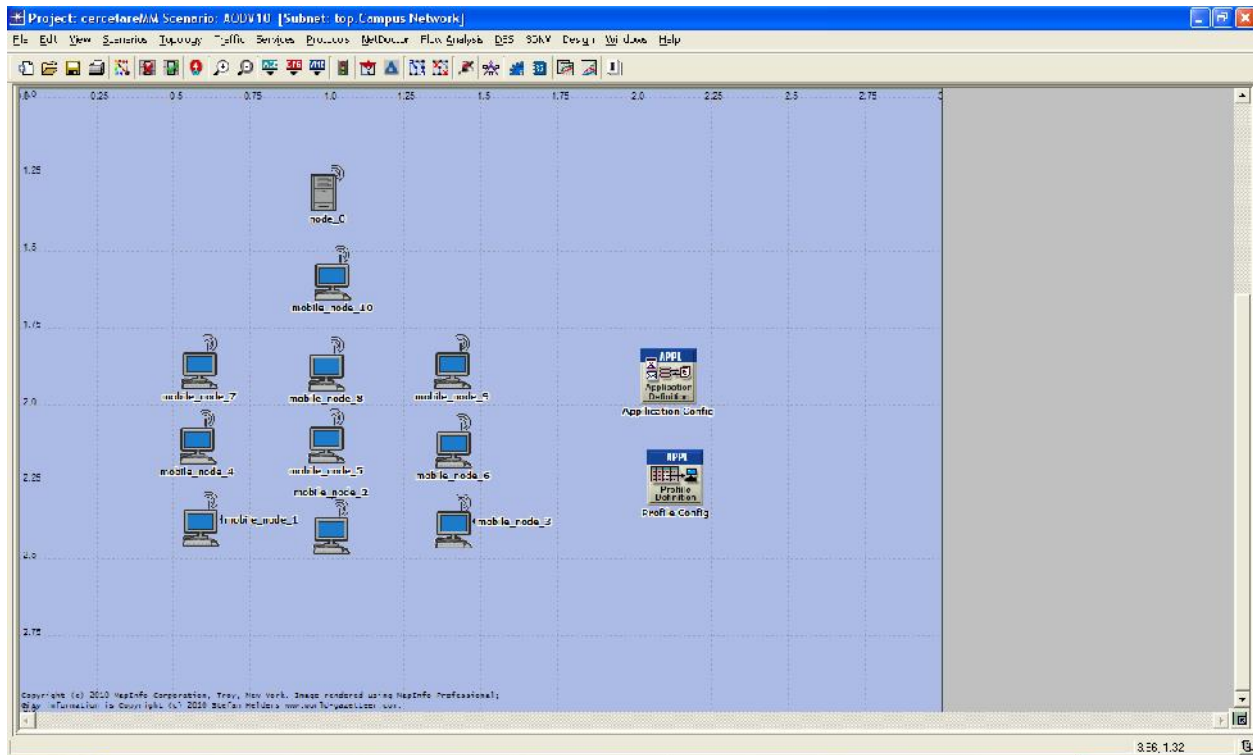
Intrebari :

- 1) Ce fel de protocol este DSR?
- 2) In ce fel de retele este folosit?
- 3) Ce avantaje prezinta?
- 4) Care sunt cele doua mecanisme principale ale protocolului DSR?
- 5) Care sunt principalele mesaje schimbate de nodurile dintr-o retea DSR?

3. Parte practica

3.1 Implementarea rețelei

În figura de mai jos este prezentat un model de simulare pentru un scenariu ce cuprinde 10 noduri:



Categoria I : Trafic mediu si 10 noduri fixe care efectueaza trafic cu serverul FTP, cu putere de transmisie de 0,001W.

Categoria II : Trafic mediu si 10 noduri mobile care se deplaseaza in jurul serverului FTP cu o viteza de 15 m/s, avand o putere de transmisie de 0,001 W.

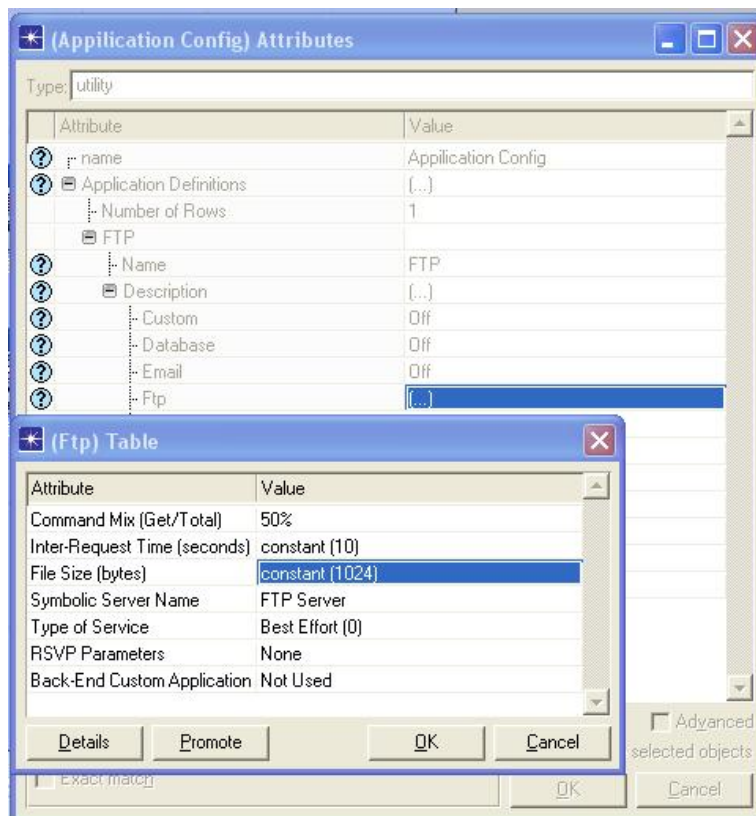
Categoria III: Trafic mediu si 42 noduri fixe care efectueaza trafic cu serverul FTP, cu putere de transmisie de 0,001W.

Categoria IV: Trafic mediu si 42 noduri mobile care se deplaseaza in jurul serverului FTP cu o viteza de 15 m/s, avand o putere de transmisie de 0,001 W.

În total au fost simulate 14 scenarii si s-a rulat fiecare simulare la 900s (simulation time). Pentru fiecare categorie, a fost simulata comportarea protocoalelor de rutare AODV, DSR, TORA si OLSR. Scopul a fost modelarea comportamentului protocoalelor de rutare variand numarul nodurilor si mobilitatea. S-au colectat statistici individuale (DES) pentru fiecare

protocol in parte, cat si pentru intreaga retea wireless. Apoi a fost examinata media statisticilor pentru traficul de rutare.

A fost selectata o retea de tip campus cu o dimensiune de 3 km X 3 km. Toate nodurile trimit si primesc trafic de la aceeasi sursa : Serverul FTP. In acest proiect am folosit un trafic TCP pentru studierea efectelor protocoalelor asupra performantei retelei. Ca profil am folosit Medium Traffic, putin modificat: timpul de trimitere/sosire al unui pachet este de 10 secunde, iar marimea pachetului este de 1024 bytes, rezultand un trafic de aproximativ 800b pentru fiecare nod. (1024 bytes/pachet*8 biti/byte*1 pachet/10 s=819,2 biti/s).



Parametrii traficului generat

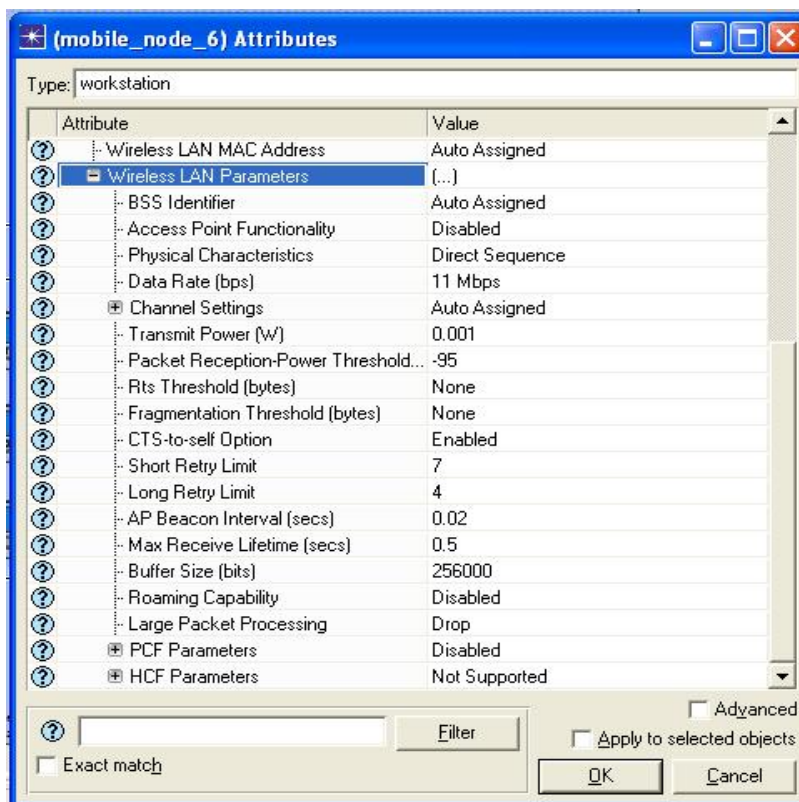
Pentru noduri s-au folosit ca model Wireless Lan Workstation (mobil), iar pentru server s-a folosit modelul Wireless Lan Server (fix). Nodurile au fost setate sa functioneze la rata de 11Mbps operand la o putere de 0.001 W. In cazul folosirii unei puterii de 0.001 W nodurile au o arie de acoperire de aproximativ 300m . Acest lucru se poate demonstra prin utilizarea formulei de mai jos :

$$P = \left(\frac{4\pi D}{0.12476} \right)^2 * 10^{-12.5}$$

Aceasta formula este aplicabila doar pentru urmatoorii parametrii :

- **Physical Characteristics:** Direct Sequence
- **Data Rate:** 11 Mbps
- **Channel Settings:** Auto Assigned
- **Packet Reception-Power Threshold:** -95dBm
- **Omnidirectional** transmitter and receiver antenna (zero gain)

Singurul parametru modificat a fost Transmit Power (0.001W), restul ramanand default. Nodurile nu suporta interfete multi-canal, toate nodurile transmitand pe acelasi canal ales aleatoriu de OPNET. Am incercat alocarea diferita a canalelor pe noduri, dar comunicatia se realiza pe portiuni, doar intre nodurile setate pe acelasi canal. Pentru interfetele multi-canal este nevoie de noi protocoale de rutare pe care versiunea 14.0 nu le suporta.



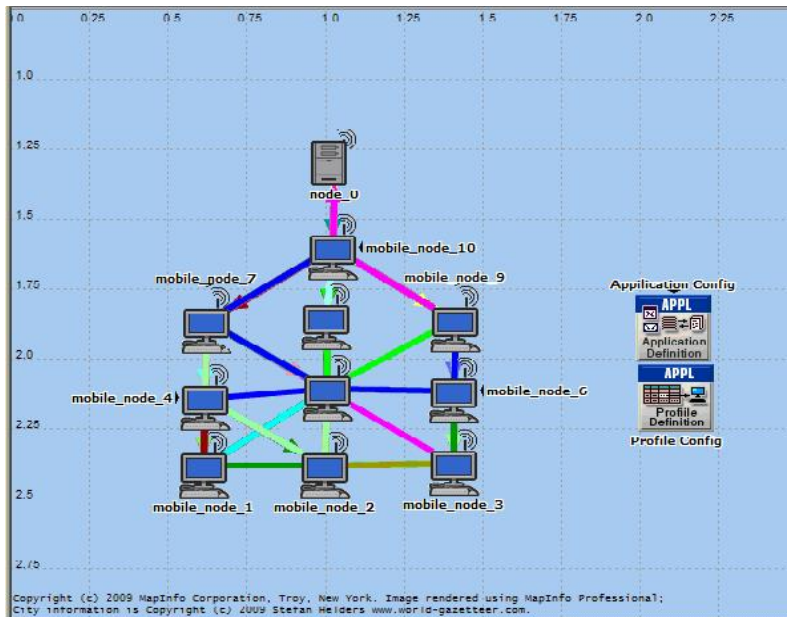
Parametrii statiilor Wireless

Modelul mobilitatii este definit manual, avand forma unui hexagon. Mai jos sunt prezentate pozitiile nodurilor:

	X Pos (km)	Y Pos (km)	Distance (m)	Altitude (m)	Traverse Time	Ground Speed	Wait Time
1	0.000000	0.000000	n/a	0.000000	n/a	n/a	00.00s
2	0.000000	-0.838547	838.547292	0.000000	55.90s	15.000846	00.00s
3	-0.393948	-1.091800	468.329144	0.000000	31.22s	15.000934	00.00s
4	-0.771013	-0.759758	502.424076	0.000000	33.49s	15.002212	00.00s
5	-0.765385	0.292647	1,052.420038	0.000000	1m10.16s	15.000286	00.00s
6	-0.433343	0.596550	450.121186	0.000000	30.01s	14.999040	00.00s
7	0.000000	0.067534	683.845538	0.000000	45.59s	14.999902	00.00s

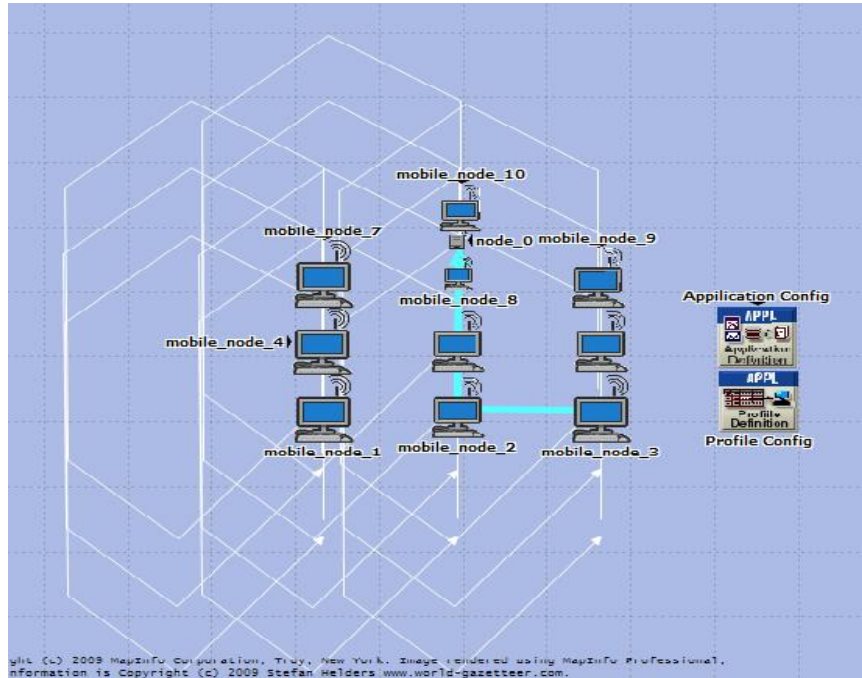
Parametrii mobilitatii

Programul OPNET permite vizualizarea rutelor in orice perioada a simularii. PROTOCOLS→ MANET→DSR→Display DSR Routes. Pentru nodul 8 am setat o putere de transmisie de 0.0005 mW, iar celelalte noduri 0.001W. Dupa cum se observa este ocolit de nodurile 7 si 9 in drumul catre serverul FTP, deoarece distanta dintre nodul 7 si 8, respectiv 8 si 9, este de 400 m, el avand o raza de acoperire radio de 100m .

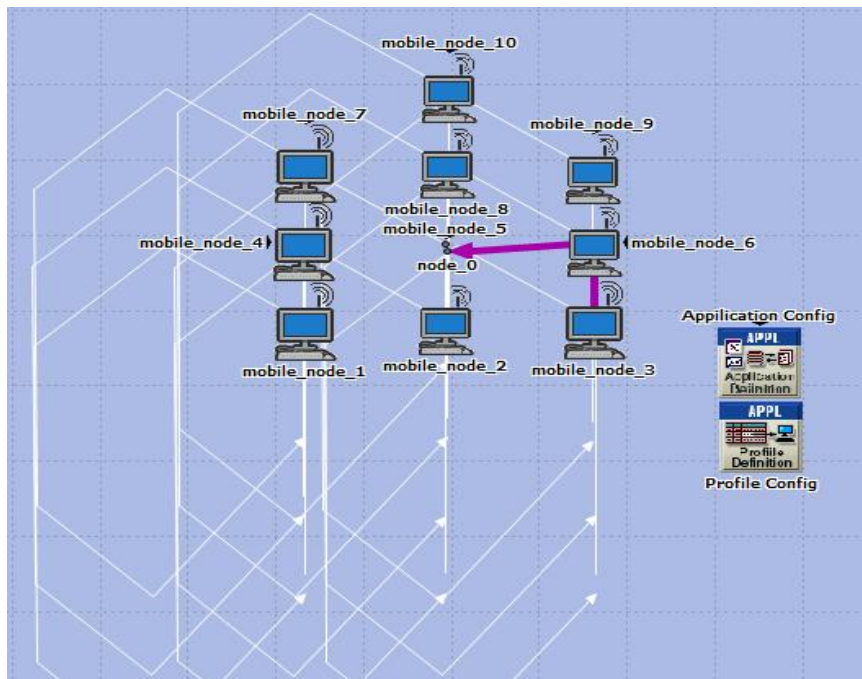


Vizualizarea rutelor pentru protocolul DSR, noduri fixe

Pentru cazul nodurilor mobile am setat ca toate statiile sa transmita cu o putere de 0.001 W. In figurile de mai jos sunt prezentate pozitiile nodurilor si rutele nodului 3 catre serverul FTP la diferite intervale de timp (25 s, 55s, 120 s):



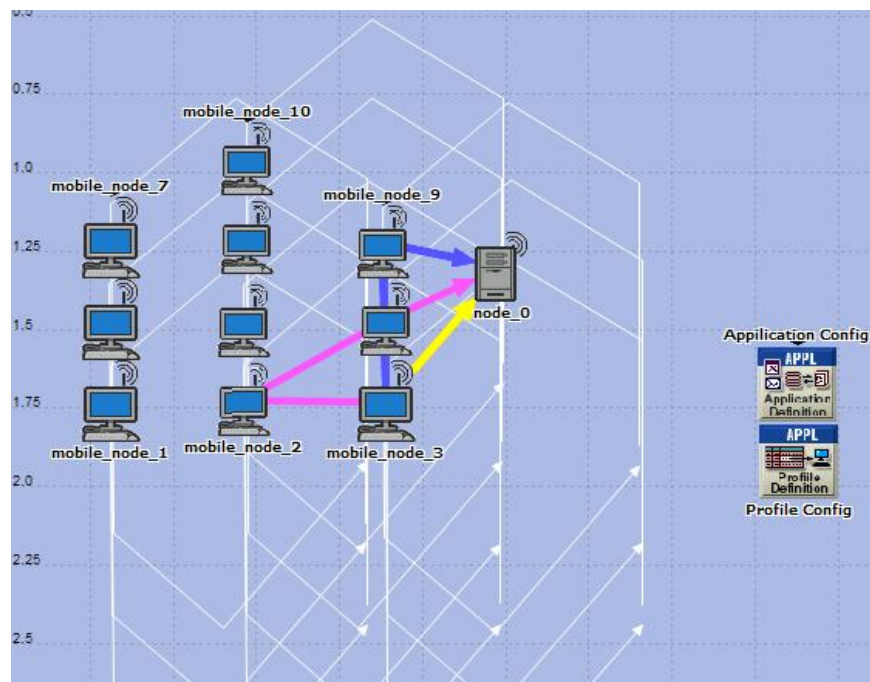
Pozitia nodurilor si ruta nodului 3 la 25 de secunde de la inceperea simularii



Pozitia nodurilor si ruta nodului 3 la 55 de secunde de la inceperea simularii

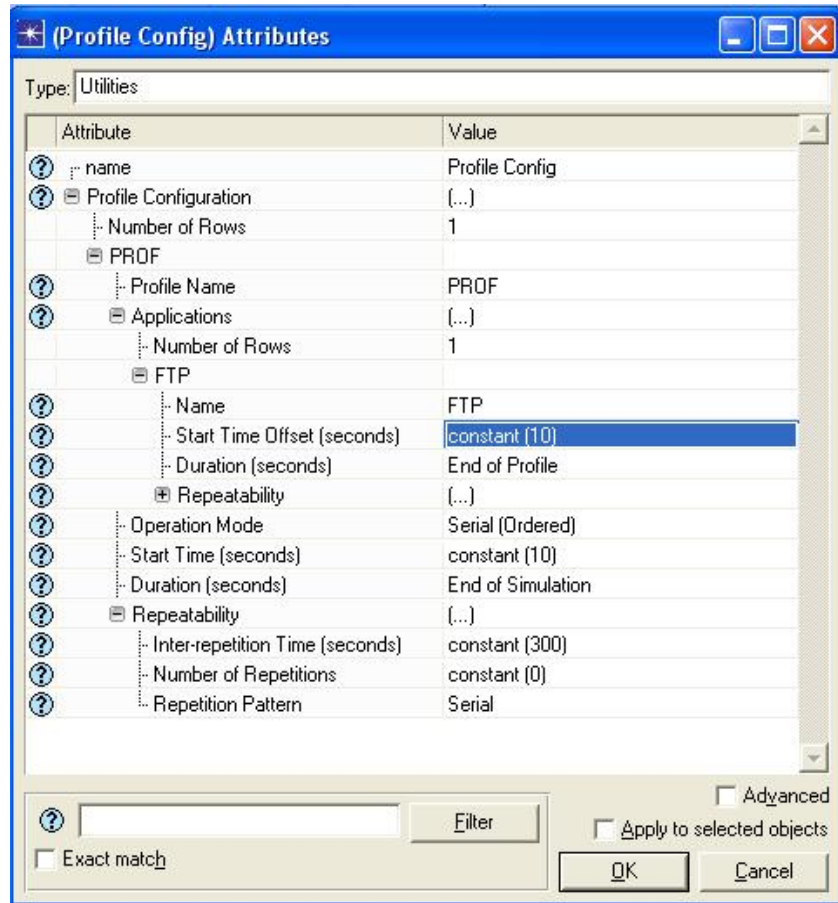
In figura de mai jos am selectat mai multe rute ale nodului 3 catre serverul FTP, la 120 de secunde dupa inceperea simularii. Protocolul DSR mentine mai multe rute catre destinatie, fapt ce se poate vedea din figura de mai jos, nodul putand ajunge catre serverul FTP prin 3 rute valide :

- mobile_node_3 → Server TCP
- mobile_node_3 → mobile_node_4 → mobile_node_9 → Server TCP
- mobile_node_3 → mobile_node_2 → mobile_node_4 → Server TCP



Pozitia nodurilor si rutele nodului 3 la 120 de secunde de la inceperea simularii

Cu ajutorul atributului Profile Config am setat urmatorii parametrii pentru serverul FTP: Start-time offset a fost setat la valoarea „constant(10)” si Duration la valoarea “End of profie”, pentru a seta acest profil sa porneasca la 10 secunde de la inceputul aplicatiei si sa se sfarseasca la finalul acesteia. Urmatorul termen, respectiv Start Time, a fost setat la constant(10), reprezentand timpul de la inceputul simularii dupa care acest profil va porni. Acesti parametrii pot fi vizualizati si in figura de mai jos:



Profile Configuration

3.2 Analize si concluzii

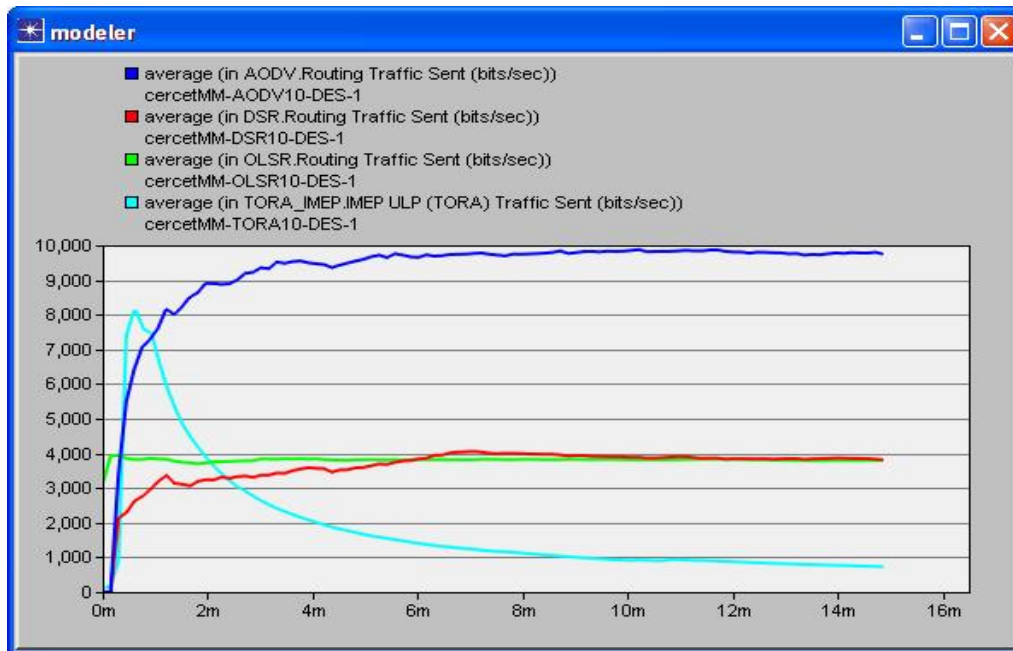
Considerand rezultatele din figurile de mai jos, observam ca AODV trimite cel mai mare trafic de rutare in retea, urmat apoi de TORA. Dupa TORA este OLSR, iar DSR se prezinta cu cel mai mic trafic de rutare trimis in retea. Acesta observatie este valida in toate scenariile prezente. Prin urmare, in cazul traficului de rutare trimis, DSR surclaseaza celelalte 3 protocoale. In retelele mici, DSR ar functiona mult mai bine decat celelalte protocoale amintite.

Superioritatea protocolului DSR vine din natura modului sau de rutare. Fiind un protocol reactiv, DSR trimite trafic in retea doar atunci cand sursa are trafic de trimis, eliminand astfel incarcarea datorita traficului de rutare inutil al altor protocoale. DSR foloseste rutarea bazata pe sursa (source routing), facand astfel ca sursa sa cunoasca inca de la inceput intregul drum pe care pachetul il va parcurge in drumul sau catre destinatie. Toate nodurile intermediare folosesc informatiile stocate in „cache” pentru rutarea pachetelor netriminand astfel raspunsuri catre sursa in timpul descoperirii rutei. Doar nodul destinatie trimite raspuns la cererea descoperire a unei rute. Prezenta multiplelor rute in DSR, reduce numarul de cereri de descoperire a unei rute in

cazul in care un link dispare. Acesti factori impreuna cu absenta update-urilor periodice in DSR, au efectul de a reduce cantitatea traficului de rutare introdus in retea.

In cazul protocolului AODV, in ciuda caracterului sau reactiv, fiecare nod trimite catre sursa raspuns la cererea decoperirii unei rute. Astfel traficul de rutare creste datorita numeroaselor raspunsuri catre nodul sursa. Mai mult, in cazul in care un nod aflat pe drumul catre destinatie dispare, un mesaj de eroare este propagat catre toti vecinii datorita absentei unor rute alternative catre destinatie. Acesta initiaza astfel un nou procedeu de redescoperire a unei noi rute, ducand astfel la marirea traficului de rutare. TORA se claseaza pe pozitia 3. Mecanismul de stare a link-ului al lui IMEP, incluzand mesajele periodice de "HELLO", duc la incarcarea retelei.

OLSR este un protocol proactiv, el inundand continuu retea cu mesaje de control si de rutare pentru a-si mentine tabela de rutare actualizata. Acest lucru duce la dimensiunea uriasa a traficului de rutare.



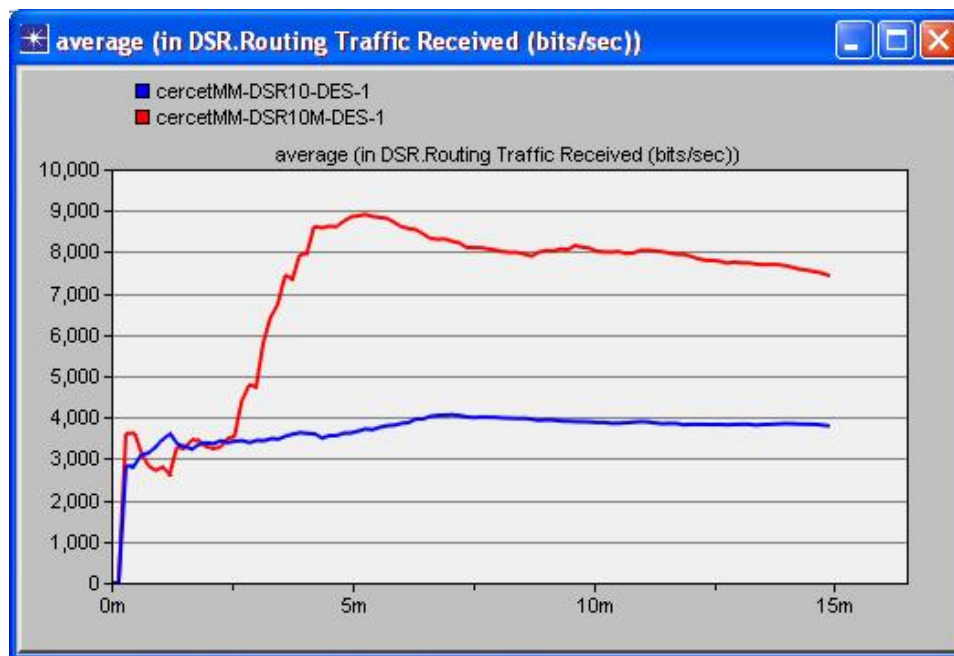
Traficul de rutare pentru AODV, DSR, TORA, OLSR pentru 10 noduri fixe

3.2.1 Efectele incarcarii si mobilitatii pentru fiecare protocol in parte

In figurile de mai jos vor fi prezentate performantele fiecarui protocol in parte in functie de numarul de noduri, puterea de transmisie si mobilitate.

3.2.1.1 Efectele numarului de noduri si mobilitatii in DSR

Dupa cum se observa mobilitatea nu are un efect mare asupra traficului de rutare trimis. La mobilitate ridicata, link-urile se schimba frecvent, cauzand ca DSR-ul sa reactioneze mai frecvent. Avand in vedere cresterea numarului de noduri, performanta unui protocol este direct afectata. Motivul ar putea fi cauzat de difuzarea in intreaga retea a mesajelor RREQ. DSR creeaza pachetele RREQ si le difuzeaza catre toti vecinii. In cazul scenariului cu 42 de noduri este evident faptul ca un nod are mai multi vecini, ducand astfel la un numar mai mare de mesaje RREQ.

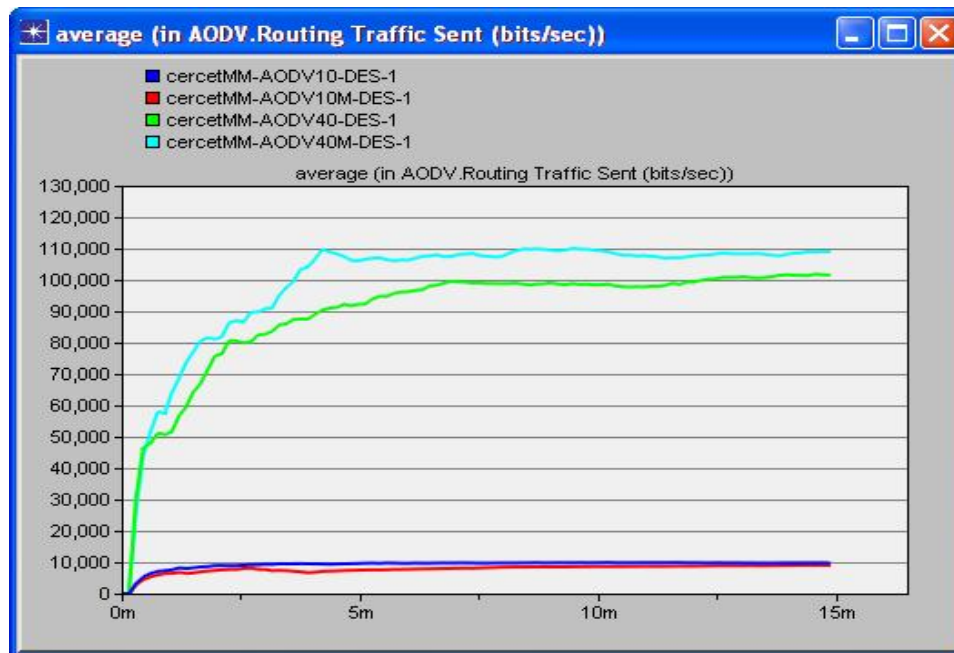


Traficul de rutare DSR → 10 noduri fixe, respective mobile

3.2.1.2 Efectele numarului de noduri si mobilitatii in AODV

Algoritmul AODV utilizeaza algoritmul DSDV si DSR, folosind rutarea hop-by-hop de la DSDV. Utilizarea algoritmului Bellman-Ford din protocolul DSDV asigura transmiterea informatiilor de rutare ale unui nod catre toti vecinii sai. Pentru orice dimensiune a retelei, fiecare router alege pachetele cu cel mai mic hop si le difuzeaza catre vecini. Acest algoritim se dovedeste eficient, indiferent de cat de mare este reteaaua.

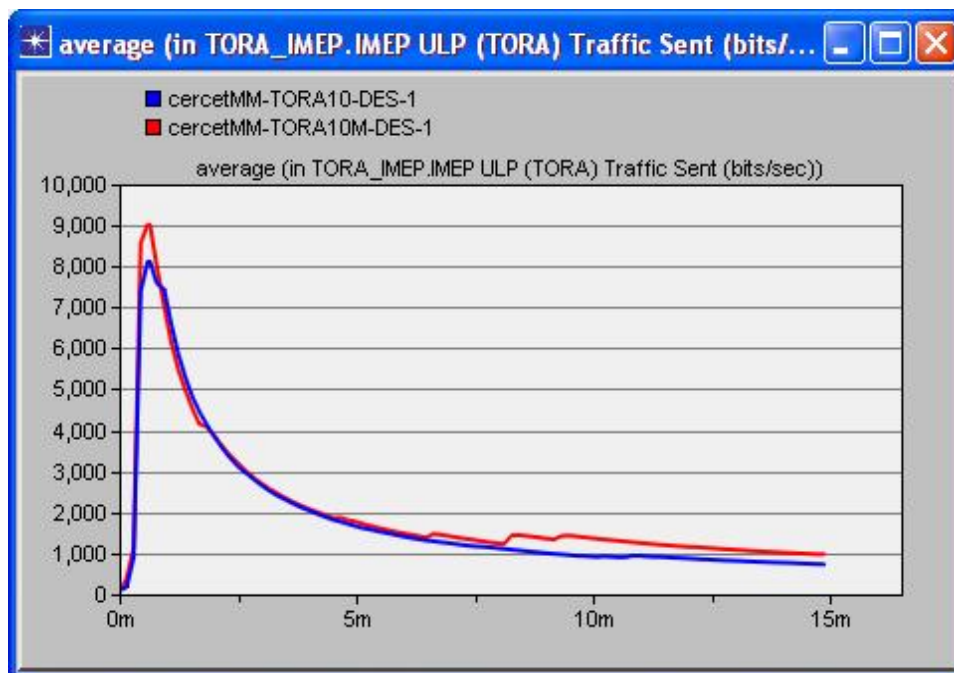
In figura de mai jos, observam ca mobilitatea nu are un efect pronuntat asupra traficului de rutare in cazul unui numar mic de noduri. In cazul retelelor mobile cu 42 de noduri traficul de rutare atinge valori destul de mari.



Trafic de rutare AODV → 10, 40 noduri fixe, mobile

3.2.1.3 Efectele numarului de noduri si mobilitatii in TORA

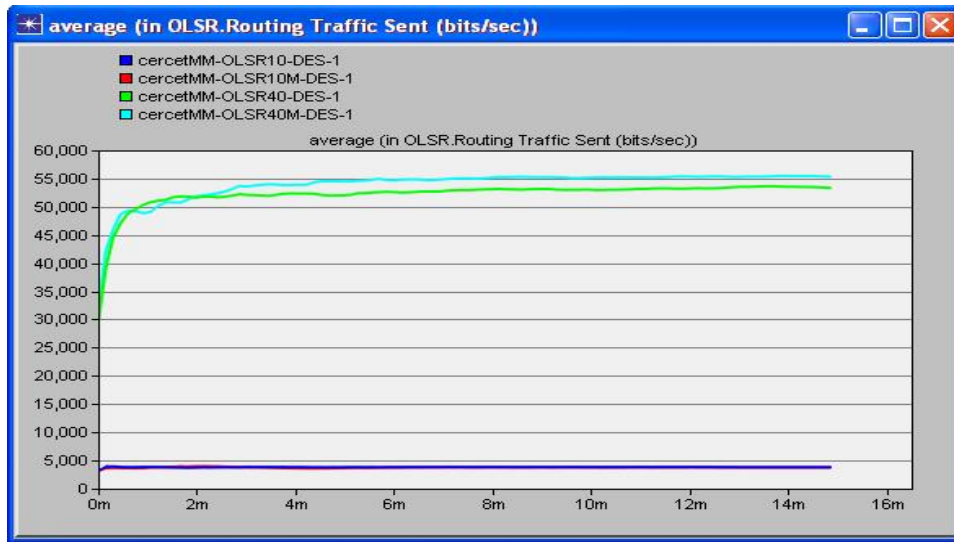
Traficul de rutare este impartit in 2 surse de trafic. Prima sursa este traficul IMEP, care este un model constant de trafic, si traficul de rutare generat numai atunci cand este nevoie de o ruta catre destinatie. Datorita folosirii protocolului IMEP ca un subnivel al protocolului TORA se genereaza o mare cantitate de trafic de rutare. Acest lucru este inutil in cazul scenariilor fixe, deoarece chiar daca nodurile nu isi modifica pozitia, acelasi trafic este generat.



Trafic de rutare TORA → 10 noduri fixe, mobile, putere de transmisie 0.001W

3.2.1.4 Efectele numarului de noduri si mobilitatii in OLSR

In figura de mai jos este prezentat traficul total de rutare in cazul protocolului OLSR. Observam ca mobilitatea nu are un efect prea mare asupra traficului de rutare introdus in retea. Consistenta traficului de rutare se datoreaza caracterului sau proactiv. Cresterea numarului de noduri are ca efect cresterea traficului de rutare dupa cum era de asteptat.



Trafic de rutare OLSR → 10, 40 noduri fixe, mobile

Tabela de rutare a protocolului OLSR poate fi vizualizata din meniul DES → Results → View results → DES Run Table. Nodurile aflate la cea mai mare distanta de server, au cea mai mare metrica, pachetele parcurgand 5 hopuri pentru a ajunge la destinatie.

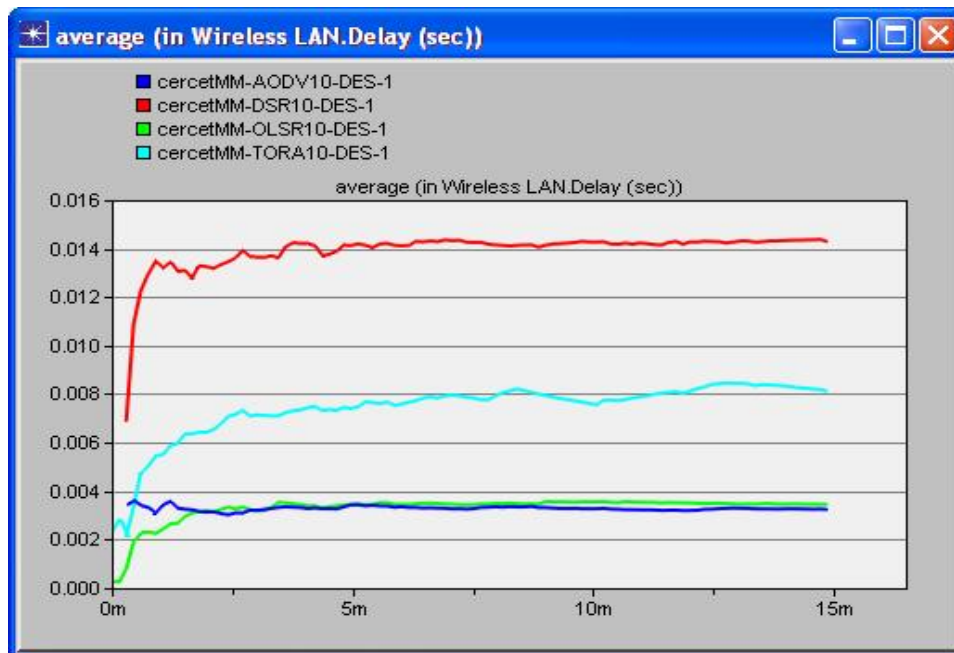
Destination	Source Protocol	Route Preference	Metric	Next Hop Address	Next Hop Node	Outgoing Interface	Outgoing LSP	Insertion Time (secs)
1 192.01.1/32	OLSR	1	5	192.0.1.13	Campus Network.mobile_node_10	FTP Server	N/A	1496.000
2 192.01.2/32	OLSR	1	4	192.0.1.11	Campus Network.mobile_node_10	FTP Server	N/A	1466.000
3 192.01.3/32	OLSR	1	5	192.0.1.13	Campus Network.mobile_node_10	FTP Server	N/A	1491.000
4 192.01.4/32	OLSR	1	4	192.0.1.11	Campus Network.mobile_node_10	FTP Server	N/A	1466.000
5 192.01.5/32	OLSR	1	3	192.0.1.13	Campus Network.mobile_node_10	FTP Server	N/A	1466.000
6 192.01.6/32	OLSR	1	4	192.0.1.11	Campus Network.mobile_node_10	FTP Server	N/A	1466.000
7 192.01.7/32	OLSR	1	4	192.0.1.13	Campus Network.mobile_node_10	FTP Server	N/A	1491.000
8 192.01.0/32	OLSR	1	2	192.0.1.13	Campus Network.mobile_node_10	FTP Server	N/A	2.000
9 192.01.9/32	OLSR	1	5	192.0.1.13	Campus Network.mobile_node_10	FTP Server	N/A	1497.000
10 192.01.10/32	OLSR	1	1	192.0.1.13	Campus Network.mobile_node_10	FTP Server	N/A	2.000
11								
12	Gateway of last resort is not set							
13								

Tabela de rutare OLSR la sfarsitul simularii

In figurile de mai jos este prezentata media intarzierilor end-to-end pentru fiecare protocol in parte.

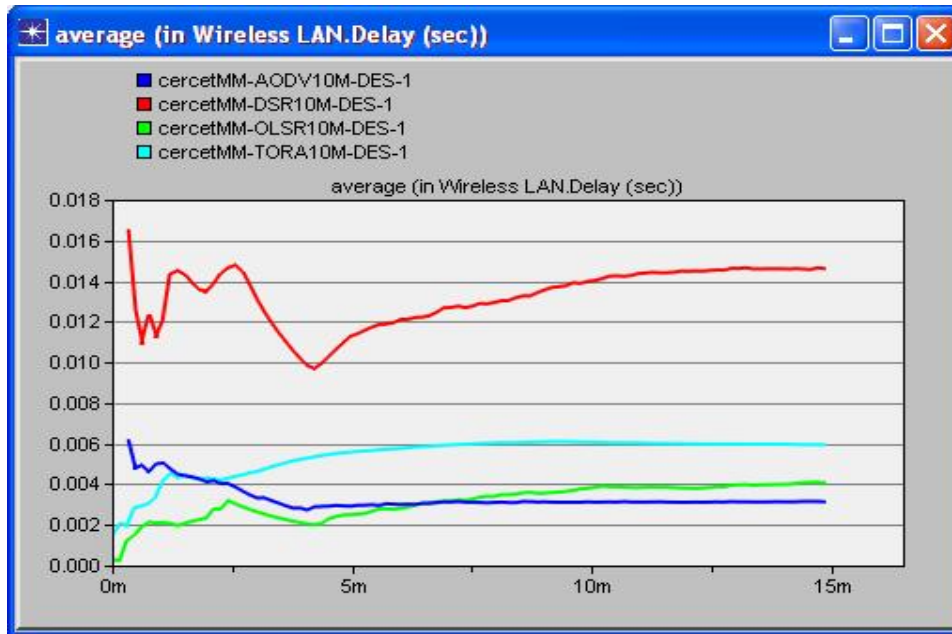
In toate scenariile considerate, observam ca OLSR-ul are cea mai mica intarziere. OLSR este un protocol proactiv, ceea ce inseamna ca rutele in retea sunt mereu gata atunci cand nivelul aplicatie are trafic de transmis. Update-uri periodice catre toate nodurile ale schimbarilor aparute in retea duc la o tabela de rutare "proaspata". In cazul folosirii unui numar mare de noduri performanta lui AODV in ceea ce priveste intarzierea in retea se apropie de OLSR.

In retelele mici, observam ca TORA are o intarziere mai mica decat DSR atat in cazul retelelor fixe cat si mobile. In cazul nodurilor mobile se distanteaza tot mai mult de DSR, apropiindu-se ca valori de AODV si OLSR. Dupa cum arata figurile de mai jos, TORA este un protocol ce poate fi folosit in retelele mobile de dimensiuni reduse.



Intarziere AODV, DSR, OLSR, TORA → 10 noduri fixe

In cazul retelelor mobile, protocolul DSR prezinta la inceput o intarziere mai mare, datorita timpului in care acesta construiește rutele catre destinatie. Si protocolul OLSR prezinta o intarziere mai mare in cazul mobilitatii, datorita timpului mai mare de construire a tabeli de rutare. Din graficul de mai jos reiese ca protocolul AODV poate fi folosit cu succes in retelele mobile de dimensiuni mici, in cazul in care intarzierea in retea este un factor critic.

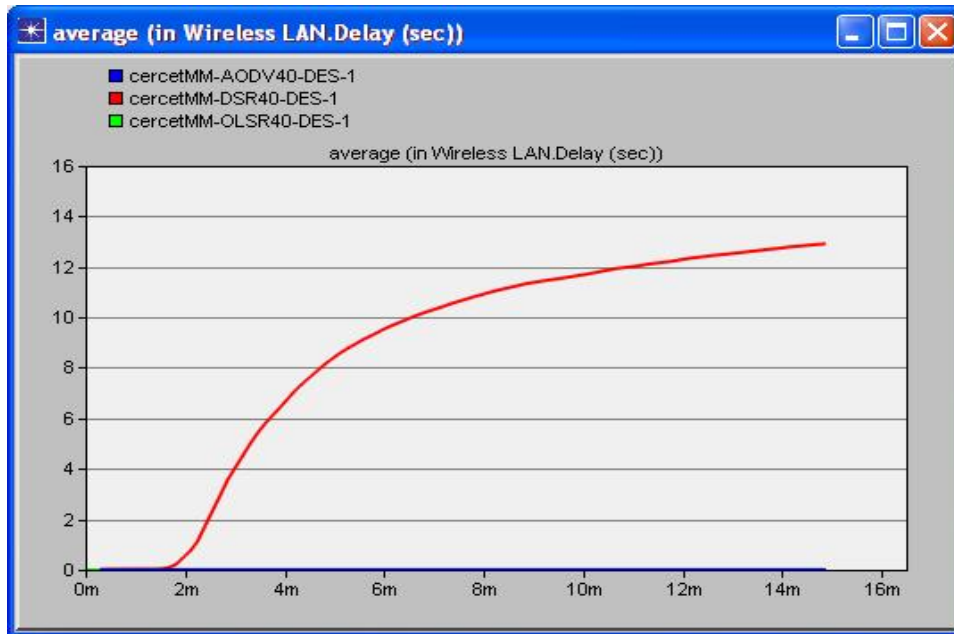


Intarziere AODV, DSR, OLSR, TORA → 10 noduri mobile

Pentru rețele mari, TORA întâmpina mari probleme în cazul congestiilor apărute în rețea forțând astfel nodurile să declare linkurile picate, chiar dacă ele încă există, deoarece se creează bucle în care numărul pachetelor trimise duc la coliziuni la nivelul MAC, iar pachetele “HELLO” și ACK (acknowledge) se pierd, ducând la ipoteza că acele linkuri vecine au fost întrerupte de către IMEP. Prin urmare, TORA reacționează la eșecuri trimițând mai multe update-uri, care la rândul lor duc la o congestie mult mai mare, și implicit la o întârziere mare.

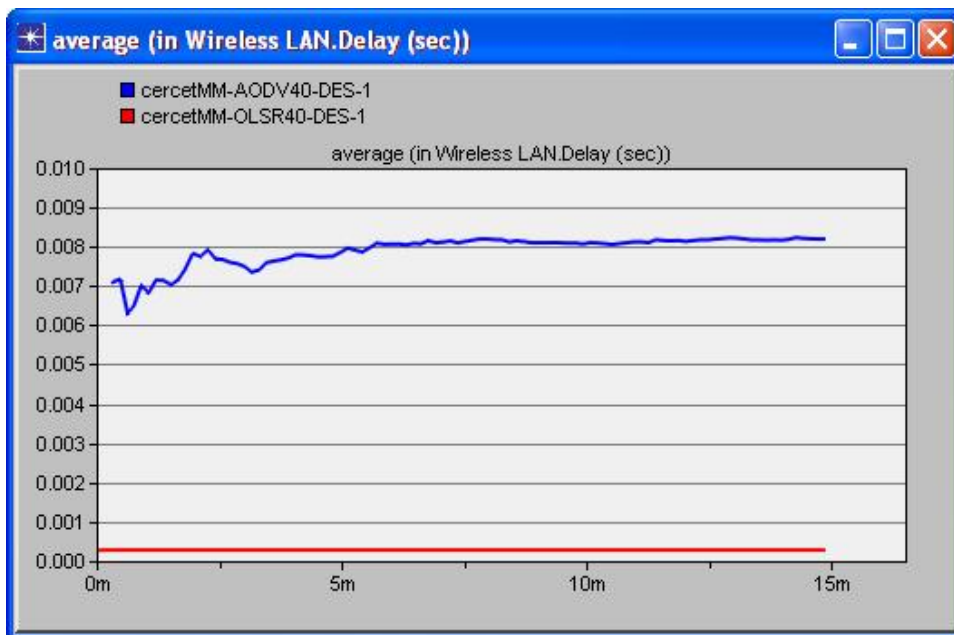
Și în cazul folosirii unui număr mare de surse de trafic, AODV surclasează protocolul DSR, deoarece nodul destinație răspunde doar la primul pachet RREQ, ceea ce are ca rezultat favorizarea traseului cel mai puțin aglomerat. DSR-ul, pe de altă parte trimite răspunsuri la toate pachetele RREQ sosite, ceea ce duce la o întârziere destul de mare. Inițializarea nod cu nod în AODV duce la reducerea întârzierii.

Protocolul DSR prezintă o întârziere consistentă în rețelele mici, atât în cazul nodurilor fixe, cât și mobile. Există o mare probabilitate ca pachetele să fie pastrate în buffer până este descoperită o cale către destinație. La folosirea unui număr de 40 de surse, întârzierea prin rețea ia valori foarte mari. DSR folosește rutele aflate în „cache”, trimițând uneori traficul pe rute vechi, care pot provoca retransmisii ducând la întârzieri excesive. Astfel, în rețelele cu trafic mare, numărul mare de rute salvate în “cache” agravează întârzierea. Pe de altă parte, DSR încearcă să minimizeze efectul rutelor “învechite” utilizând multiple cai. Cel mai probabil explicația pentru întârzierea de ordinul secundelor în cazul lui DSR, se datorează unei congestii. În cazul figurii de mai jos OLSR-ul prezintă o întârziere medie de 0.09s, iar AODV se apropie de 0.014s.



Intarziere AODV, DSR, OLSR → 40 noduri fixe

Pentru o mai buna vizualizare a intarzierii in retea a protocoloalelor AODV si OLSR, le-am reprezentat doar pe acestea intr-o scala mai potrivita. Rezultatul se poate vedea in figura de mai jos.



Intarziere OLSR, AODV → 40 noduri fixe

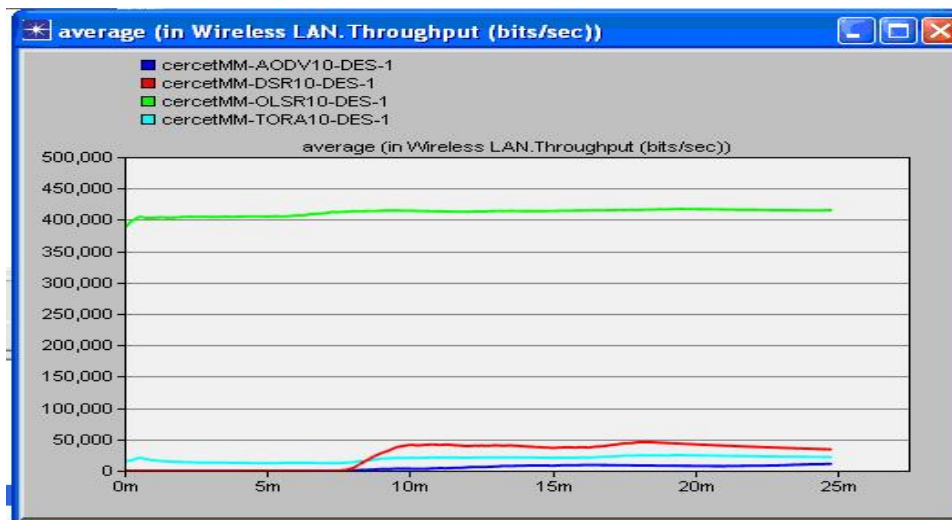
In situatia in care un nod al retelei deduce (prin *ascultarea mediului*) ca ar provoca o coliziune, desi destinatarul lui nu se afla in domeniul de transmisie al altui nod care transmite deja, acesta se opreste din transmisie pentru evitarea coliziunilor, ducand astfel la cresterea intarzierii in retea. Se poate observa ca aceasta situatie se regaseste in figura de mai jos, deoarece nivelul mic de putere transmisa reduce astfel si detectarea unui potential nod interferent.

Pentru a incheia aceasta sub-sectiune vom aminti pe scurt concluziile trase. Am observat ca OLSR-ul are cea mai mica intarziere in toate scenariile. TORA are o intarziere consistenta in cazul unei retele incarcate, mobilitatea neavand un efect foarte mare asupra intarzierii. AODV prezinta o intarziere destul de mica in cazul unui trafic mare in care mobilitatea nu a avut un efect semnificativ. Pe ultimul loc se claseaza DSR-ul care sufera o degradare destul de mare odata cu crestea retelei. Cele trei protocoale reactive prezinta intarzieri consistente in cadrul retelelor mari datorita cererilor de descoperire a traseului.

3.2.2 Throughput

In figurile de mai jos sunt prezentate performantele protocoalelor, din perspectiva numarului de pachete ajunse la destinatie. Observam ca OLSR este de departe cel mai bun protocol in aproape toate scenariile considerate. Fiind un protocol proactiv, rutele sunt pregatite dinaintea inceperii trimiterii traficului, ducand dupa cum am vazut la o intarziere a pachetelor end-to-end foarte mica. Deoarece throughput-ul este raportul dintre cantitatea totala de date pe care un receptor o primeste de la emitator si timpul total in care acesta traverseaza retea, o intarziere mica duce la un throughput mare.

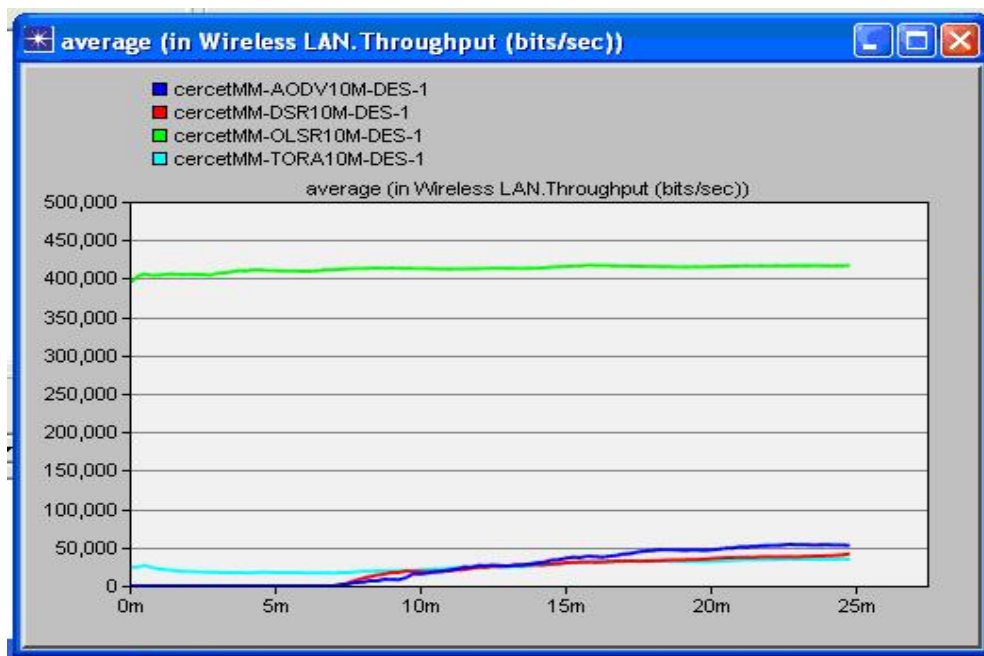
In scenariul in care se folosesc 10 noduri, protocolul DSR are un throughput mai bun decat AODV si TORA atat in cazul nodurilor fixe, chiar daca, dupa cum am vazut in subcapitolul anterior prezinta cea mai consistenta intarziere in retea.



Throughput AODV, DSR, OLSR, TORA →10 noduri fixe

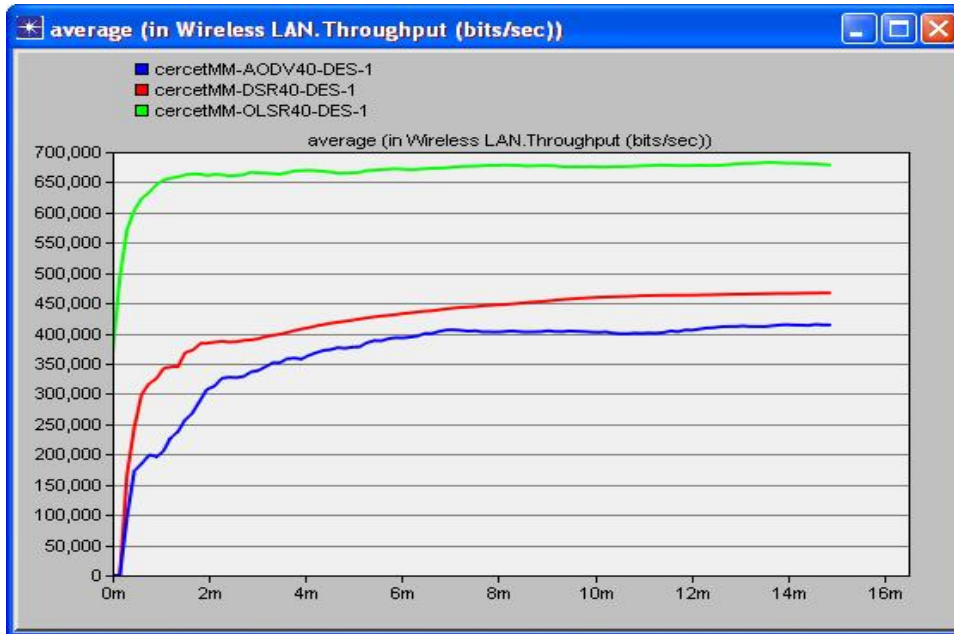
Aceasta abatere poate fi explicata prin observarea traficului de rutare. DSR prezinta cel mai mic trafic de rutare. Avand in vedere ca reseaua este mica probabilitatea intreruperii unui link este slaba, iar alti factori cum ar fi problema terminalului ascuns sau a congestiei sunt exclusi. Protocolul DSR a fost capabil sa livreze pachetele mult mai bine decat AODV, deoarece a avut deja in "cache" rutele catre destinatie, nemaiaivand nevoie de o noua redescoperire. TORA prezinta valori mai bune decat protocolul AODV, probabil deoarece a creat o diagrama a topologiei rețelei, inainte de inceperea transmiterii, fiind astfel capabila de o transmitere mai eficienta a pachetelor.

Cand comparam performanta individuala a fiecarui protocol din perspectiva mobilitatii si incarcarii, observam ca DSR se comporta mai bine , in cazul rețelei cu noduri fixe, din punctul de vedere al pachetelor ajunse la destinatie.



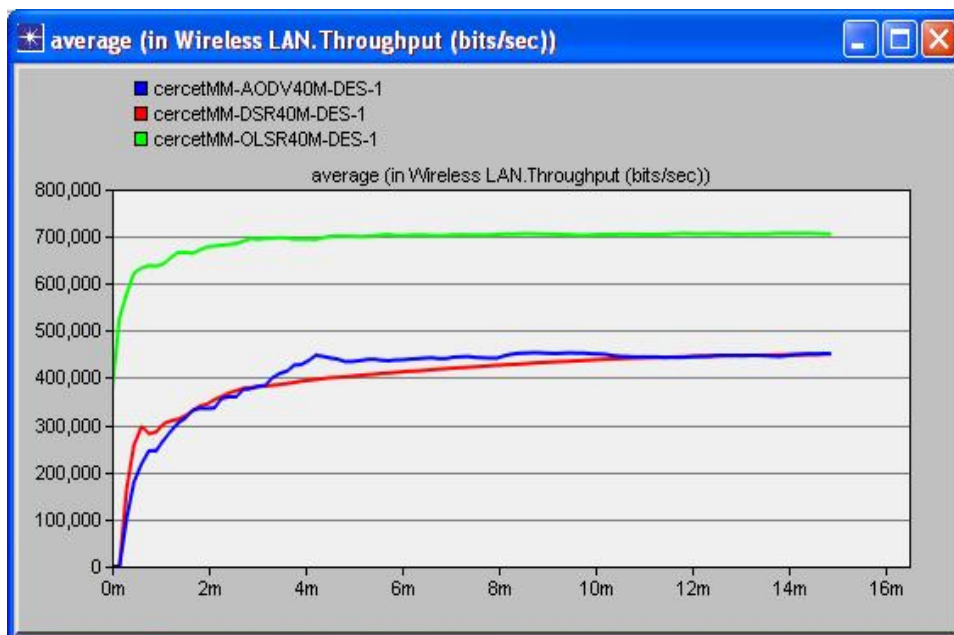
Throughput AODV, DSR, OLSR, TORA →10 noduri mobile

Odata cu cresterea numarului de surse de trafic problema congestiei si cea a terminalului ascuns pot aparea frecvent. Aceste probleme duc la o reactie diferita a protocolelor, datorita variatiei conditiilor de lucru si a intarzierii, avand un impact puternic asupra throughput-ului . Se observa din figurile de mai jos, ca performanta protocolului DSR scade in cazul unei rețele cu un numar mare de surse de trafic. De la aceste observatii, putem concluziona ca protocolul DSR surclaseaza celelalte 2 protocole (TORA, AODV) pentru rețele mici, dar nu si pentru rețele incarcate.



Throughput AODV, DSR, OLSR, TORA → 40 noduri fixe

AODV este exact opusul lui, functionand mai bine pentru retele cu trafic intens. AODV prezinta valori acceptabile, atat in cazul retelelor mici mobile, cat si in situatia in care reseaua se incarca si mobilitatea creste. In concluzie, protocolul AODV se poate folosi cu succes in retelele mobile de dimensiuni mari.



Throughput AODV, DSR, OLSR, TORA → 40 noduri mobile

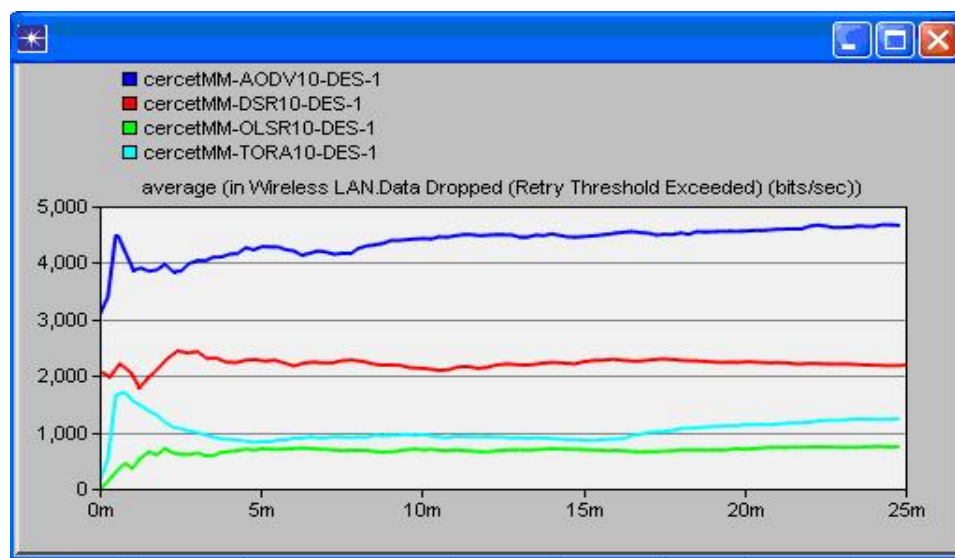
Performanta protocolului TORA in cazul retelelor incarcate nu a putut fi evidentiata datorita eroriilor aparute in timpul simularii

OLSR s-a comportat foarte bine, in toate situatiile. Dupa cum s-a demonstrat in randurile de mai sus, acesta prezinta un trafic mare de rutare si o intarziere mica. Deoarece throughput-ul este o functie de ambele metrici, este de asteptat sa aiba valori bune. Trebuie totusi mentionat ca dupa anumite valori ale numarului de noduri si a mobilitatii, performantele acestuia se degradeaza treptat datorita tabelii de rutare foarte mare.

3.2.3 Rata de pachete pierdute

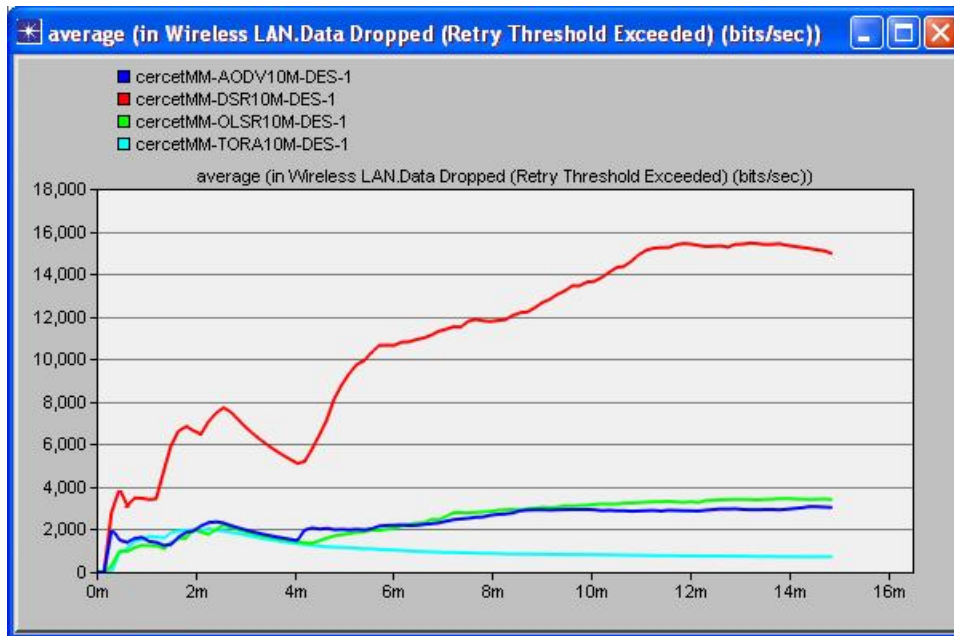
Figura de mai jos prezinta rata de pachete pierdute pentru cele 4 protocoale in scenariul cu 10 noduri fixe. OLSR, TORA si DSR prezinta un numar mic de pachete pierdute, pe intreg parcursul simularii. Acest lucru se datoreaza faptului ca atat DSR cat si TORA pastreaza mai multe trasee catre aceeasi destinatie si in cazul in care un link dispare, pachetele inca mai pot fi trimise la destinatie pe celelalte rute. Spre deosebire de DSR, AODV pastreaza doar rutele active, cele inechitate fiind eliminate. Astfel, in cazul lipsei rutelor alternative, acesta trimite nodului sursa o cerere de descoperire a rutei. In situatia in care mai multe surse trimit pachete catre aceeasi destinatie, congestia pe ruta catre singura destinatie, blocheaza pachetele de descoperire a unei noi rute (RREQ). In cazul in care nodurile intermediare nu pot trimite pachetele mai departe, AODV trimite "route error" pentru gasirea unui nou traseu catre sursa, care va avea ca efect aruncarea pachetelor, ducand la degradarea performantelor protocolului.

OLSR prezinta cea mai mica rata de pachete pierdute, datorita caracterului sau proactiv, el avand rute catre destinatie inainte de a se incepe transmiterea.



Rata de pachete pierdute AODV, DSR, OLSR, TORA → 10 noduri fixe

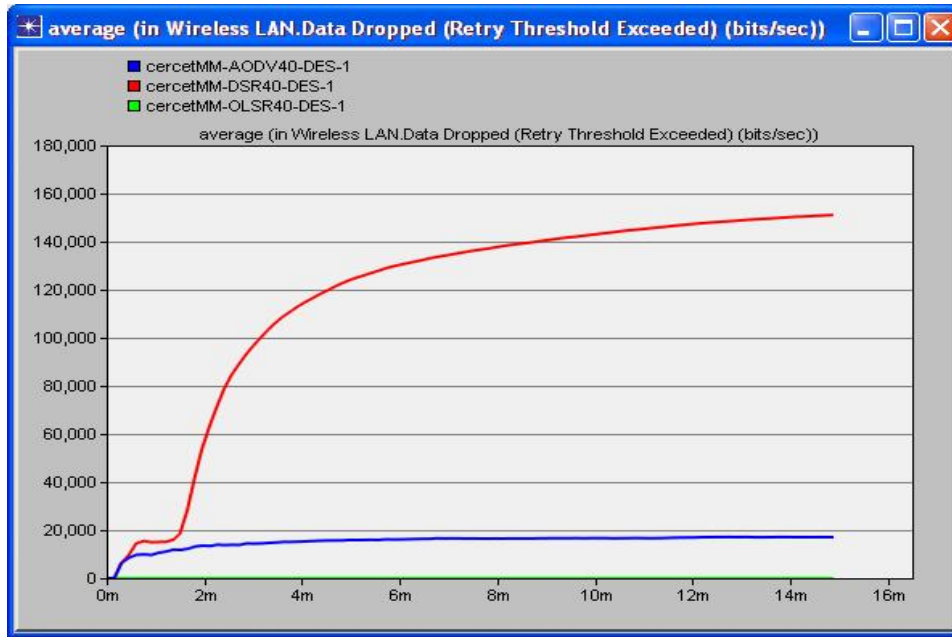
In cazul nodurilor mobile, situatia se schimba. Protocolul DSR prezinta cea mai mare rata de pachete pierdute, datorita rutelor “invechite” pe care acesta le foloseste pentru transmiterea pachetelor. AODV prezinta o performanta mai buna in cazul nodurilor mobile, deoarece pastreaza numai rutele “proaspete”. Si protocolul OLSR prezinta o mica degradare in cazul nodurilor mobile, datorita variatiei link-urilor, OLSR avand o convergenta mai lenta, timpul pentru refacerea tabelului de rutare fiind relativ mare, perioada in care pachetele sunt aruncate. Protocolul TORA se comporta foarte bine in scenariul mobil, probabil datorita multiplelor rute catre destinatie. Acest lucru se poate vedea si din figura de mai jos:



Rata de pachete pierdute AODV, DSR, OLSR, TORA →10 noduri mobile

In cazul in care numarul nodurile creste, si intervine mobilitatea, AODV prezinta la inceput cel mai mare numar de pachete aruncate, datorita inexistentei rutelor catre serverul FTP. Lipsa de rute alternative, duce la o noua actiune a nodului sursa de a descoperi rute valide. In situatia in care toate sursele trimit trafic catre aceeasi destinatie, procesul de redescoperire a rutei este intarziat, datorita congestiilor aparute pe singurele rute catre destinatie. In aceste situatii, nodurile arunca aceste pachete. Pe parcursul simularii protocolul DSR elimina cele mai numeroase pachete, atat in cazul nodurilor mobile, cat si fixe.

Protocolul OLSR, se comporta si in acest caz cel mai bine, datorita caracterului sau proactiv, el avand tabela de rutare creata in momentul inceperii transmisiunii.



Rata de pachete pierdute AODV, DSR, OLSR →40 noduri fixe, mobile

Posibile dezvoltari ale lucrarii :

1. Modificarea puterii de transmisie a nodurilor, astfel incat sa poata comunica cu 2 noduri vecine.
2. Simularea parametrilor in cazul folosirii unui trafic generat de noduri mult mai mare.
3. Folosirea nodurilor multi-channel pentru cresterea debitului binar de date.