

**ANALIZA PERFORMANTELOR PROTOCOLULUI
AODV**

Obiectivele acestei platforme de laborator sunt urmatoarele:

- **Prezentare teoretica a modului de functionare a protocolului de rutare AODV: descoperirea rutelor, mentenanta rutelor si a tabelelor de rutare, formatul mesajelor etc;**
- **Prezentare si introducere in mediul de simulare OPNET;**
- **Modelarea unei retele ad-hoc in OPNET;**
- **Interpretarea rezultatelor simulate si concluzii.**

I) Prezentare teoretica a protocolului de rutare AODV

Protocolul AODV permite rutarea multihop între mai multe noduri care doresc să comunice prin intermediul unei rețele ad hoc . AODV se bazează pe un algoritm de tip vector distanță (distance vector) . Spre deosebire de alte protocoale de acest tip, AODV este reactiv și nu proactiv . El nu menține în permanenta rute către destinațiile care nu sunt folosite în acel moment pentru comunicare și contruiește rute numai atunci când ele sunt necesare. Printre altele, AODV poate suporta multicast cu ajutorul unei extensii și evită problema 'numărării până la infinit' pe care o au protocoalele bazate pe algoritmul Bellman-Ford. Protocolul folosește mesaje diferite pentru descoperirea și mentenanța rutelor .

1.1. Descoperirea rutelor

Când un nod are nevoie de o rută către o destinație, pentru care nu detine informații, un mesaj de tip ROUTE REQUEST este difuzat către toți vecinii . Acest lucru se poate întâmpla atunci când destinația este necunoscută, sau atunci când ruta anterioară a expirat . După difuzarea unui astfel de pachet, sursa așteaptă un mesaj ROUTE REPLY. Dacă acesta nu este recepționat într-o anumită perioadă de timp, sursa poate redifuză mesajul ROUTE REQUEST sau poate presupune că nu există nici o rută către respectiva destinație .

Când un nod primește un ROUTE REQUEST și nu are o rută către respectiva destinație, el redifuzează respectivul pachet. Nodul creează și o rută inversă către

sursă, în care următorul hop este nodul de la care a primit mesajul. Acest lucru este folosit pentru a avea o rută către nodul care a inițiat căutarea, în cazul în care un pachet ROUTE REPLY va trebui să ajungă la sursa cererii . Aceste rute sunt temporare, în sensul că sunt păstrate valide mult mai puțin decât o rută normală .

Când un mesaj ROUTE REQUEST ajunge fie la destinație, fie la un nod care cunoaște o rută validă către aceasta , un mesaj ROUTE REPLY este creat și trimis ca unicast către sursa cererii. Pe măsură ce pachetul trece prin nodurile intermediare, o rută către destinație este creată, iar când mesajul a ajuns la nodul sursă, avem deja o rută contruită .

1.2. Mentenanța rutelor

Când un nod detectează că o rută către un vecin nu mai este validă, el va elimina ruta din tabela proprie și va trimite un mesaj ROUTE ERROR către vecinii săi, care folosesc ruta în momentul respectiv, prin care îi informează că respectiva rută nu mai este validă. În acest scop AODV menține o listă cu vecinii activi care folosesc o anumită rută. Nodurile care recepționează acest mesaj vor repeta procedura, iar în final toate sursele care au fost afectate vor afla despre întrerupere .

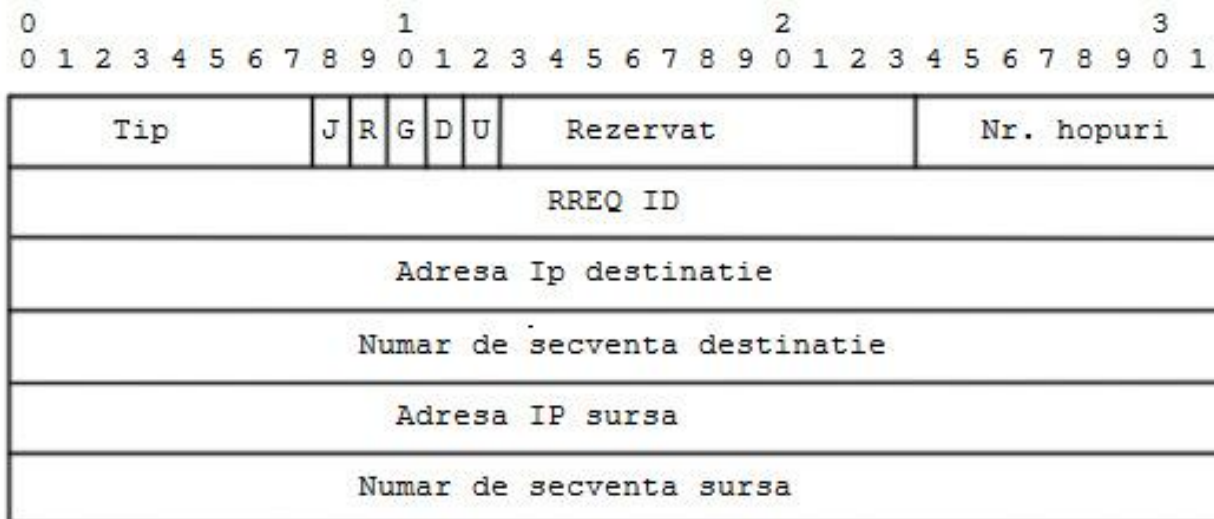
1.3. Tabela de rutare

Pentru fiecare intrare din tabela de rutare AODV păstrează următoarele informații:

- Adresa IP a destinației;
- Numărul de secvență al destinației;
- Numărul de hopuri până la destinație;
- Următorul hop – vecinul care a fost desemnat pentru transmiterea pachetelor către respectiva destinație;
- Durata de viață – durata de timp pentru care ruta rămâne validă;
- Lista vecinilor activi – vecinii care folosesc acea rută la acel moment.

2. Formatul mesajelor folosite in protocolul AODV

2.1. Formatul mesajelor Route Request (RREQ)

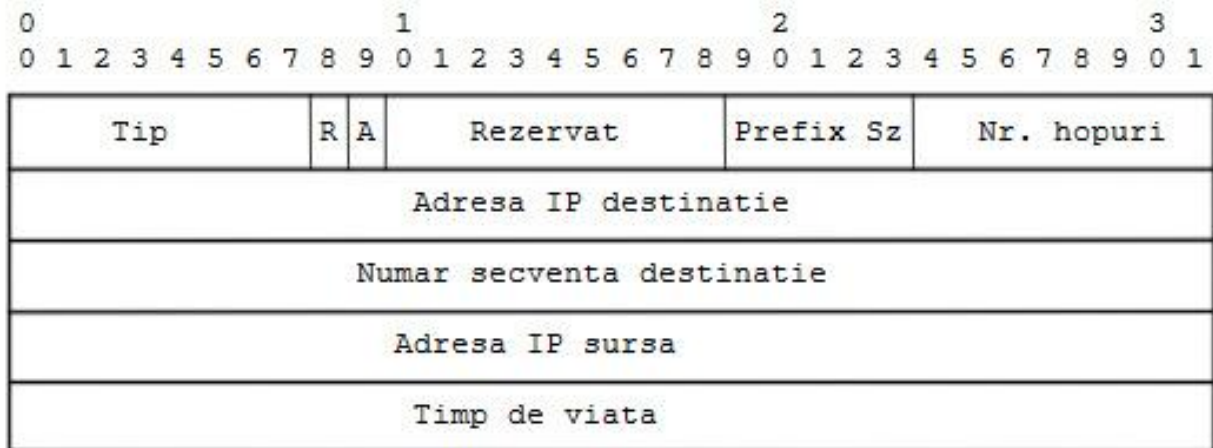


Formatul mesajelor **Route Request** este ilustrat mai sus, si contine urmatoarele campuri:

- Tip – 1
- J – flagul Join, rezervat pentru multicast;
- R – flagul Repair, rezervat pentru multicast;
- G – flagul pentru Route Reply Gratuit, indica daca un mesaj RREP gratuit trebuie trimis ca unicast catre adresa IP destinatie;
- D – flagul pentru Destinatie, indica faptul ca numai destinatia poate raspunde la acest pachet RREQ;
- U – numar de secventa necunoscut, indica faptul ca numarul de secventa al destinatiei nu este cunoscut;
- Rezervat – trimis 0, ignorat la receptie;
- Numarul de hopuri – numarul de hopuri de la IP-ul adresei sursa pana la nodul care proceseaza acest pachet;
- RREQ ID – numarul de secventa care impreuna cu adresa IP a sursei identifica in mod unic pachetul RREQ;

- Adresa IP destinatie – adresa IP a destinatiei pentru care se doreste aflarea unei rute;
- Numar de secventa destinatie – ultimul numar de secventa receptionat in trecut de catre sursa acestui pachet de la respectiva destinatie;
- Adresa IP sursa – adresa IP a nodului care a initiat pachetul RREQ;
- Numar de secventa sursa – numarul de secventa curent care trimite la originatorul pachetului RREQ

2.2. Formatul mesajelor Route Reply (RREP)



Formatul mesajelor **Route Reply**, ilustrat mai sus contine urmatoarele campuri:

- Tip – 2;
- R – flagul Repair, folosit pentru multicast;
- A – confirmare necesara;
- Prefix size – daca este diferit de 0, cei 5 biti ai acestui camp specifica ca urmatorul nod poate fi folosit de toate nodurile cu acelasi prefix de rutare ca adresa IP destinatie;
- Numar hopuri – Numarul de hopuri de la adresa IP sursa pana la adresa IP destinatie.
- In cazul unei rute de multicast, indica numarul de hopuri pana la membrul arborelui de multicast care a originat mesajul RREQ;

- Adresa IP destinatie – adresa IP a destinatiei pentru care se ofera ruta ;
- Numar de seventa destinatie – numarul de secventa asociat rutei;
- Adresa IP sursa – adresa IP a nodului care a initiat pachetul RREQ;
- Timp de viata – timpul in milisecunde pentru care nodurile care receptioneaza pachetul RREP considera ruta valida;

2.3.Formatul mesajelor Route Error (RERR)

0			1							2							3														
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Tip			N	Rezervat														ContDest													
Destinatie IP indisponibila (1)																															
Numar de secventa al destinatiei indisponibile (1)																															
Destinatie IP indisponibila aditionala (daca este necesar)																															
Numar de secventa al destinatiei indisponibile aditionala																															

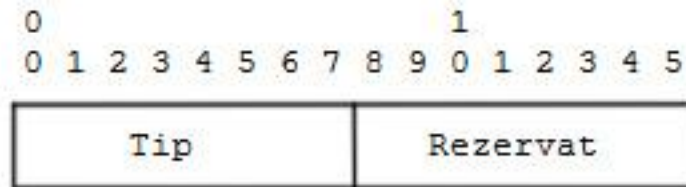
Formatul mesajelor **Route Error** ilustrat mai sus contine urmatoarele campuri:

- Tip – 3;
- N – flagul No Delete, setat atunci cand un nod a realizat o reparare locala a unui link, iar nodurile din upstream nu trebuie sa stearga aceasta ruta;
- Rezervat – trimis 0, ignorat la receptie;
- Contor Destinatiei – numarul de destinatii indisponibile continut in mesaj, trebuie sa fie minim 1;
- Destinatie IP indisponibila – adresa IP care a devenit indisponibila datorita intreruperii unui link;
- Numar de secventa al destinatiei indisponibile – numarul de secventa din tabela de rutare pentru destinatia listata in campul anterior.

Mesajul RERR este trimis oricand intreruperea unui link cauzeaza ca una sau mai multe destinatii sa devina indisponibile pentru vecini.

2.4. Formatul mesajelor Route Reply Acknowledgment (RREP-ACK)

Mesajele **RREP-ACK** trebuie trimise ca raspuns la un mesaj RREP, care are bitul 'A' setat, acest lucru fiind de obicei folosit atunci cand exista pericolul ca un link unidirectional sa previna incheierea unui ciclu de descoperire a rutei.



- Tip – 4
- Rezervat – trimis 0, ignorat la receptie;

3. Modul de functionare al protocolului AODV

In aceasta sectiune sunt descrise scenariile in care nodurile genereaza mesaje de tipul **Route Request (RREQ)**, **Route Reply (RREP)** si **Route Error (RERR)** pentru o comunicatie unicast spre o anumita destinatie, si modul in care datele sunt prelucrate. Pentru a procesa corect aceste mesaje, anumite informatii de stare trebuie mentinute in tabela de rutare pentru destinatiile de interes. Toate mesajele AODV sunt trimise pe portul 654 folosind UDP.

3.1. Mentenanta numerelor de secventa

Fiecare intrare in tabela de rutare a fiecarui nod trebuie sa contina ultima informatie disponibila despre numarul de secventa pentru adresa IP a nodului destinatie pentru care intrarea in tabela de rutare este mentinuta.

Acest numar de secventa este numit '**numar de secventa destinatie**'. Acesta este updatat de fiecare data cand nodul primeste informatii noi despre numarul de secventa de la pachete RREQ, RREP, sau RERR care au legatura cu respectiva destinatie. AODV depinde de fiecare nod din retea sa isi mentina numerele de secventa destinatie pentru a garanta ca nu exista bucle catre respectiva destinatie. Un nod destinatie isi incrementeaza numarul de secventa doar in doua circumstante :

- Imediat inainte de a initia un ciclu de descoperire a rutei, el trebuie sa isi incrementeze numarul de secventa propriu. Aceasta previne conflictele cu rutele stabilite anterior catre sursa pachetului RREQ.
- Imediat inainte ca un nod destinatie sa trimita un RREP ca raspuns la un RREQ, el trebuie sa isi updateze numarul de secventa cu maximul dintre numarul curent de secventa propriu si numarul de secventa destinatie din pachetul RREQ.

Un nod poate schimba numarul de secventa pentru o intrare din tabela de rutare numai daca :

- El este nodul destinatie si ofera o noua ruta catre el insusi
- Receptioneaza un mesaj AODV cu informatii mai noi despre numarul de secventa al destinatiei.
- Calea catre respectivul nod expira sau se intrerupe.

3.2. Tabela de rutare si lista de precursori

Cand un nod receptioneaza un pachet de control AODV de la un vecin, sau creaza sau updateaza o ruta catre o anumita destinatie, verifica daca exista o intrare in tabela de rutare pentru respectiva destinatie. In cazul in care nu exista nici o intrare in tabela de rutare pentru respectiva destinatie aceasta este creata. Numarul de secventa este determinat din informatiile continute de pachetul AODV, altfel numarul de secventa este setat false. Ruta este updata numai in cazul in care numarul de secventa este :

- Mai mare decat numarul de secventa destinatie din tabela de rutare;
- Numarul de secventa este egal, dar numarul de hopuri plus 1 este mai mic decat numarul de hopuri din tabela de rutare;
- Numarul de secventa nu este cunoscut.

Campul 'timp de viata' este fie determinat din pachetul de control, fie initializat cu valoarea ACTIVE_ROUTE_TIMEOUT. Timpul de viata pentru o ruta activa este updatat de fiecare data cand ruta catre respectiva destinatie este utilizata.

Pentru fiecare ruta valida mentinuta de catre un nod ca o intrare in tabela de rutare se mentine si o lista de precursori, care transmit pachete pe aceasta ruta. Acesti precursori vor fi instiintati in cazul in care se detecteaza intreruperea linkului catre hopul urmator.

3.3. Generarea unui pachet Route Request

Un nod genereaza un pachet Route Request atunci cand determina ca are nevoie de o ruta catre o anumita destinatie, si nu exista una disponibila. Acest lucru se poate intampla in cazul in care destinatia respectiva este necunoscuta nodului, sau daca o ruta valida pentru respectiva destinatie a expirat sau este marcata ca invalida. Campul 'numar secventa destinatie' este completat cu valoare din campul ' numar secventa destinatie' al tabelii de rutare. Daca nu este cunoscut nici un numar de secventa, flagul 'U' trebuie setat. 'Numarul de secventa sursa' este numarul propriu de secventa, care a fost incrementat inainte de crearea mesajului RREQ. Campul RREQ ID este incrementat cu 1 de la ultima valoare folosita de nodul curent. Valoarea campului ' numar de hopuri' este setata 0.

Inainte de a difuza mesajul RREQ, nodul care l-a initiat, memoreaza intr-un buffer campurile RREQ ID si adresa IP sursa pentru o valoare de timp PATH_DISCOVERY_TIME. Astfel, atunci cand pachetul receptioneaza pachetul din nou de la vecinii sai, el nu va fi reprocesat si reforwardat.

Dupa ce este transmis un pachet RREQ, nodul asteapta un mesaj RREP, sau un alt mesaj de control cu informatii despre respectiva destinatie. Daca nu se receptioneaza nimic in intervalul de timp NET_TRAVERSAL_TIME, nodul poate incerca sa descopere din nou o ruta catre respectiva destinatie, prin trimiterea unui nou mesaj RREQ, pana la un numar maxim de reincercari egal cu RREQ_TIMES. Fiecare noua incercare trebuie sa incrementeze si sa updateze campul RREQ ID.

Pachetele de date care asteapta o ruta ar trebui introduse intr-un buffer, metoda folosita fiind FIFO (first in, first out). Daca au fost RREQ_RETRIES incercari de descoperire a rutei, cu valoarea TTL maxima si nu a fost gasita nici o ruta, pachetele respective ar trebui aruncate, si un mesaj Destinatie Indisponibila ar trebui trimis la nivel aplicatie. Pentru a reduce congestiile din retea, incercarile repetate de a descoperi o ruta de catre o sursa utilizeaza un algoritm exponential de backoff.

3.4. Procesarea si forwardarea mesajelor RREQ

Atunci cand un nod primeste un mesaj RREQ, primul lucru pe care il face este sa creeze sau sa updateze o ruta catre hopul anterior, fara un numar de secventa valid, apoi se verifica daca a mai primit un pachet cu aceeasi adresa IP sursa si RREQ ID in intervalul de timp PATH_DISCOVEY_TIME. Daca acest lucru s-a intamplat, nodul arunca respectivul pachet. In continuare va fi descris ce se intampla daca acest pachet nu este un duplicat.

In primul rand , nodul incrementeaza numarul de hopuri cu 1, apoi cauta o ruta de intoarcere catre adresa IP sursa. Daca este necesar aceasta ruta este creata, sau updatata folosind 'numarul de secventa sursa' al pachetului RREQ, aceasta ruta fiind necesara in cazul in care pachetul primeste un pachet RREP ce trebuie transmis originatorului pachetului RREQ. Cand ruta este creata sau updatata, urmatoarele actiuni sunt indeplinite:

- A) Numarul de secenta sursa al pachetului RREQ este comparat cu numarul de secventa destinatie al intrarii din tabela si copiat in locul acestuia daca este mai mare.
- B) Campul numar de secventa valid este setat adevarat.
- C) Urmatorul hop din tabela de rutare devine nodul de la care a fost receptionat pachetul (este obtinut din campul adresa IP sursa al antetului pachetului IP si de cele mai multe ori este diferit de campul adresa IP sursa al mesajului RREQ).
- D) Numarul de hopuri este copiat din campul cu acelasi nume al mesajului RREQ. De fiecare data cand un mesaj RREQ este receptionat, campul 'timp de viata' al rutei de intoarcere pentru adresa IP sursa este setat sa fie maximul dintre doua valori 'timp de

viata existent' si 'timp de viata minim' (timpul de viata minim este egal cu 'timp curent'+
 $2*NET_TRAVERSAL_TIME - 2*nr_noduri*HOP_TRAVERSAL_TIME$).

Daca nodul nu trebuie sa trimita un mesaj RREP, iar valoarea campului TTL din antetul pachetului IP este mai mare de 1, nodul updateaza mesajul RREQ si apoi il difuzeaza pe toate interfetele sale cu adresa IP 255.255.255.255. Updatarea mesajului se realizeaza prin decrementarea campului TTL din antetul pachetului, iar campul 'numar de hopuri' este incrementat. Ultimul pas care trebuie facut este ca valoarea campului numar de secventa destinatie sa fie setata la maximul dintre valoarea receptionata in mesajul RREQ si valoarea pastrata in tabela de rutare pentru respectiva destinatie. Totusi, nodul nu trebuie sa isi modifice valoarea pastrata in tabela de rutare, chiar daca valoarea din pachetul RREQ este mai mare.

3.5. Generarea mesajelor Route Reply

Un nod genereaza mesaje Route Reply in doua cazuri:

- El este destinatia pachetului RREQ;
- Are o ruta activa catre respectiva destinatie, numarul de secventa destinatie din tabela de rutare este valid si mai mare sau egal cu valoarea din pachetul RREQ, iar flagul 'D' nu este setat.

Cand genereaza un mesaj RREP, nodul copiaza campurile 'adresa IP destinatie' si 'numar secventa sursa' din pachetul RREQ in campurile corespondente din mesajul RREP. Procesarea este putin diferita, depinzand de faptul ca nodul este destinatia pachetului sau doar un nod intermediar cu o ruta mai proaspata catre respectiva destinatie.

Odata creat, pachetul RREP este trimis ca unicast catre sursa pachetului RREQ, dupa cum indica tabela de rutare. Pe masura ce pachetul inainteaza, campul 'numar de hopuri' este incrementat la fiecare nod intermediar. Astfel, atunci cand pachetul ajunge la sursa cererii, campul 'numar de hopuri' va contine distanta in hopuri de la sursa la destinatie.

3.5.1. Generarea unui mesaj Route Reply de catre destinatie

Daca nodul care genereaza mesajul este chiar destinatia, el trebuie sa isi incrementeze numarul propriu de secventa cu 1 daca aceasta valoare incrementata este egala cu numarul de secventa al mesajului RREQ. Altfel , destinatia nu isi schimba numarul propriu de secventa, inainte de generarea mesajului RREP. Destinatia isi introduce numarul de secventa propriu in campul 'numar secventa destinatie' al mesajului RREP, si introduce valoarea 0 in campul 'numar de hopuri'. Nodul destinatie copiaza valoarea MY_ROUTE_TIMEOUT in campul 'timp de viata' al pachetului RREP.

3.5.2. Generarea unui mesaj Route Reply de catre un nod intermediar

Daca nodul care genereaza pachetul RREP nu este destinatia, ci doar un hop intermediar, el copiaza 'numarul de secventa destinatie' pe care il are in tabela de rutare in campul cu acelasi nume din mesajul RREP.

Apoi nodul copiaza adresa IP a hopului anterior (de la care a receptionat pachetul RREQ) in lista de precursori. Apoi nodul intermediar introduce distanta in hopuri pana la destinatie (indicata de campul cu acelasi nume din tabela de rutare). Campul 'timp de viata' din mesajul RREP este calculat extragand timpul curent din timpul de expirare al rutei.

3.6. Receptionarea si forwardarea mesajelor Route Reply

Cand un nod receptioneaza un mesaj RREP, el cauta o ruta catre hopul anterior. Daca este nevoie, o noua ruta este creata pentru hopul anterior, dar fara un numar de secventa valid. Apoi se incrementeaza campul 'numar de hopuri' din campul mesajului RREP. Apoi ruta de transmitere a mesajul este creata in cazul in care nu exista deja. Altfel, nodul compara numarul secventa destinatie din mesajul cu valoarea proprie pentru adresa IP destinatie din mesajul RREP. Dupa comparatie, in cazul in care aceasta intrare exista deja ea este updatata in urmatoarele circumstante:

- Numarul de secventa din tabela de rutare este marcat ca invalid.
- Numarul de secventa destinatie din RREP este mai mare decat valoarea pe care nodul o are pentru respectiva destinatie.
- Numarul de secventa este identic, dar ruta este marcata ca invalida.
- Numarul de secventa este identic, insa numarul de hopuri este mai mic.

Daca ruta catre destinatie este creata sau updatata, atunci urmatoarele actiuni sunt efectuate:

- Ruta este marcata ca fiind activa.
- Numarul de secventa destinatie este marcat ca fiind valid.
- Urmatorul hop pentru respectiva ruta este atribuit nodului de la care mesajul RREP a fost primit.
- Numarul de hopuri este incrementat.
- Timpul de expirare este setat ca fiind timpul curent + valoarea campului 'timp de viata' din pachet.
- Numarul de secventa destinatie ia valoarea campului 'numar de secventa destinatie' din pachetul RREP.

Nodul curent poate astfel folosi ruta pentru a trimite pachete catre respectiva destinatie.

Daca nodul nu este cel care a trimis cererea RREQ, el isi va consulta tabela de rutare pentru a determina urmatorul hop catre sursa cererii, iar apoi va forwarda respectivul pachet catre urmatorul hop. Daca nodul trimite pachetul pe un link pe care este posibil sa apara erori sau sa fie unidirectional, nodul ar trebui sa seteze flagul 'A', pentru a cere sa i se confirme primirea printr-un mesaj RREP-ACK.

II) Introducere în OPNET Modeler

Acest laborator urmărește însușirea cunoștințelor de bază necesare pentru lucrul cu programul OPNET Modeler. Cu ajutorul acestuia, studenții pot înțelege mai ușor conceptul de rețea, fiind capabili ulterior să rezolve eficient problemele ce pot apărea într-o infrastructură de rețea din lumea reală.

OPNET Modeler oferă un mediu virtual în care poate fi modelată o întreagă rețea, indiferent de dimensiunile și complexitatea ei, incluzând rutere, switch-uri, echipamente terminale, protocoale, servere sau aplicații individuale.

Înainte de a trece la realizarea efectivă a unei rețele este importantă familiarizarea cu modul în care OPNET Modeler lucrează, cu spațiul de lucru și numeroasele editoare. OPNET Modeler dispune de un *Project Editor* care permite crearea modelului de rețea folosind modele din biblioteca standard, alegerea statisticilor care urmează a fi colectate pentru fiecare obiect al rețelei sau pentru întreaga rețea. De asemenea poate fi executată o simulare și pot fi vizualizate rezultatele obținute prin simularea rețelei.

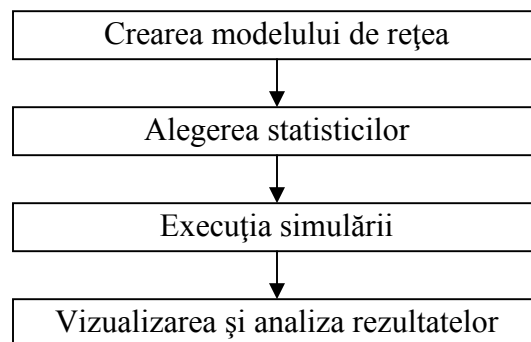


Fig. 1. Etapele de lucru pentru simularea și analiza unei rețele în OPNET Modeler

Pe lângă *Project Editor*, există și o serie de alte editoare specializate, care pot fi accesate din *Project Editor*, utilizate spre exemplu pentru: crearea nodurilor, proceselor, construirea unor formate proprii de pachete de date, aplicarea diverselor filtre sau pentru realizarea unor link-uri specifice (altele decât cele predefinite).

În momentul încărcării unui proiect, fereastra *Project Editor* (figura 2) se va deschide. Această fereastră cuprinde o serie de zone importante necesare pentru construirea și analiza unei rețele: bara de meniuri (*Menu Bar*), bara de unelte (*Tool Bar*), spațiul de lucru (*Workspace*), zona de mesaje (*Message Area*), iconița bufferului de mesaje (*Message Buffer Icon*).

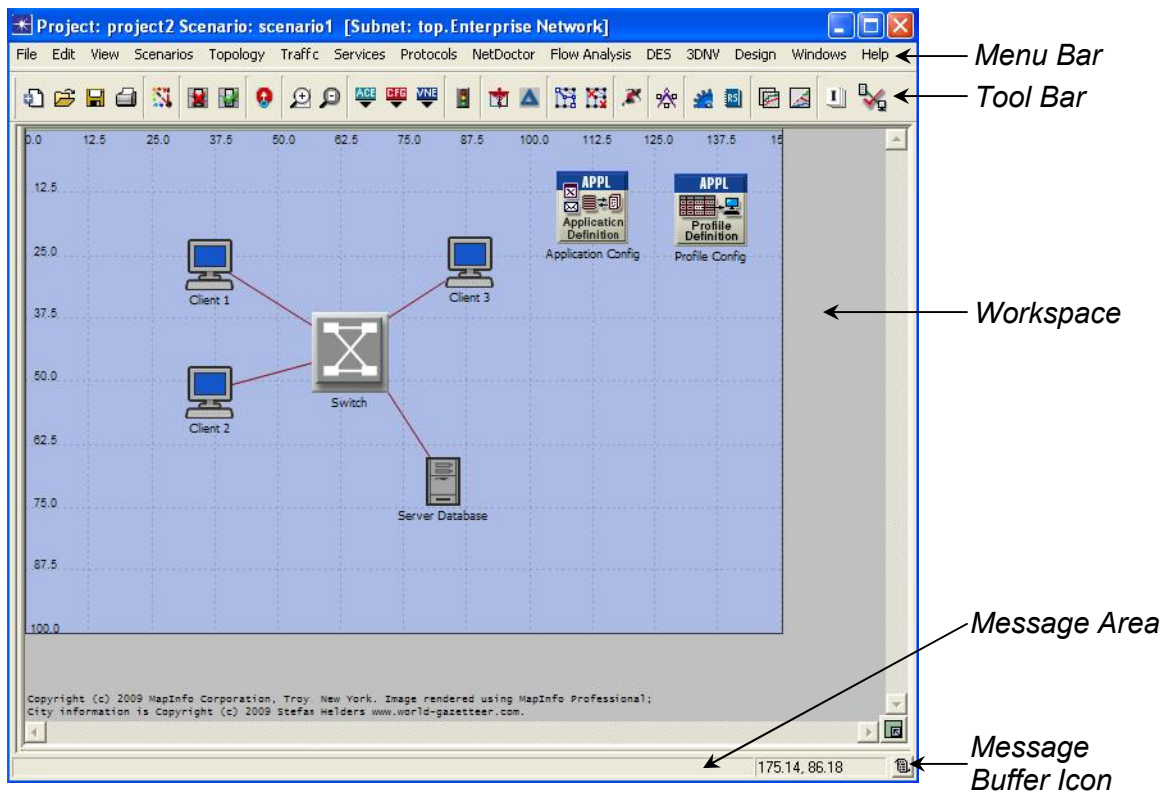


Fig. 2. Fereastra *Project Editor*

Bara de meniuri (*Menu Bar*) organizează toate operațiile editorului. Numărul și tipul operațiilor accesibile depind de modulele folosite. Operațiile editorului referitoare la context sunt accesibile prin folosirea butonului din dreapta al mouse-ului. O parte din operațiile comune, care sunt accesibile din *Menu Bar*, pot fi accesate și de pe bara de unelte (*Tool Bar* – figura 3). Lista butoanelor de pe bara de unelte este prezentată în Tabelul 1.



Fig. 3. Bara de unelte (*Tool Bar*)

Tabel 1. Semnificația butoanelor din bara de unelte

Nr. Buton	Semnificație	Nr. Buton	Semnificație
1.	New	14.	Open traffic center
2.	Open	15.	Configure/Run Doctor
3.	Save	16.	Configure/Run network differences
4.	Print	17.	Configure/Run flow analysis
5.	<i>Open object palette</i>	18.	Configure/Run survivability analysis
6.	Fail selected objects	19.	<i>Configure/Run discrete event simulation</i>
7.	Recover selected objects	20.	Configure/Run design action
8.	Go to parent subnet	21.	Configure/Run automation tasks
9.	Zoom in	22.	Web – open report server home
10.	Zoom out	23.	<i>View results</i>
11.	Import topology from ACE	24.	<i>Hide/Show graph panels</i>
12.	Import topology from device configurations	25.	Generate network inventory Summary
13.	Import topology from VNE server	26.	<i>Verify links*</i>

Notă * - În cazul în care acest buton nu este disponibil pe bară, acesta poate fi introdus astfel: Click dreapta pe bara de unelte -> *Configure Toolbar*. Se va deschide fereastra de configurare a barei de unelte, unde se selectează din categoria *Topology*, *Verify Links*. Se apasă butonul *Add>>*. În partea din dreapta-jos se setează iconița *red_x_bt*.

Notă – De reținut butoanele a căror semnificație este scrisă cu italic. Acestea vor fi folosite des pe parcursul laboratoarelor.

Workspace reprezintă spațiul de lucru situat în partea centrală a ferestrei editorului, fiind folosită pentru crearea modelului rețelei, selectarea și deplasarea obiectelor rețelei, alegerea operațiilor specifice contextului.

Zona de mesaje (*Message Area*) este plasată în partea de jos a ferestrei, având rolul de a furniza informații despre starea modelului.

Iconița bufferului de mesaje (*Message Buffer Icon*) este plasată în partea din stânga – jos a ferestrei *Project Editor*-ului. Aceasta permite accesul la o listă de mesaje, notificări și atenționări.

III) Desfășurarea lucrării

Acest laborator își propune realizarea unei rețele MANET, în cadrul careia fiecare stație mobilă folosește pentru stabilirea rutelor, de la sursă către destinație, protocolul de rutare AODV. Pentru realizarea acestei topologii, folosim biblioteca MANET disponibilă în OPNET, care este compusă din:

- **statii de lucru si servere Wireless LAN:** aceste modele de noduri sunt utilizate pentru a genera trafic cum ar fi FTP, E-mail și HTTP. Aceste noduri pot fi configurate să ruleze orice protocol de rutare MANET.
- **Statii MANET:** aceste noduri sunt folosite pentru a genera pachete peste IP peste wireless LAN. Ele funcționează ca surse de trafic sau destinații și pot fi configurate să ruleze orice protocol de rutare MANET disponibil. Pentru acest tip de nod, Layer2 este wireless LAN.

Toate nodurile MANET sunt incluse în paleta de obiecte MANET ilustrată în Figura 3.

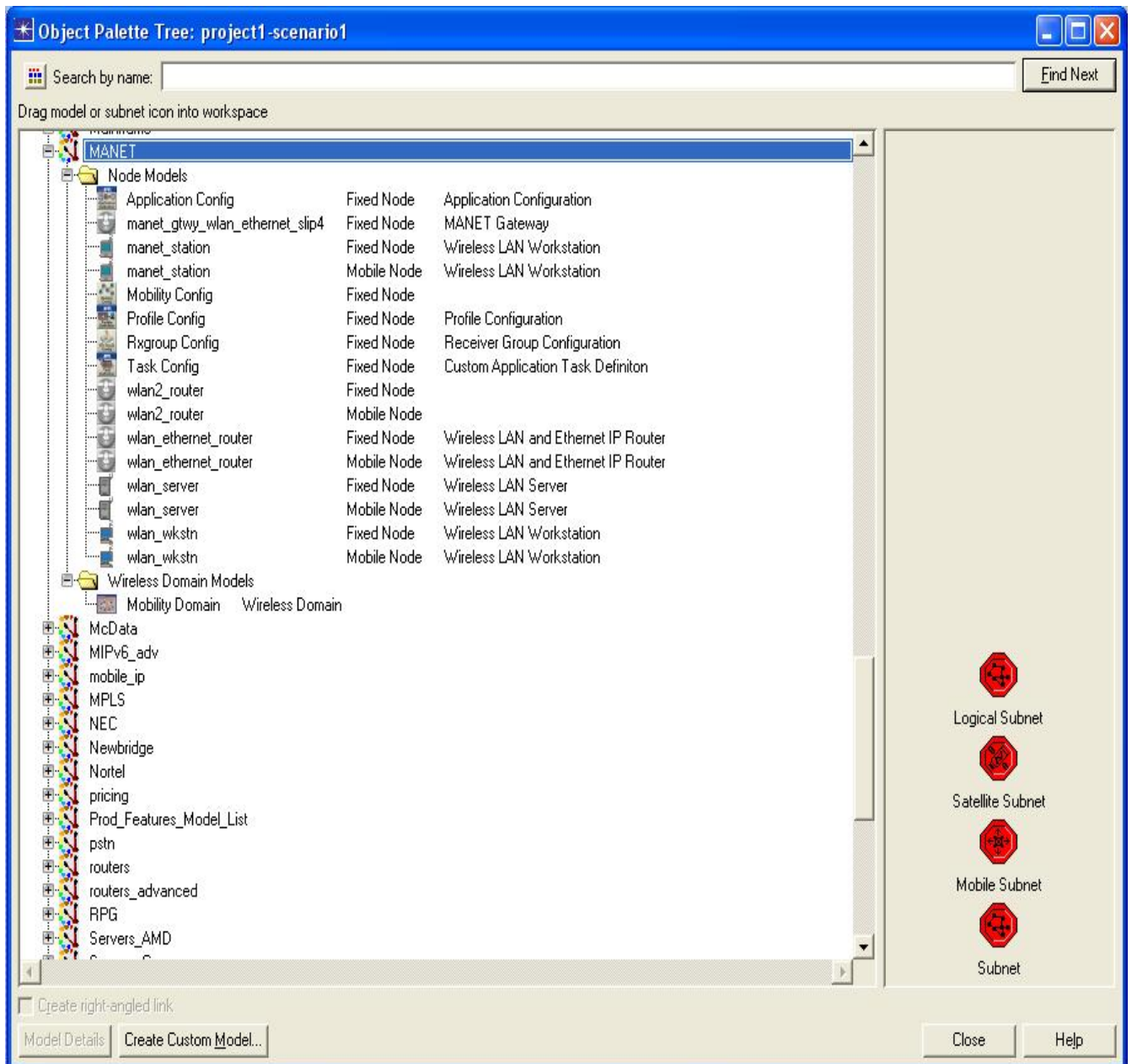


Figura 3. Paleta de obiecte MANET

Construirea scenariului *Retea_AODV*

Pentru crearea unui model nou de rețea este necesar să se creeze un nou proiect și un scenariu:

Se creează în folder-ul studentului un nou director de forma „\OPNET\GrupaXYZ\LabMM” (acesta va fi directorul în care se va salva proiectul **MANET** corespunzător laboratorului).

Se lansează în execuție programul *OPNET Modeler*.

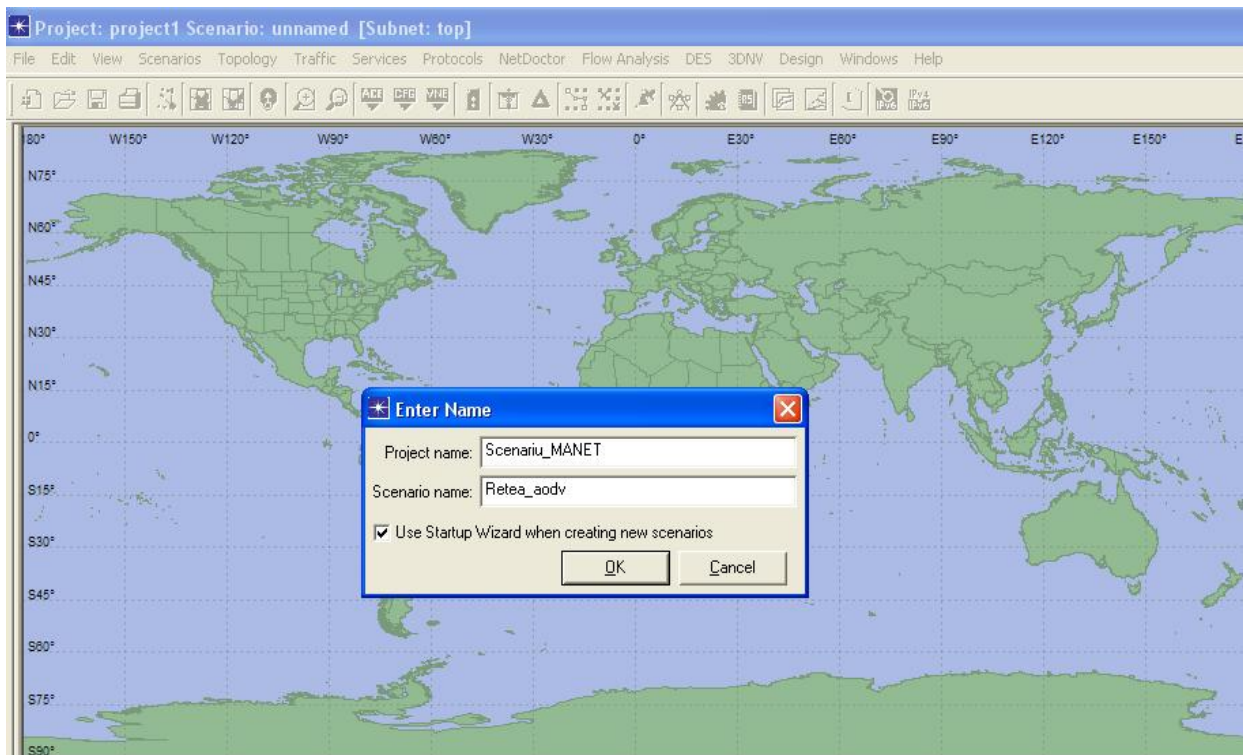


Se setează directorul în care vor fi plasate fișierele proiectului (directorul creat mai sus): *File -> Manage Model Files -> Add Model Directory*. Se bifează cele două căsuțe: *Include all subdirectories* și *Make this the default directory*.

Se selectează: *File -> New... -> Project -> OK*.

Se definesc: *Project Name: Scenariu_MANET*

Scenario Name: Retea_AODV

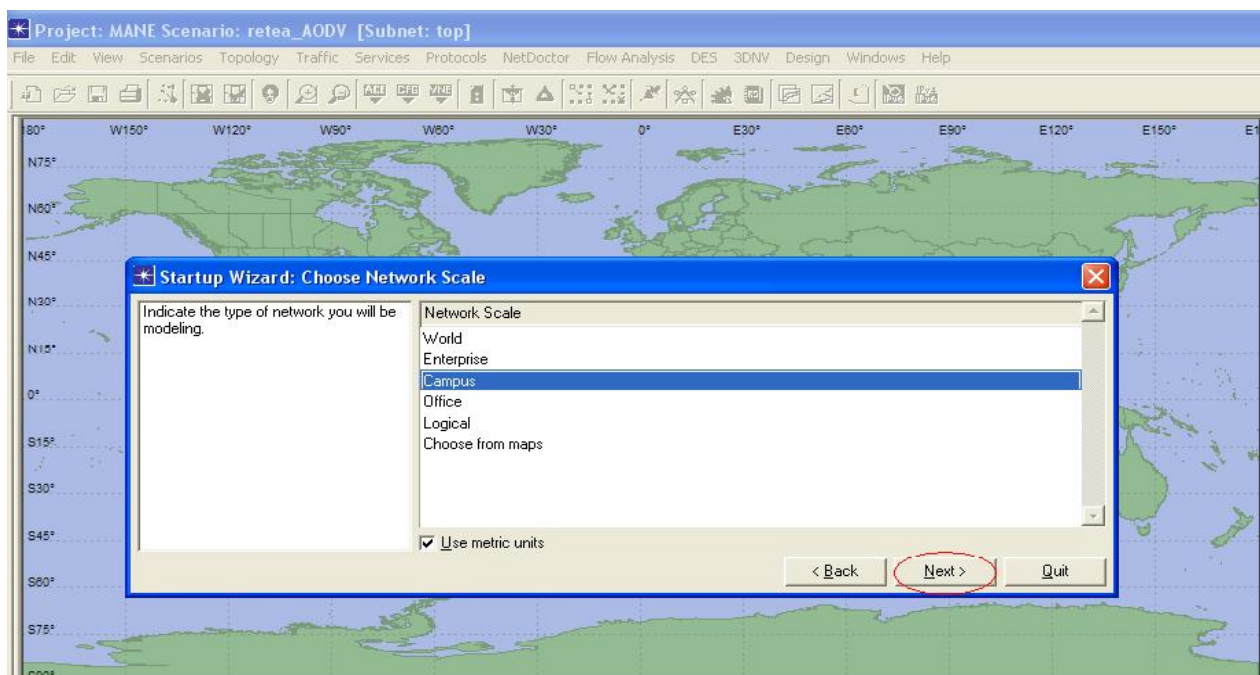
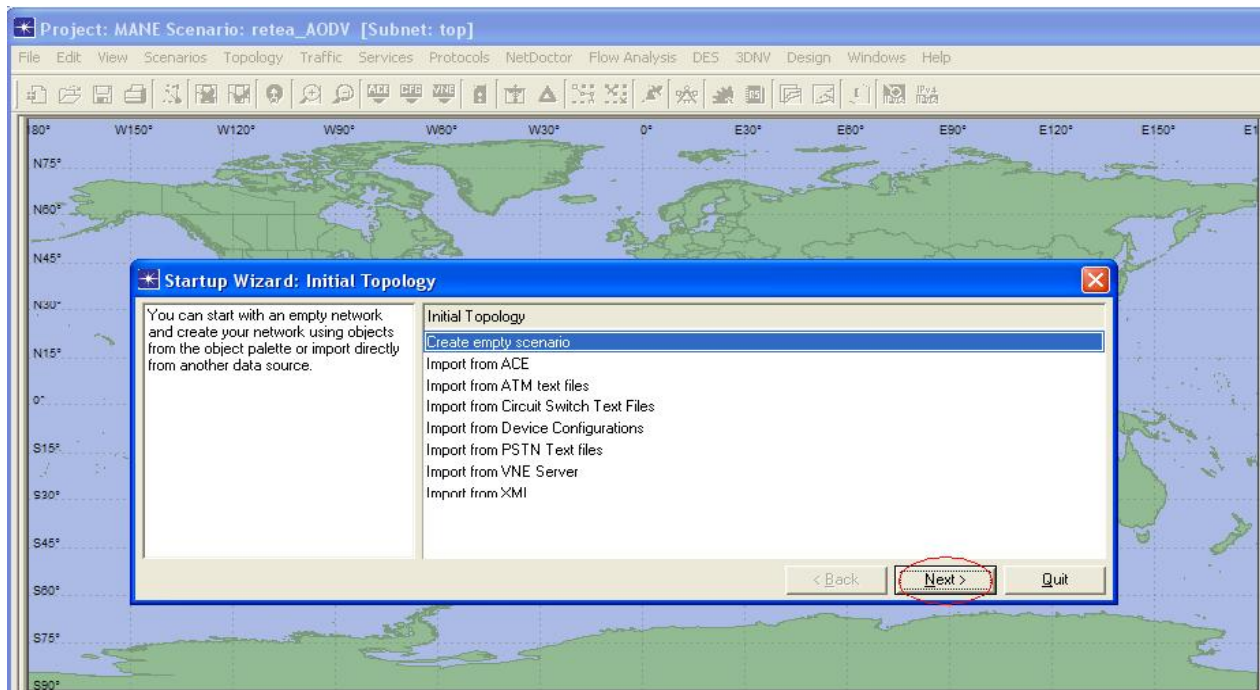


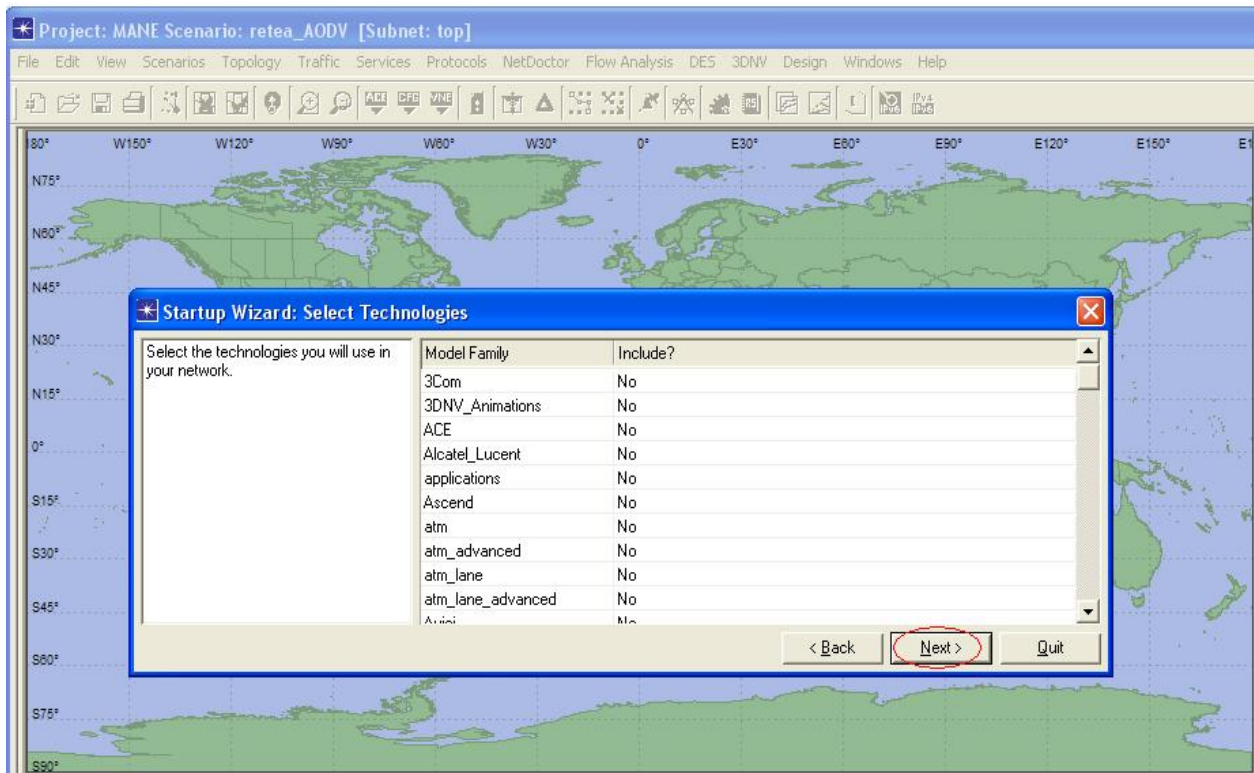
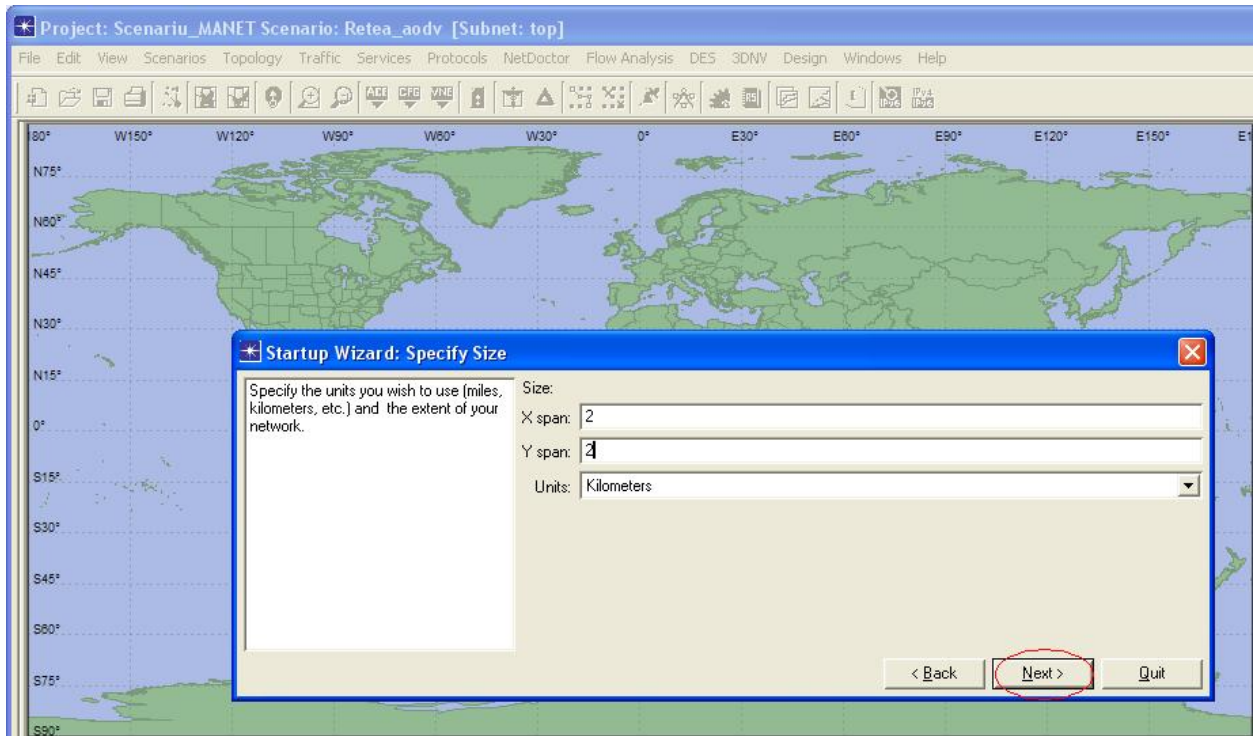
Se setează apoi: *Initial Topology: Create empty scenario*

Network Scale: Campus

Size: X: 2 / Y: 2 / Units: Kilometers

Model Family: nu se va selecta nimic, direct Next (paleta de obiecte se va construi ulterior). Urmati pasii din printscreen-urile de mai jos:





În urma parcurgerii acestor pași se va deschide fereastra *Project Editor* împreună cu paleta de obiecte disponibile (*Object Palette Tree*).

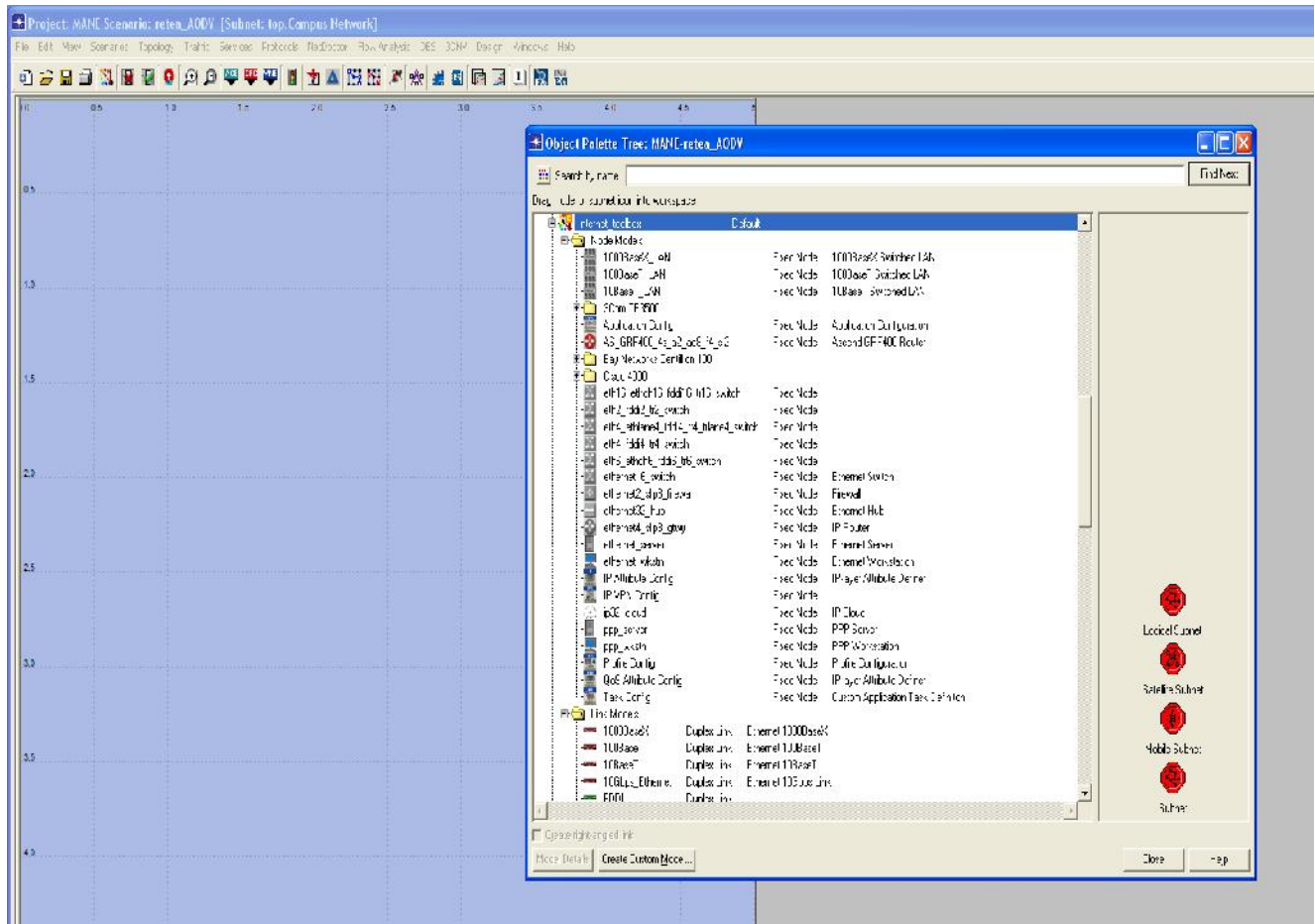


Fig 4. Fereastra de lucru și paleta de obiecte

Am folosit protocolul de rutare AODV cu parametri implicați, modelul de mobilitate *Random Waypoint*, și aplicațiile următoare: voce, conferință video și email. Au fost testate numai topologii mobile, deoarece am plecat de la ipoteza că într-o rețea ad-hoc mobilitatea nodurilor este ridicată și, în concluzie, deplasarea nodurilor nu este nulă.

Project: Scenario_MANET Scenario: Retea_oddy [Subnet: top_Campus_Network]

File Edit View Scenario Tools Traffic Services Networks Networker File Analysis DNS DHCP Config Scenario Help

Object Palette Tree: Scenario_MANET-Retea_oddy

Drag node(s) or subnet(s) into workspace

- Internet
- Link
- Link_advanced
- Link_PP
- Link_Subnet
- Trunk
- Mainframe
- Node
- Node Mode
 - Application Config: Fixed Node, Application Configuration
 - enet_qlan: Fixed Node, MANET Gateway
 - enet_ethernet_ip4: Fixed Node, Wireless LAN Workstation
 - enet_qlan: Fixed Node, Wireless LAN Workstation
 - Mobile Config: Fixed Node, Mobile Configuration
 - Task Config: Fixed Node, Device Group Configuration
 - Task Config: Fixed Node, Custom Application Task Definition
 - Wireless Config: Fixed Node, Wireless LAN and External IP Router
 - Wireless Config: Fixed Node, Wireless LAN and External IP Router
 - Wireless Config: Fixed Node, Wireless LAN Server
 - Wireless Config: Fixed Node, Wireless LAN Server
 - Wireless Config: Fixed Node, Wireless LAN Workstation
 - Wireless Config: Fixed Node, Wireless LAN Workstation
 - Wireless Config: Fixed Node, Wireless LAN Workstation
 - Wireless Domain: Wireless Domain

Mobile Detail: Create Custom Mode...

Close Help

Project: Scenario_MANET Scenario: Retea_oddy [Subnet: top_Campus_Network]

File Edit View Scenario Tools Traffic Services Networks Networker File Analysis DNS DHCP Config Scenario Help

Object Palette Tree: Scenario_MANET-Retea_oddy

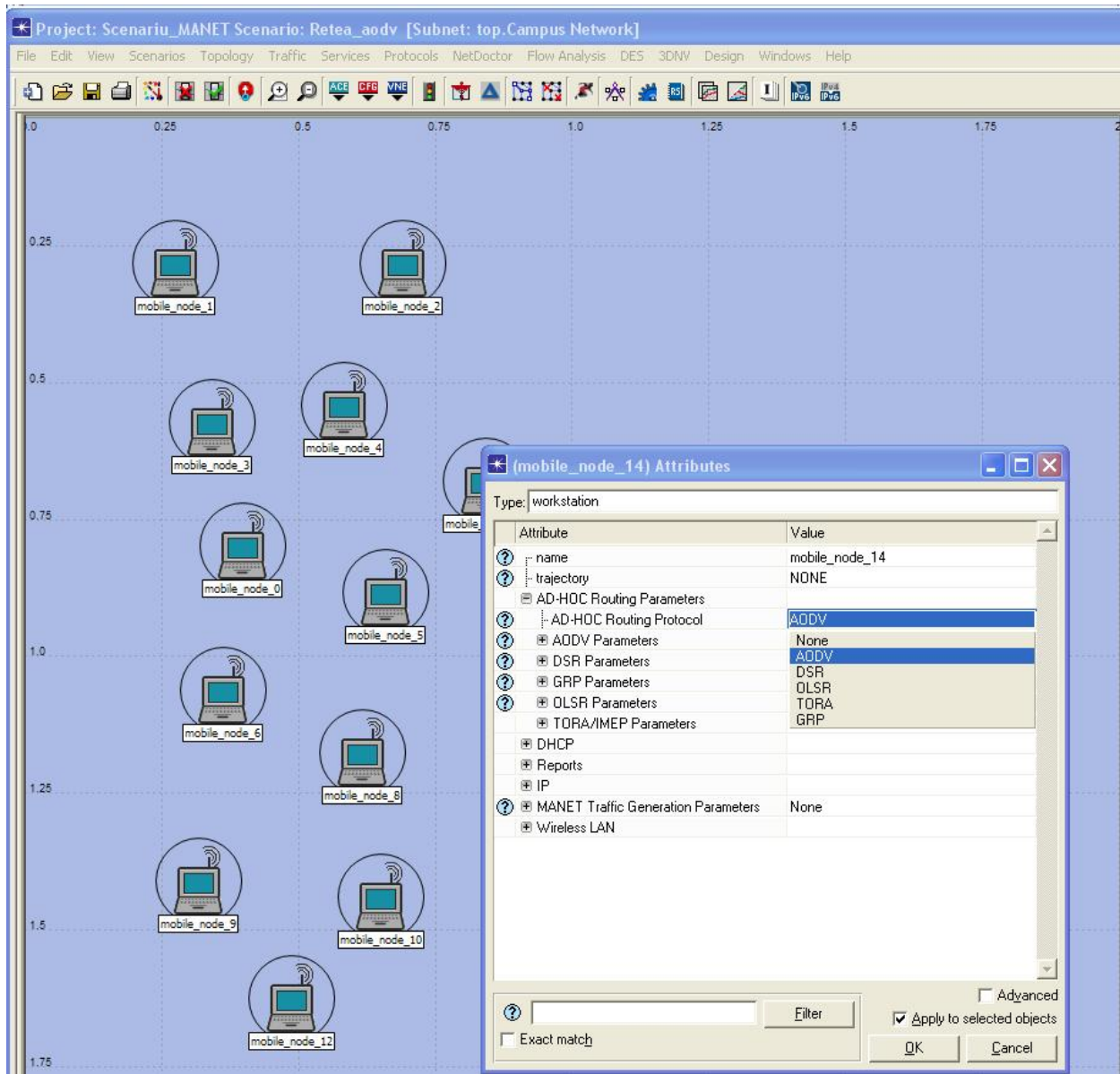
Drag node(s) or subnet(s) into workspace

- Internet
- Link
- Link_advanced
- Link_PP
- Link_Subnet
- Trunk
- Mainframe
- Node
- Node Mode
 - Application Config: Fixed Node, Application Configuration
 - enet_qlan: Fixed Node, MANET Gateway
 - enet_ethernet_ip4: Fixed Node, Wireless LAN Workstation
 - enet_qlan: Fixed Node, Wireless LAN Workstation
 - Mobile Config: Fixed Node, Mobile Configuration
 - Task Config: Fixed Node, Device Group Configuration
 - Task Config: Fixed Node, Custom Application Task Definition
 - Wireless Config: Fixed Node, Wireless LAN and External IP Router
 - Wireless Config: Fixed Node, Wireless LAN and External IP Router
 - Wireless Config: Fixed Node, Wireless LAN Server
 - Wireless Config: Fixed Node, Wireless LAN Server
 - Wireless Config: Fixed Node, Wireless LAN Workstation
 - Wireless Config: Fixed Node, Wireless LAN Workstation
 - Wireless Domain: Wireless Domain

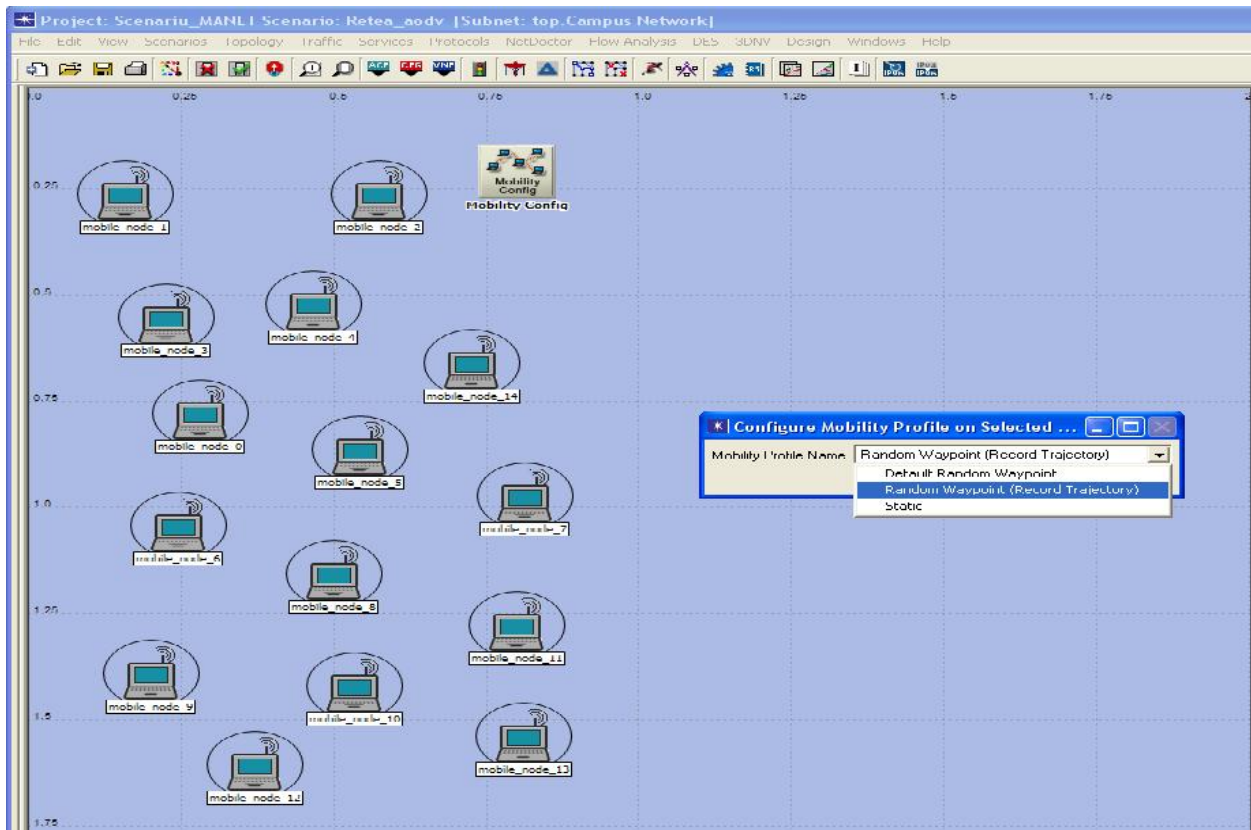
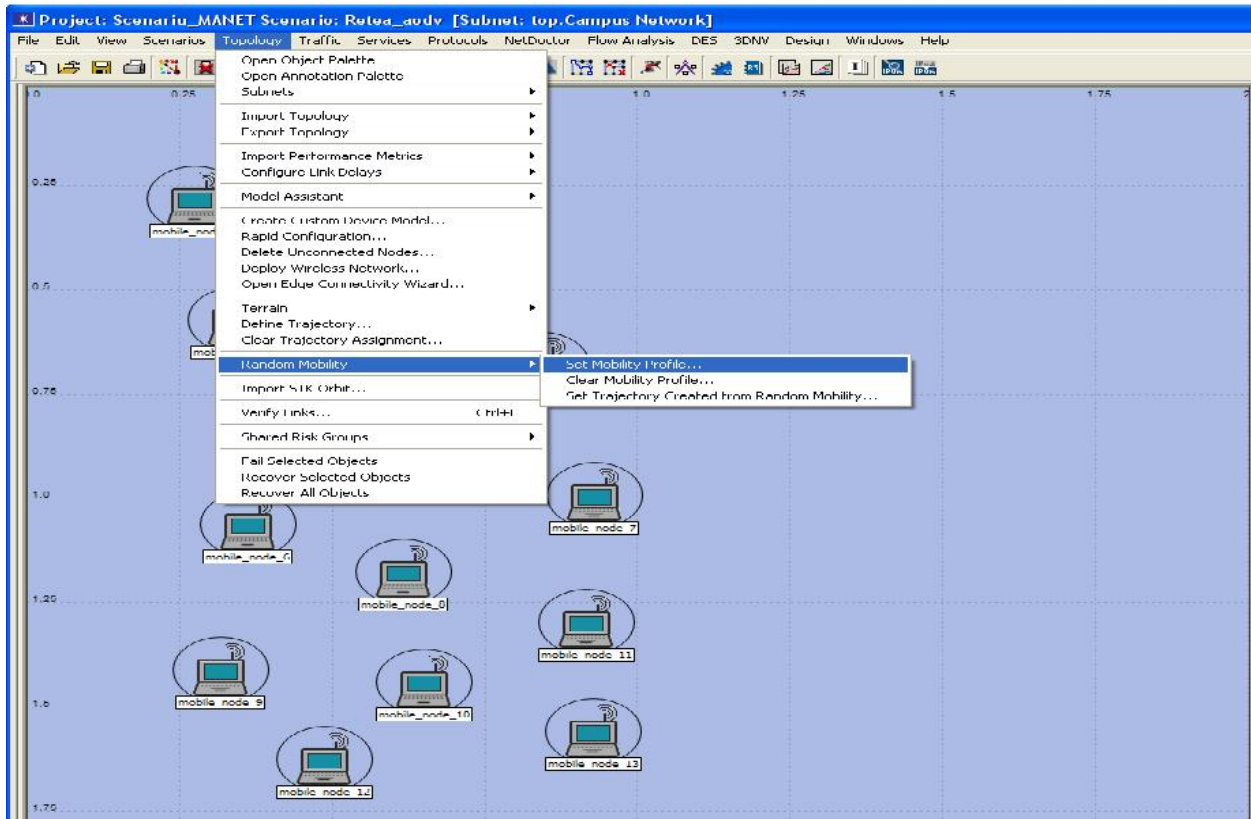
Mobile Detail: Create Custom Mode...

Close Help

Pe fiecare statie mobila se configureaza protocolul de rutare AODV, urmarindu-se pasii de mai jos. Apasati CTRL+A pentru a selecta toate host-urile. Se seteaza folosirea protocolului AODV si apoi se bifeaza “Apply to selected objects”, pentru a se face aceeasi setare pentru toate cele 15 statii mobile.



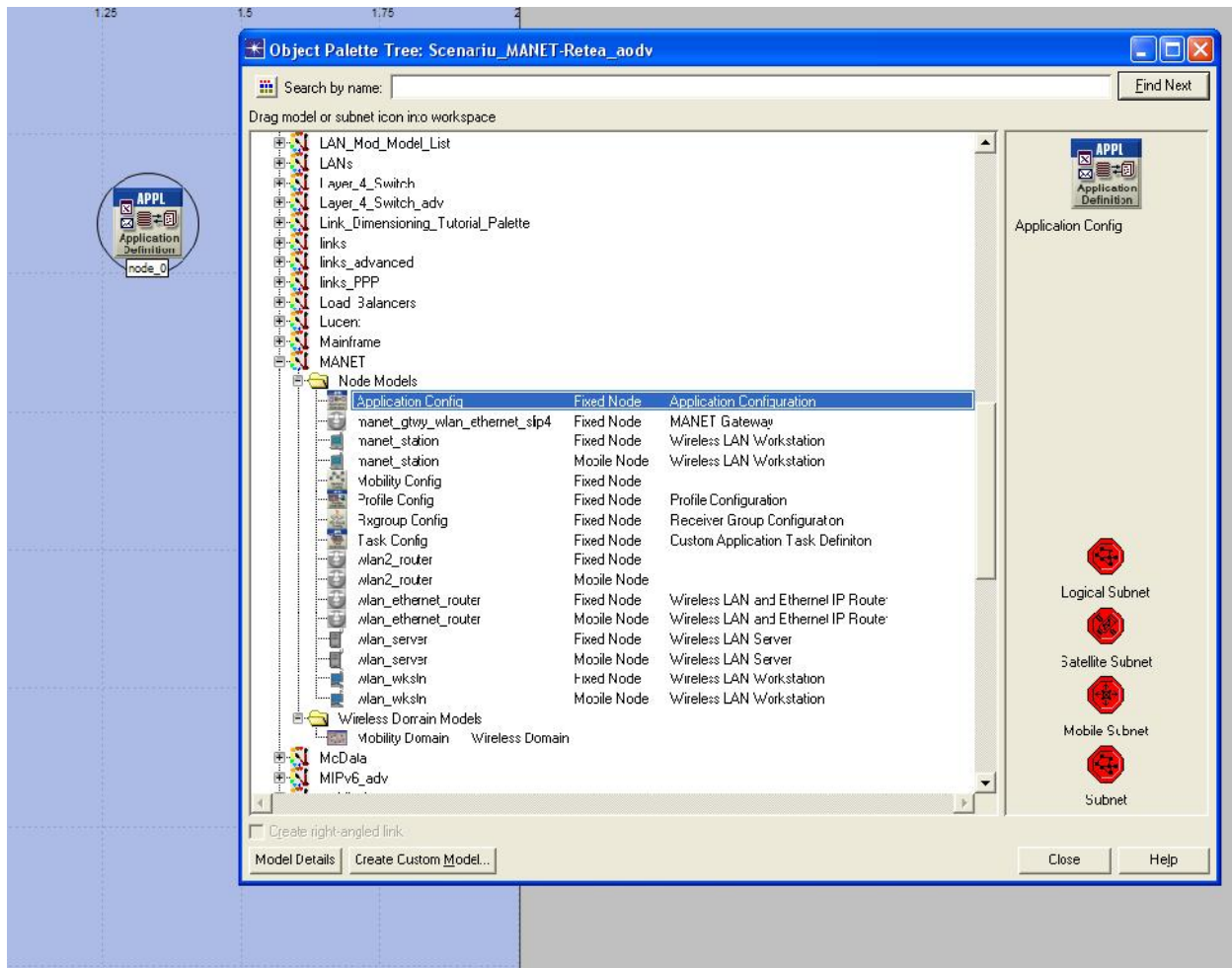
Configurarea mobilității nodurilor se poate face fie manual, fie automat, folosind mecanismele integrate. Configurarea manuală se face selectand tipul de mobilitate din meniul Topology (Topologie) -> Random Mobility (Mobilitate arbitrară) -> Set Mobility Profile (Setarea Profilului de Mobilitate). Pentru proiectul din această lucrare s-a selectat modelul de mobilitate implicită *Random Waypoint*.



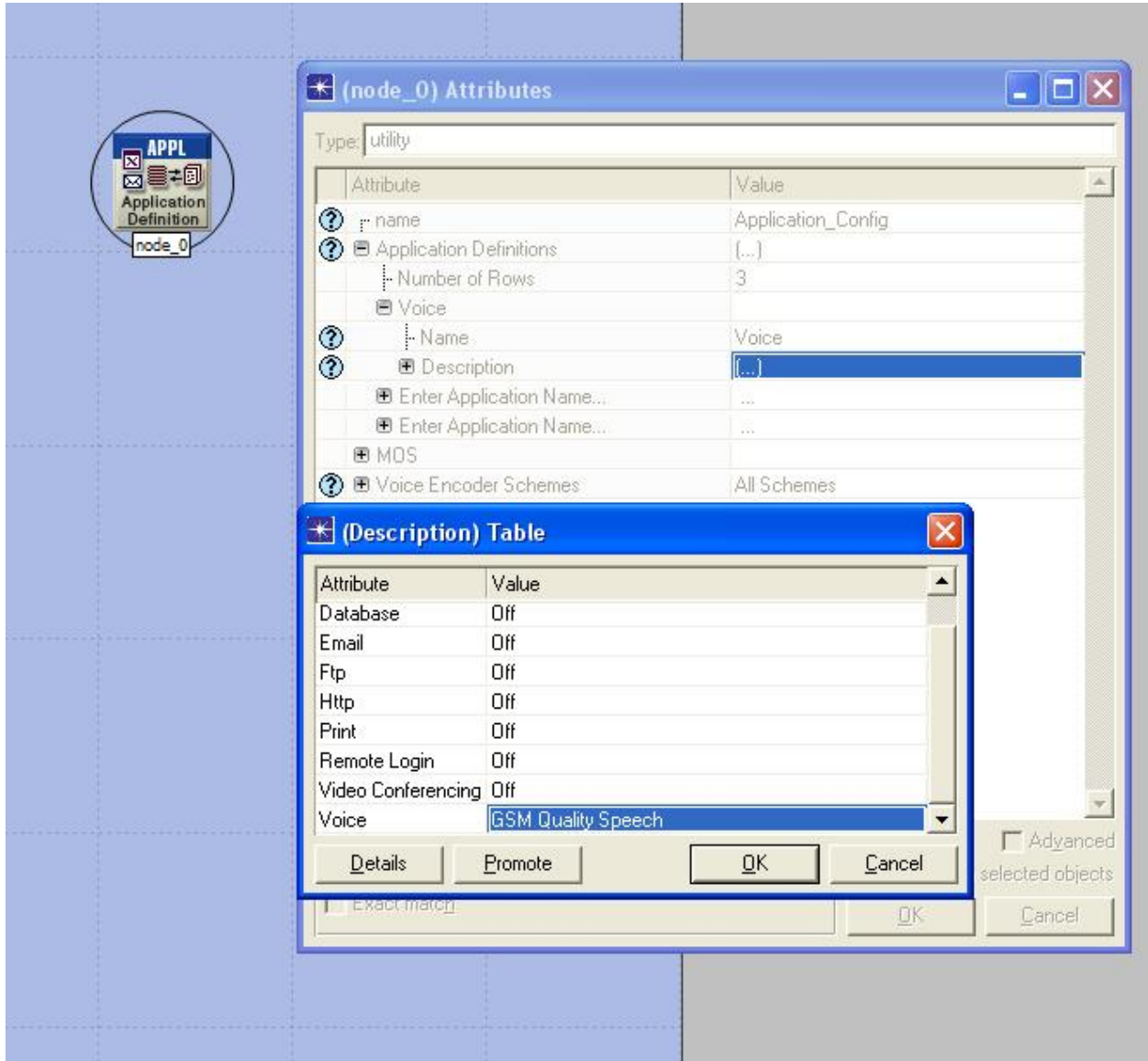
Adăugarea și configurarea aplicațiilor

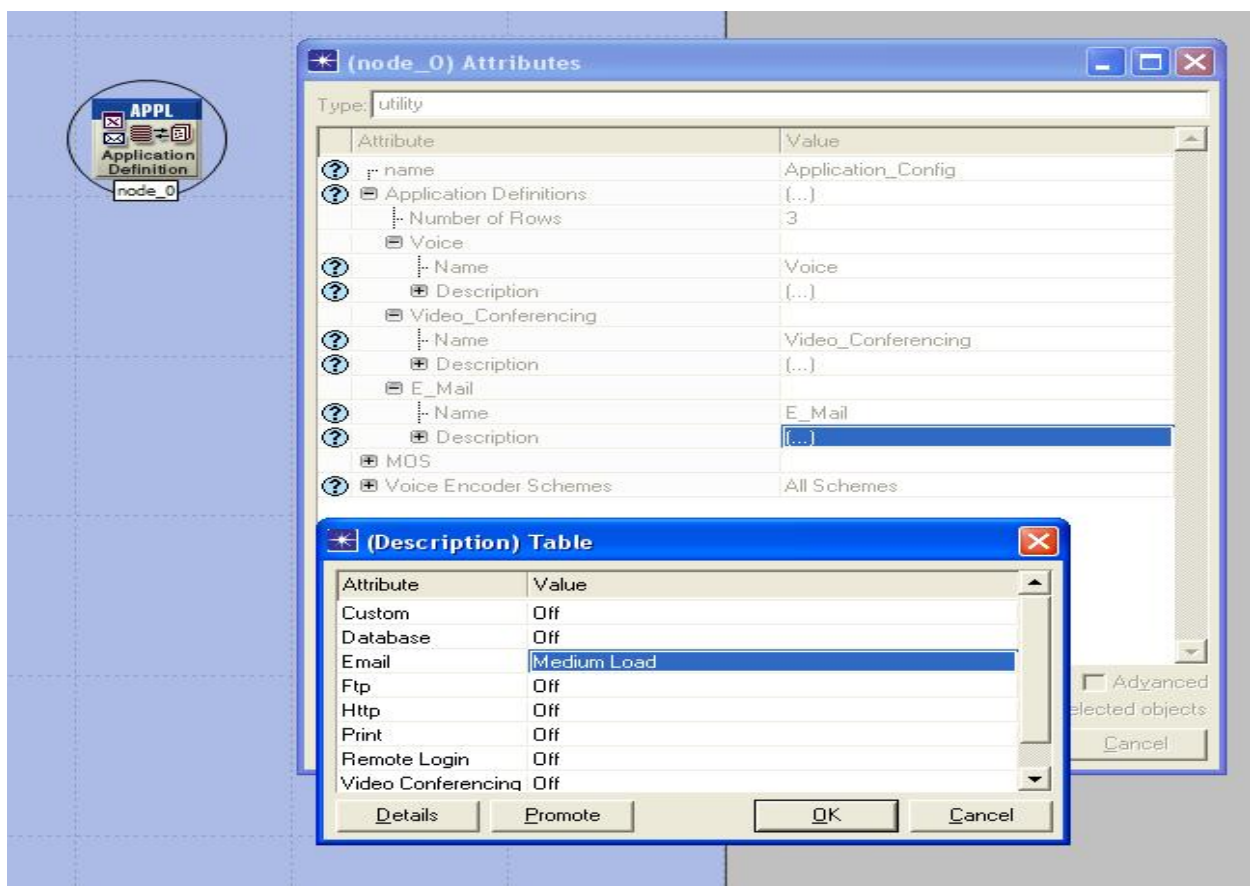
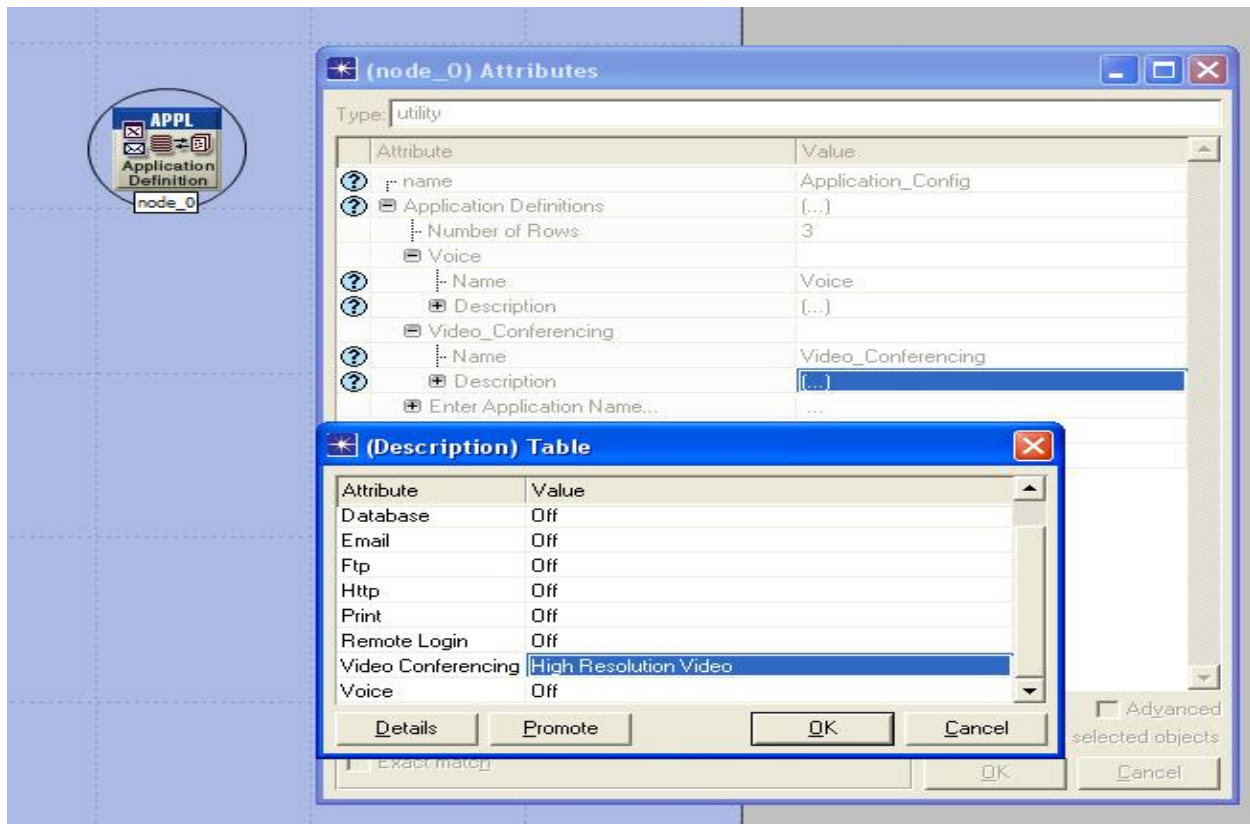
Se pot configura aplicații standard recum FTP sau Email care să genereze trafic simulat in modelul de rețea ales. Acest lucru se realizează cu ajutorul nodului Application Configuration (Configurarea Aplicațiilor). Pentru rețeaua simulată in această lucrare am ales generarea de trafic de voce, video-conferință și email, fiecare aplicație fiind definită in cadrul nodului Application Config, ca in figura de mai jos.

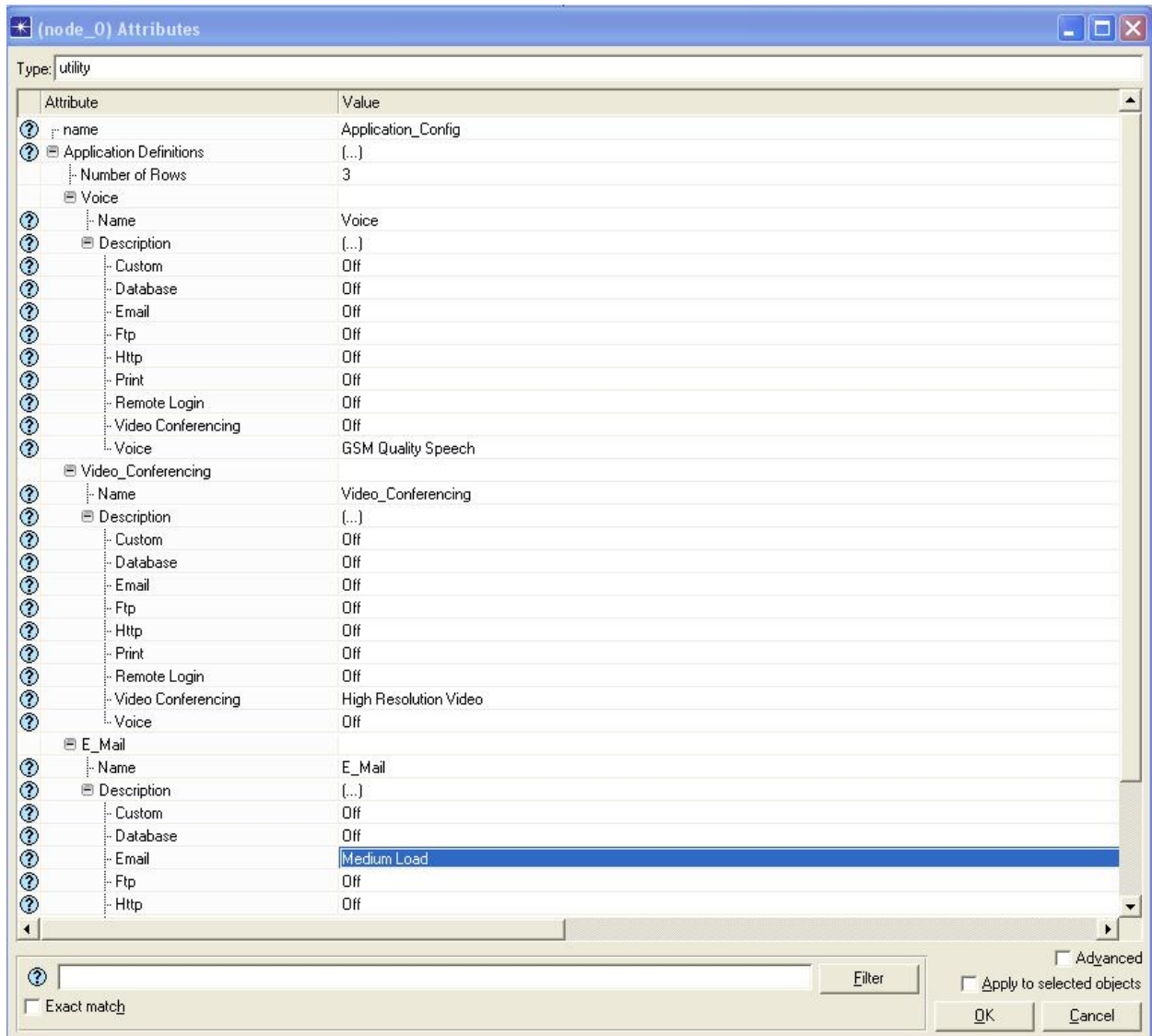
Trebuie deasemenea configurate si profilurile de aplicație, prin introducerea unui nou nod Profile Configuration.



In meniurile din figurile de mai jos sunt prezentate setarile necesare pentru configurarea celor 3 aplicatii, voce (de calitate GSM), video de rezolutie inalta, si e-mail de dimensiune medie.

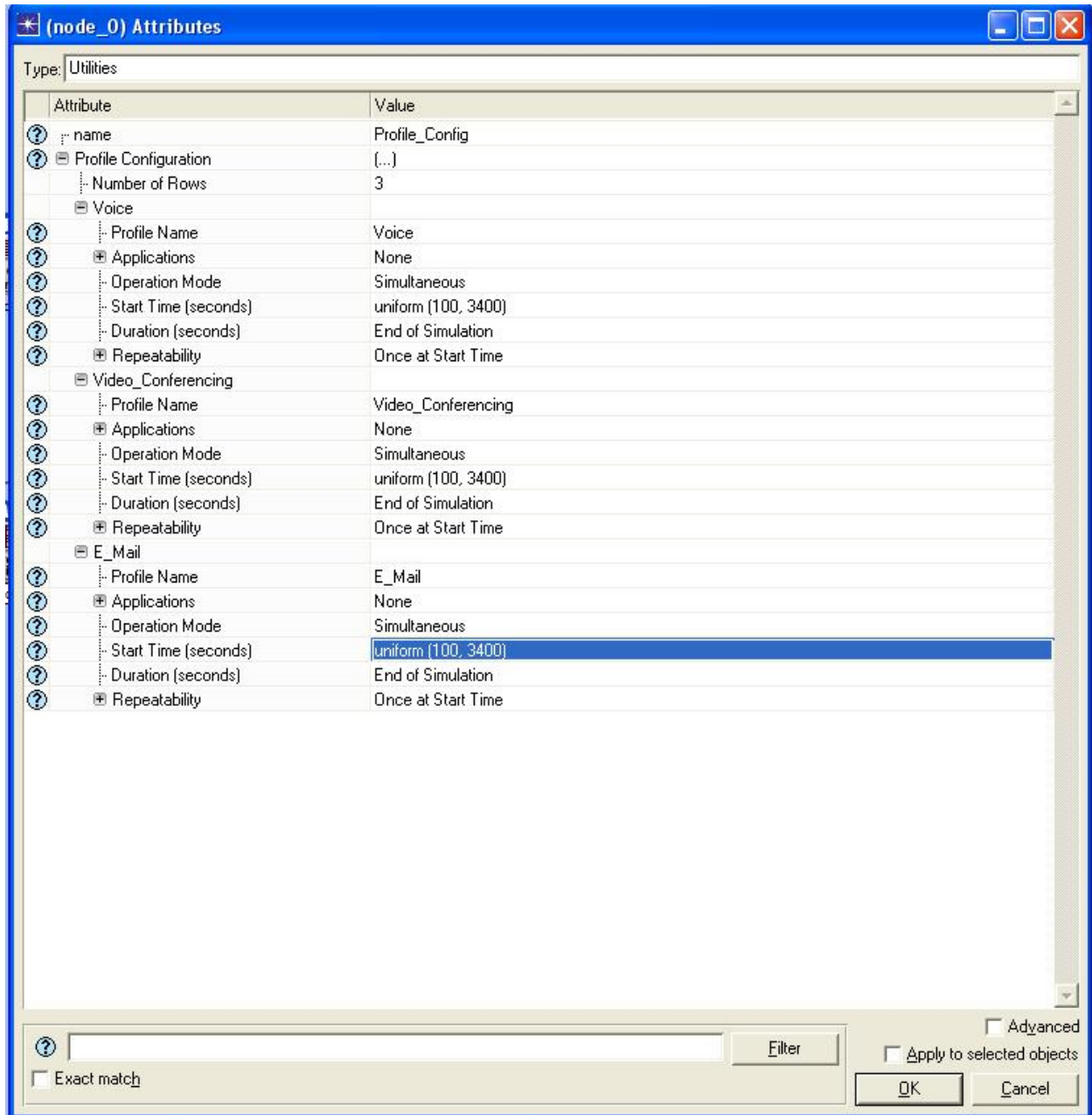




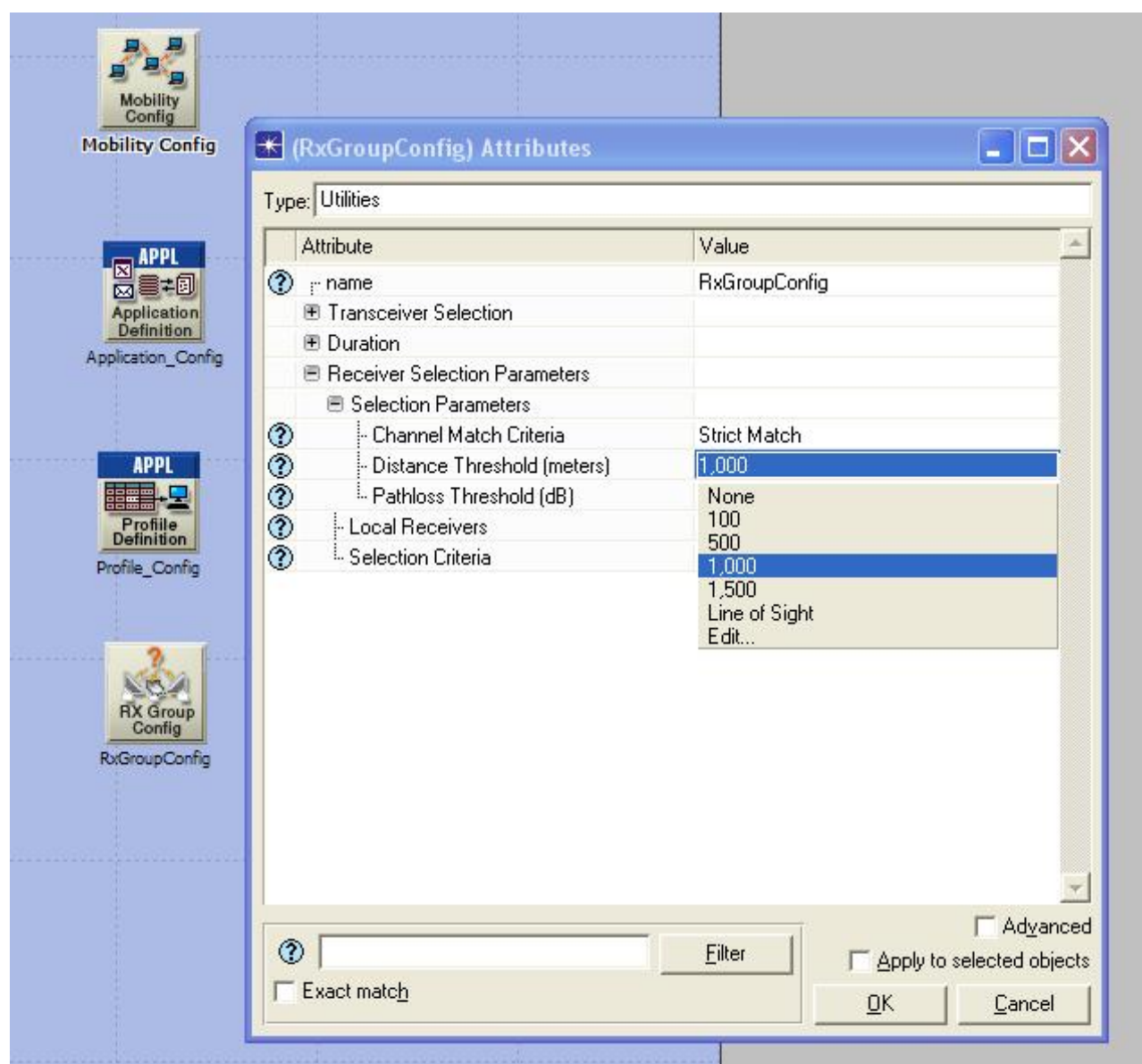


Configurarea finala a profilului aplicatie, unde se vad toate cele 3 tipuri de aplicatii setate care ruleaza in timpul simularii.

In mod analog, se configureaza Profile Config.



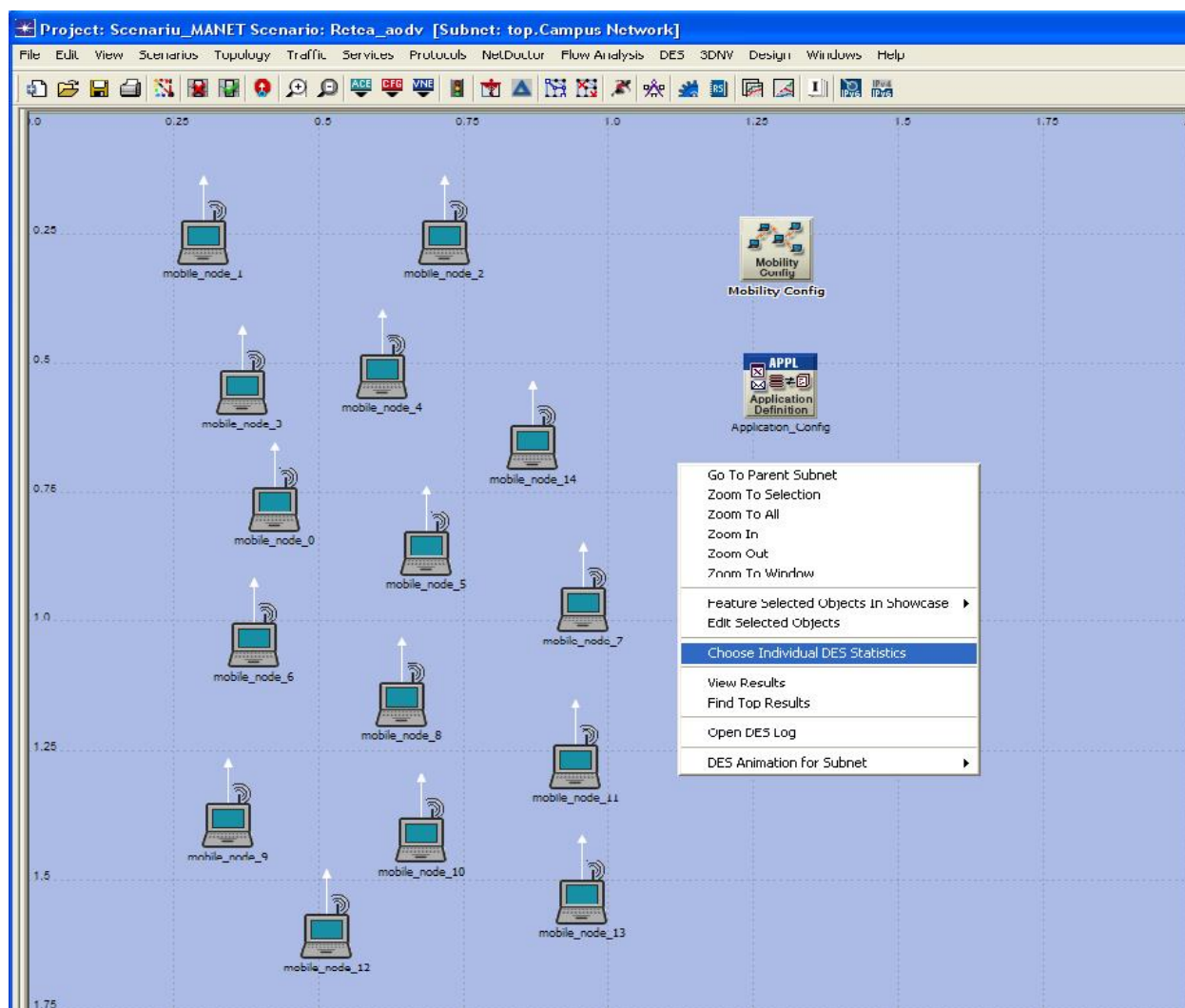
In rețeaua simulată mai este adăugat și un nod denumit Rx Group Config, cu scopul de a diminua timpul necesar simulării. Este cofigurat să elimine toate stațiile receptoare care se află la o distanță mai mare de 1000 de sursă.

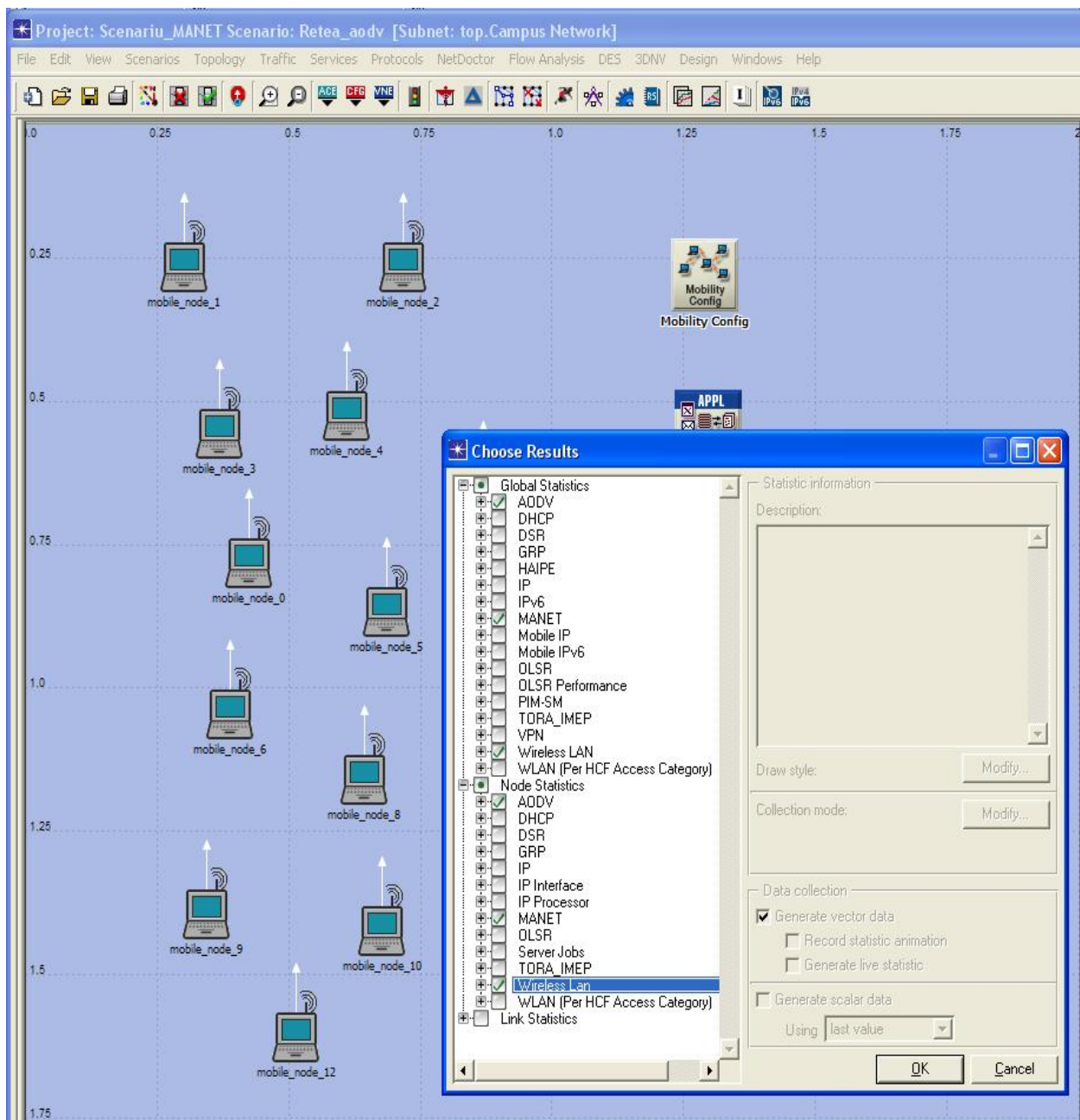


Pentru a analiza performanțele protocoalelor simulate, pot fi culese diferite statistici pentru fiecare nod din rețea, sau global, pentru întreaga rețea.

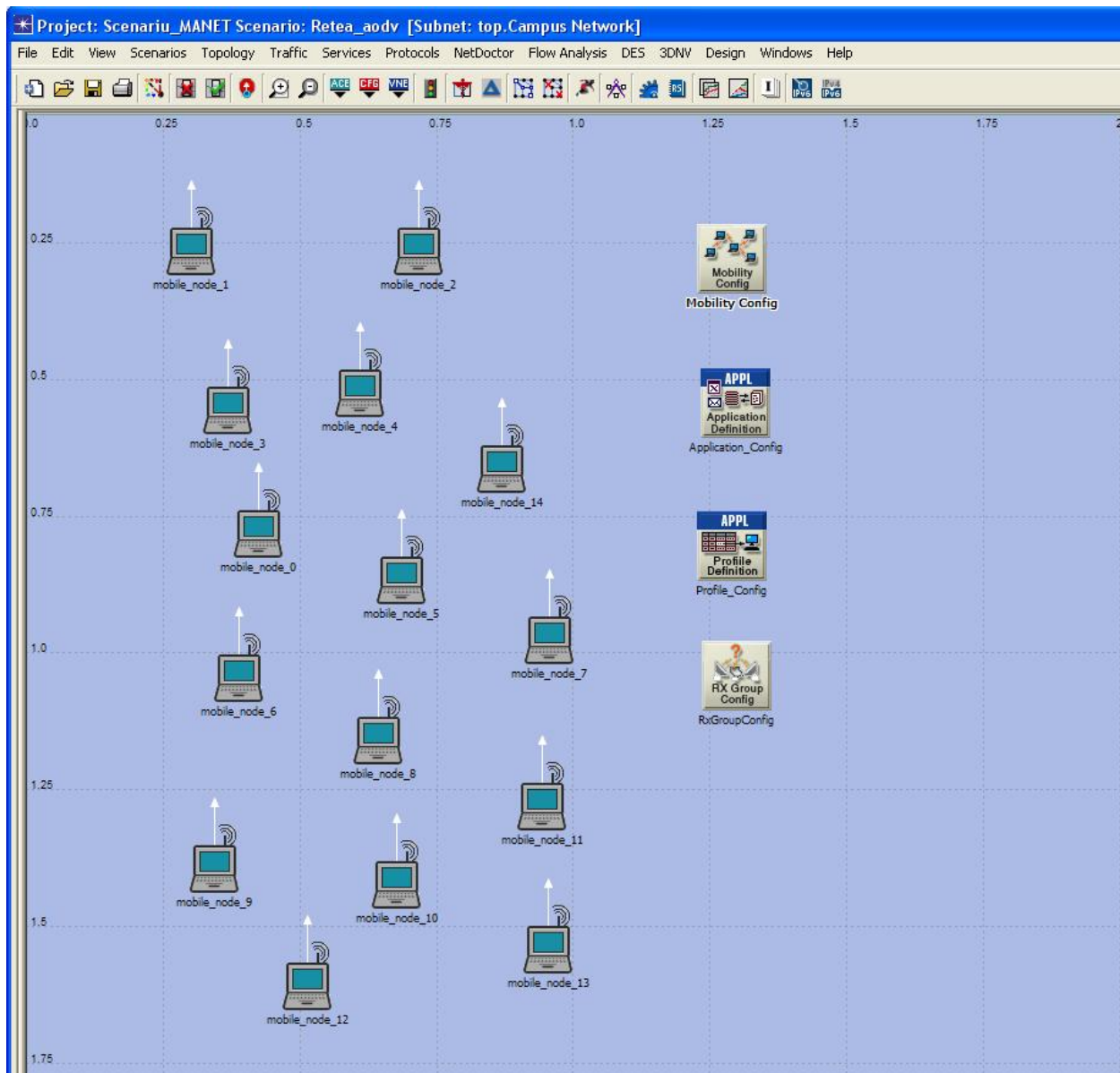
Selectarea statisticilor dorite se face din meniul DES.

După ce s-a făcut selectarea parametrilor care se doresc a fi monitorizați, se poate începe simularea propriu-zisă a funcționării rețelei. Timpul de simulare ales pentru exemplul din această lucrare a fost de 1h. După încheierea timpului de simulare, rezultatele selectate anterior pot fi vizualizate și interpretate.



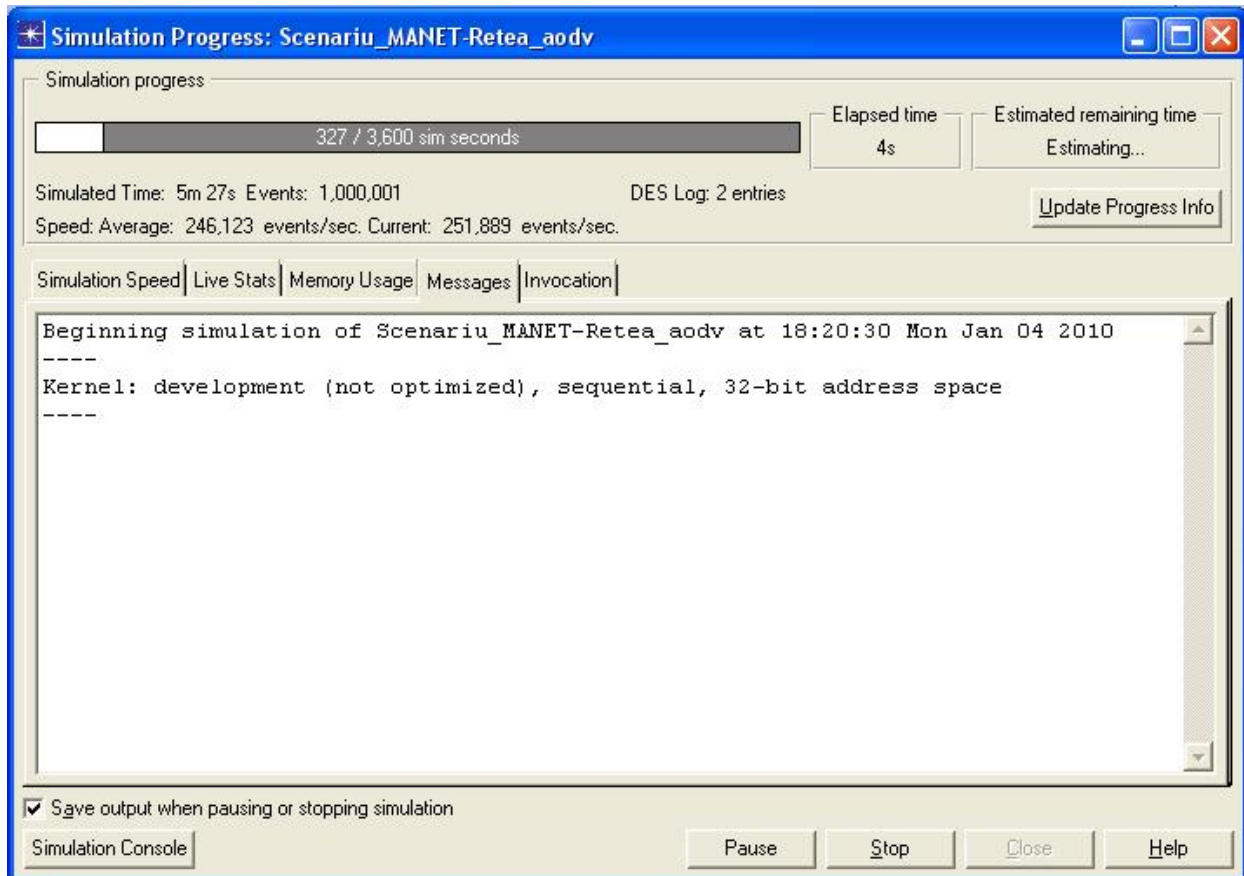
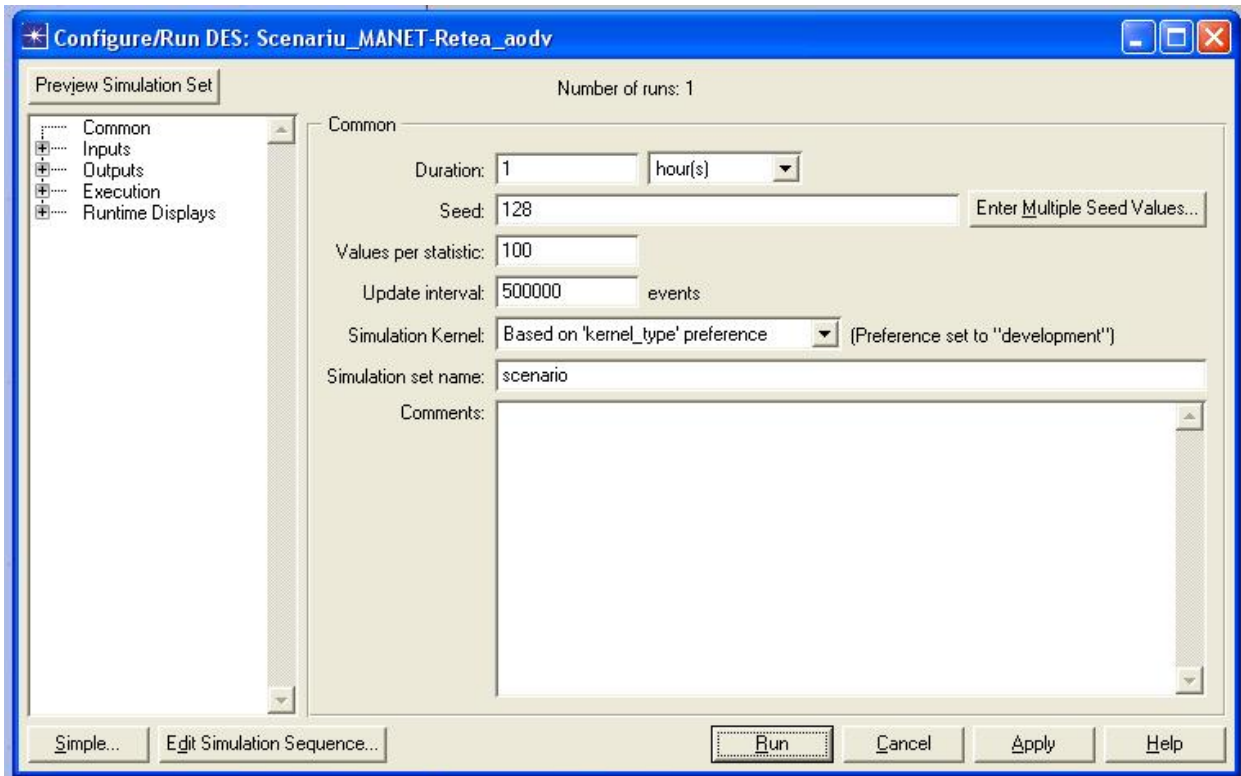


In figura de mai sus am setat nivelurile la care dorim sa vizualizam rezultate, respectiv protocolul AODV, retea MANET si WIRELESS LAN.



Acesta este scenariul final, compus din 15 statii mobile, un profil de mobilitate, un profil de aplicatie precum si profil de configurare.

Figurile de mai jos arata timpul de simulare setat si modul in care ruleaza scenariul.



IV) Interpretarea rezultatelor si concluzii

Figura de mai jos arată cantitatea de informație abandonată într-o rețea cu 20 de noduri ca urmare a folosirii numărului maxim de reîncercări pentru livrarea pachetului în cauză.

Protocolul AODV reține o singură rută pentru fiecare destinație, iar încercările de retransmitere a pachetelor sunt numeroase în cazul AODV. Totuși, protocolul nu prezintă rate de pierdere a informației utile foarte mari, deoarece rețeaua simulată conține un număr redus de noduri mobile.

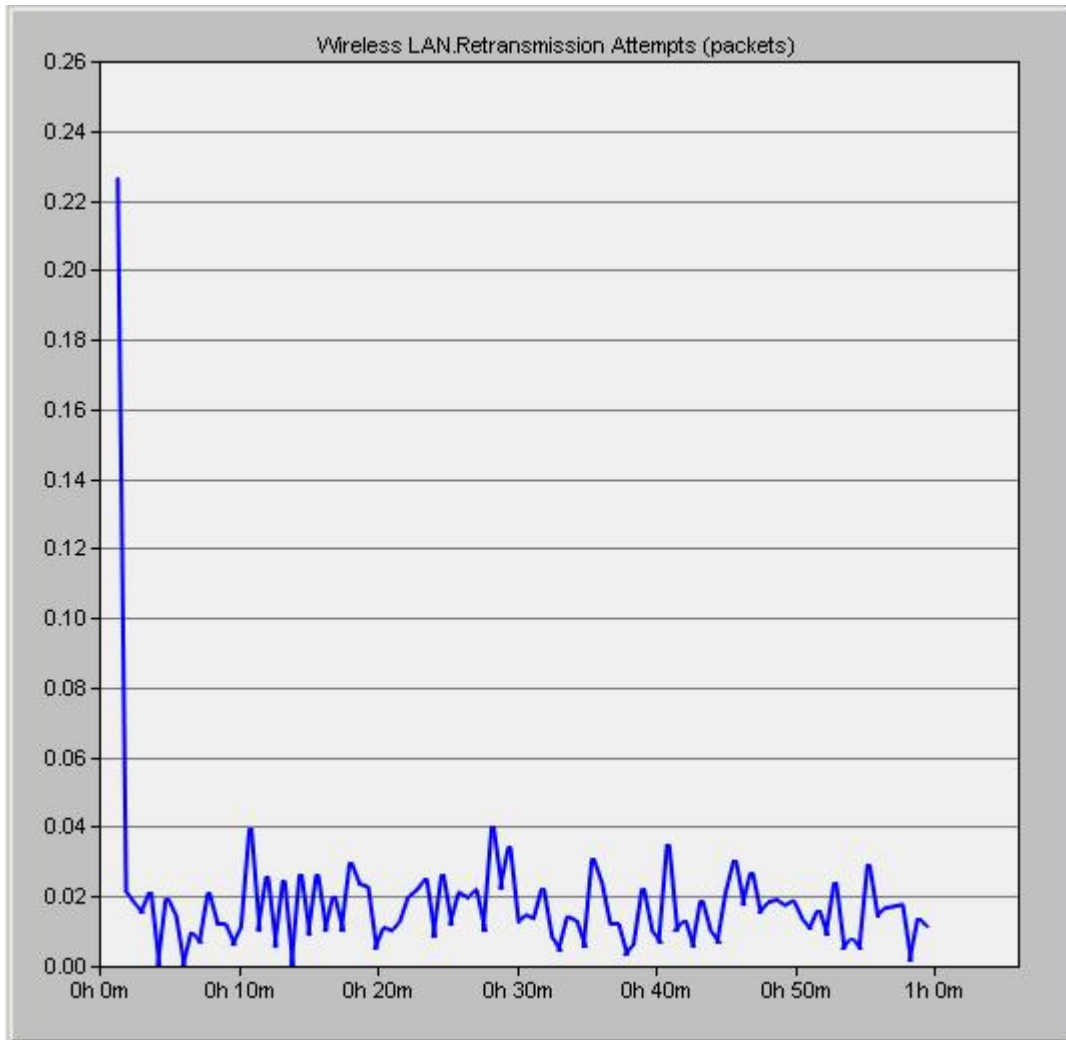
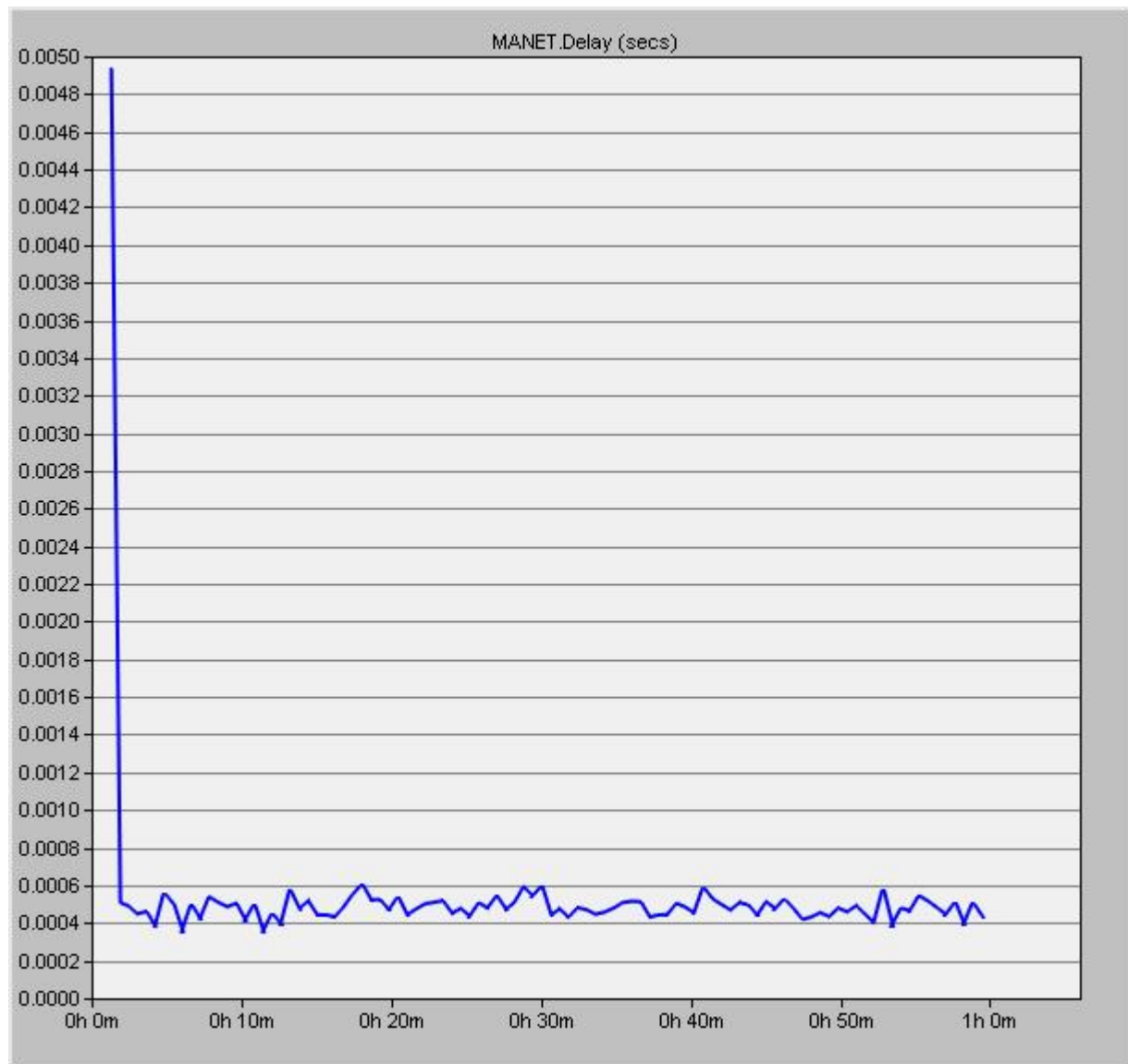
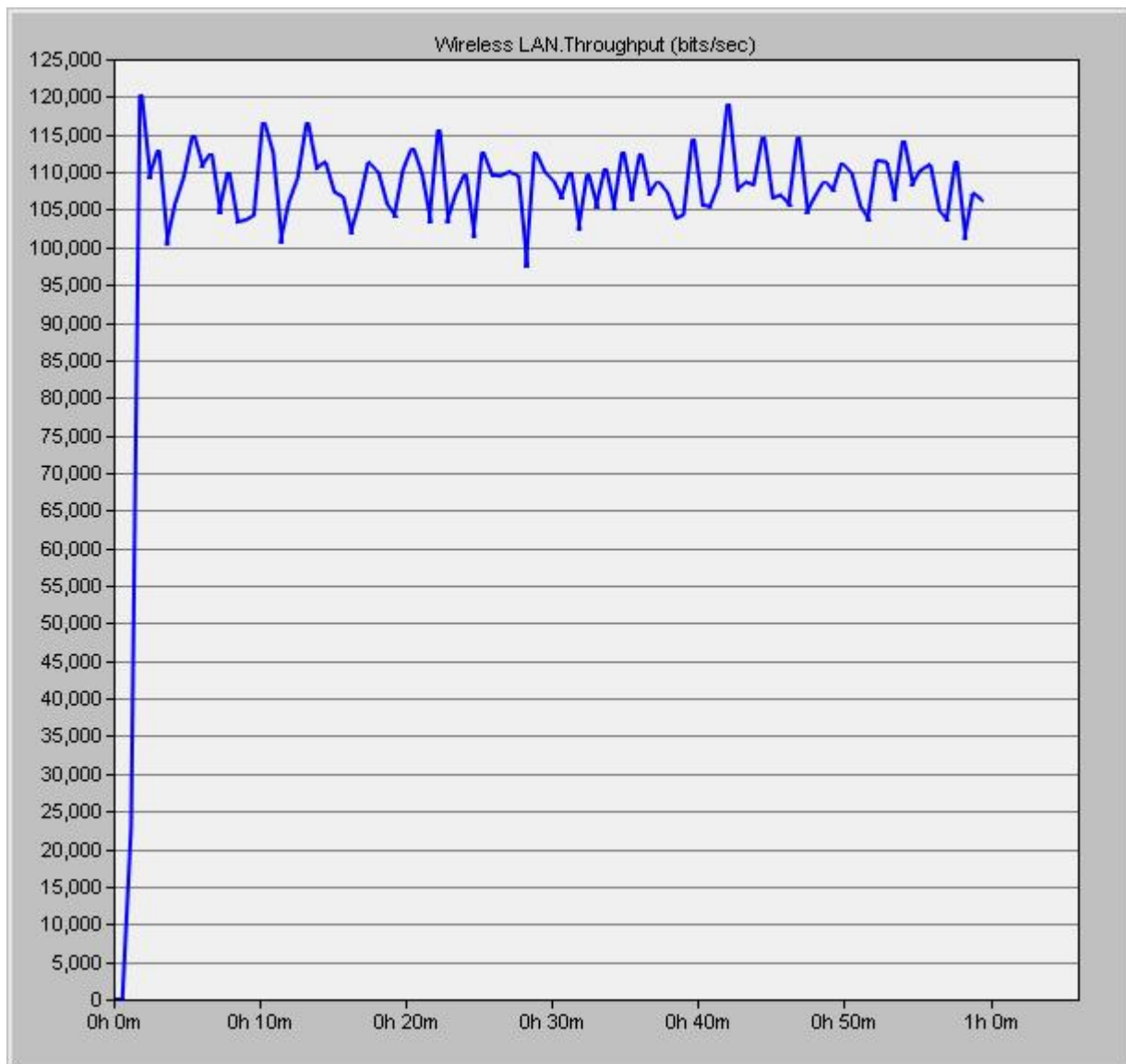


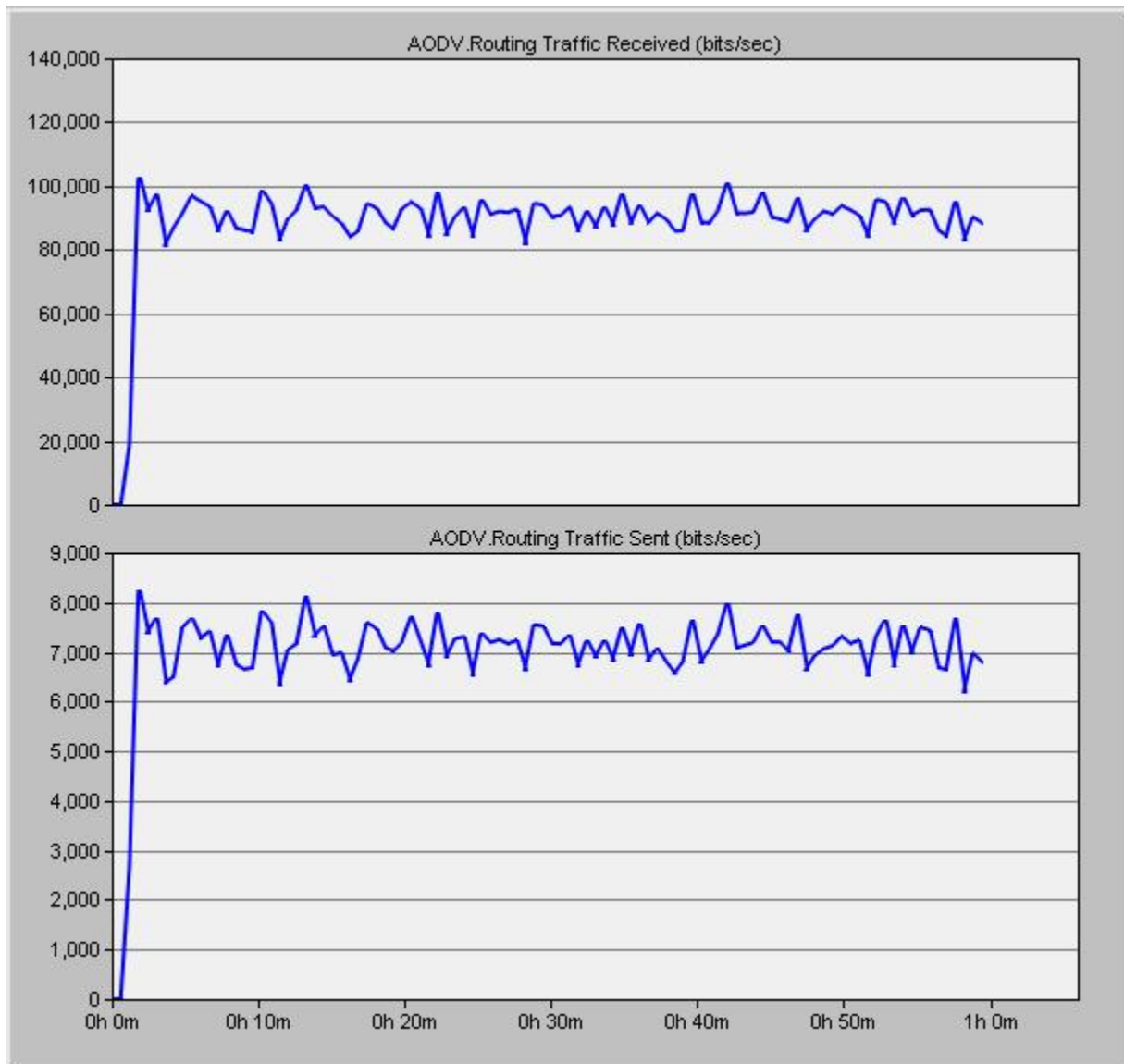
Figura următoare arată latenta introdusă de către protocolul AODV în rețeaua ad-hoc descrisă. Se observă că latenta nu este influențată de timpul necesar mecanismului de descoperire a rutelor, acest lucru fiind un avantaj al acestui protocol de rutare ad-hoc.



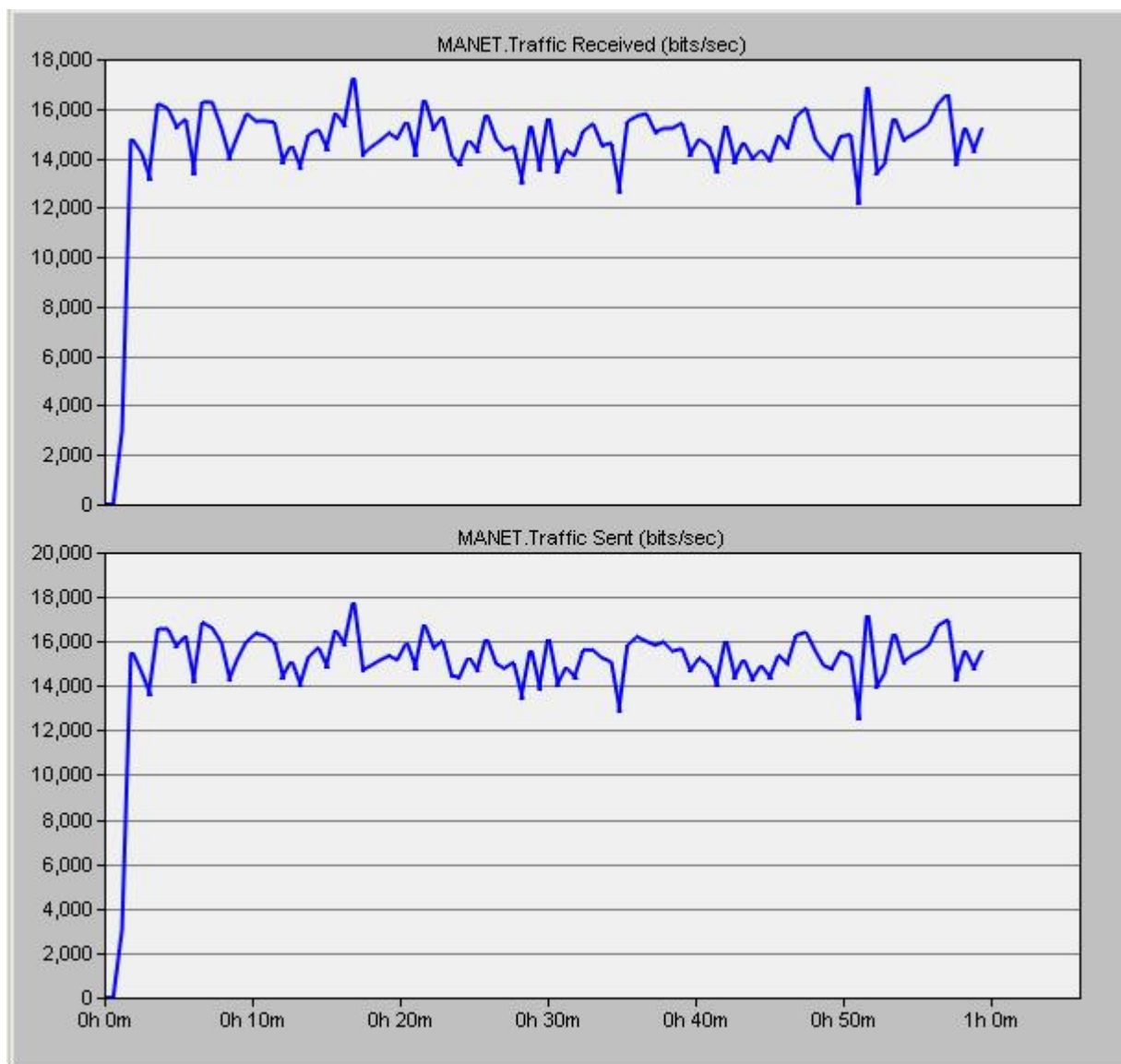
În figura următoare este afișat fluxul de date din rețea fără fir, observându-se un volum mare de informații. Acest fapt ar putea duce la o eventuală congestie a rețelei.

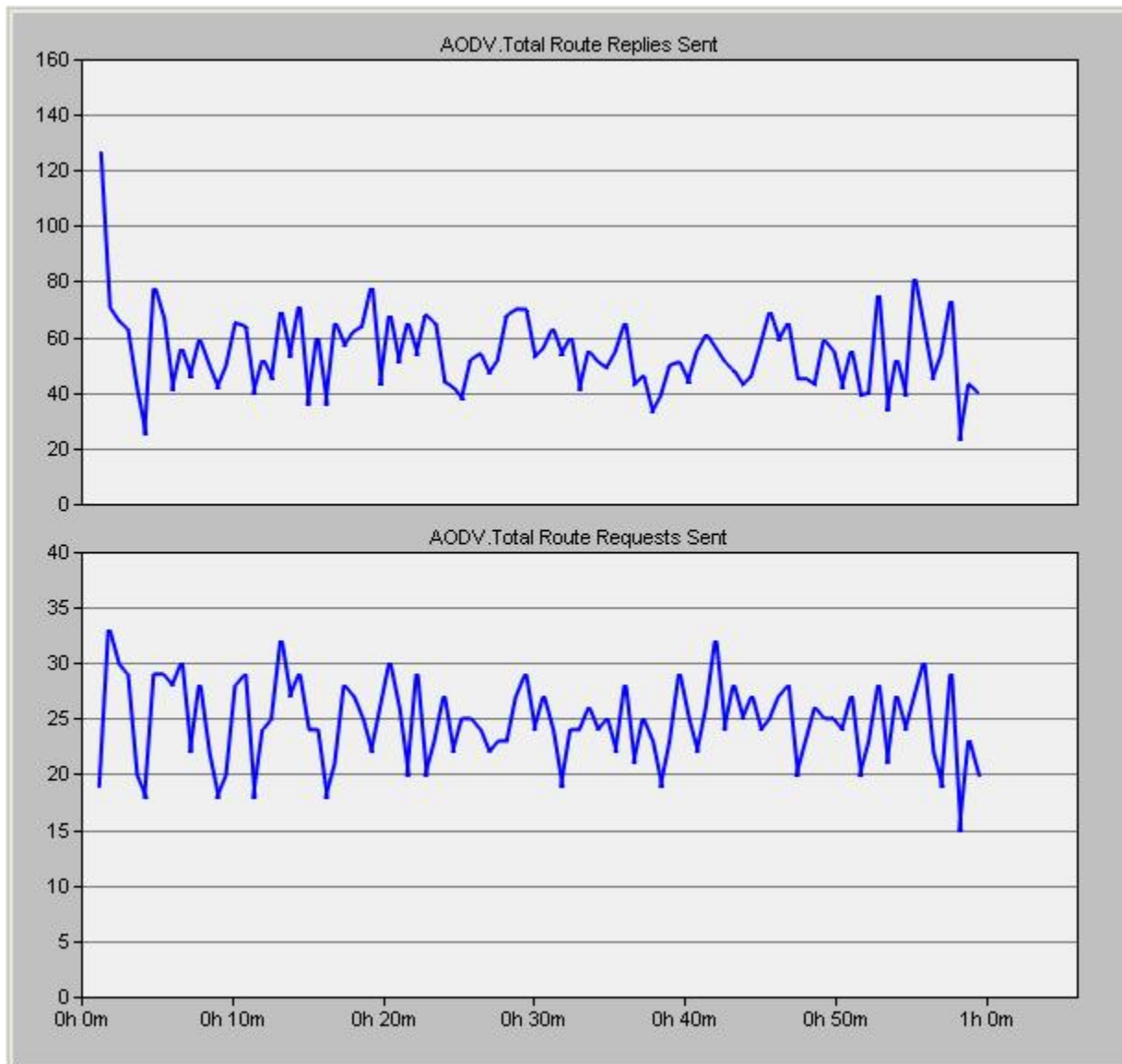


Protocolul AODV transmite în rețea o cantitate semnificativă de informație de rutare. Pachetele se propagă în toată rețeaua, în fiecare nod, cum se poate observa și în figura următoare, care arată volumul de informație suplimentară de rutare recepționat. Se poate observa de asemenea și caracterul periodic al protocolului AODV, care folosește mesaje de HELLO pentru întreținerea rutelor.



Putem observa traficul total, transmis si receptionat prin reseaua ad-hoc din scenariul nostru.





Numărul de cereri de rută este aproximativ constant pe durata simulării, variind în jurul valorii de 25 de cereri emise, după cum se poate vedea și în figură.

În urma experimentelor practice efectuate, am concluzionat că protocolul AODV îndeplinește scopul de a minimiza volumul de informație suplimentară pe care rutele prin sursă îl implică, însă pentru rate de mobilitate ridicate, AODV necesită emiterea mai multor pachete cu informație de rutare.

Cele mai importante rezultate ale studiilor efectuate în această lucrare sunt:

- Cu cât un protocol are un comportament mai puțin dependent de acțiuni periodice, cu atât performanța lui crește.
- Performanța protocolului crește pe măsură ce acesta este capabil să reducă numărul de noduri care trebuie anunțate despre schimbări apărute în topologia rețelei.

Întrebări

1. Care sunt pașii descoperirii unei rute folosind protocolul AODV?
2. Cum se realizează întreținerea rutelor în protocolul AODV?
3. Cum influențează numărul de stații din rețeaua fără fir timpul mediu de descoperire a rutelor?
4. Cum se completează tabela de rutare?
5. Când se generează mesaje Route Reply?