

Lucrarea 4.

Analiza performanței protocoalelor de rutare în sisteme wireless ad-hoc.

1. Scopul lucrării

Lucrarea are ca scop investigarea și evaluarea comparativă a două dintre protocoalele de rutare specifice sistemelor wireless ad-hoc utilizând simulatorul *GLoMoSim*. Protocoalele care vor fi analizate sunt *AODV* (*Ad-Hoc On Demand Distance Vector*) și *DSR* (*Dynamic Source Routing*), algoritmi de rutare din categoria protocoalelor reactive dar cu mecanisme de rutare diferite.

2. Modul de desfășurare al lucrării

2.1 Rutarea IP. Caracteristici generale

Scopul principal al unui protocol de rutare este acela de a livra pachetele de la sursă către destinație. Acest lucru trebuie făcut ținându-se cont de ordinea emiterii pachetelor și cu introducerea unei întârzieri minime de procesare.

Un bun algoritm de rutare are următoarele caracteristici: robust, se adaptează ușor variațiilor traficului din rețea; complexitate scăzută în calcularea rutelor optime, introducând întârzieri minime; stabil, capabil în a converge către soluții acceptabile chiar și la schimbări ale topologiei; optimal, identifică ruta cea mai potrivită pentru fiecare pachet în termeni de întârziere; echitabil, asigură aceleași privilegii tuturor nodurilor.

Ruta către destinație este estimată ținându-se cont de următoarele metrici: numărul de hop-uri, pe care un pachet trebuie să le traverseze pentru a ajunge la destinație; costul rutei, suma costurilor fiecărui traseu până la ajungerea la destinație, unde costul unei rute este înțeles în termenii de: întârzieri, lungimea cozilor de așteptare, lățime de bandă.

Sunt identificate trei tehnici de rutare de bază: rutare pe baza adresei de rețea (*routing by network address*), schimbarea etichetei (*label swpping*), rutarea sursei (*source routing*).

În cazul rutării pe baza adresei, numai adresa destinație este specificată în antetul fiecărui pachet. Fiecare nod trebuie să identifice următorul hop pe baza tabeli de rutare (*routing table*). Tabela de rutare este un set de valori asociate de tipul adresă-interfață. Această tehnică este tipică protocoalelor neorientate pe conexiune (*connectionless*). În conformitate cu locația la care tabela de rouate este consultată, distingem două categorii în cazul rutării pe baza adresei: rutare centralizată și rutare distribuită. La rutarea centralizată, consultarea tabeli de rutare este realizată într-un centru de control al rutării (*RCC*, *Routing Control Center*) ce stochează întreaga informație legată de topologie, trafic la nivelul fiecărui nod și disponibilitatea legăturilor, iar în cazul rutării distribuite, fiecare nod întreține propria tabelă de rutare pe baza informațiilor furnizate de nodurile adiacente.

În cazul tehnicii de schimbare a etichetei, fiecare pachet conține o anumită etichetă specifică pachetelor pe legătura dintre aceleași noduri. Fiecare nod înlocuiește eticheta curentă cu una nouă înainte de a transmite pachetul. Această tehnică este tipică protocoalelor orientate pe conexiune (*connection-oriented*), în particular pentru aplicațiile sensibile la întârzieri.

În cazul rutării sursei, antetul pachetului conține informații despre întreg traseul către destinație.

2.2 Mecanisme de rutare

Există două clase de protocoale cu rutare distribuită: protocoale pe baza vectorului distanță (*distance vector protocol*) sau protocoale cu rutare pe baza vectorului distanță (*DVR, Distance Vector Routing*) și protocoale pe baza stării legăturii (*link state protocol*) sau protocoale cu rutare pe baza stării legăturii (*LSR, Link State Routing*).

2.1 Protocoale pe baza vectorului distanță

În cazul protocoalelor pe bazate pe vector distanță, fiecare nod deține o tabelă de rutare conținând o listă cu cele mai scurte trasee (*shortest paths*) către celelalte noduri din rețea. La conectare, fiecare nod cunoaște doar propria adresă, putând transmite către toate nodurile vecine cu care este conectat. Prin broadcast, este transmis un pachet către toate nodurile vecine. Dacă distanța la un anumit nod este mai scurtă decât valoarea înscrisă în tabela de rutare, în tabelă este introdusă noua valoare apoi noul vector distanță este retransmis către toate nodurile. În momentul în care vectorul distanță nu mai operează schimbări în tabela de rutare, spunem că protocolul a fost convergent, topologia rețelei fiind cunoscută. Astfel, un router determină care dintre vecinii săi este următorul hop pe traseul cel mai scurt către destinație. Una din problemele acestui tip de protocoale este convergența scăzută și formarea buclelor de rutare. Buclele apar datorită faptului că fiecare router ia decizia asupra următorului hop local, pe baza unei informații care poate să nu fie tocmai actuală și datorită faptului că pachetele conținând informația de rutare (*routing update packets*) s-au propagat pe un traseu cu un număr mai mare de hop-uri.

2.2 Protocoale pe baza stării legăturii

Nodurile rețelelor ce folosesc algoritmi bazați pe starea legăturii mențin o hartă distribuită a rețelei, așa numita bază de date cu starea legăturii. Aceste hărți sunt înscrise imediat ce a fost sesizată o modificare a rețelei din punct de vedere al încărcării acesteia, al traficului vehiculat prin rețea. Fiecare nod este informat despre starea legăturii nodurilor adiacente lui și fiecare nod transmite un pachet cu starea legăturii către toate celelalte noduri. Nodurile salvează cele mai recente sosite pachete cu informații legate de starea legăturii. Deciziile de rutare sunt luate pe baza algoritmului numit alegerea traseului cel mai scurt (*shortest path first*). Convergența acestui tip de protocoale este mai rapidă decât în cazul celor bazate pe starea legăturii, însă necesită un timp mult mai mare de calcul al rutei celei mai scurte către fiecare destinație posibilă. De asemenea încărcarea rețelei cu pachete de rutare este mult mai mare deoarece acestea sunt transmise routerelor din întreaga rețea. O scurtă comparație între cele două tipuri de protocoale, scoate în evidență următoarele caracteristici ale protocoalelor pe baza stării legăturii:

1. Permite trasee multiple către destinație. Un protocol vector distanță selectează aleator unul din două trasee echivalente către destinație, în timp ce un protocol pe baza stării legăturii, având harta completă a rețelei, poate alege acel traseu ce oferă o capacitate de transfer (*bandwidth*) mai mare.
2. Rapid. Protocolul bazat pe vectorul distanță necesită, în cazul cel mai defavorabil, numărul de noduri minus unu pași în calcularea algoritmului, în timp ce în cazul protocolului pe baza stării legăturii printr-un simplu broadcast în rețea este transmisă întreaga informație, fiind necesar doar un calcul local al rutei optime.
3. Simplu. Dacă din anumite motive intervin schimbări în topologia rețelei, un alt gateway trebuind a fi selectat, acesta va trebui specificat în mod explicit în vectorul distanță. În cazul unui protocol pe baza stării legăturii, acest lucru se realizează mult mai simplu, fără o informare explicită.

Problema rutării într-o rețea are două componente: determinarea rutei (*route discovery*) și întreținerea rutei (*route maintenance*). În vederea comunicării cu o entitate destinație, sursa are nevoie de a determina o rută potrivită pe care să transmită pachetele către destinație. Dacă starea unor astfel de rute (legături, trasee), dintr-un motiv oarecare se modifică, este nevoie de uneori de determinarea unei alte rute pe care să fie transmisă informația către destinație.

Atât în cazul protocoalelor pe baza vectorului distanță cât și în cazul protocoalelor pe baza stării legăturii este îmbinată detreminarea rutei și întreținerea acesteia prin transmiterea unor pachete de informare a stării actuale a legăturii (*update packets*).

2.3 Rutarea în rețele wireless ad-hoc

În vederea comunicării cu o anume destinație, sursa trebuie să determine o rută potrivită transmiterii pachetelor către acea destinație. Așa cum am văzut, acest lucru este realizat de către protocolul de rutare. În continuare vor fi ilustrate câteva din problemele specifice ce apar în cazul unor rutări în medii wireless ad-hoc.

1. Transmitia între două noduri mobile pe un mediu wireless nu are întotdeauna aceleași caracteristici în ambele direcții. Astfel, rutele determinate pe baza protocoalelor de rutare clasice s-ar putea să nu funcționeze în anumite medii de transmisie (wireless).
2. Rețelele cablate sunt configurate în mod explicit, având un număr relativ mic de trasee între echipamentele active din rețea. Într-un mediu ad-hoc wireless, fiecare nod mobil este un ruter, putând exista trasee multiple între două entități (acest lucru crește timpul de calcul în estimarea rutei).
3. Transmiterea pachetelor de rutare în medii wireless consumă resursele rețelei, această situație putând devenii uneori critică. Transmiterea unor pachete conținând informație de rutare de către două noduri aflate la distanța unuia de altul nu influențează în nici un fel capacitatea de transfer din rețea. Însă, între două noduri aflate unul în raza celuilalt transmiterea unor astfel de pachete va afecta capacitatea de transfer dintre cele două entități.
4. În mediile ad-hoc wireless, rata de schimbare a topologiei rețelei poate fi mai mare decât aceea a transmiterii informației de rutare între noduri.
5. De asemenea, nodurile mobile într-o rețea wireless operează cu energie de emisie limitată, transmiterea informației de rutare influențând consumul de energie disponibil.

Din enumerarea problemelor ce pot apare într-un mediu wireless, putem spune că rularea unui protocol convențional într-o rețea wireless ad-hoc, considerând fiecare nod mobil un ruter, nu este o soluție optimă. În concluzie, performanța protocoalelor de rutare în cazul rețelelor wireless ad-hoc este influențată de doi factori majori: cât de des sunt inițiate procedurile de determinare a rutelor (*route discovery*) și cu ce frecvență este transmisă informația de întreținere a rutelor respective (*route maintenance*). Protocoalele de rutare în rețele wireless ad-hoc pot fi clasificate în două categorii: (1) protocoale de rutare proactive (*proactive routing protocols*) și (2) protocoale de rutare reactive (*reactive routing protocols*).

În cazul protocoalelor de rutare proactive, informația de rutare este întotdeauna cunoscută prin transmiterea unor mesaje de rutare continue. Această tehnică necesită din partea unui nod cunoașterea topologiei întregii rețele, această informație trebuind a fi transmisă în rețea. Cu alte cuvinte, aceste protocoale integrează procedura de determinare a rutei cu cea de întreținere a rutei prin transmiterea continuă a unor pachete de up-date, astfel că o rută este cunoscută a priori momentului în care este solicitată. În mediile foarte dinamice, aceste scheme de rutare sunt mai puțin eficiente. Dacă mobilitatea nodurilor crește, doar o fracțiune redusă din traficul de rutare inițial va putea fi folosit ca urmare a timpului de viață scăzut al unei rute. Avantajul protocoalelor de rutare proactive este însă acela că, o dată o rută solicitată, ea este imediat disponibilă în tabelul de rutare.

Protocoalele de rutare reactive invocă procedura de determinare a rutei numai la cerere. Atunci când o anumită rută este solicitată, este declanșată o procedură globală de căutare. Această procedură globală de căutare a protocoalelor de rutare reactive generează o cantitate semnificativă de trafic de control. Pe de altă parte însă, întârzierile în determinarea unei rute pot fi semnificative deoarece informațiile de rutare solicitate pot fi indisponibile la un moment dat al solicitării rutei respective. Cantitatea excesivă a traficului de control și întârzierile mari în determinarea rutei, nu recomandă protocoalele de rutare aplicațiilor în timp real.

3. Implementări ale protocoalelor de rutare în GloMoSim

3.1 DSDV (Destination-Sequenced Distance Vector)

În cazul protocolului de rutare **DSDV**, mesajele de rutare sunt schimbate între nodurile vecine. Fiecare nod **DSDV** menține o tabelă de rutare (*routing table*) ce indică numărul de hop-uri pentru fiecare destinație accesibilă. Tabela de rutare de la nivelul fiecărui nod menține informații pentru destinațiile din rețea. Atributele fiecărei destinații sunt: identificatorul următorului hop (*next hop ID*), numărul de hop-uri (*hop count*) și numărul secvență (*sequence number*).

DSDV este un protocol ce are la bază contorizarea numărului de hopuri și vectorul distanță (*hop-counting distance vector*). Acest lucru presupune transmiterea în mod periodic, din partea fiecărui nod, a unor tabele de rutare de tip broadcast de actualizare (*broadcast routing updates*).

PACHET UPDATE		TABELA DE ROUTARE				TABELA ACTUALIZATĂ			
DESTINAȚIE	3	DEST	METR.	URM. HOP	NR. SECV.	DEST	HOP-URI	URM. HOP	NR. SECV.
METRICĂ	6	2	4	1	24	2	4	1	24
URMĂTORUL HOP	1	3	5	5	19	3	6	1	20
NUMĂR SECVENȚĂ	20	7	1	7	58	7	1	7	58
									
		4	2	10	18	4	2	10	18

Menținerea tabelelor de rutare în cazul protocolului **DSDV**

DSDV etichetează fiecare rută cu un număr secvență și consideră ruta **ROUTE #1** mai favorabilă decât ruta **ROUTE #2**, dacă (1) ruta **ROUTE #1** are un număr secvență mai mare decât ruta **ROUTE #2** sau dacă (2) **ROUTE #1** și **ROUTE #2** au numărul secvență egal dar ruta **ROUTE #1** are o metrică mai mică.

Semnificația secvenței destinație (**DS**, *Destination Sequence*) este aceea că ruta actualizată are un număr secvență mai mare (fiecare nod din rețea incrementând unitar numărul secvență) iar vectorul distanță (**DV**, *Distance Vector*) indică metrica rețelei (numărul de hopuri până la destinație). În vederea întreținerii tabelelor de rutare, **DSDV** utilizează actualizări de rutare periodice (*periodic routing updates*) și actualizări de rutare declanșate (*triggered routing updates*).

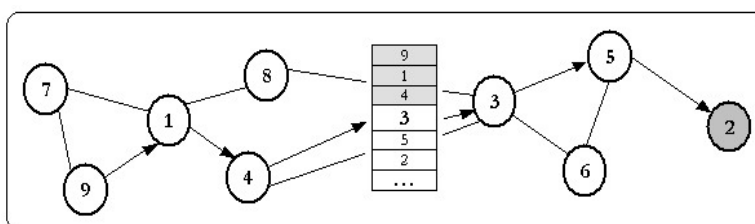
Actualizările de rutare declanșate sunt utilizate suplimentar actualizărilor de rutare periodice în vederea transmiterii cât mai rapide a informației de rutare în cazul modificării topologiei rețelei. Pachetele de rutare de actualizare includ destinațiile accesibile de către fiecare nod din rețea, numărul de hop-uri necesar atingerii fiecărei destinații cât și numărul secvență asociat fiecărei rute. La recepționarea unui pachet de rutare de actualizare, fiecare nod compară informația existentă în tabela de rutare cu informația de actualizare sosită. Rutele care au un număr secvență mai mic, sunt eliminate. În cazul rutelor având același număr secvență, informația de rutare de actualizare va fi înscrisă în tabela de rutare numai dacă metrica ce indică numărul de hopuri este mai bună. Contorul numărului de hop-uri este incrementat unitar dacă informația de actualizare sosită necesită unul sau mai multe hop-uri până la atingerea destinației. Noile rute înscrise în tabelele de rutare sunt transmise imediat nodurilor vecine din rețea. Pentru cazul în care o legătură către un nod devine inoperabilă, oricărei rute către acel nod îi este asociată o metrică infinită (*infinity metric*) și un număr secvență nou. Dacă unei rute îi este asociată metrică infinită și are un număr secvență egal sau mai mare cu al unei metrice finite, are loc transmiterea unor actualizări de rutare declanșate. În acest fel, rutele către noile destinații vor înlocui rutele cu metrica infinită.

Principalul avantaj al protocolului **DSDV** în comparație cu protocoalele de rutare clasice bazate pe vector distanță este faptul că **DSDV** garantează evitarea buclelor de rutare (*loop-free routes*). Totuși, **DSDV** are și câteva dezavantaje: încărcarea rețelei trafic excesiv destinat întreținerii rutelor și întârzieri în actualizarea tabelelor de rutare.

3.2 DSR (Dynamic Source Routing Protocol)

Protocolul **DSR** este un protocol de rutare simplu și eficient destinat utilizării în rețele wireless ad-hoc cu noduri mobile. **DSR** permite auto-organizarea și auto-configurarea rețelei fără administrarea unei infrastructuri a rețelei. Protocolul este alcătuit din trei mecanisme de bază: rutarea, determinarea rutei și întreținerea rutei.

Caracteristica de bază a protocolului **DSR** este rutarea sursei (**SR**, *Source Routing*), tehnica de rutare prin care sursa cunoaște succesiunea completă a nodurilor către destinație. Rutele către destinație sunt stocate (*route cache*) și fiecare pachet rutat conține în antetul său lista completă și ordonată a nodurilor pe care trebuie să le traverseze în traseul său către destinație. În acest fel, pachetul conține întreaga informație de rutare necesară.



Mecanismul de rutare în cazul protocolului **DSR**

Determinarea rutei este mecanismul prin care nodul care dorește să transmită un pachet către destinație, obține traseul către destinație. Pentru aceasta, nodul sursă transmite un pachet de solicitare a rutei **R-REQ** (*route request packet*) în care înscrie în lista rutelor propria-i adresă. În cadrul protocolului **DSR**, fiecare solicitare de rută este identificată în mod unic și inițial lista rutelor fiind goală. Pachetul de solicitare a rutei **R-REQ** este transmis din nod în nod, către destinație până când fie sursa, fie un alt nod este regăsit în lista de rutare. Fiecare nod de pe traseul pachetului de solicitare a rutei, de la sursă către destinație, inserează propria adresă în lista rutelor și transmite pachetul **R-REQ** către nodurile vecine. Suplimentar, fiecare nod stochează ultimele solicitări de determinare a rutei. Mecanismul de determinare a rutei este declanșat numai în cazul în care nu este regăsită o rută stocată. Când pachetul **R-REQ** atinge nodul destinație, acesta va răspunde cu un pachet de confirmare a rutei **R-REP**. Acest pachet este rutat înapoi de la sursă către destinație. Dacă o rută devine inoperabilă, nodul sursă este notificat prin transmiterea unui pachet **R-ERR**. În acest caz, nodul sursă va iniția un nou proces de determinare a rutei.

Menținerea/întreținerea rutelor este mecanismul prin care nodul care detectează o întrerupere a rutei declanșează mecanismul de determinare a unei noi rute. De cele mai multe ori, protocoalele de rutare integrează mecanismul de determinare a rutei cu cel de menținere al rutei. În cazul protocolului **DSR** nu este transmis nici un fel de mesaj de control pentru întreținerea rutei. Mecanismul de menținere a rutei este posibil numai dacă există o rută alternativă către destinație.

Principalul avantaj al protocolului **DSR** este acela că avem o cantitate foarte mică de trafic de rutare în cazul în care doar câteva surse comunică. Ceea ce generează trafic de rutare (încărcarea benzii) sunt mecanismele de determinare și întreținere a rutei și apar numai pentru caz în care se dorește stabilirea unei comunicații (necesitatea unei noi rute, modificarea topologiei rețelei, apariția unei rute inoperabile). Chiar dacă încărcarea rețelei poate fi scăzută prin stocarea rutelor în fiecare nod, rămâne în continuare sursă de trafic redundant informația de rutare a sursei inclusă în antetul fiecărui pachet. Cum în cazul rețelelor wireless ad-hoc capacitatea canalului radio este limitată, în rețele dinamice, având un număr mare de noduri, apare o cantitate semnificativă de trafic de rutare redundant și totodată întârzieri în procesul de comunicare.

Un avantaj al rutării sursei este însă acela că nu este necesar ca nodurile intermediare să rețină informația de rutare în vederea re-transmiterii pachetelor recepționate.

3.3 AODV (Ad-Hoc On-Demand Distance Vector)

Mecanismul de rutare **AODV** este o combinație a protocoalelor de rutare **DSR** (mecanism de determinare și menținere a rutelor la cerere) și **DSDV** (rutare din nod în nod, alocarea unui număr secvență și transmiterea unor pachete periodice de actualizare a rutei).

AODV adoptă un mecanism apesific de menținere a informației de rutare. Protocolul folosește tabele de rutare în care sunt introduse informații pentru fiecare destinație. Acest lucru diferențiază protocolul de rutare **AODV** de protocolul **DSR**.

În momentul în care este solicitată o anumită rută, nodul transmite un mesaj de tip broadcast (către toate nodurile vecine) într-un mesaj de tip **R-REQ**. Fără a utiliza mecanismul de rutare a sursei, la atingerea destinației sau a unui nod intermediar ce conține informația de rutare utilă, pe baza informației conținută în tabelele de rutare protocolul **AODV** răspunde cu un mesaj de tip **R-REP** către nodul sursă în vederea rutării informației către acesta. La nivelul fiecărui nod mobil este menținută o listă a rutelor active către fiecare nod vecin. Asemănător **DSDV**, protocolul de rutare **AODV** imprimă fiecărei informații de rutare transmise un număr secvență. Numărul secvență permite controlul asupra schimbărilor din topologia rețelei și evitarea buclelor de rutare. În momentul în care o rută devine inoperabilă sau au loc modificări ale topologiei rețelei este declanșată o nouă procedură de determinare a rutelor. Pentru un număr mare de noduri mobile, și în cazul protocolului **AODV**, încărcarea rețelei cu trafic de rutare devine semnificativă.

4. Experimente

Problema 1.

Analizați mecanismele de rutare în cazul protocoalelor descrise în lucrare. Rețineți în fiecare caz tipul de informație transmisă pentru determinarea rutei și modul de menținere a acesteia.

Notă

Sintetizând informația anterioară, pentru cazul protocoalelor ce trebuie analizate, putem spune că ambele protocoale sunt protocoale cu rutare la cerere (protocoale reactive). Astfel, procedura de determinare al rutei va fi declanșată doar pentru cazul în care un nod dorește să transmită informație unui alt nod din rețea. Cum ambele protocoale declanșează procedura de determinare a rutei la cerere, informația de rutare transmisă în rețea este semnificativ redusă în comparație cu protocoalele de rutare proactive. Mecanismele de rutare sunt însă diferite: **DSR** folosește mecanismul de rutare a sursei iar **AODV** are la bază tabele de rutare și numerotarea secvențelor transmise.

Problema 2.

Realizați un sistem wireless în care nodurile au implementat standardul **IEEE 802.11**. La nivel de strat fizic, pentru caracteristicile de transmisie ale nodurilor (câștigul antenei, putere de transmisie, nivel de recepție, etc.), păstrați setările implicite ale simulatorului. Ca și model de propagare va fi folosit **modelul two-ray**. Canalul va fi modelat prin intermediul unui canal de tip **AWGN**.

Realizați un scenariu wireless cu **50** de noduri mobile ce sunt inițial poziționate în mod aleator pe un teren având dimensiunile setate la valorile **1000x500 m**. Amplasarea nodurilor este aleatoare. Timpul de simulare va fi **400 s**. Ca și model de mobilitate a nodurilor se va folosi modelul **RADNOM-WAYPOINT**. Valorile parametrilor corespunzătoare modelului de mobilitate vor fi: **MOBILITY-WP-MIN-SPEED=0 m/s**, **MOBILITY-WP-MAX-SPEED=25 m/s**. Intervalului de pauză **MOBILITY-WP-PAUSE** i se vor aloca succesiv valori cuprinse între **0 s** (mobilitate continuă a nodurilor) și **400 s** (noduri fixe). Astfel, intervalului de pauză i se vor aloca următoarele valori: **0, 10, 50, 100, 200, 400 s**. Realizați **20** de sesiuni **CBR** între noduri alese în mod aleator în rețea. Fiecare sesiune va dura până la sfârșitul simulării. Transmiteți pachete având dimensiunea de **128** de octeți cu o frecvență de **5, 10 și 20** de pachete per secundă.

Scenariul 1.

Realizați setul de simulări propuse folosind protocolul de rutare *DSR*.

Scenariul 2.

Repetăți simulările folosind protocolul de rutare *AODV*.

6. Raportul tehnic

La încheierea sesiunii de laborator se va preda un raport al activității desfășurate ce va include rezultatele simulărilor pentru fiecare din scenariile rulate și răspunsul la întrebări.

Totodată, raportul de activitate va include și o analiză a rezultatelor obținute. În analiză se va acorda atenție următorilor parametri: (1) numărul de pachete recepționate, (2) încărcarea rețelei cu informație de rutare și (3) întârzierea medie a pachetelor recepționate.

Bibliografie

- [1] Amitava Mukherjee, Somprakash Bandyopadhyay, Debashis Saha, *Location Management and Routing in Mobile Wireless Networks*, Artech House, 2003
- [2] Ramjee Prasad, Marina Ruggieri, *Technology Trends in Wireless Communications*, Artech House, 2003