

# Validacija pouzdanosti OmneT++ simulatora kod bežičnih senzorskih mreža

Đorđe Banovac

**Sadržaj** — Ovaj rad sadrži informacije o simulaciji i pouzdanosti OmneT++ simulatora kada su u pitanju senzorske mreže. On sadrži informacije o različitim okruženjima koja se mogu koristiti prilikom simulacije bežičnih senzorskih mreža kao i rezultate simulacije jedne bežične senzorske mreže upotrebom OmneT++ simulacionog okruženja.

**Ključne reči** — Bežične senzorske mreže, OmneT++, simulacija, ZigBee

## I. UVOD

**U**RADU su predstavljena razičita okruženja za simulaciju, razlike između njih i poređenje ovih okruženja. Takođe, predstavljene simulacije scenarija iz [1] korišćenjem OmneT++ simulacionog okruženja. Cilj samog rada je simulacija i poređenje rezultata sa rezultima koji su dobijeni u [1]. Poređenjem se omogućava sagledavanje razlika između različitih simulacionih okruženja i koliko su rezultati bliski realnom slučaju. Glavni rezultati koji se porede su intenzitet uspešnosti i intenzitet odbačenih paketa.

Rad je organizovan tako da su u delu II predstavljene osnovne stvari koje se tiču bežičnih senzorskih mreža. Objasnjeno je šta je bežična senzorska mreža i predstavljena je struktura senzorske jedinice. U ovom delu je predstavljen i skup standarda koji se može koristiti u komunikaciji kod bežičnih senzorskih mreža. U delu III su predstavljena simulaciona okruženja i upoređena su u odnosu na efikasnost i hardverske zahteve. Deo IV je rezervisan za detaljnije predstavljanje OmneT++ okruženja pošto je to okruženje korišćeno za implementaciju simulacije. U ovom delu je u izvesnoj meri predstavljen i *Castalia* simulator jer su referentne simulacije

D. Banovac, Računarski fakultet, Srbija (telefon: +381-62-1754479; email: djordje.banovac@gmail.com)

implementirane uz pomoć ovog okruženja. U petom delu je predstavljena sama simulacija, model koji je korišćen za simulaciju. Takođe, prikazani su parametri simulacije, kao i rezultati. U poslednjem, šestom delu, su doneseni određeni zaključci koji se tiču OmneT++ simulatora i dobijenih rezultata.

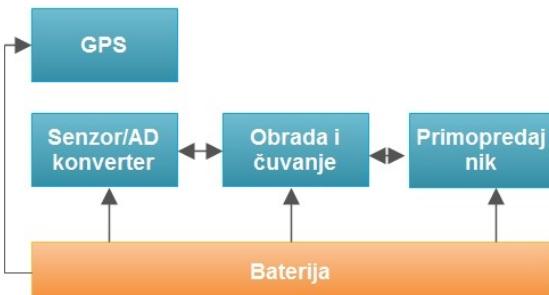
## II. BEŽIČNE SENZORSKE MREŽE

### A. Definicija bežičnih senzorskih mreža

Bežične senzorske mreže se mogu definisati kao samo-konfigurišuće bežične mreže bez infrastrukture čija je uloga da nadgledaju fizičke parametre ili parametre okruženja u kojem su postavljene. Parametri koji se prate mogu biti razni: temperatura, zvuk, vibracije, pritisak, kretanje ili zagađenje. Svi ti prikupljeni podaci se kasnije kooperativno šalju kroz mrežu putem senzorskih jedinica do glavne lokacije koja se naziva bazna stanica (*eng. sink*) ili koordinator i ona predstavlja interfejs između korisnika i mreže. Korisnik podatke dobija tako što šalje upit koordinatoru u kojem zahteva neke od prikupljenih podataka.

### B. Struktura senzorske jedinice

Senzorska jedinica je ako se gleda iz tehničke perspektive, uređaj koji prevodi parametre ili događaje iz fizičkog sveta u signale koje je moguće izmeriti i analizirati. Da bi to mogla da uradi, svaka jedinica je opremljena instrumentima koji sakupljaju informacije o fizičkom objektu ili procesu uključujući i nastanak događaja i promenu stanja objekta ili procesa (na primer promena temeprature ili promena pristiska). Međutim, mogućnosti senzorskih jedinica u bežičnoj senzorskoj mreži mogu da variraju. Jedna senzorska jedinica može da obavlja samo posao nadgledanja jednog fizičkog procesa, dok druga može da bude kompleksniji uređaj i da obavlja neki složeniji zadatak. Na primer, neki senzori mogu da koriste ultrazvuk, infracrvena sočiva ili radio-frekvenije sa različitim brzinama prenosa podataka i različitim nivoima kašnjenja koje se javlja tokom prenosa podataka. Opet, drugi senzori mogu da obavljaju jednostavan posao kao što je nadgledanje nekog parametra u okruženju poput temperature.



Sl. 1. Struktura senzorske jedinice

Senzorska jedinica se sastoji od četiri osnovne komponente: modul za prikupljanje podataka, modul za obradu, primopredajnik i izvor energije. To je prikazano na slici 1. Senzorska jedinica može da sadrži i neke dodatne module kao što je GPS. Modul za prikupljanje podataka se dalje može razložiti na dve jedinice: senzor i analogno/digitalni konvertor. Analogni signali koje generiše senzor se digitalizuju preko A/D konvertora i dalje šalju u jedinicu za obradu podataka. Jedinici za obradu podataka se jako često pridružuje i mala jedinica za čuvanje podataka. Jedinica za obradu podataka ima u sebi ugrađene procedure koje omogućavaju senzorskoj jedinici da vrši svu potrebnu komunikaciju sa drugim senzorskim jedinicama u mreži. Primopredajnik povezuje senzorsku jedinicu sa mrežom.[6] Poslednja komponenta i jedna od najvažnijih je izvor energije. Kod senzorskih jedinica, ovi izvori energije mogu imati dodatak u vidu solarnih celija čime se može značajno proizvesti vek rada senzorske jedinice. Ostale komponente koje se pojavljuju u senzorskoj jedinici su zavisne od aplikacije za koju se senzorska jedinica koristi.

### C. Skup protokola u bežičnim senzorskim mrežama

Grupa protokola koja se koristi od strane senzora je prikazana na slici 2. Ova grupa protokola je raspoređena u pet slojeva: aplikacijski sloj, transportni sloj, mrežni sloj, sloj veze podataka i fizički sloj, što odgovara TCP-IP referentnom modelu. Protokoli komunikacije se koriste u tri ravni i to: ravni za upravljanje energijom, ravni za upravljanje kretanjem i ravni za upravljanje poslovima. [6]



Sl. 2. Skup slojeva kod bežičnih senzorskih mreža

Mrežni sloj i aplikacijski sloj su definisani *ZigBee* standardom dok su sloj veze podataka i fizički sloj definisani standardom 802.15.4.

802.15.4 standard definiše bežične personalne mreže malih brzina. Cilj standarda je kreiranje baze nad kojom će se kreirati ostali viši slojevi (od mrežnog do aplikativnog sloja).[7] To znači da su ovim standardom definisani samo fizički i sloj veze podataka. Ova dva sloja potpuno definišu prenosni medijum koji se koristi za komunikaciju kao i mehanizme koji omogućavaju pristup prenosom medijumu. Sloj veze podataka je odgovoran za multipleksiranje podatka, detekciju okvira (*eng. frames*), kontrolu pristupa medijumu (*eng. Medium Access Control*), i kontrolu grešaka. Pošto senzori mogu biti mobilni, a okruženje u kojem se nalaze puno šumova, sloj veze podataka mora da ima mogućnost minimizacije kolizija sa onim što sused emituje pa se za pristup kanalu koristi CSMA-CA. Fizički sloj se bavi modulacijama koje bi trebale da budu robusne, selekcijom frekvencije, enkripcijom podataka, transmisionim i emisionim tehnikama. Sam standard omogućava korišćenje više frekvencijskih opsega, modulacija i omogućava korišćenje više različitih bitskih brzina. Ovo je prikazano u tabeli 1.

Aplikacijski sloj i mrežni sloj su definisani *ZigBee* standardom. *ZigBee* standard je standard *ZigBee* alianse i najčešće je proširenje 802.15.4 standarda. Ovaj standard unosi određena poboljšanja na nivou tri i pet. Ova poboljšanja uključuju autentifikaciju sa validnim čvorovima, enkripciju, rutiranje i prosleđivanje koje omogućuje *mesh* topologija sa višestrukim skokovima što sa 802.15.4 standardom nije definisano.

*ZigBee* standard definiše i dva tipa uređaja. To su *ZigBee* ruteri koji mogu imati dve uloge u mreži i *ZigBee* krajnji uređaji. Neophodno je imati jedan *ZigBee* ruter mreži koji će imati ulogu *ZigBee* koordinatora. *ZigBee* koordinator predstavlja glavni uređaj u mreži uz pomoć kojeg se mreža formira.

TABELA 1: FREKVENCIJSKI OPSEZI, MODULACIJE I BITSKE BRZINE

<i>Fizički sloj [MHz]</i>	<i>Frekvencija [MHz]</i>	<i>Modulacija</i>	<i>Bitska brzina [kb/s]</i>
780	779–787	O-QPSK/MPSK	250
868	868–868,6	BPSK	40
915	902–928	BPSK	40
950	950–956	GFSK	100
950	950–956	BPSK	20
2450 DSSS	2400–2483,5	O-QPSK	250

Mrežni sloj se bavi rutiranjem podataka koje dobije od transportnog sloja uz pomoć specifičnih protokola za rutiranje u bežičnim senzorskim mrežama. Takođe, uloge mrežnog sloja su i formiranje mreže, prijavljivanje i odjavljivanje uređaja sa mreže, otkrivanje i održavanje putanja.

### III. OKRUŽENJA ZA SIMULACIJU

Trenutno postoji više različitih simulatora koji omogućavaju simulaciju mrežnih okruženja. Glavne stvari na koje se obraća pažnja kod ovih simulatora su licenciranje, korisnički interfejs, efikasnost. [1] Upredni prikaz predstavljenih okruženja je prikazan u tabeli 2 na kraju ovog dela.

#### A. NS-2 Simulator

NS2 simulator (*eng. Network Simulator verison 2*) je simulator otvorenog koda i ima GNU licencu. To je simulator diskretnih događaja koji je specifično dizajniran kako bi se koristio za istraživanje mrežnih okruženja – bilo žičnih, bilo bežičnih. Jedan je od popularnijih simulatora zbog modularnosti i fleksibilnosti. Ovaj simulator sa sobom donosi veliki broj ugrađenih objekata pisanih u C++ programskom jeziku i preporučeno je korišćenje ovih objekata u implementaciji simulacija. Ostavljena je mogućnost kreiranja sopstvenih objekata što naprednim korisnicima može da bude od velike koristi. Rezultati simulacija se mogu prikazati grafički i tekstualno. U slučaju grafičkog prikazivanja rezultata potrebno je koristiti neki od alata kao što su NAM (*Network AniMator*), Xgraph ili Gnuplot.[1]

Prednosti ovog simulatora se ogleda u velikom broju dostupnih modela, realističnim modelima za mobilnost, lakoj simulaciji kompleksnih scenarija i popularnosti zbog svoje modularnosti.

NS-2 Simulator (ako se uporedi sa drugim popularnim simulatorima) nije preterano efikasan. Svaka promena implementiranog protokola dovodi do ponovnog kompjuiranja kako bi se omogućilo izvršavanje simulacije. Takođe, realni sistemi su previše kompleksni za modelovanje a i sam kod

može biti teško razumljiv i čitljiv. U slučaju kada se korsiti veliki broj čvorova, simulacija se značajno usporava.

#### B. QualNet

Ovaj simulator je komercijalna verzija GloMoSim simulatora ali takođe postoje i akademske licence ovog simulatora. Za razvoj je u potpunosti korišćen C++ i sam simulator je modelovan kao konačni automat. Slojevita arhitektura je korišćena prilikom kreiranja ovog simulatora: on se sastoji iz aplikacijskog sloja, transportnog sloja , mrežnog sloja , sloja veze podataka i fizičkog sloja. Sa ovim simulatorom je moguće implementirati modele žičnih i bežičnih mreža. Podržava veliki broj bežičnih i mobilnih protokola. Takođe podržava i senzorske mreže, MANET i WiMAX. Omogućeno je dizajniranje novih protokola, kreiranje scenarija i analiziranje performansi simularanih mreža.

Podržava izvršavanje simulacija u realnom vremenu i moguće je simuliranje mreža koje sadrže i po 20000 čvorova. Može se izvršavati i na klasterima, podžava paralelnu obradu i distribuiranu obradu, višejezgarne procesore i multi-procesorske sisteme. Obezbeđuje grafičko okruženje koje je lako za upotrebu i pruža napredne mogućnosti animiranja simulacija.

#### C. Omnet++

Omnet++ je simulator diskretnih događaja i otvorenog je koda. Besplatan je ukoliko se ne koristi u komercijalne svrhe. Potpuno je napisan u C++ jeziku i koristi se u istraživanju i u svrhe edukacije. Ovaj simulator nije specifično namenjen za simuliranje računarskih mreža. Sa ovim simulatorom je moguće simulirati bilo koji sistem u kojem se razmenjuju poruke između komponenti ili uređaja u sistemu i predstavlja moćan alat za simulaciju. Modularan je, hijerarhijski organizovan i proširiv. Iako sam simulator nema veliki broj integrisanih modula, moguće je preuzeti neku od biblioteka koje u sebi sadrže veliki broj gotovih modula koji su spremni za korišćenje. Takođe, korisnik može da kreira sopstvene module po potrebi. Za kreiranje novih modula je moguće koristiti ili C++ ili NED jezik koji se koristi za opisivanje modula. Simulator koristi grafičko okruženje *Eclipse*, dok se za izvođenje same simulacije se koristi ili grafički interfejs ili komandna linija.

Prednosti ovog simulatora su moćno grafičko okruženje. Ispravljanje grešaka koje se mogu javiti je lakše nego kod ostalih simulatora. Moguće je precizno modeliranje hardvera i fizičkih fenomena (kao što je fejding na primer).

#### D. Opnet

Opnet se može koristiti za izučavanje rada uređaja, protokola i komunikacionih mreža. Ovaj simulator koristi relativno moćno grafičko

okruženje i grafički editor za kreiranje mrežnih topologija i objekata koji mogu pripadati gotovo svim slojevima OSI modela. Objektno orijentisano programiranje se koristi za mapiranje grafičkog dizajna u implementaciju realnog sistema. Svaka topologija koja se kreira se može predstaviti i vizuelno. Sve parametre je moguće podešavati naknadno i isto tako je moguće ponavljati sve simulacije preko grafičkog interfejsa. Za organizovanje mreže se koristi hijerarhijska struktura. Opnet dolazi do izražaja kada se simuliraju mreže sa velikim brojem uređaja i kada samo mala promena može da bude kritična. Moguće je koristiti alate kao što su NetDoctor ili ACE za analizu implementirane mreže.

TABELA 2: PREGLED SIMULATORA

<i>Simulator</i>	<i>Tip</i>	<i>Operativni sistem</i>	<i>Efikasnost</i>	<i>Hardverski zahtevi</i>
Omnet++	Besplatan	Windows,Linux,Mac OS	Potrebno delimično rekompajliranje	512MB RAM, 400 MB HDD
QualNet	Komercijalni	Windows,Linux,Mac OS	Visoka	4GB RAM 5GB HDD
NS2	Besplatan	Mac OS X, Windows	Potrebno kompletno rekompajliranje	256MB RAM 5GB HDD
Opnet	Komercijalni	Windows	Visoka	256MB RAM 200 MB HDD

#### IV. OMNET++ I CASTALIA SIMULATOR

Simulacije u ovom radu su urađene u simualtoru OmneT++, iz tog razloga je ovaj simulator detaljnije obrađen o ovom delu rada. Takođe, u ovom delu je obrađen i *Castalia* simualtor pošto su referentne simulacije implementirane uz pomoć ovog okruženja.

##### A. *OmneT++ simulator*

OmneT++ je modularni objektno orijentisani simulator diskretnih događaja. Ima generičku arhitekturu i može se koristiti za rešavanje različitih vrsta problema:

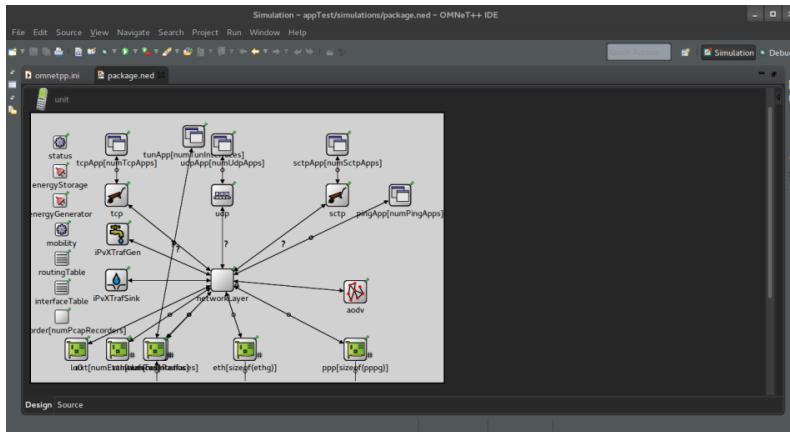
- Modelovanje žičnih i bežičnih računarskih mreža

- Modeliranje protokola
- Modeliranje redova
- Modeliranje multiprocesorskih sistema i distribuiranih hardverskih sistema
- Validiranje hardveske arhitekture
- Procenu performansi kompleksnih softverskih sistema
- Generalno, modeliranje bilo kojeg sistema koji se može predstaviti simulacijom diskretnih događaja.

OmneT++ je integrisano razvojno okruženje koje se bazira na *Eclipse* platformi i prikazano je na slici 3. Pored standardnih funkcionalnosti koje *Eclipse* pruža, OmneT++ dodaje nove funkcionalnosti koje omogućavaju konfigurisanje i kreiranje simulacionih modela uz pomoć NED jezika, grafičkog mrežnog editora i .ini fajlova koji se koriste za konfigurisanje samog modela. Takođe, moguće je i analizirati rezultate simulacija iz prikupljenih podataka.[2][3]

OmneT++ model se sastoji od modula koji komuniciraju razmenom tj. prosleđivanjem poruka. Postoje dve vrste modula: osnovni moduli (*eng. Simple Modules*) i kompozitni moduli. Za kreiranje osnovnih modula se koristi C++. Osnovni moduli se dalje mogu grupisati u kompozitne module. Treba naglasiti da je ovakva hijerarhija neograničena. Kompletan model koji predstavlja mrežu je u stvari jedan veliki kompozitni modul.

NED jezik se koristi za opisivanje strukture i simulacionog modela. NED omogućava deklarisanje osnovnih modula i povezivanje više osnovnih modula u kompozitni. Sam NED jezik je hijerarhijski organizovan što omogućava razbijanje jednog velikog modela na manje čime se olakšava sama implementacija modela. Sastoji se od modula čime se omogućava ponovno korišćenje koda i podržava interfejse i nasleđivanje.



Sl. 3. OmneT++ okruženje

Svi moduli mogu imati parametre koji se dodeljuju ili preko NED fajlova ili putem omnetpp.ini fajla koji predstavlja konfiguracioni fajl svakog modela i simulacije. Parametri određuju ponašanje svakog modula i koriste se za parametrizaciju topologije. Parametri mogu biti tekstualne, brojne vrednosti. Takođe, parametre je moguće učitati iz eksternih xml fajlova.

Poruke koje se razmenjuju između modula se prosleđuju ili preko konekcija koje se formiraju između njih ili direktno do drugih modula. Konekcije, koje predstavljaju realne fizičke linkove je moguće i parametrizovati. Takođe, parametri poput brzine prenosa, BER-a i kašnjenja usled prenosa mogu biti dodeljeni konekcijama.

Poruke koje se razmenjuju između modula mogu sadržati kompleksne strukture podatka što bi predstavljalo dodatak standardnim atributima koji sačinjavaju informacije kao što je vreme prosleđivanja poruke. U simulaciji poruke u stvari predstavljaju frejmove ili pakete jedne računarske mreže. Osnovni moduli uglavnom koriste gejtove za prosleđivanje poruka iako je moguće te iste poruke direktno poslati odredišnom modulu. Gejtovi su ulazni i izlazni interfejsi modula. Poruke se šalju kroz izlazne a primaju kroz ulazne gejtove. Gejtovi dva podmodula jednog kompozitnog modula mogu direktno da se povežu.

OmneT++ podržava paralelnu i distribuiranu simulaciju. Sami modeli nemaju neku posebnu implementaciju kako bi se simulacija izvršavala u paralelnom režimu rada. Simulacije se mogu izvršavati na dva načina: preko grafičkog interfejsa koji je koristan prilikom demonstracija i rešavanja problema koji se mogu javiti u simulaciji ili preko komandne linije koja je pogodna za *batch* izvršavanja.

### B. Castalia simulator

Ovo je robusni simulator koji se bazira na OmneT++ simulatoru. Omogućava simulaciju raznih vrsta mreža, a pogotovo je dobar za simulaciju BAN (*eng. Body Area Network*) mreža. Kao osnova je iskorišćen OmneT++, pa fokus kod ovog simulatora nije jezgro i način razmene poruka već modeli i sveukupni dizajn modela koji se koriste. Ovaj simulator, kao i OmneT++, nije dizajniran samo za određenu senzorsku platformu već se koristi za kreiranje modela koji bi se koristili za validaciju realnih implementacija. [4]

*Castalia* sadrži napredni model kanala za prenos kod kojeg su definisani gubici na prenosnom putu, kao i varijacije u snazi signala koje se javljaju tokom prenosa. Mobilnost senzora je takođe implementirana. Model koji se koristi za radio kanal se bazira na realnim merenjima. Uspešnost prijema signala zavisi od više faktora: verovatnoća prijema se bazira na SNIR vrednostima, odnosu snage signala prema snazi šuma, BER vrednostima, veličini paketa koji se šalje. Takođe, dozvoljene su različite vrednosti snage signala koju emituje predajnik. Pored svega ovoga, implementirana je i simulacija samog fizičkog procesa kojim se doprinosi poboljšanoj realističnosti same simulacije. Struktura svih modela koji se koriste je, kao i kod OmneT++ simulatora, implementirana korišćenjem NED jezika. To znači da je moguće modifikovanje parametara koje postojeći moduli imaju ili pak, kreiranje novih modula na osnovu sopstvenih potreba. Takođe, hijerarhijska struktura i korišćenje osnovnih i kompozitnih modela je preuzeto iz OmneT++ simulatora. Pošto je osnova *Castalia* simulatora OmneT++, i performanse koje ostvaruje su na istom nivou, ali za razliku od OmneT++ simulatora *Castalia* podržava samo *Linux* platformu.

## V. SIMULACIJA

Scenario koji je korišćen u simulaciji je preuzet iz [1]. U tom radu ovaj scenario je implementiran uz pomoć *Castalia* simulatora. Cilj ove simulacije je implementacija tog istog scenarija korišćenjem drugog simulatora i upoređivanje dobijenih rezultata.

Za implementaciju simulacije se koristi OmneT++ simulator i *Inet framework* verzije 3.0.0. koji sadrži sve potrebne module za simulaciju. Simulacija se izvršava na *mesh* topologiji koja se sastoji od koordinatora i dve senzorske jedinice. Sve senzorske jedinice će slati podatke prema koordinatoru (koristi se ping aplikacija). Veličina paketa koji aplikacija generiše je 56 bajta. Svi ostali osnovni parametri simulacije su prikazani u tabeli 3.

Na sloju veze podataka se koristi Inet implementacija 802.15.4 standarda. Ova 802.15.4 implementacija u sebi nema implementirane *beacon* transmisije pa samim tim nije moguće korišćenje garantovanih vremenskih okvira (*eng.*

*Guaranteed Time Slots*)[5]. Parametri 802.15.4 sloja veze podataka su prikazani u tabeli 4. Frekvencijski opseg koji se koristi u simulaciji je 780 MHz dok je za modulaciju izabrana MPSK. Modulacija DSSS-OQPSK i frekvencijski opseg od 2.4 GHz nisu implementirani u simulacionom modelu iako Inet sadrži ovaj tip modulacije. Razlog zbog kojeg ova modulacija nije implementirana u model su problemi prilikom simulacionog procesa koji dovode do preuranjenog prekida simulacije.

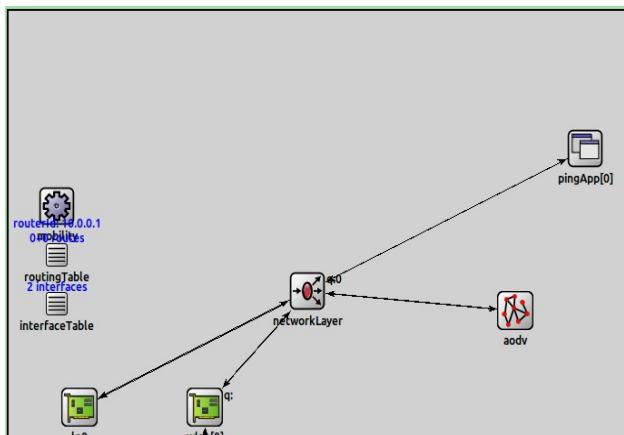
Svaka simulacija traje 120 sekundi i ponavlja se 50 puta. Aplikacija generiše sedam različitih tipova opterećenja – šalje 5, 10, 20, 40, 60, 80 i 100 paketa u sekundi. U simulaciji se koristi AODV protokol za rutiranje.

TABELA 3: PARAMETRI SIMULACIJE I HARDVER

Operativni sistem	Ubuntu 14.04
Hardver	Procesor Intel i7 QM3632, RAM 8GB
Simulator	Omnet++ verzija 4.6
Framework	Inet 3.0.0
Trajanje simulacije	120s
Broj ponavljanja	50
Aplikacija	PingApp
Sloj veze podataka	IEEE802154NarrowbandMac
Fizički sloj	Frekvencijski opseg 780 MHz, Modulacija 16MPSK
Radio Kanal	IEEE802154NarrowbandScalarRadioMedium

TABELA 4: PARAMETRI 802.15.4 SLOJA VEZE PODATAKA

MTU	118 bajta
Veličina zaglavlja	72 bita
Tip odstupanja	Eksponencijalni
Maksimalni broj retransmisija	3
Maksimalni broj CSMA odstupanja	4
Minimalni eksponent odstupanja	3
Maksimalni eksponent odstupanja	3



Sl. 4. Prikaz korišćene senzorske jedinice

Za radio prenos, odnosno za model kanala je korišćen model propagacije koji je na odgovarajući način povezan sa implementacijom fizičkog sloja korišćene bežične kartice. Parametri predajnika i prijemnika su prikazani u tabelama 5 i 6 respectivno.

TABELA 5: PARAMETRI PRIJEMNIKA

Detekcija energije	-90dbm
Osetljivost	-100dbm
SNIR prag	-8dB
Modulacija	16MPSK
Minimalna snaga interferencije	-120dbm

TABELA 6: PARAMETRI PREDAJNIKA

Snaga	0.5 mW
Brzina prenosa	250 kbps
Veličina Zaglavlja	48 bita
Modulacija	16MPSK

Simulirana su tri različita razmeštaja čvorova. U prvom su sve senzorske jedinice uključujući i koordinatora raspoređene u niz dok se u drugom koordinator nalazi između dve senzorske jedinice. Treći scenario uključuje kretanje. U ovom razmeštaju, senzorska jedinica 1 se nalazi na udaljenosti od 10 metara od koordinatora i kreće se brzinom od 0.2m/s prema koordinatoru.

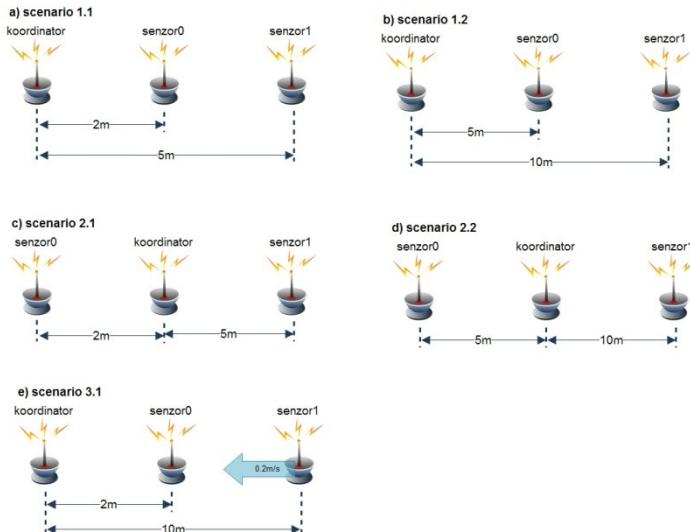
Pozicije koordinatora i svih čvorova su prikazane u tabeli 7 i slici 5, dok je i prikaz simulacije preko grafičkog interfejsa prikazan na slici 6.

Parametri koji su praćeni ovom simulacijom su:

- Procenat izgubljenih paketa
- Intenzitet uspešnosti (*eng. packet successful rate*) – definiše se kao odnos između ukupnog broja primljenih paketa i ukupnog broja emitovanih paketa
- Intenzitet odbačenih paketa (*eng. packet dropped rate*) – definiše se kao odnos između ukupnog broja izgubljenih paketa i ukupnog broja emitovanih paketa
- Opterećenje: u ovom slučaju opterećenje predstavlja različit intenzitet slanja paketa. Intenziteti su 5 paketa/s, 10 paketa/s, 20 paketa/s, 40 paketa/s, 60 paketa/s, 80 paketa/s i 100 paketa/s

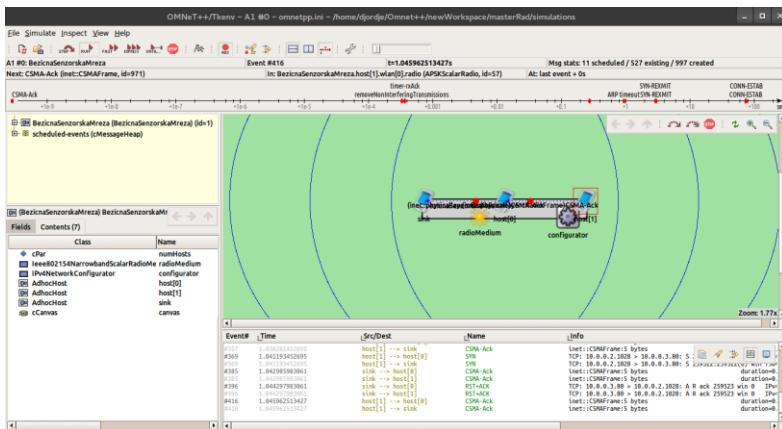
TABELA 7: PREGLED RAZMEŠTAJA SENZORA

<i>Scenario</i>	<i>Pozicija koordinatora</i>	<i>Pozicija senzora0</i>	<i>Pozicija senzora1</i>	<i>kretanje</i>
Scenario 1.1	X=1m Y=1m	X=3m Y=1m	X=6m Y=1m	Stacionarno
Scenario 1.2	X=1m, Y=1m	X=5m Y=1m	X=11m Y=1m	Stacionarno
Scenario 2.1	X=3m Y=1m	X=1m Y=1m	X=8m Y=1m	Stacionarno
Scenario 2.2	X=5m Y=1m	X=1m Y=1m	X=15m Y=1m	Stacionarno
Scenario 3.1	X=1m Y=1m	X=2m Y=1m	X=10m Y=2m	Kretanje senzora 1 prema koordinatoru brzinom 0.2m/s



Sl. 5. Prikaz razmeštaja senzora

U ovakvoj konfiguraciji modela, potrošnju energije nije moguće ispratiti jer implementacija 802.15.4 bežične kartice i implementacija modula za bateriju preko kojeg bi se pratila potrošnja energije nisu na najbolji način povezani unutar Inet paketa što dovodi do preuranjenog završetka simulacije.



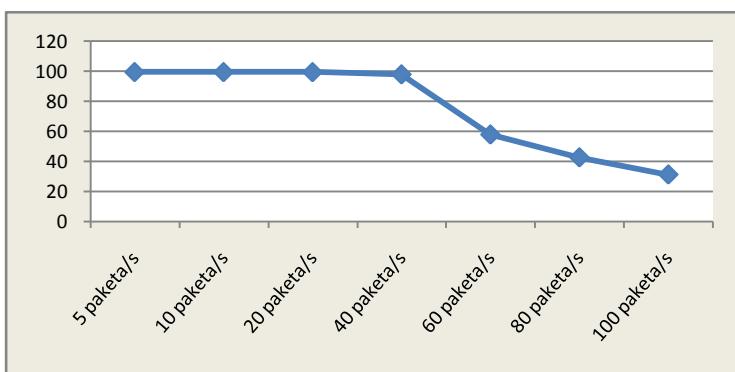
Sl. 6. Omnet++ simulacija korišćenjem grafičkog simulatora

## VI. REZULTATI

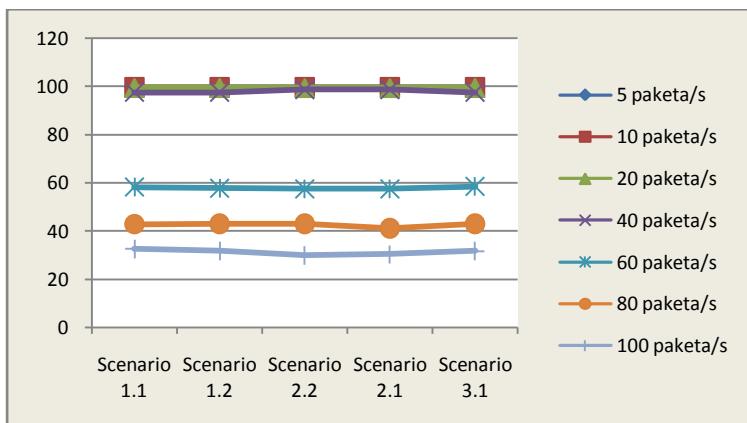
*A. Intenzitet uspešnosti*

Na slikama 7 i 8, i u tabelama 8 i 9 su prikazani rezultati simulacije koji se tiču intenziteta uspešnosti. Srednji intenzitet uspešnosti celokupne simulacije iznosi 75.52%. Na slici 6 je prikazan srednji intenzitet uspešnosti po opterećenju dok je na slici 7 prikazan intenzitet uspešnosti po svakom scenariju.

Rezultati su pokazali da je najbolji intenzitet uspešnosti pri intenzitetu od 5 paketa/s i da nema velike razlike u rezultatima između 5, 10, 20, 40 paketa/s. Drastičan pad u intenzitetu uspešnosti se javlja posle opterećenja od 40 paketa/s. Ovaj pad je uslovjen opterećenjem. Analiza je pokazala da koordinator u slučajevima visokog opterećenja pravi relativno dugačke pauze pre nego što pošalje odgovor na primljen zahtev. Takođe, javlja se velika zauzetost kanala od strane jednog ili drugog senzora koji emituje podatke.



Sl. 7. Grafik za srednji intenzitet uspešnosti



Sl. 8. Grafik srednjeg intenziteta uspešnosti za svaki scenario

TABELA 8: SREDNJI INTENZITET USPEĐNOSTI PO OPTEREĆENJU

<i>Opterećenje</i>	<i>Intenzitet uspešnosti</i>
5 paketa/s	99,62%
10 paketa/s	99,60%
20 paketa/s	99,57%
40 paketa/s	98,9%
60 paketa/s	57,92%
80 paketa/s	42,62%
100 paketa/s	31,19%

Ukoliko se posmatra intenzitet uspešnosti pojedinačnih scenarija, vidi se da različite pozicije senzora i kretanje (koje se javlja u scenariju 3.1) za isto opterećenje ne utiču na rezultate. U ovom slučaju to je tako jer je udaljenost senzora od koordinatora minimalna. Oba senzora se nalaze u dometu koordinatora za svaki razmeštaj. Zaključak je da i jedan i drugi senzor prilikom slanja paketa direktno komuniciraju sa koordinatorom što znači da je formirana topologija zvezde.

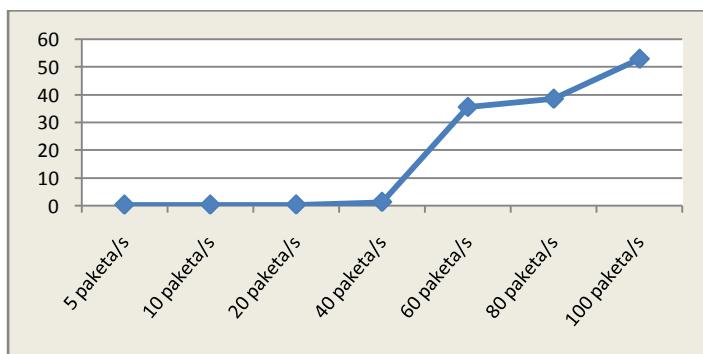
Dobijeni rezultati se razlikuju kada se uporede sa onim predstavljenim u [1]. U ovom slučaju razlika između TestBed referentnih rezultata i dobijenih rezultata iznosi 1.32% za senzor0 i 6.17% za senzor1. Razlika između referentnih TestBed i referentnih simulacionih rezultata za senzor0 je 2.51% i -1.88% za senzor1.

TABELA 9: UPOREDIVANJE REZULTATA – SREDNJI INTENZITET USPEŠNOSTI

<i>Referentni testBed rezultati</i>	Senzor0	98,32%
	Senzor1	93,44%
<i>Referentni simulacioni rezultati</i>	Senzor0	95,81%
	Senzor1	95,32%
<i>Dobijeni rezultati</i>	Senzor0	99,64%
	Senzor1	99,61%

### B. Intenzitet odbačenih paketa

Srednji intenzitet odbačenih paketa za celokupan scenario iznosi 18,54%. Slike 9 i 10, i tabele 10 i 11 prikazuju intenzitet odbačenih paketa. Slika 9 prikazuje intenzitet odbačenih paketa na osnovu opterećenja dok slika 10 prikazuje intenzitet odbačenih paketa za svaki scenario.

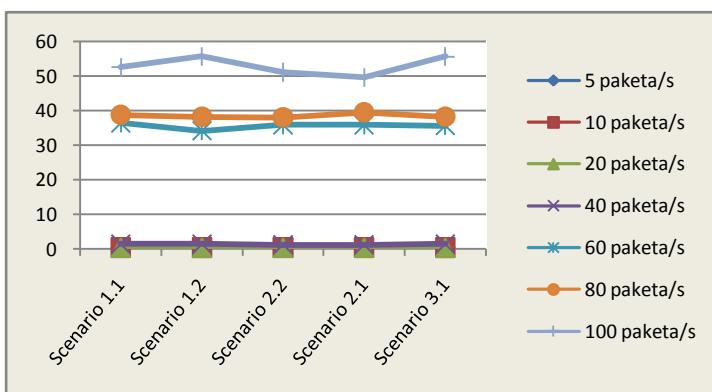


Sl. 9. Grafik srednjeg intenziteta odbačenih paketa za svako opterećenje

Opet, kao i kod intenziteta uspešnosti, i intenzitet odbačenih paketa naglo raste posle opterećenja od 40 paketa/s. To je takođe uslovljeno opterećenjem. Takođe, i u ovom slučaju različite pozicije i kretanje ne utiču na rezultate. Za različite scenarije i isto opterećenje intenzitet odbačenih paketa je približno isti što je prikazano na slici 9.

TABELA 10: SREDNJI INTENZITET ODBAČENIH PAKETA PO OPTEREĆENJU

<i>Opterećenje</i>	<i>Intenzitet odbačenih paketa</i>
5 paketa/s	0,37%
10 paketa/s	0,39%
20 paketa/s	0,42%
40 paketa/s	1,36%
60 paketa/s	35,6%
80 paketa/s	38,6%
100 paketa/s	53,01%



Sl. 10. Grafik srednjeg intenziteta odbačenih paketa za svaki scenario

I u ovom slučaju, dobijeni rezultati se razlikuju od referentnih. Razlika u dobijenim rezultatima u odnosu na refentne *TestBed* rezultate kod senzora0 iznosi -0.42%, a kod senzora1 ta razlika iznossi -0,87%. Rezlika između referentnih simulacionih i *TestBed* rezultata iznosi 3.07% za senzor0 i 3.11% za senzor1.

TABELA 11: UPOREDIVANJE REZULTATA – SREDNJI INTENZITET ODBAČENIH PAKETA

<i>Referentni testBed rezultati</i>	Senzor0	0,85%
	Senzor1	1,25%
<i>Referentni simulacioni rezultati</i>	Senzor0	3,92%
	Senzor1	4,36%
<i>Dobijeni rezultati</i>	Senzor0	0,43%
	Senzor1	0,38%

## VII. ZAKLJUČAK

Omnet++ simulator je dobro i moćno okruženje. Sama instalacija ne predstavlja problem, bez obzira koji operativni sistem se koristi. Simulacija, kada se jednom model ispravno konfiguriše, teče glatko i bez problema. Preporuka je korišćenje višejezgarnih procesora čime se simulacija značajno ubrzava. Predstavljanje rezultata nije u potpunosti automatizovano i potrebna su dodatna izračunavanja ili korišćenje nekog drugog alata kojim bi se došlo do neophodnih rezultata. Korišćeni Inet *framework* sadrži veliki broj modula pa je moguće kreirati veliki broj različitih modela za simulaciju. Ali, i dalje je u fazi razvoja i javljale su se greške i situacije u kojima moduli “ne sarađuju” dobro jedan sa drugim. U ovom slučaju primer je modulacija O-QPSK koja nije implementirana u potpunosti zbog čega se javljaju problemi, dok je drugi primer modul za bateriju koji nije povezan sa mrežnom karticom u kojoj je implementiran 802.15.4 standard.

Dobijeni rezultati u simulaciji se razlikuju u odnosu na referentne rezultate. Rezultati dobijeni OmneT++ simulatorom su procentualno bolji nego rezultati dobijeni iz referentne simulacije i samo neznatno bolji od referentnih rezultata koji su dobijeni TestBed eksperimentom. Ukoliko se uporede razlike između TestBed rezultata i rezultata koji su dobijeni simulacijama, vidi se da je kod intenziteta uspešnosti *Castalia* ostvarila bolje rezultate dok je kod intenziteta odbačenih paketa slučaj obrnut. Opet, ni jedan od pomenutih simulatora nije ponovio rezultate koji su dobijeni TestBed eksperimentom ali je pretpostavka da bi se dodatnim podešavanjem parametara rezultati još više približili realnom slučaju. Zaključak je da je kao i *Castalia* simulator, OmneT++ simulator dovoljno pouzdan i da se može primenjivati u modelovanju i validaciji senzorskih mreža. Sa daljim razvojem kako simulatora i paketa koji se koriste za simulaciju, rezultati će biti još tačniji i bliži realnom slučaju.

## LITERATURA

- [1] (Alex), ZhengYi Guan. *A Reliability Evaluation of Wireless Sensor Network*. 2011.
- [2] Varga, András. "OMNeT++ - User Guide." *OMNeT++ - Discrete Event Simulator*. <https://omnetpp.org/>.
- [3] Varga, András. "OMNeT++ - Manual." *OMNeT++ - Discrete Event Simulator*. <https://omnetpp.org/>.
- [4] Boulis, Athanassios. *Castalia - A simulator for Wireless Sensor Networks and Body Area Networks - User's Manual*. 2011.
- [5] IEEE Computer Society. "IEEE Standard Assosiation - IEEE Get Program." *IEEE-SA - The IEEE Standards Assosiation*. <http://standards.ieee.org/>
- [6] Sohraby, Kazem, Daniel Minoli, and Taeib Znati. *Wireless sensor networks: technology, protocols, and applications*. John Wiley & Sons, Inc., 2007.
- [7] Farahani, Shahin. *ZigBee Wireless Networks And Trancivers*. Oxford: Elsevier Ltd, 2008.

## ABSTRACT

Purpose of this paper is to present reliability evaluation of OmneT++ simulation enviroment regarding wireless sensory networks. This paper also contains basic information about wireless sensory networks and various simulation enviroments that can be used for wireless sensory network simulation.

## A RELIABILITY EVALUATION OF OMNET++ SIMULATOR

Dorđe Banovac