

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2020

Bc. Martin Rimeg



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

# POROVNÁNÍ SIMULAČNÍCH PROSTŘEDÍ PRO ANALÝZU BEZDRÁTOVÝCH TECHNOLOGIÍ

COMPARISON OF SIMULATION ENVIRONMENTS FOR ANALYSIS OF WIRELESS TECHNOLOGY

## DIPLOMOVÁ PRÁCE

MASTER'S THESIS

## AUTOR PRÁCE

AUTHOR

Bc. Martin Rimeg

## VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Anna Kubánková, Ph.D.

BRNO 2020

# Diplomová práce

magisterský navazující studijní obor **Telekomunikační a informační technika**

Ústav telekomunikací

**Student:** Bc. Martin Rimeg

**ID:** 186172

**Ročník:** 2

**Akademický rok:** 2019/20

**NÁZEV TÉMATU:**

## **Porovnání simulačních prostředí pro analýzu bezdrátových technologií**

**POKYNY PRO VYPRACOVÁNÍ:**

Seznamte se se standardy 802.11 a s Rate Adaptation Algoritmy (RAA) používanými v těchto standardech. Dále se seznamte se simulačními prostředími NS-3 a OMNeT. Navrhněte scénář, ve kterém bude možné získávat klíčové parametry pro analýzu Rate Adaptation Algoritmů. Vytvořte tento scénář v programech NS-3, OMNet a v reálném prostředí. Analyzujte a porovnejte funkčnost algoritmů v různých prostředích. Porovnejte možnosti simulace přenosového kanálu v porovnávaných programech.

**DOPORUČENÁ LITERATURA:**

[1] OMNet++. [online]. [cit. 2019-09-06]. Dostupné z: <https://omnetpp.org/>

[2] NS-3. [online]. [cit. 2019-09-06]. Dostupné z: <https://www.nsnam.org/>

**Termín zadání:** 3.2.2020

**Termín odevzdání:** 1.6.2020

**Vedoucí práce:** Ing. Anna Kubánková, Ph.D.

**prof. Ing. Jiří Mišurec, CSc.**  
předseda oborové rady

**UPOZORNĚNÍ:**

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Táto práca je zameraná na problematiku bezdrôtových sietí podľa štandardu IEEE 802.11. Hlavným predmetom skúmania sú Rate Adaptation Algoritmy (RAA). Práca ďalej obsahuje popis simulačných prostredí NS-3 a OMNeT z hľadiska adaptačných algoritmov. V závere práce sa nachádza zhrnutie problematiky simulácií bezdrôtových sietí v prostrediach NS-3 a OMNeT a ich porovnanie so skutočným meraním parametrov siete.

## **KLÚČOVÉ SLOVÁ**

Bezdrôtové siete, iPerf3, Meranie prenosovej rýchlosti, NS-3, OMNeT, Prenosová rýchlosť, Priepustnosť siete, Rate Adaptation Algoritmy, Štandard IEEE 802.11, Simulácie sietí, TamoSoft Throughput Test.

## **ABSTRACT**

This work is focused on the issue of wireless networks according to the IEEE 802.11 standard. The main subject of research is the Rate Adaptation Algorithms (RAA). The work also contains a description of simulation environments NS-3 and OMNeT in terms of adaptation algorithms. At the end of the work there is a summary of wireless network simulations in NS-3 and OMNeT environments and their comparison with the actual measurement of network parameters.

## **KEYWORDS**

Wireless Networks, iPerf3, Rate Measurement, NS-3, OMNeT, Rate, Network Throughput, Rate Adaptation Algorithms, IEEE 802.11 Standard, Network Simulation, TamoSoft Throughput Test.

RIMEG, Martin. *Porovnaní simulačních prostředí pro analýzu bezdrátových technologií*. Brno, 2020, 82 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedúci práce: Ing. Anna Kubánková, Ph.D.

## VYHLÁSENIE

Vyhlasujem, že svoju diplomovú prácu na tému „Porovnaní simulačních prostředí pro analýzu bezdrátových technologií“ som vypracoval samostatne pod vedením vedúceho diplomovej práce, s využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej diplomovej práce ďalej vyhlasujem, že v súvislosti s vytvorením tejto diplomovej práce som neporušil autorské práva tretích osôb, najmä som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a/alebo majetkových a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona Českej republiky č. 121/2000 Sb., o práve autorskom, o právach súvisiacich s právom autorským a o zmene niektorých zákonov (autorský zákon), v znení neskorších predpisov, vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákonníka Českej republiky č. 40/2009 Sb.

Brno .....

.....

podpis autora

## POĎAKOVANIE

Rád by som sa poďakoval vedúcej diplomovej práce pani Ing. Anne Kubánkovej, Ph.D. za odborné vedenie, konzultácie, trpezlivosť a podnetné návrhy k práci. Tiež by som chcel poďakovať všetkým mojim blízkym, ktorí mi aj v čase pandémie vírusu Covid-19 poskytli pokojné a idylické prostredie na tvorbu tejto diplomovej práce.

# Obsah

Úvod	9
<b>1 Teoretická časť študentskej práce</b>	<b>10</b>
1.1 IEEE 802.11 a Adaptačné algoritmy(RAA)	10
1.1.1 Štandard IEEE 802.11	10
1.1.2 Adaptácia rýchlosti RAA	12
1.1.3 Adaptácia rýchlosti v štandarde 802.11	14
1.2 Network Simulator NS-3	18
1.2.1 Základné prvky v NS-3	18
1.2.2 Základné súčasti programu NS-3	19
1.3 OMNeT++	20
1.3.1 Štruktúra	21
1.3.2 Grafický editor	21
1.3.3 Vnútoraná architektúra	21
1.4 Meranie rýchlosti bezdrôtovej siete	22
1.4.1 Metódy merania parametrov bezdrôtovej siete	23
1.4.2 Program iPerf3	23
1.4.3 Program TamoSoft Throughput Test	24
1.4.4 Cisco Linksys WRT610n v2	24
<b>2 Výsledky študentskej práce</b>	<b>25</b>
2.1 Simulácia v prostredí NS-3	25
2.1.1 Simulačný scenár	25
2.1.2 Inštalácia simulačného prostredia NS-3	25
2.1.3 Simulačný skript NS-3	27
2.1.4 Parametre a súčasti ovplyvňujúce výsledky simulácie.	31
2.1.5 Výsledky simulácie	35
2.2 Simulácia v prostredí OMNeT++	39
2.2.1 Simulačný scenár	39
2.2.2 Inštalácia prostredia OMNeT++ v operačnom systéme Windows 10	39
2.2.3 Vytváranie simulácií v OMNeT++	40
2.2.4 Modifikácia scenára Rate Controll Showcase	44
2.2.5 Parametre ovplyvňujúce výsledky simulácie	46
2.2.6 Výsledky simulácie	53
2.2.7 Vyhodnotenie simulácie	56
2.3 Meranie skutočných parametrov	56

2.3.1	Prvé meranie . . . . .	56
2.3.2	Druhé meranie . . . . .	58
2.3.3	Výsledky meraní . . . . .	60
2.4	Vyhodnotenie výsledkov . . . . .	67
2.4.1	Posúdenie simulácií v NS-3 . . . . .	67
2.4.2	Posúdenie simulácií v OMNeT++ . . . . .	68
2.4.3	Posúdenie merania . . . . .	69
2.4.4	Porovnanie výsledkov merania a simulácie . . . . .	71
2.4.5	Porovnanie simulačných prostredí NS-3 a OMNeT++ . . . . .	75
	<b>Záver</b>	<b>77</b>
	<b>Literatúra</b>	<b>78</b>
	<b>Zoznam symbolov, veličín a skratiek</b>	<b>80</b>
	<b>Zoznam príloh</b>	<b>81</b>
	<b>A Príloha CD</b>	<b>82</b>



# Zoznam obrázkov

1.1	Použitie bezdrôtovej siete. . . . .	11
1.2	Štruktúra NS-3. . . . .	18
1.3	Štruktúra OMNeT++.[17] . . . . .	22
2.1	Porovnanie referenčných strát pre model LogDistancePropagation-LossModel. . . . .	32
2.2	Porovnanie adaptačných algoritmov pre štandard 802.11g. . . . .	34
2.3	Výsledok simulácie pre štandard 802.11b. . . . .	36
2.4	Výsledok simulácie pre štandard 802.11g. . . . .	37
2.5	Výsledok simulácie pre štandard 802.11n. . . . .	38
2.6	Porovnanie vplyvu hodnôt SystemLoss v OMNeT++ pri štandarde 802.11g. . . . .	48
2.7	Porovnanie adaptačných algoritmov v OMNeT++. . . . .	51
2.8	Výsledok simulácie pre štandard 802.11b. . . . .	54
2.9	Výsledok simulácie pre štandard 802.11g. . . . .	55
2.10	Zapojenie pracoviska. . . . .	57
2.11	Zapojenie pracoviska. . . . .	59
2.12	Meranie štandardu 802.11b. . . . .	61
2.13	Meranie štandardu 802.11n. . . . .	62
2.14	Druhé meranie štandardu 802.11b. . . . .	64
2.15	Druhé meranie štandardu 802.11g. . . . .	65
2.16	Druhé meranie štandardu 802.11n. . . . .	66
2.17	Porovnanie meraní pre štandard 802.11n. . . . .	70
2.18	Porovnanie výsledkov simulácií a druhého merania. . . . .	72
2.19	Porovnanie pre štandard 802.11b. . . . .	73
2.20	Porovnanie pre štandard 802.11n. . . . .	74

# Úvod

Vzhľadom na čoraz väčší záujem o problematiku bezdrôtových sietí, je možné sledovať snahy o implementáciu algoritmov, ktorých cieľom je zvýšiť efektívnosť existujúcich bezdrôtových sietí. Adaptačné algoritmy možno chápať ako mechanizmy, ktorých cieľom je reagovať na zmeny v sieti. Táto práca sa venuje problematike algoritmov adaptácie rýchlosti RAA (Rate Adaptation Algorithm) v bezdrôtových sieťach definovaných v štandarde IEEE 802.11. Práca zahŕňa teoretickú analýzu bezdrôtových sietí a popis algoritmov určených na adaptáciu rýchlosti. Teoretická časť tiež obsahuje popis simulačných prostredí NS-3, OMNeT++ a nástrojov na získavanie hodnôt prenosovej rýchlosti v lokálnych sieťach.

Praktická časť práce následne popisuje inštaláciu simulačných prostredí NS-3 a OMNeT++, pomocou ktorých sú vytvárané simulačné scenáre určené na analýzu adaptačných algoritmov RAA. Tvorba simulačných scenárov a moduly použité v simuláciách sú popísané z pohľadu ich parametrov a vplyvov na výsledky simulácií. Výsledky simulácií boli následne porovnané s hodnotami získanými z dvoch meraní s prídeleným smerovačom Cisco Linksys WRT610n v2. Získané závislosti prenosových rýchlostí a vzdialenosti od prístupového bodu boli analyzované. Výsledkom práce je porovnanie simulačných prostredí a analýza adaptačných algoritmov RAA.

# 1 Teoretická časť študentskej práce

Táto časť práce obsahuje charakteristiku použitých technických riešení a teoretické spracovanie problematiky. Obsah je chronologicky delený do sekcií, ktoré pojednávajú o teoretických poznatkoch, ktoré boli použité pri riešení zadania práce. Obsah sekcií je určený predovšetkým na objasnenie problematiky realizácie simulácií, ktoré demonštrujú funkčnosť adaptačných algoritmov v sieťach IEEE 802.11.

## 1.1 IEEE 802.11 a Adaptačné algoritmy(RAA)

Z dôvodu neustáleho nárastu požiadaviek na bezdrôtové pripojenie je v dnešnej dobe zaznamenateľný masívny nárast záujmu o nové technologické riešenia v oblasti bezdrôtového pripojenia do sietí. Vzhľadom na narastajúcu potrebu zabezpečenia vysokej rýchlosti pripojenia sú v súčasnosti využívané techniky zabezpečujúce adaptáciu parametrov siete na efektívne využitie prenosového kanálu. V nasledujúcej sekcií sa nachádza orientačný popis parametrov niektorých štandardov, ktoré definujú správanie sietí v závislosti na meniacich sa parametroch prenosu. Základným prostriedkom pre efektívne riadenie bezdrôtovej komunikácie je ustanovenie ideálnej prenosovej rýchlosti, ktorá umožní spoľahlivý prenos dát pomocou bezdrôtovej siete, definovanej v štandarde IEEE 802.11.

### 1.1.1 Štandard IEEE 802.11

Technológia bezdrôtového pripojenia WLAN (Wireless Local Area Network) sa stáva realizovateľnou alternatívou k tradičným káblovým pripojeniam. Dnešné bezdrôtové siete nevyžadujú žiadne fyzické prepojenie medzi koncovým užívateľom a serverom [14]. Štandard IEEE 802.11 pozostáva zo série technologických ustanovení, ktoré boli vyvíjané v priebehu mnohých rokov. Každý nový technologický postup je definovaný zmenou a doplnením pôvodného štandardu 802.11. Štandard IEEE 802.11 sa zameriava predovšetkým na fyzickú vrstvu. V roku 1997 bol publikovaný prvý štandard IEEE 802.11. Na fyzickej (najnižšej) vrstve poskytol nové riešenia, ktoré umožňujú spoľahlivý prenos dát bezdrôtovým médiami. Novými využívanými funkciami je frekvenčný skok FHSS (frequency hopping) a spektrum s priamym frekvenčným rozptylom, ktoré zabezpečujú parametre vhodné pre spoľahlivý prenos dát. Štandard sa zameriaval na bezdrôtovú komunikáciu v nelicencovanom pásme 2,4GHz a na komunikáciu založenú na infračervenom žiarení v pásme 316 - 353THz. Pôvodný IEEE 802.11 tiež zahŕňal riešenia pre viacnásobný prístup k prenosovému médiu (CSMA/CA) bez detekcie kolízie. [14] [4]



Obr. 1.1: Použitie bezdrôtovej siete.

### **Pôvodný štandard 802.11**

Umožňuje pripojenie rýchlosťou 2Mb/s v pásme 2,4GHz. Išlo o základnú definíciu bezdrôtového štandardu pre WLAN (Wireless Local Area Network) siete.

### **Štandard 802.11b**

Je najčastejšie označovaný ako Wi-Fi. Praktická rýchlosť býva cca 6,5Mb/s a dosah približne 30m v uzatvorenom priestore. Tento štandard využíva pásma 2,4GHz (13 kanálov po 22MHz).

### **Štandard 802.11a**

Tento štandard umožňuje prenosovú rýchlosť až 54Mb/s. Je pridaná podpora prenosu v pásme 5GHz. Používa sa modulácia OFDM (Orthogonal Frequency-Division Multiplexing) tj. ortogonálny multiplex s kmitočtovým delením.

### **Štandard 802.11g**

Preberá modulačné charakteristiky z 802.11a. Maximálna prenosová rýchlosť je 24Mb/s.

## **Štandard 802.11n**

Štandard vychádzajúci z predošlej definície 802.11 využíva pásmo 2,4GHz aj 5GHz. Vďaka technológií MIMO (multiple input, multiple output) je zariadenie schopné využívať viac komunikačných tokov pre jedno spojenie. Táto technológia umožňuje zdvojnásobiť dosah zariadenia. Prenosová rýchlosť sa vďaka technológii MIMO zvýšila na hodnotu cez 100Mb/s. Maximálna prenosová rýchlosť je 600Mb/s v 4 dátových tokoch.

## **Štandard 802.11ac**

Je založený na rovnakom princípe ako 802.11n. Ponúka však maximálnu rýchlosť až okolo 1300Mb/s.

### **1.1.2 Adaptácia rýchlosti RAA**

Adaptácia rýchlosti je určenie optimálnej a najvhodnejšej prenosovej rýchlosti pre spoľahlivý prenos dát v súčasných bezdrôtových sieťach. V poslednom období bola problematika miery adaptácie značne skúmaným atribútom bezdrôtových sietí IEEE 802.11. Táto problematika zahŕňa otázky diferenciácie strát a tiež metódy snímania podmienok prenosového kanálu. Cieľom algoritmov RAA je efektívne využitie fyzického kanálu. Bežne však tieto algoritmy nezohľadňujú straty rámcov spôsobené kolíziou. Základným atribútom na účinné sledovanie parametrov siete je chybovosť rámcov. Na základe získaných dát je možné vyhodnocovať účinnosť použitých parametrov siete, ktoré vplývajú na kvalitu prenosu. Pri predpoklade voľného pohybu klienta v priestore pokrytom bezdrôtovou sieťou je možné očakávať neustále zmeny parametrov chybovosti prenosu dát. Na základe získaných dát sa pomocou adaptačných algoritmov zabezpečuje prispôbenie prenosovej rýchlosti k parametrom siete.

### **Charakteristika**

Vo všeobecnosti je možné adaptáciu rýchlosti považovať za prostriedok, ktorý odhaduje parametre prenosového kanálu. Odhad stavu prenosového kanálu je väčšinou založený na sledovaní parametrov odstupu signálu od šumu a chybovosti prenosu rámcov. Samotné sledovanie chybovosti je založené na pozorovaní počtu úspešne prenesených rámcov, prípadne skúmanie úspešnosti techniky kľzavého okna.

Techniky adaptácie rýchlosti prenosu predpokladajú, že chybovosť prenosu je spôsobená útlmom signálu a kolíziou kanálov v sieti.

## **Adaptácia rýchlosti založená na chybovosti prenosu rámcov**

**Auto Rate Fallback (ARF)** od Kamermana a Montebana je pôvodná schéma určená pre štandard 802.11. Išlo o návrh technického riešenia pre Lucent Wave-II LAN sieťový adaptér. Princíp adaptácie je založený na jednoduchom algoritme. Odosielať zaháji prenos na najnižšej možnej rýchlosti (2 Mbs v 802.11b) a spustí časovač. Ak v časovom okne prebehne prenos úspešne, alebo odosielať úspešne odošle  $N$  (konštantné číslo) po sebe idúcich rámcov, odosielať obnoví spojenie na novej vyššej prenosovej rýchlosti. Ak dochádza k zlyhaniu komunikácie, odosielať rýchlosť upravuje na nižšiu možnú. Pri každej zmene rýchlosti dochádza k nulovaniu časovača.

**Adaptive Auto Rate Fallback (AARF)** od Lacage a kol. bol navrhnutý na zvýšenie efektivity v stabilných prostrediach. Na rozdiel od ARF táto technika prispôsobuje prahovú hodnotu  $N$ . V prípade úspešného prenosu sa prahová hodnota  $N$  zdvojnásobí ( $2N$ ). Ak prenos zlyhá na novej hodnote  $2N$ , zavedie sa pôvodná hodnota  $N$ . AARF umožňuje zvýšenie intervalov medzi zmenami rýchlosti, čo vedie k väčšej stabilite prenosu.

Medzi prvé implementácie adaptácie rýchlosti v systémoch Linux patrí **Onoe** od spoločnosti MadWifi. Cieľom Onoe je nájsť vhodnú rýchlosť prenosu, pomocou ktorej je možné komunikovať so stratovým pomerom menším ako 50%.

**Sample Rate** je algoritmus adaptácie rýchlosti od spoločnosti Bicket. Na základe štatistík prenosu pomocou techniky kľavého okna Sample Rate prispôsobuje rýchlosť prenosu. Táto technika je založená na sledovaní času potrebného na úspešný prenos informácie (do potvrdenia príjemcom). Časový údaj tak zahŕňa aj čas určený na opakovaný prenos rámcov pri technike kľavého okna. Sample Rate zahajuje komunikáciu na najvyššej možnej rýchlosti. Po zaznamenaní štyroch neúspešných sekvencií prenosu je nastavená nižšia možná rýchlosť. Sample Rate tiež zavádza hodnotu priemerného času potrebného na prenos jedného rámca. Táto hodnota je upravovaná dynamicky raz za 10 sekúnd.[1]

## **Adaptácia rýchlosti založená na odstupe signálu od šumu**

**Receive Based Auto Rate (RBAR)** je adaptácia rýchlosti založená na príjemcovi. Spoločnosť Holland, Vaida a Bahl vyvinula prvý algoritmus adaptácie rýchlosti, ktorý využíva kontrolné rámce RTS/CTS vysielané základnou rýchlosťou. RBAR upravuje normu 802.11 z dvoch hľadísk. Informácia o rezervácii kanálu sa nachádza v hlavičke rámca RTS/CTS. Táto informácia je reprezentovaná dátami o veľkosti paketu a rýchlosti prenosu. Druhou úpravou normy 802.11 je prenášanie správ RSH, ktoré predchádzajú samotnému prenosu dát. Proces nadviazania komunikácie zahŕňa nasledovný algoritmus:

Odosielateľ heuristicky zvolí rýchlosť prenosu (napr. posledná úspešne použitá rýchlosť). Tieto informácie zahrnie do RTS rámcu. Prijemca obdrží RTS rámec. Na základe dát z fyzickej vrstvy príjemca odosiela informácie o odstupe signálu od šumu konvertované na rýchlosť, ktorá môže byť z pohľadu odosielateľa použitá. Následne do CTS rámcu príjemca zahrnie informácie o veľkosti paketu a o použitej rýchlosti. Následne je CTS rámec rozoslaný na stanice. Všetky stanice, ktoré obdržia CTS rámec sú tak schopné evidovať informácie o rezervácii prenosového kanálu.[1]

**Opportunistic Auto Rate (OAR)** reaguje na anomálie v sieťach s viacerými rýchlosťami. Heusse a kol. pozorovali anomáliu v sieťach, ktoré zabezpečujú komunikáciu s viacerými stanicami na rôznych rýchlostiach. Všetky stanice dosahujú priepustnosť približne 30% napriek rôznym sadzbám rýchlosti. Tento jav je spôsobený rovnosťou pravdepodobnosti prístupu k prenosovému médiu v sieti CS-MA/CA. OAR sa snaží tento problém riešiť pomocou rámcov RTS/CTS. Technika využíva možnosť fragmentácie rámcov podľa štandardu 802.11. Cieľom tohto procesu je viacnásobné zasielanie RTS/CTS rámcov v závislosti od prenosovej rýchlosti. Týmto mechanizmom je možné poskytnúť časovú spravodlivosť pomocou umelého udržiavania prenosu dát do úplného zaslania fragmentovaných dát.

**Full Auto Rate (FAR)** zahŕňa vyššie spomínané mechanizmy RBAR. V FAR je však zahrnutý mechanizmus automatického odhadu rýchlosti. Nečinný prijímač zaznamenáva rámce okolitých staníc. Prijímač tieto dáta využije k automatickému odhadu rýchlosti. Dáta o zvolených parametroch následne zasiela pomocou RTS rámcov.

**Loss-differentiating-ARF (LD-ARF)** je technika adaptácie, ktorá zahŕňa aj diferencovanie strát. Straty pri komunikácii môžu mať rôznu príčinu. Táto technika je nadstavbou na už spomínané ARF. LD-ARF sleduje stav hlavičky. Ak je hlavička neporušená a telo správy je chybné, možno predpokladať chybu spôsobenú útlmom kanála. V prípade porušenia celej správy sa dá predpokladať kolízia správ v prenosovom kanáli.

Detekcia a analýza kolízie je zahrnutá aj v ďalších technikách adaptácie rýchlosti. Medzi ne patrí napríklad: **Collision-Aware Rate Adaptation (CARA)**, **Robust Rate Adaptation Algorithm (RRAA)**, **Loss Differentiation Rate Adaptation (LDRA)**)[1]

### 1.1.3 Adaptácia rýchlosti v štandarde 802.11

Pri vytváraní adaptačných algoritmov implementovaných v štandarde 802.11 boli navrhnuté mechanizmy, ktoré vychádzajú z vyššie spomenutých algoritmov adaptácie rýchlosti. Podobne ako pri všeobecnom pohľade na samotnú adaptáciu rýchlosti, aj v prípade štandardizácie sa kládol dôraz na prispôbenie parametrov bez-drôtovej

komunikácie v sieti. Hlavným cieľom je zefektívniť samotnú prevádzku v sieti. Pri implementovaní funkčného modelu adaptácie rýchlosti bolo nutné klásť dôraz na funkčnosť samotnej fyzickej vrstvy a tiež zabezpečiť podmienky pre linkovú vrstvu. Výrobcovia sieťových zariadení väčšinou ponúkajú vlastné adaptačné algoritmy, ktoré sú schopné zabezpečiť kompatibilitu zariadenia v rámci daného štandardu a zároveň zlepšujú parametre prenosu dát v bezdrôtovej sieti. Tento prístup však komplikuje podrobnú analýzu použitých adaptačných algoritmov v rámci štandardov 802.11. Neopomenuteľným faktom je, že väčšina autorov adaptačných algoritmov zdôrazňuje skutočnosť, že štandard 802.11 bližšie nešpecifikuje parametre adaptačných algoritmov. V nasledujúcej sekcii sú popísané základné známe mechanizmy, ktoré sú implementované pri adaptácii rýchlosti v štandardoch 802.11.

### **Adaptácia rýchlosti Minstrel**

Minstrel je adaptačný algoritmus, ktorý bol v roku 2006 vyvíjaný ako reakcia na postupné zvyšovanie prenosových rýchlostí v bezdrôtových sieťach. Podstatná časť vývoja bola zameraná na riešenie nedostatkov spojených s protokolom TCP. Existujúce algoritmy negatívne vplývali na komunikáciu TCP a svojou činnosťou zhoršovali bezproblémový chod siete. V nasledujúcich rokoch bol vyvíjaný adaptačný algoritmus Minstrel, ktorý sa snaží eliminovať problémy spôsobené inými algoritmami. Postupne došlo k implementácii algoritmu Minstrel do jadier distribúcií operačného systému Linux. Adaptačný algoritmus Minstrel sleduje okrem sily signálu aj odstup signálu od šumu. Cieľom vývojárov bolo využiť všetky použiteľné informácie, ktoré odzrkadľujú aktuálnu kondíciu siete. Adaptačný algoritmus Minstrel reaguje na fakt, že väčšina komunikačných protokolov pri detekcii zvýšenia strát zahľucuje komunikačný kanál správami, ktoré obmedzujú zlepšenie fungovania samotného prenosu. Minstrel sa tak snaží predchádzať prehnane vysokým sadzbám rýchlosti, ktoré zvyšujú chybovosť a straty rámcov.[6]

### **Robust Rate Adaptation Algorithm (RRAA) v 802.11a/b**

V prípade štandardov 802.11a a 802.11b bol kladený dôraz predovšetkým na komplexné sledovanie parametrov siete. Model adaptácie bol navrhnutý tak, aby dokázal reagovať na náhodné, nešpecifikované zhoršenie parametrov prenosu. Model dokáže reagovať aj na rýchle skokové zmeny, spôsobené pohybom užívateľov v priestore.

Samotná spoľahlivosť modelu je zabezpečená malou veľkosťou časového okna. V prípade štandardu 802.11b dochádza k využitiu RTS/CTS rámcov pre zefektívnenie chodu systému. RRAA pozostáva z dvoch prvkov: adaptácia rýchlosti a detekcia kolízií. Základným sledovaným parametrom v sieti však zostáva pomer (P) medzi počtom neúspešne zaslaných rámcov a súčtom celkového počtu rámcov. Stanica začne



vysielat maximálnou možnou rýchlosťou. Veľkosť časového okna sa môže počas činnosti dynamicky meniť. RRAA tiež zavádza dve prahové hodnoty: PMTL a PORI. Ak je nameraný pomer (P) väčší ako hodnota PMTL, pre nasledujúci prenos je zvolená nižšia možná rýchlosť. Ak je hodnota pomeru (P) menšia ako hodnota PORI, rýchlosť sa zvýši. V prípade, že je pomer (P) väčší ako PORI a zároveň menší ako PMTL, rýchlosť zostane nezmenená. Súčasťou RRAA je tiež mechanizmus pre kolíziu detekcií A-RTS, zabezpečujúci riadenie parametrov prenosu v závislosti na zistených kolíziách. A-RTS je schopné riadiť parameter počtu po sebe idúcich paketov jednej komunikačnej sekvencie. Táto hodnota sa upravuje v závislosti na úspešnosti predošlej komunikácie. Model robustnej adaptácie rýchlosti je schopný rýchlejšie reagovať na skokové zmeny v sieti a v porovnaní s bežnými modelmi zabezpečuje vyššiu prenosovú rýchlosť.[5]

### **MIMO Rate Adaptation v 802.11n**

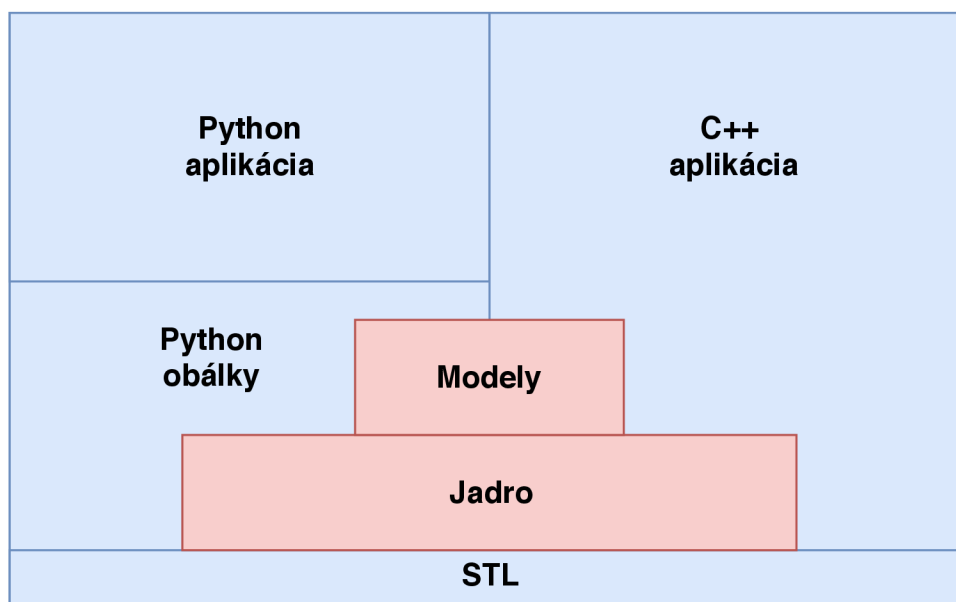
Štandard 802.11n je pomerne nová generácia bezdrôtovej siete LAN. Tento štandard využíva vyššiu rýchlosť prenosu dát a tiež zabezpečuje vyšší dosah. MIMO je kľúčový mechanizmus implementovaný na fyzickej vrstve sieťového modelu. Mechanizmus MIMO pracuje v dvoch režimoch. Priestorová diverzita umožňuje vysielat jeden dátový tok pomocou jednej antény. Použitím väčšieho počtu antén dochádza k zvyšovaniu spoľahlivosti prenosu. Riadenie priestorového multiplexovania dátového toku pri väčšom počte antén je však výpočtovo náročné a často nie je podporované sieťovými kartami zariadení v sieti. Mechanizmus MIMO je schopný v prípade potreby umožniť aj štandardnú komunikáciu pomocou jednej antény a jedného dátového toku.[13] Druhým režimom, pomocou ktorého MIMO zabezpečuje zlepšenie parametrov v bezdrôtovej sieti, je samotná adaptácia rýchlosti známa z predošlých štandardov. Mechanizmus adaptácie však v tomto prípade musí zohľadňovať fakt, že na prenos môže byť použitý väčší počet antén. MIMO v mechanizme adaptácie implementuje aj použitie sledovania odstup signálu od šumu. V porovnaní so štandardmi 802.11a a 802.11b je nutné pri úprave rýchlosti zohľadniť možnosť prepínania do režimu nepodporujúceho vysielanie vo viacerých tokoch. Samotná problematika adaptácie rýchlosti v štandardoch 802.11n je závislá predovšetkým na použitej základnej schéme. Väčšina výrobcov sieťových zariadení disponuje vlastným algoritmom pre adaptáciu rýchlosti v štandarde 802.11n. V prípade zariadení Cisco je použitý algoritmus DRA (Dynamic Rate Adaptation), ktorý definuje spoločné parametre, ktoré algoritmus sleduje pri vyhodnocovaní potrieb zmeny prenosovej rýchlosti v bezdrôtovej sieti. Základným sledovaným parametrom je odstup signálu od šumu (SNR). Zariadenia pracujú s hraničnými hodnotami odstup signálu

od šumu, ktoré určujú prepnutie do inej rýchlosti. Tieto hodnoty sú závislé na použítom zariadení. Jedným z ďalších algoritmov pre adaptáciu rýchlosti v štandarde 802.11n je MiRA. Pri analýze fungovania adaptačných algoritmov v sieťach podporujúcich MIMO je možné sledovať časté prepínanie módov komunikácie vo viacerých tokoch a komunikáciu zabezpečenú len jednou anténou. Skoková zmena rýchlosti je často nazývaná zigzag. Tento jav je charakterizovaný náhlou zmenou rýchlosti, ktorá je však po zavedení adaptačného algoritmu pre daný mód upravená na konštantnú hodnotu. Tento stav je nutné zohľadňovať pri implementácii adaptačných algoritmov v sieťach 802.11n. Algoritmus MiRA je založený na sledovaní chybovosti prenesených rámcov. Implementuje algoritmus RRAA. Samotný proces adaptácie rýchlosti je zavedený v prípade, že súčasný mód MIMO nie je schopný zabezpečiť spoľahlivý prenos dát. Adaptácia rýchlosti je tak uprednostnená pred procesom prepnutia módu MIMO. Algoritmus MiRA eliminuje prípadné časté výpadky spôsobené zmenou módu MIMO. Vo všeobecnosti je tak možné považovať samotnú problematiku značne závislú na samotnom výrobcovi bezdrôtového zariadenia. Väčšina mechanizmov však vychádza z bežných algoritmov použiteľných aj pri 802.11b a 802.11a, ktoré sú upravené pre použitie v kombinácii s technológiou MIMO.[11][2][5][3]

## 1.2 Network Simulator NS-3

NS-3 je sieťový simulátor diskretných udalostí určený na výskumné a vzdelávacie účely. NS-3 je software so slobodnou licenciou GNU GPLv2.[12]

Cieľom projektu NS-3 je vyvinúť simulačné prostredie vhodné pre výskum v oblasti sietí. Projekt sa zaväzuje vybudovať robustné simulačné jadro, ktoré je možné zdokumentovať a ladiť. Jadro musí vyhovovať pracovným potrebám užívateľov a zabezpečiť postupy pre analýzu a dokumentovanie simulačných dát. Softvérová infraštruktúra NS-3 tiež podporuje vývoj simulačných modelov, ktoré sú dostatočne realistické na emulovanie siete v reálnom čase. NS-3 je možné prepojiť so skutočnou sieťou. Simulačné jadro podporuje vytváranie scenárov založených na IP protokole. Väčšina používateľov sa zameriava na simuláciu bezdrôtových sietí. NS-3 tiež podporuje plánovač v reálnom čase, ktorý uľahčuje možnosti simulácie v slučke. Ďalšou vlastnosťou NS-3 je možnosť opätovného spustenia aplikácie. Vytváranie simulácie je možné pomocou vytvárania aplikácií v programovacom jazyku C alebo C++. [9]



Obr. 1.2: Štruktúra NS-3.

### 1.2.1 Základné prvky v NS-3

Pri vytváraní simulácií v prostredí NS-3 je nutné využívať nástroje, ktoré sú súčasťou samotného simulačného prostredia. NS-3 obsahuje veľké množstvo modulov a nástrojov, ktoré slúžia na zabezpečenie čo najväčšej škálovateľnosti. Fungovanie NS-3 je možné zjednodušené popísať ako softvér bežiaci na počítači, ktorý v reálnom čase simuluje deje v simulovanej sieti. Základnou abstrakciou užívateľského

programu je teda aplikácia, ktorá generuje simulovanú aktivitu. Z pohľadu programovacieho jazyka C++ sa jedná predovšetkým o súbor tried a metód, pomocou ktorých je možné vytvárať objekty [9] zabezpečujúce abstrakciu a následný prístup ku konfiguračným parametrom pomocou metód.

### **Komunikačný kanál (Channel)**

NS-3 zabezpečuje prepojenie objektov pomocou konektivity a objektu kanál. Ak je nutné realizovať pripojenie uzlu k sieti, je použitý objekt kanál. Túto abstrakciu reprezentuje v jazyku C++ objekt Channel. Trieda Channel obsahuje metódy zabezpečujúce konfiguráciu samotnej podsiete. Vzhľadom na podporu objektovo orientovaného programovania je vývojom umožnené vytvárať špecifické kanály ako napríklad PointToPointChannel a WifiChannel.[9]

### **Sieťové zariadenie (Net Device)**

Simulačné prostredie zabezpečuje abstrakciu sieťového zariadenia pomocou triedy Net Device. Táto trieda zastrešuje aplikačné a hardvérové vlastnosti konkrétneho zariadenia v sieti. Podobne ako pri triede Channel existuje niekoľko špecializovaných tried Net Device ako napríklad CsmaNetDevice alebo WifiNetDevice.[9]

### **Pomocník topológie (Topology Helper)**

Pri vytváraní reálnej siete dochádza k pomerne zdĺhavej konfigurácii sieťových zariadení, ktoré sú na základe IP adries, MAC adries a podpory protokolov schopné komunikovať v sieti. Tento proces konfigurácie sieťových zariadení je pri väčšine simulačných scenárov značne zdĺhavý a aj pri využití algoritmickej, plynúcej zo samotného programovacieho jazyka C++, dochádza k značnej redundancii. NS-3 túto neefektivitu odstraňuje pomocou sady pomocníkov. Ide o nástroje, ktoré sú schopné zabezpečiť automatizovaný proces konfigurácie sieťových prvkov a samotné spustenie simulovanej siete do prevádzky.[9]

## **1.2.2 Základné súčasti programu NS-3**

Pri vytváraní simulačných skriptov v NS-3 je možné sledovať značnú podobnosť s vytváraním programov v jazyku C++. Existujúca platforma NS-3 kopíruje bežnú štruktúru aplikácie v jazyku C++.[9]

## Informácie o kódovaní a licencií

Súčasťou každého štandardizovaného skriptu v NS-3 je úvodná hlavička obsahujúca informácie o použítom kódovaní a licencii GNU (General Public License).[12] Simulátor NS-3 je licencovaný na základe GNU. V záhlaví každého súboru je možné vidieť právnych zástupcov licencie a text informujúci o autorských právach.

## Zahrnutie modulov (Include)

V záhlaví každého skriptu sa nachádza pole, obsahujúce zahrnutie potrebných hlavičkových súborov pre použitie príslušných modulov. NS-3 je navrhnuté tak, aby umožňovalo modularitu samotného simulačného prostredia. Zahrnutie potrebných hlavičkových súborov zabezpečuje možnosť použitia príslušného modulu.[9]

## Menný priestor NS3

Celý projekt NS-3 je implementovaný v jednom mennom priestore. Táto vlastnosť umožňuje združovať všetky deklarácie v rámci jedného menného priestoru.

## Funkcia main

Podobne ako pri bežnom programe v C++, tak aj v prípade NS-3 je telo samotnej simulácie vytvorené vo funkcii main. Štandardne ide o priestor obsahujúci nastavenie časového rozlíšenia a samotné nasadenie prvkov v simulácii. Postupným pridávaním vyššie spomenutých tried vzniká samotný simulačný scenár, ktorý je možné pomocou spustenia simulácie zahájiť.[9]

## 1.3 OMNeT++

OMNeT++ (Objective Modular Network Testbed in C++) je objektovo-orientovaný modulárny systém určený pre vytváranie simulácií diskretných udalostí v sieti. Tento simulačný nástroj sa využíva na vytváranie simulácií káblových a bezdrôtových sietí. Taktiež sa využíva na modelovanie multiprocessorov a iných distribuovaných systémov. Je často využívaný na posudzovanie topológií a architektúr. OMNeT++ je šíriteľný na základe verejnej licencie a je určený na akademické účely. Cieľom vývoja OMNeT++ bolo vytvoriť nástroj, ktorý nahrádza licencované komerčné produkty. Softvér je prístupný na všetkých bežných platformách ako Linux, MacOS/X a Windows. Vzhľadom na modulárnu koncepciu simulačného prostredia je možné využívať rôzne moduly reprezentované rámcami, ktoré sú vyvinuté nezávisle od samotného simulačného prostredia.[17] [16]

### 1.3.1 Štruktúra

Sieťový simulátor OMNeT++ pozostáva z modulov, ktoré majú medzi sebou zabezpečenú výmenu informácií. Samotné moduly môžu byť združované do zložených modulov. Tieto prvky sú vytvorené v programovacom jazyku C++, využitím simulačných tried a knižníc. Počet úrovní hierarchie nie je obmedzený. Modulárna hierarchia umožňuje užívateľovi vytvárať sieťový model, ktorý v sebe zahŕňa všetky potrebné parametre a sieťové prvky, ktoré sú reprezentované modulmi. Komunikácia medzi modulmi je zabezpečená pomocou správ, ktoré obsahujú bežné sieťové atribúty, ako časová pečiatka a voliteľné dáta. Z dôvodu zabezpečenia opakovateľnosti realizácie simulácií je komunikácia povolená len prostredníctvom brán. Týmto opatrením sa zabezpečuje plne funkčná modulárna hierarchia samotnej simulácie. Parametre komunikácie medzi modulmi je možné definovať pomocou komunikačného kanálu, ktorý združuje všetky atribúty súvisiace s daným spojom. Vzhľadom na potrebu zabezpečiť komplexné riešenie definície samotných modulov a ich parametrov, bol vytvorený jazyk NED (network description), ktorý umožňuje škálovať parametre sieťových modulov.[16]

### 1.3.2 Grafický editor

Súčasťou OMNeT++ je aj integrované vývojové prostredie, ktoré obsahuje grafický editor. Tento nástroj využíva jazyk NED ako svoj natívny formát zápisu. Samotný grafický editor dokáže využívať aj ručne písané zdrojové kódy v jazyku NED. Užívateľ je tak schopný editovať kód pomocou grafického editoru, alebo priamo prepisovať zdrojový kód. Grafický editor umožňuje okrem vytvárania statických sieťových topológií využívať aj deklaratívne konštrukcie, ktoré pripomínajú slučku. Takéto riešenie umožňuje náhodne, alebo v istej sekvencii meniť parametre konfigurácie.[17]

### 1.3.3 Vnútoraná architektúra

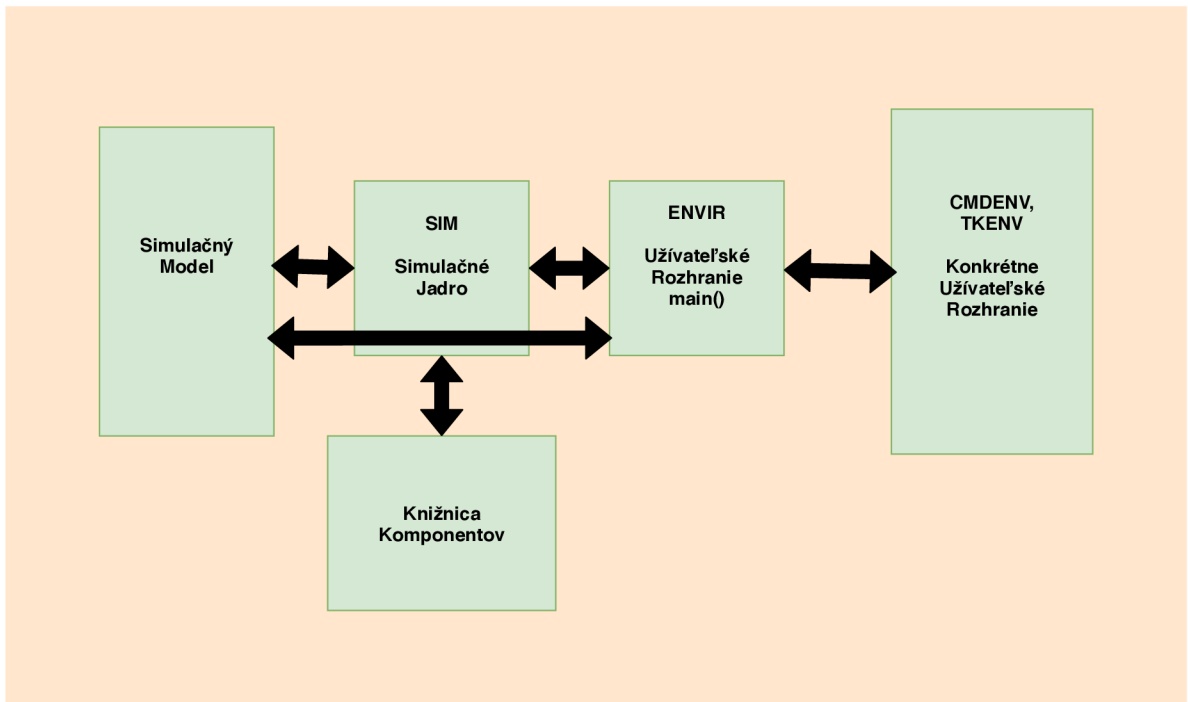
Ako už bolo spomenuté, simulačné programy OMNeT++ majú modulárnu štruktúru. Samotná štruktúra simulačného prostredia pozostáva z piatich hlavných častí.[17][16]

#### **Knižnica komponentov**

Pozostáva z kompilovaného kódu, ktorý obsahuje jednoduché a zložené moduly.

#### **Simulačné jadro**

Jadro obsahuje potrebné moduly a inštanciu triedy knižnice Sim. V simulačnom jadre sa vykonávajú všetky simulácie.



Obr. 1.3: Štruktúra OMNeT++.[17]

### Užívateľské rozhranie a konkrétne užívateľské rozhranie

Do jadra vnášajú informácie o pôvode modulov a dát. Zabezpečujú spracovanie vstupov a výstupov simulácie.

### Simulačný model

Znázorňuje vytvorenú topológiu. Väčšinou sa jedná o vizualizovaný model siete.

## 1.4 Meranie rýchlosti bezdrôtovej siete

Pre prevádzkovateľa siete je dôležitým ukazovateľom funkčnosti danej topológie realizácia merania parametrov siete. Pri zavádzaní nových prvkov a aplikácií v sieti dochádza k zmenám, ktoré môžu spôsobiť zhoršenie parametrov siete. Pri zisťovaní parametrov siete je možné sledovať rôzne parametre súvisiace priamo s fyzickou vrstvou sieťového modelu. V súvislosti s rýchlosťou prenášaných paketov je nutné uvažovať reálne využiteľnú šírku pásma ponúknutú na prenos informácií na aplikačnej vrstve. Pri realizácii meraní je pomerne náročné brať do úvahy len samotnú aplikačnú časť celej sieťovej hierarchie. Oveľa častejšie sa pri meraní parametrov lokálnych sietí zavádza pojem priepustnosť siete. Táto veličina je reprezentovaná počtom prenesených bitov za sekundu. Priepustnosť meranej siete priamo súvisí

so šírkou pásma a umožňuje objektívne a opakovateľné prevedenie meraní. Parameter priepustnosti siete priamo súvisí nielen s prenosovou rýchlosťou, ale aj s počtom uzlov a počtom pripojených zariadení.

### **1.4.1 Metódy merania parametrov bezdrôtovej siete**

Na realizáciu meraní parametrov siete je možné vzhliadať rôznymi spôsobmi. Zhromažďovanie údajov o parametroch siete môže byť aktívne a pasívne. V nasledujúcom texte sú v krátkosti popísané obe metódy merania parametrov siete.

#### **Aktívne metódy**

Princíp aktívneho merania parametrov siete je založený na vkladaní testovacích paketov do bezdrôtovej siete. Tieto testovacie pakety sú následne využité na zber dát o chybovosti prevádzky a o samotnej priepustnosti siete. Aktívne merania bezdrôtovej siete umožňujú lepšie charakterizovať kvalitu služieb vnímanú užívateľom. Presnosť nameraných dát je založená na skutočne realizovanom spojení pomocou testovacích paketov v bezdrôtovej sieti. Nevýhodou samotného procesu je fakt, že aktívne meranie vytvára skutočný spoj, čím môže spôsobiť narušenie normálnej prevádzky v bezdrôtovej sieti. Zavádzanie testovacej komunikácie v bezdrôtovej sieti vyžaduje zdieľanie rovnakej šírky pásma ako pri skutočnej prevádzke. [10]

#### **Pasívne metódy**

Pri pasívnych metódach merania bezdrôtovej siete je prevádzka monitorovaná pomocou zachytávania paketov existujúcich služieb. Tieto metódy umožňujú jednoduchý prístup k zisťovaniu parametrov bezdrôtovej siete. Pri realizácii meraní v bezdrôtovej sieti sú pasívne metódy často používané. Väčšina sieťových prvkov má predinštalované nástroje umožňujúce realizáciu pasívneho merania.[10]

### **1.4.2 Program iPerf3**

Pri realizácii merania parametrov bezdrôtovej siete je možné využiť rôzne nástroje, ktoré zabezpečujú aktívnu testovaciu prevádzku v sieti. Väčšina nástrojov je navrhnutá ako komunikačná schéma pozostávajúca zo servera a klienta. iPerf3 je nástroj na meranie maximálne dosiahnuteľnej priepustnosti v IP sieťach. Podporuje ladenie rôznych parametrov týkajúcich sa samotnej komunikácie, protokolov a časovania paketov. Výsledkom testu je zhrnutie parametrov siete. Nástroj iPerf3 je vyvinutý laboratóriom ESnet/Lawrence Berkley. Nástroj bol vydaný na základe licencie BSD.

iPerf3 podporuje využitie testovacej komunikácie pomocou TCP a UDP protokolu. Nástroj obsahuje podporu pre veľkosť TCP okna pomocou vyrovnávacej



pamäte soкетов. V prípade využitia UDP protokolu môže klient vysielat UDP datagramy so špecifikovanými parametrami. Následne je proces zameraný na samotnú analýzu straty paketov a časovú odozvu.

### **1.4.3 Program TamoSoft Throughput Test**

Podobne ako iPerf3, je program TamoSoft Throughput Test určený na aktívne zisťovanie parametrov siete. Tento nástroj je zameraný na meranie priepustnosti bezdrôtových a káblových sietí. Komunikačná schéma pozostáva z aplikácie servera a z aplikácie určenej pre vyhodnotenie dát z pohľadu klienta. Po spustení nástroj generuje TCP a UDP prevádzku s vopred stanovenou hodnotou kvality služieb. Klientský program následne získané dáta graficky aj číselne zobrazuje. Ďalším pomerne užitočným získaným parametrom je aj zobrazená chybovosť komunikácie. Počas realizácie merania je teda možné sledovať aj samotnú úspešnosť prevádzky. Nástroj TamoSoft Throughput Test je kompatibilný so sieťami verzie IPv4 a IPv6. Hlavným cieľom využívania tohto nástroja bolo získanie výsledkov v prehľadnejšej a podrobnejšej forme ako pri meraniach s nástrojom iPerf3.

### **1.4.4 Cisco Linksys WRT610n v2**

Pri realizácii merania rýchlosti bezdrôtovej komunikácie bolo nutné využiť komerčne prístupné zariadenie, použiteľné v konfigurácii prístupového bodu podľa štandardov 802.11. Pre realizáciu merania závislosti rýchlosti pripojenia na vzdialenosti stanice od prístupového bodu bol použitý bezdrôtový smerovač Linksys WRT610N v2.

#### **Charakteristika**

Linksys WRT610N v2 je dvojpásmový bezdrôtový smerovač s podporou pásiem 2,4Ghz a 5GHz. Zariadenie podporuje viac štandardov. Pri použití zariadenia pre meranie je možné voliť medzi štandardmi: 802.11a, 802.11b, 802.11g a 802.11n. Zariadenie tiež umožňuje voliť šírku kanála medzi štandardným (20MHz) a širokým kanálom (40MHz). Konfigurácia smerovača je možná pomocou pripojenia na nezabezpečenú lokálnu sieť. Smerovač je možné konfigurovať pomocou grafického rozhrania prístupného z bežného internetového prehliadača. Zariadenie je napájané pomocou adaptéra s napätím 12V. Smerovač je vybavený anténami RIFA so ziskom 4bBi pri 2.4GHz a ziskom 3,5dBi pri frekvencii 5GHz.

## 2 Výsledky študentskej práce

Táto časť študentskej práce obsahuje zhrnutie a podrobný popis praktických poznatkov použitých pri riešení študentskej práce. Text je rozdelený do tematických sekcií obsahujúcich popis riešenia problematiky simulácie a merania vlastností bezdrôtovej komunikácie podľa štandardov IEEE 802.11.

Skúmanie adaptácie rýchlosti podľa štandardov 802.11 bolo chronologicky rozdelené do postupnej analýzy problematiky a následného prevedenia merania.

Pri simulácii boli využité predošlé teoretické poznatky pre použitie vhodného scenára a jeho úpravu pre naše podmienky. Merania skutočných parametrov súvisiacich s adaptáciou rýchlosti boli realizované pomocou prideleného smerovača Cisco Linksys WRT610n v2.

### 2.1 Simulácia v prostredí NS-3

Na základe zadania študentskej práce bolo nutné po analýze prostredia NS-3 zahájiť prácu na vytváraní vhodného programového riešenia pre demonštráciu funkčnosti adaptácie rýchlosti v bezdrôtových sieťach.

#### 2.1.1 Simulačný scenár

Samotný simulačný scenár pozostával z pomerne jednoduchého modelu. Skúmanými veličinami bola priepustnosť siete a vzdialenosť od prístupového bodu. Tieto predpoklady jasne predurčovali charakter simulačného scenára. V simulovanom prostredí sa nachádzal prístupový bod s pevnou polohou. Samotné meranie prebiehalo opakovane pri každej rovnomernej zmene vzdialenosti jednej stanice od prístupového bodu.

V simulovanej sieti sa nenachádzali okrem prístupového bodu a stanice žiadne iné uzly. Scenár tiež predpokladal následné grafické spracovanie závislosti medzi priepustnosťou siete a vzdialenosťou pripojenej stanice od prístupového bodu. Scenár zahrňoval parametre prostredia, ktoré je možné simulovať aj v reálnom prostredí na pridelenom technickom vybavení.

#### 2.1.2 Inštalácia simulačného prostredia NS-3

Za účelom prevedenia simulácie v prostredí NS-3 boli preštudované softvérové požiadavky pre vytváranie simulačných skriptov. V prvom kroku bolo nutné preštudovať samotnú inštaláciu prostredia NS-3, kompatibilitu a vybrať vhodné aplikácie pre úpravu a preklad skriptov NS-3.

## Výber operačného systému

NS-3 je primárne vyvíjané na platforme GNU/Linux. Prístup k samotnému balíčku súborov pre inštaláciu NS-3 je voľne dostupný na oficiálnej stránke komunity NS-3. Kompiláciu zdrojových súborov pre NS-3 je možné štandardne spustiť na bežných distribúciách operačného systému Linux, MacOS a FreeBSD. Ďalšou softvérovou požiadavkou pre funkčnosť NS-3 je podpora programovacieho jazyka Python vo verzii 2.7 a vyššie, prípadne využitie verzie 3.4 a vyššie. Po preštudovaní väčšiny odporúčaných softvérových platforiem bola vybraná distribúcia operačného systému Linux Ubuntu 19.04 použitá vo virtualizovanej forme. Pre zabezpečenie virtuálneho stroja bol využitý voľne prístupný softvér VMware Workstation 15 Player.

## Inštalácia NS-3 v operačnom systéme.

Pre inštaláciu NS-3 bol použitý voľne prístupný balíček súborov s najnovšou verziou simulačného prostredia. Následne bol využitý odporúčaný postup a inštrukcie zverejnené komunitou NS-3. Obsahom týchto procedúr bola predovšetkým inštalácia nástrojov a súčastí, ktoré umožňujú vytváranie simulácií v NS-3. Pre samotnú inštaláciu NS-3 je možné využiť viac postupov. Komunita NS-3 poskytuje podrobný postup inštalácie pomocou predpripraveného nástroja Bake. Po spustení skriptu `bake.py` je možné sledovať automatickú inštaláciu NS-3 do operačného systému. Manuálna inštalácia je možná pomocou prostredia `ns-3-allinone`. Pri dodržaní odporúčaného postupu je po spustení príkazu `./build.py` spustený proces zostavenia súborov NS-3. Po nastavení správnej konfigurácie je proces inštalácie NS-3 ukončený. Pri inštalácii prostredia NS-3 boli odskúšané oba odporúčané spôsoby.

## Kompilácia skriptov a vývojové prostredie Eclipse

Po inštalácii prostredia NS-3 je možné pomocou príkazu `./waf` spustiť kompiláciu testovacieho skriptu. Pri vytváraní simulácií sa bežne používa priečinok `scratch`, v ktorom sa bežne nachádzajú skripty určené pre kompiláciu a spustenie. Samotný skript je možné označiť za bežný zdrojový kód pre programovací jazyk C++. Ďalšou, pomerne dôležitou časťou konfigurácie, je použitie vhodného spôsobu editácie zdrojových kódov. Pri vytváraní simulácií je editácia zdrojových kódov pomocou jednoduchého textového editora pomerne komplikovaná a vyžaduje preklad a spúšťanie skriptov priamo z príkazového riadku operačného systému. Toto riešenie nie je vhodné na vytváranie rozsiahlejšieho projektu. Uživateľsky oveľa prívetivejším spôsobom editácie skriptov je využitie vývojového prostredia. Komunita NS-3 odporúča pre vytváranie simulácií využiť programy: Eclipse, NetBeans a QtCreator. Pre našu prácu bolo vybrané vývojové prostredie Eclipse. Eclipse je integrované vývojové prostredie vhodné pre programovanie v jazykoch C++ a Java. Obsahuje

základný pracovný priestor a rozšíriteľný systém doplnkov pre prispôbenie prostredia. Po správnej konfigurácii Eclipse je možné kompilovať a spúšťať skripty simulácie priamo z grafického rozhrania. Táto koncepcia umožňuje prehľadnú editáciu vytváraného scenára priamo v adresárovej štruktúre NS-3.

### **2.1.3 Simulačný skript NS-3**

Pri realizácii simulácie bolo v prvom kroku nutné zostaviť spustiteľný skript obsahujúci dva sieťové uzly komunikujúce medzi sebou prostredníctvom bezdrôtového pripojenia podľa štandardov 802.11. Základným predpokladom prevedenia presnej simulácie je použitie vhodného komunikačného kanálu. Komunikačný kanál by mal byť schopný zahrnúť potrebné parametre pre spustenie vierohodnej simulácie bezdrôtovej siete. Ostatná problematika je zameraná hlavne na vytváranie modelu mobility, ktorý krokovo mení vzdialenosť medzi prístupovým bodom a stanicou. Vytvorená simulačná sieť tak musí reagovať na postupné zhoršovanie parametrov komunikácie. Vhodným parametrom samotného skriptu je aj možnosť výberu použitého štandardu pre multifunkčnosť samotného simulačného scenára. Záverečnou súčasťou simulácie je zaznamenávanie nameraných hodnôt rýchlosti. Tieto parametre je vhodné následne graficky zobrazit.

#### **Využitie skriptu `rate-adaptation-distance.cc`**

V súborovom systéme NS-3 sa okrem použiteľných balíčkov nachádzajú aj vytvorené vzorové scenáre. V rámci príkladov skriptu pre realizáciu simulácie bezdrôtových sietí je súčasťou NS-3 aj skript zameraný na skúmanie adaptácie rýchlosti pre rôzne verzie štandardu 802.11. Po otestovaní simulačného skriptu `rate-adaptation-distance.cc` boli namerané hodnoty graficky vykreslené pomocou programu Gnuplot. Na základe posúdenia použiteľnosti vytvorených dát bola ďalšia práca zameraná na modifikáciu a skúmanie samotného skriptu. Graficky znázornené hodnoty rýchlosti boli zaznamenávané pri každej zmene. V rámci úprav skriptu bola testovaná verzia, ktorá na rozdiel od pôvodného skriptu využíva prevádzku protokolom TCP. Táto špecifikácia bola zvolená ako najvhodnejšia. Za predpokladu využitia testovacieho programu `iperf3` je možné očakávať TCP komunikáciu aj pri realizácii skutočného merania.

#### **Práca s NS-3**

Po výbere vhodného programového riešenia pre prácu v NS-3 bolo možné plne využívať výhody vývojového prostredia Eclipse a zároveň používať potrebné nástroje

pre prácu v NS-3. Skript `rate-adaptation-distance.cc` využíval priame vstupy parametrov z terminálu. Tieto argumenty boli odovzdané spolu s príkazom `./waf`. Väčšinou bola teda využitá kombinácia prostredia Eclipse a terminálového okna operačného systému. Pri detekcii chýb bolo však vhodnejšie využiť možnosť spúšťať kompiláciu skriptov priamo v prostredí Eclipse. Užívateľsky prívetivé prostredie je schopné poskytnúť dostatok informácií pre prehľadnú a jednoduchú identifikáciu chybných častí skriptu. Po odstránení syntaktických chýb však bolo prostredie Eclipse využité väčšinou len ako editor samotného skriptu. Po úspešnej kompilácii skriptu je spustená samotná simulácia. V prípade `rate-adaptation-distance.cc` tento proces trvá niekoľko minút. Po ukončení simulácie je možné zaznamenané výstupné dáta graficky vykresliť pomocou nástroja `gnuplot`. Skript `rate-adaptation-distance.cc` bol vytvorený tak, aby bol zaznamenaný každý údaj o zmene rýchlosti v čase. Vykreslené dáta bolo následne možné porovnať s dátami získanými pri realizácii skutočného merania.

### **Popis simulačného scenára a použité súčasti**

Pri vytváraní simulačného scenára bolo v prvom kroku nutné posúdiť potrebné súčasti skriptu, ktoré umožnia vytvorenie samotnej simulácie. Ďalším podstatným atribútom scenára bol aj fakt, že výsledky simulácie bude nutné porovnať s dátami získanými pri skutočnom meraní na pridelenom bezdrôtovom smerovači. Táto skutočnosť viedla k zvoleniu jednoduchého scenára obsahujúceho jeden prístupový bod, na ktorý je pripojená stanica. Vzdialenosť medzi prístupovým bodom a stanicou sa zväčšuje v pravidelnom intervale. Základným nastavením samotnej simulácie možno chápať zvolenie vhodnej počiatočnej pozície oboch uzlov siete, veľkosť kroku a počet vykonaných krokov. Ostatné parametre nesúvisia s nastaveniami mobility, ale priamo ovplyvňujú parametre siete a použitý štandard. Vzhľadom na fakt, že boli realizované dve rôzne merania, ktoré boli vykonané v rôznych podmienkach, je možné očakávať, že samotný simulačný scenár vyžaduje modifikácie, ktoré dokážu výsledky priblížiť

k samotným výsledkom skutočných meraní. Skript `rate-adaptation-distance.cc` bol navrhnutý tak, aby umožňoval pri spustení skriptu definovať potrebné atribúty, ktoré aktívne vplývajú na samotný výstup simulácie. Skript následne definuje použitý protokol (TCP/UDP) a vytvára komunikačný kanál s definovanými parametrami. Následujúca časť popisuje najdôležitejšie súčasti použité v skripte.[7]

**Použité `YansWifiPhyHelper` a `YansWifiChannelHelper`** sú moduly umožňujúce vytvorenie objektov, ktoré reprezentujú komunikačný kanál a fyzické parametre bezdrôtovej siete.

šírka pásma	20MHz
minimálna energia prijímaného signálu	-101dBm
maximálna dostupná úroveň prenosu	16,0206dBm
úroveň šumu (Signal-to-Noise-Ratio)	7dB
oneskorenie medzi dvoma rámcami	250000,0ns
počet antén	1

Tab. 2.1: Parametre fyzickej siete.

```
YansWifiPhyHelper wifiPhy=YansWifiPhyHelper::Default();
YansWifiChannelHelper wifiChannel=YansWifiChannelHelper::Default();
```

Pri definovaní parametrov fyzickej vrstvy boli použité vopred nastavené parametre (Default). V hore uvedenej tabuľke sa nachádzajú orientačné informácie o definícii fyzickej vrstvy. Tieto parametre je možné upravovať podľa požiadaviek, ktoré sú kladené na základe použitého reálneho zariadenia.

Pri testovaní a modifikovaní simulačného skriptu boli odskúšané rôzne nastavenia, ktoré priamo zasahovali do simulačného scenáru a ovplyvňovali výstupné dáta. Najväčší podiel na funkčnosti samotného algoritmu adaptácie rýchlosti malo použitie rôznych tried WifiManager. Problematika správnej voľby wifi manažéra je založená predovšetkým na podpore adaptácie rýchlosti. Implementácia adaptácie rýchlosti nie je štandardne implementovaná v každom modeli. Pri vytváraní simulačného skriptu je nutné na základe dokumentácie posúdiť možnosti použitia rôznych tried WifiManager. **MinstrelHtWifiManager** je trieda podporujúca manažment adaptácie rýchlosti pre štandardy 802.11n a 802.11a/c. Použitie tohto manažéra je pre podporované rýchlosti bezproblémové. Pre vytváranie simulácií, ktoré zahŕňajú staršie štandardy ako napríklad 802.11b a g, bolo nutné voliť triedy WifiManager, ktoré majú implementovanú podporu adaptácie rýchlosti pre staršie štandardy. **RraaWifiManager** má implementovaný algoritmus adaptácie rýchlosti RRAA (Robust Rate Adaptation Algorithm). Tento druh adaptácie vychádza z pôvodnej definície, ktorá bola zverejnená v štandarde 802.11. RraaWifiManager tak spĺňa parametre pre použitie pri simulácii starších štandardov. Jeho hlavnou nevýhodou je nemožnosť využitia pre novšie štandardy. Podobne ako pri RraaWifiManager je aj pri **AarfWifiManager** možné realizovať simulácie starších verzií štandardov be-

zdrôtových sietí. AarfWifiManager simuluje AARF (adaptive auto rate fallback) adaptáciu rýchlosti. Jeho využitie je taktiež obmedzené len na staršie verzie štandardu 802.11.

**ConstantSpeedPropagationDelayModel** a **LogDistancePropagationLossModel** definujú parametre komunikačného kanála. Pri vytváraní simulačného skriptu bol odskúšaný vplyv atribútov, ktoré vytvárajú straty v komunikačnom kanáli. Problematika stratových modelov je však pomerne rozsiahla. Pri testovaní simulačného skriptu boli odskúšané viaceré stratové modely ako napríklad: ThreeLogDistancePropagationLossModel, FixedRssLossModel a ďalšie. Základným parametrom pre použiteľnosť stratového modelu je jeho schopnosť reagovať na vzdialenosť stanice od prístupového bodu. Väčšina testovaných stratových modelov umožňuje nastavenie parametrov útlmu a strát. Model šírenia signálu je pomerne podstatným parametrom simulácie. Podrobnejší popis problematiky modelov šírenia signálu je popísaný v sekcii určenej na popis parametrov ovplyvňujúcich výsledky simulácií. V doleuvedenom príklade sa nachádza príklad definície modelu šírenia a zahrnutie modelu oneskorenia signálu.

```
wifiChannel.  
SetPropagationDelay ("ns3::ConstantSpeedPropagationDelayModel");
```

```
wifiChannel.  
AddPropagationLoss ("ns3::LogDistancePropagationLossModel");
```

**TcpSocketFactory** a **UdpSocketFactory** sú triedy umožňujúce definíciu druhu komunikácie a použitie protokolov TCP a UDP

```
PacketSinkHelper sink ("ns3::TcpSocketFactory" ...
```

Vzhľadom na potrebu krokovej zmeny vzdialenosti medzi prístupovým bodom a stanicou bola použitá trieda **MobilityHelper**.

```
MobilityHelper mobility;  
positionAlloc ->Add( Vector( ap1_x, ap1_y, 0.0 ));  
positionAlloc ->Add( Vector( sta1_x, sta1_y, 0.0 ));  
mobility.SetPositionAllocator( positionAlloc );  
mobility.SetMobilityModel( "ns3::ConstantPositionMobilityModel " );  
mobility.Install( wifiApNodes.Get( 0 ));  
mobility.Install( wifiStaNodes.Get( 0 ));
```

Po postupnej definícii všetkých parametrov simulovanej bezdrôtovej siete je použitý nástroj **Gnuplot**, ktorý umožňuje grafické spracovanie získaných dát.

### **Riadenie náhodných premenných v simulácii**

Simulácie v prostredí NS-3 je možné konfigurovať tak, aby poskytovali deterministické výsledky simulácií alebo zabezpečili náhodné výsledky simulácie. Ak je simulácia v NS-3 nakonfigurovaná na použitie pevného deterministického semena s rovnakým číslom cyklu, mala by pri každom spustení poskytnúť rovnaký výstup. Simulácie v NS-3 štandardne používajú pevné počiatočné číslo a počiatočné semeno. Tieto hodnoty sú uložené v dvoch inštanciách verejných premenných: `g_rngSeed` a `g_rngRun`. Zmenu hodnôt globálnych premenných je možné vykonať pri spúšťaní simulácie.

```
$ NS_GLOBAL_VALUE="RngRun=3" ./waf --run rate-adaptation-distance.cc
```

Pomocou zavedenia modifikácie globálnych premenných sa do výsledkov simulácie vnáša istá miera neurčitosti, ktorá je vhodná pri štatistickom vyhodnotení väčšieho množstva získaných dát.

#### **2.1.4 Parametre a súčasti ovplyvňujúce výsledky simulácie.**

V nasledujúcej sekcii sú podrobnejšie popísané súčasti, ktoré svojimi parametrami ovplyvňujú výsledky simulácie. Okrem základných parametrov, týkajúcich sa samotnej mobility sú prítomné simulačné modely, ktoré aktívne zasahujú do kondície komunikačného kanálu. Je nutné konštatovať, že aj samotná analýza adaptácie rýchlosti je závislá od použitého nastavenia komunikačného kanálu. Problematika je najviac závislá od charakteru simulácie v súvislosti s fyzickou vrstvou. Simulačné prostredie NS-3 ku komunikácii vzhliada z pohľadu generovaných paketov v sieti. Následne sú posudzované rôzne atribúty, ktoré zabezpečia úspešné doručenie správy adresátovi. Z pohľadu užívateľa je teda nutné počítať s faktom, že existuje objekt, ktorý posúdi momentálnu situáciu v sieti, zahrnie úroveň šumu a na základe použitého modelu rušenia k danej správe priradí určitú úroveň chybovosti a pravdepodobnosti doručenia správy k adresátovi.[7]

#### **Model chybovosti**

Ako už bolo v tejto práci popísané, model chybovosti šírenia signálu je štandardne zahrnutý v simulačných scenároch prostredia NS-3. Cieľom tohto objektu je modelovať v čo najreálnejšej miere šírenie signálu v priestore.



## LogDistancePropagationLossModel

Tento model zahŕňa straty, ktoré sú bežné predovšetkým v budovách a v husto obývaných oblastiach. Model počíta výkonovú úroveň prijatého signálu na základe nasledovného vzorca. [8]

$$L = L_0 + 10n \log \left( \frac{d}{d_0} \right) \quad (2.1)$$

$n$  - stratový exponent vzdialenosti

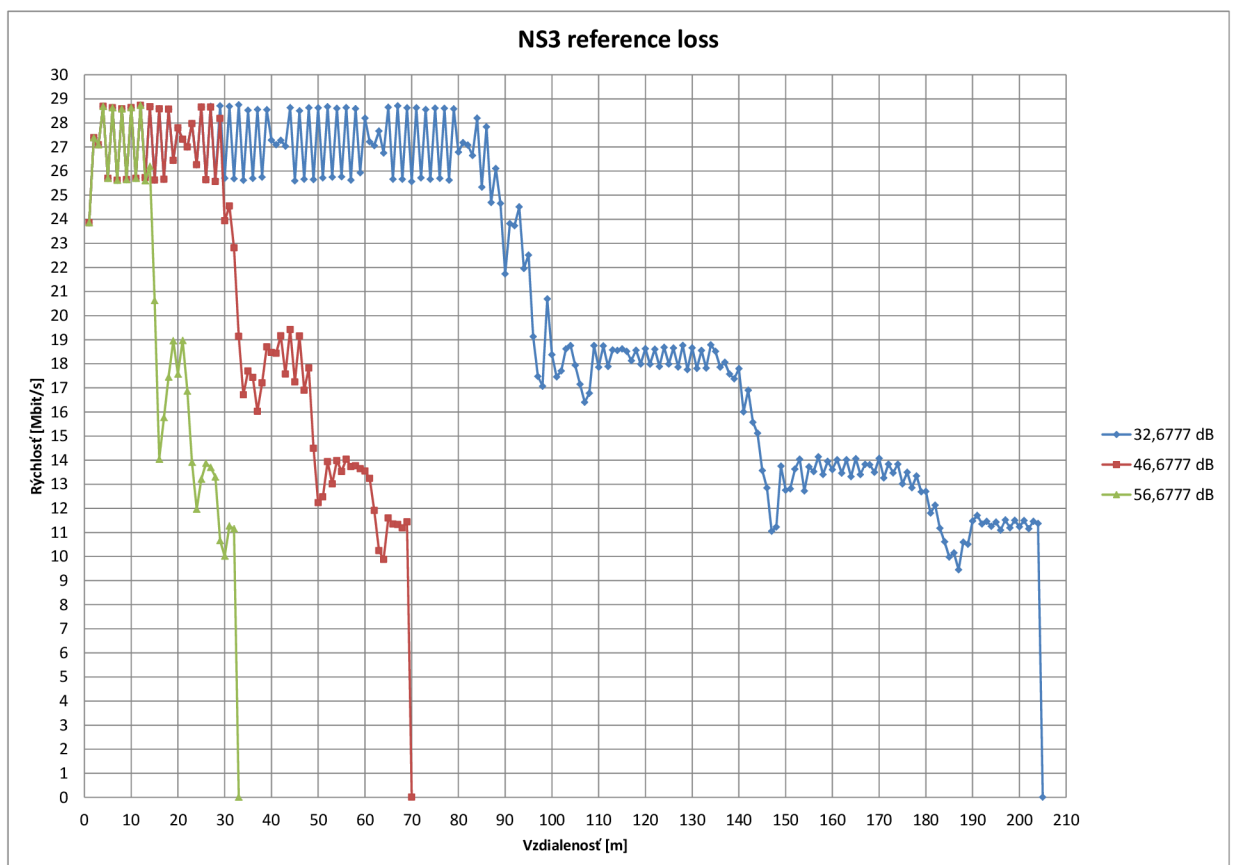
$d_0$  - referenčná vzdialenosť

$L_0$  - strata pri referenčnej vzdialenosti

$d$  - vzdialenosť

$L$  - strata

V predvolenom nastavení komunikačného kanála sa práve tento model chybovosti využíva s referenčnou hodnotou straty  $L_0 = 46,6777 \text{ dB}$  pri vzdialenosti jeden meter od vysielača. Úpravou referenčnej hodnoty strát je možné modifikovať dosah bezdrôtovej siete. V nižšie uvedenom grafe sa nachádza porovnanie dosahu siete v závislosti na nastavenej referenčnej strate.



Obr. 2.1: Porovnanie referenčných strát pre model LogDistancePropagationLossModel.

## FriisPropagationLossModel

V prípade potreby simulácie prostredia na otvorenom priestranstve bolo nutné kalkulovať s parametrami, ktoré nie sú typické pre mestské, husto zastavané plochy. FriisPropagationLossModel je svojim charakterom určený hlavne pre telekomunikačné simulácie na otvorenom priestranstve.[8]

$$P_r = \frac{P_t G_t G_r \lambda^2}{(4\pi d)^2 L} \quad (2.2)$$

$P_t$  - prenosový výkon ( $W$ )

$P_r$  - prijímaný výkon ( $W$ )

$G_t$  - zisk prenosu

$G_r$  - zisk príjmu

$\lambda^2$  - vlnová dĺžka

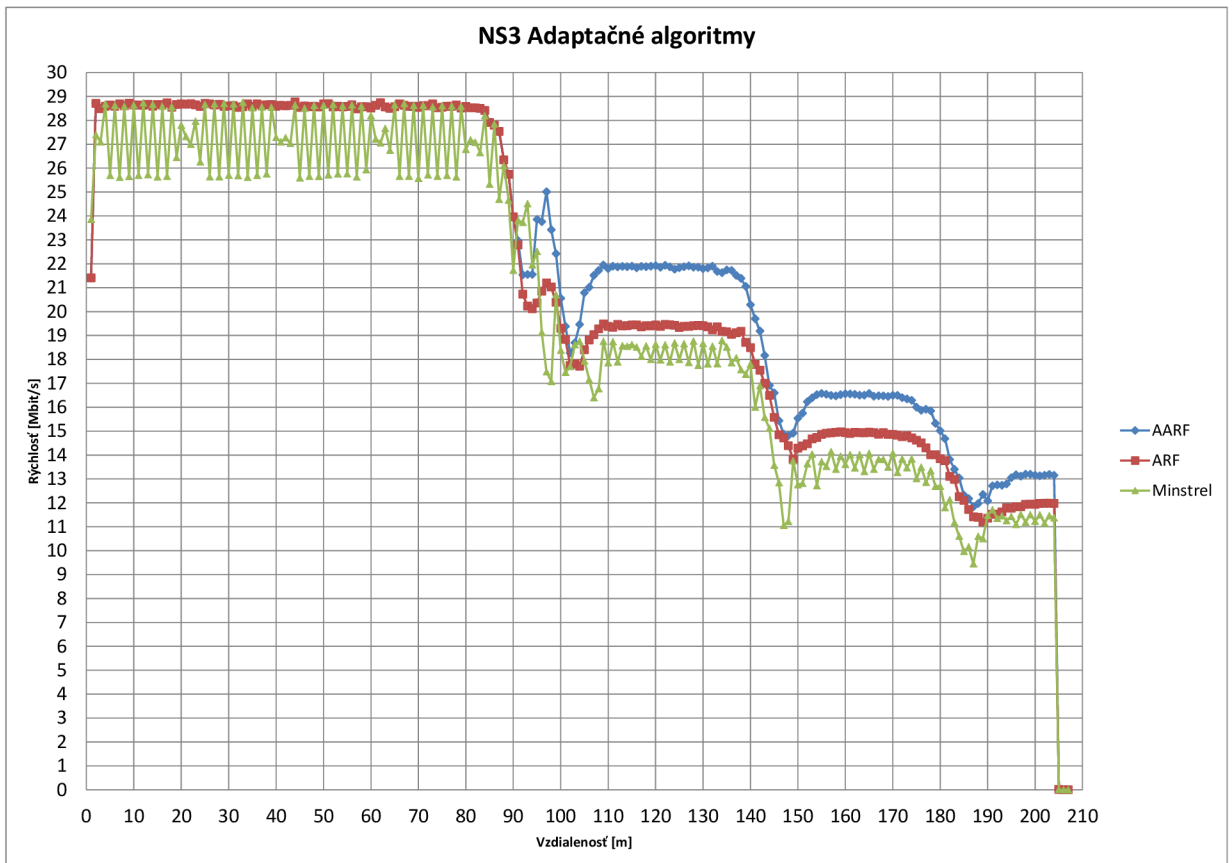
$d$  - vzdialenosť

$L$  - strata systému

Použitie tohto modelu je z hľadiska realistického pohľadu na simulácie pomerne problematické. Je nutné konštatovať, že bez úprav atribútov minimálnej straty a modifikácie tohto modelu je výsledok simulácie analytický. Pri snahe o realistickú simuláciu bezdrôtovej siete je vhodné preferovať iné modely straty pri šírení signálu. Pomerne zásadným faktorom je však to, že k skúmaniu tohto konkrétneho modelu došlo na základe výsledkov druhého merania v reálnom prostredí. Dosahy bezdrôtovej siete sa aj v skutočnom svete môžu bližšie k stovkám metrov. V závere platí, že bližšie skúmanie tohto modelu a jeho modifikácia nie sú priamo cieľom tejto práce.[8]

## Adaptačné algoritmy

Z hľadiska problematiky adaptácie rýchlosti sú pomerne podstatnou súčasťou simulačného scenára objekty tried *WifiManager*. Simulačné prostredie NS-3 ponúka rôzne varianty tejto triedy určené na rôzne špecifické simulácie. V rámci prístupnej dokumentácie je možné analyzovať predovšetkým tie triedy, ktoré zahŕňajú jednoznačnú definíciu adaptačných algoritmov. Experiment bol zameraný na algoritmy ARF, AARF a komunitou preferovaný algoritmus Minstrel. Tieto zvolené algoritmy boli zahrnuté v triedach: *AarfWifiManager*, *ArfWifiManager*, *MinstrelWifiManager*. Testované triedy boli volené tak, aby zahŕňali bežne používané adaptačné algoritmy. Cieľom experimentov bolo porovnať vplyv zvoleného algoritmu na výsledné závislosti prenosovej rýchlosti na vzdialenosti. Z doleuvedeného grafu vyplýva, že použitý adaptačný algoritmus viditeľne ovplyvňuje výsledné závislosti.[9]



Obr. 2.2: Porovnanie adaptačných algoritmov pre štandard 802.11g.

## Model oneskorenia

**ConstantSpeedPropagationDelayModel** zabezpečuje základné vlastnosti šírenia signálov v prostredí. Model počíta s konštantnou rýchlosťou šírenia signálu. Oneskorenie signálu je odvodené zo vzdialenosti medzi vysielačom a príjemcom signálu. Vzhľadom na bežné vzdialenosti využívané v bezdrôtových sieťach tento model nezahrňuje oneskorenie spôsobené zakrivením Zeme.

## Vysielací výkon

Pri vytváraní funkčného a realistického scenára bolo nutné zabezpečiť definíciu úrovne výkonu vysielača. Je nutné konštatovať, že dosah bezdrôtovej siete priamo závisí na použitej výkonovej úrovni vysielača. V tomto prípade bola analýza pomerne jednoznačná. Atribút výkonu bol zvolený podľa technickej špecifikácie zariadenia Cisco linksys wrt610n v2.

```
Config :: Set ("/NodeList/*/DeviceList*/  
$ns3::WifiNetDevice/Phy/TxPowerEnd", DoubleValue (20));
```

```
Config :: Set ( "/NodeList/*/DeviceList/*/  
$ns3::WifiNetDevice/Phy/TxPowerStart" , DoubleValue (20));
```

### 2.1.5 Výsledky simulácie

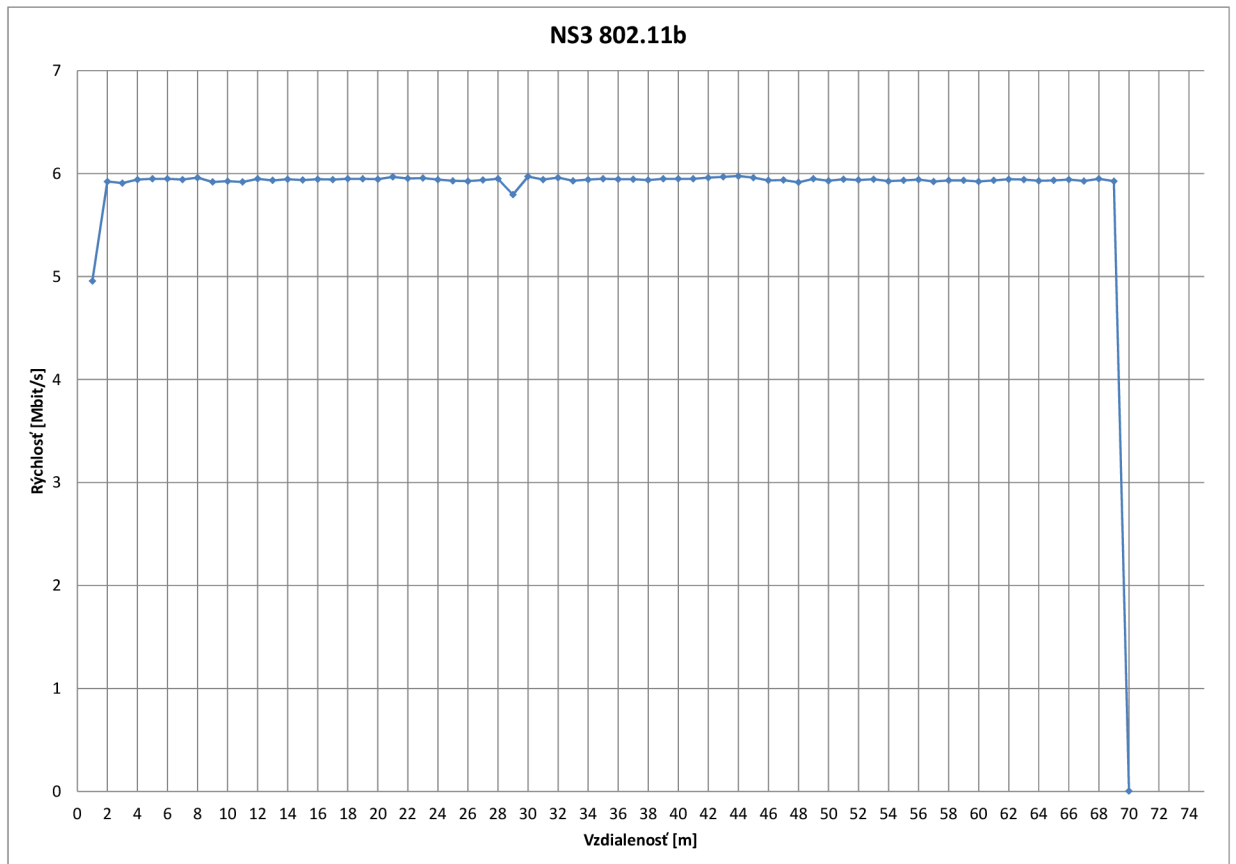
Po odskúšaní pôvodnej konfigurácie skriptu rate-adaptation-distance.cc bol pôvodný skript upravený tak, aby sa čo najviac približoval parametrom skutočného merania. Hlavným výstupom simulácie bola závislosť vzdialenosti od rýchlosti bezdrôtovej siete. Pre účely overenia presnosti simulovaných dát boli zvolené tri štandardy: 802.11b, 802.11g a 802.11n.

#### Vyhodnotenie simulácie

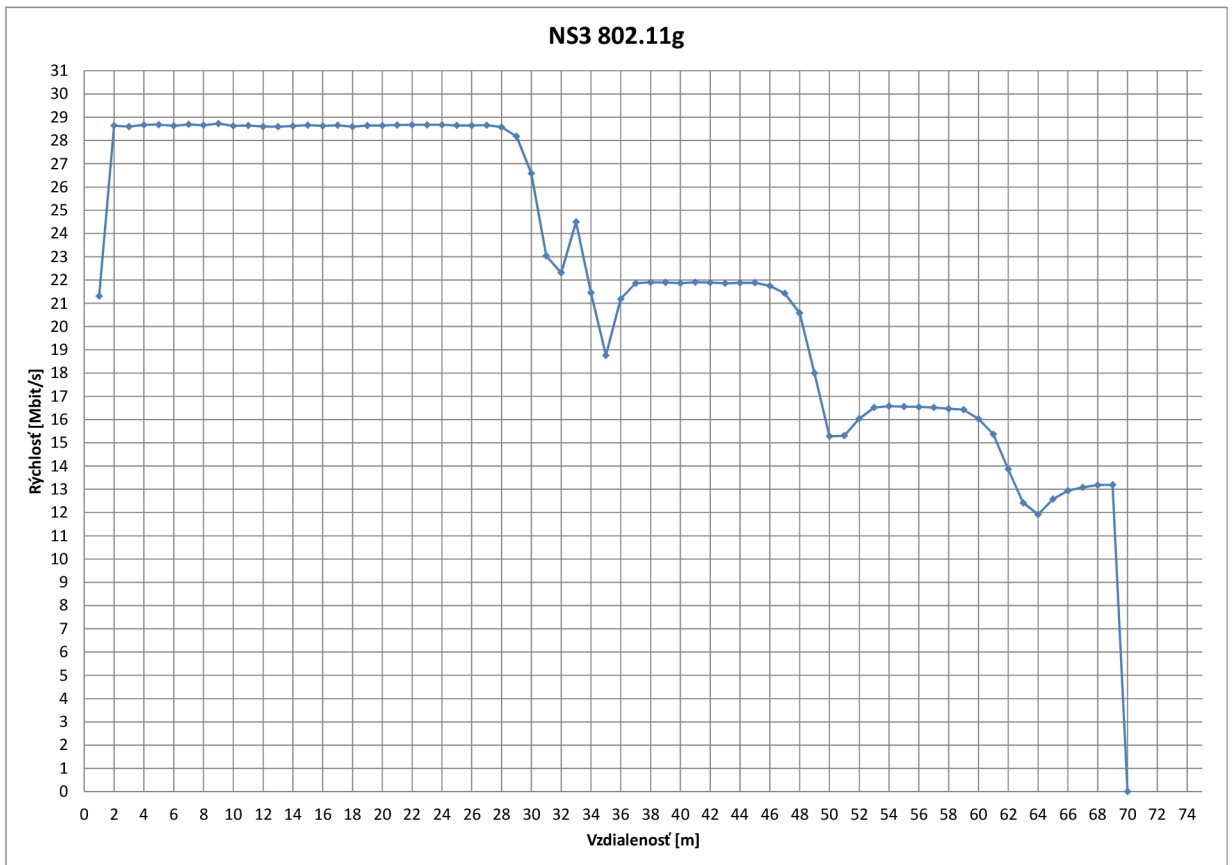
Simulačný model umožňujúci realizáciu simulačného scenára bezdrôtovej siete v NS-3 bol schopný aktívne reagovať na zmeny v sieti. Z nameraných dát je zrejmé, že samotný simulátor reálne upravuje prenosovú rýchlosť pri znižovaní kvality signálu podľa štandardov IEEE. Pri samotnej simulácii je však možné sledovať pomerne veľkú zmenu rýchlosti v závislosti na použitom druhu simulačnej prevádzky. V prípade štandardu 802.11b bolo pri použití UDP komunikácie zreteľné, že je zvolená konštantná rýchlosť 2,5Mb/s. Toto nastavenie nebolo upravené počas celej doby simulácie. Zastavenie komunikácie bolo spôsobené zrejme až stratou signálu z prístupového bodu.

V prípade štandardu 802.11n dochádza aj pri použití protokolu UDP k postupnému znižovaniu rýchlosti. V oboch prípadoch je však možné sledovať zreteľne nižšie nastavené rýchlosti od úplného započatia komunikácie v bezprostrednej vzdialenosti od prístupového bodu.

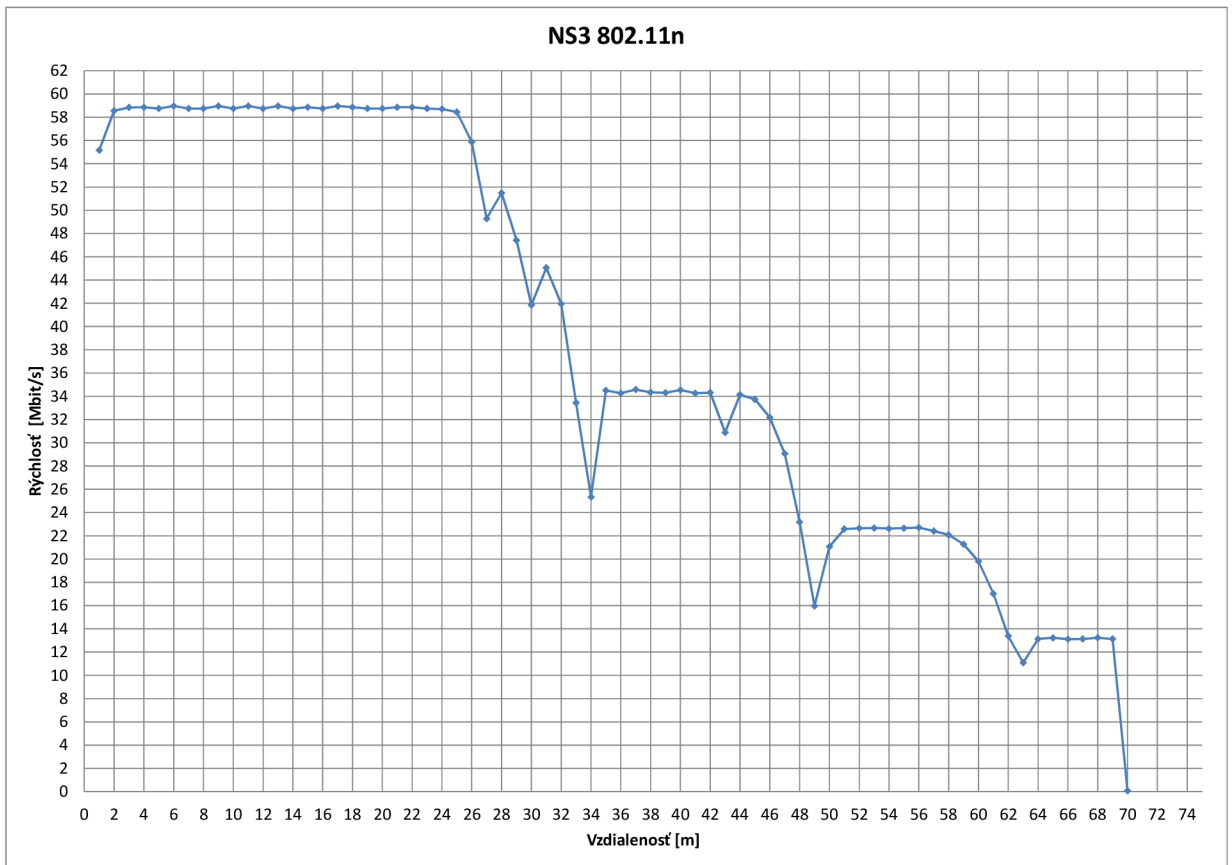
## Grafy



Obr. 2.3: Výsledok simulácie pre štandard 802.11b.



Obr. 2.4: Výsledok simulácie pre štandard 802.11g.



Obr. 2.5: Výsledok simulácie pre štandard 802.11n.

## 2.2 Simulácia v prostredí OMNeT++

Po teoretickom spracovaní problematiky simulácií v prostredí OMNeT++ bol zahájený proces inštalácie simulačného prostredia a následné testovanie jeho funkčnosti. Cieľom tejto činnosti bolo predovšetkým otestovať správanie prostredia OMNeT++ pri vytváraní simulačných scenárov, ktoré sú porovnateľné s reálnym meraním.

### 2.2.1 Simulačný scenár

Pri realizácii simulácií bol kladený dôraz predovšetkým na demonštráciu funkcie adaptačných algoritmov v OMNeT++. Simulovaný scenár očakáva existenciu jedného prístupového bodu a jednej stanice, ktorá je pripojená do bezdrôtovej siete. Testovanie samotnej adaptácie rýchlosti bolo realizované na základe zmeny parametrov komunikácie, ktorá bola spôsobená zmenou vzdialenosti stanice od prístupového bodu. Dôraz sa kládol predovšetkým na dodržanie podobnosti s predošlým postupom, ktorý bol realizovaný v prostredí NS-3. Cieľom bolo tiež vytvoriť názorné grafické zobrazenie nameraných hodnôt. Vzhľadom na predpoklad, že simulačné prostredie OMNeT++ umožňuje názorné grafické zobrazenie prebiehajúcej simulácie v reálnom čase, boli otestované rôzne možnosti zobrazenia získaných hodnôt rýchlosti a priepustnosti siete.

### Inštalácia simulačného prostredia OMNeT++

Po teoretickom preštudovaní softvérových požiadaviek bol zvolený vhodný postup na inštaláciu simulačného prostredia. OMNeT++ je možné využívať na rôznych operačných systémoch. Komunita OMNeT++ poskytuje rôzne verzie simulačného prostredia voľne prístupné na oficiálnej webovej stránke. Pre naše účely bola zvolená najnovšia verzia simulačného prostredia OMNeT++ určená pre operačné systémy Windows 7 a 10 (64-bit). OMNeT++ je však platforma prístupná aj vo verziách určených pre iné bežne rozšírené operačné systémy.

### 2.2.2 Inštalácia prostredia OMNeT++ v operačnom systéme Windows 10

Po výbere príslušnej verzie prostredia OMNeT++ bol dodržaný postup, ktorý je odporúčaný komunitou OMNeT++. Samotná inštalácia simulačného prostredia je po rozbalení všetkých súborov umožnená pomocou pripraveného skriptu mingwenv.cmd. Po spustení je možné pomocou konzolového vstupu zahájiť preklad a rozbalenie všetkých potrebných súborov pre zabezpečenie funkčnosti OMNeT++. Následne,



podľa odporúčaného postupu, boli spustené testovacie simulácie, ktoré overili funkčnosť simulačnej platformy. Posledným krokom bolo spustenie grafického prostredia OMNeT++ IDE, ktoré umožňuje editovanie zdrojových kódov a tiež vizualizáciu samotného simulačného scenáru.

## **Knižnica INET**

INET je voľne prístupná knižnica modelov pre simulačné prostredie OMNeT++. Poskytuje modely protokolov a agentov určených pre vytváranie simulácií v komunikačných sieťach. INET je odporúčaný predovšetkým pri analýze protokolov a ich súčastí. Pomocou knižníc INET je možné realizovať podporu protokolov na všetkých sieťových vrstvách. Vďaka tejto vlastnosti je možné pri simulácii zabezpečiť analýzu protokolov na všetkých vrstvách sieťového modelu. INET obsahuje prvky, ktoré simulujú signalizáciu v sieti. Súčasťou tohto balíka je aj množstvo aplikačných modelov, ktoré definujú prevádzku v sieti. INET je postavený na koncepcii modulov, ktoré komunikujú odovzdávaním správ. Agenti a sieťové protokoly sú zastúpené komponentmi, ktoré je možné ľubovoľne kombinovať. Výhodou je aj podrobná definícia komunikačného prostredia a parametrov fyzickej vrstvy. Tieto predpoklady podstatnou mierou vplývajú na definíciu simulovaného scenára bezdrôtovej siete s dôrazom na analýzu adaptačných algoritmov. Knižnicu INET je možné pomocou nastavení zahrnúť do bežného projektu prostredia OMNeT++.

### **2.2.3 Vytváranie simulácií v OMNeT++**

Simulačné prostredie po zahrnutí knižnice INET obsahuje adresárovú štruktúru, zahŕňajúcu okrem simulačných modelov aj vopred pripravené scenáre, ktoré je možné využiť na oboznámenie sa s problematikou vytvárania simulácií v OMNeT++. Samotné simulačné prostredie vychádza z vývojového prostredia Eclipse. OMNeT++ však vo vizuále Eclipse zahrňuje aj grafické prvky reprezentujúce samotnú simulovanú sieť. Bežný simulačný scenár je definovaný pomocou súboru .ned, ktorý definuje samotnú štruktúru siete. Užívateľ má možnosť definovať sieť pomocou skriptov, alebo sa zamerať na grafické rozhranie. Pri tejto práci bol kladený dôraz predovšetkým na vytváranie skriptov bez použitia grafických formulárov. Ďalšou súčasťou štandardného scenára je inicializačný súbor .ini, ktorý obsahuje definíciu parametrov pre každý simulačný model a objekt. Tento súbor je taktiež možné upravovať pomocou skriptov a grafických formulárov. Aj v tomto prípade však bol kladený dôraz na vytváranie skriptov bez použitia grafických formulárov. Po spustení prekladu vytvoreného skriptu sa v oddelenom okne otvorí aplikácia, ktorá reprezentuje samotnú spustiteľnú simuláciu. Simulačné okno disponuje rôznymi nástrojmi, ktoré sú využiteľné pre analýzu simulácie v reálnom čase. OMNeT++ umožňuje v inicializačnom

súbore definovať parametre, ktoré zabezpečujú grafickú prezentáciu signálových tokov a umožňujú užívateľovi sledovať celý priebeh simulácie. Tieto nástroje je možné pokladať za pomerne podstatnú súčasť prostredia OMNeT++. Samotná vizualizácia simulácie diskretných udalostí v sieti je doplnená možnosťou zberu dát, ktoré je možné znázorňovať pomocou grafov. Táto funkcia je štandarde zabezpečená pomocou súborov .anf. Po využití užívateľského dialógu vo vývojovom prostredí je možné sledovať dáta simulácie reprezentované pomocou skalárnych a vektorových grafov. [15][16]

## Definícia siete

Štruktúra simulačného prostredia OMNeT++ umožňuje definovať prvky siete pomocou súboru .ned. Prostredie umožňuje okamžitú vizualizáciu definovanej siete a prístup k zdrojovému skriptu, ktorý popisuje všetky moduly v sieti. Užívateľ má možnosť použiť moduly v závislosti na simulačnom scenári. Okrem sieťových prvkov je súčasťou definície siete aj zahrnutie štatistických nástrojov a definícia komunikačného kanálu pomocou fyzického prostredia. Po zahrnutí knižnice INET je možné využiť pomerne širokú škálu nástrojov a modulov pre bezdrôtové siete.

Pri vytváraní simulačných scenárov bezdrôtovej siete má väčšinou užívateľ možnosť využiť viac špecifických modulov, ktoré umožňujú vytvárať sieťové prvky. Samotný uzol bezdrôtovej siete je možné definovať pomocou modulu **WirelessHost**. Po vložení sieťového uzla do scenára sa v zdrojovom kóde nachádza definícia samotného uzla a príslušné parametre zobrazenia. Podobne, ako pri definícii sieťového uzla, je možné postupovať aj pri vkladaní modulov fyzického prostredia **PhysicalEnvironment** a prenosového média **Ieee80211ScalarRadioMedium**. Následne je do simulačného scenára štandardne zahrnutý aj sieťový konfiguratívny **Ipv4NetworkConfigurator** umožňujúci jednoduché definovanie konfigurácie vytvorenej siete. Bežnou súčasťou definície je aj modul zabezpečujúci vizualizáciu príslušných modulov a dejov v sieti.

submodules :

```
sinkHost: WirelessHost {
    parameters:
        @display("p=252.65,169.51999;r=, ,#707070");
}
physicalEnvironment: PhysicalEnvironment {
    parameters:
        @display("p=131.53847,86.15385");
}
radioMedium: Ieee80211ScalarRadioMedium {
    parameters:
```

```

        @display("p=130.76924,168.46155");
    }
    configurator: Ipv4NetworkConfigurator {
        @display("p=131.53847,253.84616");
    }
    visualizer: IntegratedCanvasVisualizer {
        @display("p=131.53847,336.15387");
    }
}

```

## Inicializácia a parametre siete

Pri realizácii simulácie v prostredí OMNeT++ je nutné po zahrnutí sieťových modulov zabezpečiť inicializáciu a nastavenie parametrov samotnej siete. Táto procedúra je štandardne viazaná na súbor `omnetpp.ini` obsahujúci popis použitých vlastností a protokolov v sieti. Užívateľ pomocou vkladania nastavení modulov do súboru `omnetpp.ini` realizuje nastavenie všetkých potrebných parametrov, ktoré definujú správanie simulovanej siete. Okrem definície parametrov každej vrstvy bežného sieťového modelu je možné realizovať nastavenia parametrov mobility prvkov a tiež filtrovať vizualizačné nástroje, ktoré po spustení samotnej simulácie graficky zobrazujú príslušné deje v sieti. Nižšie uvedený skript reprezentuje príklad definície parametrov mobility a vizuálnych vlastností simulácie.

```

# visualization settings
*.visualizer.obstacleLossVisualizer.displayIntersections = true
*.visualizer.obstacleLossVisualizer.fadeOutTime = 40ms
*.visualizer.obstacleLossVisualizer.fadeOutMode = "animationTime"
*.visualizer.dataLinkVisualizer.displayLinks = true
*.visualizer.dataLinkVisualizer.packetFilter = "Udp*"
*.visualizer.dataLinkVisualizer.fadeOutMode = "animationTime"
*.visualizer.dataLinkVisualizer.fadeOutTime = 200ms

# mobility settings
**Host.mobility.typename = "RectangleMobility"
*.sourceHost.mobility.startPos = 0
*.sourceHost.mobility.speed = 200mps
*.sinkHost.mobility.startPos = 3
*.sinkHost.mobility.speed = 100mps

```

Vzhľadom na nutnosť využiť adaptačné algoritmy je nutné v nastavení parametrov

zahrnúť definíciu adaptačného algoritmu a nastavenie parametrov najnižších vrstiev sieťového modelu. Inicializácia simulácie umožňuje využiť argumenty, ktoré je možné definovať pri spúšťaní simulácie.

V nižšie uvedenom príklade sú uvedené nastavenia, ktoré sú viazané na vstupný argument inicializácie reprezentované premennou **rateControl**, ktorá ovplyvňuje parametre fyzickej vrstvy a zavedenie adaptačného algoritmu (**AarfRateControl**).

```
# nic settings
**Host.wlan[*].opMode = "g(erp)"
**Host.wlan[*].radio.transmitter.power = 50mW
**Host.wlan[*].mgmt.typename = "Ieee80211MgmtAdhoc"
**Host.wlan[*].agent.typename = ""
**Host.wlan[*].mac.*.rateControl.typename =
    ${rateControl="" , "AarfRateControl"}

**Host.wlan[*].mac.*.rateControl.initialRate = 54Mbps
**Host.wlan[*].mac.dataBitrate =
    (${rateControl} == "" ? 54Mbps : -1bps)

**Host.wlan[*].mac.*.rateSelection.dataFrameBitrate =
    (${rateControl} == "" ? 54Mbps : -1bps)

**Host.wlan[*].mac.dcf.rateControl.increaseThreshold = 20
**Host.wlan[*].mac.dcf.rateControl.decreaseThreshold = 5
**Host.wlan[*].mac.dcf.rateControl.interval = 1s
```

Podstatnou súčasťou inicializácie parametrov siete je aj zavedenie prevádzky na aplikačnej vrstve. Táto procedúra je v nižšie uvedenom príklade reprezentovaná definíciou parametrov aplikácie využívajúcej UDP prevádzku, ktorá ovplyvňuje nižšie položené vrstvy sieťového modelu.

```
# application settings
**Host.numApps = 1

*.sourceHost.app[0].typename = "UdpBasicApp"
*.sourceHost.app[0].destAddresses = "sinkHost"
*.sourceHost.app[0].destPort = 5000
*.sourceHost.app[0].messageLength = 1000B
```

```
*.sourceHost.app[0].sendInterval = 0.1ms
```

```
*.sinkHost.app[0].typename = "UdpSink"
```

```
*.sinkHost.app[0].localPort = 5000
```

V závislosti od nastavenej inicializácii je po spustení simulácie možné sledovať grafickú vizualizáciu, ktorá znázorňuje sieťové deje podľa nastavenej vizualizácie.

## 2.2.4 Modifikácia scenára Rate Controll Showcase

Vzhľadom na pomerne rozsiahlu škálu simulačných modulov je v knižnici INET implementovaných množstvo vytvorených scenárov, ktoré obsahujú potrebné simulačné moduly a nástroje na plné využitie možností samotného simulačného prostredia.

V súborovej štruktúre sú tieto scenáre delené do kategórií, ktoré zahrňujú väčšinu bežných reálnych situácií, ktoré je možné predpokladať pri testovaní siete. V prípade bezdrôtových sietí definovaných štandardom IEEE 802.11 sú prístupné rôzne scenáre, ktoré mapujú parametre a funkčnosť bezdrôtovej siete.

### Popis pôvodného scenára Rate Controll Showcase

Pôvodná definícia simulovanej siete obsahuje dva sieťové uzly, ktoré medzi sebou komunikujú pomocou UDP prevádzky. Fyzické prostredie obsahuje prekážku, ktorá je umiestnená v strede priestoru. Pomocou nástrojov mobility sú sieťové uzly v neustálom pohybe. Po spustení simulácie je možné sledovať pomocou vizuálnych nástrojov aktuálnu zvolenú prenosovú rýchlosť v sieti. Tento scenár tak demonštruje funkčnosť nástrojov mobility a funkčnosť adaptačného algoritmu v bezdrôtovej sieti.

Inicializačný súbor obsiahnutý v priečinku scenára Rate Controll Showacase predstavuje spôsob, akým užívateľ môže priamo ovplyvňovať parametre siete. Pri snahe o priblíženie sa reálnemu scenáru, ktorý je možné porovnať s predchádzajúcimi výsledkami simulácií sa dá odstrániť preddefinovanú fyzickú prekážku v priestore a modifikovať model mobility tak, aby dochádzalo k postupnému vzdalovaniu sa stanice od prístupového bodu. Taktiež je možné upraviť protokoly na vyšších vrstvách sieťového modelu tak, aby boli získané dáta porovnateľné s ostatnými hodnotami rýchlosti. Samotný adaptačný algoritmus je založený na modele **Adaptive Auto Rate Fallback (AARF)**.

## Spúšťanie simulácie

Užitočnou vlastnosťou prostredia OMNeT++ je možnosť vizualizácie simulovaných dejov, ktoré prebiehajú v reálnom čase. Po spustení pôvodnej simulácie Rate Controll Showcase sa užívateľovi zobrazujú okrem sieťových prvkov aj grafické objekty, ktoré umožňujú sledovanie prenosovej rýchlosti v reálnom čase. Na základe dát z inicializačného súboru je možné definovať maximálny čas simulácie. Počas simulácie sú štatistické dáta zaznamenávané v podobe súborov, ktoré je možné následne graficky zobrazit.

## Úpravy pôvodného scenára

V súvislosti s realizáciou porovnania nameraných dát bolo nutné pôvodný scenár Rate Controll Showcase upraviť tak, aby získané výsledky simulácie boli porovnateľné s ostatnými meraniami. Prvotnou úpravou bola modifikácia modelu mobility sieťových prvkov. Po úprave je počas simulácie v pohybe len jeden sieťový prvok, ktorý sa pohybuje lineárne, konštantnou rýchlosťou.

```
*.sinkHost.mobility.typename = "LinearMobility"  
*.sinkHost.mobility.startPos = 0  
*.sinkHost.mobility.speed = 1mps  
*.sinkHost.mobility.initialMovementHeading = 0deg
```

Ďalšou úpravou pôvodného scenára bolo odstránenie fyzickej prekážky z priestoru simulácie. Úpravou rozmerov simulovaného priestoru vzniká voľné priestranstvo, umožňujúce testovanie rýchlosti v celom dosahu bezdrôtovej siete.

Problematika samotnej prevádzky v bezdrôtovej sieti je pomerne závislá na definícii prevádzky na aplikačnej vrstve sieťového modelu. Povaha tejto komunikácie určuje aj prenosové rýchlosti, ktoré sú v bezdrôtovej sieti nastavené. Počas testovania možností simulačného scenára boli okrem protokolu UDP testované aj možnosti prevádzky pomocou protokolu TCP. Nižšie uvedená definícia je príkladom bežnej komunikácie TCP formou klienta a servera.

```
*.*Host.numApps = 1  
*.sinkHost.app[*].typename = "TcpSinkApp"  
*.sinkHost.app[0].localPort = 1000  
*.sourceHost.app[*].typename = "TcpEchoApp" # ftp
```

```

*.sourceHost.app[0].active = true
*.sourceHost.app[0].connectAddress = "sinkHost"
*.sourceHost.app[0].connectPort = 1000
*.sourceHost.app[0].tOpen = 0s
*.sourceHost.app[0].tSend = 0s
*.sourceHost.app[0].sendBytes = 100MiB
*.sourceHost.app[0].tClose = 0s
**.app[*].localAddress = ""
**.app[*].localPort = -1
**.app[*].sendScript = ""

```

## 2.2.5 Parametre ovplyvňujúce výsledky simulácie

Cieľom tejto sekcie je popísať všetky nastavenia, ktoré priamo ovplyvňujú výsledky simulácie. Po definícii simulovanej siete je možné realizovať úpravy, ktoré menia fyzikálne parametre v sieti. Okrem možností modifikácie parametrov sieťových prvkov má v simulácii silné zastúpenie aj samotný komunikačný kanál, ktorý je reprezentovaný pomocou modelov. Konečné výsledky simulácie sú následne po zavedení konštantných nastavení ovplyvnené náhodnými premennými, ktoré dokáže modifikovať deterministický pohľad na výsledky simulácie. V tomto kroku analýzy pôvodného scenára Rate Control Showcase dochádzalo k modifikáciám, ktoré zásadne menili povahu samotného scenára. V nižšie uvedených príkladoch sú spracované zmeny a dodatočné atribúty, ktoré boli postupne zavádzané a modifikované, s cieľom čo najširšej analýzy použiteľných modulov.

### Komunikačný kanál a prenosové médium

Štruktúra simulačného prostredia OMNeT++ je navrhnutá tak, aby bolo možné modularitou použitých modelov pokryť množstvo situácií a technických riešení, ktoré zabezpečujú reálne výsledky simulácie. Definícia bezdrôtovej siete má okrem samotných sieťových prvkov a použitých aplikačných protokolov obsiahnutú aj problematiku prenosového média. Pri bezdrôtovej komunikácii je cieľom implementovať taký model komunikačného kanálu, ktorý obsiahne čo najväčšie množstvo atmosférických a fyzikálnych zákonitostí, ktoré v reálnom svete ovplyvňujú šírenie signálu vo vzduchu. OMNeT++ považuje bezdrôtový komunikačný kanál za zdieľané fyzikálne médium, v ktorom prebieha komunikácia v sieti. Model prenosového média je neustále ovplyvňovaný komunikáciou prijímača a vysielača. Simulačný model taktiež reaguje na rušenie v prenosovom kanáli. Samotný komunikačný kanál je možné

rozdeliť na súčasti, ktoré sú reprezentované pomocou simulačných modelov. V tomto ohľade je badateľná podobnosť so skúmaným simulačným prostredím NS-3. Tieto modely sú súčasťou objektu **RadioMedium**.

**Model šírenia signálu** popisuje deje, ktoré priamo súvisia so šírením signálu v čase. Účelom modulu je zahrnúť informácie o časopriestore a polohe simulovaných zariadení, na ktoré sa nahliada ako na zdroj a prijímač signálu. V doleuvedenom príklade sa nachádza definícia modelu šírenia signálu, ktorá zahŕňa rýchlosť šírenia signálu (rýchlosť svetla) a vzdialenosť.

```
*.radioMedium.propagation.typeName = "ConstantSpeedPropagation"  
*.radioMedium.propagation.propagationSpeed = 299792458 mps
```

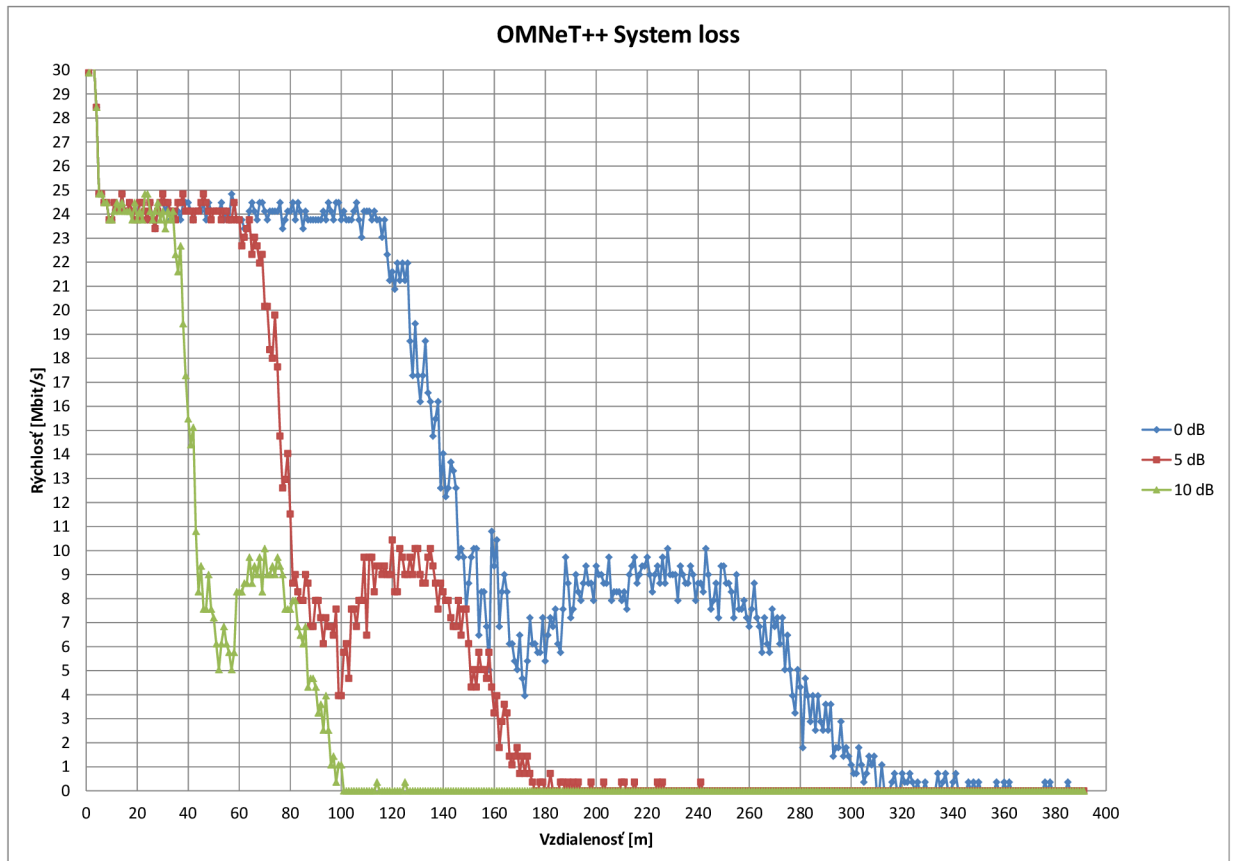
**Model strát cesty** je určený na popis energetického úbytku prenášaného signálu. Počas cesty médiom modul simuluje lom, difrakciu, odrazy a absorpciu prenášaného signálu v priestore. V odbornej literatúre sa nachádza veľké množstvo modelov straty ciest. Knižnica INET implementuje pomerne širokú škálu modelov straty ciest. Každý model je definovaný špecifickým algoritmom, ktorý priamo určuje vplyv prostredia na prenášaný signál. INET zahŕňa nasledovné algoritmy: *FreeSpacePathLoss*, *BreakpointPathLoss*, *LogNormalShadowing*, *TwoRayGroundReflection*, *TwoRayInterference*, *RicianFading*, *RayleighFading*, *NakagamiFading*. Po analýze horeuvedených modelov je možné konštatovať, že cieľom algoritmov je popis funkcie závislosti rýchlosti prenosu na vzdialenosti. Úplný výpadok komunikácie je v zásade ovplyvňovaný aj inými parametrami komunikačného kanálu. V doleuvedenom príklade sa nachádza definícia algoritmu pre simulácie v zastavaných oblastiach. Model *LogNormalShadowing* bol považovaný za najvhodnejší pre simuláciu bezdrôtových sietí. Po preštudovaní problematiky a následnom testovaní modulov strát cesty boli modifikované niektoré parametre modelu, ktoré viedli aj k úpravám dosahu bezdrôtovej siete. Získaný priebeh a prítomnosť adaptácie rýchlosti je pri použití modelu *LogNormalShadowing* najrealistickejšia. Je však nutné podotknúť, že výsledné dosahy bezdrôtovej siete boli realistické až po definovaní strát systému. Táto hodnota dokáže ovplyvniť počiatkové hodnoty simulácie tak, aby výsledné priebehy čo najviac vystihovali reálne výsledky meraní. Hodnota 5dB však bola určená len pomocou série experimentov, ktorých cieľom bolo určiť čo najvhodnejšiu hodnotu straty systému. Dá sa predpokladať, že túto hodnotu je možné získať aj pomocou meraní, ktoré sú realizované na skutočnom vybavení.

```
*.radioMedium.pathLoss.typeName = "LogNormalShadowing"
```



```
*.radioMedium.pathLoss.sigma = 1.1  
*.radioMedium.pathLoss.systemLoss = 5dB
```

V nižšie uvedenom grafe sa nachádza závislosť vplyvu strát systému na výsledný dosah simulovanej siete.



Obr. 2.6: Porovnanie vplyvu hodnôt SystemLoss v OMNeT++ pri štandarde 802.11g.

**Model strát překážek** Pri šíření signálu v simulovanom priestore je v reálnom svete častý prienik signálu cez fyzické překážky. Cieľom tohto modulu je vypočítať stratu energie na základe dráhy prejdenej v překážke a frekvencie signálu. V knižnici INET sa nachádzajú dva modely strát. *IdealObstacleLoss* zahŕňa úplnú alebo žiadnu stratu energie, na základe vypočítanej trasy signálu. *DielectricObstacleLoss* vypočíta stratu výkonu na základe presnej dielektrickej a odrazovej straty pozdĺž priamej dráhy s ohľadom na tvar, polohu, orientáciu a materiál překážajúcich fyzických objektov. V testovanom scenári však neboli prítomné žiadne překážky, ktoré by mohli zasahovať do dráhy vysielaného signálu.

**Model strát rušením z okolia** je určený na simuláciu šumu okolitého prostredia. Bežnú prevádzku komunikačného kanála ovplyvňuje tepelný šum, kozmický šum a náhodné výkyvy elektromagnetického poľa. Tento šum nepochádza z konkrétneho zdroja, preto nemá zmysel modelovať jeho šírenie v priestore. Model šumu tak popisuje len jeho zmenu v čase. V nižšie uvedenom príklade sa nachádza definícia modelu šumu a nastavenie výkonovej úrovne. Zvolená úroveň -86dBm je pomerne zásadným parametrom celej simulácie. Jej hodnota bola určená tak, aby nedochádzalo k extrémnemu rušeniu okolia. Zároveň je táto hodnota zvolená tak, aby v simulovanom prostredí boli prítomné reálne straty spôsobené rušením z okolia. Po množstve experimentov je zvolená hodnota prakticky výsledkom, ktorý čo najviac približuje získané dáta simulácie k reálnym meraniam. Je priamo viazaná na citlivosť prijímača. V nižšie uvedenom príklade sa okrem nastavenia úrovne šumu nachádza aj definícia citlivosti prijímača. Úroveň šumu a citlivosti prijímača sú zvolené tak, aby neznemožňovali komunikáciu v simulovanej sieti.

```
*.radioMedium.backgroundNoise.typeName =  
"IsotropicScalarBackgroundNoise"  
*.radioMedium.backgroundNoise.power = -86 dBm  
  
*.*Host.wlan[*].radio.receiver.sensitivity = -85dBm
```

**Model sledovania susedných prenosov** je určený predovšetkým na sledovanie susedných komunikácií. Model prepočítava množinu ovplyvnených prijímačov pre daný prenos.

**Analógový model signálov** určuje, ako sú signály reprezentované počas prenosu, šírenia a prijímania. Analógový model kombinuje parametre antény, modelovania strát a strát překážok. Jeho hlavným účelom je vypočítať prijímaný signál z vysielaného signálu. Simulačné prostredie bežne využíva analógový model, ktorý využíva reprezentáciu sily skalárneho signálu vo frekvenčnom a časovom intervale.

```
*.radioMedium.analogModel.typename = "ScalarAnalogModel"
```

**Model limitov média** sleduje určité prahové hodnoty, ktoré ovplyvňujú parametre rozsahov, výkonu a prenosovej rýchlosti. Na základe informácií z ostatných modulov v sieti je model limitov schopný definovať potrebné hraničné hodnoty. Pri definícii bezdrôtovej siete platí, že na základe údajov získaných z modelov sieťových prvkov sa vytvára sada parametrov hraničných hodnôt.

**Model vyrovnávacej pamäte** sa využíva ako vyrovnávacia pamäť všetkých výpočtov komunikácie. Simulačné prostredie podporuje tri modely vyrovnávacej pamäte: *ReferenceCommunicationCache*, *MapCommunicationCache*, *VectorCommunicationCache*.

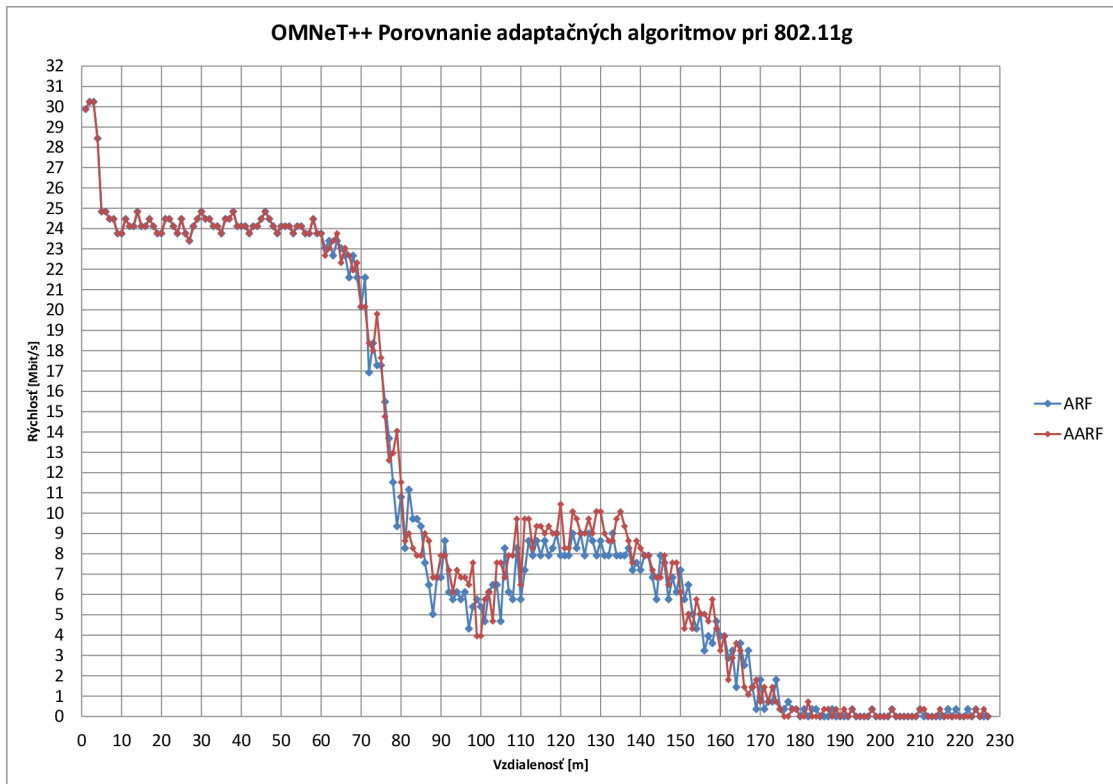
Po analýze a spustení série experimentálnych simulácií je zrejmé, že pri realizácii simulačného scenára nie je nutné manuálne definovať všetky uvedené modely. Predovšetkým Model limitov a Model vyrovnávacej pamäte neboli vzhľadom na oblasť výskumu bližšie analyzované. Ich predvolené nastavenie umožňuje dostatočnú funkčnosť simulačných scenárov.

## Adaptačný algoritmus

Jedným z parametrov, ktorý priamo ovplyvňuje výsledky simulácie je pochopiteľne aj zahrnutie adaptačného algoritmu. Komunita OMNeT++ v súvislosti so simuláciami bezdrôtových sietí 802.11 odporúča použitie algoritmov *ArfRateControl*, *AarfRateControl*, a *OnoeRateControl*. Posledný spomínaný algoritmus, *OnoeRateControl* však nebol použiteľný pri danej konfigurácii simulácie. Vzhľadom na fakt, že simulačné prostredie má pomerne veľké množstvo nedostatkov, tento algoritmus nebol ďalej skúmaný. Analýza adaptačných algoritmov bola zameraná na ostatné dva, ktoré od počiatku vykazovali funkčnosť. V nižšie uvedenom príklade sa nachádza príklad definície adaptačného algoritmu a nastavenie inicializačnej rýchlosti.

```
*Host.wlan[*].mac.*.rateControl.typename =  
${rateControl="", "AarfRateControl"}  
*.*Host.wlan[*].mac.*.rateControl.initialRate = 54Mbps
```

Použitie *ArfRateControl* a *AarfRateControl* bolo analyzované pomocou doleuvedeného grafu. Na základe výsledkov je možné konštatovať, že oba použité algoritmy mali veľmi podobné reakcie na zmeny v sieti. Získané dáta však potvrdzujú ich funkčnosť.



Obr. 2.7: Porovnanie adaptačných algoritmov v OMNeT++.

### Veľkosť rámcov a interval odosielania

Vzhľadom na pomerne širokú škálu parametrov ovplyvňujúcich výsledky simulácie bol proces analýzy zameraný predovšetkým na tie parametre, ktoré priamo súvisia s problematikou adaptačných algoritmov a dosahu bezdrôtovej siete. Po prvotnom definovaní parametrov komunikačného kanála a rádiového signálu boli získané výsledky simulácie pomerne nereálne. Dosah bezdrôtového signálu bol napriek nastaveniu reálnych výkonových úrovní neprímeraný. Prvé testovacie simulácie vykazovali dosah signálu približne 800 metrov. Táto hodnota bola považovaná za jednoznačne prehnanú. Nemožno predpokladať, že by v reálnej situácii dochádzalo k takémuto rádovo vyššiemu dosahu signálu bezdrôtovej siete. Ďalšia analýza simulačného prostredia tak bola zameraná na odstránenie tejto nezrovnalosti. Z pohľadu definície siete a jej parametrov je nutné konštatovať, že prostredie OMNeT++ ponúka predpripravené simulačné scenáre, určené na jednoduchšiu orientáciu v súčastiach a nástrojoch simulačného prostredia OMNeT++. Táto forma technickej dokumentácie však vnáša do prostredia OMNeT++ istú dávku nejasnosti. Samotnú logiku simulačných modelov je často pomerne náročné pochopiť pomocou predpripravených scenárov. Výsledky predpripravených simulácií taktiež vykazujú extrémne dosahy bezdrôtových sietí aj pri použití nižších výkonových úrovní vysielачa. Po podrobnej

analýze vplyvu nastavení na dosah bezdrôtovej siete bolo zistené, že veľkosť prenášaných správ a interval ich odosielania podstatnou mierou vplyvajú na dosah a prenosovú rýchlosť bezdrôtovej siete. Simulačné prostredie má pri použití správ s menšou veľkosťou (asi 1000B) tendenciu kalkulovať s dosahom v stovkách metrov. Táto skutočnosť je zrejme podmienená modelom strát cesty v závislosti na dĺžke trasy signálu. Je možné predpokladať, že miera chybovosti pri extrémne málo zaťaženej sieti je tak nízka, že dosah bezdrôtovej siete je rádovo väčší ako v reálnom svete. Pri definícii siete je však možné samotnú prevádzku regulovať nastavením parametrov aplikačného modulu prevádzky v bezdrôtovej sieti. Dostatočne zahľtená sieť so správami s veľkosťou asi 4000B má pri bežných verziách štandardu 802.11 dosah asi 90 metrov. Táto hodnota je porovnateľná s predpokladanými parametrami simulovanej siete. Z hľadiska nastavenia intervalu a veľkosti odosielaných dát nemožno presne špecifikovať ich potrebné parametre. V závislosti na verzii štandardu 802.11 a na definícii strát systému v modeli strát cesty je nutné hodnoty nastaviť tak, aby sa svojimi parametrami približovali k reálnej sieti. V nižšie uvedenom príklade sa nachádza definícia UDP prevádzky, ktorá zabezpečuje primeraný dosah signálu.

```
*.sourceHost.app[0].typename = "UdpBasicApp"  
*.sourceHost.app[0].destAddresses = "sinkHost"  
*.sourceHost.app[0].destPort = 5000  
*.sourceHost.app[0].messageLength = 4000B  
*.sourceHost.app[0].sendInterval = 1ms  
  
*.sinkHost.app[0].typename = "UdpSink"  
*.sinkHost.app[0].localPort = 5000
```

## Riadenie náhodných premenných v simulácii

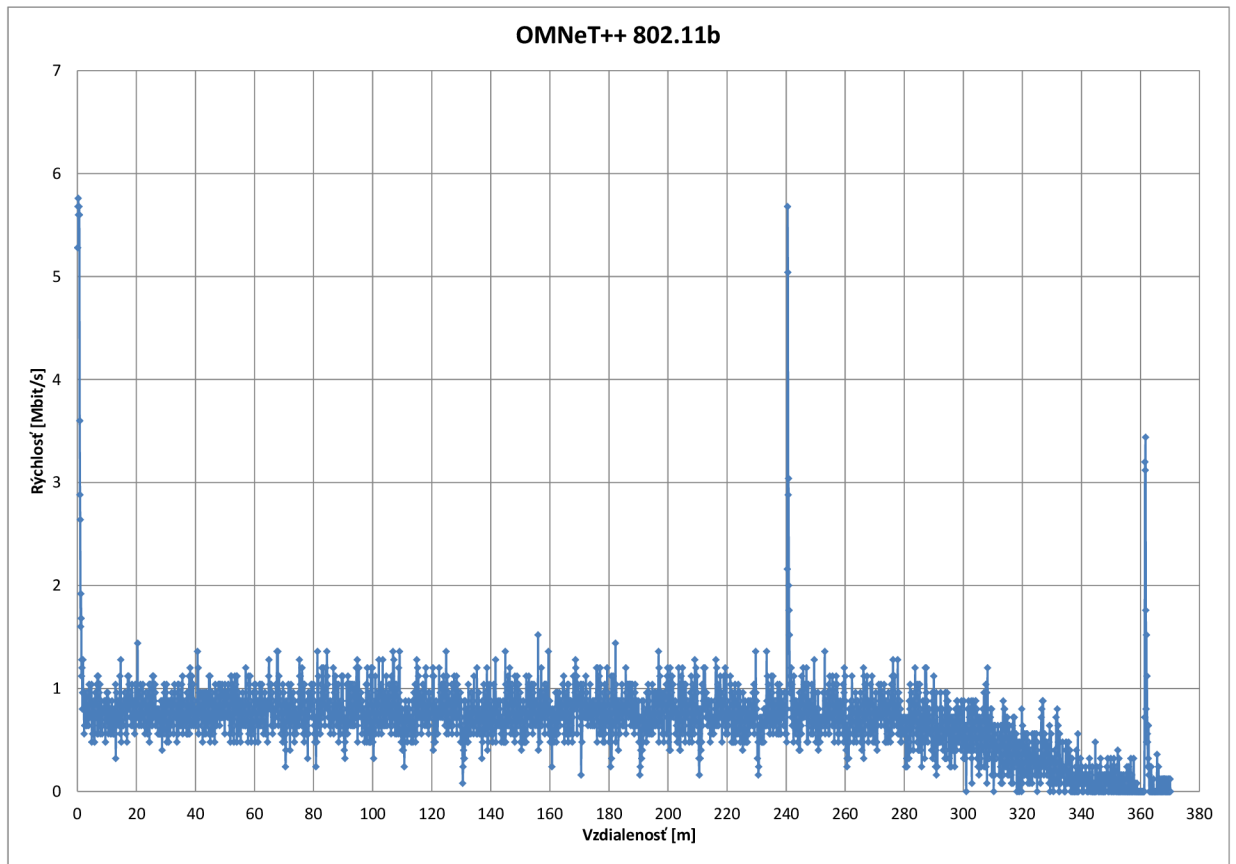
V prípade potreby nedeterministického prístupu k výsledkom simulácií prostredie OMNeT++ umožňuje definovanie hodnoty premennej, ktorá reprezentuje iníciačné semeno simulácie. Užívateľ tak má možnosť definovať pomocou nastavenia v inicializačnom súbore .ini počet náhodných čísiel a tiež definovať vstup inicializačného semena. Tieto možnosti sú vhodné pre zabezpečenie väčšieho množstva výstupov simulácie, ktoré je možné pomocou vhodnej štatistiky následne spracovať. Okrem manuálnej úpravy náhodného semena je vhodným spôsobom modifikácie deterministického výsledku aj zmena počtu náhodných premenných v simulácii. Vý-

sledné grafické zobrazenie je tak výsledkom troch po sebe idúcich simulácií s rôznymi hodnotami náhodných premenných.

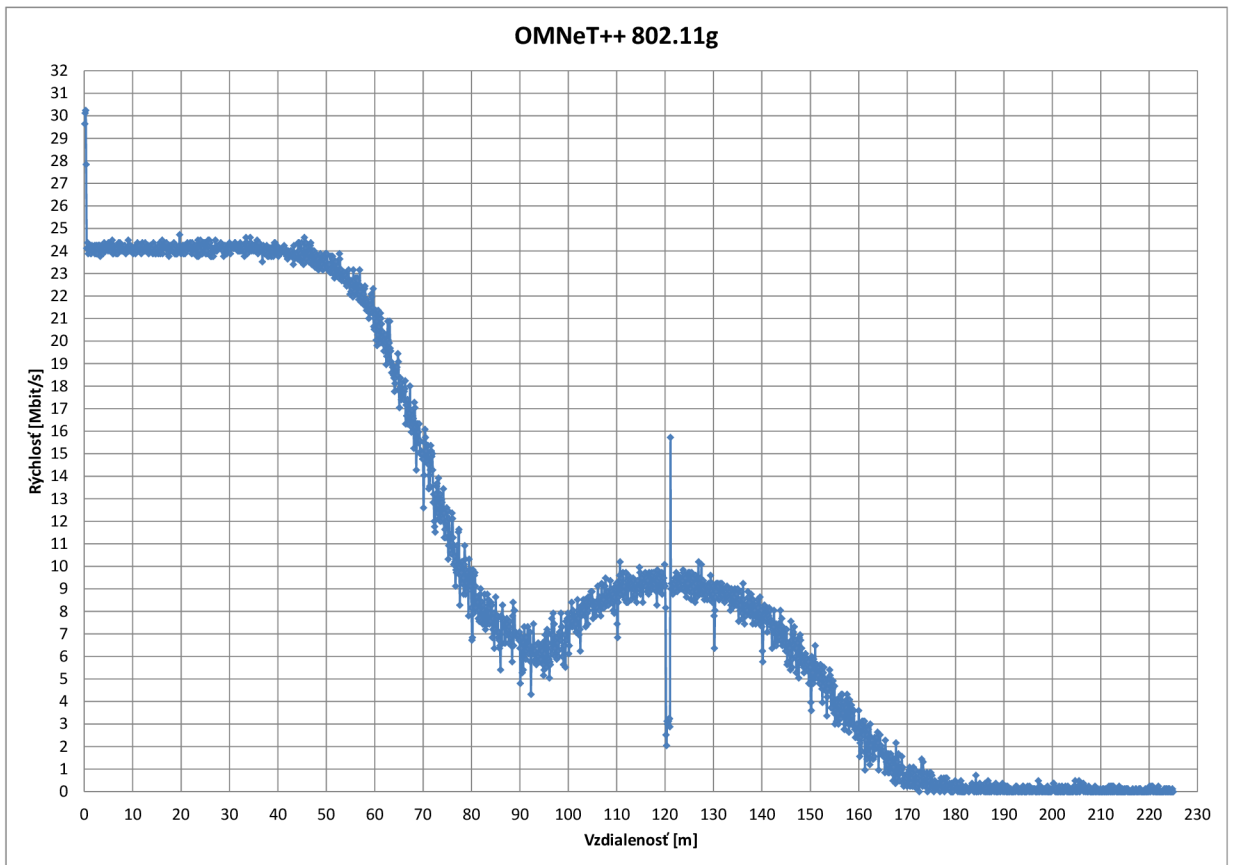
### **2.2.6 Výsledky simulácie**

Po analýze simulačného prostredia OMNeT++ bol vytvorený simulačný scenár, pomocou ktorého je možné realizovať simulácie vhodné pre porovnanie výsledkov so skutočnosťou a so simulačným prostredím NS-3. Hlavným výstupom simulácie bola závislosť vzdialenosti od rýchlosti bezdrôtovej siete. Pre účely overenia presnosti simulovaných dát boli zvolené také verzie štandardu, ktoré bolo možné odskúšať aj v iných prostrediach.

## Grafy



Obr. 2.8: Výsledok simulácie pre štandard 802.11b.



Obr. 2.9: Výsledok simulácie pre štandard 802.11g.



## 2.2.7 Vyhodnotenie simulácie

Z výsledkov simulácie je zrejmé, že prostredie OMNeT++ dokáže realizovať adaptáciu rýchlosti na základe využitia štandardných algoritmov. Za účelom porovnania bol volený algoritmus AARF, ktorý je najrozšírenejší. Tvorba simulačného scenára pozostávala z kombinácie nastavení a definícií, ktoré boli demonštrované v predpripravených scenároch. Finálna verzia simulačného scenára tak zachytáva kombináciu viacerých simulačných modelov a definícií, ktoré komunita OMNeT++ odporúča pri realizácii čo najpresnejšej simulácie.

## 2.3 Meranie skutočných parametrov

Súčasťou zadania študentskej práce je aj analýza adaptácie rýchlosti bezdrôtovej siete v reálnom prostredí. Ďalšia sekcia je zameraná na popis metódy merania a vyhodnocuje získané namerané hodnoty. Pri samotnej realizácii merania bolo nutné v prvom kroku preštudovať parametre prideleného testovaného zariadenia Cisco Linksys WRT610n v2 a následne zvoliť vhodnú konfiguráciu zariadenia. Na zabezpečenie čo najväčšej dôveryhodnosti získaných hodnôt boli realizované dve merania v rôznej lokalite a za použitia rôznych meracích nástrojov.

V oboch prípadoch boli využité bežne prednastavené parametre šírky pásma (20MHz) a použitie frekvencie 2,4GHz.

### 2.3.1 Prvé meranie

Cieľom merania je zistiť závislosť rýchlosti komunikácie na vzdialenosti stanice od prístupového bodu. Meranie bolo nutné realizovať tak, aby namerané hodnoty v čo najväčšej miere zachytávali samotný algoritmus adaptácie rýchlosti. Meranie bolo uskutočnené pre štandardy 802.11b a 802.11n.

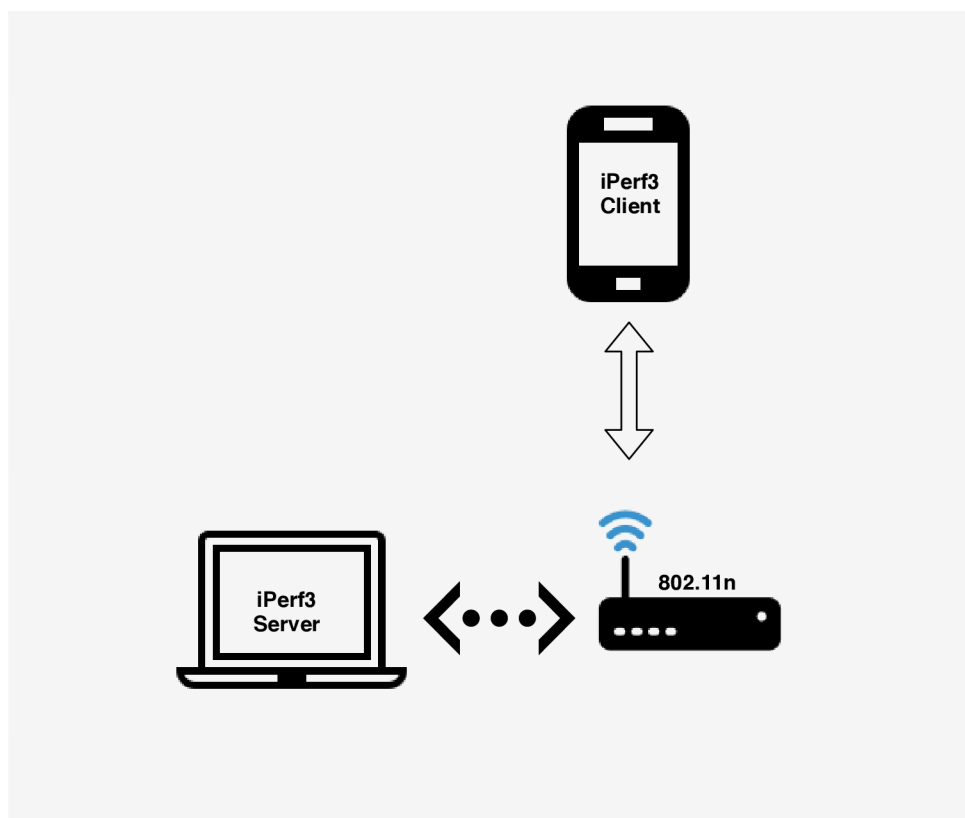
#### Výber prostredia

Cieľom merania bolo porovnanie výstupov zo sieťových simulačných prostredí so skutočnými nameranými hodnotami. Šírenie signálu bezdrôtových sietí je v skutočnosti závislé od množstva rušivých elementov nachádzajúcich sa v meranom prostredí. Tieto parametre rušenia je možné zahrnúť aj do simulačných scenárov. Cieľom práce je však posúdenie dôveryhodnosti dát súvisiacich s adaptačnými algoritmami v sieťach 802.11. Z tohto dôvodu neboli do simulačných scenárov zahrnuté žiadne objekty súvisiace s prekážkami v trase merania. Možno tak predpokladať, že pri realizácii merania na otvorenom priestranstve dôjde k eliminácii nepresností spôsobených nepresnosťami súvisiacimi s modelom fyzických prekážok v simulačnom

prostredí. Ďalším parametrom, ktorý môže spôsobiť nepresnosť nameraných dát, je interferencia

s inými zdrojmi signálu v prostredí. Na základe horeuvedených skutočností bolo zvolené voľné priestranstvo, na ktorom je čo najmenšia pravdepodobnosť ovplyvnenia parametrov. Ďalšou, pomerne podstatnou skutočnosťou, ktorá viedla k zvoleniu merania na otvorenom priestranstve, bola potreba získania dát v čo najväčšej možnej vzdialenosti stanice od prístupového bodu. Po posúdení všetkých možností využitia voľného priestranstva bola zvolená voľná plocha, ktorá je súčasťou športového areálu VUT Brno na ulici Technická 14.

### Použitú technické vybavenie a merací postup



Obr. 2.10: Zapojenie pracoviska.

Po teoretickom spracovaní problematiky bol na realizáciu samotného merania zvolený program iPerf3. Tento nástroj je možné používať v dvoch režimoch. Nástroj v režime servera je schopný reagovať na testovaciu prevádzku siete a zaznamenávať namerané hodnoty rýchlosti. Po posúdení poskytnutého technického vybavenia bolo zvolené zapojenie pracoviska, ktoré pozostáva z bezdrôtového smerovača, ktorý je priamo prepojený s prenosným osobným počítačom pomocou metalického kábla.

Osobný počítač pripojený cez ethernet k smerovaču zastával funkciu iPerf3 servera, ktorý nepretržite čakal na podnet na porte 5001 (TCP). Smerovač tak mohol dotazy z bezdrôtovej siete priamo presmerovať na osobný počítač so spusteným serverom iPerf3. Do vytvorenej bezdrôtovej siete bol následne pripojený mobilný telefón s operačným systémom Android. Toto zariadenie je schopné po inštalácii nástroja iPerf3 zastávať funkciu meracieho zariadenia, ktoré produkuje testovacie dotazy na iPerf3 server.

Ďalším potrebným vybavením meracieho pracoviska bolo meracie pásmo, ktoré umožňovalo presné zaznamenanie vzdialenosti stanice od prístupového bodu. Samotné meranie prebiehalo opakovanými sekvenciami testovacej komunikácie v bezdrôtovej sieti, ktorých výsledkom bol údaj o rýchlosti komunikácie v sieti. Táto hodnota bola zaznamenaná spolu s príslušnou vzdialenosťou stanice od prístupového bodu. Krok merania bol počas merania upravovaný podľa stavu v sieti. Cieľom bolo zaznamenanie čo najväčšieho počtu údajov použiteľných pri vyhodnotení merania. Ukončeniu merania predchádzal úplný výpadok komunikácie a strata signálu spôsobená prekročením vzdialenosti dosahu signálu použiteľného pre komunikáciu v sieti.

### **2.3.2 Druhé meranie**

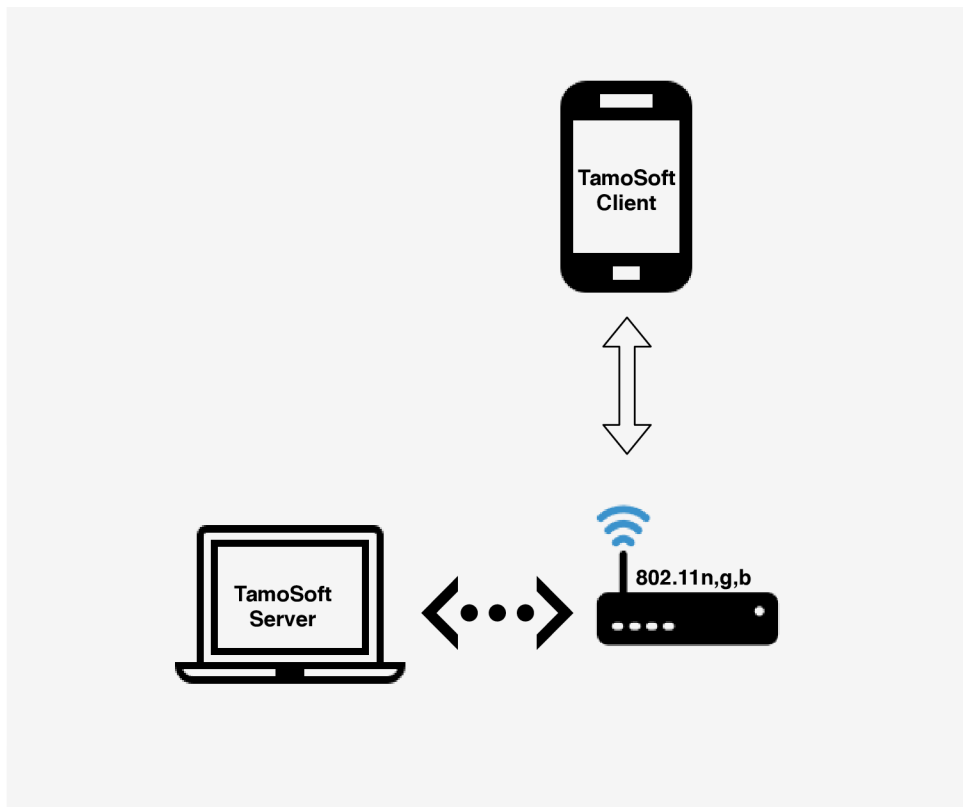
Cieľom druhého merania bolo opakovať meranie pre možnosť porovnania získaných dát. Meranie bolo realizované tak, aby postup merania kopíroval prvé meranie. Základným rozdielom bolo využitie odlišného nástroja pre meranie priepustnosti siete. Taktiež došlo k zmene zariadení v sieti. Meranie bolo uskutočnené pre štandardy 802.11b, 802.11g a 802.11n. Štandard 802.11g bol zvolený z dôvodu snahy zaznamenať adaptáciu rýchlosti pri starších štandardoch, ktoré však podporujú vyššie prenosové rýchlosti.

#### **Výber prostredia**

Vzhľadom na vopred určený cieľ a merací postup bol v prípade druhého merania kladený dôraz na zabezpečenie čo najmenej rušeného prostredia. Na základe simulácií a orientačných meraní bolo možné očakávať, že dosah bezdrôtového signálu veľkej miere závisí od zvoleného prostredia. Otvorené priestranstvo v zastavanej oblasti môže často vykazovať parametre rušenia, ktoré možno očakávať v uzavretých priestoroch s veľkým množstvom rušivých elementov. Vzhľadom na to, že adaptačné algoritmy primárne sledujú chybovosť v sieti, je možné predpokladať, že na testovanie adaptácie rýchlosti je nutné zvoliť také prostredie, ktoré má homogénne vlastnosti a jeho parametre možno vopred definovať. Vzhľadom na tento fakt bola zvolená

lokalita poľného letiska so súvislou asfaltovou pristávacou plochou v katastri obce Komoča v Slovenskej republike. V čase merania sa na letisku nenachádzali žiadne rádiolokačné ani rádiokomunikačné zariadenia, ktoré by mohli ovplyvniť výsledky merania.

### Použité technické vybavenie a merací postup



Obr. 2.11: Zapojenie pracoviska.

Pri druhom meraní došlo k zmenám, ktorých cieľom bolo v čo najväčšej možnej miere eliminovať nepresnosti spôsobené chybou merania a nepresnosťou technického vybavenia. Podobne ako v prvom meraní bol použitý bezdrôtový smerovač Cisco Linksys WRT610n, ktorý bol prepojený s osobným počítačom pomocou metalického kábla. Na rozdiel od prvého merania bol pre realizáciu merania použitý program TamoSoft Throughput Test, ktorý bol spustený vo verzii server na osobnom počítači. Tento server neustále načúval na porte 27100(TCP/UDP). Do bezdrôtovej siete bol zapojený mobilný telefón s operačným systémom Android. Toto zariadenie po inštalácii nástroja TamoSoft Throughput Test Client zastávalo funkciu meracieho zariadenia, ktoré produkuje a zaznamenáva testovaciu prevádzku. Zavedeniu nástroja TamoSoft Throughput Test predchádzalo porovnanie nameraných výsledkov s použitým nástrojom iPerf3. Je nutné konštatovať, že namerané hodnoty oboch

nástrojov sú pri rovnakom nastavení prakticky zhodné. TamoSoft Throughput Test však ponúka spoľahlivejší priebeh meracej sekvencie a získané údaje obsahujú ucelený súbor výsledkov pre protokoly TCP aj UDP. Z užívateľského hľadiska bol teda tento nástroj považovaný za vhodnejší. Je však nutné podotknúť, že toto rozhodnutie je možné chápať ako subjektívny postoj autora, ktorý mal za cieľ eliminovať hrubé chyby merania. Náhodné chyby merania, spôsobené kolísaním rušivých vplyvov, bolo možné odstrániť použitím priemeru nameraných hodnôt za určitú sekvenciu. V tomto ohľade TamoSoft Throughput Test aj iPerf3 ponúkajú rovnakú mieru eliminácie náhodných chýb. Bezdrôtový smerovač bol napájaný z akumulátora, ktorý bol inštalovaný v motorovom vozidle.

Bezdrôtový smerovač bol umiestnený približne jeden meter nad úrovňou terénu. Ďalším vybavením meracieho pracoviska bolo meracie pásmo, ktoré umožňovalo presné zaznamenanie vzdialenosti od prístupového bodu. Nameraná hodnota rýchlosti prenosu bola priemerom sekvencie výmeny dát pomocou komunikačných protokolov TCP a UDP v oboch smeroch. K tejto zistenej hodnote bola zaznamenaná hodnota vzdialenosti od prístupového bodu. Krok medzi jednotlivými meraniami bol volený na základe stavu komunikácie v sieti. Snahou bolo zaznamenať podrobne skokové zmeny v sieti. Okrem spomínaných hodnôt bola orietračne sledovaná aj hodnota chybovosti a časový interval vzájomnej výmeny dát. Ukončeniu merania predchádzala úplná strata signálu a nemožnosť opätovného nadviazania komunikácie. Meranie bolo realizované pre štandardy 802.11g, b a n.

### **2.3.3 Výsledky meraní**

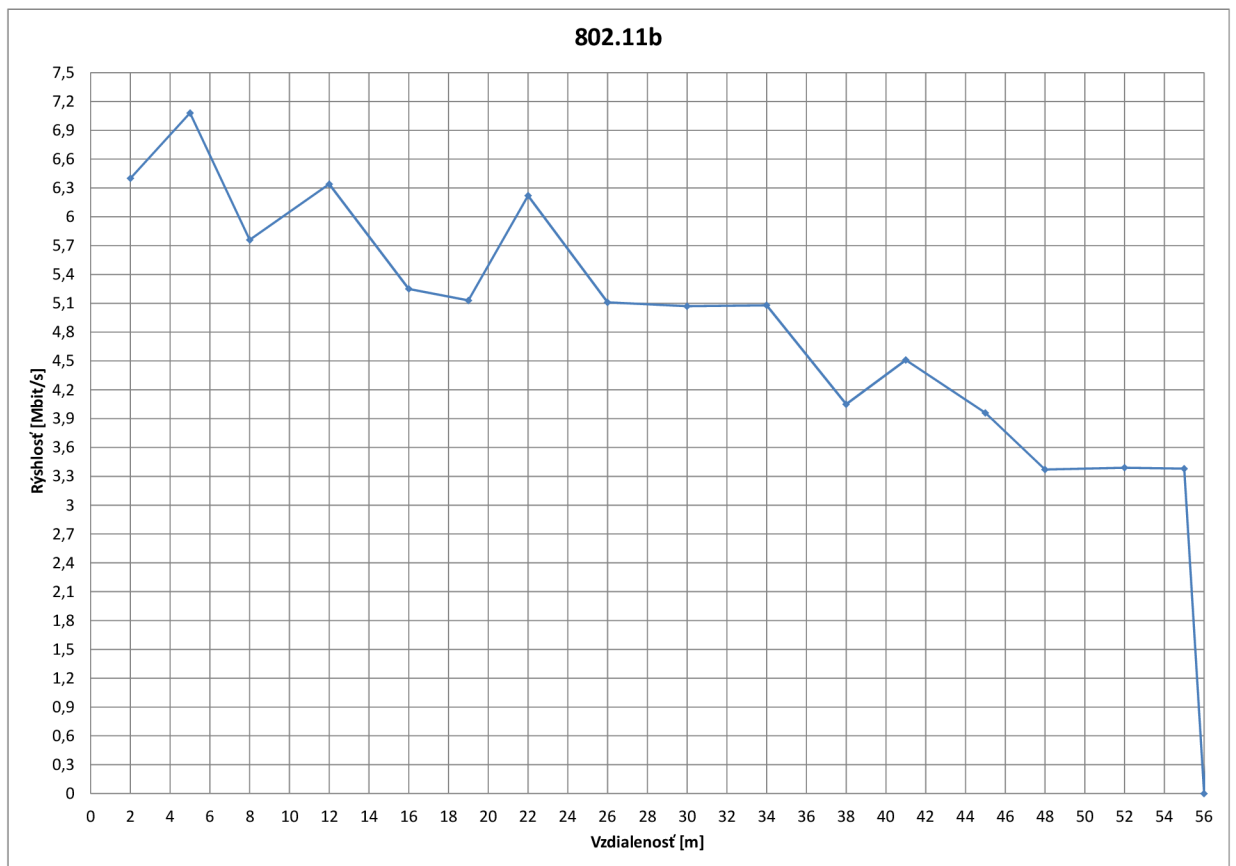
Po ukončení procesu merania boli zaznamenané údaje graficky spracované. Hodnoty získané pri meraní boli posúdené a následne porovnané s výsledkami simulácie.

#### **Vyhodnotenie prvého merania**

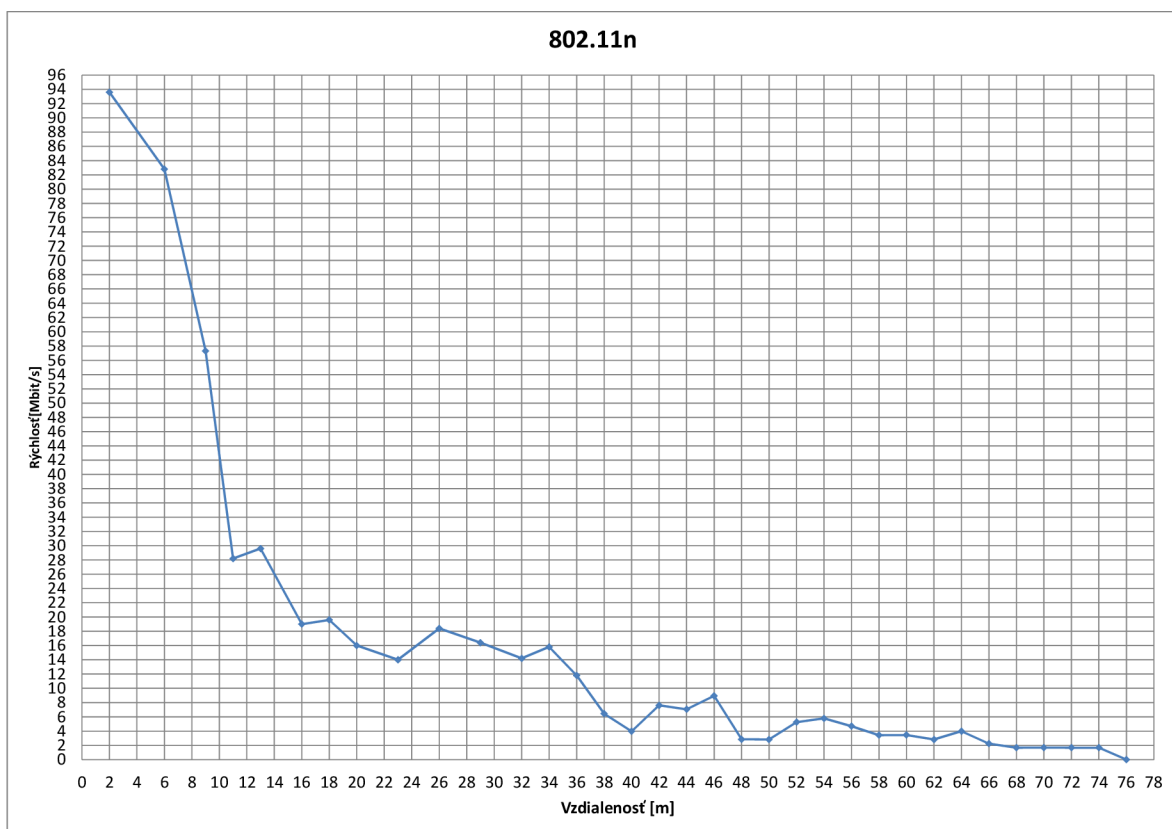
Na základe získaných dát je možné konštatovať, že v prípade prvého merania bola zreteľne badateľná tendencia postupného poklesu prenosovej rýchlosti pri postupnom vzdalovaní sa od prístupového bodu. V prípade prvého merania štandardu 802.11b došlo v druhej polovici merania ku krátkemu výpadku. Testovacia relácia nástroja iPerf3 bola následne obnovená a meracia sekvencia pokračovala ďalšími meraniami. Vzhľadom na zachovanie kontinuity a postupnosti bol tento výpadok evidovaný ako záznam o krátkodobom výpadku, ktorý spôsobil sekvenciu desiatich nulových meraní na danej vzdialenosti. Toto meranie bolo uskutočnené v zastavanej mestskej oblasti. Je možné predpokladať, že boli prítomné veľké rušivé vplyvy, ktoré majú za následok pomerne nízky dosah bezdrôtovej siete. Toto meranie možno považovať

za orientačné. Získané hodnoty boli použité predovšetkým na posúdenie relevancie simulácií a zoznámenia sa s problematikou merania prenosovej rýchlosti.

### Grafy prvého merania



Obr. 2.12: Meranie štandardu 802.11b.



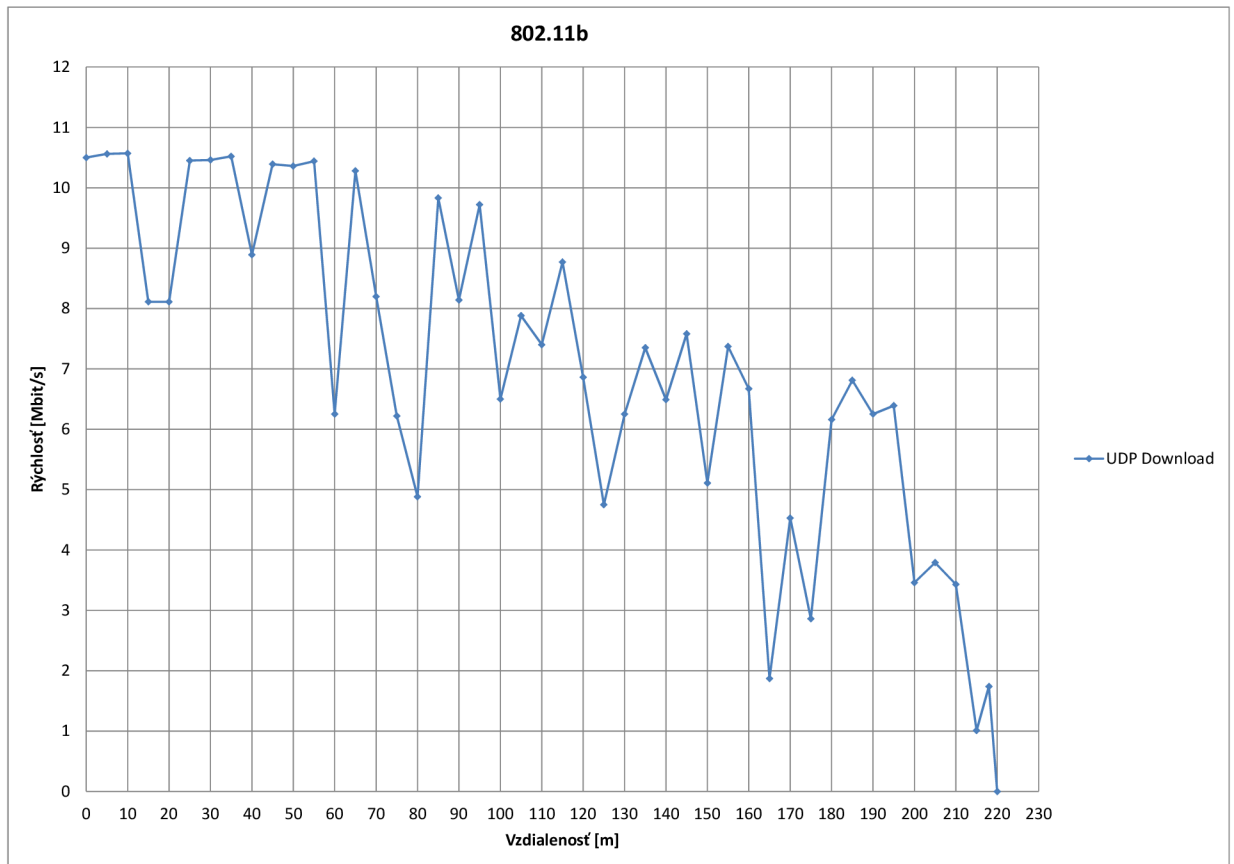
Obr. 2.13: Meranie štandardu 802.11n.

## Vyhodnotenie druhého merania

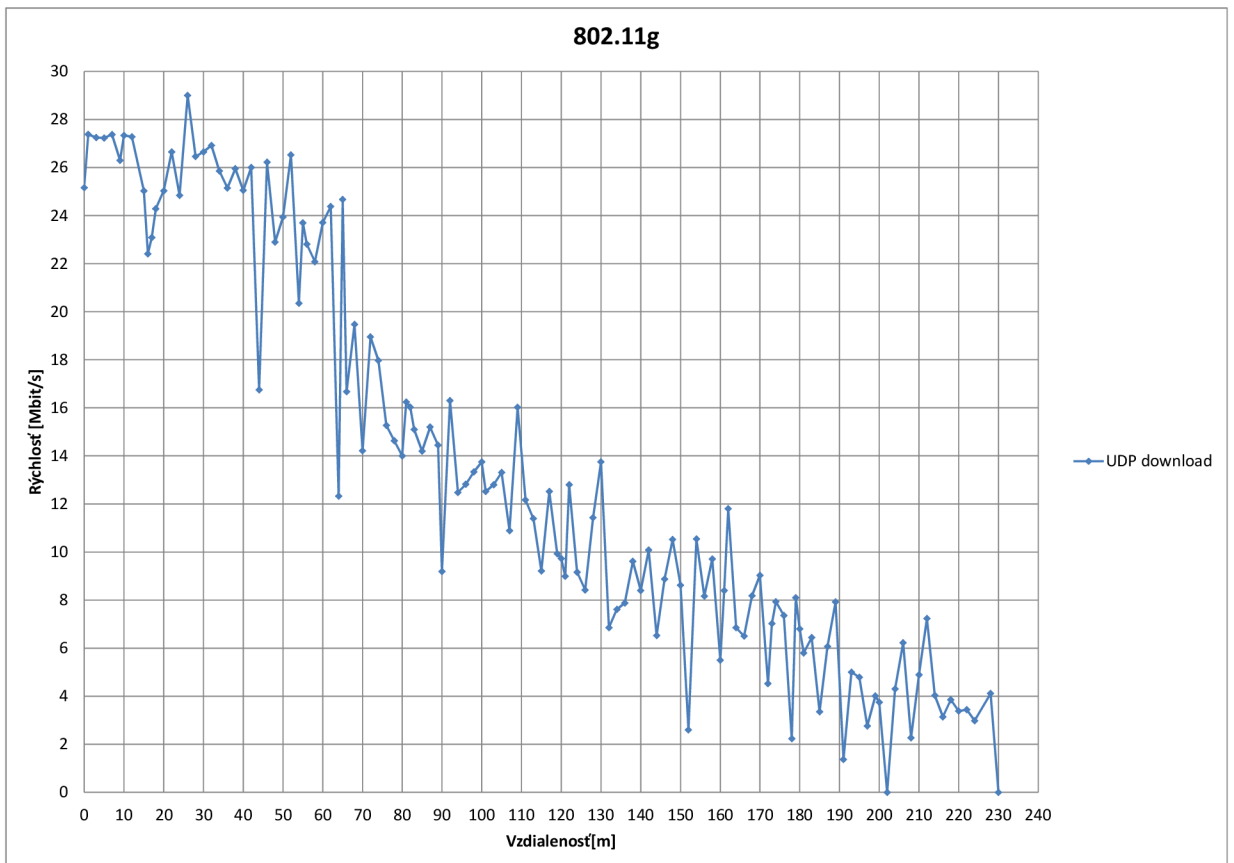
Vzhľadom na fakt, že druhé meranie nebolo možné vykonať v rovnakej lokalite ako prvé meranie, nebolo cieľom ďalej skúmať rušivé vplyvy a chyby, prítomné pri prvom meraní. Druhé meranie bolo realizované v lokalite bez zastavaných plôch. Rušivé vplyvy v tomto prostredí možno považovať za atmosférický šum, prípadne rušenia zo vzdialených zdrojov, ktoré sa nedajú bližšie lokalizovať. Pri posudzovaní použitého vybavenia a softvérových nástrojov je možné považovať druhé meranie za dokonalejšie. Hodnoty získané z druhého merania sú vzhľadom na zvolené prostredie reálne. V prípade všetkých troch štandardov boli pri druhom meraní zistené skokové zmeny prenosovej rýchlosti. Tieto dáta je možné porovnávať s výsledkami simulácií. Samotnému meraniu predchádzalo podrobnejšie skúmanie rušivých vplyvov a prípadných nedostatkov spojených s použitým vybavením. Počas druhého merania boli okrem zaznamenávaných hodnôt prenosovej rýchlosti a vzdialenosti sledované aj hodnoty chybovosti. Táto veličina sa v čase rýchlo menila a nebolo možné zaznamenať konkrétne hodnoty. Pri posudzovaní chybovosti a prenosovej rýchlosti je jednoznačné, že úprave prenosovej rýchlosti na nižšiu sadzbu vždy predchádzalo zaznamenanie nadmernej chybovosti prenosu. Zistené fakty potvrdzujú očakávanie, že zníženie prenosovej rýchlosti je v rámci adaptačného algoritmu podmienené detekciou zvýšenia strát rámcov v sieti. Druhé meranie obsahuje ucelený súbor získaných dát, ktoré je možné porovnávať so simuláciami.



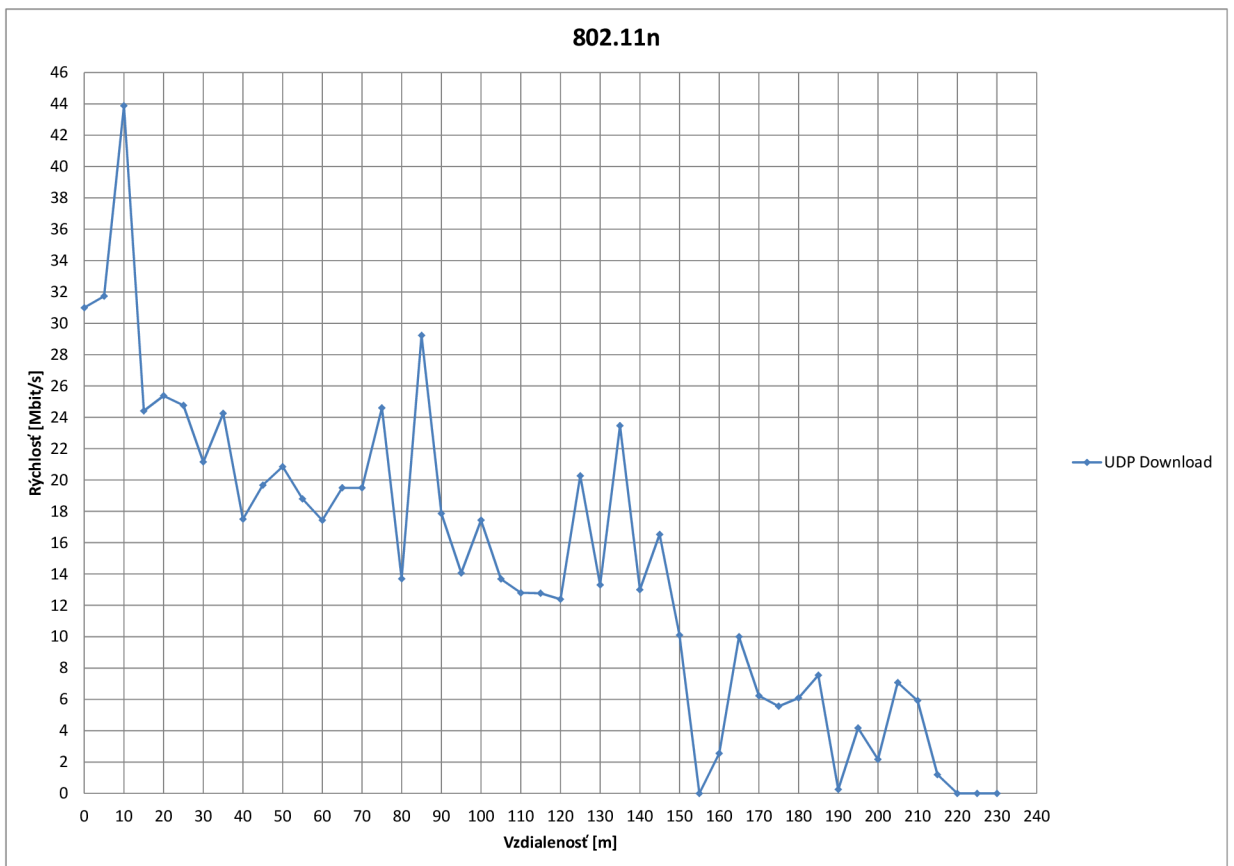
## Grafy druhého merania



Obr. 2.14: Druhé meranie štandardu 802.11b.



Obr. 2.15: Druhé meranie štandardu 802.11g.



Obr. 2.16: Druhé meranie štandardu 802.11n.

## 2.4 Vyhodnotenie výsledkov

Hlavným cieľom študentskej práce bolo porovnanie dát získaných z merania závislosti rýchlosti prenosu na vzdialenosti od prístupového bodu a dát, získaných zo simulácií v programe NS-3. Táto sekcia sa zaoberá porovnaním získaných dát a posúdením validity samotných dát. Sekcia taktiež popisuje výhody a nevýhody použitého riešenia.

### 2.4.1 Posúdenie simulácií v NS-3

Ako už bolo v tejto práci popísané, vytváranie simulácií v prostredí NS-3 je značne závislé od schopnosti autora využiť voľne prístupné programové súčasti, ktoré podľa dokumentácie zahrňujú potrebné parametre pre realizáciu simulácie. Použitý simulačný scenár bol zvolený na základe prvotného posúdenia získaných výsledkov.

V prípade použitia odporúčaných tried `MinstrelHtWifiManager` boli získané dáta pri štandarde 802.11n validné a boli považované za dostatočné. Hlavným nedostatkom použitého scenára je problematickosť implementácie adaptácie rýchlosti v štandardoch 802.11g a 802.11b. Simulačný scenár bol navrhnutý tak, aby bol schopný simulovať rôzne štandardy a umožniť použitie rôznych tried skupiny `WifiManager`. Na základe prístupnej dokumentácie boli vybrané viaceré triedy, zahŕňajúce adaptáciu rýchlosti pre štandardy 802.11g a 802.11b. Výsledky simulácie však ani v jednom prípade nenaznačovali funkčnosť adaptačných algoritmov pri štandarde 802.11b. Štandard 802.11g však zreteľne vykazoval funkčnosť adaptačných algoritmov. Vzhľadom na veľký počet doporučených modulov a tried použiteľných na realizáciu simulácií bezdrôtovej siete nebolo možné zaznamenávať všetky kombinácie nastavení a použitých tried. Je však možné konštatovať, že výsledky sú vo veľkej miere podobné. Ďalším skúmaným parametrom simulácie bol simulovaný komunikačný kanál. Pri základnej konfigurácii simulačného scenára bol prednastavený komunikačný kanál, ktorý reprezentuje normálne podmienky v simulovanom prostredí. Cieľom skúmania tohto objektu bolo posúdenie vplyvu modulov zahrňujúcich rušenie v komunikačnom kanáli na výsledky simulácie. Všetky skúmané riešenia simulovali rušenie

a interferenciu signálov z iných zdrojov v komunikačnom kanáli. Na základe výsledkov simulácie je evidentné, že zahrnutie ďalšieho rušenia spôsobuje značný útlm signálu a znemožňuje komunikáciu vo väčších vzdialenostiach. Získané výsledky simulácie boli najreálnejšie pri simulovaní štandardu 802.11n a 802.11g pri normálnych podmienkach komunikačného kanála. Výkon vysielača taktiež aktívne vplýval na výsledky simulácií. Pri nastavení výkonu sieťových prvkov, ktorý korešponduje s reálnymi hodnotami boli získané reálne výsledky. Poslednou skúmanou súčasťou

simulačného scenáru bol druh komunikácie. Získané výsledky naznačujú, že rozdiel medzi použitým protokolom UDP alebo TCP spôsobuje len rovnomerný pokles rýchlosti v prípade použitia protokolu TCP. Na základe výsledkov simulácií je však zrejmé, že prostredie NS-3 obsahuje prvky zahrňujúce aj adaptačné algoritmy pre bezdrôtové siete. Je možné konštatovať, že hlavnou nevýhodou prostredia NS-3 je pomerne komplikovaná štruktúra skriptov, ktorá môže užívateľovi komplikovať vytváranie scenárov. Po získaní dostatočných skúseností s prostredím NS-3 je však možné vytvoriť simulácie, ktoré sú porovnateľné so skutočnosťou. Akékoľvek úpravy konfigurácie siete boli sprevádzané podrobnou teoretickou prípravou a následným testovaním. Je zrejmé, že prostredie NS-3 vyžaduje dlhodobé a postupné štúdium problematiky. Výsledné grafické závislosti však potvrdzujú dostatočnú implementáciu adaptačných algoritmov.

## 2.4.2 Posúdenie simulácií v OMNeT++

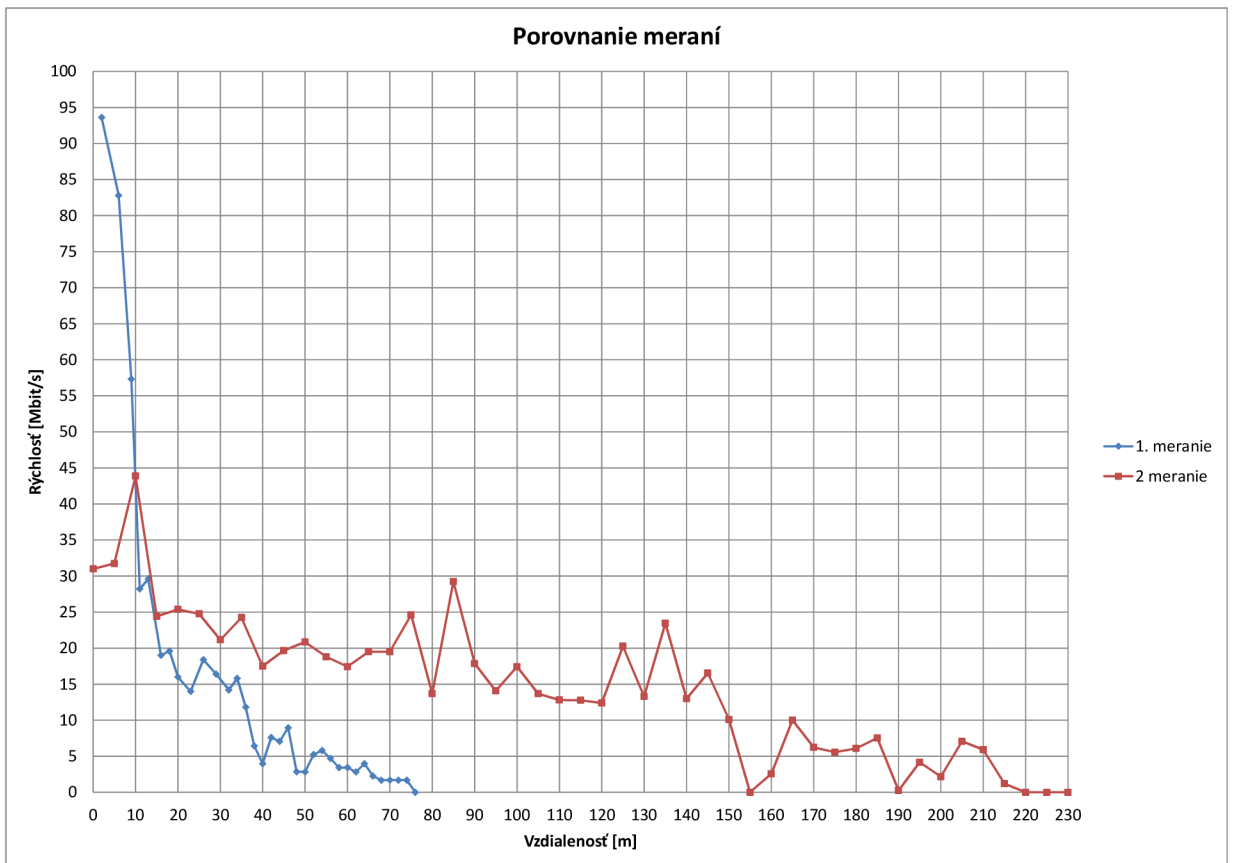
Vzhľadom na zadanie práce, bolo nutné analyzovať simulačné prostredie OMNeT++ z pohľadu bežného užívateľa s cieľom vytvoriť simulačný scenár demonštrujúci adaptáciu rýchlosti v bezdrôtových sieťach. Po inštalácii a spustení simulačného prostredia OMNeT++ bolo možné konštatovať, že tento nástroj ponúka prehľadné a užívateľsky prívetivé prostredie pre vytváranie simulácií sietí. Graficky znázornené prvky siete a definícia parametrov formou grafických formulárov sú veľmi užitočnou vlastnosťou, ktorá umožňuje prehľadné zoznámenie sa so simulačným prostredím. Tvorba skriptov je značne závislá na znalostiach užívateľa. Veľkou výhodou prostredia OMNeT++ je však snaha autorov o minimalizáciu potrebných prerekvizít užívateľa. Užívateľ by mal byť schopný pochopiť a následne vytvoriť definíciu simulovanej siete aj bez predošlého štúdia programovacieho jazyka, prípadne analýzy samotnej štruktúry simulačného prostredia. Hlavnou nevýhodou tohto prostredia je pomerne neprehľadná dokumentácia, ktorá sa skladá predovšetkým z názorných ukážok simulácií. Užívateľ je nútený analyzovať hneď niekoľko simulačných scenárov a dbať na dôsledné štúdium zvolených simulačných modelov. Pri volení simulačných modelov užívateľ často naráža na chyby v zdrojovom kóde, ktoré sú zrejme spôsobené nedostatočným vývojárskym zastúpením v posledných rokoch. Vytvorenie simulačného scenára určeného na demonštráciu adaptačných algoritmov v bezdrôtovej sieti bolo sprevádzané množstvom ťažkostí, ktoré boli spôsobené predovšetkým neznalosťou problematiky a slabou technickou dokumentáciou. Po vytvorení funkčného simulačného scenára je na mieste diskusia, či sú získané výsledky skutočne výsledkom simulácie, alebo sú výsledkami, ktoré boli získané pri snahe priblížiť sa k očakávaným parametrom siete.

### 2.4.3 Posúdenie merania

Výsledky merania na otvorenom priestranstve potvrdili predpokladané parametre bezdrôtovej siete. Problematika merania rýchlosti v bezdrôtových sieťach je však pomerne rozsiahla. Pri meraní rýchlosti je nutné zohľadniť fakt, že získané namerané hodnoty sú výstupom softvéru, určeného na meranie rýchlosti v sieťach. Je možné predpokladať, že posudzované dáta zahŕňujú rýchlosť siete na aplikačnej vrstve. Použitie rôzneho softvérového a hardvérového vybavenia tak vnáša

do výsledkov merania istú neurčitost, ktorá je vzhľadom na charakter skúmaných veličín zanedbateľná. Pri voľbe vhodného priestoru bolo zvolené otvorené priestranstvo, ktoré umožňovalo uskutočňovať merania pri väčších vzdialenostiach od prístupového bodu. Dáta získané z meraní je možné považovať za dostatočné. Zariadenie Cisco Linksys WRT610n v2 počas merania neustále upravovalo prenosovú rýchlosť na základe zmeny vzdialenosti od prístupového bodu. Vplyv prostredia je badateľný predovšetkým pri sledovaní dosahu siete. Vo všeobecnosti je dosah v zhoršených podmienkach približne 70 metrov. V prípade druhého merania sa dosah bezdrôtovej siete pohyboval na úrovni asi 220 metrov. Použitie nástroja TamoSoft Throughput Test sa pri druhom meraní odzrkadlilo na množstve získaných dát. Taktiež tento nástroj zaznamenal reálnejšie výsledky, ktoré sú porovnateľné s očakávanými hodnotami.

Z pohľadu adaptačných algoritmov bolo pri meraniach možné sledovať prakticky neustálu zmenu parametrov siete, ktoré spôsobujú aj zmenu sadzieb prenosovej rýchlosti. V reálnom čase je kondícia siete značne premenlivá. Získané priemerné hodnoty spravidla vychádzajú zo série meraní na danej vzdialenosti od prístupového bodu. Výsledky druhého merania boli použité pri porovnávaní s výsledkami simulačných prostredí.

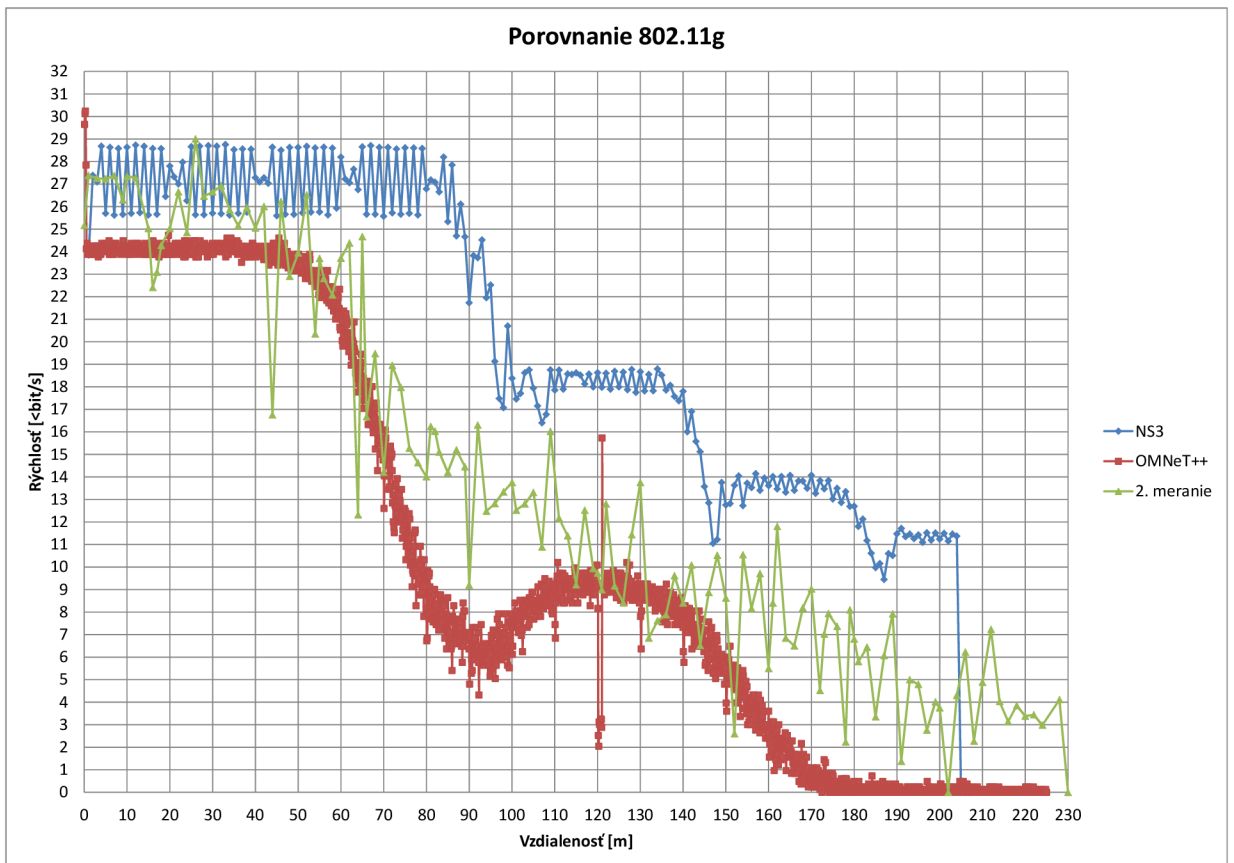


Obr. 2.17: Porovnanie meraní pre štandard 802.11n.

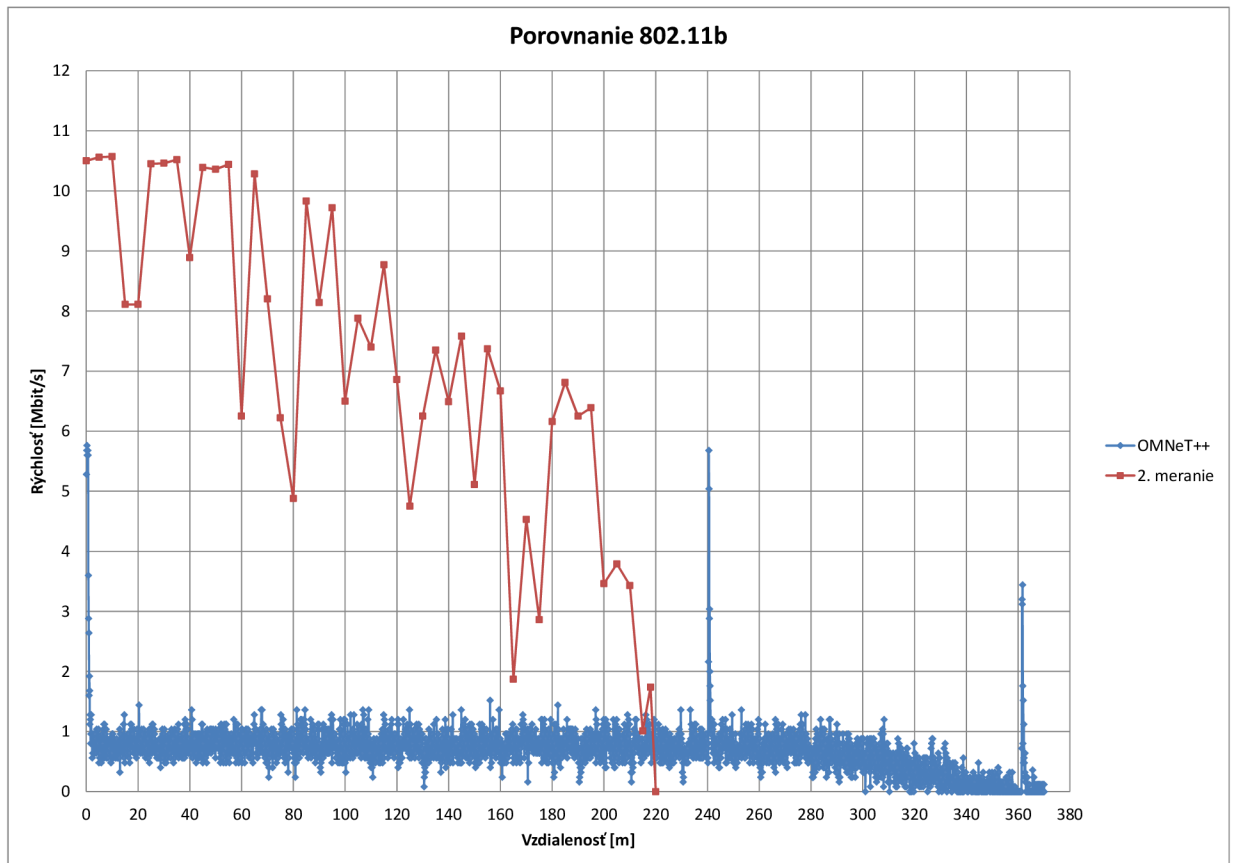
## 2.4.4 Porovnanie výsledkov merania a simulácie

Po posúdení získaných dát bolo možné vytvoriť grafické porovnanie nameraných hodnôt rýchlosti s dátami získanými pomocou simulácií v prostrediach NS-3 a OMNeT++. Pre potreby porovnania získaných dát bol vybraný štandard 802.11g. Dáta získané zo simulácií štandardu 802.11b a 802.11n neboli považované za porovnateľné pre všetky skúmané časti, vzhľadom na nízke prenosové rýchlosti v prípade štandardu 802.11b a problémom so simuláciami štandardu 802.11n v prostredí OMNeT++. Získané dáta však boli znázornené pre porovnanie s výsledkami druhého merania. V prípade štandardu 802.11g sú dáta porovnateľné. Z hľadiska podobnosti simulácií a výsledkov druhého merania je zjavné, že napriek snahám o čo najdokonalejšie simulácie bezdrôtových sietí je pri NS-3 a OMNeT++ výsledný priebeh omnoho menej skokový, ako pri skutočnom meraní. Adaptácia rýchlosti je v prípade simulačných prostredí zreteľne rozdelená na tri úrovne. V prípade nameraných hodnôt z druhého merania je výsledná krivka omnoho nesúmernejšia a počas celého priebehu sú viditeľné výkyvy, ktoré v simulačných prostrediach nie sú tak extrémne. Z hľadiska snahy o čo najrealistickejšiu simuláciu je však nutné konštatovať, že obe simulačné prostredia dokážu odhadnúť hodnotu prenosovej rýchlosti, ktorá by v reálnej situácii mohla byť zvolená. Použitý hardvér a samotný model smerovača značne ovplyvňuje získané hodnoty. Dá sa predpokladať, že získané priebehy z reálneho merania by boli odlišné pri použití iného modelu smerovača. Ďalším, pomerne podstatným faktorom je, že zariadenie Cisco Linksys WRT610n nemá vo svojej technickej špecifikácii jednoznačne popísaný adaptačný algoritmus. Musíme sa tak obmedziť na konštatovanie, že Cisco Linksys WRT610n má v sebe nejakou formou implementovaný adaptačný algoritmus, sledujúci chybovosť rámcov a odstup signálu od šumu. Adaptačné algoritmy sa však v každej zo skúmaných častí nachádzajú. Simulácie sú v oboch prípadoch značne modifikovateľné a na rozdiel od skutočného merania vystavované rôznym náhodným výkyvom. Výsledky simulácie v NS-3 majú zo všetkých skúmaných častí najvyššie sadzby prenosovej rýchlosti. Naopak, prostredie OMNeT++ pri danej konfigurácii zvolilo nižšie sadzby prenosovej rýchlosti. Z hľadiska snahy o čo najrealistickejšiu simuláciu sú výsledky simulácií z prostredia OMNeT++ dokonalejšie.

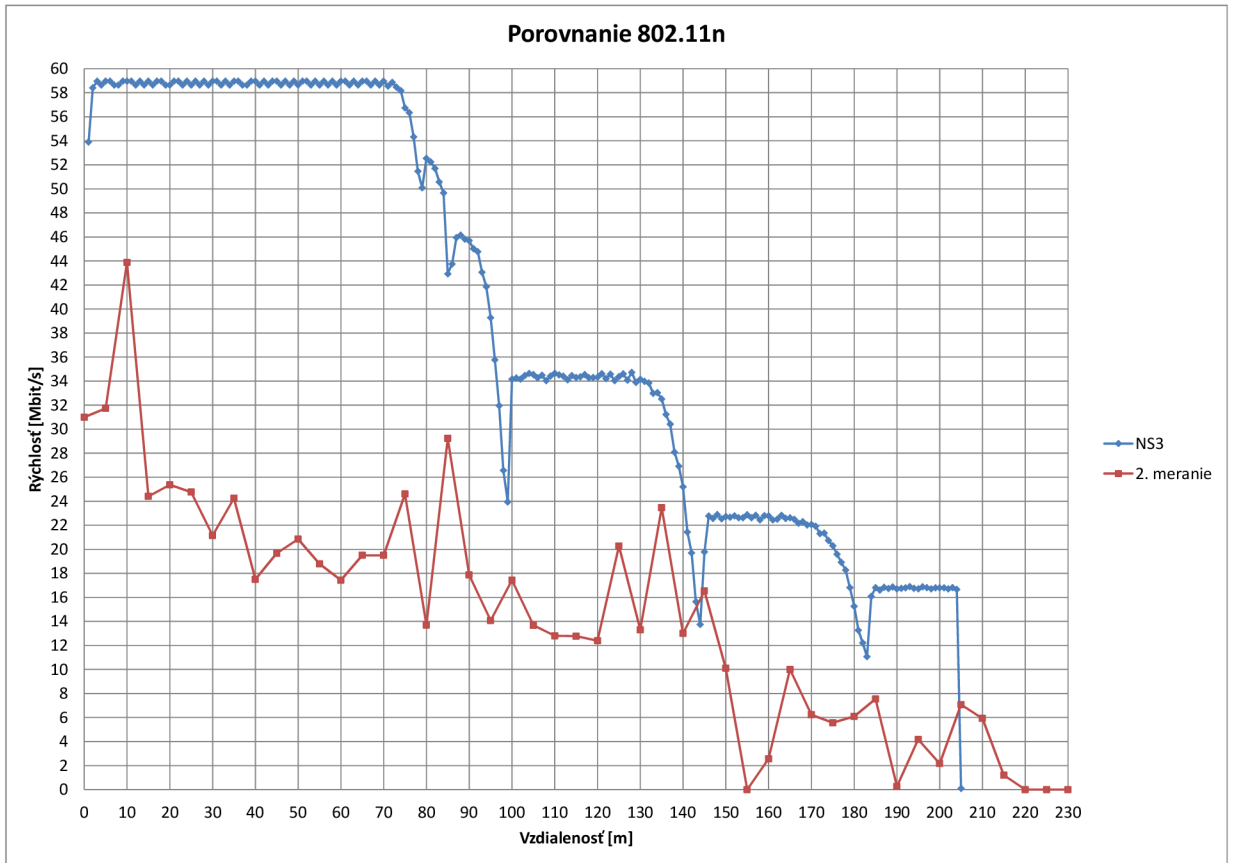




Obr. 2.18: Porovnanie výsledkov simulácií a druhého merania.



Obr. 2.19: Porovnanie pre štandard 802.11b.



Obr. 2.20: Porovnanie pre štandard 802.11n.

## 2.4.5 Porovnanie simulačných prostredí NS-3 a OMNeT++

Vzhľadom na vopred určený cieľ porovnať simulačné prostredia NS-3 a OMNeT++, bol od počiatku práce kladený dôraz na čo najdetailnejšie porovnanie spomínaných simulačných prostredí. Z užívateľského hľadiska je možné obe prostredia posudzovať od inštalácie až po konkrétne adaptačné algoritmy. Ak porovnáваме simulačné prostredia z hľadiska intuitívnosti a prehľadnosti, je v tomto ohľade dokonalejšie simulačné prostredie OMNeT++. Jeho samotná inštalácia a spustenie prvotných testovacích simulácií nevyžadujú extrémnu teoretickú prípravu. Získanie grafických výsledkov simulácií je už náročnejšie. Po štúdiu prístupnej dokumentácie je však tento proces bezproblémový a prostredie OMNeT++ dokáže v pomerne krátkom čase užívateľovi ponúknuť dostatočný priestor na realizáciu cieľových simulačných scenárov. Prostredie NS-3 je v tomto ohľade omnoho komplikovanejšie. Samotná viazanosť na distribúcie operačných systémov Linux, MacOS a FreeBSD vyžaduje istú dávku znalostí užívateľa, ktoré nemusia byť samozrejmosťou. Následná inštalácia NS-3 je problematrická. Existuje množstvo návodov a doporučení, ktoré však môžu spôsobiť ďalšie ťažkosti pri inštalácii simulačného prostredia.

Z hľadiska prístupu k technickej dokumentácii je v oboch spomínaných prostrediach možné hovoriť o pomerne svojskom prístupe autorov. Pri NS-3 užívateľ musí čeliť nástrahám neprehľadnej dokumentácie, ktorá môže zavádzať. V konečnom dôsledku však platí že prostredie NS-3 obsahuje obrovské množstvo funkčných modulov, ktoré v oblasti bezdrôtových sietí dokážu sprostredkovať prínosné simulácie. Kombinácia štruktúry programovacieho jazyka a knižníc je v prostredí NS-3 kombinovaná so samotnými sieťovými modelmi. Za predpokladu dostatočnej znalosti používaných knižníc a objektov je užívateľ schopný vytvoriť realistické simulácie. Prostredie OMNeT++ je v porovnaní s NS-3 jednoduchšie na obsluhu. V tejto práci už bola spomenutá štruktúra prostredia a aj samotný simulačný scenár. Technická dokumentácia OMNeT++ a knižnica INET pôsobí celistvým dojmom a užívateľovi ponúka aj predpripravené simulačné scenáre, ktoré názorne popisujú konkrétnu oblasť.

Samotné vytváranie simulácií v prostredí NS-3 je za predpokladu dostatočnej znalosti problematiky dostatočné. Užívateľ má možnosť simulovať rôzne deje a implementovať širokú škálu adaptačných algoritmov. V rámci problematiky bezdrôtových sietí tak prostredie NS-3 ponúka veľmi užitočné moduly, ktoré realistickým spôsobom simulujú adaptačné algoritmy. Dosah simulovanej bezdrôtovej siete a prenosové rýchlosti sú však závislé od množstva premenných a atribútov. Užívateľ získa použiteľný výsledok len za predpokladu, že dostatočne ovláda problematiku a dokáže vhodným spôsobom modifikovať atribúty simulačných modulov. V prostredí OMNeT++ je vytváranie scenárov síce menej časovo náročné, no užívateľ môže narážať na nedostatočne zdokumentované chyby vývojového prostredia, ktoré vedú

k nereálnym, alebo žiadnym výsledkom simulácií. Za predpokladu, že užívateľ v danej oblasti nenarazí na problematické moduly a chyby vývojového prostredia, môže získať realistické výsledky simulácií. Je nutné však dodať, že aj funkčná simulácia musí byť prispôsobená pomocou modifikácií atribútov. Často však tieto nastavené hodnoty atribútov nie sú odvoditeľné zo známych veličín, ktoré sú bežne zistiteľné z prístupných zdrojov.

Vo všeobecnosti je teda možné toto porovnanie zhrnúť do konštatovania, že užívateľ musí byť zoznámený s problematikou a musí poznať simulačné mechanizmy, nezávisle od toho, ktoré prostredie využije. Prostredie NS-3 však ponúka širšiu škálu možností a je vhodnejšie na dlhodobé testovanie a skúmanie problematiky. Prístupné moduly prostredia sú modifikovateľné a pri znalosti programovacieho jazyka C++ je možné vytvoriť veľmi kvalitné a užitočné simulácie. Prostredie OMNeT++ je však prehľadnejšie a užívateľ môže rýchlo získať použiteľné simulácie, ktoré môže ľahko prispôbiť. Výsledné horeuvedené grafické porovnania tak jednoznačne potvrdzujú, že za ideálnych podmienok je prostredie OMNeT++ jednoduchší a prehľadnejší nástroj než NS-3. Jeho výsledky môžu byť po modifikácii realistickejšie než neprehľadné skripty vytvorené v prostredí NS-3.

## Záver

Po záverečnom porovnaní získaných výsledkov je možné konštatovať, že práca zachytáva proces oboznamovania sa s problematikou adaptácie rýchlosti v bezdrôtových sieťach štandardu IEEE 802.11. Teoretické znalosti boli využité na vytvorenie scenárov v simulačných prostrediach, ktoré boli porovnané s nameranými hodnotami. Po

preštudovaní problematiky adaptačných algoritmov je badateľné, že sú podstatnou mierou zastúpené v oblasti bezdrôtových sietí. Ako reakciu na túto skutočnosť je možné chápať implementáciu adaptačných algoritmov do simulačných prostredí.

Z hľadiska analýzy adaptačných algoritmov je pri oboch prostrediach badateľné pomerne silné zastúpenie modulov, ktoré umožňujú simulácie a analýzu adaptačných algoritmov. Po porovnaní získaných dát sa dá považovať prostredie NS-3 za dokonalejšie z hľadiska množstva použiteľných modulov a modifikácií. Z užívateľského pohľadu je pri implementácii adaptačných algoritmov prostredie OMNeT++ vhodné na získanie orientačných dát, ktoré sú porovnateľné so skutočnými meraniami.

Z hľadiska perspektívneho štúdia a analýzy je naopak simulačné prostredie NS-3 dokonalejšie. Dá sa očakávať, že v nasledujúcom období bude možné sledovať zvyšujúci sa trend podpory rôznych modulov, určených pre simuláciu bezdrôtových sietí podľa štandardov 802.11. Rate Adaptation Algoritmy sa dajú považovať za bežnú súčasť bezdrôtových smerovačov, ktoré užívateľovi automaticky ponúkajú najvhodnejšiu prenosovú rýchlosť. Tento fakt potvrdzuje predpoklad, že výrobcovia kladú dôraz na čo najefektívnejšie využitie bezdrôtových sietí. Napriek pomerne zreteľnej snahe výrobcov využiť skutočnosť, že samotný štandard 802.11 bližšie nešpecifikuje parametre adaptačných algoritmov, je možné v prístupnej literatúre získať informácie, ktoré dokážu priblížiť základné zákonitosti v tejto skúmanej oblasti.

Záverom sa dá zhrnúť že táto práca zachytila proces analýzy Rate Adaptation Algoritmov a ich následnú implementáciu v simulačných prostrediach. Získané grafické závislosti sú výsledkom dlhodobého skúmania a snahy o získanie čo najrealistickejších dát.

# Literatúra

- [1] BIAZ, S.; WU, S.: Rate adaptation algorithms for IEEE 802.11 networks: A survey and comparison. In *2008 IEEE Symposium on Computers and Communications*, July 2008, ISSN 1530-1346, s. 130–136, doi:10.1109/ISCC.2008.4625680.
- [2] CHEN, J.; LI, H.; ZHANG, F.; aj.: MIMO Mode Switching Scheme for Rate Adaptation in 802.11n Wireless Networks. In *2011 IEEE Global Telecommunications Conference - GLOBECOM 2011*, Dec 2011, ISSN 1930-529X, s. 1–6, doi:10.1109/GLOCOM.2011.6133521.
- [3] Cisco Systems, Inc.: *Cisco Wireless Mesh Access Points, Design and Deployment Guide, Release 8.5*. December 2012.  
URL <<https://www.cisco.com/c/en/us/td/docs/wireless/technology/mesh/7-3/design/guide/Mesh.pdf>>
- [4] Juniper Networks: *Understanding the IEEE 802.11 Standard for Wireless Networks - Technical Documentation - Support - Juniper Networks*. November 2004.  
URL <[https://www.juniper.net/documentation/en\\_US/junos-space-apps/network-director3.2/topics/concept/wireless-80211.html](https://www.juniper.net/documentation/en_US/junos-space-apps/network-director3.2/topics/concept/wireless-80211.html)>
- [5] LACAGE, M.; MANSHAEI, M. H.; TURLETTI, T.: IEEE 802.11 Rate Adaptation: A Practical Approach. In *Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '04, New York, NY, USA: ACM, 2004, ISBN 1-58113-953-5, s. 126–134, doi:10.1145/1023663.1023687.  
URL <<http://doi.acm.org/10.1145/1023663.1023687>>
- [6] MCGREGOR, A.; SMITHIES', D.: Rate Adaptation for 802.11 Wireless Networks: Minstrel. *Blog CerWrt notebook*, 2019.  
URL <<http://blog.cerowrt.org/post/minstrel/>>
- [7] NS-3, A Discrete-Event Network Simulator: *Design Documentation*. 2020.  
URL <<https://www.nsnam.org/docs/models/html/wifi-design.html#interferencehelper>>
- [8] NS-3, A Discrete-Event Network Simulator: *Propagation*. 2020.  
URL <<https://www.nsnam.org/docs/models/html/propagation.html>>
- [9] NSAM: *NS-3 Network simulator*. May 2020.  
URL <<https://www.nsnam.org/>>

- [10] PAN, D.: Analysis of Wi-Fi performance data for a Wi-Fi throughput prediction approach. *KTH ROYAL INSTITUTE OF TECHNOLOGY SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY*, 2017.
- [11] PEFKIANAKIS, I.; HU, Y.; WONG, S. H.; aj.: MIMO Rate Adaptation in 802.11N Wireless Networks. In *Proceedings of the Sixteenth Annual International Conference on Mobile Computing and Networking, MobiCom '10*, New York, NY, USA: ACM, 2010, ISBN 978-1-4503-0181-7, s. 257–268, doi:10.1145/1859995.1860025.  
URL <<http://doi.acm.org/10.1145/1859995.1860025>>
- [12] PETÁK (GPL.sk): *GNU General Public License*. May 2020.  
URL <<http://www.gpl.sk/>>
- [13] PURANDARE, R. G.; KOLI, S. M.; KSHIRSAGAR, S. P.: Adaptive framework for rate adaptation in 802.11n wireless link. In *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPEs)*, Oct 2016, ISSN null, s. 173–177, doi:10.1109/SCOPEs.2016.7955743.
- [14] SALIGA, S. V.: An introduction to IEEE 802.11 wireless LANs. In *2000 IEEE Radio Frequency Integrated Circuits (RFIC) Symposium Digest of Papers (Cat. No.00CH37096)*, June 2000, s. 11–14, doi:10.1109/RFIC.2000.854406.
- [15] SARKAR, N.; MEMBARTH, R.: *Modeling and simulation of IEEE 802.11g using OMNeT++*. 01 2010, doi:10.4018/978-1-60566-774-4.ch017.
- [16] VARGA, A.: *OMNeT++ Simulation Manual*. Copyright © 2016 András Varga and OpenSim Ltd., 2016.
- [17] VARGA, A.; HORNIG, R.: An overview of the OMNeT++ simulation environment. 01 2008, str. 60, doi:10.1145/1416222.1416290.



## Zoznam symbolov, veličín a skratiek

<b>Mbit/s</b>	Megabit za sekundu (jednotka prenosovej rýchlosti)
<b>Hz</b>	Hertz (základná jednotka frekvencie)
<b>dBi</b>	Jednotka energetického zisku antény
<b>C++</b>	Viacparadigmaticý programovací jazyk vyššej úrovne
<b>RTS/CTS</b>	Request To Send / Clear To Send
<b>ARF</b>	Auto Rate Fallback
<b>RAA</b>	Robust Rate Adaptation Algorithm
<b>AARF</b>	Adaptive Auto Rate Fallback
<b>RBAR</b>	Receive Based Auto Rate
<b>OAR</b>	Opportunistic Auto Rate
<b>FAR</b>	Full Auto Rate
<b>LD-ARF</b>	Loss-differentiating-ARF
<b>CARA</b>	Collision-Aware Rate Adaptation
<b>LDRA</b>	Loss Differentiation Rate Adaptation
<b>RAA</b>	Rate Adaptation Algorithm
<b>NS-3</b>	Network Simulator 3
<b>OMNeT++</b>	Objective Modular Testbed in C++
<b>NED</b>	Network Description - jazyk pre nastavenie parametrov v OMNeT++
<b>IEEE</b>	Institute of Electrical and Electronics Engineers
<b>BSD</b>	Permisívna licencia používaná pre open source softvér
<b>MIMO</b>	MIMO - multiple input, multiple output
<b>TCP</b>	Transmission Control Protocol - Protokol riadenia prenosu
<b>UDP</b>	user datagram protocol - Používateľský datagramový protokol
<b>GNU</b>	GPL - General Public License

# Zoznam príloh

A Príloha CD

82

# A Príloha CD

```
/ ..... kořenový adresář přiloženého CD
├── omnet ..... simulačné skripty a grafy pre OMNeT++
│   ├── omnetpp.ini
│   ├── RateControllShowcase.ned
│   ├── 80211bOmnet.pdf
│   └── 80211gOmnet.pdf
├── ns3 ..... simulačné skripty a grafy pre NS-3
│   ├── rate-adaptation-distance.cc
│   ├── RAA80211b.cc
│   ├── RAA80211g.cc
│   ├── RAA80211n.cc
│   ├── NS380211b.pdf
│   ├── NS380211g.pdf
│   └── NS380211n.pdf
├── meranie ..... výsledky meraní
│   ├── meranie1 ..... výsledky prvého merania
│   │   ├── MeranieB.pdf
│   │   └── MeranieN.pdf
│   └── meranie2 ..... výsledky druhého merania
│       ├── 80211bTCPdown.pdf
│       ├── 80211bTCPup.pdf
│       ├── 80211bUDPdown.pdf
│       ├── 80211bUDPup.pdf
│       ├── 80211gTCPdown.pdf
│       ├── 80211gTCPup.pdf
│       ├── 80211gUDPdown.pdf
│       ├── 80211gUDPup.pdf
│       ├── 80211nTCPdown.pdf
│       ├── 80211nTCPup.pdf
│       ├── 80211nUDPdown.pdf
│       └── 80211nUDPup.pdf
```