

**ANALISIS PERBANDINGAN UNJUK KERJA ALGORITME TCP *NEW RENO*
DENGAN TCP *WESTWOOD+* UNTUK MEREDUKSI KONGESTI PADA JARINGAN**

WLAN

SKRIPSI

untuk memenuhi salah satu persyaratan mencapai derajat Sarjana S1



Disusun oleh:

M. Ryandy Ghonim Asgar

14524049

**Jurusan Teknik Elektro
Fakultas Teknologi Industri
Universitas Islam Indonesia
Yogyakarta**

2018

LEMBAR PENGESAHAN

ANALISIS PERBANDINGAN UNJUK KERJA ALGORITME TCP *NEW RENO*
DENGAN TCP *WESTWOOD+* UNTUK MEREDUKSI KONGESTI PADA JARINGAN
WLAN

TUGAS AKHIR

Diajukan sebagai Salah Satu Syarat untuk Memperoleh
Gelar Sarjana Teknik
pada Program Studi Teknik Elektro
Fakultas Teknologi Industri
Universitas Islam Indonesia

Disusun oleh:

M. Ryandy Ghonim Asgar
14524049



Yogyakarta, 28 Maret 2018

Menyetujui,

Pembimbing 1

Ida Nurcahyani, S.T.M.Eng
155240104

LEMBAR PENGESAHAN PENGUJI

ANALISIS PERBANDINGAN UNJUK KERJA ALGORITME TCP *NEW RENO*
DENGAN TCP *WESTWOOD+* UNTUK MEREDUKSI KONGESTI PADA JARINGAN
WLAN

TUGAS AKHIR

Disusun oleh :

M. Ryandy Ghonim Asgar

14524049

Telah Dipertahankan Didepan Sidang Penguji Syarat Untuk Memperoleh Gelar Sarjana
Konsentrasi Telekomunikasi Jurusan Teknik Elektro Fakultas Teknologi Industri

Universitas Islam Indonesia

Yogyakarta, 30 April 2018

Tim Penguji,

Ketua

Ida Nurcahyani, S.T, M.Eng

Anggota I

Yusuf Aziz Amrullah, S.T, M.Sc, Ph.D

Anggota II

Tito Yuwono, S.T, M.Sc

Mengetahui,

Ketua Jurusan Teknik Elektro

Universitas Islam Indonesia



Indira Setiawan S.T, M.T

PERNYATAAN

Dengan ini Saya menyatakan bahwa:

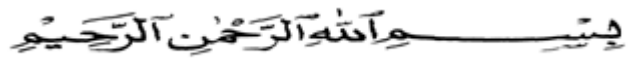
1. Skripsi ini tidak mengandung karya yang diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan Saya juga tidak mengandung karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.
2. Informasi dan materi Skripsi yang terkait hak milik, hak intelektual, dan paten merupakan milik bersama antara tiga pihak yaitu penulis, dosen pembimbing, dan Universitas Islam Indonesia. Dalam hal penggunaan informasi dan materi Skripsi terkait paten maka akan diskusikan lebih lanjut untuk mendapatkan persetujuan dari ketiga pihak tersebut diatas.

Yogyakarta, 28 Maret 2018



M. Ryandy Ghonim Asgar

KATA PENGANTAR



Assalamu'alaikum warahmatullahi wabarakatuh,

Alhamdulillahirabbil'alam, segala puji syukur bagi Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya, sehingga pada kesempatan ini penulis dapat menyelesaikan laporan tugas akhir yang berjudul “ANALISIS PERBANDINGAN UNJUK KERJA ALGORITME TCP NEW RENO DENGAN TCP WESTWOOD+ UNTUK MEREDUKSI KONGESTI PADA JARINGAN WLAN” sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Teknik Elektro Universitas Islam Indonesia.

Penulis juga menyadari bahwa selama berlangsungnya penelitian, sampai dengan tahap penyelesaian laporan skripsi ini tidak lepas dari bantuan dan dukungan oleh berbagai pihak. Oleh karena itu do'a dan ucapan terimakasih penulis sampaikan kepada:

1. Orang tua tercinta Ibu Hj.Zohriah, serta saudariku Andria Apridyana Safitri, dan tante Nazwa Ratna Santi yang telah memberikan nasihat, do'a, dan dukungan moril maupun materil kepada penulis dalam menuntut ilmu, sehingga penyusunan laporan skripsi ini dapat terselesaikan.
2. Dr.Eng.Hendra Setiawan,ST,MT. selaku ketua Jurusan Teknik Elektro Universitas Islam Indonesia (UII).
3. Ibu Ida Nurcahyani S.T, M.Eng. selaku dosen pembimbing yang telah banyak memberikan arahan, masukan, serta semangat dalam membimbing penulis untuk dapat menyelesaikan skripsi ini dengan baik.
4. Seluruh dosen Jurusan Teknik Elektro atas segala ilmu dan bimbingannya yang telah diberikan kepada penulis.
5. Saudara-saudara Teknik Elektro, khususnya angkatan 2014 yang telah saling memotivasi dan membantu terselesainya skripsi ini.
6. Seluruh pihak yang tidak bisa penulis sebutkan satu persatu.

Semoga skripsi ini dapat bermanfaat bagi pembaca, dan semua orang maupun kepada penulis sendiri, Aamiin.

ARTI LAMBANG DAN SINGKATAN

Wifi	: <i>Wireless Fidelity</i>
OSI	: <i>Open System Interconnection</i>
TCP	: <i>Transmission Control Protocol</i>
UDP	: <i>User Datagram Protocol</i>
LAN	: <i>Local Area Network</i>
WLAN	: <i>Wireless Local Area Network</i>
WPAN	: <i>Wireless Personal Area Network</i>
WMAN	: <i>Wireless Metropolitan Area Network</i>
WWAN	: <i>Wireless Wide Area Network</i>
NS	: <i>Network Simulator</i>
QoS	: <i>Quality of Service</i>
OFDM	: <i>Orthogonal Frequency Division Multiplexing</i>
AIAD	: <i>Additive Increase, Addaptive Decrease</i>
AIMD	: <i>Additive Increase, Multiplicative Decrease</i>
Mbit/s	: <i>Megabit per second</i>
Kbit/s	: <i>Kilobit per second</i>
s	: <i>second</i>

ABSTRAK

Wifi merupakan teknologi yang perkembangannya cukup pesat di era globalisasi. Seiring berkembangnya wifi, jumlah pengguna juga semakin bertambah. Ini menyebabkan peningkatan pada lalu lintas data yang disebabkan oleh bertambahnya jumlah pengguna tersebut. Peningkatan lalu lintas data tentunya memberikan efek negatif pada suatu jaringan, karena dapat memperlambat lalu lintas jaringan data atau yang biasa disebut dengan kongesti pada jaringan. Penulis menggunakan suatu teknik algoritme *Transmission Control Protocol* (TCP) untuk mereduksi kongesti. Penelitian ini membandingkan kinerja dari algoritme TCP *New Reno* dengan TCP *Westwood+* dalam mereduksi kongesti pada jaringan *Wireless Local Area Network* (WLAN). Metode penelitian yang dilakukan dengan menganalisis kinerja *throughput*, *packet delivery ratio*, dan *delay* pada masing-masing algoritme saat melakukan pengiriman data. Hasil penelitian menunjukkan bahwa kinerja *throughput* dan *packet delivery ratio* TCP *New Reno* lebih baik dibandingkan TCP *Westwood+* dengan nilai perbandingan *throughput* sebesar 0.63 Kbit/s dan nilai perbandingan *packet delivery ratio* sebesar 4%. Kemudian *delay* yang dihasilkan TCP *Westwood+* lebih besar 0.004s dibandingkan TCP *New Reno*.

Kata Kunci : Wifi, kongesti, TCP *New Reno*, TCP *Westwood+*, QoS

DAFTAR ISI

LEMBAR PENGESAHAN.....	i
LEMBAR PENGESAHAN PENGUJI	ii
PERNYATAAN.....	iii
KATA PENGANTAR.....	iv
ARTI LAMBANG DAN SINGKATAN	v
ABSTRAK	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	2
BAB 2 TINJAUAN PUSTAKA	4
2.1 Studi Literatur	4
2.2 Tinjauan Teori.....	5
2.2.1 Jaringan Komunikasi Wireless	5
2.2.2 WLAN (<i>Wireless Local Area Network</i>).....	5
2.2.3 Standar IEEE 802.11a.....	6
2.3 TCP	7
2.3.1 TCP New Reno	7

2.3.2 TCP Westwood+	8
2.4 Kongesti	8
2.5 FTP.....	9
2.6 Network Simulator 3.....	9
BAB 3 METODOLOGI	10
3.1 Alur Penelitian	10
3.2 Studi Literatur	11
3.3 Merancang Skenario	11
3.3.1 Skenario 1	11
3.3.2 Skenario 2	11
3.4 Proses Simulasi	12
3.5 Hasil Ujicoba	12
3.6 Pengukuran dan Perhitungan QoS	12
3.7 Analisa Data Hasil Ujicoba.....	13
3.8 Penyusunan Laporan.....	13
3.9 Perancangan Simulasi	13
3.9.1 Konfigurasi Parameter Simulasi	14
3.9.2 Pembuatan <i>Node</i>	14
3.9.3 Perancangan Topologi	14
3.9.4 Menentukan Protokol.....	15
3.9.5 Menentukan IP Address dan Jaringan	16
3.9.6 QoS	16
BAB 4 HASIL DAN PEMBAHASAN.....	18
4.1 Perbandingan Throughput.....	18
4.2 Perbandingan Packet Delivery Ratio	19
4.3 Perbandingan Delay	20
4.4 Perbandingan Jumlah <i>Node</i> Gagal	21

BAB 5 KESIMPULAN DAN SARAN.....	23
5.1 Kesimpulan	23
5.2 Saran	23
DAFTAR PUSTAKA	24
LAMPIRAN	26

DAFTAR GAMBAR

Gambar 2.1 Topologi BSS	6
Gambar 2.2 Topologi IBSS	6
Gambar 2.3 Kontrol kongesti pada <i>New Reno</i>	7
Gambar 2.4 Kontrol kongesti pada <i>Westwood+</i>	8
Gambar 3.1 <i>Flowchart</i> Penelitian.....	10
Gambar 3.2 <i>Flowchart</i> Simulasi	13
Gambar 3.3 Topologi yang digunakan	15
Gambar 4.1 Perbandingan <i>Throughput</i>	18
Gambar 4.2 Perbandingan <i>Packet Delivery Ratio</i>	19
Gambar 4.3 Perbandingan <i>delay</i>	20
Gambar 4.4 Perbandingan node gagal.....	21

DAFTAR TABEL

Tabel 2.1 Jangkauan jarak IEEE 802.11a	7
Tabel 3.1 Parameter Simulasi.....	14
Tabel 3. 2Parameter QoS	17

BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

Wireless Fidelity (wifi) adalah sebuah teknologi yang menggunakan peralatan elektronik untuk bertukar data secara nirkabel [1]. Istilah wifi pertama kali diciptakan oleh suatu organisasi yang bernama WI-FI Alliance yang bekerja untuk menguji dan memberikan sertifikasi untuk perangkat-perangkat WLAN [2]. Wifi juga bisa digunakan untuk membuat jaringan diperusahaan, oleh karena itu banyak orang mengasosiasikan teknologi wifi kepada pemakainya untuk mengakses internet atau mentransfer data dari kampus, sekolah, dan hotel, atau tempat-tempat yang bertanda wifi *Hot Spot* [3]. Seiring berkembangnya wifi, jumlah lalu lintas jaringan juga terus meningkat. Peningkatan tersebut karena bertambahnya jumlah pengguna pada suatu *access point* sehingga menyebabkan kongesti pada jaringan.

Kongesti adalah perlambatan laju paket-paket data yang disebabkan ketika sebuah jaringan mempunyai beban yang banyak atau berlebih [4]. Terdapat beberapa cara untuk mengatasi kongesti, antara lain dengan memperbesar *bandwith* dan paket data, memperbesar jumlah memori pada *router*, dan menggunakan algoritme *Transmission Control Protocol* (TCP) [5]. Namun masih terdapat kekurangan pada masing-masing cara, salah satunya dengan memperbesar *bandwith* dan paket data membutuhkan biaya yang tidak sedikit. Disisi lain, memperbesar jumlah memori pada *router*, mampu mengatasi masalah kongesti dalam jaringan tetapi dapat memberikan efek semakin tingginya *delay* antara pengirim dan penerima [5]. Oleh karena itu, penulis menggunakan suatu teknik algoritme TCP untuk mengatasi kongesti dalam jaringan karena biaya yang terjangkau dan mampu meningkatkan *Quality of Service* (QoS) tergantung dari jenis algoritme yang digunakan. Terdapat beberapa algoritme dasar yang disediakan oleh TCP, diantaranya adalah *Tahoe* dan *Reno* [6]. Salah satu pengembangan algoritme *Reno* adalah *New Reno*, dengan kontrol kongesti yang lebih berkembang daripada *Reno* [7]. Adapun TCP *Westwood+* yang merupakan evolusi dari TCP *Westwood*, dan merupakan pengembangan dari TCP *New Reno* [7]. Perbedaan dari TCP *New Reno* dan TCP *Westwood+* terletak pada kontrol kongesti [7].

Pada penelitian ini penulis menganalisis dan membandingkan kinerja dari algoritme TCP *New Reno* dengan TCP *Westwood+* pada jaringan *Wireless Local Area Network* (WLAN) dengan menggunakan perangkat lunak *Network Simulator* (NS) 3.27. Penelitian ini bertujuan

untuk mengetahui kinerja TCP terbaik antara TCP *New Reno* dan TCP *Westwood+* dalam mereduksi kongesti pada jaringan WLAN.

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini :

1. Bagaimana pengaruh penambahan jumlah pengguna pada algoritme *New Reno* dan *Westwood+* pada jaringan WLAN dalam mengatasi kongesti?
2. Bagaimana mengimplementasikan algoritme *New Reno* dan *Westwood+* pada jaringan WLAN untuk mengatasi kongesti?
3. Bagaimana menganalisis kinerja algoritme *New Reno* dan *Westwood+* pada jaringan WLAN saat mengatasi kongesti?
4. Bagaimana perbandingan kinerja algoritme *New Reno* dan *Westwood+* pada jaringan WLAN dalam mengatasi kongesti?

1.3 Batasan Masalah

Adapun batasan-batasan masalah pada penelitian ini :

1. Parameter *Quality of Service* (QoS) yang dianalisis adalah *throughput*, *packet delivery ratio*, dan *delay*.
2. Simulasi hanya dilakukan pada jaringan WLAN 802.11a dengan menggunakan perangkat lunak *Network Simulator* (NS) 3.27.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini :

1. Mereduksi kongesti dengan menggunakan algoritme *New Reno* dan *Westwood+* pada jaringan WLAN.
2. Menganalisis dan membandingkan kinerja algoritme *New Reno* dan *Westwood+* yang lebih baik untuk mereduksi kongesti pada jaringan WLAN.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini :

1. Mereduksi kongesti dengan menggunakan algoritme *New Reno* dan *Westwood+* pada jaringan WLAN.

2. Mengetahui kinerja algoritme yang terbaik antara *New Reno* dan *Westwood+* dalam mereduksi kongesti pada jaringan WLAN.

BAB 2

TINJAUAN PUSTAKA

2.1 Studi Literatur

Pada penelitian [8] dilakukan pengimplementasian mekanisme kontrol kongesti untuk mengatur lalu lintas jaringan saat terjadinya pemakaian berlebih. Penelitian ini membandingkan empat model TCP, yaitu TCP *Cubic*, *Reno*, *Vegas*, dan *Westwood+*. Metode pengujian yang dilakukan adalah menganalisis *latency*, *jitter*, *packet loss*, dan *throughput* pada masing-masing model TCP pada jaringan *Local Area Network* (LAN) dengan *bandwidth* 100 Mbit/s. Hasil dari penelitian ini menunjukkan bahwa TCP *Westwood+* adalah TCP yang terbaik diantara ketiga TCP lainnya dalam mengatasi kongesti pada jaringan LAN berdasarkan *latency*, *jitter*, *packet loss*, dan *throughput*.

Pada penelitian R. Fahrizal [9] dilakukan pengujian algoritme TCP pada jaringan dengan dua topologi yang berbeda. Beberapa algoritme TCP yang diujikan ialah algoritme *BIC*, *Cubic*, dan *HTCP*. Algoritme-algoritme tersebut diujikan dengan menggunakan dua topologi yang berbeda yaitu topologi *parkinglot* dan topologi *multihome* dengan menganalisa *throughput* masing-masing TCP. Hasil dari penelitian ini menunjukkan bahwa algoritme TCP *BIC* adalah TCP yang terbaik diantara kedua TCP lainnya pada topologi *parkinglot* dan topologi *multihome*.

Penelitian [10] melakukan pengujian terhadap kinerja dari algoritme TCP *Tahoe* dan TCP *New Reno*. Pada penelitian ini menggunakan skenario simulasi jaringan *wired* dan *wireless* dengan parameter kinerja yang diujikan ialah *throughput*, *delay*, dan *packet drop*. Hasil pengujian pada jaringan *wired* maupun *wireless* menunjukkan bahwa algoritme TCP *New Reno* mampu mengungguli kinerja dari TCP *Tahoe* dalam seluruh parameter kinerja yang diujikan.

Penelitian Rineta Inelia dan kawan-kawan [11] melakukan pengimplementasian mekanisme untuk mengontrol kongesti. Penelitian yang dilakukan menggunakan skema perbandingan TCP dengan *Pragmatic General Multicast Congestion Control* (PGMCC) dengan parameter yang diuji ialah *throughput*, *packet loss*, dan *delay*. Hasil pengujian ini menunjukkan beberapa keunggulan dan kekurangan dari PGMCC. PGMCC mampu menangani *packet loss* dengan baik daripada TCP, namun PGMCC memiliki *delay* yang besar dan *throughput* yang kecil dibandingkan dengan TCP.

2.2 Tinjauan Teori

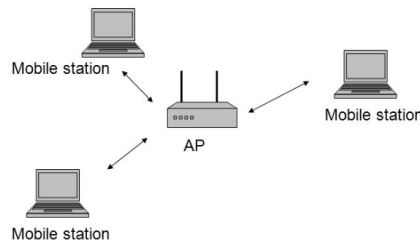
2.2.1 Jaringan Komunikasi Wireless

Jaringan komunikasi *wireless* ialah jaringan komunikasi yang memanfaatkan udara sebagai media transmisinya, atau bisa disebut jaringan yang mampu menghubungkan satu atau lebih perangkat-perangkat elektronik untuk berkomunikasi tanpa menggunakan kabel [12]. Hal ini tentunya memudahkan para pengguna dalam menggunakan perangkat tanpa dipengaruhi oleh keterbatasan penggunaan oleh media transmisi fisik. Teknologi wireless ini dibagi menjadi 4 kategori berdasarkan *coverage*, yakni *Wireless Personal Area Network* (WPAN), *Wireless Local Area Network* (WLAN), *Wireless Metropolitan Area Network* (WMAN), dan *Wireless Wide Area Network* (WWAN) [12].

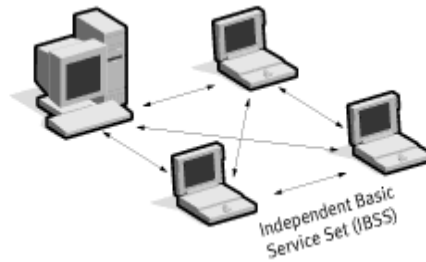
2.2.2 WLAN (*Wireless Local Area Network*)

WLAN merupakan bentuk jaringan yang sesuai standar IEEE 802.11. WLAN menggunakan teknologi frekuensi radio sebagai media penyimpanan data dan juga dapat memberikan kemudahan bagi pengguna dalam penerapannya, antara lain:

1. Pengguna dapat mengakses informasi dimanapun sepanjang masih berada dalam *coverage* jaringan WLAN.
2. Instalasi jaringan WLAN lebih cepat dibandingkan jaringan dengan menggunakan kabel karena perangkat yang digunakan tidak terlalu banyak dan mudah untuk dikonfigurasi.
3. Instalasi jaringan dapat dilakukan ditempat dimana jaringan *Local Area Network* (LAN) tidak dapat dipasang karena kendala kondisi geografis.
4. Jaringan WLAN dapat dikonfigurasi dengan beberapa bentuk topologi, sesuai dengan kebutuhan pengguna. Topologi WLAN antara lain, *Independent Service Set* (IBSS) dan *Based Service Set* (BSS). IBSS adalah jenis topologi yang biasa dikenal dengan *ad-hoc network* dimana seluruh node saling terhubung dan tidak ada node yang berfungsi sebagai server. Sedangkan BSS adalah *wireless client* dihubungkan oleh sebuah perangkat *access point* dan setiap *wireless client* yang ingin terhubung dengan *client* lainnya harus terhubung dengan *access point* terlebih dahulu.



Gambar 2.1 Topologi BSS



Gambar 2.2 Topologi IBSS

Terdapat beberapa standar IEEE 802.11, diantaranya adalah 802.11a, 802.11b, 802.11c, 802.11d, 802.11e, 802.11f, 802.11g, 802.11h, 802.11i, 802.11r, 802.11j [13]. Namun dari banyaknya standar yang dimiliki oleh IEEE 802.11, terdapat tiga standar yang paling banyak diimplementasikan pada perangkat WLAN ialah IEEE 802.11a, 802.11b, 802.11g [14]. Walaupun memiliki banyak kemudahan yang diberikan oleh teknologi WLAN, Teknologi ini juga memiliki kelemahan dalam implementasinya, dimana terdapat pengaruh interferensi radio dan halangan akibat bangunan maupun pepohonan dan lain-lain [12].

2.2.3 Standar IEEE 802.11a

Standar IEEE 802.11a adalah protokol jaringan WLAN yang dipublikasikan pada tahun 1999 [12]. 802.11a bekerja pada band frekuensi 5 GHz dengan teknik *Orthogonal Frequency Division Multiplexing* (OFDM) dan menggunakan 52 *sub-carrier* yang dimodulasi dengan teknik *Binary Phase Shift Keying* (BPSK), *Quadrature Phase Shift Keying* (QPSK), *16-Quadrature Amplitude Modulation* (16-QAM), atau *64-Quadrature Amplitude Modulation* (64-QAM) [14]. *Data rate* yang dimiliki IEEE 802.11a ialah 6 Mbps, 9 Mbps, 12 Mbps, 18 Mbps, 24 Mbps, 36 Mbps, 48 Mbps, dan 54 Mbps [12]. Adapun parameter jarak dari *data rate* IEEE 802.11a dapat dilihat pada Tabel 2.1:

Tabel 2.1 Jangkauan jarak IEEE 802.11a [14]

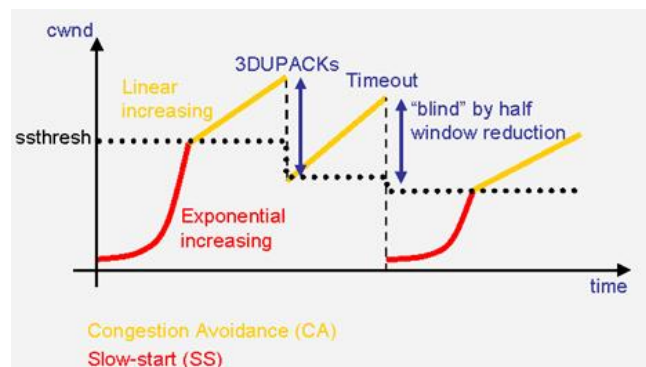
Data Rate (Mbps)	Jarak indoor (m)	Jarak outdoor (m)
6	100	198
9	91	190
12	84	183
18	76	168
24	69	152
36	60	130
48	45	91
54	24	30

2.3 TCP

Transmission Control Protocol (TCP) merupakan suatu protokol yang berada pada *Layer* ke-4 *Open System Interconnection* (OSI) [15]. TCP ialah salah satu protokol utama dalam protokol di internet yang menyediakan pengiriman paket yang dapat diandalkan, sehingga mampu membuat aliran data yang diterima oleh TCP *receiver* tidak rusak [8]. Algoritme dalam TCP *congestion control* merupakan faktor utama yang mampu berperan penting dalam baik buruknya kinerja suatu jaringan [8].

2.3.1 TCP New Reno

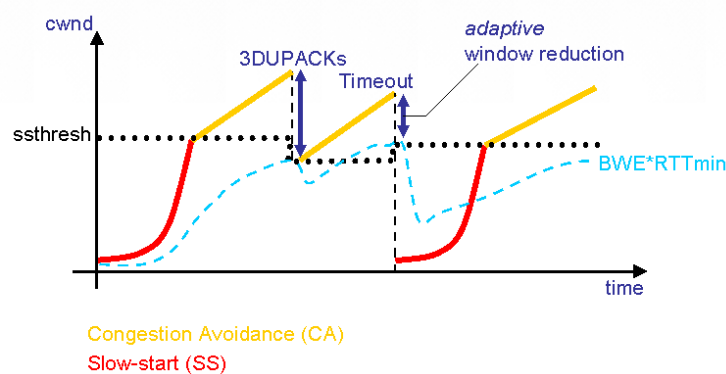
TCP *New Reno* ialah pengembangan algoritma dari TCP *Reno*. TCP *New Reno* hanya mampu menangani satu segmen paket data yang hilang sehingga dapat menangani pengiriman ulang paket data hilang lebih dari satu dalam satu *window* tanpa menurunkan *ssthresh* berkali-kali karena tidak akan meninggalkan fase *fast recovery* sebelum semua paket dalam satu *window* di *ack* semua [7].



Gambar 2.3 Kontrol kongesti pada *New Reno* [7]

2.3.2 TCP Westwood+

TCP Westwood+ ialah pengembangan dari TCP westwood yang dikembangkan oleh Saverio Mascolo dan teman-teman [7]. TCP Westwood+ juga merupakan pengembangan lebih lanjut dari TCP New Reno, namun terdapat perbedaan yang terletak dibagian kontrol kongesti. TCP Westwood+ menggunakan AIAD (*Additive Increase, Addaptive Decrease*), dimana paket hilang tidak akan direspon secara ekstrim karena nilai *ssthresh* dan *cwnd* akan diestimasi secara adaptif berdasarkan bandwidth *end-to-end* pada jaringan [7]. Sedangkan TCP New Reno menggunakan AIMD (*Additive Increase, Multiplicative Decrease*), dimana saat terjadi paket hilang akan direspon dengan menurunkan *ssthresh* dan *cwnd* menjadi setengah sehingga pita yang dipakai menurun drastic [7].



Gambar 2.4 Kontrol kongesti pada Westwood+ [7]

2.4 Kongesti

Kongesti adalah perlambatan laju paket-paket data yang disebabkan ketika sebuah jaringan mempunyai beban yang banyak atau berlebih [4]. Kontrol kongesti difokuskan pada lalu lintas data yang masuk ke sebuah jaringan telekomunikasi, untuk menghindari kegagalan pengiriman data dilakukan pengurangan tingkat pengiriman paket [11].

Kongesti merupakan permasalahan yang muncul di jaringan akibat banyak pengguna yang saling berbagi *bandwidth* dan jumlah permintaan akan *bandwidth* melebihi kapasitas *link* yang tersedia. Kongesti melibatkan beberapa hal berikut [11]:

1. Kemampuan jalur komunikasi data dalam mentransfer data terbatas.
2. Beban jaringan, jumlah data yang dapat dikirim dapat bervariasi tergantung jumlah pengguna dan kecepatan pengiriman data.

2.5 FTP

File Transfer Protocol (FTP) adalah protokol internet yang digunakan untuk pengiriman dan penerimaan data pada jaringan komputer, misalnya *upload* dan *download* file yang dilakukan oleh FTP *client* dan FTP *server* [16]. FTP ialah protokol internet yang paling awal dikembangkan, dan sampai saat ini masih digunakan untuk melakukan pengunduhan maupun pengunggahan [17]. FTP menggunakan *Protokol Transmission Control Protocol* (TCP) untuk komunikasi data yang dilakukan antara *client* dan *server* [17].

2.6 Network Simulator 3

NS-3 merupakan simulator jaringan yang dapat digunakan untuk mensimulasikan jaringan komputer dan telekomunikasi. NS-3 bersifat *open source* dan dikembangkan menggunakan bahasa C++. Versi stabil NS-3 pertama sekali diperkenalkan pada tahun 2008 [18]. Simulasi dari jaringan nirkabel dan protokol, seperti algoritma routing, TCP, dan UDP dapat diselesaikan dengan baik menggunakan aplikasi ini. NS-3 banyak digunakan sebagai media pengajaran protokol jaringan. Terdapat beberapa keunggulan yang dimiliki aplikasi ini, yaitu *Network simulator* dilengkapi dengan *tools validasi*, pembuatan simulasi dengan menggunakan *Network Simulator* jauh lebih mudah daripada menggunakan *software developer* seperti Dhelphi atau C++ [7].

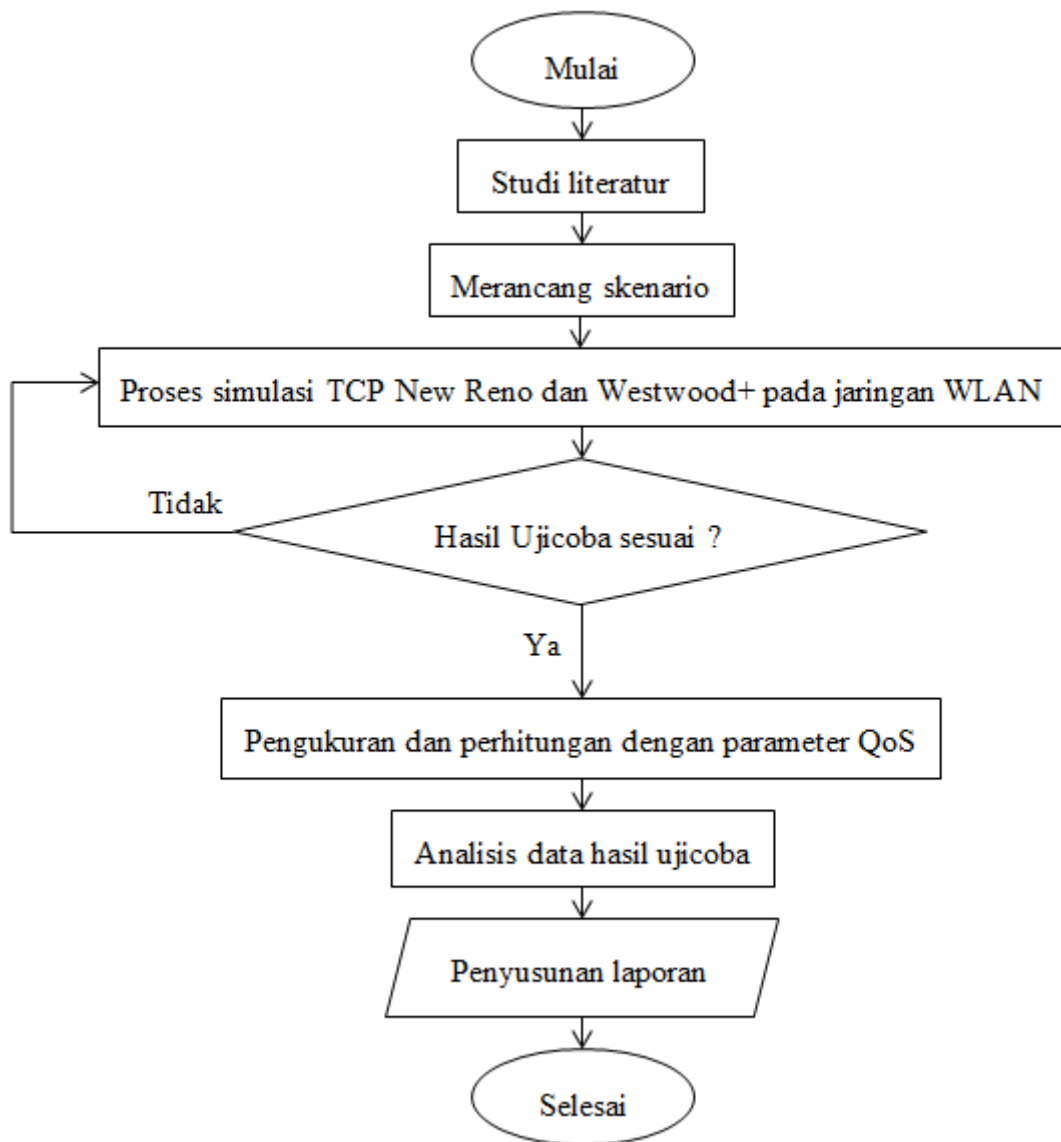
BAB 3

METODOLOGI

Dalam BAB ini menjelaskan hal-hal pokok yang dikaji dalam penelitian ini dan disampaikan dalam bentuk diagram alir (*flowchart*). Metodologi ini diuraikan menjadi beberapa tahapan, yakni :

3.1 Alur Penelitian

Alur pada penelitian ini diuraikan dalam diagram alir (*flowchart*) yang ditunjukkan pada Gambar 3.1:



Gambar 3.1 *Flowchart* Penelitian

3.2 Studi Literatur

Pada proses ini penulis mencari beberapa jurnal dari berbagai sumber untuk dijadikan acuan penelitian yang dilakukan. Selain berfungsi sebagai acuan kepada penulis, jurnal tersebut dapat menambah wawasan penulis dan dapat dijadikan parameter untuk penelitian.

3.3 Merancang Skenario

Pada penelitian ini penulis menggunakan dua skenario, skenario pertama adalah proses simulasi menggunakan *TCP New Reno*, sedangkan skenario kedua adalah proses simulasi menggunakan *TCP Westwood+*. Penjelasan dari masing-masing skenario dijelaskan pada sub-bab berikut:

3.3.1 Skenario 1

Pada proses simulasi 1 menggunakan *TCP New Reno*. Penulis menguraikan tentang implementasi algoritma *TCP New Reno* yang digunakan, jumlah *node* yang digunakan pada saat simulasi, waktu proses simulasi yang digunakan, dan jenis jaringan yang digunakan.

Pada penelitian ini menggunakan aplikasi *Network Simulator 3*, pada proses simulasi 1 penulis menggunakan algoritma *TCP New Reno* dengan menggunakan *data rate* 6 Mbps dan *bandwidth* 10 Mhz dengan jumlah *node* yang berbeda-beda dengan waktu simulasi masing-masing *node* ialah 300 detik dan dilakukan pada jaringan WLAN. Percobaan ke-1 menggunakan algoritma *TCP New Reno* dengan jumlah *node* 50, kemudian pada percobaan ke-2 dengan menggunakan 60 *node*, kemudian percobaan ke-3 kembali mengganti jumlah *node* dengan jumlah 70 *node*, percobaan ke-4 menggunakan 100 *node*, dan yang terakhir percobaan ke-5 menggunakan 120 *node*.

3.3.2 Skenario 2

Pada proses simulasi 2 menggunakan *TCP Westwood+* ini tidak berbeda jauh dengan simulasi *TCP New Reno* sebelumnya, hanya saja yang membedakannya terletak pada algoritme yang penulis gunakan. Dengan kata lain jumlah *node* yang digunakan pada saat simulasi, waktu proses simulasi yang digunakan, dan jenis jaringan yang digunakan sama seperti simulasi pada *TCP New Reno*.

Pada penelitian ini menggunakan aplikasi *Network Simulator 3*, pada proses simulasi ke-2 ini penulis menggunakan algoritma *TCP Westwood+* dengan menggunakan jumlah *node* yang

berbeda-beda dengan waktu simulasi masing-masing *node* ialah 300 detik dan dilakukan pada jaringan WLAN. Sama halnya dengan percobaan-percobaan pada TCP *New Reno*, di percobaan ke-1 TCP *Westwood+* menggunakan jumlah *node* 50, kemudian pada percobaan ke-2 menggunakan 60 *node*, kemudian percobaan ke-3 kembali mengganti jumlah *node* dengan 70 *node*, percobaan ke-4 menggunakan 100 *node*, dan yang terakhir percobaan ke-5 menggunakan 120 *node*.

3.4 Proses Simulasi

Pada proses ini simulasi digunakan dengan skenario 1 dan skenario 2 yang sudah ditentukan oleh penulis, kemudian simulasi dijalankan dan mendapatkan hasil simulasi.

3.5 Hasil Ujicoba

Pada proses ini, penulis mendapatkan hasil simulasi yang sudah dijalankan dengan menggunakan skenario 1 dan skenario 2. Jika hasil ujicoba berhasil maka dilanjutkan ke proses selanjutnya yaitu pengukuran dan perhitungan QoS, tetapi jika hasil ujicoba dianggap gagal maka dilakukan pengujian ulang pada proses simulasi hingga hasil ujicoba yang didapatkan berhasil.

3.6 Pengukuran dan Perhitungan QoS

Pada proses ini pengambilan data QoS yang dilakukan dengan beberapa tahapan, yaitu :

- a. Seluruh *node* mengirimkan paket pada *node* yang telah ditentukan yaitu, *node* 0 (IP Address : 10.1.1.2) , kemudian letak *node* 0 merupakan jarak terjauh antar *node* dengan *access point* (IP Address: 10.1.1.1).
- b. Proses pengukuran dilakukan menggunakan FTP dengan beberapa perubahan *node* pada setiap percobaan yang dilakukan. *Node* yang diberikan pada masing-masing percobaan antara lain 50 *node*, 60, *node*, 70, *node*, 100 dan 120 *node*.
- c. Melakukan pengambilan data dengan parameter QoS berupa *throughput*, *packet delivery ratio*, dan *delay*.

3.7 Analisa Data Hasil Ujicoba

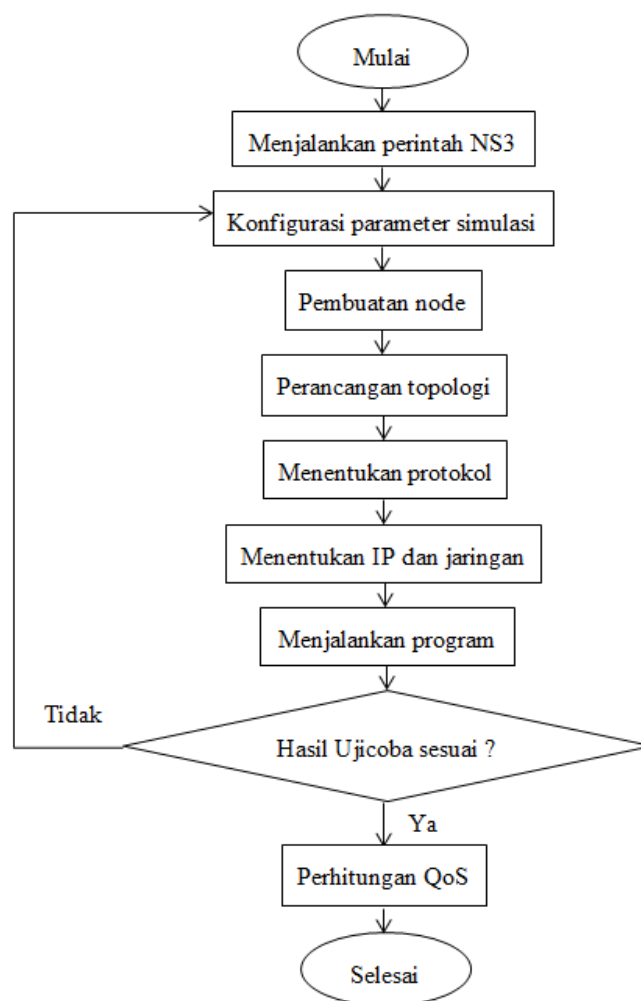
Pada proses ini analisis dilakukan dengan mengamati kualitas parameter-parameter QoS yang telah ditentukan dari kedua TCP yang digunakan dengan masing-masing jumlah *node* yang digunakan.

3.8 Penyusunan Laporan

Ini adalah proses terakhir dalam alur penelitian, penyusunan laporan ini berisi tentang awal mula penulis mencari studi literatur hingga mendapatkan hasil ujicoba yang telah dilakukan dan kemudian membuatnya dalam bentuk laporan.

3.9 Perancangan Simulasi

Perancangan simulasi ini di uraikan dalam diagram alir (*flowchart*) yang ditunjukkan pada Gambar 3.2:



Gambar 3.2 *Flowchart* Simulasi

Berikut ini adalah parameter yang digunakan untuk menjalankan simulasi, dapat dilihat pada Tabel 3.1:

Tabel 3.1 Parameter Simulasi

Parameter	Nilai
Waktu Simulasi	300 detik
Data Rate	6 Mbps
Bandwith	10 MHz
Ukuran Paket	1024 bytes
Traffic Source	FTP
Posisi Node	Random walk position
Kecepatan Node	1 m/s
Dimensi Ruang	100m x 50m
Tipe Kanal	WLAN (802.11a)
Skenario 1	TCP New Reno (Jumlah Node : 50,60,70,100, 120)
Skenario 2	TCP Westwood+ (Jumlah Node : 50,60,70,100,120)

3.9.1 Konfigurasi Parameter Simulasi

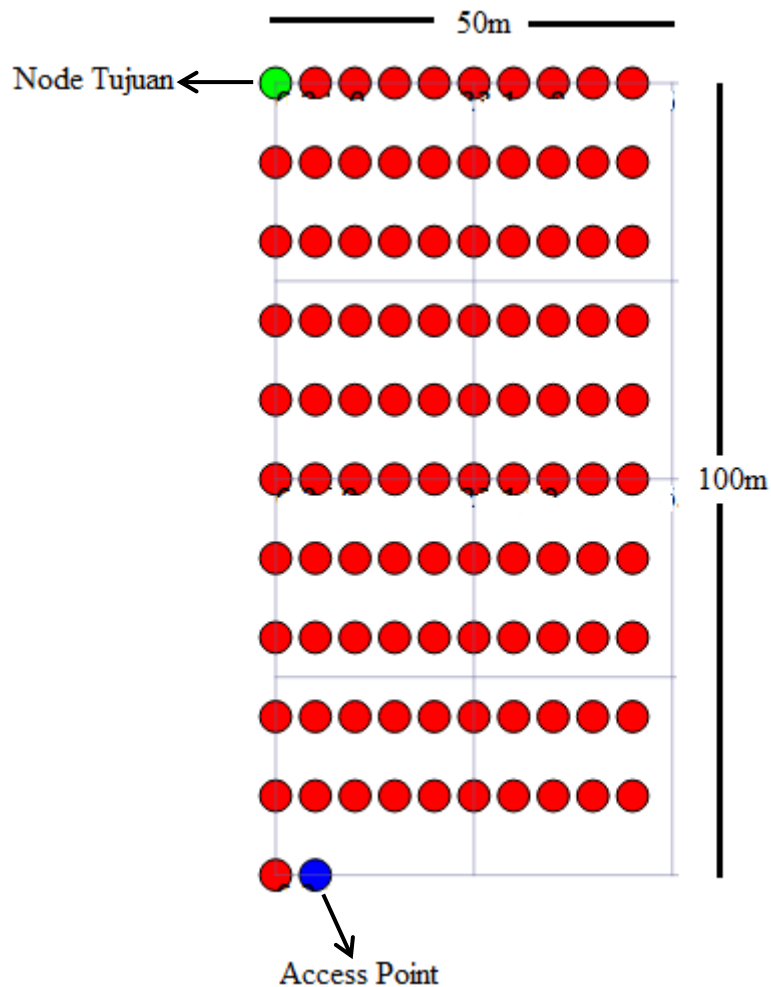
Ini adalah proses penulis memasukkan parameter-parameter yang digunakan dalam melakukan simulasi. Parameter-parameter yang dimasukkan berfungsi untuk memudahkan penulis dalam mengimplementasikan skenario yang diujikan.

3.9.2 Pembuatan *Node*

Pada proses ini penulis menggunakan 50, 60, 70, 100 , dan 120 *node* yang digunakan pada masing-masing algoritma TCP *New Reno* dan TCP *Westwood+*. Penambahan *node* pada setiap percobaan berfungsi untuk menganalisis kualitas QoS dari algoritma TCP *New Reno* dan TCP *Westwood+* yang digunakan tersebut.

3.9.3 Perancangan Topologi

Pada proses ini penulis akan menggunakan jenis topologi *Basic Service Set* (BSS). Koneksi antara *wireless client* pada topologi ini dihubungkan oleh sebuah perangkat *access point* dan setiap *wireless client* yang ingin terhubung dengan *client* lainnya harus terhubung dengan *access point* yang digunakan terlebih dahulu [19]. Kemudian setelah semua *client* terhubung dengan *access point* yang digunakan, maka semua *node* mengirim data FTP kepada 1 *client* yang telah ditentukan. Berikut adalah contoh topologi yang digunakan pada simulasi dengan 100 pengguna dapat dilihat pada Gambar 3.3:



Gambar 3.3 Topologi yang digunakan untuk 100 pengguna

Keterangan :

- → Node Tujuan
- → Node Pengirim
- → Access Point

Pada percobaan ini menggunakan ruangan dengan luas 100m x 50m dengan posisi access point diletakkan berjauhan dengan posisi node penerima, dengan tujuan untuk memperbesar terjadinya kongesti pada jaringan tersebut.

3.9.4 Menentukan Protokol

Pada proses ini penulis menggunakan jenis protokol TCP (*Transmission Control Protocol*). Terdapat berbagai jenis algoritme TCP yang disediakan di aplikasi NS3, namun penulis hanya membandingkan 2 jenis algoritme TCP, yaitu TCP *New Reno* dengan TCP *Westwood+*.

3.9.5 Menentukan IP Address dan Jaringan

Pada proses ini penulis menggunakan jaringan WLAN dengan standar IEEE 802.11a dengan menggunakan parameter *data rate* 6 Mbps dan *bandwidth* 10 MHz. Penulis juga membuat IPV4 *address* pada setiap *node*, *access point* menggunakan IP 10.1.1.1, kemudian *node* 0 sebagai *node* tujuan *upload* dari setiap *node* menggunakan IP 10.1.1.2 dan *node* yang lain akan mengikuti perubahan pada digit terakhir IP.

3.9.6 QoS

Quality of Service (QoS) ialah kemampuan dalam menyediakan tingkatan layanan untuk mentransmisi data pada suatu jaringan. QoS jaringan *wireless* antara lain berdasarkan *throughput*, *packet delivery ratio*, dan *delay* [12].

1. *Throughput* ialah jumlah kedatangan paket yang berhasil yang diamati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut dengan satuan *bit/second* (bps) [20]. Persamaan *throughput* yang digunakan adalah:

$$\text{Throughput} = \text{rxBytes} / (\text{Time last RxPacket} - \text{Time first TxPacket}) \quad (3.1)$$

Keterangan :

rxBytes = Paket yang diterima (*bytes*)

Time last RxPacket = Waktu terakhir paket diterima

Time first TxPacket = Waktu pertama pengiriman paket

2. *Packet delivery ratio* disini ialah jumlah persentase paket-paket yang berhasil dikirim dalam melakukan proses pengiriman data dari sumber ke tempat tujuan yang dituju [20].

$$PDR = \frac{\text{rxPackets}}{\text{txPackets}} \times 100\% \quad (3.2)$$

Keterangan :

rxPackets = Paket yang diterima

txPackets = Paket yang dikirim

3. *Delay* merupakan waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tempat tujuan [20].

$$Delay = \frac{delaySum}{rxPackets} \quad (3.3)$$

Keterangan :

delaySum = Waktu paket tiba

rxPackets = Paket yang diterima

Berikut ini adalah parameter QoS berdasarkan standar ETSI yang ditunjukkan pada Table 2.2:

Tabel 3.2 Parameter QoS [19]

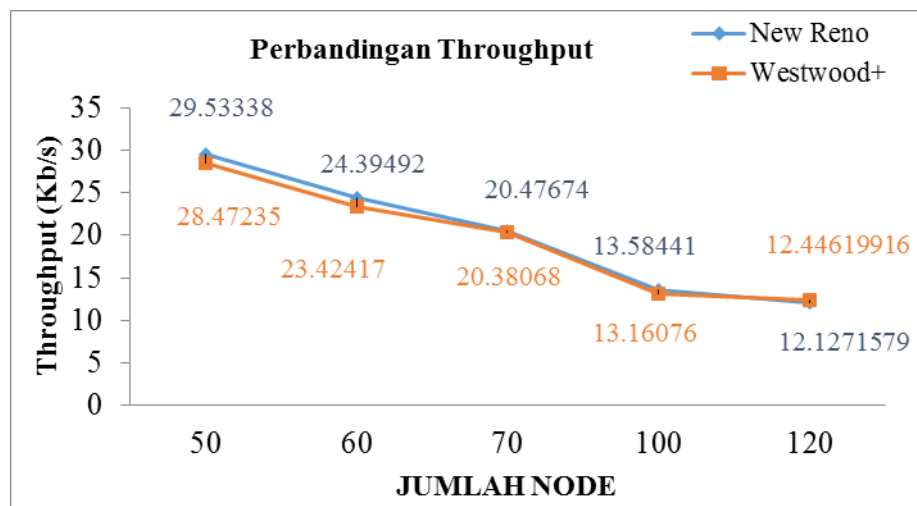
Kategori Degradasi	<i>Packet Delivery Ratio</i>	Besar <i>Delay</i>	Indeks
Sangat Bagus	98%-100%	<150ms	4
Bagus	86%-97%	150 s/d 300ms	3
Sedang	76-85%	300 s/d 450ms	2
Buruk	<75%	>450ms	1

BAB 4

HASIL DAN PEMBAHASAN

Pengujian ini bertujuan untuk mengetahui kinerja terbaik antara TCP *New Reno* dengan TCP *Westwood+* dalam mereduksi kongesti yang dilakukan pada jaringan WLAN. Kedua TCP tersebut diujikan dengan menggunakan masing-masing 50, 60, 70, dan 100 *node* dengan 1 *access point*. Parameter QoS yang diujikan ialah *throughput*, *packet delivery ratio*, dan *delay*.

4.1 Perbandingan Throughput

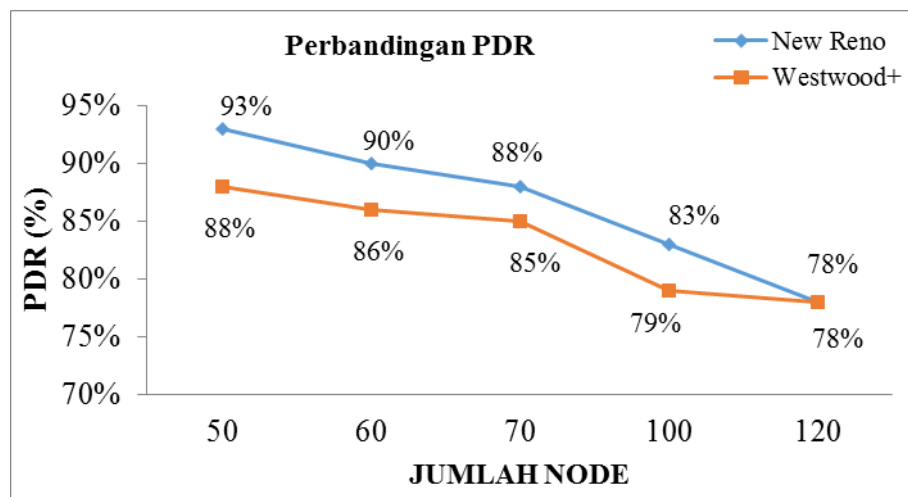


Gambar 4.1 Perbandingan *Throughput*

Dapat dilihat pada Gambar 4.1 adalah hasil *throughput* yang telah didapatkan pada percobaan TCP *New Reno* dan TCP *Westwood+*. Pada saat menggunakan 50 *node*, *throughput* yang dihasilkan oleh TCP *New Reno* adalah 29.5 Kbit/s, sedangkan *throughput* yang dihasilkan oleh TCP *Westwood+* adalah 28.4 kbit/s. Kemudian saat menggunakan 60 *node*, *throughput* yang dihasilkan oleh TCP *New Reno* mengalami penurunan menjadi 24.3 Kbit/s, begitu juga dengan TCP *Westwood+* juga mengalami penurunan menjadi 23.4 Kbit/s. Setelah itu saat menggunakan 70 *node*, *throughput* yang dihasilkan oleh kedua TCP *New Reno* dan TCP *Westwood+* kembali mengalami penurunan hingga mencapai 20.4 Kbit/s dan 20.3 Kbit/s. Saat menggunakan 100 *node* juga *throughput* yang dihasilkan mengalami penurunan, *throughput* yang dihasilkan TCP *New Reno* adalah 13.5 Kbit/s dan *throughput* yang dihasilkan TCP *Westwood+* adalah 13.1 Kbit/s. Kemudian saat menggunakan 120 *node* kedua TCP mengalami penurunan, namun *throughput* yang dihasilkan TCP *Westwood+* lebih bagus daripada TCP *New Reno*, yakni 12.4 Kbit/s dan 12.1 Kbit/s.

Dari hasil perbandingan data *throughput* TCP *New Reno* dan TCP *Westwood+*, maka dapat diambil kesimpulan bahwa *throughput* yang dihasilkan oleh TCP *New Reno* lebih bagus 0.44 Kbit/s daripada *throughput* yang dihasilkan TCP *Westwood+*. Namun jika dilihat dari kinerja saat penambahan *node*, kinerja TCP *New Reno* menurun dengan signifikan daripada kinerja TCP *Westwood+* saat menggunakan jumlah *node* yang lebih besar. Ini dikarenakan kontrol kongesti pada TCP *New Reno* menggunakan teknik AIMD dimana jika terjadi paket yang gagal maka direspon dengan menurunkan *ssthres* dan *cwnd* menjadi setengah lebar *bandwidth*, sedangkan pada TCP *Westwood+* menggunakan teknik AIAD dimana jika terdapat paket yang gagal tidak di respon dengan ekstrim. Sehingga pada *node* 60, 70, 100, dan 120 nilai *throughput* yang dihasilkan TCP *New Reno* menurun drastis karena terdapat paket yang gagal.

4.2 Perbandingan Packet Delivery Ratio



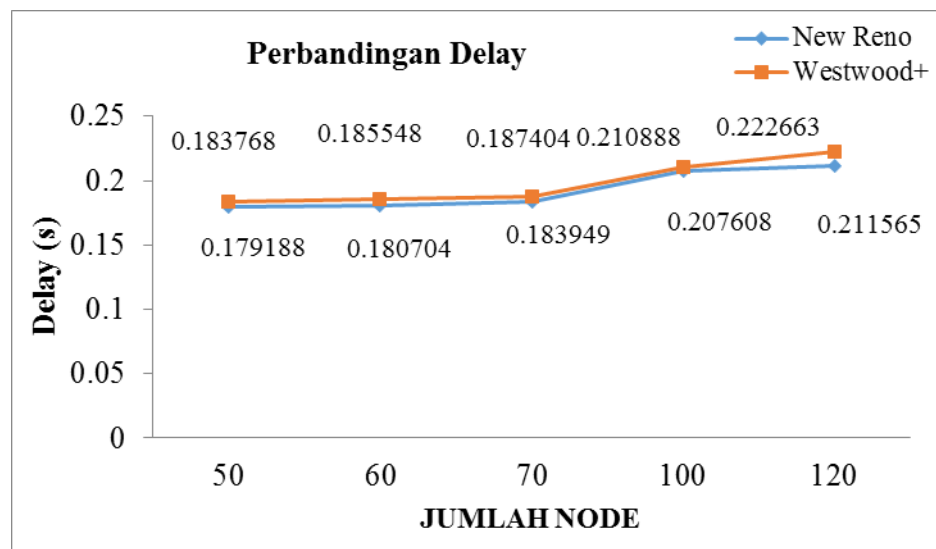
Gambar 4.2 Perbandingan *Packet Delivery Ratio*

Dapat dilihat pada Gambar 4.2 adalah hasil PDR pada percobaan TCP *New Reno* dan TCP *Westwood+*. Pada saat percobaan menggunakan jumlah *node* 50, nilai PDR yang dihasilkan oleh TCP *New Reno* adalah 93%, sedangkan PDR yang dihasilkan oleh TCP *Westwood+* adalah 88%. Kemudian setelah melakukan penambahan *node* mulai dari menggunakan 60 dan 70 *node*, PDR yang dihasilkan TCP *New Reno* menurun 2% hingga 3%, yaitu menjadi 90% dan 88%. Sedangkan pada TCP *Westwood+* juga mengalami penurunan pada saat menggunakan *node* 60 dan 70, PDR yang dihasilkan menjadi 86% dan 85%. Saat percobaan menggunakan 100 *node*, PDR yang dihasilkan oleh TCP *New Reno* dan TCP *Westwood+* kembali mengalami penurunan. PDR yang dihasilkan TCP *New Reno* adalah 83%, sedangkan PDR yang dihasilkan TCP

Westwood+ adalah 79%. Kemudian pada percobaan terakhir saat menggunakan 120 *node*, PDR yang dihasilkan kedua TCP ini bernilai sama yaitu 78%.

Oleh karena itu dapat diambil kesimpulan bahwa PDR yang dihasilkan TCP *New Reno* lebih bagus dibandingkan dengan PDR yang dihasilkan oleh TCP *Westwood+*. Perbedaan PDR yang dihasilkan mencapai 3%. Dapat dilihat pada Gambar 4.2 bahwa, penambahan jumlah *node* sangat mempengaruhi kinerja PDR yang dihasilkan oleh kedua TCP, baik TCP *New Reno* maupun TCP *Westwood+*. Walaupun PDR yang dihasilkan oleh kedua TCP tersebut mengalami penurunan saat penambahan *node*, PDR yang dihasilkan oleh kedua TCP tersebut masih tergolong sedang menurut ETSI, ini menunjukkan bahwa kinerja dari kedua TCP ini pada *packet delivery ratio* tidak buruk.

4.3 Perbandingan Delay



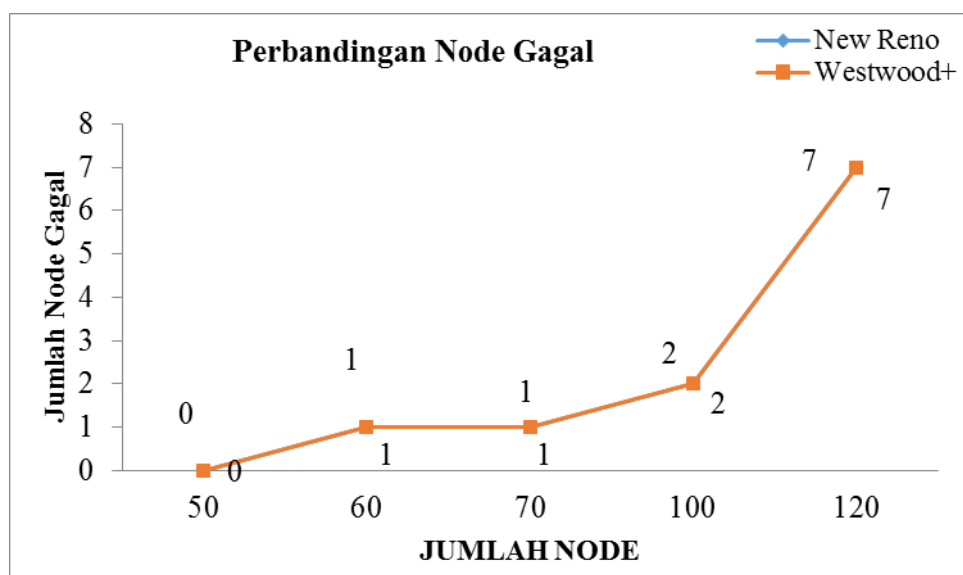
Gambar 4.3 Perbandingan *delay*

Dapat dilihat pada Gambar 4.3 *delay* yang dihasilkan oleh TCP *New Reno* dan TCP *Westwood+*. Pada saat menggunakan TCP *New Reno* dengan jumlah *node* 50, nilai *delay* yang dihasilkan adalah 0.179s, sedangkan *delay* yang dihasilkan TCP *Westwood+* saat jumlah *node* 50 adalah 0,183s. Kemudian *delay* yang dihasilkan TCP *New Reno* mengalami kenaikan menjadi 0.180s setelah melakukan percobaan dengan menambahkan jumlah *node* menjadi 60. Sedangkan *delay* pada TCP *Westwood+* mengalami kenaikan menjadi 0.185s. Kemudian *delay* yang dihasilkan TCP *New Reno* kembali mengalami kenaikan menjadi 0.183s saat menggunakan 70 *node*, dan *delay* yang dihasilkan TCP *Westwood+* juga mengalami kenaikan menjadi 0.187s saat menggunakan 70 *node*. Saat Percobaan menggunakan 100 *node*, kedua TCP *New Reno* dan TCP

Westwood+ mengalami kenaikan *delay* yang cukup besar menjadi 0.207s pada TCP *New Reno* dan 0.210s pada TCP *Westwood+*. Kemudian saat menggunakan 120 *node*, kembali terjadi kenaikan *delay* pada kedua TCP yaitu menjadi 0.211s pada TCP *New Reno* dan 0.222s pada TCP *Westwood+*.

Dari hasil perbandingan *delay* antara TCP *New Reno* dengan TCP *Westwood+* dapat diambil kesimpulan bahwa TCP *Westwood+* memiliki *delay* yang lebih besar daripada TCP *New Reno*. Perbedaan *delay* yang dihasilkan oleh TCP *Westwood+* mencapai 0.005s lebih besar daripada TCP *New Reno*. Ini dikarenakan teknik kontrol kongesti yang digunakan TCP *Westwood+* menggunakan teknik AIAD, dimana teknik AIAD mempunyai *delay* yang cukup tinggi. Walaupun *delay* yang dihasilkan TCP *Westwood+* tidak seburuk *delay* yang dihasilkan TCP *New Reno*, namun menurut parameter ETSI *delay* yang dihasilkan TCP *New Reno* dan TCP *Westwood+* masih tergolong bagus.

4.4 Perbandingan Jumlah Node Gagal



Gambar 4.4 Perbandingan *node* gagal

Pada penelitian ini juga membahas jumlah *node* yang gagal untuk mengirim data kepada *node* yang sudah ditentukan. Dapat dilihat pada Gambar 4.4 adalah grafik perbandingan jumlah *node* yang gagal pada TCP *New Reno* dan TCP *Westwood+*, pada saat menggunakan jumlah *node* 50, seluruh *node* pada TCP *New Reno* maupun TCP *Westwood+* berhasil menjalankan tugasnya untuk mengirim data kepada *node* tujuan. Namun pada saat TCP *New Reno* menggunakan jumlah *node* 60 terdapat 1 *node* yang gagal mengirim data kepada *node* yang dituju, begitu juga pada TCP *Westwood+* terdapat 1 *node* yang gagal mengirimkan data saat

menggunakan *node* 60. Pada percobaan selanjutnya saat menggunakan 70 *node*, pada algoritme TCP *New Reno* kembali terdapat 1 *node* yang gagal mengirimkan data. Begitu juga pada TCP *Westwood+* terdapat 1 *node* yang gagal mengirimkan data kepada tujuan saat menggunakan 70 *node*. Pada percobaan menggunakan jumlah *node* 100, *node* yang gagal bertambah menjadi 2 *node* pada TCP *New Reno* maupun TCP *Westwood+*. Saat menggunakan jumlah *node* 120, *node* yang gagal bertambah menjadi 7 *node* pada TCP *New Reno* dan TCP *Westwood+*.

Dari data yang didapatkan dapat diambil kesimpulan bahwa jumlah *node* yang gagal pada TCP *New Reno* dan TCP *Westwood+* sama, yaitu pada saat menggunakan jumlah *node* 60 dan 70, jumlah *node* yang gagal pada masing-masing TCP ialah 1 *node*. Pada saat menggunakan jumlah *node* 100, jumlah *node* yang gagal pada masing-masing TCP ialah 2 *node*. Kemudian pada saat menggunakan 120 *node*, jumlah *node* yang gagal menjadi 7 pada masing-masing TCP. Ini menandakan bahwa jika semakin banyak *node* yang digunakan maka, kemungkinan jumlah *node* yang gagal pada kedua TCP semakin bertambah, begitu juga sebaliknya jika semakin sedikit *node* yang digunakan maka kemungkinan jumlah *node* yang gagal berkurang.

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah melakukan penelitian ini, maka dapat diambil kesimpulan bahwa :

1. Semakin banyak jumlah *node* mempengaruhi kinerja masing-masing TCP.
2. *Throughput* yang dihasilkan TCP *New Reno* lebih bagus 0.44 Kbit/s dibandingkan dengan TCP *Westwood+*.
3. PDR yang dihasilkan oleh TCP *New Reno* lebih bagus 3% daripada PDR yang dihasilkan oleh TCP *Westwood+* dan menurut parameter ETSI, PDR yang dihasilkan kedua TCP ini tergolong sedang.
4. *Delay* yang dihasilkan TCP *Westwood+* lebih besar 0.005s dibandingkan *delay* yang dihasilkan TCP *New Reno*. Namun menurut parameter ETSI, *delay* yang dihasilkan kedua TCP pada semua jumlah *node* masih tergolong bagus.
5. Jumlah *node* yang gagal pada kedua TCP sama.
6. Dari butir 1 hingga 6 dapat disimpulkan bahwa TCP *New Reno* merupakan TCP yang lebih bagus dalam mereduksi kongesti pada jaringan wifi dibandingkan TCP *Westwood+*.

5.2 Saran

Setelah melakukan penelitian ini, penulis memiliki beberapa saran untuk penelitian selanjutnya agar mampu menambah performa dari sistem ini :

1. Model WLAN yang digunakan pada penelitian ini menggunakan IEEE 802.11a, sebaiknya penelitian selanjutnya menggunakan IEEE 802.11g untuk menambah jarak lingkup wifi.
2. Penelitian ini menggunakan WLAN, sebaiknya penelitian selanjutnya menggunakan LTE agar mampu mengetahui performa kedua TCP pada jaringan LTE.

DAFTAR PUSTAKA

- [1] E. N. I. Nurkayati, “Sejarah wifi,” 2014.
- [2] A. Insani, “Studi Awal Teknologi WIFI Untuk Diimplementasikan Pada Pembuatan Prototipe Sistem Remote Terminal Unit Multi Sensor Dengan Energi Mandiri,” vol. 10, no. 3, pp. 225–240, 2012.
- [3] elib unikom, “Sejarah Dan Perkembangan WIFI (Wireless Fidelity).”
- [4] P. H. Rusmin, C. Machbub, and A. Harsoyo, “Sistem Kendali Kongesti Di Internet,” vol. 12, no. 1, pp. 19–26, 2008.
- [5] S. A. Wahyu Putri Utami Yuliana, Jusak, “Analisis Perbandingan Unjuk Kerja Algoritma Congestion Control Pada Tcp Tahoe , Reno Dan Sack (Selective Acknowledgment),” pp. 1–12, 2008.
- [6] H. Hilary Zhang, “Evaluation of Different Tcp Congestion Control Algorithms Using Ns-2,” 2002.
- [7] A. G. Fallis, “Perbandingan kinerja TCP Westwood+ dan TCP New Reno pada jaringan WIMAX,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.
- [8] A. Tanzila Harfard, “Analisa Kinerja Algoritma TCP Congestion Control Cubic, Reno, Vegas dan Westwood+,” vol. 2, no. 3, pp. 1099–1108, 2018.
- [9] R. Fahrizal, W. Dewanto, and S. Sumaryono, “Perbandingan Kinerja Algoritma Bic , Cubic Dan Htcp Pada Topologi Parkinglot Dan Multihome,” vol. 2012, no. semnasIF, pp. 38–44, 2012.
- [10] R. Kukuh, “Analisis Perbandingan unjuk kerja TCP Tahoe dan TCP New Reno pada Jaringan wired dan wireless,” 2017.
- [11] R. Inelia, S. N. Hertiana, and I. Wahidah, “Analisis dan simulasi perfomansi kendali kongesti dengan skema pgmcc,” pp. 1–15, 2011.
- [12] G. Sukadarmika, N. I. Er, and N. W. Saputra, “Analisis Coverage WLAN (Wireless Local Area Network) 802.11a menggunakan Opnet Modeler,” vol. 9, no. 2, 2010.
- [13] S. A. Sarah, “IEEE 802.11g,” pp. 2–6, 2003.
- [14] F. Helmy, “Simulasi Kinerja Jaringan Nirkabel IEEE-802.11a dan IEEE-802.11g

Menggunakan NS-2,” vol. 10, no. 4, 2013.

- [15] D. T. Britt and C. Matthews, “TCP / IP Tutorial and Technical Overview,” *Contract*, vol. 1, no. December 2006, p. 1004, 2006.
- [16] S. Kam and A. Takagi, “File Transfer And Access (FTP , TFTP , NFS) History of FTP.”
- [17] Ma. Nathalia, “Analisis Unjuk Kerja TCP Reno di jaringan Single HOP Wireless Link,” 2016.
- [18] T. Y. Arif, “Simulasi Throughput WiFi menggunakan Model,” 2014.
- [19] B. Raman, “Introduction 802.11,” no. May, pp. 9–13, 2005.
- [20] R. Wulandari, “Analisis QoS (Quality of Service) Pada Jaringan Internet (Studi Kasus : UPT Loka Uji Teknik Penambangan Jampang Kulon - LIPI),” *J. Tek. Inform. dan Sist. Inf.*, vol. 2, pp. 162–172, 2016.

LAMPIRAN