

ANALISIS KINERJA JARINGAN *WIRELESS LAN* BERDASARKAN MEKANISME *LOAD BALANCING* DENGAN ALGORITMA *LEAST CONNECTION* MENGGUNAKAN *SIMULATOR OPNET 14.5*

Hauril Maulida Nisfari^{*)}, Sukiswo, and Ajub Ajulian Zahra

Jurusan Teknik Elektro, Universitas Diponegoro Semarang
Jl. Prof. Sudharto, SH, kampus UNDIP Tembalang , Semarang 50275, Indonesia

^{*)} Email: hauril.nisfari@gmail.com

Abstrak

Penggunaan teknologi Wireless sudah tidak bisa dielakkan lagi saat ini. Teknologi Wireless yang banyak digunakan saat ini adalah WLAN (Wireless Local Area Network). Peningkatan jumlah user ini tentu saja berpengaruh terhadap kinerja server sebagai penyedia layanan. Mekanisme Load Balancing adalah salah satu strategi yang digunakan oleh pengelola jaringan untuk menyeimbangkan beban antar server. Pada penelitian ini dirancang sebuah jaringan Wireless LAN yang menerapkan mekanisme Load Balancing beserta pengaturan algoritma Least Connection menggunakan software OPNET 14.5. Perancangan dilakukan untuk membandingkan kinerja layanan WLAN tanpa Load Balancing dan dengan Load Balancing dengan parameter-parameter yang digunakan adalah throughput, delay jaringan, Response Time dan fairness index untuk tiap pengguna aplikasi yang berbeda. Dari hasil simulasi diperoleh nilai Throughput pada jaringan dengan Load Balancing untuk layanan HTTP berkisar antara 100-120 Kbps, untuk layanan FTP antara 80-100 Kbps dan untuk Email berkisar 50-60 Kbps. Algoritma Least Connection memberikan cukup adil alokasi pembagian lebar pita, dibuktikan dengan fairness index untuk ketiga skenario berkisar 0,7-0,8 dibandingkan dengan jaringan tanpa Load Balancing dengan fairness index berkisar 0,4-0,7. Mekanisme Load Balancing bekerja optimal pada skenario 3 dengan Delay dan Response Time yang lebih kecil.

Kata kunci : OPNET Modeler v14.5, Load Balancing, Least Connection Algorithm

Abstract

The use of wireless technology has become inevitable at this time. Wireless technology that is widely used today is WLAN (Wireless Local Area Network). Load Balancing Mechanism is one of the strategies used by network managers to balance the load among servers. In this research has been designed a wireless LAN network that implements Load Balancing mechanism and its Least Connection algorithm settings using OPNET 14.5 software. The design is performed to compare the performance of WLAN services without Load Balancing and with Load Balancing with parameters used are the throughput, the network delay, response time and fairness index for each user of different applications. From the simulation results obtained that the throughput on the network with Load Balancing for HTTP service ranges from 100-120 Kbps, for FTP services between 80-100 Kbps and for Email ranged 50-60 Kbps. Least Connection algorithm provides fair allocation of bandwidth sharing, evidenced by the fairness index for the three scenarios range 0.7-0.8 compared to networks without Load Balancing with fairness index ranges from 0.4-0.7. Load Balancing mechanism works optimally in Scenario 3 with smaller Response and Time Delay.

Keywords : OPNET Modeler v14.5, Load Balancing, Least Connection Algorithm

1. Pendahuluan

Saat ini WLAN telah banyak digunakan untuk layanan akses internet *broadband* nirkabel. WLAN telah menjadi pilihan, terlihat dari banyaknya WLAN yang terpasang di berbagai tempat seperti sekolah, kampus, kafe, perpustakaan dan tempat umum lainnya. Jumlah

penggunanya cenderung meningkat dari tahun ke tahun[1].

Disamping dampak positif terhadap mudahnya akses informasi, didapatkan pengetahuan dengan bertambahnya pengguna WLAN setiap waktunya, menyebabkan terjadinya peningkatan akses informasi yang penting oleh semua orang. Peningkatan jumlah pengguna ini tentu saja berpengaruh terhadap kinerja *server* sebagai penyedia

layanan. *Server* yang pada mulanya hanya bisa melayani koneksi beberapa klien saja, kini dipaksa harus melayani banyak klien. Oleh karena itu, dibutuhkan suatu mekanisme yang mengatur kinerja *server* untuk melayani klien dengan seimbang. Mekanisme *Load Balancing* adalah salah satu strategi yang digunakan oleh pengelola jaringan untuk menyeimbangkan beban antar server, sehingga tidak terjadi *overload* pada salah satu *server*[2]. Dalam penelitian ini akan dilakukan perancangan jaringan WLAN berdasarkan Mekanisme *Load Balancing* dengan Algoritma *Least Connection* menggunakan Simulator OPNET 14.5[3]. OPNET adalah salah satu software yang banyak digunakan pengelola jaringan untuk melakukan simulasi perancangan, analisis maupun prediksi jaringan.

2 Metode

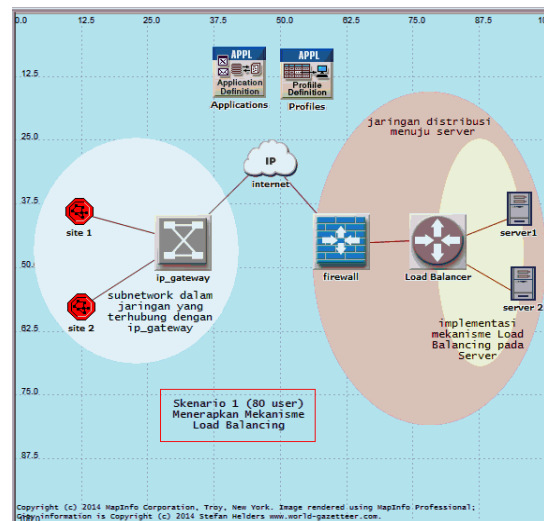
2.1 Simulasi Jaringan WLAN

Simulasi yang dilakukan pada Penelitian ini menggunakan tiga skenario yang berbeda, yang membedakan antara skenario satu dengan yang lainnya adalah jumlah pengguna yang digunakan. Aplikasi yang dijalankan yaitu HTTP, FTP dan *Email*. Pada Skenario 1 terdapat 80 pengguna dengan rincian 30 akses ke layanan HTTP, 24 akses ke layanan FTP dan 26 akses ke layanan *Email*. Sedangkan pada Skenario 2 terdapat 128 pengguna yang juga menggunakan satu layanan, yang terdiri dari 41 akses ke layanan HTTP, 45 akses ke layanan FTP dan 42 akses ke layanan *Email*. Dan Skenario 3 terdapat 160 pengguna yang menggunakan satu layanan, yang terdiri dari 53 akses ke layanan HTTP, 52 akses ke layanan FTP dan 55 akses ke layanan *Email*.

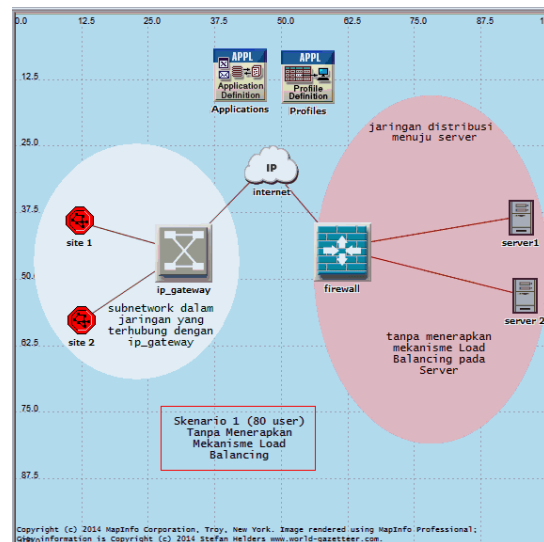
Gambar 1 dan 2 merupakan tampilan skenario yang dirancang pada Penelitian ini, yang merupakan perancangan topologi *Wireless LAN* dengan mekanisme *Load Balancing* dan tanpa *Load Balancing* pada simulator OPNET. Perbedaan kedua topologi tersebut adalah adanya penambahan node *Load Balancer* pada jaringan dengan mekanisme *Load Balancing*.

Topologi yang di buat terdiri dari 2 server dan 2 site masing-masing terdapat 8 *Access Point*. Untuk masing-masing topologi di atas selain node *Load Balancer*, kedua topologi tersebut menggunakan parameter yang sama agar bisa dibandingkan performansi kinerjanya.

Masing – masing pengguna pada topologi ini hanya menggunakan 1 jenis aplikasi. Untuk setting mekanisme *Load Balancing* ini, semua permintaan dari *client* akan diteruskan ke node *Load Balancer* terlebih dahulu, selanjutnya akan di atur dan diarahkan ke server mana yang akan melayani, sesuai dengan algoritma yang digunakan.



Gambar 1 Skenario dengan *Load Balancing*



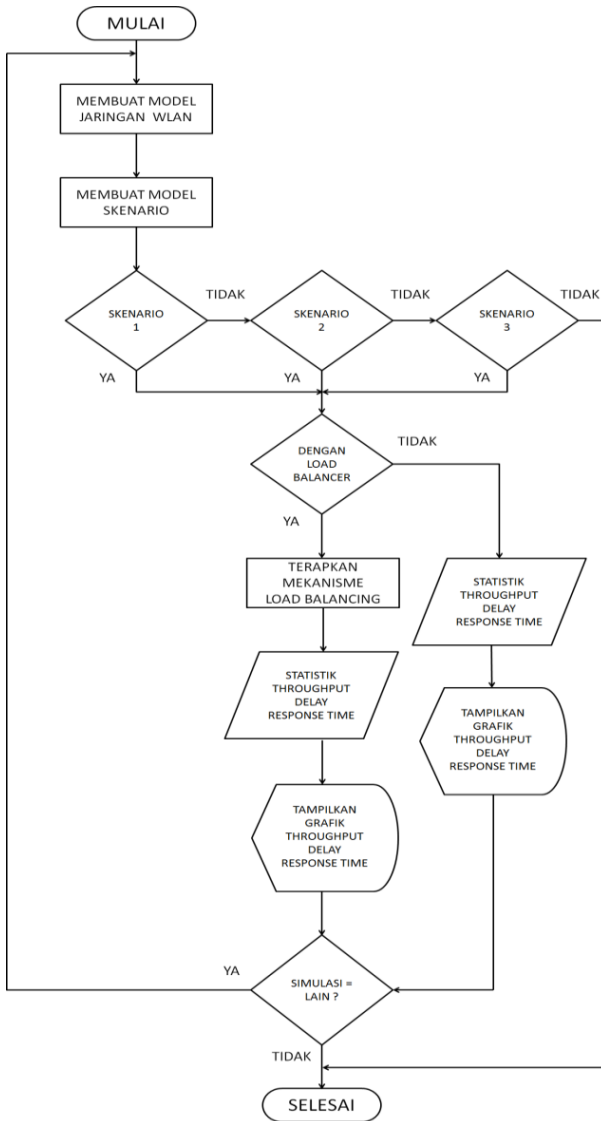
Gambar 2 Skenario tanpa *Load Balancing*

2.2 Perancangan Sistem

Pada Penelitian ini dibuat suatu jaringan WLAN dengan menggunakan simulator OPNET modeler v14.5. Diagram alir perancangan dapat dilihat pada Gambar 3. Dari Gambar 3 dapat diketahui bahwa perancangan simulasi diawali dengan membuat model jaringan *Wireless LAN*. Dalam Penelitian ini menggunakan *Wireless LAN* dengan standar 802.11g. Kemudian membuat beberapa skenario berdasarkan jumlah klien. Skenario 1 adalah skenario dengan jumlah klien 80, skenario 2 adalah skenario dengan jumlah klien 128 dan skenario 3 adalah skenario dengan jumlah klien 160. Ketiga skenario tersebut akan dijalankan dengan mengimplementasikan *Load Balancer* dan tanpa mengimplementasikan *Load Balancer*. Hasil simulasi yang didapat berupa *Throughput*, *Delay* dan *Response Time*.

Alur kerja penggunaan OPNET dalam Penelitian ini, yaitu :

1. Membuat Model Jaringan
2. Memilih Statistik
3. Menjalankan Simulasi
4. Mengamati dan Menganalisis Hasil



Gambar 3 Diagram alir simulasi

3 Hasil dan Analisis

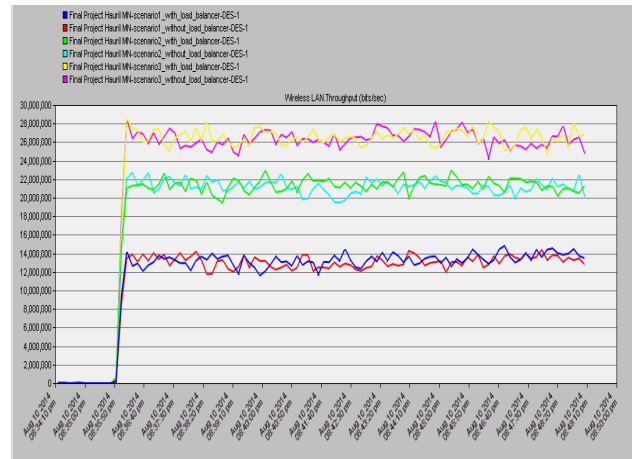
3.1 Throughput

Setelah dilakukan simulasi pada ketiga skenario didapatkan hasil seperti gambar 4. Dari gambar 4 dapat dilihat perolehan *Throughput Global* yang ditunjukkan pada tabel 1.

Berdasarkan perolehan *Throughput* pada tabel 1, skenario 2 dan 3 memperoleh total *Throughput* pada jaringan yang menerapkan mekanisme *Load Balancing*

lebih besar di banding tanpa *Load Balancing*. Hal ini menunjukkan bahwa kinerja *Load Balancing* akan bagus jika diimplementasikan pada jaringan trafik padat.

Algoritma *Least Connection* mengarahkan koneksi ke server yang tersedia sehingga bisa mengoptimalkan jalur koneksi dan menyebabkan jumlah *Throughput* yang diterimapun bisa maksimal.



Gambar 4 Grafik *Throughput* pada Skenario 1

Tabel 1 *Throughput Global*

Skenario	Dengan LB	Tanpa LB
1	11,687	11,487
2	18,751	18,599
3	23,281	23,095

3.2 Waktu Tunda (Delay)

Hasil total *delay* (waktu tunda) rata-rata jaringan dapat dilihat pada tabel 2.

Tabel 2 Rata-rata total *delay* ketiga skenario

Skenario	Dengan LB	Tanpa LB
1	0.00126	0.00129
2	0.00136	0.00140
3	0.00146	0.00150

Dari tabel 2, dapat disimpulkan bahwa untuk ketiga skenario, rata-rata *delay* total pada jaringan dengan *Load Balancing* lebih kecil di banding tanpa *Load Balancing*. Hal ini disebabkan adanya *delay* proses, khususnya *delay* antrian pada jaringan tanpa *Load Balancing* lebih besar karena tidak adanya metode antrian seperti halnya yang ada pada jaringan dengan *Load Balancing*, sehingga dapat terjadi adanya kemacetan trafik. Sedangkan algoritma *Least Connection* dalam *Load Balancing* ini akan mengatur dan mengarahkan jalannya trafik yang masuk ke server sehingga dapat mencegah terjadinya *congestion* yang menyebabkan *delay* dan *collision* yang menyebabkan *Throughput* rendah.

3.3 Response Time

Tabel 3,4 dan 5 berikut merupakan hasil simulasi dari ketiga skenario.

Tabel 3 Response Time Aplikasi Skenario 1

Skenario 1 (80 User)	Dengan LB	Tanpa LB
HTTP Object Response Time (sec)	0.0101	0.0108
FTP DL Response Time (sec)	0.0836	0.0852
FTP UL Response Time (sec)	0.0831	0.0795
Email DL Response Time (sec)	0.0231	0.0242
Email UL Response Time (sec)	0.0194	0.0196

Tabel 4 Response Time Aplikasi Skenario 2

Skenario 2 (128 User)	Dengan LB	Tanpa LB
HTTP Object Response Time (sec)	0.0110	0.0116
FTP DL Response Time (sec)	0.0895	0.0919
FTP UL Response Time (sec)	0.0910	0.0946
Email DL Response Time (sec)	0.0283	0.0273
Email UL Response Time (sec)	0.0241	0.0235

Tabel 5 Response Time Aplikasi Skenario 3

Skenario 3 (160 User)	Dengan LB	Tanpa LB
HTTP Object Response Time (sec)	0.0114	0.0124
FTP DL Response Time (sec)	0.0937	0.0972
FTP UL Response Time (sec)	0.0969	0.0980
Email DL Response Time (sec)	0.0293	0.0298
Email UL Response Time (sec)	0.0253	0.0285

Pada tabel 3,4 dan 5 menunjukkan bahwa jaringan dengan *Load Balancing* telah bekerja optimal pada skenario 3 di lihat dari *Response Time* pada ketiga aplikasi memiliki nilai lebih kecil dibandingkan pada jaringan tanpa *Load Balancing*. Hal ini disebabkan karena terjadinya kemacetan trafik, sehingga waktu antrian pada jaringan tanpa *Load Balancing* lebih besarseiring dengan meningkatnya trafik. Sedangkan algoritma *Least Connection* pada *Load Balancing* akan mencegah terjadinya penumpukan data dan membagi beban secara *service dedicated*.

3.4 Fairness Index

Nilai *Fairness Index IntraClass* pada ketiga aplikasi berbeda dapat dihitung dengan rumus :

$$\text{MinMax Fairness} = \frac{\text{Throughput}_{\min}}{\text{Throughput}_{\max}}$$

Hasil *fairness index* dapat di lihat pada tabel 6 berikut.

Tabel 6 Fairness Index pada ke-3 skenario

Fairness Index (IntraClass)	Skenario 1	
	Dengan LB	Tanpa LB
HTTP	0,78	0,65
FTP	0,70	0,70
Email	0,83	0,54

Fairness Index (IntraClass)	Skenario 2	
	Dengan LB	Tanpa LB
HTTP	0,77	0,80
FTP	0,73	0,44
Email	0,78	0,75

Fairness Index (IntraClass)	Skenario 3	
	Dengan LB	Tanpa LB
HTTP	0,83	0,64
FTP	0,70	0,74
Email	0,77	0,78

Dari tabel 6 di atas, secara *Global Indeks* keseimbangan pada jaringan dengan *Load Balancing* lebih besar atau stabil dibanding jaringan tanpa *Load Balancing*, terbukti pada indeks keseimbangan yang bernilai 0,44 di jaringan tanpa *Load Balancing*. Hal ini disebabkan karena algoritma *Least Connection* pada *Load Balancing* menangani tiap permintaan klien untuk disalurkan ke server dengan mengoptimalkan jalur yang ada dan menjamin ketersediaan layanan, sehingga meminimalisir terjadinya paket yang *drop* pada klien.

Algoritma *Least Connection* ini memberikan efisiensi waktu tunggu dan jumlah *Throughput* yang lebih seimbang untuk tiap klien dibandingkan jaringan tanpa *Load Balancing*.

4 Kesimpulan

Kesimpulan yang dapat diambil dari penelitian ini adalah bahwa *Throughput* pada jaringan dengan *Load Balancing* relatif lebih seimbang dibandingkan dengan tanpa *Load Balancing* dibuktikan dengan *fairness index* berkisar 0,7-0,8, sedangkan jaringan tanpa *Load Balancing* hanya berkisar 0,4-0,7. Pada ketiga skenario, kinerja *Load Balancing* lebih optimal pada skenario 3, dibuktikan dengan nilai *delay* dan *response time* yang lebih kecil dibandingkan dengan jaringan tanpa *Load Balancing*. Penggunaan *Load Balancing* akan lebih terlihat kinerjanya pada jaringan dengan trafik yang padat. Algoritma *Least Connection* pada mekanisme ini akan bekerja dengan baik ketika beberapa server yang digunakan memiliki *resource* atau kemampuan yang sama

Referensi

- [1]. Singh, Jasmeet. "Quality of Service in Wireless LAN Using OPNET MODELER". Thesis, Computer Science and Engineering Department Thapar University. 2009
- [2]. Sundawa, Bakti Viyata. "Peningkatan Kinerja Jaringan WLAN Dengan Load Balancing Menggunakan Teknologi Agent". Laporan Tesis Teknik Elektro Universitas Sumatera Utara. 2012
- [3]. Bourke, T., "Server Load Balancing", Published by O'Reilly & Associates, Inc., 101 Morris Street, Sebastopol, 2001.

- [4]. Koppurapu, Chandra, "Load Balancing Servers, Firewalls, and Caches", Published by John Wiley & Sons, Inc., 605 Third Avenue, New York, 2002.
- [5]. Geier, Jim, "Wireless Networks first-step" (edisi bahasa Indonesia), Penerbit Andi, Yogyakarta, 2005.
- [6]. Tri Kuntoro Priyambodo dan Dodi Heriadi, "Jaringan Wi-Fi, Teori dan Implementasi", Penerbit Andi, Yogyakarta, 2005.
- [7]. Sofana, Iwan, "CISCO CCNA & Jaringan Komputer", Penerbit Informatika, Bandung, 2012.
- [8]. Rajendra K. Jain, Dah-Ming W, dan William R. Hawe. "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System. Digital Equipment Corporation, 77 Reed Road (HLO2-3/N03) Hudson, MA 01749, 1984
- [9]. Gu, Qianping, "Delay Analysis". CMPT765/408 08-1
- [10]. Manju Sharma, Manoj Kumar, Ajay K Sharma. "HTTP and FTP Statistics for Wireless and Wire-Line Network With and Without Load Balance Based on Opnet". International Journal of Information and Systems Sciences Volume 5, Number 1, Pages 112-125. 2009
- [11]. "_____", IEEE 802.11. http://en.wikipedia.org/wiki/IEEE_802.11, diakses pada tanggal 23 juli 2014
- [12]. "_____", The Least Connection Method. <http://support.citrix.com/proddocs/topic/netScaler-load-balancing-93/ns-lb-customizing-about-leastconnection-con.html>, diakses tanggal 6 Juni 2014
- [13]. Rajan Vohra, R.S Sawheney, Sunandika Mann. *Statistics Comparison in Wireless Network Environment for Balanced and Unbalanced Network*. International Conference on Recent Advances and Future Trends in Information Technology (iRAFIT2012)