

PEMILIHAN NODE TETANGGA YANG HANDAL DENGAN MEMPERHITUNGGAN SIGNAL STRENGTH DAN LINK QUALITY PADA ZONE ROUTING PROTOCOL DI LINGKUNGAN MANET

Ida Bagus Ary Indra Iswara¹, Waskitho Wibisono²

¹ Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember
Kampus ITS Keputih Sukolilo, Surabaya, Jawa Timur

² Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember
Kampus ITS Keputih Sukolilo, Surabaya, Jawa Timur
Email: aryindraiswara@gmail.com

ABSTRACT

Zone Routing Protocol (ZRP) is a hybrid routing protocol on MANET, because the routing protocol is based on two protocols, namely IntraZone Routing Protocol (IARP) and InterZone Routing Protocol (IERP). Through IARP, each node identifies distance to all nodes in its routing zone. Although the network is very large, the update process is only distributed locally in its routing zone only and not the whole network. While IERP protocol is responsible to find a route to a node that is located outside the zone. The selection of neighboring nodes is completely done by Neighbor Discovery Protocol (NDP), which is performed by the node selection NDP is sending beacons broadcast message. NDP only uses TTL to decide whether the neighboring nodes are broken or maintained.

In this research, the method of signal strength based link-sensing stability adaptation is integrated into the work process of the NDP. This method works based on the signal strength of a node. Signal strength of a node is reliable if the node does not exceed the predetermined threshold value. Reliable nodes that will be stored in the neighbor table and will be a reference to the IARP routing process. Nodes that are not reliable will be disconnected and will not send beacons at the message again.

The selection of reliable neighbor nodes by calculating signal strength and link quality is able to improve the performance of ZRP routing protocols. Protocols tested in scenarios designed by the wide variation area. The test results showed an increase of 4.13 kbps increase in throughput, a decrease of 9.98 ms *e2e* delay and routing overhead decreased by 3.83%.

Keywords: MANET, ZRP, Link Quality, Signal Strength, JiST/SWANS

ABSTRAK

Zone Routing Protocol (ZRP) merupakan protokol routing hybrid pada MANET, karena protokol routing ini didasarkan pada dua protokol yaitu *IntraZone Routing Protokol* (IARP) dan *InterZone Routing Protokol* (IERP). Melalui IARP, setiap node mengidentifikasi jarak ke semua node di zona routing-nya. Meskipun jaringannya sangat besar, proses update hanya disebar secara lokal pada zona routing-nya saja dan tidak pada keseluruhan jaringan. Sedangkan protokol IERP bertanggung jawab untuk menemukan rute ke node yang terletak di luar zona. Pemilihan node tetangga sepenuhnya dilakukan oleh *Neighbor Discovery Protocol* (NDP), pemilihan node yang dilakukan oleh NDP adalah mem-broadcast beacons message. NDP hanya memanfaatkan TTL untuk memutuskan apakah node tetangga tersebut di putus atau dipertahankan.

Pada penelitian ini dilakukan adaptasi metode *signal strength based link-stability sensing*, ke dalam proses kerja NDP. Metode ini bekerja berdasarkan dari kekuatan signal suatu node. Kekuatan signal (*signal strength*) suatu node dikatakan *reliable* apabila node tersebut tidak melebihi nilai *threshold* yang sudah ditentukan. Node yang *reliable* akan disimpan pada tabel *neighbor* dan selanjutnya akan dijadikan acuan untuk melakukan proses routing pada IARP. Node yang tidak *reliable* akan diputus dan tidak akan di kirim *beacons message* lagi.

Pemilihan node tetangga yang handal dengan memperhitungkan *signal strength* dan *link quality* mampu memperbaiki kinerja dari protokol routing ZRP. Protokol diujikan ke dalam skenario yang dirancang berdasarkan variasi luas area. Hasil pengujian menunjukkan peningkatan peningkatan *throughput* sebesar 4.13 kbps, penurunan *e2e delay* sebesar 9.98 ms dan penurunan *routing overhead* sebesar 3.83 %.

Kata kunci: MANET, ZRP, Link Quality, Signal Strength, JiST/SWANS

1 PENDAHULUAN

Mobile Ad-hoc Networks (MANET) terdiri dari *host mobile* yang dapat berkomunikasi satu sama lain menggunakan *link* nirkabel. Dalam lingkungan ini rute antara dua *host* dapat terdiri dari banyak hop melalui satu atau lebih *node* di MANET (Ravilla dan Putta C., 2012). *Node* pada MANET memiliki beberapa kekurangan, seperti *bandwidth* terbatas, perangkat *power* yang rendah, perubahan topologi jaringan yang dinamis. (Jain, dkk., 2012). Komunikasi *link* di MANET juga rentan dan sensitif terhadap karakteristik saluran yang bervariasi. Pembentukan rute dan rekonstruksi rute merupakan dua fase penting dalam MANET yang harus ditangani dengan hati-hati (C. Biradar dan S. Manvi, 2012). MANET akan mendapatkan kinerja yang maksimal apabila setiap *node* dapat memanfaatkan sumber dayanya sendiri dan sebagai gantinya juga dapat memanfaatkan sumber daya *node* yang lain (Wrona dan Mahonen, 2004)

Routing pada MANET secara garis besar dibagi kedalam tiga kelompok yaitu proaktif *routing*, reaktif *routing* dan *hybrid routing* (Asha, dkk., 2012). Protokol routing proaktif contohnya adalah DSDV, OLSR., protokol routing yang reaktif antara lain AODV, ODMRP, dan AOMDV, sedangkan protokol *routing hybrid* merupakan protokol *routing* yang menggabungkan kedua kelebihan dari proaktif dan reaktif *routing*, salah satu protokol *routing hybrid* adalah *Zone Routing Protocol* (ZRP).

Zone Routing Protocol (ZRP) merupakan protokol routing *hybrid*, karena protokol *routing* ini didasarkan pada dua protokol yaitu *IntraZone Routing Protokol* (IARP) dan *InterZone Routing Protokol* (IERP) (J. Haas, dkk., 2003). Melalui IARP, setiap *node* mengidentifikasi jarak ke semua *node* di zona routing-nya. Meskipun jaringannya sangat besar, proses update hanya disebar secara local pada zona routing-nya saja dan tidak pada keseluruhan jaringan. Sedangkan protokol IERP bertanggung jawab untuk menemukan rute ke *node* yang terletak di luar zona (Beijar, 2002). Selain kedua protokol tersebut ZRP juga didukung *Broadcast Resolution Protocol* (BRP) yang bertugas sebagai perantara komunikasi antara IARP dengan IERP dan *Neighbor Routing Protocol* (NDP) yang bertugas untuk mencari dan mencatat *node* tetangga yang ada pada IARP (Beijar, 2002)

Neighbor Routing Protocol (NDP) memiliki peranan yang sangat penting pada proses *routing* di ZRP. NDP merupakan sumber informasi dari keberadaan *node-node* yang berada pada *zone* IARP. NDP melakukan pendataan *node* tetangga dengan melakukan *broadcast beacons message* yang disebar ketetangga yang terkoneksi, *beacon message* yang dikirim berisikan *MAC Address*, *IP Address*, *Destination*, *Source*, apabila *broadcast*

beacons message mendapatkan jawaban maka jawaban tersebut dijadikan indikator bahwa *node* tersebut merupakan tetangga dan akan disimpan pada tabel *neighbor* yang dimiliki NDP. *Node* dianggap tidak valid apabila *node* tersebut tidak mengirimkan balasan selama batasan waktu yang dialokasikan, dan tabel tersebutlah yang akan menjadi acuan bagi IARP untuk melakukan proses *routing* (Beijar, 2002; Shafiq, Mahmud, Khan, Sayyed, & Al-Raweshidy, 2012).

Mengingat MANET adalah jaringan yang bersifat *mobile* maka *node-node* yang berada pada suatu *zone* di ZRP sangat mungkin mengalami perubahan tempat. Perubahan tempat suatu *node* akan berdampak pada koneksi antar *node* yang sebelumnya sudah terbangun. Sistem *discovery node* yang ditawarkan oleh ZRP klasik adalah NDP, NDP seperti dijelaskan diatas hanya menampung MAC, dan IP Address dari *node* tetangga, hal ini akan menyebabkan sebuah masalah ketika *node-node* yang berada dalam *zone* mengalami pergerakan. NDP hanya mampu mengetahui IP dan MAC saja dan tidak mengetahui apakah *node* tersebut masih bisa menerima *beacons message* atau tidak. Indikator yang digunakan NDP saat ini untuk menentukan apakah sebuah *node* merupakan tetangga yang valid atau tidak adalah menunggu balasan *beacons message* yang dikirimkan selama batas waktu yang ditentukan (Beijar, 2002). Berdasarkan hal yang disebutkan diatas maka peneliti mengusulkan penerapan metode baru untuk melakukan pemilihan *node* tetangga dengan *signal strength based link-stability sensing*. Pemilihan *node* dengan metode ini berdasarkan dari kekuatan signal suatu *node*. Kekuatan signal (*signal strength*) Suatu *node* dikatakan *reliable* apabila *node* tersebut tidak melebihi nilai *threshold* yang sudah ditentukan. *Node* yang *reliable* akan disimpan pada tabel *neighbor* dan selanjutnya akan dijadikan acuan untuk melakukan proses *routing* pada IARP. *Node* yang tidak *reliable* akan diputus dan tidak akan dikirimkan *beacons message* lagi. Dengan metode *signal strength based link stability sensing* diharapkan dapat meningkatkan kinerja dari protokol *routing* ZRP.

2 METODE SIGNAL STRENGTH LINK-STABILITY SENSING

Metode *signal strength based link-stability sensing* dibagi dalam tiga tahapan yaitu tahap perhitungan *signal strength* tahap penentuan *threshold* dan tahap perhitungan *link quality*. Tahap-tahap tersebut akan diterangkan lebih rinci pada sub-bab berikut.

2.1 Tahap Perhitungan Signal Strength

Perhitungan *signal strength* yang digunakan pada penelitian ini adalah dengan menggunakan persamaan dari *two ray ground reflection model*. *Two ray ground reflection model* mempertimbangkan jalur langsung dan jalur refleksi tanah. Hal ini menunjukkan bahwa model ini memberikan prediksi yang lebih akurat pada jarak jauh dibandingkan dengan model ruang bebas.

Penelitian ini menggunakan Persamaan 1 untuk melakukan perhitungan *signal strength*. Persamaan 1 akan berdampak pada *signal strength* dari suatu *node* ketika nilai *d* semakin besar maka *ss* akan menjadi semakin kecil. Oleh karena itu dibuatlah *threshold* untuk membatasi suatu *node* apakah masih bisa berkomunikasi atau tidak.

$$(ss) = \frac{1}{d^4}$$

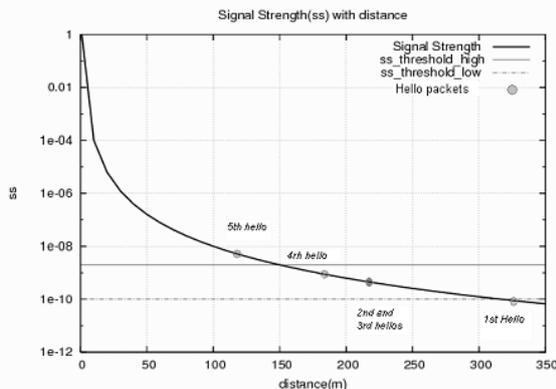
Dimana :

ss = Signal Strength

d = Jarak antar *node*

Hasil yang didapatkan dari perhitungan dengan menggunakan Persamaan 1 digunakan sebagai acuan untuk digunakan pada perhitungan *link-quality*.

Perhitungan *link-quality* sangat berpengaruh dari hasil perhitungan *signal strength* dengan menggunakan Persamaan 1. Hasil dari perhitungan menggunakan Persamaan 1 dapat dilihat pada Gambar 1, pada Gambar tersebut diperlihatkan penurunan kualitas signal seiring bertambahnya nilai jarak yang semakin jauh.



Gambar 11 Signal Strength vs Distance

2.2 Tahap Penentuan Threshold

Penentuan nilai *Threshold* ditentukan berdasarkan Gambar 1. Pada Gambar 1 terlihat nilai *threshold_high* dan nilai *threshold_low* ditunjukkan oleh garis horizontal yang membentang pada Gambar 1. *threshold_high* digambarkan dengan garis horizontal lurus tanpa putus-putus, sedangkan untuk *threshold_low* digambarkan dengan garis horizontal lurus dengan putus-putus. Nilai *threshold_high* adalah diantara nilai *ss* $1e-08$ dengan $1e-$

09, sedangkan *threshold_low* berada diantara ss $1e-09$ dengan $1e-10$.

Nilai epsilon yang didapatkan berdasarkan dari Gambar 1 berikutnya akan dirubah kedalam nilai logarithmic.

Berdasarkan nilai *threshold* yang digunakan pada paper yang dibuat oleh (Ali, dkk., 2009) yaitu untuk nilai *threshold_high* adalah -8.9 dan untuk nilai *threshold_low* adalah -9.3. Untuk nilai *threshold_high* didapatkan dari persamaan dibawah ini

$$\begin{aligned} \text{threshold_high} &= \log(1e - 8.9) = \log(1 \times 10^{-8.9}) \\ &= -8.9 \end{aligned}$$

Sedangkan untuk *threshold_low* juga menggunakan persamaan yang sama untuk memperoleh nilai logarithmic dari nilai *ss* yang bernilai $1e-9.3$, untuk perhitungan *threshold_low* dapat dilihat seperti dibawah ini (1)

$$\begin{aligned} \text{threshold_low} &= \log(1e - 9.3) = \log(1 \times 10^{-9.3}) \\ &= -9.3 \end{aligned}$$

Nilai *threshold* inilah yang selanjutnya akan digunakan dalam perhitungan *link-quality* pada sub-bab berikut.

2.3 Tahap Perhitungan Link Quality

Tahapan perhitungan *link-quality* dilakukan berdasarkan informasi *signal strength* yang diterima dari node-node sekitarnya. Ada beberapa indikator yang digunakan pada perhitungan *link-quality* ini antara lain adalah adanya *threshold_high*, *threshold_low*, *Hyst_ss_scaling* dan juga delta. Tahapan perhitungan *link-quality* adalah sebagai berikut :

1. Ketika paket hello pertama diterima dengan *signal strength* di bawah *ss_threshold_high*, maka paket tersebut tidak dianggap sebagai *reception* jadi *link-quality* tidak diinisialisasi. Hanya paket dengan *signal strength* di atas *ss_threshold_low* yang akan diinisialisasi *link-quality*-nya
2. Paket Hello kedua berada di atas *ss_threshold_low* dengan demikian *link-quality* diinisialisasi, tetapi *link_status* masih tertunda oleh karena itu, *link* tidak dapat digunakan untuk mengirimkan pesan kepada siapapun. Ketika node menerima paket lain pada sekitar tingkat daya yang sama, tidak berdampak pada *link-quality*.
3. Jadi Setiap kali perbedaan dalam *signal strength*, yang terakumulasi, melintasi nilai tertentu dari Δ , *link-quality rewarded* atau *punished* berdasarkan peningkatan atau penurunan masing-masing.
4. Penggunaan Δ memungkinkan untuk *rewarded* hanya node yang mendekati dan node yang tidak akan tinggal pada jarak yang sama antar hello berturut-turut.

Pemberiaan *rewarded* dan *Punished* antar ambang untuk setiap perubahan Δ , dilakukan untuk membuat link lebih kuat dan mengurangi waktu yang dibutuhkan dalam mengubah status link. Ketika sebuah paket dengan *signal strength* di atas *ss_threshold_high* diterima, *link-quality rewarded* dan selama *link-quality* melintasi ambang batas, link menjadi aktif (*link_pending=false*).

3 DISAIN SISTEM

Pada penelitian ini protokol *Link-Stability Zone Routing Protocol* (LS-ZRP) dirancang dengan mengintegrasikan *Signal Strength Based Link-Stability Sensing*, untuk melakukan pemilihan node tetangga yang *reliable* berdasarkan *signal strength* dari node tetangga. Metode ini diterapkan pada sub-protokol pada *Zone Routing Protocol* (ZRP). Desain ini diharapkan mampu menjaga *link-stability* antar *node* untuk mendapatkan rute yang stabil untuk dilalui pada ZRP. Untuk menjaga *link-stability* antar *node* diterapkanlah metode *Signal Strength Based Link-Stability Sensing* pada *Neighbor Discovery Protocol* (NDP) dengan memodifikasi *HELLO Beacons* yang dikirimkan ke semua *node* tetangga. *HELLO Beacons* yang dimodifikasi seperti pada Gambar 3.2 akan memberikan informasi tetangga mengenai *ip*, jarak, *signal strength* dll. Hal ini dijadikan acuan untuk memilih apakah *node* tetangga yang dikirim *HELLO Beacons* merupakan *node* yang *reliable* atau tidak, jika *node* tersebut *reliable* maka akan disimpan pada tabel *node* yang *reliable*. Tabel *node reliable* yang dihasilkan oleh NDP selanjutnya akan menjadi acuan *IntraZone Routing Protocol* (IARP) dalam proses penentuan rute *routing* menuju ke *node* tujuan

Pengembangan Protokol Routing LS-ZRP dilakukan dengan cara menurunkan sifat ZRP asli dan dimodifikasi dengan metode *Signal Strength Based Link-Stability Sensing*. Pada paket header ZRP ditambahkan *signal strength status* untuk menampung nilai dari *signal strength* sebuah *node*. *Signal strength* ini akan ditambahkan seperti terlihat pada Gambar 2.

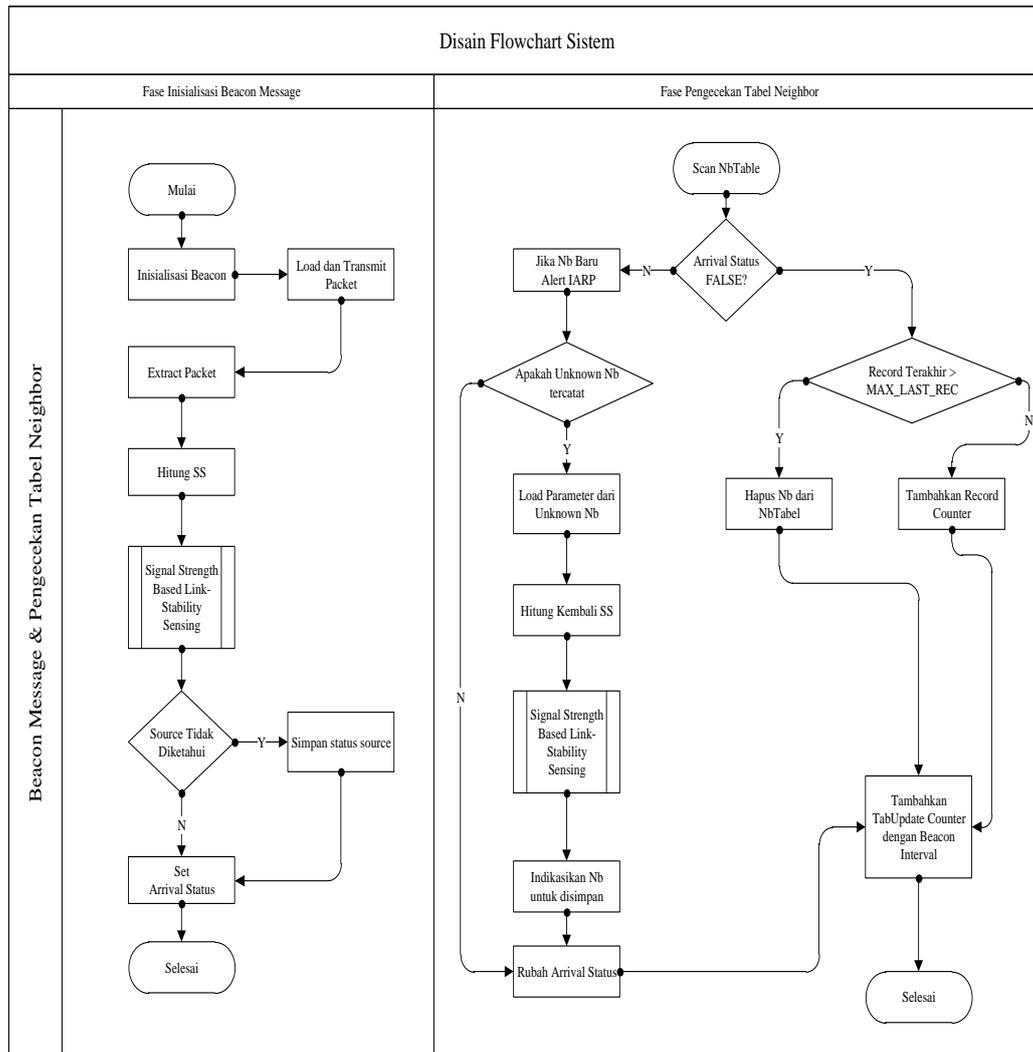
Nilai yang didapat dari *signal strength status* pada paket header digunakan untuk melakukan pencarian *node* tetangga yang *reliable* pada penerapan *Signal Strength Based Link-Stability Sensing*.

3.1 Pengembangan Protocol NDP dengan Link-Stability Sensing

Pada tahapan ini dipaparkan *flowchart* dari hasil pengembangan NDP dengan metode *Signal Strength Based Link-Stability Sensing*.

Flowchart sistem yang terlihat pada Gambar 3, merupakan modifikasi dari *flowchart* NDP. Modifikasi dilakukan dengan menambahkan perhitungan *signal strength* dan *link-stability* pada dua tahapan NDP yaitu pada tahap Inisialisasi *beacons message* dan pada proses pengecekan tabel *neighbor*.

Perhitungan *signal strength* (SS) dilakukan ketika paket yang diterima di ekstrak kemudian dari data paket yang diterima dilakukan perhitungan SS dari *node* yang mengirimkan paket. Selanjutnya adalah melakukan perhitungan dengan menggunakan metode *Signal Strength Based Link-stability Sensing* (Ali, dkk., 2009).



Gambar 13 Flowchart NDP dengan Link Stability Sensing

Untuk mengimplementasikan LS-ZRP pada simulator JiST/SWANS beberapa kode program perlu ditambahkan. Penambahan kode program dilakukan pada Class yang berkaitan dengan ZRP seperti RouteZrpNdp.java, RouteZrp.java, NetMessage.java. serta penambahan pada driver simulator yang berfungsi sebagai scenario simulasi.

4 PERANCANGAN SIMULASI SISTEM

Sesuai dengan tujuan penelitian ini, yaitu untuk menghasilkan sebuah perbaikan mekanisme routing pada protokol ZRP dengan metode *signal strength based link stability sensing*, dinamai dengan LS-ZRP. Protokol tersebut diharapkan menghasilkan performa yang lebih tinggi dalam proses routing dibandingkan dengan protokol ZRP standar.

Perlu ditentukan parameter-parameter yang diamati untuk kedua protokol dalam melakukan perbandingan tersebut. Parameter-parameter ini diamati dan dibandingkan nilainya

antara routing protokol ZRP dan routing protokol LS-ZRP. Terdapat 3 (tiga) parameter yang diamati dalam penelitian ini *end-to-end delay (e2e delay)*, *throughput*, dan *routing overhead*.

- a. Parameter pertama adalah *end-to-end delay* atau *e2e delay*, merupakan waktu yang diperlukan mulai dari sebuah paket dikirimkan hingga paket tersebut diterima dan memberikan *acknowledgement* kepada pengirim paket. Semakin kecil nilai *e2e delay*, maka semakin bagus performa dari protokol tersebut. *End-to-end delay* dihitung dari file trace simulator JiST/SWANS dengan cara mencari selisih antara waktu paket tersebut dikirimkan dengan waktu paket tersebut diterima oleh node penerima. Perhitungan *routing overhead* dapat dilakukan dengan Persamaan 2.

$$e2edelay = \frac{\sum Waktu_Tempuh_Paket}{\sum Rerieved_Packet} \quad (2)$$

- b. Parameter kedua adalah *throughput*, merupakan total jumlah *packet* data yang diterima per detik oleh penerima. *Throughput* ini diperoleh dari dengan menghitung banyaknya paket yang diterima oleh penerima dibagi dengan waktu simulasi. Satuan yang digunakan adalah *kilo-bit-per-second* (kbps). Semakin besar nilai *throughput* semakin baik. Perhitungan *throughput* dapat dilakukan dengan Persamaan 3.

$$throughput = \frac{\sum Packet_size}{(stop_time - start_time)} \quad (3)$$

- c. Parameter ketiga adalah *Routing Overhead*, merupakan total jumlah paket yang di *routing*-kan berbanding dengan jumlah paket yang diterima oleh penerima. *Routing overhead* ini dapat diperoleh dengan cara menghitung banyaknya paket yang di-*routing* dibagi dengan banyaknya paket yang diterima oleh penerima. Hasil pembagian tersebut kemudian dikali 100% untuk mendapatkan hasil dari *routing overhead* dalam bentuk persen (%). Perhitungan *routing overhead* dapat dilakukan dengan Persamaan 4.

$$Routing\ Overhead = \frac{\sum Routing_packet}{\sum Packet_Received} (\%) \quad (4)$$

Tabel 2 Keterangan simbol hasil packet trace

Trace Tipe	Event		Tipe Data	Keterangan	
Packet		-t	time	Time	
		-s	int	Source ID	
		-d	int	Destination ID	
		-e	int	Extent	
		-a	int	Packet Color Attribute ID	
		-i	int	ID	
		-l	int	Energy	
		+	-c	string	Conversation
		Enqueue	-	comment	Comment
		-:	x		
		Dequeue	-	string	Packet Type
			-	string	Packet Type
			-	comment	
			-	int	
			-	int	
			-	int	

```

r -t 10.390309 -s 24 -d 48 -p zrp -e -2147483648 -i 650 -a 0
r -t 10.390309 -s 24 -d 26 -p zrp -e -2147483648 -i 650 -a 0
r -t 10.390309 -s 24 -d 21 -p zrp -e -2147483648 -i 650 -a 0
+ -t 10.395349 -s 41 -d -1 -p zrp -e -2147483648 -i 651 -a 0
- -t 10.395349 -s 41 -d -1 -p zrp -e -2147483648 -i 651 -a 0
h -t 10.395349 -s 41 -d -1 -p zrp -e -2147483648 -i 651 -a 0
r -t 10.395352 -s 41 -d 21 -p zrp -e -2147483648 -i 651 -a 0
r -t 10.395352 -s 41 -d 34 -p zrp -e -2147483648 -i 651 -a 0
r -t 10.395352 -s 41 -d 49 -p zrp -e -2147483648 -i 651 -a 0
d -t 10.395352 -s 41 -d 45 -p zrp -e -2147483648 -i 651 -a 0
r -t 10.395352 -s 41 -d 47 -p zrp -e -2147483648 -i 651 -a 0
r -t 10.395352 -s 41 -d 37 -p zrp -e -2147483648 -i 651 -a 0
r -t 10.395352 -s 41 -d 46 -p zrp -e -2147483648 -i 651 -a 0
r -t 10.395352 -s 41 -d 8 -p zrp -e -2147483648 -i 651 -a 0
    
```

Gambar 14 Hasil Trace File

Hasil *trace* yang dihasilkan, kemudian akan diolah kembali untuk mendapatkan hasil sesuai dengan parameter-parameter yang diamati. Pengolahan dilakukan dengan menggunakan fungsi statistik yang dimiliki oleh JiST/SWANS dan kemudian dihitung dengan rumus dari masing-masing parameter yang diamati.

4.1 Pembangunan Skrip untuk Pengujian

Pada saat pembangunan skrip untuk pengujian, parameter-parameter simulasi ditentukan terlebih dahulu. Parameter-parameter simulasi tersebut adalah atribut-atribut tetap untuk setiap skenario pengujian, sehingga setiap skrip pengujian yang digunakan untuk simulator JiST/SWANS, dibangun berdasarkan parameter-

parameter tersebut. Nilai-nilai untuk parameter-parameter simulasi tersebut dijelaskan secara lebih terperinci dalam Tabel 2.

Tabel 3 Parameter-parameter Simulasi untuk Pengujian

No	Parameter	Spesifikasi
1.	Simulator	JiST/SWANS
2.	Banyaknya Node	50
3.	Area Simulasi	21000m x 1000m, 1500m x 1500m, 2000m x 2000m, 2500m x 2500m, 3000m x 3000m
4.	MAC protokol	IEEE 802.11b DCF
5.	Mode Propagasi	Two-ray ground reflection model
6.	Mobility Model	Random Way Point
7.	Antennas Model	Omnidirectional
8.	Tipe Trafik	Constant Bit Rate (CBR)
9.	Kecepatan Node	5m/s – 50m/s
10.	Simulation Time	900 s (waktu simulasi)
11.	Bordercast Transmission	5, 10, 15, 25, 35, 45
12.	Start Bordercast	100s (Setelah simulasi berjalan sampai akhir simulasi)
13.	Radio Radius	250 m
14.	Source/Destination	Random dalam

		Node
15.	ZRP Zone	2 hope node
16.	Packet Drop	Uniform 0.02
17.	Threshold_high	-8.9 (logarithmic value)
18.	Delta	0.2
19.	Hyst_ss_scaling	0.5

Atribut-atribut pada Tabel 2 digunakan untuk membangun skrip pengujian pada skenario pengujian. Adapun skrip skenario pengujian sistem dijelaskan pada sub-bab berikut.

4.2 Skrip Pengujian

Skrip pengujian ini merupakan hasil dari penentuan nilai yang dilakukan. Penentuan nilai pengujian kemudian diimplementasikan ke simulator dalam bentuk skrip pengujian. Pada Gambar 5 diperlihatkan skrip yang berisikan nilai yang digunakan untuk pengujian. Pada skenario pengujian sistem yang dilakukan, adalah memvariasikan luas wilayah yaitu 1000x1000 s/d 3000x3000. Nilai tersebutlah yang dimasukkan kedalam skrip pada Gambar 2, nilai yang dirubah adalah nilai yang ditandai dengan warna hijau yaitu luas wilayah dari simulasi yang dilakukan. Sedangkan nilai yang lain di setting sama dengan luas wilayah yang lain.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<launchConfiguration type="org.eclipse.jdt.launching.localJavaApplication">
<listAttribute key="org.eclipse.debug.core.MAPPED_RESOURCE_PATHS">
<listEntry value="/LSZRP/src/jist/runtime/Main.java"/>
</listAttribute>
<listAttribute key="org.eclipse.debug.core.MAPPED_RESOURCE_TYPES">
<listEntry value="1"/>
</listAttribute>
<stringAttribute key="org.eclipse.jdt.launching.MAIN_TYPE"
value="jist.runtime.Main"/>
<stringAttribute key="org.eclipse.jdt.launching.PROGRAM_ARGUMENTS"
value="jist.swans.Main driver.lsZrp -p zrp:2 --iarp=iarp:inf -e 900 -n 50 -f
1000x1000 -l uniform:0.02 -w -a random -b 10,100,20 -m waypoint:0,2,5,40"/>
<stringAttribute key="org.eclipse.jdt.launching.PROJECT_ATTR" value="LSZRP"/>
</launchConfiguration>
```

Gambar 15 Contoh skrip pengujian

Skenario sistem berdasarkan skrip pada Gambar 2 akan dijalankan pada simulator JiST/SWANS. Skrip pengujian sistem tersebut akan menghasilkan file trace yang kemudian diaolah kembali untuk mendapatkan hasil sesuai dengan parameter yang diamati yaitu *e2e delay*, *throughput*, dan *routing overhead*.

5 HASIL DAN PEMBAHASAN

Pada bagian ini akan dilakukan evaluasi kinerja dari metode pemilihan node tetangga yang diajukan dengan melihat hasil parameter yang diamati dan perbandingan dengan protokol ZRP standar. Metode yang diajukan diimplementasikan

dengan bahasa pemrograman Java dan disimulasikan menggunakan simulator JiST/SWANS, pada sebuah komputer dengan spesifikasi Processor Intel Core i3-2120 3,3 Ghz, 4 Core, RAM 4 GB, (2,92 GB useable), Hardisk 500 GB, Sistem Operasi Windows 7 32-bit.

5.1 Parameter Uji Coba

Adapun parameter pengujian simulator yang digunakan untuk menguji sistem yang dibuat adalah seperti terlihat pada tabel 3.1

Tabel 4 Parameter Pengujian pada Simulator

No	Parameter	Spesifikasi
1.	Simulator	JiST/SWANS
2.	Banyaknya Node	50
3.	Area Simulasi	21000m x 1000m, 1500m x 1500m, 2000m x 2000m, 2500m x 2500m, 3000m x 3000m
4.	MAC protokol	IEEE 802.11b DCF
5.	Mode Propagasi	Two-ray ground reflection model
6.	Mobility model	Random Way Point
7.	Antennas Model	Omnidirectional
8.	Traffic type	Constant Bit Rate (CBR)
9.	Speed of node	5m/s – 50m/s
10.	Simulation Time	900 s
11.	Bordercast Transmision	5, 10, 15, 25, 35, 45
12.	Start Bordercast	100s (Setelah simulasi berjalan sampai akhir simulasi)
13.	Radio Radius	250 m
14.	Source/Destination	Random dalam Node
15.	Waktu Simulasi	900 detik (waktu simulator)

5.2 Skenario Uji Coba

Skenario pengujian pada penelitian ini dibagi menjadi 3 sesuai dengan pembahasan pada bab 3. Skenario pengujian ini dilakukan untuk membuktikan kebenaran dan mengetahui kinerja protokol routing ZRP yang telah dimodifikasi dengan menambahkan metode *signal strength based link-stability sensing* yang kemudian menjadi LS-ZRP (*Link-Stability Zone Routing Protocol*).

Pada skenario ini wilayah simulasi dirancang sedemikian rupa dengan bentuk persegi dan luas wilayahnya divariasikan yaitu 1000x1000, 1500x1500, 2000x2000, 2500x2500 dan 3000x3000, luas wilayah ini dalam satuan meter. Jumlah node yang disimulasikan adalah 50 node dengan jumlah transmisi bordercast sebanyak 10, start bordercast 100s setelah simulasi berjalan

(waktu simulasi) delay bordercast 10, waktu simulasi mengacu pada tabel 3.1 parameter uji yaitu serta untuk kecepatan node divariasikan mulai 5m/s – 40m/s.

Pada JiST/SWANS skenario diatas diimplementasikan pada antarmuka driver. Protokol lapisan MAC yang digunakan adalah IEEE 802.11b *Distributed Coordination Function* (DCF). DCF didesain dengan menggunakan mekanisme CSMA/CA dan algoritma *binary exponential backoff* untuk mengurangi kemungkinan terjadinya *collision*. Selain itu, DCF juga menerapkan skema *positive acknowledge*, yang mana jika sebuah frame telah diterima dengan benar oleh node tujuan, node tujuan harus mengirimkan frame ACK kepada node sumber.

Model propagasi radio yang digunakan adalah model *Two Ray Ground*. Model ini mempertimbangkan dua aspek, yaitu lintasan lurus antara *transmitter* node dengan *receiver* node dan pantulan permukaan bumi. Lintasan antara *transmitter* node dengan *receiver* node diasumsikan memenuhi kondisi *line of sight*, yang artinya tidak ada penghalang di antara node tujuan dengan node sumber. Sehingga sesuai untuk simulasi yang memerlukan jangkauan transmitter yang jauh. Pada penelitian ini jangkauan transmisi adalah 250 meter, tinggi 43statist adalah 1.5 meter yang ditempatkan sedemikian rupa tepat di tengah mobile node.

Untuk melakukan uji stressing terhadap kemampuan routing protocol ketika melakukan prosedur *route discovery* dan *route maintenance* tanpa menimbulkan kongesti yang berlebihan, *Constant Bit Rate* (CBR) dipilih sebagai model komunikasi antar node yang *uniform*. CBR merupakan aplikasi yang berjalan di atas *transport UDP*. Node sumber akan mengirim trafik CBR dalam paket-paket data sebesar 512 bytes dengan kecepatan 4 paket per detik. Selama simulasi berlangsung.

Model mobilitas menggambarkan bagaimana node bergerak pada area simulasi. Model mobilitas yang digunakan pada penelitian ini adalah *random waypoint*. Model mobilitas *random waypoint* dipilih karena mayoritas simulasi menggunakan model ini dan merupakan model pergerakan standar pada JiST/SWANS. Pada model *random waypoint*, posisi awal setiap mobile node dipilih secara acak pada area simulasi. Kecepatan *mobile* node dipilih secara acak di antara kecepatan minimal dan kecepatan maksimal. Kecepatan minimal diberi nilai 1 m/s agar kecepatan mobile node dapat mencapai kondisi yang stabil dan konvergen dengan cepat.

Setelah pembuatan program skenario selesai, maka langkah selanjutnya adalah menjalankan simulasi program tersebut. Simulasi program menghasilkan trace file dengan ekstensi .tr. Trace file ini berisi informasi-informasi yang diperlukan untuk analisa kinerja routing protocol.

Dalam simulator JiST/SWANS, pembuatan trace file untuk mendapatkan hasil berupa animasi jaringan atau disebut dengan NAM, membutuhkan library tambahan yaitu library guiTrace. guiTrace akan menghasilkan file berekstensi .nam. NAM merupakan singkatan dari Network Animator, sehingga simulasi driver dengan tambahan library guiTrace dapat dianimasikan dengan program aplikasi NAM.

5.3 Hasil Uji Coba

Hasil uji coba dari skenario yang telah dilakukan akan ditampilkan dalam bentuk tabel yang berisikan data dari parameter-parameter yang diamati, yaitu *routing overhead*, *throughput*, dan *e2e delay*.

Tabel 5 Hasil Uji Coba

Luas Area	Routing Overhead		Throughput		e2e delay	
	LS-ZRP	ZRP	LS-ZRP (kbps)	ZRP (kbps)	LS-ZRP (ms)	ZRP (ms)
1000x1000	63.99%	50.82%	35.56	44.77	204.76	162.63
1500x1500	23.69%	24.19%	96.06	94.06	75.80	77.41
2000x2000	29.30%	34.83%	77.66	65.34	93.76	111.45
2500x2500	45.62%	55.19%	49.86	41.10	146.04	164.00
3000x3000	66.64%	83.33%	33.51	26.76	217.33	272.11

5.4 Hasil Uji Coba Parameter Routing Overhead

Routing Overhead adalah total jumlah paket yang di *routing*-kan berbanding dengan jumlah paket yang diterima oleh penerima. Untuk mendapatkan nilai presentase maka hasil *Routing Overhead* dikalikan 100%.

Untuk mendapatkan nilai dari *routing overhead* setiap skenario, digunakanlah rumus sebagai berikut:

$$\begin{aligned} \text{Routing Overhead} &= \frac{\sum \text{Routing_Packet}}{\sum \text{Packet_Recieved}} (\%) \\ \text{Routing Overhead} &= \frac{50896}{89899} (\%) \\ \text{Routing Overhead} &= 0.5661 \times 100 \% \\ \text{Routing Overhead} &= 56.61\% \end{aligned}$$

Hasil yang didapatkan dari rumus diatas selanjutnya akan disimpan ke dalam bentuk tabel yang kemudian diolah menjadi grafik yang bisa dianalisa.

Data *Routing Overhead* pada Tabel 5 adalah data hasil pengujian skenario yang telah dilakukan dengan variasi luas area yang disimulasikan. Terdapat 5 variasi luas area yang terdapat pada 5. Dari data hasil pengujian skenario yang telah dilakukan pada Tabel 5 terlihat *Routing Overhead* LS-ZRP lebih tinggi dibandingkan dengan ZRP pada luas wilayah 1000x1000 meter. Nilai *Routing Overhead* LS-ZRP sebesar 63.99% sedangkan nilai *Routing Overhead* ZRP sebesar 50.82% pada luas area 1000x1000 meter. Untuk luas area 1500x1500 meter *Routing Overhead* dari LS-ZRP mengalami penurunan yang sangat signifikan yaitu dari 63.99% menjadi 23.69%. Penurunan

Routing Overhead juga terlihat pada ZRP yaitu dari 50.82% menjadi 24.19%.

Routing Overhead mengalami peningkatan setelah luas area menjadi 2000x2000 meter. LS-ZRP dan ZRP sama-sama mengalami peningkatan *Routing Overhead*, pada luas area 2000x2000 meter, namun peningkatan nilai *Routing Overhead* dari LS-ZRP lebih rendah dibandingkan dengan nilai *Routing Overhead* ZRP. Peningkatan nilai *Routing Overhead* LS-ZRP dari 23.96% menjadi 29.30% sedangkan nilai *Routing Overhead* ZRP dari 24.19% menjadi 34.83%. Trend nilai *Routing Overhead* dari protokol LS-ZRP dan ZRP mengalami peningkatan seiring dengan meningkatnya luas area. Secara keseluruhan nilai *Routing Overhead* LS-ZRP masih lebih baik dengan rata-rata nilai *Routing Overhead* sebesar 45.85% sedangkan rata-rata nilai *Routing Overhead* dari ZRP adalah 49.67%. Selisih rata-rata nilai *Routing Overhead* dari LS-ZRP dan ZRP adalah 3.83%.

Grafik dari Gambar 6 memiliki trend yang sama dengan grafik *e2e delay* pada Gambar 8. grafik *e2e delay* trend-nya adalah mengalami penurunan pada saat luas wilayah 1500x1500 meter dan grafik *Routing Overhead* pada Gambar 6 juga mengalami penurunan pada area yang sama. Grafik *Routing Overhead* kembali mengalami peningkatan seiring dengan naiknya nilai *e2e delay* yang diikuti oleh bertambahnya luas area. Jadi ketika nilai *e2e delay* mengalami penurunan maka nilai *Routing Overhead* juga akan mengalami penurunan, hal ini disebabkan oleh semakin tingginya *Routing Overhead* maka nilai *e2e delay* juga akan meningkat. Karena pada proses *Routing*, paket akan mengalami pengecekan node, jadi semakin lama pengecekan

yang terjadi maka nilai *e2e delay* juga akan bertambah semakin lama.

5.5 Hasil Uji Coba Parameter *Throughput*

Throughput merupakan jumlah paket yang diterima dalam setiap detik. Hasil perhitungan *Throughput* yang didapatkan dari skenario yang telah dilakukan dimuat dalam satu tabel antara *Throughput* LS-ZRP dan ZRP.

Perhitungan *Throughput* bisa dilakukan dengan menggunakan rumus sebagai berikut:

$$\begin{aligned} \text{throughput} &= \frac{\sum \text{packet_size}}{(\text{stop}_{time} - \text{Start}_{time})} \text{ kbps} \\ \text{throughput} &= \frac{29448}{800} \text{ kbps} \\ \text{throughput} &= 36.81 \text{ kbps} \end{aligned}$$

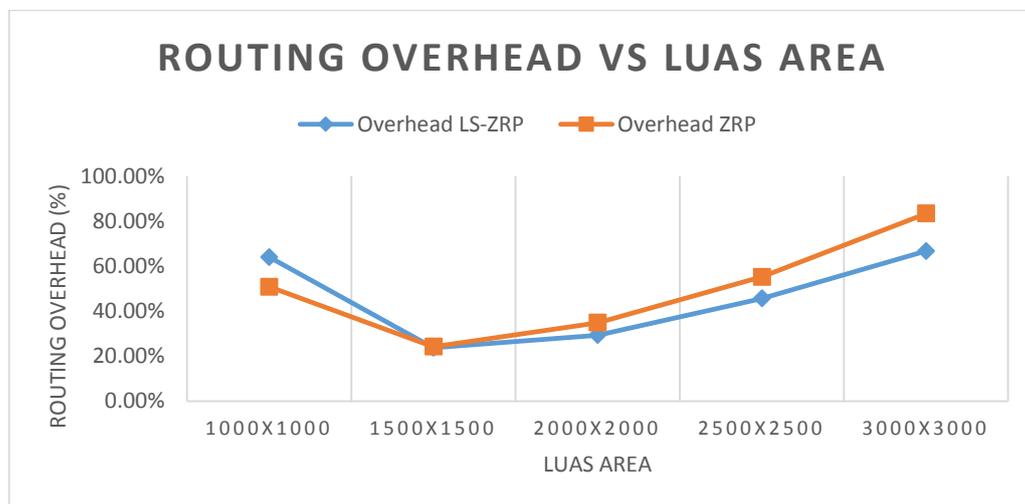
Dari perhitungan dengan menggunakan rumus diatas maka didapatlah data yang selanjutnya dijadikan sebuah tabel yang terlihat seperti Tabel 5. Tabel hasil *throughput* selanjutnya akan dianalisis dan disajikan dalam bentuk grafik.

Data yang diperlihatkan pada Tabel 5 adalah data hasil pengujian yang telah dilakukan dengan variasi luas area yang disimulasikan. Terdapat 5 variasi luas area yang terdapat pada Tabel 5. Dari data hasil pengujian yang telah dilakukan pada Tabel 5 terlihat *Throughput* LS-ZRP lebih kecil dibandingkan dengan ZRP pada luas wilayah 1000x1000 meter. Nilai *Throughput* LS-ZRP sebesar 35.56 kbps sedangkan nilai *Through-*

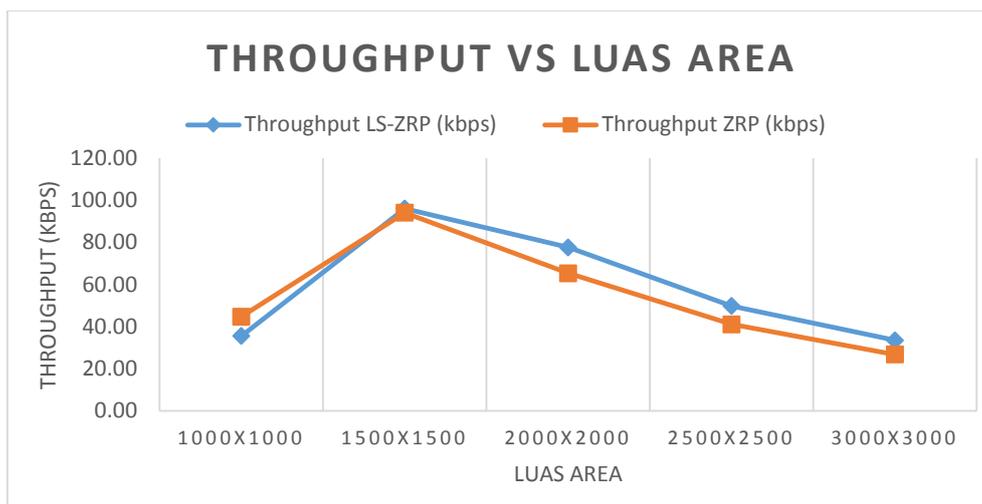
put ZRP sebesar 44.77 kbps pada luas area 1000x1000 meter. Untuk luas area 1500x1500 meter *Throughput* dari LS-ZRP mengalami peningkatan yang sangat signifikan yaitu dari 35.56 kbps menjadi 96.06 kbps. Peningkatan *Throughput* juga terlihat pada ZRP yaitu dari 44.77 kbps menjadi 94.06 kbps.

Nilai *Throughput* mengalami penurunan setelah luas area menjadi 2000x2000 meter. LS-ZRP dan ZRP sama-sama mengalami penurunan *Throughput*, pada luas area 2000x2000 meter, namun penurunan nilai *Throughput* dari LS-ZRP lebih rendah dibandingkan dengan nilai *Throughput* ZRP. Penurunan nilai *Throughput* LS-ZRP dari 96.06 kbps menjadi 77.66 kbps sedangkan nilai *Throughput* ZRP dari 94.06 kbps menjadi 65.43 kbps. Trend nilai *Throughput* dari protokol LS-ZRP dan ZRP mengalami penurunan seiring dengan meningkatnya luas area. Secara keseluruhan nilai *Throughput* LS-ZRP masih lebih baik dengan rata-rata nilai *Throughput* sebesar 58,53 kbps sedangkan rata-rata nilai *Throughput* dari ZRP adalah 54,41 kbps. Selisih rata-rata nilai *Throughput* dari LS-ZRP dan ZRP adalah 4.13 kbps.

Grafik dari Gambar 7 memiliki trend yang sama dengan grafik *e2e delay* pada Gambar 8 tetapi dengan nilai yang terbalik. Pada grafik *e2e delay* trend-nya adalah mengalami penurunan pada saat luas wilayah 1500x1500 meter sedangkan untuk grafik *Throughput* pada Gambar 4.17 mengalami peningkatan pada area yang sama.



Gambar 16 Grafik *Routing Overhead* vs Luas Area

Gambar 17 Grafik *Throughput* vs Luas Area

Grafik *Throughput* kembali mengalami penurunan seiring dengan naiknya nilai *e2e delay* yang diikuti oleh bertambahnya luas area. Jadi ketika nilai *e2e delay* mengalami penurunan maka nilai *Throughput* akan mengalami peningkatan, hal ini disebabkan oleh semakin tingginya *e2e delay* maka jumlah data yang diterima akan semakin sedikit, begitu juga sebaliknya semakin sedikit ren-

5.6 Hasil Uji Coba Parameter *e2e delay*

End-to-end delay (e2e delay) merupakan perbandingan antara jumlah waktu transmisi packet dari *source* ke *destination* dengan jumlah *packet* yang diterima. Perhitungan *e2e delay* bisa dilakukan dengan menggunakan rumus sebagai berikut:

$$e2edelay = \frac{\sum Waktu_Tempuh_Paket}{\sum Recieved_Packet}$$

$$e2edelay = \frac{325399 \text{ ms}}{5880}$$

$$e2edelay = 55.3399 \text{ ms}$$

Dari hasil perhitungan di atas maka nilai dari *e2e delay* 55.3399 ms, yang didapatkan dari keseluruhan waktu tempuh packet, yakni 325.399 ms dibagi dengan banyaknya packet yang diterima yakni 5880 packet.

Hasil *e2e delay* dari keseluruhan skenario disajikan ke dalam bentuk yaitu Tabel 5. Tabel tersebut dibagi menjadi 3 bagian skenario dan masing-masing skenario dibagi menjadi 2 bagian yaitu LS-ZRP dan ZRP.

Pada Tabel 5 terlihat *e2e delay* LS-ZRP lebih besar dibandingkan dengan ZRP pada luas wilayah 1000x1000 meter. Nilai *e2e delay* LS-ZRP sebesar 204.76ms sedangkan nilai *e2e delay* ZRP sebesar 162.63ms pada luas area 1000x1000 meter. Untuk luas area 1500x1500 meter *e2e delay* dari LS-ZRP mengalami penurunan yang sangat signifikan yaitu dari 204.76 ms menjadi 75.80 ms. Penurunan *e2e delay* juga terlihat pada ZRP yaitu dari 162.63 ms menjadi 77.41 ms.

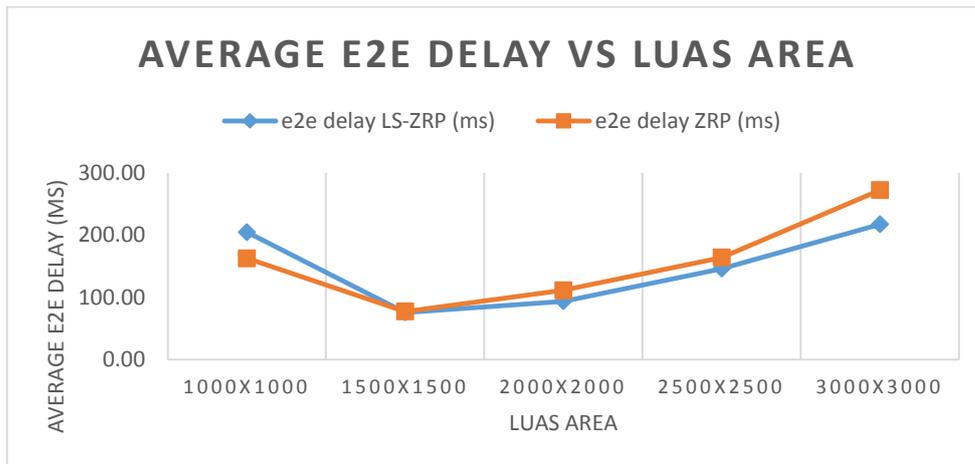
Trend peningkatan nilai *e2e delay* terjadi setelah luas area menjadi 2000x2000 meter. LS-

dah nilai *e2e delay* maka jumlah data yang bisa dilewatkan akan semakin besar juga.

keseluruhan *Throughput* LS-ZRP masih lebih baik dari ZRP terbukti dari nilai rata-rata *throughput* LS-ZRP yaitu sebesar 80.31 kbps, serta untuk nilai rata-rata *throughput* ZRP adalah sebesar 67.29 kbps, dengan selisih rata-rata antara LS-ZRP dengan ZRP yaitu sebesar 13.02 kbps.

ZRP dan ZRP sama-sama mengalami peningkatan *e2e delay*, pada luas area 2000x2000 meter, namun peningkatan nilai *e2e delay* dari LS-ZRP lebih rendah dibandingkan dengan nilai *e2e delay* ZRP. Peningkatan nilai *e2e delay* LS-ZRP dari 75.80 ms menjadi 93.76 ms sedangkan nilai *e2e delay* ZRP dari 77.41 ms menjadi 111.45 ms. Trend nilai *e2e delay* dari protokol LS-ZRP dan ZRP mengalami peningkatan seiring dengan meningkatnya luas area. Secara keseluruhan nilai *e2e delay* LS-ZRP masih lebih baik dengan rata-rata nilai *e2e delay* sebesar 147.54 ms sedangkan rata-rata nilai *e2e delay* dari ZRP adalah 157,52 ms. Selisih rata-rata nilai *e2e delay* dari LS-ZRP dan ZRP adalah 9.98 ms.

Representasi dari analisis yang dilakukan terhadap data *e2e delay* hasil pengujian yang telah dilakukandapat dilihat pada Gambar 8. Grafik *e2e delay* dari pengujian yang telah dilakukan mengalami penurunan pada saat luas area berubah dari 1000x1000 meter menjadi 1500x1500 meter. Tingginya delay pada luas area 1000x1000 meter disebabkan oleh kepadatan node (density), dimana pada saat itu jumlah node yang disimulasikan berjumlah 50 dengan luas wilayah 1000x1000 meter. Area ini dikategorikan sempit untuk jumlah node 50, seperti yang terlihat pada Gambar 8 nilai *e2e delay* dari LS-ZRP lebih tinggi dibandingkan dengan nilai *e2e delay* dari ZRP pada area 1000x1000 meter..



Gambar 18 Grafik Average e2e delay vs Luas Area

LS-ZRP memiliki nilai *e2e delay* lebih tinggi dikarenakan oleh adanya perhitungan *signal strength* dan *link-stability* yang diadaptasi pada protokol tersebut, jadi dengan adanya penambahan perhitungan kalkulasi *signal strength* dan *link-stability* pada area yang memiliki kepadatan node, menyebabkan LS-ZRP mendapatkan nilai *e2e delay* lebih tinggi dibandingkan dengan ZRP yang tidak menggunakan perhitungan tersebut

Dapat dilihat juga pada grafik Gambar 8 nilai *e2e delay* antara LS-ZRP dengan ZRP sama-sama mengalami penurunan pada luas wilayah 1500x1500 meter, penurunan nilai *e2e delay* dari kedua protokol sangat signifikan, hal ini disebabkan karena pada luas wilayah 1500x1500 meter dengan jumlah node 50 dan kecepatan node bervariasi dari 5m/s – 40m/s serta *Bordercast Transmission* 10 dengan zone hop adalah 2 node, merupakan setingan yang paling optimun. Jadi perbandingan antara jumlah node, kecepatan node, *bordercast transmission* dan zona hop, merupakan setingan yang paling tepat untuk digunakan pada saat area berukuran 1500x1500 meter. Grafik pada Gambar 8 memperlihatkan trend meningkat setelah luas wilayah berubah menjadi lebih besar. Kedua protokol menunjukkan trend yang sama akan tetapi untuk LS-ZRP pada trend naik ini masih lebih bagus dari pada ZRP.

6 KESIMPULAN

Berdasarkan hasil uji coba yang telah dilakukan, metode yang diterapkan pada protokol ZRP, mampu menghasilkan nilai lebih baik dibandingkan dengan protokol ZRP standar. Hal ini dibuktikan dengan peningkatan nilai *throughput* sebesar 4.13 kbps, penurunan *e2e delay* sebesar 9.98 ms, dan penurunan *routing overhead* sebesar 3.83%.

7 DAFTAR PUSTAKA

- Ali, H. M. i., Naimi, A. M., Busso, A. & Veque, V., (2009). Signal Strength Based Link Sensing for Mobile Ad-Hoc Networks. *Telecommunication Systems*, Desember, 42(3-4), hal. 201-212.
- Asha, A., Sharma, H. & Mohabey, V. K., (2012). Improved AODV Protocol For Solving Link Failure in MANET. *International Journal of Scientific & Engineering Research*, Oktober, 3(10), hal. 1-6.
- Bejar, N., (2002). *Zone Routing Protocol (ZRP)*. [Online] Tersedia di: <http://www.netlab.hut.fi/opetus/s38030/k02/Papers/08-Nicklas.pdf> [Accessed 4 Januari 2013].
- C. Biradar, R. & S. Manvi, S., (2012). Neighbor supported reliable multipath multicast routing in MANETs. *Journal of Network and Computer Applications*, Volume 35, hal. 1074-1085.
- J. Haas, Z., Pearlman, M. R. & Samar, P., (2003). *The Zone Routing Protocol (ZRP) for Ad Hoc Networks*. [Online] Tersedia di: <http://tools.ietf.org/html/draft-ietf-manet-zone-zrp-04> [Accessed 3 Januari 2013].
- Jain, S., Kokate, S., Thakur, P. & Takalkar, S., (2012). A Study of Congestion Aware Adaptive Routing Protocols in MANET. *Computer Engineering and Intelligent Systems*, 3(4), hal. 64-73.
- Ravilla, D. & Putta C., S. R., (2012). Energy Management in Zone Routing Protocol

(ZRP). *International Journal of Emerging Technology and Advanced Engineering*, Mei, 2(5), hal. 61-68.

Shafiq, Z. et al., (2012). *Zone Routing Protocol: How does it Perform the How does it Perform the*. Peshawar, s.n., hal. 71-77.

Wrona, K. & Mahonen, P., (2004). Analytical model of cooperation in ad hoc Networks. *Telecommunication Systems*, Volume 27, hal. 347-369.