

ANALISIS PERFORMANSI MODIFIKASI BINARY SPRAY AND WAIT MENGGUNAKAN PROPHET PADA DTN

Bima Adhiguna*, Tody Ariefianto Wibowo, Leanna Vidya Yovita

Program Studi Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

*Corresponding author, e-mail: bima26.adhiguna@gmail.com¹

Abstrak—*Delay Tolerant Network* (DTN) memungkinkan komunikasi dalam lingkungan di mana mungkin tidak ada jalur *end-to-end*, peluang berkomunikasi datang dan pergi dan jarak antara pengirim dan penerima bisa sangat panjang dan bahkan tidak diketahui sebelumnya. Ada beberapa *Routing Protocol* pada DTN, diantaranya *Binary Spray and Wait* yang merupakan pengembangan dari *Spray and Wait*, dan *PROPHET*. Penelitian ini membahas tentang modifikasi *Binary Spray and Wait* pada tahap *spray* menggunakan teknik *delivery predictability* yang ada di *PROPHET*. Simulasi menggunakan *ONE (Opportunistic Network Environment) Simulator* dan menganalisa *Delivery Probability*, *Overhead Ratio*, *Buffer Time*, *Average Latency* dan *Average Remaining Energy* pada pergerakan node *Shortest Path Map Based* dan *Random Way Point*. Berdasarkan pengamatan, dihasilkan Modifikasi *Binary Spray and Wait* meningkatkan performansi *average latency* dan *buffer time* dibanding dengan *Binary Spray and Wait*, namun menurunkan sedikit *delivery probability* dan memperbesar *overhead ratio*.

Kata Kunci : *Delay Tolerant Network*, *Spray and Wait*, *PROPHET*, *Shortest Path Map Based*, *Random Way Point*.

Abstract—*Delay Tolerant Network* (DTN) enabling communication in an environment where there may be no end-to-end link, opportunities to communicate come and go and the distance between sender and receiver can be very long and even previously unknown. There are some *Routing Protocol* on DTN, including *Binary Spray and Wait* is a development of *Spray and Wait*, and *PROPHET*. Research discuss about modification of *Binary Spray and Wait* in *spray* phase using *delivery predictability* in *PROPHET*. Simulation used *ONE (Opportunistic Network Environment) Simulator* and analyze *Delivery Probability*, *Overhead Ratio*, *Buffer Time*, *Average Latency* and *Average Remaining Energy* in *Shortest Path Map Based* dan *Random Way Point* movement. Based on observation, the resulting modification of *Binary Spray and Wait* improve performance and *buffer time* *average latency* compared with *Binary Spray and Wait*, but slightly lower probability delivery and improving *overhead ratio*.

Keywords : *Delay Tolerant Network*, *Spray and Wait*, *PROPHET*, *Shortest Path Map Based*, *Random Way Point*.

Copyright © 2017 JNTE. All rights reserved

1. PENDAHULUAN

Fungsi utama pada jaringan komunikasi berkaitan dengan routing yang bertujuan untuk menemukan jalur dari sumber ke beberapa tujuan [2]. Ada dua tipe jaringan, yaitu *Traditional Network* dan *Delay Tolerant Network*. Dalam jaringan tradisional kita asumsikan bahwa ada jalur *end-to-end* antara *node* yang berkomunikasi. Akan tetapi, jalur yang dilalui tidak selamanya dapat diandalkan atau tersedia. Hal tersebut bisa menyebabkan paket yang dikirim tidak dapat diteruskan. Pada komunikasi *Delay Tolerant Network* memungkinkan paket tersebut bisa diteruskan sampai ke *node* tujuan.

Delay Tolerant Network (DTN) adalah paradigma jaringan komunikasi yang memungkinkan komunikasi dalam lingkungan di mana mungkin tidak ada jalur *end-to-end*, peluang komunikasi datang dan pergi dan intervalnya bisa sangat panjang dan bahkan tidak diketahui sebelumnya [4]. DTN menggunakan mekanisme *Store and Forward* untuk mengirim pesan dimana jika *node* sumber memiliki paket data yang akan dikirim tetapi jalur *node* berikutnya tidak tersedia, maka paket data tersebut akan disimpan sampai kontak berikutnya tersedia dan kemudian diteruskan. DTN memiliki beberapa protokol perutean, diantaranya adalah *Spray and Wait* dan *PROPHET*.

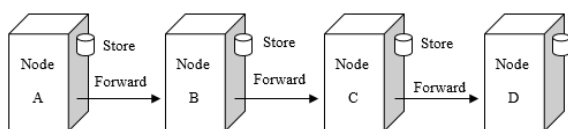
Spray and Wait mempunyai dua tahap, tahap *Spray* dan tahap *Wait* [2]. Pada tahap *Spray*, *node* sumber meneruskan paket ke L *node* yang berbeda. Jika jalur ditemukan maka pesan atau transfer paket berhasil. Jika tidak maka tahap *Wait* dijalankan sampai menemukan *node* tujuan. Namun, pada *Spray and Wait* tidak diketahui bagaimana awalnya salinan pesan L disebar. Oleh sebab itu, muncul *Binary Spray and Wait* yang merupakan pengembangan dari *Spray and Wait* dimana ada inisialisasi jumlah salinan pesan di *node* sumber maupun *node* relay dahulu dan memberikan sebanyak $L/2$ salinan pesan untuk *node* yang ditemui. Pada *PROPHET*, paket diteruskan berdasarkan perhitungan probabilitas oleh setiap *node* ke *node* tujuan. Probabilitas ini disebut *Delivery Predictability*.

Berdasarkan ulasan di atas maka tercipta ide untuk memodifikasi *Binary Spray and Wait* yaitu pada tahap *spray* meneruskan salinan pesan ke *node* yang mempunyai nilai *delivery predictability* lebih tinggi seperti pada *PROPHET*, dengan tujuan agar data yang diteruskan lebih cepat sampai *node* yang dituju.

2. TINJAUAN PUSTAKA

2.1. Delay Tolerant Network (DTN)

Delay Tolerant Network adalah paradigma jaringan komunikasi yang memungkinkan komunikasi dalam lingkungan di mana mungkin tidak ada jalur *end-to-end*, peluang komunikasi datang dan pergi dan intervalnya bisa sangat panjang dan bahkan tidak diketahui sebelumnya [4]. DTN menggunakan mekanisme *Store and Forward* untuk mengirim pesan dimana jika *node* sumber memiliki paket data yang akan dikirim tetapi jalur *node* berikutnya tidak tersedia, maka paket data tersebut akan disimpan sampai kontak berikutnya tersedia dan kemudian diteruskan.



Gambar 1. Proses *Store and Forward*

Dalam DTN, proses *Store and Forward* dilakukan pada sebuah *layer* tambahan yang disebut *Bundle Layer*. *Bundle layer* adalah sebuah *layer* tambahan untuk memodifikasi

paket data dengan fasilitas-fasilitas yang disediakan DTN. *Bundle Layer* memuat data dari sumber, informasi kontrol dan *Bundle Header*.

Bundle Layer terletak langsung di bawah *application layer*. Dalam *bundle layer*, data dari *application layer* akan dipecah-pecah menjadi *bundle*. *Bundle* inilah yang akan dikirim ke *transport layer* untuk diproses lebih lanjut.

Beberapa contoh jaringan berbasis DTN adalah ZebraNet untuk *Wildlife tracking*, Daknet yang memungkinkan penggunaan aplikasi internet di daerah pedesaan tanpa infrastruktur, OpraCom untuk *monitoring* populasi pada area yang luas, dll.

2.2. Routing Protocol Pada DTN

Routing Protocol (Protokol Perutean) merupakan sekumpulan aturan/ standar untuk menentukan bagaimana router dalam sebuah jaringan bisa berkomunikasi dan melakukan pertukaran informasi antara satu dan yang lainnya. Dalam teknologi *Delay Tolerant Network*, *routing protocol* dibagi menjadi 2 yaitu *single copy routing protocol* dan *multicopy routing protocol*. *Single copy* merupakan jenis *routing protocol* dimana hanya membentuk satu *bundle* untuk dikirim di dalam jaringan. Contoh *routing protocol single copy* yaitu *first contact* dan *direct delivery*. Skema berbeda ditunjukkan oleh *multicopy*, dimana protokol ini menggandakan *bundle* di dalam jaringan untuk memperbaiki probabilitas pengiriman dan mengurangi *delay* akibat pengiriman namun menambah *bandwidth* dan penyimpanan. Contoh *routing protocol* jenis *multicopy* yaitu *Spray and Wait*, *Epidemic*, *MaxProp* dan *PROPHET*.

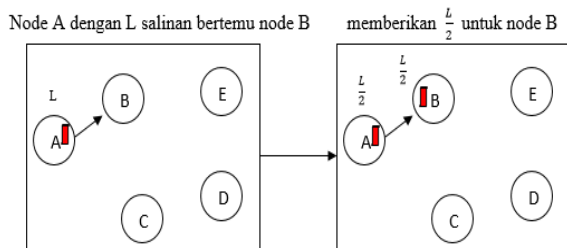
2.3. Spray and Wait

Spray and Wait (SaW) memiliki 2 tahap, yaitu tahap *spray* dan tahap *wait* [4]. *Node* sumber awalnya menginisialisasi jumlah salinan pesan sejumlah L . Pada tahap *spray*, *node* sumber mengirimkan 1 salinan pesan ke L *node* yang berbeda. Jika *node* tujuan ditemukan maka pesan akan sukses dikirim. Apabila tidak, akan masuk tahap *wait* sampai *direct transmission* ke *node* tujuan.

2.4. Binary Spray and Wait

Binary Spray and Wait (BSW) merupakan peningkatan dari *Spray and Wait*. Pada BSW sumber pesan awalnya memiliki L salinan pesan. Ketika menemukan *node* pertama yang belum

memiliki salinan, maka *node* sumber akan memberikan $\frac{L}{2}$ salinan kepada *node* tersebut dan $\frac{L}{2}$ tetap disimpan di *node* sumber. Proses diulang untuk *node* yang mempunyai $L > 1$, dan ketika tersisa hanya 1 salinan kemudian beralih ke tahap wait sampai transmisi langsung ke tujuan.



Gambar 2. Pengiriman Pesan BSW

2.5. PROPHET

Pada *PROPHET* pesan diteruskan berdasarkan perhitungan probabilitas/ peluang oleh setiap *node* ke setiap *node* tujuan [3]. Probabilitas ini disebut *Delivery Predictability*. Ketika dua *node* ditemukan, pesan akan diteruskan ke *node* yang mempunyai nilai *delivery predictability* tertinggi. *Delivery Predictability* ini menunjukkan seberapa besar kemungkinan *node* akan mengirim pesan ke tujuan. Ketika dua *node* bertemu, mereka akan bertukar vektor ringkasan (*summary vector*) yang mana pada kasus ini juga terdiri dari informasi *delivery predictability* yang tersimpan pada *node-node*. Informasi ini digunakan untuk memperbarui vektor internal *delivery predictability*, dan informasi pada vektor ringkasan digunakan untuk menentukan pesan mana yang diminta dari *node* lain. Tahap perhitungan *delivery predictability* [3] :

- Update *delivery matrix* saat *node* bertemu, hal ini diperlukan karena *node-node* yang sering bertemu mempunyai *delivery predictability* tinggi.

$$P_{(a,b)} = P_{(a,b)} + (1 - P_{(a,b) \text{ old}}) \times P_{\text{init}}$$

$P_{\text{init}} \in [0,1]$: konstanta inisialisasi

- Pengiriman *delivery predictability* harus ada batas waktu.

$$P_{(a,b)} = P_{(a,b) \text{ old}} \times \gamma^k$$

$\gamma^k \in [0,1]$: konstanta batas waktu

- *Delivery predictability* juga mengikuti transitivitas, jika *node* A sering bertemu dengan *node* B dan *node* B sering bertemu dengan *node* C maka *node* C kemungkinan

adalah *node* yang bagus untuk meneruskan pesan ke tujuan dari *node* A.

$$P_{(a,c)} = P_{(a,b) \text{ old}} + (1 - P_{(a,c) \text{ old}}) \times P_{(a,b)} \times P_{(b,c)} \times \beta$$

- β : konstanta skala yang memutuskan seberapa besar pengaruh transitivitas yang harus dimiliki *delivery predictability*.

2.6. Modifikasi Binary Spray and Wait menggunakan Delivery Predictability

Modifikasi *Binary Spray and Wait* menggunakan *Delivery Predictability* terdapat dua tahap yaitu :

- Tahap *Spray*, pada tahap ini setiap *node* yang bertemu akan mengupdate *delivery predictability* seperti halnya di *PROPHET*, dan *node* akan memberikan $\frac{L}{2}$ salinan pesan kepada *node* yang mempunyai nilai *delivery predictability* lebih tinggi.
- Tahap *Wait*, pada tahap ini sama seperti *BSW* yaitu setelah masing-masing *node* hanya memiliki satu salinan pesan akan mengirim pesan secara *direct transmission*.

3. METODOLOGI

3.1. Perancangan Sistem

Simulasi ini membandingkan *routing protocol* pada DTN yaitu *Binary Spray and Wait*, *PROPHET* dan modifikasi *Binary Spray and Wait*, dengan ketentuan:

- Perancangan dan simulasi diterapkan pada *Opportunistic Network Environment (ONE) Simulator*.
- *Node* menggambarkan pejalan kaki dan kendaraan.
- Mobilitas *node* menggunakan *Shortest Path Map Based Movement* dan *Random Way Point*. Untuk *Shortest Path Map Based Movement* menggunakan map Helsinki city Jepang.
- Pada saat konfigurasi *node*, yang diperhatikan yaitu jumlah *node* dan ukuran *buffer* masing-masing *node*. Parameter keluaran yaitu : 1. *delivery probability*, merupakan perbandingan antara bundle yang dikirim dan dibentuk , 2. *overhead ratio* merupakan Perbandingan jumlah salinan pesan yang dibuat dengan pesan yang terkirim, 3. *average latency*,

rata-rata waktu yang digunakan untuk mengirimkan pesan dari sumber ke tujuan, 4. *buffer time* yang mana merupakan lama waktu yang dibutuhkan pesan untuk antri di buffer node.

Tabel 1. Parameter Simulasi

No	Parameter Uji Simulasi	
1.	Waktu Simulasi	36000 s
2.	Jumlah Node	10, 50, 100, 200
3.	Interface	Wifi Interface
4.	Tipe Interface	Simple Broadcast
5.	Kecepatan Transmisi	6 Mbps
6.	Transmit Range	100 m
7.	Mobilitas	Shortest Path Map Based Movement, Random Way Point
8.	Ukuran Buffer	5 M, 10 M, 15M
9.	Kecepatan Node	25 to 27m/sec
10.	Ukuran Pesan	512 Kb
11.	Message Interval	25 to 35 sec
12.	Message TTL	300 minutes

3.2. Alur Simulasi

Pada simulasi penelitian ini dilakukan modifikasi *routing protocol Spray and Wait* pada bagian fase *Spray*. Kemudian *script* yang telah berhasil dimodifikasi tersebut disimulasikan dan dianalisis hasilnya berdasarkan parameter keluaran seperti *delivery probability*, *overhead ratio*, *average latency*, dan *buffer time*.

3.3. Pengaruh Jumlah Node

Tabel 2. Skenario 1 Pengaruh Jumlah Node

Skenario	Mobilitas	Jumlah Node
Skenario 1	Random Way Point, Map Based Movement	10
		50
		100
		200

Skenario perubahan jumlah *node* dilakukan untuk melihat performansi dari masing-masing *routing protocol* yang di analisis yaitu *Binary Spray and Wait*, *PROPHET* dan modifikasi *Binary Spray and Wait*, terhadap perubahan mobilitas *node*.

3.4. Pengaruh Ukuran Buffer

Skenario perubahan ukuran *buffer*

dilakukan untuk melihat performansi dari masing-masing *routing protocol* yang di analisis yaitu *Binary Spray and Wait*, *PROPHET* dan modifikasi *Binary Spray and Wait*, terhadap perubahan mobilitas *node*.

Tabel 3. Skenario 2 Pengaruh Ukuran Buffer

Skenario	Mobilitas	Ukuran Buffer
Skenario 2	Random Way Point, Map Based Movement	5 MB
		10 MB
		15 MB

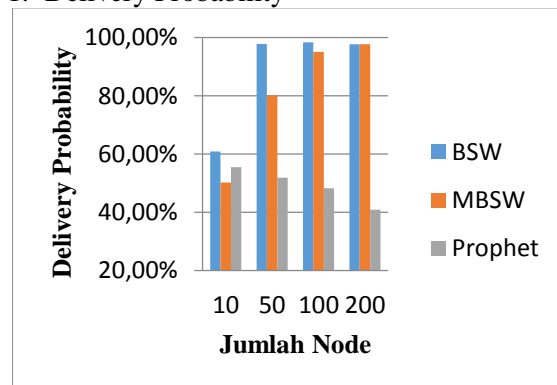
4. HASIL DAN PEMBAHASAN

Simulasi menggunakan ONE Simulator dengan keluaran data .txt kemudian diolah ke dalam bentuk grafik untuk dianalisis.

4.1. Analisis Performansi Protokol Routing Terhadap Perubahan Jumlah Node.

4.1.1. Random Way Point

1. Delivery Probability



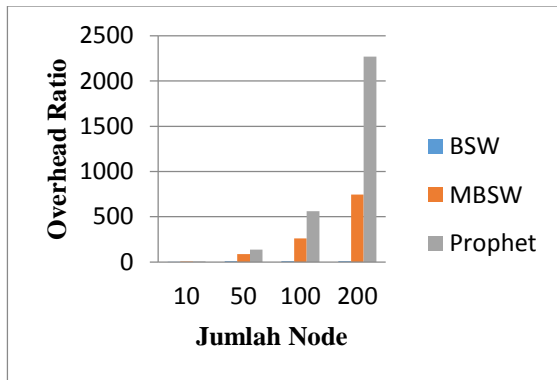
Gambar 3. Pengaruh Perubahan Jumlah Node Terhadap Delivery Probability

Untuk *Binary Spray and Wait* dan modifikasi tahap *spray* pada Modifikasi *Binary Spray and Wait*, nilai *delivery probability* semakin meningkat seiring dengan meningkatnya jumlah node. Hal tersebut karena semakin banyak jumlah node maka akan semakin banyak intensitas pertemuan node, namun pada modifikasi tahap *spray* pada *MBSW* nilai *delivery probability* tidak sebesar pada *Binary Spray and Wait* karena beban pada *buffer* yang lebih tinggi menyebabkan pesan yang diteruskan lebih sedikit. Sedangkan pada *PROPHET*, semakin meningkat jumlah node nilai *delivery probability*

semakin menurun, karena selain beban pada *buffer* yang lebih tinggi, juga tidak ada batasan jumlah salinan pesan, yang akan menambah beban pada *buffer*. Semakin banyak beban pada *buffer* semakin banyak juga pesan yang didrop, karena ukuran *buffer* terbatas.

Modifikasi Binary Spray and Wait menurunkan nilai *delivery probability* pada Binary Spray and Wait yaitu pada 10 node sebesar 17.51%, 50 node sebesar 18.12%, 100 node sebesar 3.34% dan pada node 200 sebesar 0%.

2. Overhead Ratio

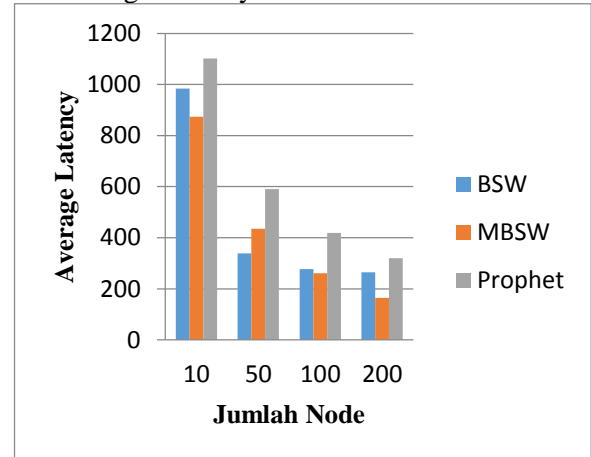


Gambar 4. Pengaruh Perubahan Jumlah Node Terhadap *Overhead Ratio*

Dari data Gambar 4 di atas dapat dilihat bahwa bahwa semakin bertambah jumlah node maka akan semakin tinggi nilai *overhead ratio*, karena akan semakin banyak jumlah pesan yang dikirim. Pada Binary Spray and Wait memiliki nilai *overhead ratio* paling kecil dibanding kedua *routing protocol* yang lain, karena salinan pesan yang dikirim terbatas dan hanya mengirimkan salinan pesan saja sehingga perbandingan jumlah pesan yang diteruskan dengan jumlah pesan yang terkirim semakin kecil. Berbeda dengan modifikasi tahap *spray* pada Modifikasi Binary Spray and Wait dimana selain mengirimkan salinan pesan yang sudah ditentukan, juga saling bertukar informasi *delivery predictability* yang menyebabkan perbandingan jumlah pesan yang diteruskan dengan pesan yang terkirim semakin lebih tinggi dibanding Binary Spray and Wait. Sedangkan pada *PROPHET* memiliki nilai *overhead ratio* paling tinggi dari kedua *routing protocol* yang lain, karena selain selama simulasi berlangsung saling bertukar informasi *delivery predictability*, *PROPHET* juga mengirimkan

pesan yang jumlahnya tidak ditentukan dan akan terus mengirim pesan ke node yang mempunyai nilai *delivery predictability* lebih tinggi sampai ke tujuan yang menyebabkan perbandingan jumlah pesan yang diteruskan dengan jumlah pesan yang terkirim besar.

3. Average Latency

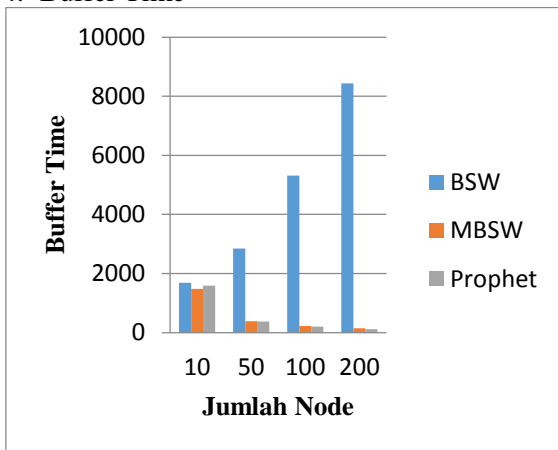


Gambar 5. Pengaruh Perubahan Jumlah Node Terhadap *Average Latency*

Dari data dapat dilihat bahwa semakin bertambah jumlah node semakin menurun nilai *average latency* masing-masing *routing protocol*. Pada *PROPHET* memiliki nilai *average latency* paling tinggi, karena mengirimkan pesan ke node yang mempunyai nilai *delivery predictability* lebih tinggi, dan hal tersebut membutuhkan waktu untuk menemukan node yang mempunyai *delivery predictability* lebih tinggi. *BSW* melakukan *direct transmission* pada tahap *wait* yang mana juga membutuhkan waktu sampai bertemu node yang dituju. Modifikasi tahap *spray* pada *MBSW* membuat nilai *average latency* kecil, karena pada tahap *spray* salinan pesan dikirimkan ke node yang mempunyai nilai *delivery predictability* paling tinggi sehingga tidak memerlukan waktu yang lama untuk *direct transmission* di tahap *wait*.

Modifikasi Binary Spray and Wait menyebabkan nilai *average latency* semakin cepat dibandingkan dengan Binary Spray and Wait, untuk 10 node lebih cepat sebesar 11.19%, namun pada node 50 MBSW lebih lama sebesar 28.69%, untuk 100 node lebih cepat sebesar 5.57%, dan untuk 200 node lebih cepat sebesar 37.91%.

4. Buffer Time



Gambar 6. Pengaruh Perubahan Jumlah Node Terhadap *Buffer Time*

Data menunjukkan bahwa untuk modifikasi tahap *spray* pada *MBSW* dan *PROPHET* cenderung menurun seiring pertambahan jumlah node. Hal tersebut karena pada kedua *routing protocol* tersebut mengirim pesan ke node yang mempunyai *delivery predictability* lebih tinggi. Berbeda dengan *Binary Spray and Wait* yang mana pada tahap *spray* mengirimkan salinan pesan ke setiap node yang ditemuinya yang menyebabkan pada tahap *wait* akan lama berada di *buffer* sampai *direct transmission*.

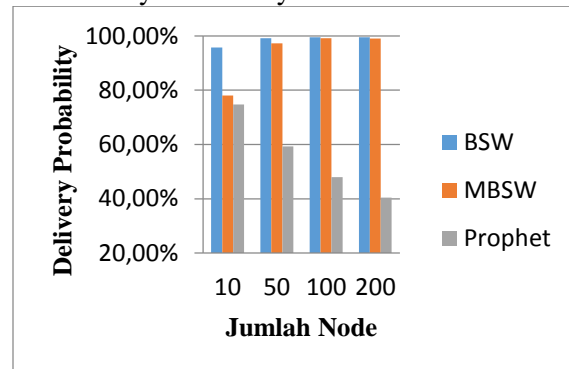
Modifikasi *Binary Spray and Wait* menyebabkan nilai *Buffer Time* semakin kecil dibandingkan dengan *Binary Spray and Wait*, untuk 10 node lebih kecil sebesar 11.96%, untuk 50 node lebih kecil sebesar 86.37%, untuk 100 node lebih kecil sebesar 95.88% dan untuk 200 node lebih kecil sebesar 98.24%.

4.1.2. Map Based Movement

Dari data Gambar 7 dapat dilihat bahwa *Binary Spray and Wait* dan modifikasi tahap *spray* pada Modifikasi *Binary Spray and Wait*, nilai *delivery probability* semakin meningkat seiring dengan meningkatnya jumlah node. Hal tersebut karena semakin banyak jumlah node maka akan semakin banyak *contact* antar node, namun pada modifikasi tahap *spray* pada *MBSW* nilai *delivery probability* tidak sebesar pada *Binary Spray and Wait* karena. Sedangkan pada *PROPHET*, semakin meningkat jumlah node nilai *delivery probability* semakin menurun, karena selain beban pada *buffer* yang lebih tinggi,

juga tidak ada batasan jumlah salinan pesan, yang akan menambah beban pada *buffer*. Semakin banyak beban pada *buffer* semakin banyak juga pesan yang didrop, karena ukuran *buffer* terbatas.

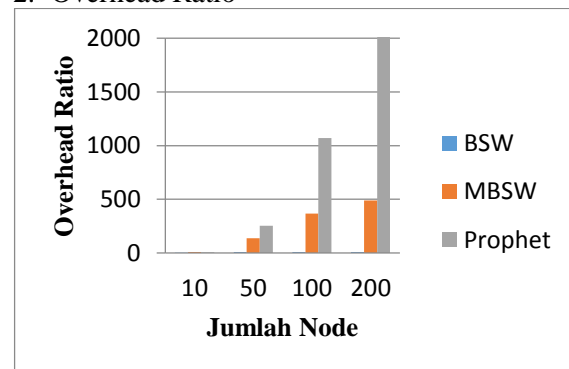
1. Delivery Probability



Gambar 7. Pengaruh Perubahan Jumlah Node Terhadap *Delivery Probability*

Modifikasi *Binary Spray and Wait* menurunkan nilai *delivery probability* pada *Binary Spray and Wait* yaitu untuk 10 node sebesar 18.52%, 50 node sebesar 1.98 %, 100 node sebesar 0.33 % dan 200 node sebesar 0.5 %.

2. Overhead Ratio

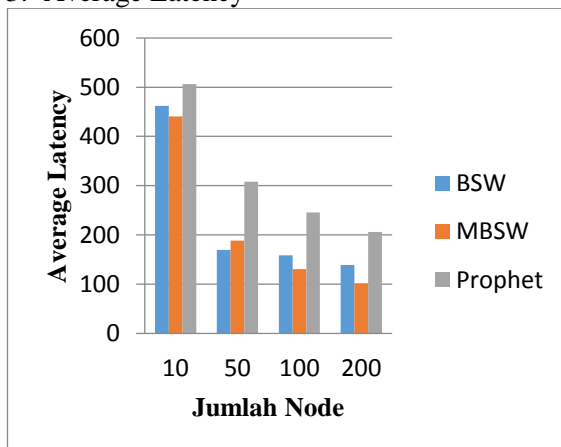


Gambar 8. Pengaruh Perubahan Jumlah Node Terhadap *Overhead Ratio*

Dari data Gambar 8 di atas dapat dilihat bahwa semakin bertambah jumlah node maka akan semakin tinggi nilai *overhead ratio*, karena akan semakin banyak jumlah pesan yang dikirim. Pada *Binary Spray and Wait* memiliki nilai *overhead ratio* paling kecil dibanding kedua *routing protocol* yang lain, karena salinan pesan yang dikirim terbatas dan hanya mengirimkan salinan pesan saja sehingga perbandingan jumlah pesan yang diteruskan dengan jumlah pesan yang

terkirim semakin kecil. Berbeda dengan modifikasi tahap *spray* pada Modifikasi *Binary Spray and Wait* dimana selain mengirimkan salinan pesan yang sudah ditentukan, juga saling bertukar informasi *delivery predictability* yang menyebabkan perbandingan jumlah pesan yang diteruskan dengan pesan yang terkirim semakin lebih tinggi dibanding *Binary Spray and Wait*. Sedangkan pada *PROPHET* memiliki nilai *overhead ratio* paling tinggi dari kedua *routing protocol* yang lain, karena selain selama simulasi berlangsung saling bertukar informasi *delivery predictability*, *PROPHET* juga mengirimkan pesan yang jumlahnya tidak ditentukan dan akan terus mengirim pesan ke node yang mempunyai nilai *delivery predictability* lebih tinggi sampai ke tujuan yang menyebabkan perbandingan jumlah pesan yang diteruskan dengan jumlah pesan yang terkirim besar.

3. Average Latency



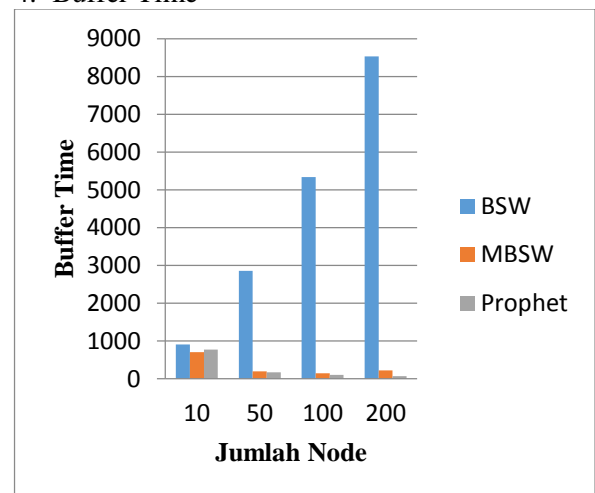
Gambar 9. Pengaruh Perubahan Jumlah Node Terhadap *Average Latency*

Data menunjukkan semakin bertambah jumlah node, semakin menurun nilai *average latency* masing-masing *routing protocol*, karena akan semakin banyak intensitas pertemuan antar node. Pada *PROPHET* memiliki nilai *average latency* paling tinggi, karena mengirimkan pesan ke node yang mempunyai nilai *delivery predictability* lebih tinggi, dan hal tersebut membutuhkan waktu untuk menemukannya. *BSW* melakukan *direct transmission* pada tahap *wait* yang mana juga membutuhkan waktu sampai ketemu node tujuan. Modifikasi tahap *spray* pada *MBSW* membuat nilai *average latency* lebih kecil dari *BSW*, karena pada tahap

spray salinan pesan dikirimkan ke node yang mempunyai nilai *delivery predictability* lebih tinggi sehingga tidak memerlukan waktu yang lama untuk *direct transmission* di tahap *wait*.

Modifikasi *Binary Spray and Wait* menyebabkan nilai *average latency* semakin cepat dibandingkan dengan *Binary Spray and Wait*, untuk 10 node lebih cepat sebesar 4.68%, namun pada node 50 *MBSW* lebih lama sebesar 11.05%, untuk 100 node lebih cepat sebesar 17.88%, dan untuk 200 node lebih cepat sebesar 27.31%.

4. Buffer Time



Gambar 10. Pengaruh Perubahan Jumlah Node Terhadap *Buffer Time*

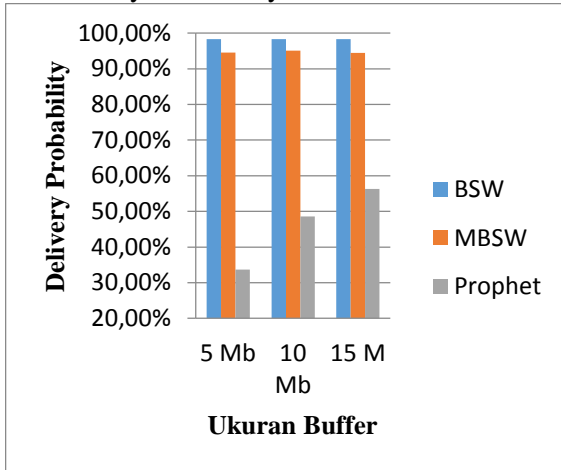
Data menunjukkan bahwa untuk modifikasi tahap *spray* pada *MBSW* dan *PROPHET* nilai *Buffer Time* cenderung menurun seiring pertambahan jumlah node. Hal tersebut karena pada kedua *routing protocol* tersebut mengirim pesan ke node yang mempunyai *delivery predictability* lebih tinggi. Berbeda dengan *Binary Spray and Wait* yang mana pada tahap *spray* mengirimkan salinan pesan ke setiap node yang ditemuinya yang menyebabkan pada tahap *wait* akan lama berada di *buffer* sampai *direct transmission*.

Modifikasi *Binary Spray and Wait* menyebabkan nilai *Buffer Time* semakin kecil dibandingkan dengan *Binary Spray and Wait*, untuk 10 node lebih kecil sebesar 22.15%, untuk 50 node lebih kecil sebesar 92.94%, untuk 100 node lebih kecil sebesar 97.19% dan untuk 200 node lebih kecil sebesar 97.37%.

4.2. Analisis Performansi Protokol Routing Terhadap Perubahan Ukuran Buffer.

4.2.1. Random Way Point

1. Delivery Probability

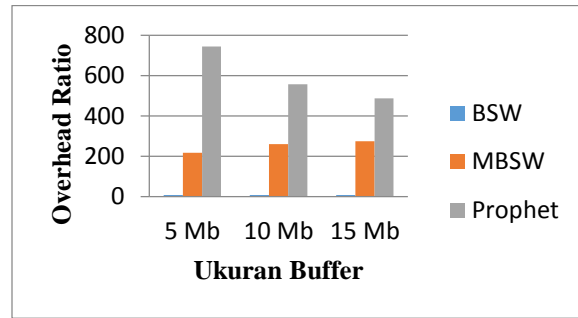


Gambar 11. Pengaruh Perubahan Ukuran Buffer Terhadap Delivery Probability

Dari data Gambar 11 dapat dilihat bahwa pada Binary Spray and Wait nilai delivery probability konstan, karena jumlah salinan pesan yang dikirim dan yang ditampung di buffer sama untuk setiap penambahan ukuran buffer. Modifikasi tahap spray pada MBSW memiliki nilai delivery probability yang meningkat seiring dengan penambahan ukuran buffer, karena walaupun jumlah salinan pesan sama, namun pada MBSW juga saling bertukar informasi delivery predictability yang mana membutuhkan ruang. Pada PROPHET nilai delivery probability meningkat seiring pertambahan ukuran buffer, sama halnya dengan MBSW melakukan pertukaran informasi delivery predictability, namun bedanya pada PROPHET jumlah salinan pesan yang dikirimkan tidak ada batasannya sehingga akan terus mengirim pesan ke node yang mempunyai nilai delivery predictability lebih tinggi sampai ke tujuan yang menyebabkan beban semakin banyak dan pesan yang didrop semakin banyak juga, sehingga nilai delivery probability nya lebih kecil dibandingkan dengan kedua routing protocol lainnya.

Modifikasi Binary Spray and Wait menurunkan nilai delivery probability pada Binary Spray and Wait yaitu untuk ukuran buffer 5 Mb sebesar 3.84%, ukuran buffer 10 Mb sebesar 3.34% dan untuk ukuran buffer 15 Mb sebesar 4.01%.

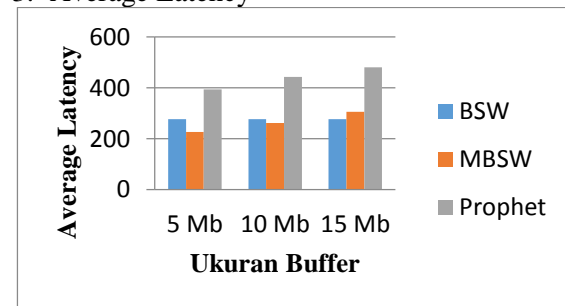
2. Overhead Ratio



Gambar 12. Pengaruh Perubahan Ukuran Buffer Terhadap Overhead ratio

Hasil menunjukkan overhead ratio pada PROPHET paling tinggi dari kedua routing protocol lainnya, karena pada PROPHET perbandingan jumlah pesan yang diteruskan dengan pesan yang terkirim paling besar diantara keduanya.

3. Average Latency

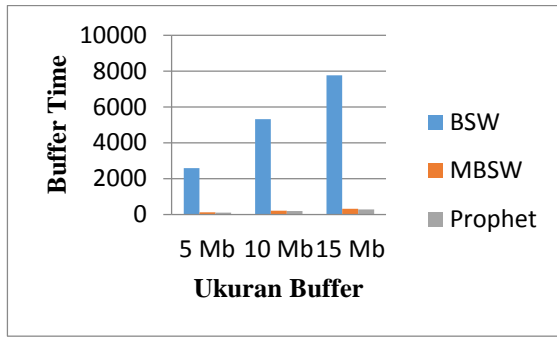


Gambar 13. Pengaruh Perubahan Ukuran Buffer Terhadap Average Latency

Hasil menunjukkan routing protocol modifikasi tahap spray pada MBSW mempunyai nilai average latency paling kecil karena pada tahap spray mengirimkan pesan ke node yang mempunyai nilai delivery predictability lebih tinggi sehingga direct transmission pada fase wait akan semakin cepat. BSW memiliki nilai average latency paling tinggi karena BSW mengirimkan salinan pesan ke setiap node yang ditemuinya tanpa melihat probabilitas pertemuan node sehingga lama sampai ke tujuan.

Modifikasi Binary Spray and Wait menyebabkan nilai average latency semakin cepat dibandingkan dengan Binary Spray and Wait, pada ukuran buffer 5 Mb lebih cepat sebesar 18.13%, ukuran buffer 10 Mb lebih cepat 5.57%, namun pada ukuran 15 Mb lebih lambat sebesar 10.32%.

4. Buffer Time



Gambar 14. Pengaruh Perubahan Ukuran Buffer Terhadap Buffer Time

Hasil menunjukkan bahwa *PROPHET* mempunyai nilai *buffer time* paling kecil, karena *routing protocol* tersebut mengirimkan pesan ke node yang mempunyai nilai *delivery predictability* tinggi sehingga pesan tidak lama berada di *buffer*. *Routing protocol* modifikasi tahap *spray* pada *Binary Spray and Wait* mempunyai nilai *buffer time* kecil namun masih di atas *PROPHET*, sama halnya dengan *PROPHET* modifikasi pada tahap *spray* pada *MBSW* meneruskan salinan pesan ke node yang mempunyai nilai *delivery predictability* lebih tinggi. *Binary Spray and Wait* mempunyai nilai *buffer time* paling tinggi dari kedua *routing protocol* yang lain, karena pada *BSW* meneruskan salinan pesan ke setiap node yang ditemuinya tanpa melihat probabilitas pertemuannya, sehingga akan membuat pesan lama berada di *buffer*.

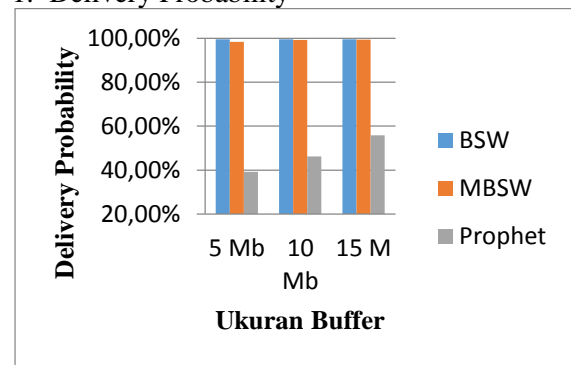
Modifikasi *Binary Spray and Wait* menyebabkan nilai *Buffer Time* semakin kecil dibandingkan dengan *Binary Spray and Wait*, pada ukuran *buffer* 5 Mb lebih kecil sebesar 95.16%, pada ukuran *buffer* 10 Mb lebih kecil sebesar 95.88% dan pada ukuran *buffer* 15 Mb lebih kecil sebesar 95.91%.

4.2.2. Map Based Movement

Dari data Gambar 15 dapat dilihat bahwa pada *Binary Spray and Wait* nilai *delivery probability* konstan, karena jumlah salinan pesan yang dikirim dan yang ditampung di *buffer* sama untuk setiap penambahan ukuran *buffer*. Modifikasi tahap *spray* pada *MBSW* memiliki nilai *delivery probability* yang meningkat seiring dengan penambahan ukuran *buffer*, karena walaupun jumlah salinan pesan sama, namun pada *MBSW* juga saling bertukar informasi

delivery predictability yang mana membutuhkan ruang. Pada *PROPHET* nilai *delivery probability* meningkat seiring pertambahan ukuran *buffer*, sama halnya dengan *MBSW* melakukan pertukaran informasi *delivery predictability*, namun bedanya pada *PROPHET* jumlah salinan pesan yang dikirimkan tidak ada batasannya sehingga akan terus mengirim pesan ke node yang mempunyai nilai *delivery predictability* lebih tinggi sampai ke tujuan yang menyebabkan beban semakin banyak dan pesan yang didrop semakin banyak juga, sehingga nilai *delivery probability* nya lebih kecil dibandingkan dengan kedua *routing protocol* lainnya.

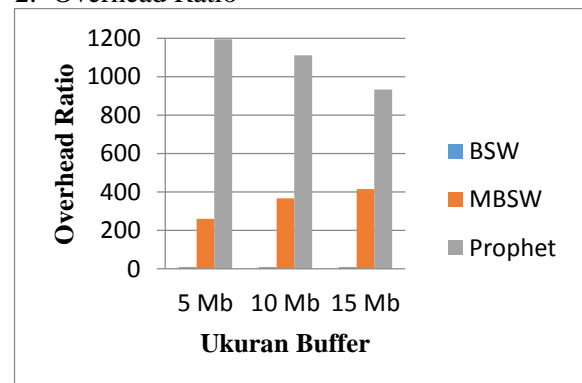
1. Delivery Probability



Gambar 15. Pengaruh Perubahan Ukuran Buffer Terhadap Delivery Probability

Modifikasi *Binary Spray and Wait* menurunkan nilai *delivery probability* pada *Binary Spray and Wait* yaitu untuk ukuran *buffer* 5 Mb sebesar 1.16%, ukuran *buffer* 10 Mb sebesar 0.33% dan untuk ukuran *buffer* 15 Mb sebesar 0.17%.

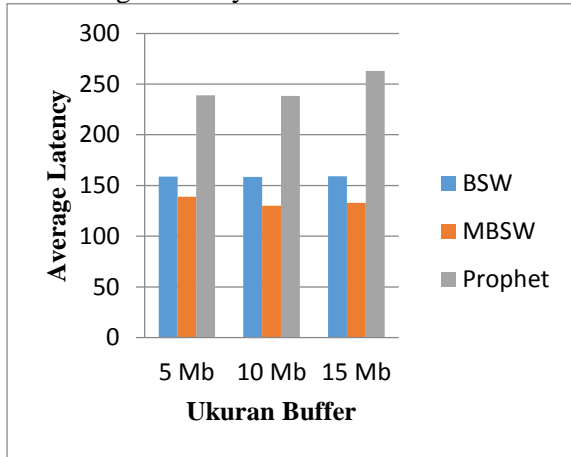
2. Overhead Ratio



Gambar 16. Pengaruh Perubahan Ukuran Buffer Terhadap Overhead ratio

Hasil menunjukkan *overhead ratio* pada *PROPHET* paling tinggi dari kedua *routing protocol* lainnya, karena pada *PROPHET* perbandingan jumlah pesan yang diteruskan dengan pesan yang terkirim paling besar diantara keduanya.

3. Average Latency

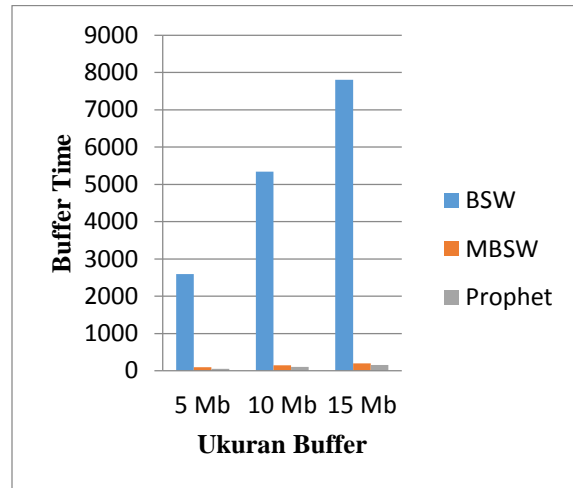


Gambar 17. Pengaruh Perubahan Ukuran *Buffer* Terhadap *Average Latency*

Hasil menunjukkan *PROPHET* mempunyai nilai *average latency* paling besar dibanding kedua *routing protocol* yang lain, karena pada *PROPHET* mengirimkan pesan ke node yang mempunyai nilai *delivery predictability* lebih tinggi sampai ke tujuan, yang mana membutuhkan waktu yang lama. *BSW* memiliki nilai *average latency* di bawah *PROPHET* karena *BSW* pada tahap *spray* mengirimkan salinan pesan ke setiap node yang ditemuinya tanpa melihat probabilitas pertemuan node sehingga pada tahap *wait* akan membutuhkan waktu yang lama sampai *direct transmission*. Modifikasi tahap *spray* pada *MBSW* mempunyai nilai *average latency* paling kecil karena pada tahap *spray* mengirimkan pesan ke node yang mempunyai nilai *delivery predictability* lebih tinggi sehingga *direct transmission* pada tahap *wait* akan semakin cepat.

Modifikasi *Binary Spray and Wait* menyebabkan nilai *average latency* semakin cepat dibandingkan dengan *Binary Spray and Wait*, pada ukuran buffer 5 Mb lebih cepat sebesar 12.53%, ukuran buffer 10 Mb lebih cepat 17.88%, dan pada ukuran 15 Mb lebih cepat sebesar 16.43%.

4. Buffer Time



Gambar 18. Pengaruh Perubahan Ukuran *Buffer* Terhadap *Buffer Time*

Hasil menunjukkan bahwa semakin besar ukuran *buffer* akan menyebabkan nilai *buffer time* untuk masing-masing *routing protocol* meningkat, karena jumlah pesan yang ditampung di *buffer* akan semakin banyak. *PROPHET* mempunyai nilai *buffer time* paling kecil, karena *routing protocol* tersebut mengirimkan pesan ke node yang mempunyai nilai *delivery predictability* tinggi sehingga pesan tidak lama berada di *buffer*. *Routing protocol* modifikasi tahap *spray* pada *MBSW* mempunyai nilai *buffer time* kecil namun masih di atas *PROPHET*, sama halnya dengan *PROPHET* modifikasi pada tahap *spray* meneruskan salinan pesan ke node yang mempunyai nilai *delivery predictability* lebih tinggi akan tetapi pada tahap *wait* membutuhkan waktu sedikit lebih lama sampai *direct transmission*. *Binary Spray and Wait* mempunyai nilai *buffer time* paling tinggi dari kedua *routing protocol* yang lain, karena pada tahap *spray* meneruskan salinan pesan ke setiap node yang ditemuinya tanpa melihat probabilitas pertemuannya, sehingga akan membuat pesan lama berada di *buffer*.

Modifikasi *Binary Spray and Wait* menyebabkan nilai *Buffer Time* semakin kecil dibandingkan dengan *Binary Spray and Wait*, pada ukuran *buffer* 5 Mb lebih kecil 96.12%, pada ukuran *buffer* 10 Mb lebih kecil sebesar 97.19% dan pada ukuran *buffer* 15 Mb lebih kecil sebesar 97.43%.

5. KESIMPULAN

Kesimpulan yang dapat diambil dari seluruh proses perancangan, simulasi dan analisis adalah sebagai berikut:

- Performansi modifikasi tahap *spray* pada Modifikasi *Binary Spray and Wait* menggunakan *delivery predictability* lebih unggul pada sisi *average latency* dan *buffer time* dibanding *Binary Spray and Wait*. Namun untuk *delivery probability* sedikit di bawah *BSW* dan *overhead ratio* jauh lebih tinggi dari *BSW*. Sedangkan jika modifikasi tahap *spray* pada *MBSW* dibandingkan dengan *PROPHET*, performansi *delivery probability*, *overhead ratio*, dan *average latency* lebih unggul namun *buffer time* lebih bagus *PROPHET*.
- Performansi *MBSW*, *Binary Spray and Wait*, dan *PROPHET* bagus pada model pergerakan node *Shortest Path Map Based Movement* dibandingkan pada *Random Way Point*.

DAFTAR PUSTAKA

- [1] Anjula Mehto and Meenu Chawla, Ph.D, "Modified Different Neighbor History Spray and Wait using PROPHET in Delay Tolerant Network", International Journal of Computer Applications (0975 – 8887), January 2014.
- [2] Bijal Patel, Krupa Dave and Vyomal Pandya, "Spray and Wait Routing Protocol in Delay Tolerant Networks", International Journal of Emerging Technology and Advanced Engineering, May 2014.
- [3] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks. SIGMOBILE Mob," Comput. Commun. Rev. vol. 7, no. 3, 2003
- [4] Keranen, Ari. "Opportunistic Network Environment Simulator", Helsinki University of Technology, May 2008.
- [5] Bhed Bahadur Bista and Dand B. Rawat. "Energy Consumption and Performance of Delay Tolerant Network Routing Protocols under Different Mobility Models", 7th International Conference on Intelligent Systems, Modelling and Simulation, 2016.
- [6] Carlo Caini, Haitham Cruickshank, Stephen Farrell and Mario Marchese, "Delay- and Disruption-Tolerant Networking (DTN): An Alternative Solution for Future Satellite Networking Applications", 2011.
- [7] Thrasyvoulos Spyropoulos, Konstantinos Psounis and Cauligi S. Raghavendra, "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks", 2005.
- [8] Sebastian Schildt and Lars Wolf, "Recent Trends: DTN Introduction & Applications", 2011.

Biodata Penulis

Bima Adhiguna, lahir di Padang, 26 November 1994. Menamatkan jenjang S1 di Telkom University pada bulan Juli 2016. Bekerja sebagai Laboran di Fakultas Teknik Elektro Telkom University dan saat ini sedang melanjutkan jenjang S2 di Universitas Komputer Indonesia.

Tody Ariefianto Wibowo, menamatkan jenjang S1 di prodi Teknik Elektro STT Telkom tahun 2006 dan S2 di prodi Teknik Elektro ITB tahun 2010. Saat ini bekerja sebagai dosen di Fakultas Teknik Elektro, Universitas Telkom. Aktif pada kegiatan riset mengenai Delay Tolerant Network dan Network Virtualization.

Leanna Vidya Yovita, menamatkan jenjang S1 di prodi teknik elektro STT Telkom tahun 2006 dan S2 di prodi teknik elektro IT Telkom tahun 2011. Bekerja sebagai dosen di Fakultas Teknik Elektro Universitas Telkom. Fokus pada riset mengenai Ad hoc Network, terutama Delay Tolerant Network.