

MEKANISME PEMILIHAN MPR DENGAN CONGESTION DETECTION DALAM OLSR PADA MANET

I Nyoman Buda Hartawan, Waskitho Wibisono

Jurusan Teknik Informatika, Fakultas Teknologi Informasi,
Institut Teknologi Sepuluh Nopember (ITS), Surabaya, 60111
E-mail: buda.hartawan@gmail.com, waswib@if.its.ac.id

ABSTRACT

MANET (Mobile Ad-hoc Network) is a collection of mobile nodes which communicate each others using wireless network. MANET does not have fixed network infrastructure. OLSR (Optimized Link State Routing) is a routing protocol in MANET. In OLSR there is a selection of neighbor nodes as MPR (Multi Point Relay) for forwarding packet data from source to destination. MANET characteristics which shared medium to communicate wirelessly, causing congestion vulnerable occurs in intermediate node when packet data transmission from source to destination. The congestion that occurs in network can lead to high packet loss and long delay, thus causing reduction of network performance. This study aims to perform the selection of MPR in OLSR with congestion detection methods in MANET. Results of the study showed that MPR selection with congestion detection in OLSR can improve packet delivery ratio and network throughput when packet transmission from source to destination. This study simulated on Network Simulator version 2 (NS-2).

ABSTRAK

MANET (Mobile Ad-hoc Network) terdiri dari mobile node yang saling berkomunikasi menggunakan jaringan wireless. MANET tidak memiliki infrastruktur jaringan yang tetap. OLSR (Optimized Link State Routing) adalah protokol routing proaktif pada MANET. Pada OLSR dilakukan pemilihan node tetangga sebagai MPR (Multi Point Relay) untuk dapat digunakan dalam meneruskan packet data dari source ke destination. Karakteristik MANET yang melakukan sharing media untuk berkomunikasi secara wireless menyebabkan kongesti rentan terjadi pada intermediate node ketika pengiriman paket data dari source ke destination. Kongesti yang terjadi pada jaringan dapat menyebabkan tingginya packet loss dan delay yang panjang, sehingga pada akhirnya akan mengakibatkan penurunan kinerja jaringan. Penelitian ini bertujuan untuk melakukan pemilihan MPR dalam OLSR dengan metode congestion detection pada MANET. Hasil dari penelitian menunjukkan pemilihan MPR dengan congestion detection pada OLSR mampu meningkatkan packet delivery ratio dan throughput jaringan ketika pengiriman packet dari source ke destination. Penelitian ini disimulasikan pada Network Simulator versi 2 (NS-2).

Kata Kunci: MANET, OLSR, Multi-Point Relay, congestion detection, NS-2.

1 PENDAHULUAN

MANET (*Mobile Ad-hoc Network*) merupakan jaringan yang terdiri dari *mobile node* yang saling berkomunikasi secara *wireless*. *Node-node* pada MANET dapat bergerak secara bebas, bergabung atau keluar dari jaringan setiap waktu. Hal ini menyebabkan topologi MANET mengalami perubahan secara cepat. Selain itu keterbatasan sumber daya seperti *bandwidth*, *memory*, dan *energy*, serta adanya *sharing media* dalam komunikasi menyebabkan MANET rentan mengalami kegagalan komunikasi (Jain dkk., 2012). Membangun rute *routing* yang efektif adalah salah satu tantangan untuk meningkatkan kualitas jaringan pada MANET.

Protokol *routing* pada MANET diklasifikasikan menjadi 3 (tiga) yaitu *reactive*, *proactive* dan *hybrid* (Abolhasan dkk., 2004). OLSR (*Optimized Link State Routing*) adalah sebuah protokol *routing* yang dikembangkan untuk MANET dan tergolong kedalam protokol *routing proactive*. Protokol *routing proactive* membangun tabel *routing* untuk menyimpan rute pengiriman *packet* data. Pada protokol OLSR, setiap *node* memilih *node* tetangganya sebagai MPR (*Multi-Point Relay*) yang bertugas meneruskan transmisi ke seluruh jaringan. MPR digunakan untuk membangun rute dari *source* ke *destination*. Setiap *node* mengelola informasi tentang *set* tetangganya yang dipilih sebagai MPR (Jacquet dkk., 2001).

Pada MANET kongesti rentan terjadi pada *intermediate node*. Kongesti dalam jaringan dapat

terjadi karena jumlah *packet* yang tiba pada *node* melebihi kapasitas *buffer node* tersebut, sehingga *node* menjadi sesak dan mulai kehilangan *packet* data (Kumaran dan Sankaranarayanan, 2011). Kongesti yang terjadi pada jaringan *wireless* dapat menyebabkan *packet loss* dan *delay* yang tinggi ketika pengiriman *packet* data dari *source* ke *destination* (Jain dkk., 2012). Hal ini kemudian mengakibatkan kinerja jaringan menurun. Pemilihan MPR yang tepat sangat menentukan kualitas pengiriman *packet* data pada OLSR.

Untuk itu diusulkan penerapan metode *congestion detection* untuk pemilihan MPR dalam membangun rute *routing* pada OLSR. Pemilihan MPR diprioritaskan pada *node* yang menggunakan *buffer* lebih sedikit. Kongesti pada MANET dideteksi dengan mengamati nilai *Threshold* pada *buffer*. Pendeteksian kongesti dalam suatu jaringan dengan menggunakan *buffer* juga dilakukan oleh (Popa dkk., 2006; Ahmad dan Turgut, 2008; Razzaque dan Hong, 2009; Kumaran dan Sankaranarayanan, 2011). Penggunaan *buffer* yang tinggi dapat menyebabkan antrean padat dan meningkatkan *delay* (Jamshaid dkk., 2011).

Apabila semua *node* dalam jaringan mengalami kongesti, maka pemilihan MPR selanjutnya didasarkan pada *Energy level* yang dimiliki oleh setiap *node*. *Energy level* pada *node* akan diberikan nilai *Threshold* sebesar 30% untuk menentukan bahwa *node* tersebut mampu meneruskan *packet* data. Dalam komunikasi pada OLSR, *node-node* akan menyebarkan status *level* kongesti dan *Energy level*-nya ke *node-node* tetangganya secara periodik untuk meng-*update* tabel *routing*. Dengan informasi tersebut maka diperoleh rute pengiriman *packet* dari *source* ke *destination*.

Penerapan *congestion detection* untuk pemilihan MPR pada OLSR diharapkan dapat meningkatkan kinerja protokol OLSR pada MANET. Sistem yang diusulkan selanjutnya disebut CD-OLSR (*Congestion Detection Optimized Link State Routing*).

2 METODE

Algoritma RED (*Random Early Detection*) merupakan algoritma *active queue management* (AQM). RED menghitung rata-rata ukuran *queue* dengan memberikan nilai ambang batas (*Threshold*). Rata-rata ukuran *queue* dibandingkan untuk dua nilai *Threshold*, *minimum Threshold* dan *maximum Threshold* (Floyd dan Jacobson, 1993).

RED digunakan oleh (Kumaran dan Sankaranarayanan, 2011) untuk mendeteksi kongesti pada protokol reaktif, AODV (*Ad-hoc On-Demand Distance Vector*). Nilai *Threshold* yang digunakan pada penelitian tersebut adalah:

$$Min_Thr = 0.25 * Buffer_Size \quad (1)$$

$$Max_Thr = 3 * Min_Thr \quad (2)$$

$$Avg = ((1 - wq) * Avg) + (Cur_Que * wq) \quad (3)$$

$$Status_Que = Cur_Que - Avg \quad (4)$$

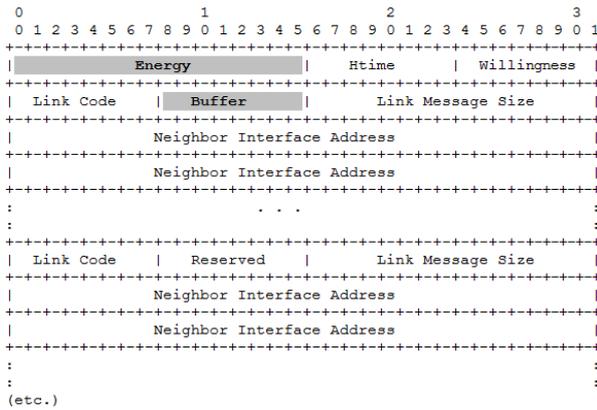
Min_Thr merupakan nilai batas *minimum Threshold*, sedangkan *Max_Thr* merupakan nilai batas maksimum *Threshold*. *Avg* merupakan nilai rata-rata dari ukuran *queue* (*Average Queue*). *wq* merupakan bobot (*weight*) *queue* dengan nilai konstan 0.002. *Cur_Que* merupakan ukuran *queue* saat ini (*Current Queue*), sedangkan *Status_Que* merupakan kondisi dari nilai *queue* yang sebenarnya.

Paket data yang datang dalam jaringan akan memenuhi *queue*, sehingga semakin banyak *packet* yang datang semakin berkurang ukuran *queue*. Jika *Status_Que* kurang dari *Min_Thr*, maka *node* diklasifikasikan ke dalam Area I (area aman dari kongesti) dengan nilai 0 (nol). Apabila *Status_Que* lebih dari *Min_Thr* dan *Cur_Que* kurang dari *Max_Thr*, maka *node* diklasifikasikan ke dalam Area II (Area mendekati kongesti) dengan nilai 1 (satu), sedangkan jika *Cur_Que* lebih dari *Max_Thr* maka *node* diklasifikasikan ke dalam Area III (Area dalam keadaan kongesti) dengan nilai 2 (dua).

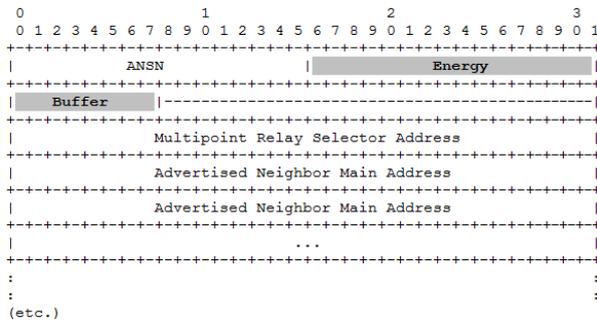
Dalam penelitian diterapkan metode pendeteksian kongesti dengan algoritma RED pada protokol OLSR. Penerapan algoritma RED digunakan pada sistem pemilihan *node* sebagai MPR pada OLSR yang didasarkan pada *buffer level* pada setiap *node*. Hal ini bertujuan untuk membangun rute pengiriman *packet* data dengan *level* kongesti yang lebih rendah dari *source* ke *destination*. Rute *routing* dengan *level* kongesti yang lebih rendah mampu meningkatkan kualitas pengiriman *packet* data. Untuk mengukur kualitas pengiriman *packet* data digunakan parameter pengujian seperti *packet delivery ratio*, *end-to-end delay* dan *throughput*.

Protokol OLSR untuk MANET pertama kali diusulkan oleh Jacquet dkk (2001). Protokol OLSR adalah optimasi dari protokol *link state* murni untuk MANET. Pada protokol OLSR, setiap *node* memilih *node* tetangganya sebagai *Multi-Point Relay* (MPR) yang bertugas meneruskan transmisi ke seluruh jaringan. MPR digunakan untuk membangun rute dari *source* ke *destination*. Untuk membangun tabel *routing* OLSR menggunakan informasi dari pesan kontrol, seperti pesan HELLO dan pesan TC (*Topology Control*). Pesan HELLO digunakan untuk melakukan *link sensing*, *neighbor detection*, dan MPR *signaling*. Pesan TC digunakan untuk mendeklarasikan topologi ke seluruh *node* jaringan.

Pada penelitian ini, *format* pesan HELLO dan pesan TC dimodifikasi dengan menambahkan *field buffer* dan *Energy* untuk menyimpan nilai status *buffer* dan *Energy* dari setiap *node* (lihat Gambar 1 dan Gambar 2). Ketika menerima pesan ini *node* akan mengetahui status *buffer* dan *Energy node* tetangganya.



Gambar 1 Modifikasi Packet Pesan HELLO



Gambar 2 Modifikasi Packet Pesan TC

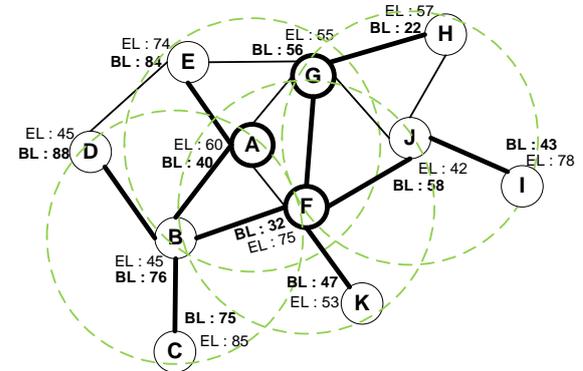
2.1 Pemilihan Node Sebagai MPR

Pemilihan *node* sebagai MPR bertujuan untuk memperoleh MPR yang sesuai untuk membangun rute pengiriman *packet* data dari *source* ke *destination*. Dalam pemilihan MPR informasi yang diperoleh dari pesan *HELLO* mengenai status *buffer level* dan *Energy level* setiap *node* tetangga menjadi faktor penting untuk menentukan sebuah *node* terpilih sebagai MPR. Untuk lebih memahami proses pemilihan MPR yang terjadi, pada tahapan ini diuraikan beberapa kasus yang mungkin terjadi selama proses pemilihan MPR.

2.1.1 Kasus 1, Buffer kurang dari Max_Thr dan Energy lebih dari Threshold

Gambar 3 merupakan contoh MANET dengan nilai *buffer level* (BL) dan *Energy level* (EL) pada setiap *node*-nya. Misalkan *node* B merupakan *source* yang akan mengirimkan *packet* data. Pada tahapan ini *node* B akan memilih *node* tetangganya sebagai MPR untuk membangun rute pengiriman *packet* data ke *destination* H. Pada Gambar 3, *node* A, C, D, F merupakan 1-hop tetangga dari *node* B. *Node* E, G, J, K adalah 2-hop tetangga dari *node* B. Dalam pemilihan *node*

tetangga sebagai MPR, *buffer level* dan *Energy level* pada *node* tersebut dibandingkan. Dalam kasus ini *buffer level* dan *energy level* pada *node* A, D, F dibandingkan untuk dipilih sebagai MPR.



Gambar 3 Buffer kurang dari Max_Thr dan Energy lebih dari Threshold

Pada Gambar 3 dapat dilihat *Node* F memiliki *buffer level* paling kecil (BL:32) dibandingkan dengan *node* A (BL:40) dan D (BL:88), sedangkan *node* A memiliki *buffer level* lebih kecil daripada *node* D. *Node* F memiliki *Energy level* tertinggi (EL:75) dibandingkan dengan *node* A (EL:60) dan D (EL:45), dan *node* A memiliki *Energy level* lebih tinggi dari *node* D. *Node* F memiliki derajat tertinggi (5), kemudian diikuti oleh *node* A (4) dan D (2).

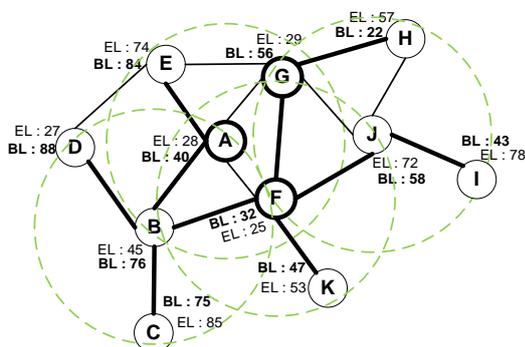
CD-OLSR yang diusulkan memprioritaskan *buffer level* dibandingkan parameter yang lainnya, sehingga B memilih F sebagai MPR untuk *node* G, J, K. Karena *node* E masih belum ter-cover oleh *node* F maka B melakukan pemilihan lagi. Dalam kasus ini A dipilih sebagai MPR untuk E karena *buffer level* dari A lebih kecil daripada D. *Node* F pun juga melakukan hal yang sama dengan *node* B dalam memilih MPR untuk 2-hop tetangganya *node* H dan I. *Node* F memilih *node* G sebagai MPR untuk 2-hop tetangganya karena *node* G memiliki *buffer level* lebih kecil daripada *node* J. Sehingga apabila B mengirimkan *packet* data ke H, rute yang dilalui adalah B→F→G→H.

2.1.2 Kasus 2, Buffer kurang dari Max_Thr dan Energy kurang dari Threshold

Pada Gambar 4 *node* B memilih *node* F sebagai MPR untuk 2-hop tetangga dari *node* B yaitu G, J, K karena *buffer level*-nya paling kecil walaupun *Energy level*-nya lebih kecil dari *node* A dan D.

Karena *node* E yang merupakan 2-hop tetangga dari *node* B belum dapat ter-cover oleh *node* F maka pemilihan MPR dilakukan kembali, dan *node* B memilih *node* A sebagai MPR untuk *node* E karena memiliki *buffer level* lebih rendah dari *node* D walaupun *Energy level* *node* A lebih kecil dari *node* D. *Node* F juga melakukan hal yang

sama seperti *node* B. Sehingga rute pengiriman *packet* data ketika *node* B akan mengirim *packet* data ke *node* H pada kasus ini adalah $B \rightarrow F \rightarrow G \rightarrow H$.

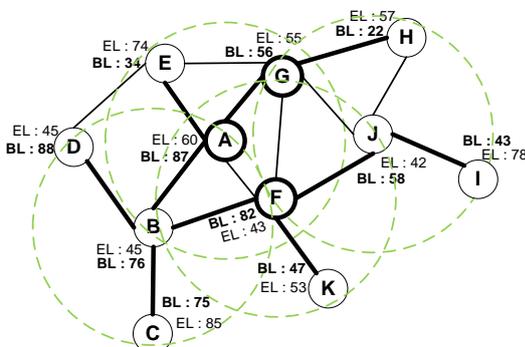


Gambar 4 *Buffer* kurang dari *Max_Thr* dan *Energy* kurang dari *Threshold*

2.1.3 Kasus 3, *Buffer* lebih dari *Max_Thr* dan *Energy* lebih dari *Threshold*

Pada Gambar 5 *node* A, D, F memiliki *buffer level* yang mencapai *Threshold* maksimum namun *Energy level*-nya belum mencapai *Threshold*. Ketika *node* B mengirim *packet* data ke *node* H, maka *node* B pada kasus ini akan memilih *node* dengan *Energy level* tertinggi. Hal ini dilakukan karena *buffer level* sudah mencapai *Threshold* maksimum, atau dengan kata lain semua *node* mengalami kongesti.

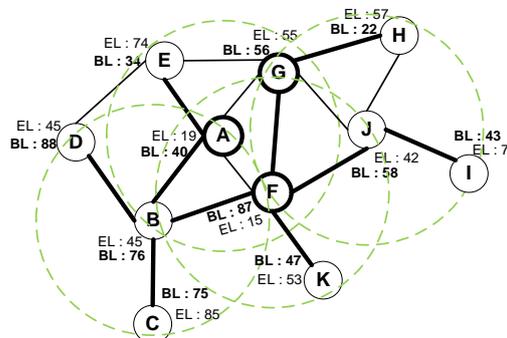
Node B memilih *node* A sebagai MPR untuk 2-hop tetangganya yaitu *node* E dan G. Untuk sisa 2-hop tetangga yang lainnya *node* B memilih *node* F sebagai MPR. Pemilihan rute ini berdasarkan *node* yang memiliki *Energy* lebih tinggi dibandingkan dengan *node* yang lainnya, karena dalam kondisi kongesti *node* akan lebih cepat kehilangan *Energy*. Sehingga rute yang dilalui oleh *node* B ketika mengirim *packet* data ke *node* H adalah $B \rightarrow A \rightarrow G \rightarrow H$.



Gambar 5 *Buffer* lebih dari *Max_Thr* dan *Energy* lebih dari *Threshold*

2.1.4 Kasus 4, *Buffer* lebih dari *Max_Thr* dan *Energy* kurang dari *Threshold*

Pada kasus ini prioritas pemilihan MPR didasarkan pada derajat *node* tertinggi yang mampu meng-cover 2-hop tetangganya.



Gambar 6 *Buffer* lebih dari *Max_Thr* dan *Energy* kurang dari *Threshold*

Derajat *node* merupakan jumlah *node* tetangga 1-hop yang dimiliki oleh sebuah *node*. Pada Gambar 6 *node* B memilih *node* F sebagai MPR untuk 2-hop tetangganya yaitu *node* G, J, K, karena *node* F memiliki derajat tertinggi yaitu 5 (A, B, G, J dan K) dibandingkan dengan *node* A yang memiliki derajat 4 (B, E, F, G) dan D yang memiliki derajat 2 (B dan E). Sehingga rute pengiriman *packet* data dari B ke H adalah $B \rightarrow F \rightarrow G \rightarrow H$.

Pada penelitian ini pemilihan MPR diprioritaskan pada *node* yang memiliki nilai penggunaan *buffer* lebih rendah dan mampu mencakup 2-hop tetangganya. Hal ini berkaitan dengan kemampuan *node* tersebut untuk meneruskan *packet* data ke *node* selanjutnya. *Node* dengan penggunaan *buffer* yang rendah mampu meningkatkan pengiriman *packet* data dengan mengurangi *packet loss* yang mungkin terjadi yang diakibatkan oleh *packet* yang datang tidak mendapatkan ruang pada *node* untuk menunggu akan diproses. Semakin rendah penggunaan *buffer* pada sebuah *node*, semakin besar kemungkinan *node* tersebut dipilih sebagai MPR.

Dalam pemilihan MPR informasi yang diperoleh dari pesan *HELLO* mengenai status *buffer level* dan *Energy level* setiap *node* tetangga menjadi faktor penting untuk menentukan sebuah *node* terpilih sebagai MPR. Berikut ini dijelaskan secara lebih terperinci tentang flowchart yang ditunjukkan oleh Gambar 7.

1. *Start MPR set* semua anggota N dengan memberikan nilai $N_willingness$ sama dengan *WILL_ALWAYS*. *Willingness* adalah tingkat kerelaan sebuah *node* untuk meneruskan *packet* data ke *node* selanjutnya. $N_willingness$ sama dengan *WILL_ALWAYS* berarti setiap *node* selalu

dapat dipilih sebagai MPR untuk meneruskan *packet data*;

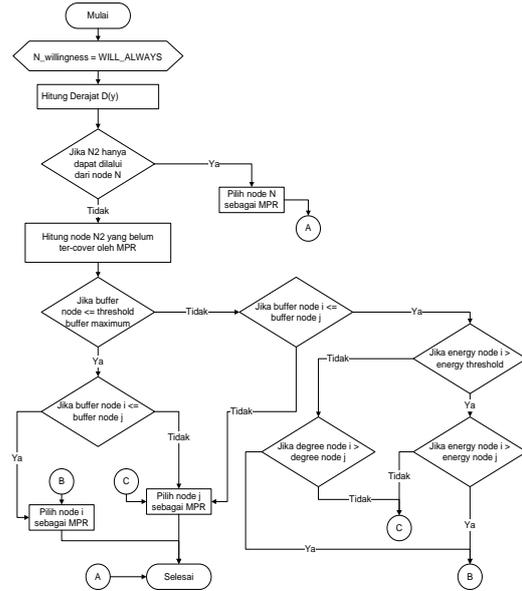
2. Hitung derajat y (disimbolkan dengan $D(y)$), dimana y adalah anggota N untuk semua *node* N ;
3. Pilih sebagai MPR *node* pada N dimana hanya *node* tersebut yang menyediakan *link* ke *node* di $N2$. Contoh, jika *node* B di $N2$ dapat dicapai hanya melalui *link symmetric* ke *node* A di N , maka pilih *node* A sebagai MPR *set*;
4. Apabila terdapat *node* di $N2$ yang belum ter-cover oleh MPR, maka:
 - i). Setiap *node* di N , hitung jumlah *node* di $N2$ yang belum ter-cover oleh MPR dan dapat dicapai melalui 1-hop tetangga.

- ii). Pilih sebagai MPR *node* pada N tersebut yang men meng-cover *node* $N2$ dan memiliki nilai penggunaan *buffer* lebih sedikit. Apabila *buffer level* mencapai *Threshold* maksimum maka pemilihan MPR diprioritaskan pada nilai *Energy level* tertinggi. Apabila *buffer level* mencapai *Threshold* maksimum dan *Energy level* mencapai *Threshold* minimum pada semua *node*, maka akan dipilih *node* yang memiliki derajat $D(y)$ lebih tinggi sebagai MPR.

5. Jika semua *node* di $N2$ masih ter-cover oleh paling sedikitnya satu MPR *set* selain *node* y dan jika $N_willingness$ dari *node* y lebih kecil daripada $WILL_ALWAYS$ maka *node* y dihapus dari MPR *set*.

6. SKENARIO UJI COBA

Untuk menguji kinerja CD-OLSR yang diusulkan dilakukan pengujian dengan memberikan variasi jumlah koneksi *end-to-end* dari *source* ke *destination* dengan jumlah koneksi *end-to-end* 10, 20, 30, 40, dan 50 koneksi *end-to-end* pada 100 *node*. Koneksi *end-to-end* adalah jumlah koneksi yang terjadi dari *source* ke *destination* selama waktu simulasi. Jumlah koneksi *end-to-end* diatur terus meningkat dengan tujuan untuk meningkatkan pertukaran *packet data* pada jaringan, sehingga kepadatan jaringan menjadi lebih tinggi. Hal ini dilakukan untuk mengetahui pengaruh jumlah koneksi *end-to-end* terhadap *packet delivery ratio*, *end-to-end delay*, dan *throughput*.



Gambar 7 Flowchart Komputasi MPR

CBR load yang digunakan pada skenario ini adalah 4 *packet/s*. CBR merupakan nilai konstan yang diberikan pada setiap pengiriman *packet data*. Setiap *node* akan bergerak dengan kecepatan mulai dari 1m/s sampai 10m/s mengikuti model pergerakan *Random Waypoint* yang digunakan dengan *pause time* 30s. Artinya setiap *node* dalam jaringan akan bergerak secara bebas dan akan berhenti selama 30s ketika sudah mencapai tujuan. Jika *pause time* sudah terpenuhi maka *node* akan bergerak kembali ke tujuan yang baru. Hal ini dilakukan berulang-ulang sampai waktu simulasi selesai.

2.2 Parameter Simulasi

Parameter simulasi merupakan parameter-parameter yang digunakan dalam simulasi pengujian CD-OLSR.

Tabel 1 Parameter Simulasi

No	Parameter	Spesifikasi
1.	<i>Number of nodes</i>	100 node
2.	<i>Simulation area</i>	1600 m x 1600 m
3.	<i>Radio range</i>	250 m with 2Mbps Bandwidth
4.	<i>MAC protocol</i>	IEEE 802.11 DCF
5.	<i>Propagation Model</i>	Two-ray ground reflection model
6.	<i>Mobility model</i>	Random Way Point
7.	<i>Interface queue</i>	50 packets max
8.	<i>CBR</i>	4 packet/s
9.	<i>Number of flows</i>	10 - 50 flows
10.	<i>Node speed maximum</i>	10m/s
11.	<i>Simulation time</i>	900 s
12.	<i>Initial Energy</i>	1000 joule

2.3 Packet Delivery Ratio (PDR)

Nilai PDR adalah perbandingan antara jumlah *packet* yang diterima oleh *destination* dan jumlah *packet* yang dikirimkan oleh *source*. Satuan yang digunakan adalah persen (%). Semakin besar nilainya semakin baik. Nilai PDR dihitung menggunakan rumus berikut.

$$PDR(\%) = \frac{\sum received_packets}{\sum sent_packets} \times 100\% \quad (5)$$

2.4 End-to-End Delay

Nilai *end-to-end delay* adalah waktu yang dibutuhkan oleh *packet* ketika pengiriman *packet* dari *source* ke *destination*. Satuan yang digunakan adalah *second* (s). Semakin rendah nilainya semakin baik. Nilai *end-to-end delay* dihitung menggunakan rumus berikut.

$$E2eDelay = \frac{\sum waktu_tempuh_seluruh_packets}{\sum received_packets} \quad (6)$$

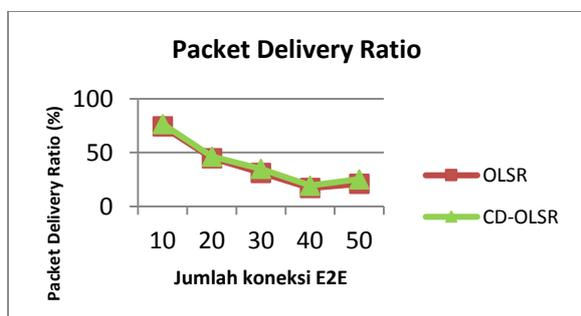
2.5 Throughput

Nilai *throughput* merupakan total jumlah *packet* data yang diterima per detik. Satuan yang digunakan adalah *kilo-bit-per-second* (kbps). Semakin besar nilai *throughput* semakin baik. Nilai *throughput* dihitung dengan menggunakan rumus:

$$Throughput = \frac{\sum received_packets_size}{(StopTime - StartTime)} \quad (7)$$

3 EVALUASI KINERJA

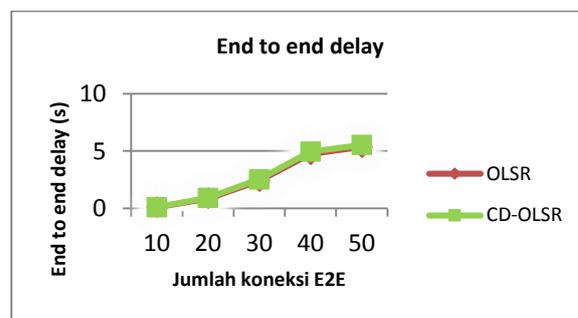
Untuk mengevaluasi kinerja CD-OLSR analisis dilakukan terhadap nilai parameter-parameter pengujian yang digunakan seperti *packet delivery ratio*, *end-to-end delay* dan *throughput* jaringan yang telah diperoleh.



Gambar 8 Pengaruh Variasi Koneksi *End-to-End* Terhadap *PDR*

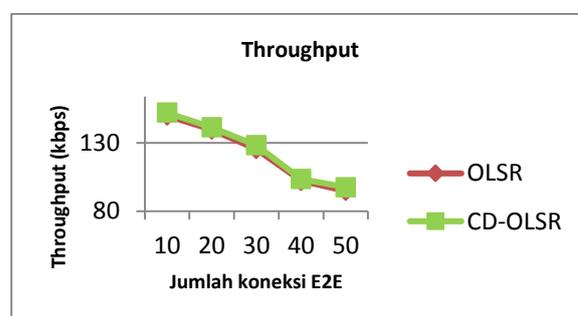
Pada Gambar 8 dapat dilihat bahwa penurunan PDR terjadi pada kedua protokol OLSR dan CD-OLSR seiring dengan meningkatnya

jumlah koneksi *end-to-end* yang terjadi. Hal ini disebabkan semakin meningkatnya jumlah koneksi *end-to-end* yang diberikan, *packet* data yang berada dalam jaringan semakin banyak sehingga mengakibatkan kepadatan jaringan semakin tinggi. Namun demikian CD-OLSR memiliki *packet delivery ratio* yang lebih tinggi dibandingkan dengan OLSR. Ini berarti pemilihan *node* sebagai MPR pada OLSR mengalami perbaikan sehingga mampu meningkatkan kualitas pengiriman *packet* data.



Gambar 9 Pengaruh Variasi Koneksi *End-to-End* Terhadap *End-to-End Delay*

Pada Gambar 9 dapat dilihat bahwa *end-to-end delay* pada kedua protokol OLSR dan CD-OLSR mengalami peningkatan seiring dengan peningkatan koneksi *end-to-end*. Artinya waktu yang dibutuhkan dalam pengiriman *packet* dari *source* ke *destination* semakin meningkat. Protokol OLSR memiliki *end-to-end delay* lebih rendah dibandingkan dengan CD-OLSR. Hal ini disebabkan pada CD-OLSR terdapat penambahan proses perhitungan pada pemilihan *node* sebagai MPR untuk membangun rute dengan *level* kongesti yang lebih rendah. Adanya penambahan proses perhitungan tersebut mengakibatkan meningkatnya waktu proses pada setiap *node*. Sehingga *end-to-end delay* yang dihasilkan menjadi lebih tinggi.



Gambar 10 Pengaruh Variasi Koneksi *End-to-End* Terhadap *Throughput*

Pada Gambar 10 dapat dilihat bahwa *throughput* pada kedua protokol OLSR dan CD-OLSR mengalami penurunan seiring dengan pen-

ingkatan jumlah koneksi *end-to-end*. CD-OLSR memiliki *throughput* yang lebih tinggi daripada OLSR. Hal ini membuktikan bahwa dengan adanya perbaikan mekanisme pemilihan MPR maka CD-OLSR mampu menyediakan rute yang lebih handal dalam pengiriman *packet* data dari *source* ke *destination*.

4 KESIMPULAN

Pada paper ini telah diusulkan mekanisme pemilihan *node* sebagai MPR dalam OLSR dengan metode *congestion detection*. Adaptasi metode *congestion detection* untuk pemilihan *node* sebagai MPR memberikan perbaikan terhadap *packet delivery ratio* dan *throughput* jaringan pada protokol OLSR. Namun, dari hasil penelitian menunjukkan bahwa *end-to-end delay* pada CD-OLSR lebih tinggi dibandingkan dengan protokol OLSR.

5 DAFTAR PUSTAKA

- Abolhasan, M., Wysocki, T. dan Dutkiewicz, E. (2004), "A review of routing protocols for mobile ad hoc networks", *Ad Hoc Networks* 2, hal. 1-22.
- Ahmad, M.Z. dan Turgut, D. (2008), "Congestion avoidance and fairness in wireless sensor networks", *Global Telecommunication Conference*, New Orleans, USA.
- Floyd, S. dan Jacobson, V. (1993), "Random Early Detection Gateways for Congestion Avoidance", *IEEE/ACM Transactions Networking*, vol. 1, no. 4, hal. 397-413.
- Jacquet, P. dkk. (2001), "Optimised Link State Routing Protocol for Ad hoc Networks", *IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International*, Le Chesnay, Prancis, hal.62- 68.
- Jain, S., Kokate, S., Thakur, P. dan Takalkar, S. (2012), "A Study of Congestion Aware Adaptive Routing Protocols in MANET", *Computer Engineering and Intelligent Systems*, vol. 3, no. 4.
- Jamshaid, K., Shihada, B., LiXia dan Levis, P. (2011), "Buffer Sizing in 802.11 Wireless Mesh Networks", *MASS '11 Proceedings of the 2011 IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems*, Washington, DC, USA, hal.272-281.
- Kumaran, S.T. dan Sankaranarayanan, V. (2011), "Early congestion detection and adaptive routing in MANET", *Egyptian Informatics Journal*, vol. 12, no. 3, Nopember, hal. 165–175.
- Lochert, C., Scheuermann, B. dan Mauve, M. (2007), "A Survey on Congestion Control for Mobile Ad-Hoc Networks", *Wiley Wireless Communications and Mobile Computing* 7 (5), vol. 7, no. 5, Juni, hal. 655 - 676.
- Popa, L., Raiciu, C., Stoica, I. dan Rosenblum, D. (2006), "Reducing congestion effects in wireless networks by multipath routing", *Proceedings of the 14th International Conference on Network Protocols*, California, USA.
- Razzaque, M.A. dan Hong, C.S. (2009), "Congestion detection and control algorithms for multipath data forwarding in sensor networks", *Advanced Communication Technology, 2009. ICACT 2009. 11th International Conference on*, Gangwon-Do, Korea, hal.651- 653.
- Zhai, H., Chen, X. dan Fang, Y. (2005), "Rate-Based Transport Control for Mobile Ad Hoc Networks", *IEEE*, vol. 0-7803-8966-2/05.