

# PEMILIHAN NODE REBROADCAST UNTUK MENINGKATKAN KINERJA PROTOKOL MULTICAST AODV (MAODV) PADA VANETS

Mohamad Dimiyati<sup>1)</sup>, Radityo Anggoro<sup>2)</sup>, dan Waskitho Wibisono<sup>3)</sup>

<sup>1)</sup>Politeknik Negeri Banyuwangi

<sup>2,3)</sup> Jurusan Teknik Informatika, Fakultas Teknologi Informasi

Institut Teknologi Sepuluh Nopember (ITS)

Kampus ITS, Keputih-Sukolilo, Surabaya-60111

e-mail: [ayt.dimiyati@gmail.com](mailto:ayt.dimiyati@gmail.com)<sup>1)</sup>, [onggo@if.its.ac.id](mailto:onggo@if.its.ac.id)<sup>2)</sup>, [waswib@if.its.ac.id](mailto:waswib@if.its.ac.id)<sup>3)</sup>

## ABSTRAK

MAODV merupakan routing protokol yang dapat digunakan pada VANETs. Setiap paket RREQ yang diterima node pada protokol MAODV dikirim secara broadcast. Proses ini dapat mengakibatkan flooding pada jaringan yang dapat mengakibatkan congestion. Selain itu, hal ini juga mengakibatkan routing overhead pada setiap node. Jumlah node yang melakukan broadcast dapat dikurangi dengan menentukan node tertentu yang dapat melakukan rebroadcast paket RREQ. Algoritma PGB digunakan untuk mengurangi jumlah node yang melakukan broadcast paket RREQ. Pemilihan node yang melakukan broadcast paket RREQ didasarkan pada signal strength. Protokol MAODV yang menggunakan algoritma PGB pada proses route discovery bertujuan untuk mengurangi jumlah node yang melakukan rebroadcast. Network Simulator 2 (NS-2) digunakan untuk mensimulasikan penggunaan protokol MAODV-PGB dan Simulation of Urban Mobility (SUMO) digunakan untuk mensimulasikan mobilitas kendaraan. Hasil pengujian pada protokol MAODV-PGB kemudian dibandingkan dengan protokol MAODV. Pengujian dilakukan dengan beberapa skenario untuk menguji kinerja protokol MAODV-PGB. Hasil uji coba yang dilakukan menunjukkan peningkatan kinerja protokol MAODV-PGB yang ditunjukkan dengan meningkatnya nilai PDR, menurunnya average end-to-end delay, berkurangnya routing overhead dan meningkatnya goodput ratio. Oleh karena itu penggunaan protokol MAODV pada VANETs dapat mengurangi congestion yang diakibatkan oleh broadcast paket RREQ.

**Kata Kunci:** MAODV, protokol, routing, simulator

## ABSTRACT

MAODV is a routing protocol that can be used on VANETs. Every package RREQ received any node on the protocol MAODV sent broadcast. This process could lead to flooding on the network that can lead to congestion. In addition, it is routing overhead at each node, the number of nodes that perform broadcast can be reduced by specifying a particular node that can perform rebroadcast RREQ packet. PGB algorithm is used to reduce the number of nodes to broadcast RREQ packet. Selection node broadcasts RREQ packets based on signal strength. MAODV protocol that uses algorithms PGB on the route discovery process aims to reduce the number of nodes that do rebroadcast. Network Simulator 2 (NS-2) is used to simulate the MAODV-PGB protocol usage and Simulation of Urban Mobility (SUMO) is used to simulate the vehicle mobility. The results of the testing MAODV-PGB protocol compared with MAODV protocol. Testing is done with a few scenarios to test the performance of MAODV-PGB protocol. Result of experiment showed the improvement of MAODV-PGB protocol performance which indicated by the increased value of the PDR, decreasing average of end-to-end delay, decreasing of routing overhead and increasing the goodput ratio. Therefore the use of MAODV protocol on VANETs can reduce the congestion caused by the broadcast RREQ packet.

**Keywords:** MAODV, protocol, routing, simulator

## I. PENDAHULUAN

PERKEMBANGAN teknologi informasi yang begitu cepat memudahkan manusia untuk mendapatkan informasi tentang segala hal, tak terkecuali informasi yang berkaitan dengan lalu lintas. Seorang pengemudi dapat memanfaatkan *Google map* untuk menunjukkan rute dan jarak yang ditempuh untuk suatu lokasi tertentu. Pengemudi tidak perlu khawatir tentang lokasi yang belum pernah ditempuh atau dilewati. Hal ini memberikan kenyamanan dalam menempuh perjalanan.

Beberapa tahun terakhir dikembangkan teknologi yang berkaitan dengan komunikasi antar kendaraan menggunakan teknologi *wireless*. Tiap kendaraan dapat memberi atau menerima informasi yang berkaitan dengan kondisi lalu lintas. Masing-masing kendaraan yang dilengkapi perangkat *wireless* bergerak menempuh perjalanan dapat diasumsikan sebuah pergerakan node. Node yang bergerak ini, membentuk jaringan ad hoc yang disebut *Vehicular Ad-hoc Networks* (VANETs) [3]. VANETs merupakan jaringan ad hoc yang dapat melakukan konfigurasi sendiri secara terpusat pada *Medium Access Control* (MAC) dan membentuk topologi jaringan, serta mempunyai kemampuan untuk berkomunikasi dengan kecepatan tinggi. Meskipun VANETs bekerja pada jaringan *wireless ad hoc*, akan tetapi berbeda dengan MANET. Beberapa karakteristik yang membedakan antara VANETs dan

MANET adalah pergerakan dan kecepatan node. Selain itu VANETs tidak mempunyai masalah dengan media penyimpanan dan suplai energi.

Beberapa artikel menjelaskan bahwa routing protokol dalam VANETs dapat dikelompokkan menjadi lima kategori [7] yaitu: *driven protocols*, *position-based*, *cluster-based*, *broadcast* dan *geocast routing*. Beberapa *routing* protokol ad hoc reaktif telah ditawarkan untuk digunakan dalam VANETs seperti *Ad hoc On demand Distance Vector* (AODV), *Dynamic Source Routing* (DSR), dan *Algoritma Temporally Ordered Routing Algorithm* (TORA). Semua routing protokol tersebut bekerja secara *unicast*. Penggunaan routing protokol *ad hoc* secara *multicast* menunjukkan kinerja yang konsisten dengan meningkat jumlah penerima. Protokol *multicast* pada VANETs untuk menangani beberapa transmisi yang tidak ada spesifikasi lokasi geografis [2].

*Multicast* adalah sebuah transmisi data yang ditujukan ke sebuah alamat yang dibagi ke beberapa node. Meskipun *multicast* pada awalnya diperuntukan jaringan kabel, tapi dalam perkembangannya diperuntukan juga untuk jaringan *wireless* termasuk *ad hoc* [5]. *Multicast Ad-hoc On-Demand Distance Vector* (MAODV) adalah sebuah routing protokol *multicast* pada MANET. Pada MAODV, sebuah node akan mengirim paket RREQ secara *broadcast* ke semua node yang ada, ketika ingin bergabung atau meneruskan paket ke sebuah grup *multicast* yang belum diketahui rutenya. *Broadcast* paket RREQ dibatasi waktu, apabila dalam waktu yang diberikan tidak mendapatkan jawaban berupa paket RREP maka akan dilakukan *broadcast* ulang dengan menambahkan TTL (*Time To Live*) untuk menjangkau node yang lebih jauh. Akibat dari *broadcast* ini terjadi *flooding* pada jaringan. Sehingga jaringan akan terbebani yang diakibatkan oleh proses *route discovery* sebagai dampak dari pergerakan *node*.

Salah satu solusi untuk menyelesaikan permasalahan diatas adalah mengurangi jumlah node yang melakukan *broadcast* paket RREQ pada proses *route discovery*. Pengurangan ini dilakukan dengan cara menentukan atau memilih node tertentu yang dapat meneruskan *broadcast* paket RREQ. Pemilihan node yang dapat meneruskan paket *broadcast* RREQ dapat menggunakan algoritma *Preferred Group Broadcasting* (PGB) yang bertujuan mengurangi *overhead* paket dengan menghilangkan transmisi yang berlebihan [4]. Setiap node pada PGB bekerja berdasarkan tingkat kekuatan *signal sender* pada saat melakukan *broadcast* paket. Kekuatan *signal* ini digunakan untuk menentukan *hold off time* dari sebuah node untuk melakukan *rebroadcast* paket. Node dengan *hold off time* terpendek yang akan melakukan *rebroadcast* paket lebih dulu. Sedangkan node dengan *hold off time* lebih lama menunggu sampai batas waktu *hold off time* untuk melakukan *broadcast*. Proses ini akan berulang sampai node menemukan anggota dari sebuah *group multicast* atau *multicast tree* dan mengirimkan paket RREP sebagai balasan. Karena pada *multicast* hanya node yang menjadi anggota *multicast* atau *multicast tree* yang dapat mengirimkan paket RREP sebagai balasan.

Berdasarkan penjelasan diatas maka diusulkan untuk memodifikasi proses *route discovery* yang berkaitan dengan *broadcast* pada *protocol* MAODV menggunakan algoritma PGB (*Preferred Group Broadcasting*). Penggunaan algoritma PGB bertujuan untuk mengurangi jumlah *rebroadcast* yang berdampak pada pengurangan *routing overhead*. Sehingga meningkatkan jumlah paket yang diterima dan efisiensi penggunaan *bandwidth*.

## II. KAJIAN PUSTAKA

### A. *Multicast Ad-hoc On-Demand Distance Vector Protocol*

MAODV (*Multicast Ad-hoc On-Demand Distance Vector*) adalah sebuah protokol *multicast* yang dibentuk berdasarkan permintaan rute. MAODV menggunakan mekanisme *broadcast* untuk membentuk sebuah rute baru. Sebuah node mengirimkan paket *Route Request* (RREQ) ketika ingin bergabung ke grup *multicast* atau ketika node tersebut mempunyai sebuah paket yang dikirim ke grup *multicast* tetapi node tidak memiliki rute ke grup *multicast*. Setiap node yang mendengarkan paket RREQ melakukan *rebroadcast*. *Broadcast* berhenti ketika node anggota grup *multicast* atau node sebagai *multicast tree* mendengarkan *broadcast* paket RREQ. Paket *Reply Request* (RREP) pada protokol MAODV hanya dapat dikirim oleh node yang menjadi anggota grup *multicast* atau node sebagai *multicast tree*. Setelah node anggota grup *multicast* atau node *member tree* mendengarkan paket RREQ maka paket *Route Reply* (RREP) dikirim ke node sumber secara *unicast*.

Paket RREQ dikirim bertujuan untuk membangun sebuah rute. Ketika node menerima paket RREQ, maka data disimpan ke dalam *routing table* untuk mempersiapkan *reverse route*. Setiap *node* menerima paket RREQ, sebelum melakukan *rebroadcast* paket RREQ, node mengubah *sequence number* dan *next hop* untuk *node* sumber. Hal ini dilakukan untuk mempersiapkan diri jika nanti *node* merespon pengembalian atas paket RREQ. *Node* menerima paket RREQ diberi tambahan *join* disimpan dalam *multicast route table*, sedangkan node yang tidak diberi pesan tambahan merupakan node sebagai *multicast tree*. Jika paket RREQ bertanda *join* diterima oleh *node multicast tree* maka node akan mengirim paket RREP sebagai balasan, jika *node multicast tree* mempunyai *sequence number* sama besar seperti yang ada dalam paket RREQ. Jika paket RREQ bertanda *join* diterima node anggota grup *multicast* maka node penerima paket RREQ akan membalas, jika node adalah anggota grup *multicast* dan mempunyai *sequence number* lebih besar dari yang ada dalam paket RREQ. *Node* yang membalas paket RREQ mengubah *multicast route tables* pada bagian *next hop* dengan node yang mengirimkan paket RREQ yang didengar.

Selanjutnya paket RREP dikirim secara *unicast* ke *node* sumber. *Node* yang menerima paket RREP mengubah *route table* dan *multicast route table* dengan *node* yang telah mengirimkan paket RREP. Sehingga terbentuklah sebuah rute sebuah grup *multicast*.

*Node* sumber yang melakukan *broadcast* paket RREQ untuk bergabung dengan grup *multicast* akan menerima banyak balasan paket RREP dari beberapa *node*. *Node* sumber melakukan seleksi dengan memilih *sequence number* terbesar dan *hop* terpendek dari *multicast tree* dan mengabaikan rute yang lain. Selanjutnya *node* sumber mengirimkan paket *Multicast Activation* (MACT) ke rute yang dipilih secara *unicast*. Sebaliknya jika *node* sumber bukan grup *multicast* melainkan *member tree* maka *node* sumber memilih rute terbaik dengan mengirimkan paket MACT secara *unicast* ke *next hop*. Proses ini akan berlangsung sampai *node* yang mengirim RREP tercapai. Paket MACT dibuat untuk memastikan bahwa hanya satu rute *node* sumber untuk mencapai *member multicast* dan mengaktifkan *multicast route tables*. Paket data dikirim berdasarkan pada *multicast route tables*. Selama MACT belum diterima *node* pengirim paket RREP maka *multicast route tables* belum dirubah.

*Multicast* dikontrol oleh *Group Leader* (GL). GL adalah *node* yang pertama menjadi anggota *multicast*. GL bertugas menjaga *sequence number* dan melakukan *broadcast sequence number* ke semua anggota *multicast*. aktifitas tersebut dilakukan GL dengan mengirimkan sebuah paket *Hello Group*. Paket *Hello Group* berisi alamat *multicast* dan *sequence number*. Beberapa *node* menggunakan informasi *Hello Group* untuk mengubah *request table*. Jika sebuah *node* ingin bergabung dengan grup *multicast* maka paket RREQ didasarkan dari informasi *Hello Group*.

### B. Vehicular Ad-Hoc Networks

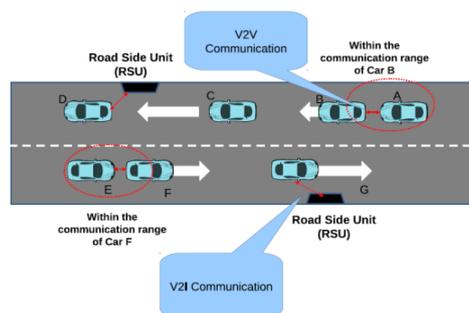
*Vehicular ad-hoc networks* (VANETs) adalah sebuah komunikasi secara langsung antara kendaraan bergerak menggunakan jaringan *Ad hoc* [8]. Setiap kendaraan yang dilengkapi dengan perangkat *wireless* dapat membentuk suatu jaringan VANETs. Kendaraan diasumsikan adalah sebuah *node* yang bergerak. Karakteristik VANETs memiliki kemiripan dengan *mobile Ad Hoc networks* (MANETs), diantaranya : tidak ada infrastruktur tetap, topologi selalu berubah, jangkauan komunikasi terbatas. Akan tetapi VANETs dan MANETs memiliki beberapa perbedaan, yaitu : pergerakan, kecepatan dan sensor On-board seperti GPRS [1].

Bentuk komunikasi pada VANETs dikategorikan menjadi dua tipe yaitu komunikasi yang dilakukan antara kendaraan dengan kendaraan yang dilengkapi dengan peralatan komunikasi disebut *Vehicle-to-Vehicle* (V2V) dan komunikasi antara kendaraan dengan infrastruktur yang diletakkan di tepi jalan disebut *Vehicle-to-Infrastructure* (V2I) seperti pada Gambar 1. Perangkat yang diletakkan di tepi sepanjang jalan untuk mendukung terjadinya komunikasi antara kendaraan disebut *Roadside Unit* (RSU). Ada tiga system utama yang mendukung VANETs selain RSU, yaitu *On Board Unit* (OBU) dan *Application Unit* (AU) [7].

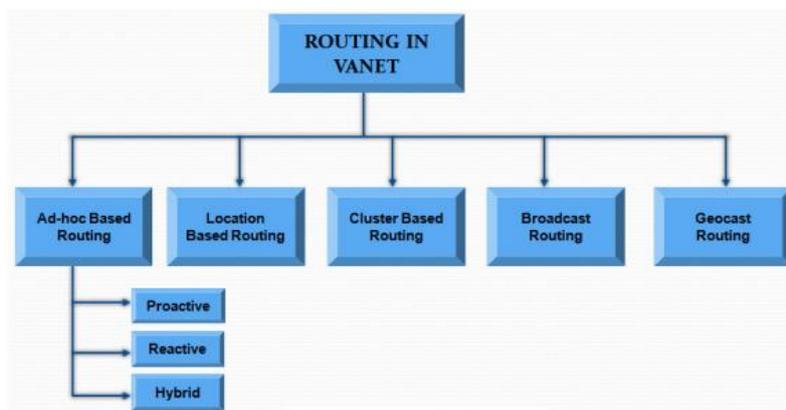
Salah satu tantangan utama dalam merancang jaringan VANETs adalah mengembangkan sebuah *routing protocols* yang dapat menyebarkan informasi antara *node* dengan topologi yang berubah secara dinamis [6]. Beberapa *routing protocols* pada MANETs telah diuji untuk diimplementasikan pada lingkungan VANETs, tetapi perlu dikembangkan lagi. Topologi pada VANETs berubah sangat dinamis. *Node* bergerak dengan kecepatan tinggi yang menyebabkan *routing protocol* pada MANETs, perlu pengembangan untuk diimplementasikan pada VANETs.

Tantangan penerapan *routing protocols* MANETs pada VANETs adalah mengurangi *delay* berkaitan dengan penyampaian informasi dari satu *node* ke *node* yang lain. Selain itu mengurangi *overhead* perlu juga dilakukan. Tantangan yang paling sulit adalah mengoptimalkan *route discovery* dan *maintenance* di lingkungan VANETs terkait dengan perubahan topologi yang sangat dinamis [6].

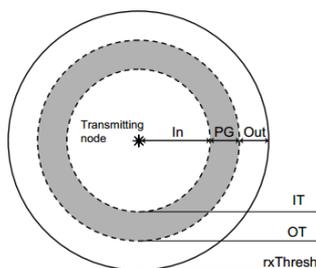
*Routing protocol* pada VANETs terkait dengan tipe arsitektur jaringan dan kinerja setiap ada perubahan dalam topologi jaringan seperti pada Gambar 2. *Routing protocols* pada VANETs diklasifikasikan ke dalam lima kategori: *driven protocols*, *position-based*, *cluster-based*, *broadcast* dan *geocast routing*) [7].



Gambar 1. Tipe Komunikasi pada VANETs [6]



Gambar 2. Type routing pada VANETs [6]



Gambar 3. Ilustrasi Signal sebagai Klasifikasi [4]

### C. Preferred Group Broadcasting

Algoritma *Preferred Group Broadcasting* (PGB) adalah sebuah mekanisme *broadcast* untuk menurunkan *overhead* pesan dengan mengurangi transmisi yang berulang dan untuk mendapatkan rute yang stabil dengan kemampuan *auto-correct* [4]. Salah satu beban *routing* pada jaringan *wireless ad hoc* adalah menentukan rute dan menjamin kestabilan rute, karena node pada jaringan ini terus bergerak. Jumlah atau kepadatan node yang rendah menandakan jarak antar node panjang. Sehingga apabila satu node keluar dari jangkauan transmisi mengakibatkan rute yang sudah terbentuk menjadi rusak. Perbaikan rute memerlukan *broadcast* ke jaringan yang mengakibatkan *congestion*.

*Broadcast* pada saat *route discovery* tidak memiliki kriteria khusus untuk menentukan node *intermediate* yang meneruskan *broadcast* paket RREQ. Setiap node yang menerima *broadcast* paket RREQ akan melakukan *broadcast* ulang sampai sebuah node mengirimkan pesan RREP. PGB memberikan kriteria pada node yang melakukan *broadcast* ulang ketika menerima sebuah pesan RREQ. Meskipun beberapa node menerima pesan RREQ akan tetapi hanya node tertentu yang dapat melakukan *broadcast* ulang pesan RREQ. Hal ini akan mengurangi *overhead* jaringan dalam melakukan *broadcast* pesan RREQ. Sehingga *congestion* yang diakibatkan *broadcast storms* dapat dikurangi.

Algoritma PGB bekerja berdasarkan kekuatan *signal* tetangga. Sebuah node pada saat mentransmisikan sebuah paket maka node yang berada sekitarnya akan mendengarkan dan merasakan kekuatan *signal* yang dipancarkan dari node tersebut. Berdasarkan kekuatan *signal* ini, node melakukan perhitungan sebelum meneruskan sebuah paket.

Setiap node dalam PGB akan melakukan klasifikasi setiap mendengarkan sebuah node melakukan *broadcast* paket RREQ. Klasifikasi didasarkan pada kekuatan *signal node* yang melakukan *broadcast* seperti yang terlihat pada Gambar 3. Setiap node yang mendengarkan *broadcast* pesan RREQ akan menangkap kekuatan *signal* dari node yang melakukan *broadcast*. Kekuatan *signal* yang diterima dibandingkan dengan dua nilai yaitu *Inner Threshold* (*IT*) dan *Outer Threshold* (*OT*) [4]. Kedua ambang batas tersebut dikaitkan dengan ambang batas penerima (*rxTh*) dan kekuatan *signal* yang sesuai dengan jangkauan transmisi maksimum. Ambang batas penerima *rxTh* dalam satuan watt. Nilai *IT* dan *OT* dapat dihitung dengan persamaan berikut:

$$IT = rxTh + f_{IT} \tag{1}$$

$$OT = rxTh + f_{OT} \tag{2}$$

*IT* dan *OT* merupakan kekuatan *signal* jangkauan maksimal pada kelompoknya dalam watt, seperti terlihat pada Gambar 3. Nilai  $f_{IT}$  dan  $f_{OT}$  merupakan bilangan integer positif dengan satuan dB, yang ditentukan awal saat konfigurasi. Persamaan diatas dapat digunakan untuk melakukan klasifikasi dari sebuah node sebagai berikut:

1. PG :  $OT < rxP < IT$  sebagai grup 1
2. OUT :  $rxP < OT$  sebagai grup 2
3. IN :  $IT \leq rxP$  sebagai grup 3

Node dengan klasifikasi sebagai PG merupakan prioritas utama. Apabila tidak ada node dengan klasifikasi PG maka OUT sebagai pilihan ke dua dan IN sebagai pilihan ketiga. Pada ketiga kelompok tersebut, node yang mendekati nilai  $OT$  merupakan prioritas utama di masing-masing grup.

Setelah masing-masing node melakukan klasifikasi dirinya maka node menunggu untuk periode *hold off* sebelum mengambil keputusan untuk melakukan *rebroadcast* paket atau membuang paket. *Hold off* dan strategi untuk membuat keputusan *rebroadcast* tergantung grup masing-masing node. *Hold off time* (*hoTime*) masing-masing node tergantung dari grup mana node berada. Perhitungan *hoTime* dari masing node sesuai dengan fomulasi berikut:

$$hoTime = T_i + jitterT_i \text{ dengan } jitterT_i \in [0, dt_i) \tag{3}$$

$$\Delta P_i = |rxTh - rxP| - f_i \tag{4}$$

$$T_i = \Delta P_i \cdot dt_i + minT_i \tag{5}$$

$f_i$  adalah nilai  $f_{IT}$  atau  $f_{OT}$  tergantung dari grup node tersebut. Dimana untuk  $f_{1,2} = f_{OT}$  dan  $f_3 = f_{IT}$ .  $dt_i$  adalah nilai maksimum *jitter* untuk grup  $i$ . *Jitter* digunakan untuk mencegah *rebroadcast* secara bersamaan dari beberapa node yang memiliki nilai  $rxP$  sama [4].

Perhitungan nilai *jitter* untuk masing-masing grup adalah sebagai berikut:

- a. Preferred group

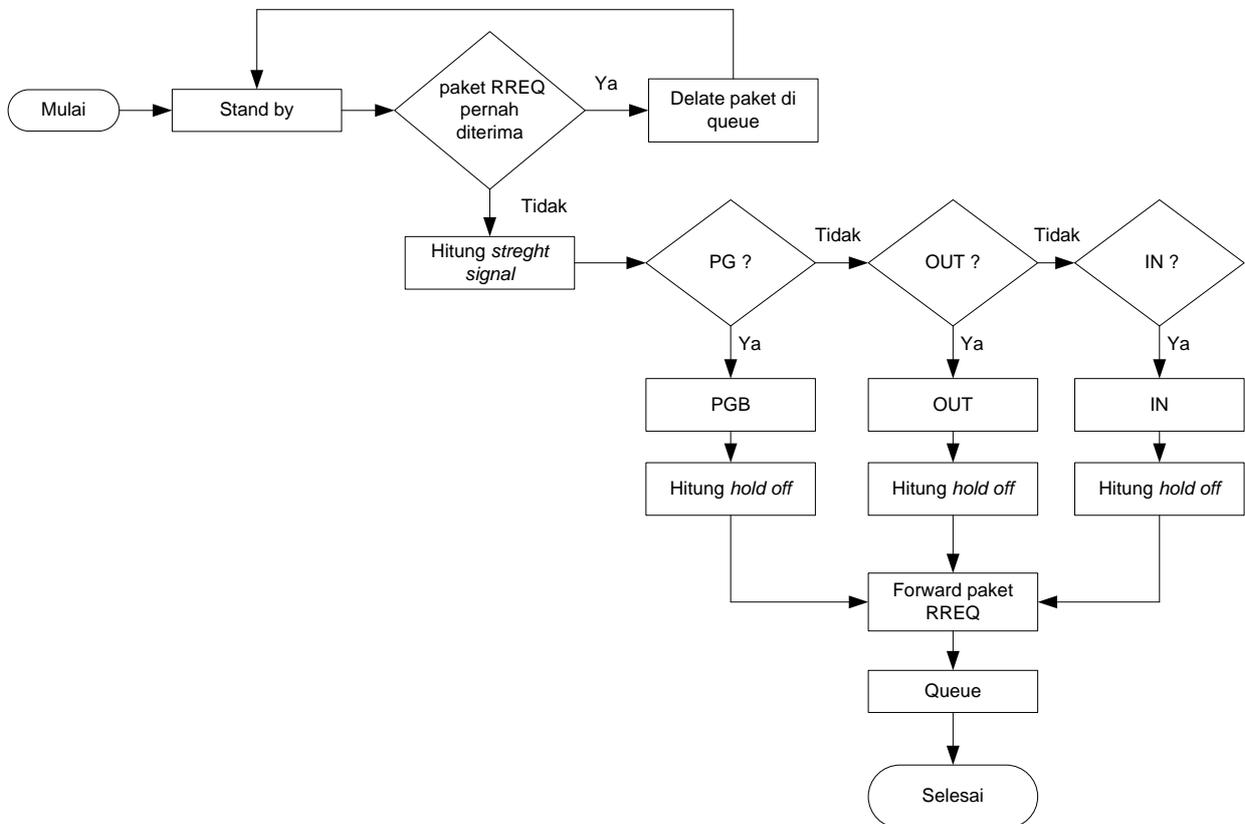
$$dt_1 = \frac{maxT_1 - mixT_1}{f_{IT} - f_{OT}} \tag{6}$$

- b. Out Group

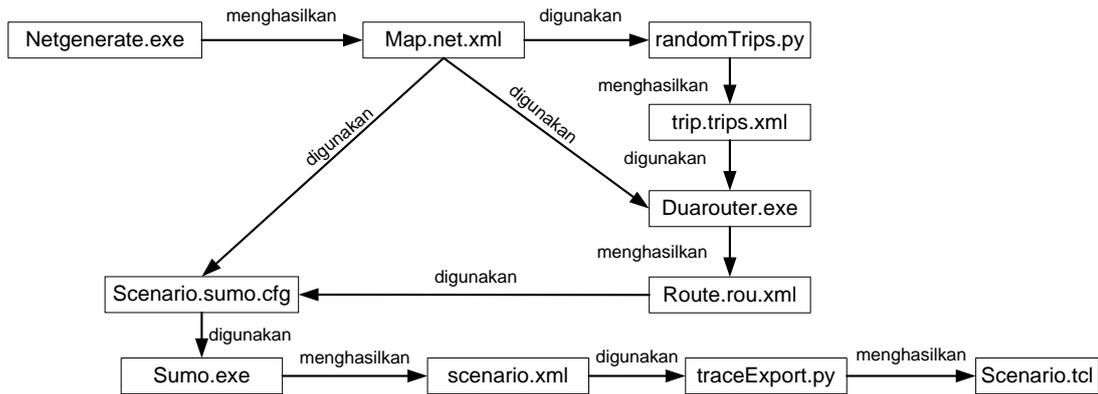
$$dt_2 = \frac{maxT_2 - mixT_2}{f_{IT}} \tag{7}$$

- c. In group

$$dt_3 = \frac{maxT_3 - mixT_3}{(txP - rxTh) - f_{IT}} \tag{8}$$



Gambar 4. Diagram alir mekanisme pemilihan node rebroadcast RREQ



Gambar 5. Alur pembuatan peta grid

TABEL I  
PARAMETER MOBILITAS KENDARAAN

Parameter	Spesifikasi
Jumlah grid x	5 buah
Jumlah grid y	5 buah
Jarak grid x	200 m
Jarak grid y	200 m
Kecepatan	10m/s, 20m/s, 30m/s
Waktu simulasi	1000 detik
Jumlah kendaraan	50 buah
Jarak antar kendaraan	2.5 m (default)
Panjang kendaraan	5 m (default)

### III. METODOLOGI

#### A. Perumusan masalah

Dalam pembentukan rute baru MAODV melakukan broadcast paket RREQ. Setiap node yang menerima RREQ melakukan *rebroadcast*. Proses ini berdampak pada *broadcast storm*, secara konvensional MAODV membatasi dengan memberikan TTL (*Time To Live*) setiap melakukan *broadcast*. Meskipun teknik ini masih dapat mengakibatkan *broadcast storm*.

Pembatasan jumlah node yang melakukan *broadcast* pada protokol MAODV merupakan permasalahan yang diangkat dalam penelitian ini. Dengan membatasi node yang melakukan *broadcast* dapat mengurangi *broadcast storm*. Sehingga *congestion* yang diakibatkan oleh *rebroadcast* paket RREQ dapat diminimalkan. Selain itu kerja *routing* protokol akan lebih ringan yang dapat mengurangi *routing overhead*.

#### B. Perancangan metode

Pada protokol MAODV rute dibentuk berdasarkan permintaan. Ketika sebuah node mempunyai paket yang ditujukan ke MAODV atau sebuah node ingin bergabung menjadi anggota MAODV. Rute baru dibuat dengan cara melakukan *broadcast* paket RREQ. Setiap node yang menerima paket RREQ melakukan *rebroadcast* paket tersebut. Node yang melakukan *rebroadcast* paket RREQ dipilih berdasarkan kekuatan signal node terhadap sender. Diagram alur tentang mekanisme pemilihan node *rebroadcast* paket RREQ menggunakan protokol MAODV-PGB ditunjukkan pada Gambar 4.

Algoritma PGB digunakan untuk memilih node yang melakukan *rebroadcast* paket RREQ. Setiap node yang menerima paket RREQ menangkap *signal strength* dari node *sender*. Nilai *signal strength* ( $rxP$ ) kemudian dibandingkan dengan nilai  $f_{IT}$  dan  $f_{OT}$  untuk menentukan posisi node. Nilai  $f_{IT}$  dan  $f_{OT}$  ditentukan 12 dB dan 6 dB [4].  $f_{IT}$  dan  $f_{OT}$  dikonversikan ke dalam watt untuk mendapatkan nilai  $f_{IT}$  dan  $f_{OT}$ .

Setelah mengetahui keberadaan node terhadap *sender*, node melakukan perhitungan *hoTime*. Nilai  $\Delta T=10$ ms,  $maxT_1=15$  ms,  $maxT_2=40$  ms dan  $maxT_3=100$  ms [4]. Perhitungan *hoTime* dapat menggunakan persamaan 2.3 sampai 2.8. Nilai  $rxTh$  berdasarkan transmisi 250 m maka nilai  $rxTh=3.65e-10$  watt.

Setiap node yang menerima paket RREQ melakukan broadcast paket RREQ, akan tetapi ketika menggunakan PGB setiap node mempunyai waktu berbeda-beda untuk broadcast paket RREQ. Perbedaan waktu broadcast ditentukan oleh *hoTime*. Node dengan waktu *hoTime* lebih pendek melakukan lebih dulu. Pada algoritma PGB waktu terpendek diperoleh oleh grup PG, kedua grup OUT dan terakhir IN.

Ketika node menerima paket RREQ dilakukan pengecekan terhadap paket tersebut. Apabila paket RREQ belum pernah diterima maka node melakukan perhitungan waktu *hoTime* seperti diatas. Sebaliknya, ketika node sudah pernah menerima paket RREQ maka paket RREQ yang berada dalam antrian (*queue*) untuk dikirim akan dihapus. Node tidak perlu melakukan broadcast paket RREQ, karena paket tersebut sudah di broadcast oleh node lain yang berada disekitar. Sehingga *routing overhead* pada dapat dikurangi. Selain itu jumlah paket RREQ terkirim menjadi berkurang.

#### IV. MODEL MOBILITAS KENDARAAN

Skenario mobilitas kendaraan pada VANETs dibagi dalam 3 tahapan yaitu perancangan peta pergerakan kendaraan, pembuatan rute pergerakan dan lalu lintas kendaraan, dan implementasi pergerakan. Dalam penelitian ini skenario mobilitas node menggunakan sebuah simulator SUMO [10] untuk membangun sebuah peta grid. SUMO sebagai simulator mobilitas kendaraan menyerupai situasi nyata sebuah lalu lintas kendaraan. Sistem lalu lintas yang mampu digambarkan oleh SUMO meliputi kecepatan, tikungan, pengaturan *traffic lights*, kepadatan kendaraan dan jarak. Alur pembuatan skenario mobilitas kendaraan menggunakan SUMO dalam bentuk peta grid dapat dilihat pada Gambar 5.

Peta yang digunakan dalam penelitian ini dalam bentuk grid dengan parameter yang digunakan sesuai dengan Tabel I. Jumlah peta yang dihasilkan sebanyak 3 yaitu peta dengan kecepatan 10 m/s, 20 m/s dan 30 m/s. Mobilitas kendaraan yang dihasilkan SUMO sebuah file *tcl* yang digunakan untuk pergerakan node pada NS-2 [9].

#### V. SIMULASI

Dalam penelitian ini, perangkat lunak simulasi jaringan digunakan NS-2 [9] yang banyak digunakan untuk penelitian. NS-2 merupakan perangkat lunak *open source* yang dikembangkan dalam dua bahasa yaitu *otcl* dan *C++*. Hasil simulasi berupa file trace yang dapat dianalisis dengan program skrip AWK. Hasil analisis dapat digunakan untuk mengevaluasi kinerja dari protokol MAODV-PGB yang dibandingkan dengan protokol MAODV. Berikut merupakan beberapa indikator yang digunakan untuk mengevaluasi kinerja protokol yang diusulkan:

##### 1) Packet Delivery Ratio (PDR)

*Packet Delivery Ratio* (PDR) adalah perbandingan antara total paket data yang berhasil diterima dan jumlah total paket data yang dihasilkan oleh sumber. PDR menunjukkan tingkat keberhasilan sebuah protokol melakukan pengiriman paket dari sumber ke tujuan. Nilai PDR semakin besar memberikan tingkat keberhasilan yang baik, sebagai tanda bahwa *routing protocol* lebih lengkap dan benar. Penghitungan PDR untuk multicast adalah sebagai berikut:

$$PDR = \left( \frac{\sum CBR_{delivered}}{(CBR_{sent} \times CBR_{delivered})} \right) \times 100\% \quad (9)$$

##### 2) Average end to end delay

*Delay* adalah waktu yang dibutuhkan untuk mengirimkan paket data, terhitung dari saat dikirim sampai saat data diterima. *Average end to end delay* adalah waktu rata-rata yang dibutuhkan untuk mentransfer data dari sumber ke tujuan. Beberapa hal yang menyebabkan *delay* adalah *buffering* selama proses *route discovery*, antrian (*queue*), transmisi ulang, propagasi dan waktu transfer.

Perhitungan *average end to end delay* didapat dari selisih waktu antara setiap paket CBR (*Constant Bit Rate*) terkirim dan diterima, dibagi total waktu paket CBR diterima. Nilai *average end to end delay* semakin rendah memberikan kinerja yang lebih baik.

$$Average\ End - to - End\ Delay = \frac{\sum(CBR_{rcvTime} - CBR_{sentTime})}{\sum_{receiver}} \quad (10)$$

##### 3) Routing Overhead (RO)

*Routing Overhead* adalah jumlah paket kontrol *routing* yang ditransmisikan per data paket yang terkirim ke tujuan selama pengambilan data. RO dihitung berdasarkan paket routing yang ditransmisikan. Baris yang mengandung *routing over head* pada trace file ditandai dengan paket yang bertipe *s* (*send*) atau *f* (*forward*) dan terdapat header paket dari protokol MAODV.

$$RO = \sum Packet_{send\ MAODV} + \sum Packet_{forward\ MAODV} \quad (11)$$

##### 4) Goodput ratio

*Goodput* bisa diartikan sebagai bagian dari *throughput* yang yang terdiri dari *traffic* yang berguna. *Goodput* tidak menyertakan duplikat paket, paket yang dibuang pada *downstream* dan paket terfragmentasi. Pada jaringan

*nirkabel*, *goodput ratio* dapat didefinisikan sebagai data yang berguna dikirimkan ke pengguna sebagai sebagian kecil dari jumlah total data yang dikirimkan pada jaringan.

$$\text{Goodput Ratio} = \frac{\frac{\sum CBR_{recv}}{Recv_{multicast}}}{T_{simulation} - T_{CBR-start}} \quad (12)$$

## VI. ANALISA HASIL SIMULASI

Analisa dilakukan menggunakan skrip AWK untuk menelusuri file trace yang dihasilkan oleh NS-2. Analisa meliputi PDR, RO, *average end-to-end delay* dan *goodput*. Simulasi dilakukan sebanyak lima belas kali untuk masing-masing kecepatan. Hasil simulasi dirata-rata yang disajikan dalam bentuk grafik.

Berdasarkan Gambar 6, prosentase rata-rata PDR menunjukkan peningkatan pada masing-masing kecepatan, ketika menggunakan protokol MAODV-PGB. Prosentase peningkatan sekitar 4% dengan menggunakan MAODV. Pada kedua protokol mempunyai pola yang sama pada titik stabil dan penurunan. Kondisi stabil dihasilkan pada kecepatan 10 m/s dan 20 m/s, sedangkan penurunan tajam terjadi pada kecepatan 30 m/s. Kecepatan pada kendaraan berpengaruh pada kestabilan topologi jaringan. Sehingga berpengaruh terhadap jumlah paket yang dapat diterima.

*Average end-to-end delay* merupakan waktu tunda yang dihitung mulai waktu paket dikirim oleh node sumber sampai paket diterima node tujuan. Hasil simulasi menunjukkan penurunan yang tajam pada kecepatan 20 m/s untuk delay protokol MAODV seperti terlihat pada Gambar 7. Sedangkan pada kecepatan 10 m/s dan 30 m/s mempunyai waktu tunda yang lebih lama. Pada protokol MAODV-PGB mempunyai delay yang lebih lama pada kecepatan 30 m/s. Pada kecepatan 30m/s perubahan topologi jaringan lebih sering terjadi, dapat meningkatkan *routing overhead* yang mengakibatkan delay yang lebih lama.

Berdasarkan Gambar 8, Rata-rata RO pada kedua protokol MAODV-PGB dan MAODV mempunyai pola yang sama. Dimana kecepatan bertambah mengakibatkan jumlah rata-rata RO semakin meningkat. Kecepatan yang tinggi mengakibatkan perubahan topologi jaringan lebih sering terjadi, topologi jaringan tidak stabil. Perubahan *routing table* diperlukan proses *route discovery* dan *maintenance*. Aktivitas ini cukup membebani RO, yang mengakibatkan RO meningkat pada kecepatan yang tinggi. Meskipun demikian MAODV-PGB dapat menurunkan RO. Prosentase penurunan sekitar 35% dari jumlah ketika menggunakan MAODV.

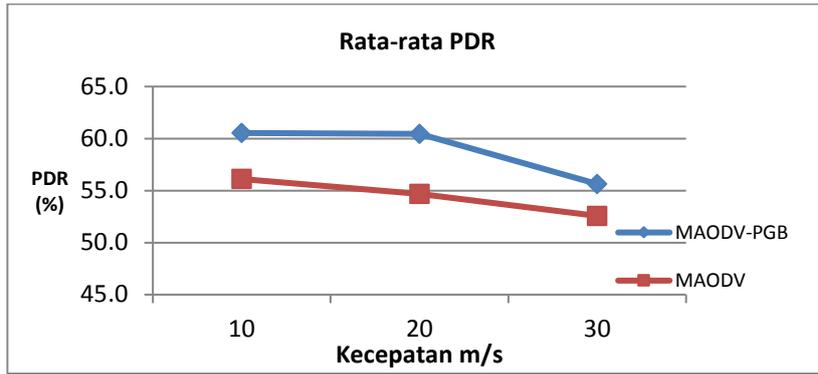
Protokol MAODV-PGB memberikan rasio *goodput* lebih baik pada semua kecepatan yang digunakan simulasi seperti terlihat pada Gambar 9. Pada kecepatan 30 m/s rasio *goodput* menurun, dan disebabkan oleh kestabilan topologi jaringan yang mempengaruhi nilai rasio *goodput*. Kejadian ini tidak hanya pada protokol MAODV-PGB tapi juga MAODV.

## VII. KESIMPULAN

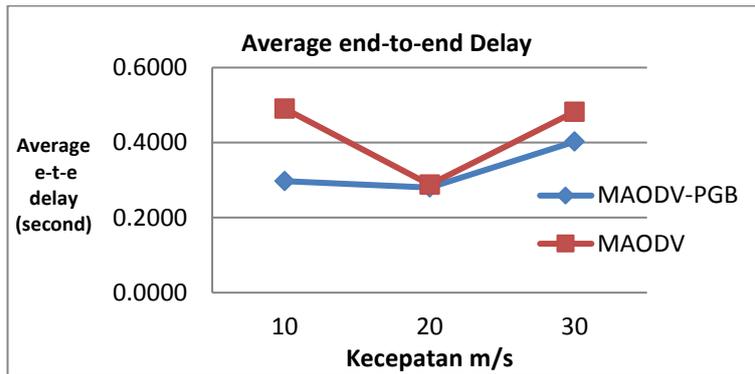
Algoritma PGB yang bekerja berdasarkan *signal strength* dapat mengurangi jumlah node yang melakukan *rebroadcast* paket RREQ pada protokol MAODV. Sehingga tidak semua node yang mendengarkan paket RREQ melakukan *rebroadcast*. Berkurangnya jumlah *rebroadcast* paket RREQ dapat menurunkan *routing overhead* (RO) pada protokol MAODV hingga 64% sampai 69% yang berdampak pada menurunnya *congestion*. Hal ini dibuktikan dengan meningkatnya PDR hingga 5.8%, berkurangnya *average end-to-end delay* hingga 193 ms dan meningkatnya *goodput ratio* hingga 26 byte/detik. Perhitungan waktu *hold off* yang berdampak pada penundaan *rebroadcast* paket RREQ masih cukup besar. Oleh karena itu, pada penelitian berikutnya perlu dicari formulasi perhitungan yang lebih singkat yang berkaitan dengan waktu *hold off*.

## DAFTAR PUSTAKA

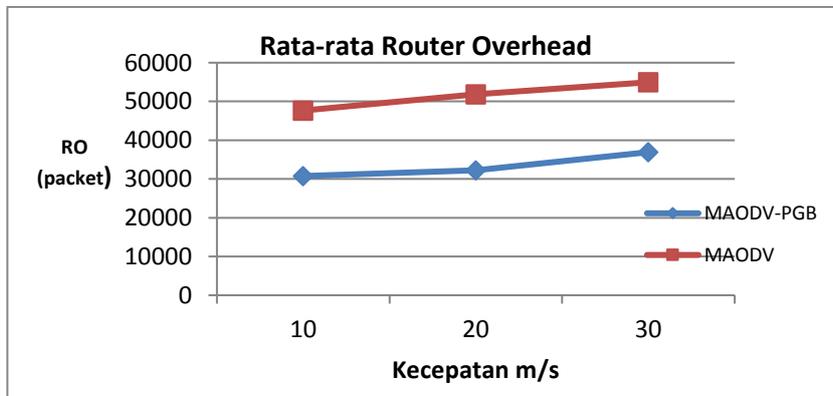
- [1] Ding, B., Chen, Z., Yan W, Yan. dan Yu,H. (2011), "An Improved AODV Routing Protocol for VANETs", *IEEE*, Vol. 978-1-4577-1010-0/11.
- [2] Hassan, A. ,Mohamed H. Ahmed, M.H dan Rahman,M.A (2011) "Performance Evaluation For Multicast Transmissions In Vanet ", *IEEE CCECE*, Vol. 978-1-4244-9789-8/11, hal 1106-1108.
- [3] Muñoz, A.G.(2009), "Multicast over Vehicle Ad Hoc Networks"
- [4] Naumov, V., Baumann, R. dan Gross.,T (2006), "An Evaluation of Inter-Vehicle Ad Hoc Networks Based on Realistic Vehicular Traces", *MobiHoc'06*, Vol. 1-59593-368-9/06/0005.
- [5] Kunz,T. dan Cheng,E.(2001), "Multicasting in Ad-Hoc Networks: Comparing MAODV and ODMRP" *Proceedings of the Workshop on Ad hoc Communications*
- [6] Sabih ur Rehman,S. Khan,M.A. dan A. Zia,A. dan Zheng, L.(2013), " Vehicular Ad-Hoc Networks (VANETs) - An Overview and Challenge" , *Journal of Wireless Networking and Communications 2013*, Vol.3,No.3, Hal. 29-38.
- [7] Sangari, M.S. dan Baskaran, K. (2014) , "A Comprehensive Survey on Efficient Routing Protocols And Simulation Tools For VANET", *International Journal of Computer Science and Information Technologies* , Vol. 5 (3) , hal. 2729-2737.
- [8] Yousefi,S., Miousavi,S.M dab Fathy,M.(2006), "Vehicular Ad Hoc Networks (VANETs):Challenges and perspectives", *IEEE*, Vol. 0-7803-9586-7/06, Hal.761-766.
- [9] NS-2. Tersedia: <http://www.isi.edu/nsnam/ns/>.
- [10] SUMO. Tersedia: <http://sumo.dlr.de/wiki/SUMO>



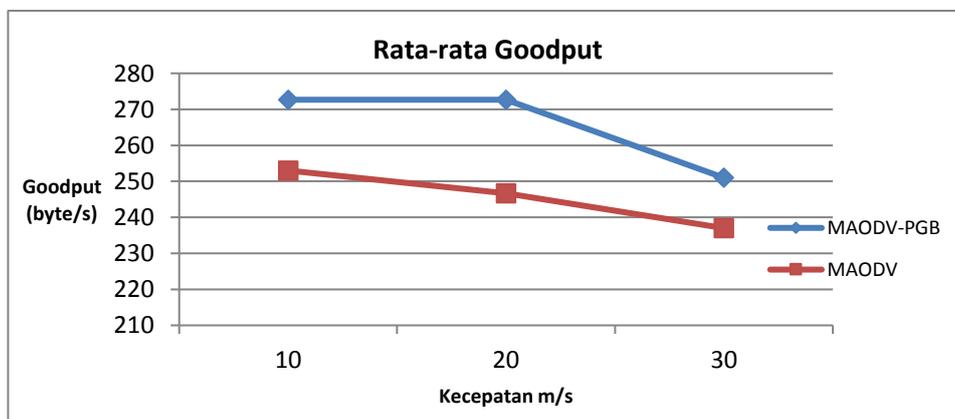
Gambar 6. Rata-rata PDR



Gambar 7. Average end-to-end delay



Gambar 8. Average end-to-end delay



Gambar 9. Rata-rata Goodput