

Analisis QoS Jaringan Adhoc Menggunakan TORA (Temporally Ordered Routing Algorithm)

Mukminatun Ardaisi*

Fakultas Teknik, Program Studi Elektro, Universitas Tridianti, Palembang, Indonesia

*mukminatunardaisi@univ-tridianti.ac.id

ABSTRACT

An ad hoc network is a collection of wireless mobile host forming a temporary network without the aid of any established infrastructure or centralized administration that has dynamic topologies characteristics. Every node in network can be act as a host and router forwarding the packet to the other node. That's why needed routing protocol. Ad hoc network Qos (Quality of service) is affected by routing protocol works. With dynamic topology as the basic of this network made a problem at routing. This happened because the topology easily change with higher mobility will cause more computation at every node for storage, memory consume, CPU processing, and power consume. TORA (Temporally Ordered Routing Algorithm) is one of ad hoc routing protocol which has multiple route so the data packet can be send through the different path from the source to the destination. It has good performance when higher mobility and the network become large. This research simulates and analyze the effect of TORA to route the data packet with performance parameter throughput, delay, packet delivery ratio and route overhead. The result show that Throughput and Packet Delivery Ratio (PDR) increase when the mount of node bigger when using TORA protocol. The biggest throughput is got by CBR traffic rather than Exponential and Pareto traffic when speed become high (5 m/s, 20 m/s). The biggest Packet Delivery Ratio (PDR) got by Exponential traffic rather than two other types.

Keywords: TORA, QoS, Adhoc, Routing, Wireless

ABSTRAK

Jaringan ad hoc adalah kumpulan dari beberapa mobile host yang membentuk suatu jaringan yang bersifat sementara tanpa ada infrastruktur dan administrasi terpusat dengan karakteristik topologi yang dinamis. Setiap node dalam jaringan dapat berperan sebagai host dan router dan harus dapat memforward paket menuju node lainnya. Untuk tujuan seperti itu, suatu protokol routing diperlukan. Quality of service (QoS) pada jaringan ad hoc dipengaruhi oleh kinerja protokol routing. Dengan topologi dinamis sebagai karakteristik dasar pada jaringan ini menimbulkan masalah dalam hal routing. Hal ini disebabkan karena pada topologi yang mudah berubah atau pada jaringan dengan mobilitas tinggi akan menyebabkan komputasi yang besar pada setiap node dalam hal penyimpanan (storage), konsumsi memori, pemrosesan data oleh CPU, dan konsumsi daya. TORA (Temporally Ordered Routing Algorithm) merupakan salah satu jenis protocol routing pada jaringan ad hoc yang memiliki multiple route dimana pengiriman data dilakukan melalui rute yang berbeda dari pengirim ke penerima. Protokol ini memiliki performansi yang baik ketika mobilitas tinggi dan jaringan bertambah besar. Pada penelitian ini akan disimulasikan dan dianalisa efek dari penggunaan protokol TORA dalam merutekan paket – paket data dengan parameter performansi throughput, delay, packet delivery ratio dan routing overhead. Pada penelitian, didapat bahwa Throughput dan Packet Delivery Ratio (PDR) menggunakan protocol routing TORA semakin meningkat ketika jumlah node bertambah besar. Throughput terbesar didapat oleh trafik CBR dibandingkan trafik Pareto dan Eksponensial. Walaupun begitu delay trafik CBR lebih besar daripada trafik Eksponensial dan Pareto terutama pada saat kecepatan tinggi (5 m/s, 20 m/s). Sedangkan untuk PDR terbesar didapat oleh trafik Eksponensial dibandingkan kedua trafik lain.

Kata Kunci : TORA, QoS, Ad-hoc, Routing, Nirkabel

INFORMASI ARTIKEL

Submit
20, September 2025

Diterima
20, Oktober 2025

Publish Online
30, November 2025

PENDAHULUAN

Jaringan Ad hoc merupakan jenis jaringan wireless tanpa infrastruktur. Pada jaringan ini tidak memiliki fixed router dan semua node atau mobile host dalam jaringan ini dapat bergerak dengan bebas serta dapat dikoneksikan secara dinamis pada setiap keadaan[1][2]. Sehingga dapat dikatakan dalam jaringan ini memiliki topologi yang berubah-ubah sesuai dengan pergerakan node didalam jaringan tersebut. Jaringan Ad Hoc pada awalnya diperuntukan untuk kepentingan aplikasi militer yaitu komunikasi pada wilayah peperangan dan untuk daerah yang sedang terjadi bencana. Peningkatan jumlah pengguna menuntut performansi QoS yang lebih baik lagi dalam mengatasi routing yang semakin kompleks. Tuntutan ini menjadi sulit sehubungan dengan tidak terkoneksi secara tetapnya (*fixed* atau *wire*) jaringan tersebut.

Quality of service (QoS) pada jaringan ad hoc dipengaruhi oleh kinerja protokol routing. Dengan topologi dinamis sebagai karakteristik dasar pada jaringan ini menimbulkan masalah dalam hal routing[3]. Hal ini disebabkan karena pada topologi yang mudah berubah atau pada jaringan dengan mobilitas tinggi akan menyebabkan komputasi yang besar pada setiap node dalam hal ukuran memori, power processing, dan konsumsi power[4]. Oleh karena itu, pengembangan protokol routing yang efisien dan handal sangat penting untuk meningkatkan kualitas layanan dalam jaringan Ad hoc. Beberapa penelitian telah dilakukan untuk mengatasi tantangan ini, termasuk pengembangan algoritma routing yang adaptif dan toleran terhadap perubahan topologi.

TORA (Temporally Ordered Routing Algorithm) merupakan salah satu jenis protokol routing pada jaringan ad hoc yang mendukung skema multipath routing dengan memiliki multiple route dimana pengiriman data dilakukan melalui rute yang berbeda dari node pengirim ke penerima[5][6]. Berdasarkan latar belakang tersebut, maka permasalahan yang diangkat dalam penelitian ini dapat dirumuskan diantaranya, (1) bagaimana kinerja protokol TORA sebagai protokol routing jaringan Ad-Hoc, (2) bagaimana performansi TORA (Temporally Ordered Routing Algorithm) dalam meningkatkan QoS jaringan Ad-Hoc yang terdiri dari: *throughput*, *packet delivery ratio (PDR)*, *end-to-end delay*, dan *routing overhead*.

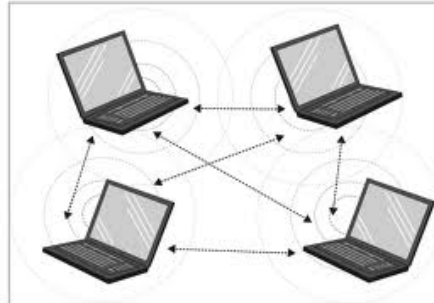
Tujuan umum dari penelitian ini adalah untuk memahami konsep dasar jaringan Ad Hoc, mengetahui konsep routing pada jaringan Ad Hoc menggunakan *protocol routing TORA (Temporally Ordered Routing Algorithm)*. Secara khusus penelitian ini bertujuan untuk mengetahui performansi *TORA* dalam pengiriman paket data pada Jaringan Ad Hoc dengan parameter - parameter *QoS* seperti *throughput*, *PDR (Packet Delivery Ratio)*, *end-to-end delay*, dan *routing overhead*. Pada penelitian ini akan di simulasikan dan di analisa efek dari penggunaan *protocol TORA* dalam merutekan paket-paket data pada WLAN mode Ad Hoc dengan standar 802.11. Dengan keluaran dari pemodelan simulasi ini berupa *throughput*, *Packet Delivery Ratio (PDR)*, *end-to-end delay*, serta *routing overhead* yang ada di diharapkan mampu meningkatkan *Qos (Quality of Service)* pada jaringan Ad Hoc.

TINJAUAN PUSTAKA

2.1 Jaringan Ad-hoc

Konfigurasi jaringan ad hoc terdiri dari beberapa mobile node yang dapat saling berkomunikasi secara peer-to-peer tanpa menggunakan infrastruktur seperti access point maupun base station[7][1]. Setiap mobile node memiliki wireless network interface dan

saling berkomunikasi dengan memanfaatkan media radio atau infra merah. Node-node pada konfigurasi jaringan ad hoc dapat bergerak dengan bebas atau diam pada posisinya.



Gambar 1. Konfigurasi Jaringan Adhoc

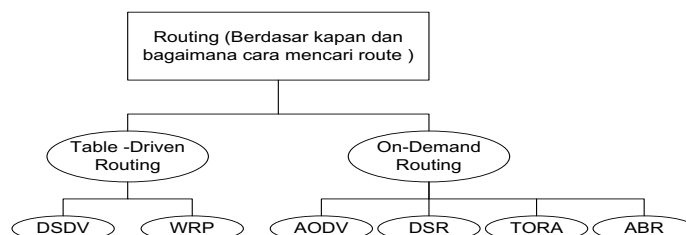
2.2 Quality of Service (QoS)

Kemampuan menyediakan jaminan performansi dan diferensiasi layanan dalam network sering diacu dengan istilah QoS (quality of service). ITU, dalam rekomendasi E.800, mendefinisikan QoS sebagai pengaruh kolektif atas performansi layanan yang menentukan tingkat kepuasan pemakai layanan. QoS dalam jaringan ad hoc dapat dilihat dalam beberapa bagian level yang saling tergantung yaitu:

- Pada level medium access protocols (MAC), dengan menambahkan fungsional dari QoS ke layer MAC dengan maksud agar lebih memberikan jaminan kualitas.
- Pada level routing protocol, dengan melihat performa route.
- Pada level signaling dengan mekanisme reservasi sumber daya yang bebas-routing protocol. QoS pada level ini bertanggungjawab atas koordinasi antara level QoS level yang lain sebaik komponen yang lainnya.

2.3 Routing

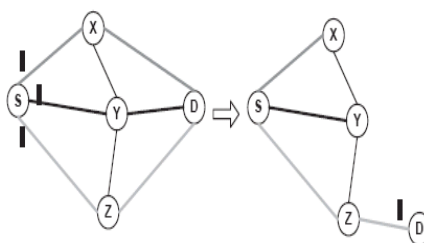
Routing adalah suatu fungsi dari lapisan network dimana fungsi ini merupakan suatu mekanisme untuk menentukan path dari suatu node (*source node*) menuju node lain (*destination node*) [8]. Protokol routing diperlukan karena untuk mengirimkan data menuju node tujuan harus melewati beberapa node *intermediate (multihops)* dan protokol *routing* ini berfungsi untuk memilih *route* terbaik untuk menyampaikan data menuju node tujuannya. Pada jaringan ad hoc, setiap node dapat bergerak dan menyebabkan topologi berubah dari waktu ke waktu. Hal ini akan menimbulkan permasalahan dalam pengiriman data dari node 2 pengirim ke node penerima. Permasalahan ini dapat diatasi dengan penggunaan protokol routing, yang dibagi berdasarkan kapan dan bagaimana *route* menemukan tujuannya. Protokol *routing* itu adalah : *table-driven* dan *on demand routing*.



Gambar 2. Kategori Protokol Routing pada Jaringan Adhoc

2.4 Multipath Routing dengan TORA (*Temporally Ordered Routing Algorithm*)

Skema *multipath routing* memberikan beberapa *path* dari sumber ke tujuan sehingga volume trafik dapat terbagi dan berkomunikasi melalui *path* yang telah dipilih[9][10]. Hal ini dapat meningkatkan performansi jaringan dilihat dari keseimbangan distribusi paket dan dapat menurunkan tingkat kongesti serta *end-to-end delay*. contoh dapat dilihat pada gambar 3. Node S mengirim paket yang sama pada tiga *path* yaitu SXD, SYD, SZD. Ketika node D bergerak, rute SXD dan SYD pun hilang (tak terhubung) tetapi rute SZD masih mampu mengirim paket ke node D. Selama *path* tersebut masih terhubung, maka node D akan tetap menerima paket. Algoritma yang digunakan pada *routing* bergantung pada protokol yang digunakan. TORA (*Temporally Ordered Routing Algorithm*) mendukung skema *multipath routing* dikarenakan protokol ini menyediakan *multiple route* dari sumber ke tujuan[11].



Gambar 3. Multipath Routing

2.5 TORA (*Temporally Ordered Routing Algorithm*)

Seperti telah dijelaskan di atas, TORA termasuk jenis protocol routing *on-demand* dimana proses *routing* akan dibentuk ketika suatu node akan berhubungan dengan node lain atau mengirim paket data. Protokol ini didesain untuk mengurangi terjadinya reaksi terhadap perubahan topologi jaringan. Algoritma *routing* merupakan distribusi dimana node di dalam jaringan memerlukan pemeliharaan informasi mengenai node tetangganya. Protokol menyediakan *multiple route* dari *source* menuju *destination*. Keuntungan *multiple route* menuju tujuan adalah pada saat topologi berubah tidak perlu lagi proses *route discovery* ketika terdapat rute yang rusak untuk ruting data. TORA memiliki bagian utama diantaranya : *route discovery (creating routes)*, *route maintenance*, dan *route erasure* [12]. Tiga tipe paket yang digunakan dalam TORA diantaranya : paket QRY (*Query*) yang digunakan untuk pembangunan route, paket UPD (*Update*) yang digunakan untuk pembangunan dan pemeliharaan route, dan paket CLR (*Clear*) yang digunakan untuk penghapusan rute. Pada TORA, protocol melakukan penghapusan rute ketika mendeteksi bagian yang tidak terhubung (putus).

Setiap node memiliki elemen yang terdiri dari:

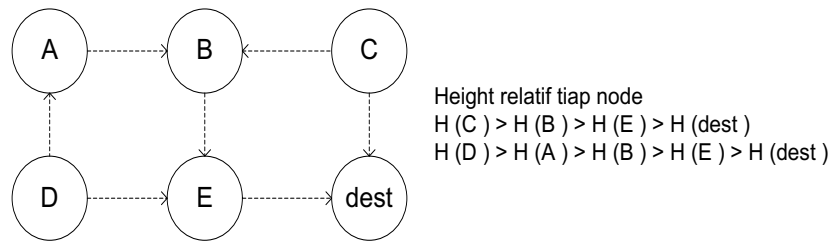
- t, waktu terjadi kegagalan
- oid, originator id
- r, indikasi bit refleksi (0 = level original, 1 = level refleksi)
- d, level referensi
- i, id node

$H(\text{height}) = (t, \text{oid}, r, d, i)$. Height dari suatu node menentukan path untuk tujuan yang dimaksud. (t, oid, r) adalah level referensi sedangkan (d, i) adalah offset. TORA menggunakan DAG (*Directed Acyclic Graph*) yaitu jenis graph direct (langsung) tanpa ada directed cycles;

dapat dikatakan untuk setiap node *i*, tidak ada path yang bermula dan berakhir di node yang sama.

Setiap node memelihara tabel informasi yang berisikan height node – node tetangga. Pada kondisi awal, height dari semua node adalah null (“-“), sedangkan height tujuan adalah (0,0,0,0,dest). Tiap node memberitahukan height kepada tetangganya. Jika tidak terdapat info mengenai tetangga, height adalah null. Jika tetangga adalah tujuan, height adalah 0. Tiap node memelihara kondisi link sekitarnya. Jika height suatu node lebih besar dari tetangga *j*, link menuju *j* disebut downstream. Lainnya disebut upstream.

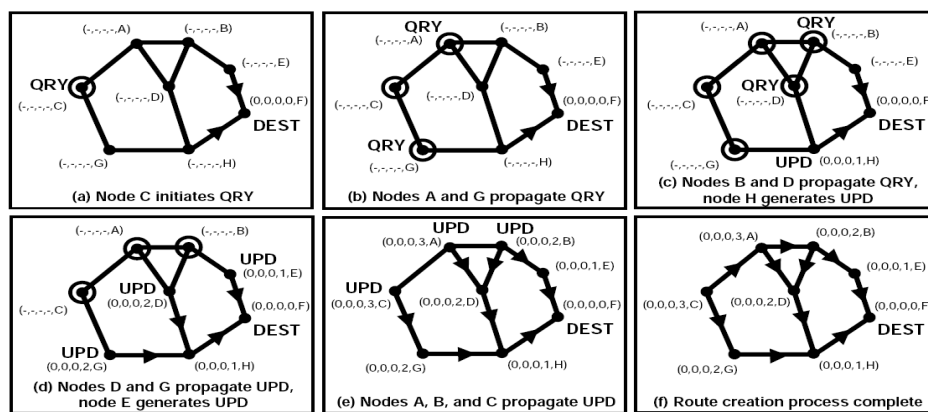
Singkatnya, height dari setiap node dan link yang terhubung membentuk, jalur bebas loop, struktur multipath routing dimana semua node mengarah downstream ke tujuan. Pada gambar 2.5 perhatikan bahwa C lebih dekat pada tujuan daripada B dalam arti jumlah hop, tetapi height C lebih besar dari B.



Gambar 4. Representasi DAG

2.5.1 Route Discovery

Pembentukan rute dilakukan dengan menggunakan paket *query* (QRY) dan *update* (UPD). Paket QRY berisikan id node tujuan yang dimaksud. Balasan untuk paket *query* (QRY) disebut paket *update* (UPD) yang berisi lima elemen (t,oid,r,d,i). Node yang membutuhkan link menuju tujuan karena tidak memiliki tetangga downstream untuk mengirimkan QRY akan mengatur atau mengeset *RR flag* (Route required Flag).



Gambar 5. Proses Route Discovery

Dari gambar 5 di atas node C sebagai node sumber dan node F sebagai node tujuan. Node C mengirimkan paket QRY kepada node tetangga, A dan G. Node A meneruskan paket ke node B dan D, sedangkan node G meneruskan paket tersebut ke node H. node B dan D selanjutnya meneruskan paket QRY kepada node tetangga. Node D,G, dan E yang mendapat paket QRY meng-update tabel routing, sekaligus mengirim paket UPD kepada node yang

bersebelahan (neighbor). Selanjutnya, Node A, B, dan C juga meng-update tabel routing sekaligus mengirimkan paket UPD. Proses pembangunan rute berakhir ketika semua node telah meng-update tabel routing.

2.5.2 Route Maintenance

Route maintenance proses pemeliharaan route jika terjadi perubahan topologi dapat dikarenakan oleh penambahan jumlah node ataupun pergerakan node. Proses pemeliharaan route menggunakan paket *update* (UPD).

2.5.3 Route Erasure

Suatu node akan mendeteksi adanya bagian yang terpisah dalam suatu jaringan yang disebabkan topologi jaringan yang berubah. Node akan melakukan penghapusan rute ketika mendeteksi bahwa bagian tersebut terpisah atau tidak ada path menuju tujuan. TORA menggunakan paket *clear* (CLR) dalam proses route erasure.

METODE PENELITIAN

3.1 Analisa Sistem

Simulasi ini terdiri dari tiga bagian utama yaitu program utama yang tersusun dengan bahasa skrip OTCL, skenario mobilitas node dan skenario generator trafik. Skenario mobilitas terdiri dari dua bagian yaitu kecepatan dengan interval V_{min} dan V_{max} [V_{min} , V_{max}], sedangkan skenario trafik terbagi menjadi tiga bagian yaitu model sumber trafik CBR, Exp dan Pareto. Untuk setiap model trafik karakteristik pembangkitan datanya berbeda. Trafik CBR membangkitkan data secara kontinyu, sedangkan untuk trafik Exp dan Pareto, terdapat periode On dan Off pada pembangkitan data, sesuai dengan fungsi distribusinya masing-masing. Sehingga jumlah paket data yang dikirimkan untuk trafik Exp dan Pareto lebih sedikit dibandingkan trafik CBR.

3.2 Parameter QoS Jaringan Adhoc

Beberapa parameter untuk QoS jaringan Ad Hoc didapat dari performansi protocol routing yang digunakan. Hal tersebut dapat dilihat dari :

a. Packet Delivery Ratio

Packet delivery ratio dihitung dengan membagi jumlah paket yang diterima oleh tujuan berbanding paket yang dikirim pada layer aplikasi dari sumber (contoh : constant Bit Rate /CBR). Ini menunjukkan tingkat packet loss, yang membatasi throughput maksimum dari jaringan. Semakin baik delivery ratio, maka semakin bagus suatu protocol routing. Dapat dirumuskan sebagai berikut:

$$PDR = \frac{\sum \text{Paket data yang diterima}}{\sum \text{Paket data yang dikirim}} \times 100\%$$

b. Normalized Routing Overhead

Normalized Routing Overhead adalah jumlah paket routing dibagi jumlah paket data yang diterima. Routing overhead hanyalah perbandingan antara paket routing dan paket data yang dapat bervariasi nilainya. Paket routing merupakan paket yang digunakan oleh

protocol routing untuk mengetahui kondisi jaringan sekitar dengan saling bertukar informasi antar node (cth : UPD, QRY, CLR). Routing Overhead dapat dirumuskan sebagai berikut:

$$Overhead = \frac{\sum \text{Paket routing}}{\sum \text{Paket data yang diterima}}$$

c. Troughput

Throughput (pesan/detik) adalah jumlah total paket data yang dikirim dibagi waktu durasi simulasi. Dalam hal ini, dievaluasi troughput yang dapat dicapai oleh dengan menggunakan protocol routing TORA. Secara matematis dituliskan sebagai berikut:

$$Throughput = \frac{\sum \text{Paket data yang diterima}}{\text{waktu simulasi}}$$

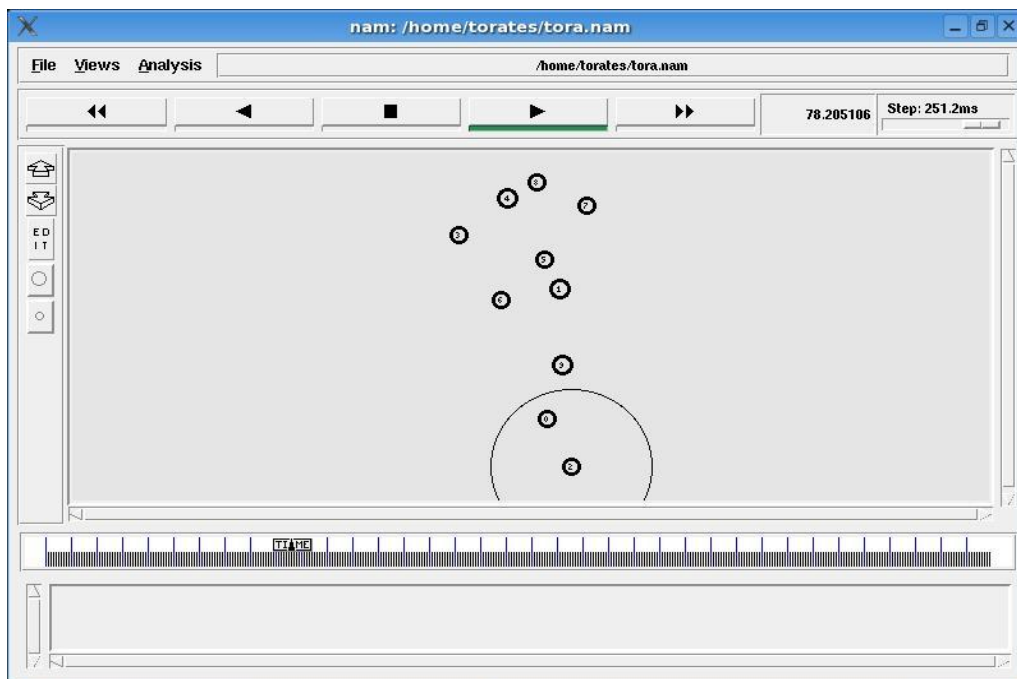
d. Average End-to-end Delay

Average end - to - end delay (detik) adalah waktu rata -rata suatu paket data untuk mencapai tujuan. Parameter ini dihitung dengan mengurangi waktu pertama kali paket sampai ditujuan dengan waktu pertama kali paket dikirim. Kondisi ini dapat dirumuskan sebagai berikut:

$$End -to-end delay = t_{\text{terima}} - t_{\text{kirim}}$$

3.3 Model Simulasi

Pada simulasi ini dibagi menjadi tiga bagian yaitu pemodelan mobilitas menggunakan Random WayPoint Mobility model, pemodelan trafik (CBR, Exp dan Pareto), yang terakhir pemodelan node yang dibuat oleh cmu monarch project.



Gambar 6. Tampilan Simulasi

3.4 Skenario Umum Simulasi

Tabel 1. Parameter Simulasi Utama

Parameter	Nilai
Area Simulasi	300 X 600 meter
Jumlah node	10,20,30
Kapasitas Link Wireless	2 Mb/s
Model Koneksi	Peer to peer
Model Pergerakan	Random Way Point
Kecepatan	0,1,5,10, 15,20 m/s
Pause time	25 s
Model Trafik (generator trafik)	CBR, Pareto, Eksponensial

Dalam penelitian ini, akan dicoba beberapa skenario untuk mengetahui kinerja routing TORA pada kondisi jaringan yang berbeda. Parameter simulasi diperlihatkan pada tabel 1.

3.4.1 Skenario Simulasi

Node pengirim dan penerima dipilih pada node yang memiliki pergerakan paling banyak diantara node yang lain. Pada simulasi ini hanya terdapat dua koneksi dari empat node yang saling bertukar paket.

Tabel 2. Parameter Simulasi Skenario

Skenario	Pause Time	Kecepatan	Jumlah node	Sumber Trafik	Keterangan
Skenario 1, 2 dan 3	25 s	0 m/s	10,20,30	CBR,	Tanpa Background trafik
		1 m/s	10,20,30	Pareto,	
		5 m/s	10,20,30	Eksponensial	
		10 m/s			
		15 m/s			
	25 s	20 m/s			
		0 m/s	10,20,30		Dengan Background trafik
		1 m/s	10,20,30		
		5 m/s	10,20,30		
		10 m/s			
15 m/s					
		20 m/s			

Berikut ini adalah gambaran dari skenario:

- T = 0.0 : setiap node diletakkan pada area simulasi dengan posisi acak dimana tidak ada pergerakan dan pengiriman paket.
- T = 2.0 : node pengirim (source) data akan mengirimkan data menuju node penerima (destination).
- T = 25.0 : Seluruh node yang ada dalam simulasi bergerak secara random menuju kearah tertentu dengan kecepatan moderat untuk setiap skenario simulasi.
- T = 300.0 : Seluruh node berhenti bergerak dan waktu simulasi selesai.

Background Traffic yang akan digunakan memiliki parameter sebagai berikut:

- Transport agent : UDP, protocol yang bersifat connectionless, cocok untuk mentransferkan multimedia dan flooding pada jaringan
 - Ukuran paket : 10000 byte
 - Rate yang dikirimkan : 256 Kbps, 512 Kbps, 1024 Kbps
 - Generator trafik : CBR, berfungsi membangkitkan data dengan bit rate yang konstan
- Untuk ukuran paket data akan dijelaskan sebagai berikut pada table 3:

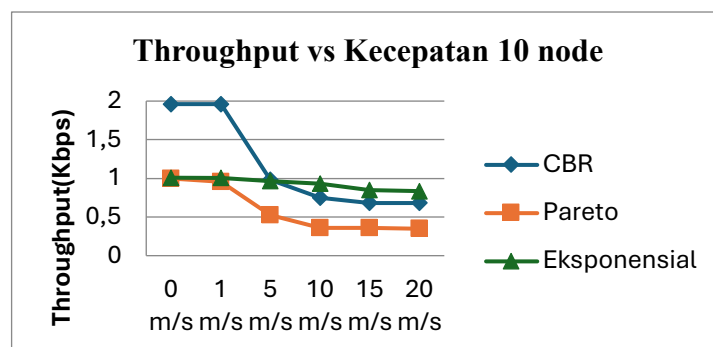
Tabel 3. Parameter Paket Data

<i>CBR Parameter</i>	Distribusi	Nilai
<i>Packet Size</i>	Konstan	64 bytes
<i>Rate</i>	Konstan	2 kb /s
<i>Pareto parameter</i>	Distribusi	Nilai
<i>Packet Size</i>	Konstan	64 bytes
<i>Burst time</i>	Pareto	315 ms
<i>Idle time</i>	Pareto	325 ms
<i>Rate</i>	Konstan	2 kb /s
<i>Shape parameter</i>	Konstan	1,5
<i>Exponensial parameter</i>	Distribusi	Nilai
<i>Packet Size</i>	Konstan	64 bytes
<i>Burst time</i>	Ekspensial	315 ms
<i>Idle time</i>	Ekspensial	325 ms
<i>Rate</i>	Konstan	2 kb /s

HASIL DAN PEMBAHASAN

4.1 Analisa Troughput

4.1.1 Analisa Troughput terhadap Kecepatan

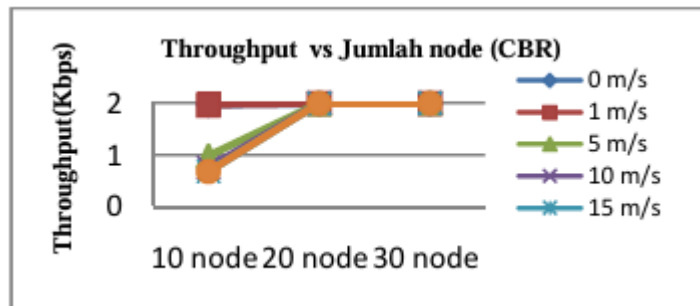


Gambar 7. Troughput terhadap Kecepatan 10 Node

Dengan bertambahnya kecepatan gerak node maka akan sering terjadi perubahan posisi node (topologi berubah-ubah), hal ini akan menyebabkan rute akan semakin sering berganti. Pada saat pencarian rute yang baru maka paket data akan tertahan di buffer. Semakin lama pencarian rute maka semakin besar pula jumlah paket data yang tertahan di

buffer, ketika buffer penuh (overflow) maka paket data akan dibuang (Drop), sehingga throughput semakin turun.

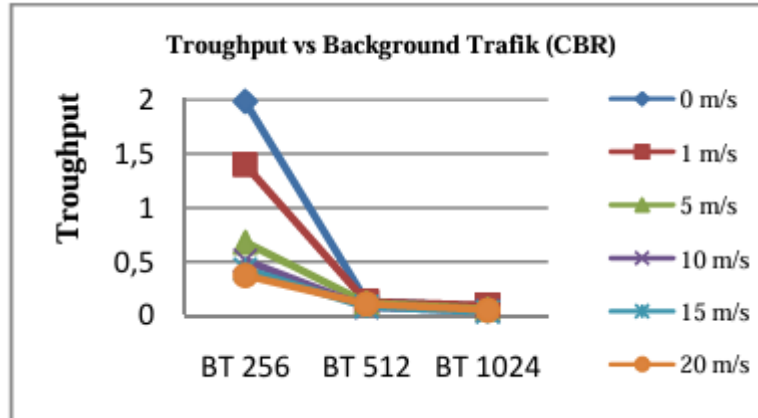
4.1.2 Analisa *Throughput* terhadap Jumlah Node



Gambar 8. *Throughput* terhadap Jumlah Node

Semakin bertambahnya jumlah node maka trafik *routing* akan semakin padat. Hal ini disebabkan karena protokol routing TORA menggunakan mekanisme flooding pada proses *route discovery* yang membanjiri jaringan dengan route request. Karena protokol ini mendukung “*multipath*”, dengan bertambahnya node maka semakin banyak rute yang bisa dibuat sehingga proses pencarian rute bisa 5 menjadi lebih cepat dan jumlah paket data yang tertahan di buffer pun berkurang, di hal ini meningkatkan *throughput*.

4.1.3 Analisa *Throughput* terhadap Background Trafik

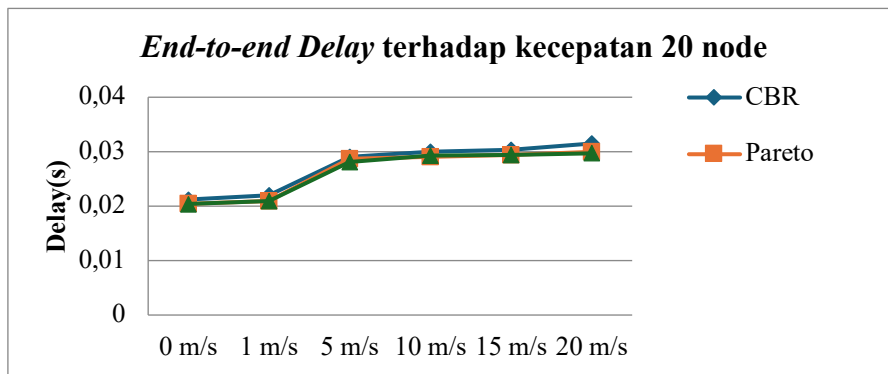


Gambar 9. *Throughput* terhadap Background Trafik 10 Node (CBR)

Dengan peningkatan background trafik, maka kanal banyak digunakan untuk mengirimkan background trafik dan ketersediaan kanal semakin sedikit untuk mengirimkan paket data, sehingga hal tersebut dapat menurunkan *throughput*.

4.2 Analisa *End-to-end Delay*

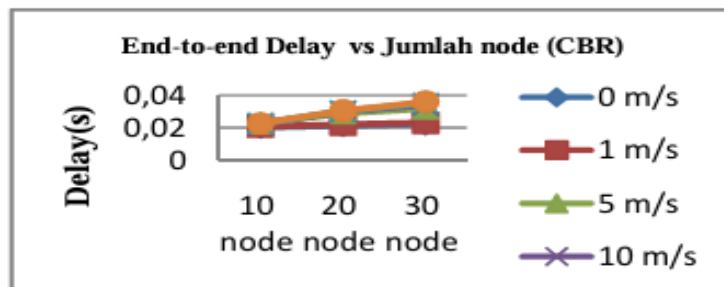
4.2.1 Analisa *End-to-end Delay* terhadap Kecepatan



Gambar 10. End-to-end Delay terhadap Kecepatan 20 Node

Dengan bertambahnya kecepatan gerak node maka akan sering terjadi perubahan posisi node (topologi berubah-ubah), hal ini akan menyebabkan rute akan semakin sering berganti. Pada saat pencarian rute yang baru maka paket data akan tertahan di *buffer* sementara sampai rute baru ditemukan. Seringnya perubahan rute ini akan mengakibatkan banyaknya paket data yang tertahan di *buffer*, sehingga rata-rata *delay* pengiriman paket data akan semakin besar.

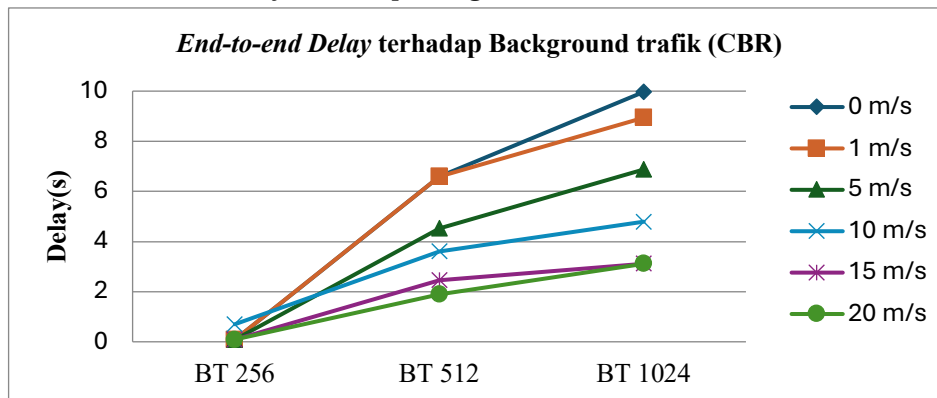
4.2.2 Analisa End-to-end Delay terhadap Jumlah Node



Gambar 11. End-to-end Delay terhadap Jumlah Node

Dengan bertambahnya jumlah node pada jaringan, kemungkinan besar akan meningkatkan jumlah hop yang diperlukan untuk mencapai node tujuan. Hal ini akan menimbulkan *delay* tambahan. Sehingga *delay* total pengiriman paket data akan semakin tinggi.

4.2.3 Analisa End-to-end Delay terhadap Background Trafik

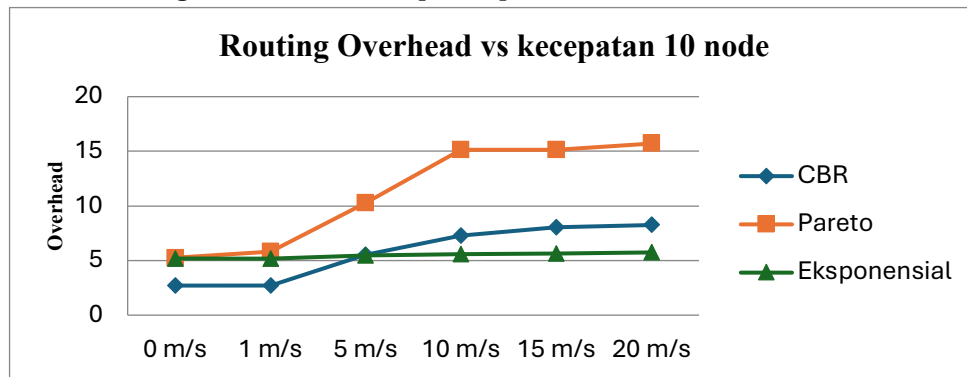


Gambar 12. End-to-end Delay terhadap Background Trafik

Dengan peningkatan *background* trafik, maka kanal banyak digunakan untuk mengirimkan *background* trafik dan ketersediaan kanal semakin sedikit untuk mengirimkan paket data, sehingga paket data banyak tertahan di *buffer* sebelum sampai ke tujuan. Hal ini dapat meningkatkan *end-to-end delay*.

4.3 Analisa Routing Overhead (ROV)

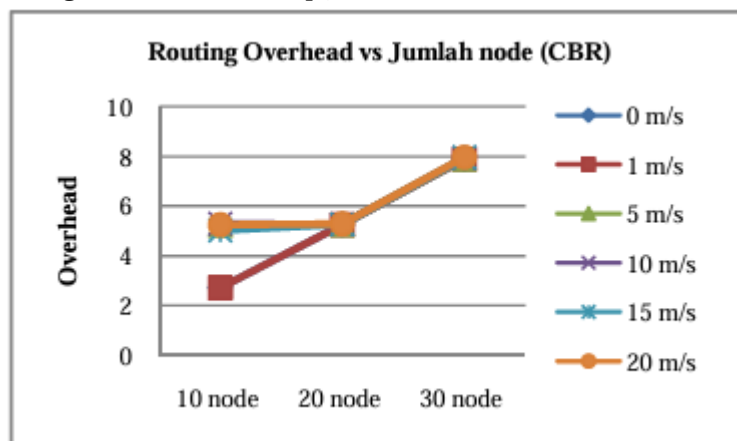
4.3.1 Analisa Routing Overhead terhadap Kecepatan



Gambar 13. Routing Overhead terhadap kecepatan 10 node

Ketika kecepatan gerak node bertambah, maka posisi node juga semakin mudah berubah dan memungkinkan suatu node tidak lagi dalam *coverage area* dari suatu node sebelumnya dan menyebabkan konfigurasi jaringan sering berubah-ubah. Sehingga node pengirim akan lebih banyak melakukan *broadcast* paket *routing* pada proses *route discovery* (RREQ dan RREP) untuk mendapatkan rute yang baru. Dengan bertambahnya kecepatan gerak node akan meningkatkan jumlah paket *routing*.

4.3.2 Analisa Routing Overhead terhadap Jumlah Node

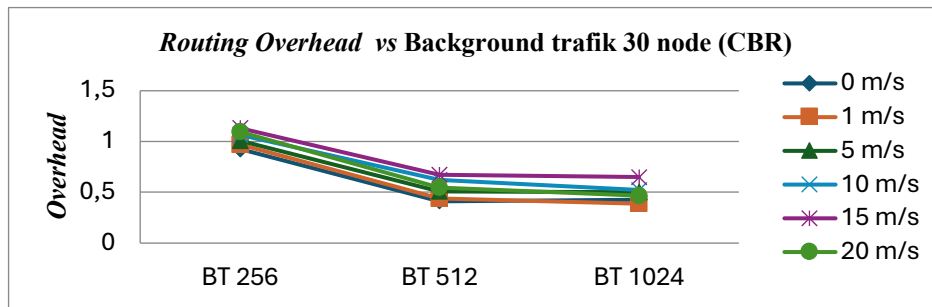


Gambar 14. Routing Overhead terhadap Jumlah Node

Routing overhead masing-masing trafik juga meningkat seiring dengan penambahan jumlah node. Hal ini dikarenakan : Adanya mekanisme *flooding* pada penggunaan algoritma *routing* TORA. Seperti yang sudah dijelaskan pada pembahasan sebelumnya *flooding* adalah suatu proses pada protokol *routing* untuk melakukan *broadcast* paket kontrol informasi pada seluruh node di jaringan. Node pengirim akan mengirimkan paket kontrol informasi tersebut pada node tetangganya (semua node yang termasuk didalam *coverage area*nya), kemudian node juga mem-*broadcast* paket tersebut

sampai seluruh node di jaringan menerima paket tersebut. Pada protokol *routing* TORA akan melakukan mekanisme *flooding* dengan mengirimkan paket RREQ dan RREP pada saat *route discovery* dan ketika jumlah node bertambah maka jumlah paket RREQ dan RREP yang dikirimkan juga semakin bertambah besar.

4.3.3 Analisa Routing Overhead terhadap Background Trafik

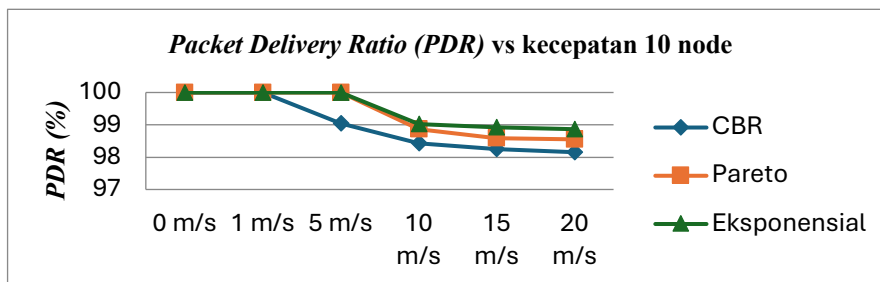


Gambar 15. Routing Overhead terhadap Jumlah Node

Dengan peningkatan *background* trafik, trafik *routing* akan semakin padat. Hal ini disebabkan karena protokol *routing* TORA menggunakan mekanisme *flooding* pada proses *route discovery* yang membanjiri jaringan dengan *route request*. Karena trafik data dan *background* trafik dilewatkan pada jaringan yang dihasilkan oleh node sumber dengan node tujuan yang sama serta banyaknya paket yang dikirimkan, maka rute untuk mencapai tujuan tersimpan pada *route cache*. Dengan menggunakan *route cache* tersebut setiap pengiriman paket data dapat menggunakan *route* yang masih tersimpan tanpa harus mengirimkan paket RREQ sehingga akan mengurangi *overhead* paket kontrol informasi routing, sehingga *overhead* turun.

4.4 Analisa Packet Delivery Ratio (PDR)

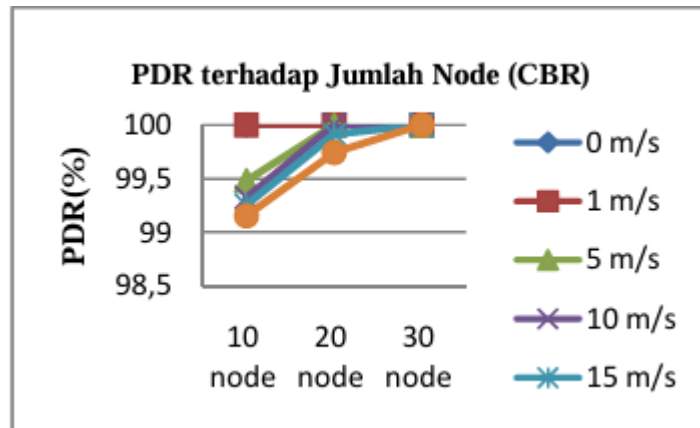
4.4.1 Analisa Packet Delivery Ratio (PDR) terhadap Kecepatan



Gambar 16. PDR terhadap Kecepatan

Dengan bertambahnya kecepatan gerak node maka akan sering terjadi perubahan posisi node (topologi berubah-ubah), hal ini akan menyebabkan rute akan semakin sering berganti. Pada saat pencarian rute yang baru maka paket data akan tertahan di *buffer*. Semakin lama pencarian rute maka semakin besar pula jumlah paket data yang tertahan di *buffer*, ketika *buffer* penuh (*overflow*) maka paket data akan dibuang (*Drop*), sehingga PDR semakin turun.

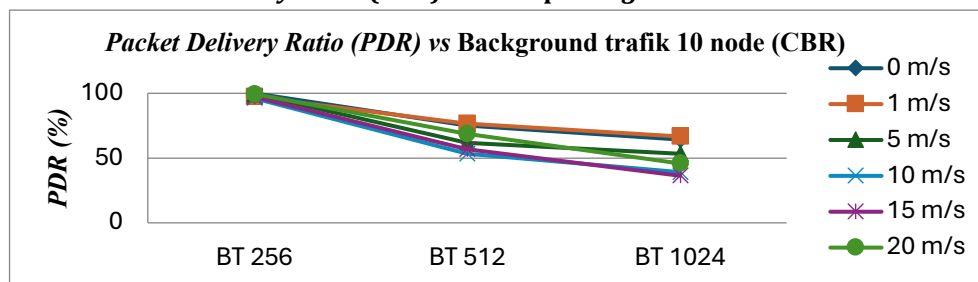
4.4.2 Analisa Packet Delivery Ratio (PDR) terhadap Jumlah Node



Gambar 17. PDR terhadap Jumlah Node

Semakin bertambahnya jumlah node maka trafik *routing* akan semakin padat. Hal ini disebabkan karena protokol *routing* TORA menggunakan mekanisme *flooding* pada proses *route discovery* yang membanjiri jaringan dengan *route request*. Karena protokol ini mendukung “*multipath*”, dengan bertambahnya node maka semakin banyak rute yang bisa dibuat sehingga proses pencarian rute bisa menjadi lebih cepat dan jumlah paket data yang tertahan di *buffer* pun berkurang, hal ini dapat meningkatkan PDR.

4.4.3 Analisa Packet Delivery Ratio (PDR) terhadap Background Trafik



Gambar 18. PDR terhadap Background Trafik

Dengan peningkatan *background* trafik, maka kanal banyak digunakan untuk mengirimkan *background* trafik dan ketersediaan kanal semakin sedikit untuk mengirimkan paket data, sehingga paket data banyak tertahan di *buffer* sebelum sampai ke tujuan. Ketika *buffer* penuh (*overflow*) maka paket data akan dibuang (*Drop*), sehingga PDR semakin turun.

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari penelitian yang telah dilakukan, baik melalui simulasi maupun analisa kualitatif dapat diambil beberapa kesimpulan sebagai berikut:

1. *Throughput* dan *Packet Delivery Ratio (PDR)* menggunakan *protocol routing* TORA semakin meningkat ketika jumlah node bertambah besar. *Throughput* terbesar didapat oleh trafik CBR dibandingkan trafik Pareto dan Eksponensial, yaitu sebesar 1,96096 Kbps(10 node), 1,98656 Kbps(20 node), dan 1,98997(30 node). Walaupun begitu delay trafik CBR lebih besar dari pada trafik Exp dan Pareto terutama pada saat kecepatan

- tinggi (5 m/s, 20 m/s) yaitu 0,0290198 s dan 0,0305027 s, Pareto (0,0286456 s dan 0,0299725 s), Eksponensial (0,0281603 s dan 0,0296971 s). Sedangkan untuk PDR terbesar didapat oleh trafik Eksponensial dibandingkan kedua trafik lain yaitu 98,8647319%, 98,55072464% (pareto) dan 98,15272235% (CBR).
2. Pada skenario peningkatan kecepatan dan *background* trafik akan menurunkan performansi dari *protocol routing* TORA pada *throughput* dan PDR. Hal ini dapat dilihat dari *throughput* dan *Packet Delivery Ratio (PDR)* yang semakin turun pada masing – masing trafik ketika kecepatan bertambah dan intensitas *background* trafik meningkat. Untuk *throughput* dan PDR pada trafik CBR didapat 1,96096 Kbps dan 100 % (1 m/s), 0,98304 Kbps dan 99,03846154% (5 m/s), 0,68096 dan 98,15272235%.
 3. *Routing overhead* menggunakan *protocol routing* TORA pada masing – masing trafik akan semakin meningkat ketika kecepatan bertambah dan jumlah node bertambah besar. Untuk trafik CBR sebesar 2,7075718 (1 m/s), 5,23125 (5 m/s), 5,24812035 (20 m/s) dan 2,7075718 (10 node), 5,24978541(20 node), dan 7,88517566(30 node). Untuk trafik pareto 5,25641026 (10 node), 11,10036496 (20 node), dan 13,11781609 (30 node). Pada trafik eksponensial 5,18274112 (10 node), 9,63116057 (20 node), dan 14,2125 (30 node).
 4. Secara keseluruhan, protokol *routing* TORA cocok digunakan untuk jaringan dengan banyak user (banyak node) dan kondisi trafik padat.

5.2 Saran

Penggunaan protokol *routing* TORA pada jaringan ad hoc dapat dikembangkan lagi dengan memperhatikan beberapa hal sebagai berikut:

1. Penggunaan protokol *routing* TORA pada jaringan yang memiliki karakteristik trafik yang lebih padat, dengan cara menambah jumlah node dan koneksi antar node.
2. Memperbesar waktu pengamatan (simulasi) untuk masing-masing sumber trafik, sehingga diperoleh performansi yang paling baik untuk protokol routing TORA.

DAFTAR PUSTAKA

- [1] Shreya Mane, "Conceptual Aspects on Mobile Ad-Hoc Network System," *Int. J. Eng. Technol. Manag. Sci.*, vol. 6, no. 6, pp. 555–564, 2022, doi: 10.46647/ijetms.2022.v06i06.095.
- [2] M. A. Al-Absi, A. A. Al-Absi, M. Sain, and H. Lee, "Moving ad hoc networks—a comparative study," *Sustain.*, vol. 13, no. 11, 2021, doi: 10.3390/su13116187.
- [3] M. A. Khan, I. U. Khan, A. Safi, and I. M. Quershi, "Dynamic routing in flying Ad-Hoc networks using topology-based routing protocols," *Drones*, vol. 2, no. 3, pp. 1–15, 2018, doi: 10.3390/drones2030027.
- [4] A. H. Wheeb, R. Nordin, A. A. Samah, M. H. Alsharif, and M. A. Khan, "Topology-Based Routing Protocols and Mobility Models for Flying Ad Hoc Networks: A Contemporary Review and Future Research Directions," *Drones*, vol. 6, no. 1, 2022, doi: 10.3390/drones6010009.
- [5] D. T. W. AM, R. Munadi, and R. Mayasari, "Analisis Pengaruh Dynamic Source Routing

- Dan Temporally Ordered Routing Algorithm Terhadap Tabrakan Data Pada Vanet,” *Transmisi*, vol. 20, no. 4, p. 138, 2019, doi: 10.14710/transmisi.20.4.138-144.
- [6] Y. Sarada Devi and M. Roopa, “Performance Analysis of Routing Protocols in Vehicular Ad hoc Networks,” *Lect. Notes Electr. Eng.*, vol. 1155, no. 3, pp. 413–424, 2024, doi: 10.1007/978-981-97-0644-0_36.
- [7] A. S. Kasem Mohammed and K. Hamid Bilal, “Survey of mobile ad hoc networks (manets),” *Int. J. Eng. Comput. Sci.*, vol. 8, no. 02, pp. 24484–24490, 2019, doi: 10.18535/ijecs.v8i02.4229.
- [8] *Chapter 5 IP Routing Fundamentals*.
- [9] A. Bhardwaj and H. El-Ocla, “Multipath routing protocol using genetic algorithm in mobile ad hoc networks,” *IEEE Access*, vol. 8, pp. 177534–177548, 2020, doi: 10.1109/ACCESS.2020.3027043.
- [10] S. Chaudhari, “A Survey on Multipath Routing Techniques in Wireless Sensor Networks,” *International Journal of Networking and Virtual Organitations* 24, no.3, 2021: 267-328.
- [11] G. Vignesh, R. Vengatesan, R. Mohandas, and S. Balasubramanian, “A Review of Temporally Ordered Routing Algorithm Protocol’s Challenges and Future Directions,” *Proc. 9th Int. Conf. Commun. Electron. Syst. ICCES 2024*, pp. 733–740, 2024, doi: 10.1109/ICCES63552.2024.10859533.
- [12] Z. Zhai, J. Du, and Y. Ren, “The Application and Improvement of Temporally Ordered Routing Algorithm in Swarm Network with Unmanned Aerial Vehicle Nodes,” *Ninth Int. Conf. Wirel. Mob. Commun.*, no. c, pp. 7–12, 2013.