



ANALISIS KINERJA MEKANISME ROUTING FLOODING DAN CONVERGECAST PADA WIRELESS SENSOR NETWORK

SKRIPSI

Untuk memenuhi sebagian persyaratan

Memperoleh gelar Sarjana Komputer

Disusun oleh:

Kharisma Fadillah

NIM:135150201111126



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA

MALANG

2020

PENGESAHAN

ANALISIS KINERJA MEKANISME ROUTING FLOODING DAN CONVERGECAST PADA
WIRELESS SENSOR NETWORK

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Kharisma Fadillah
NIM:135150201111126

Skripsi ini telah diuji dan dinyatakan lulus pada
22 Juli 2020

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Ir. Heru Nurwarsito, M.Kom
NIP:196504021990021001

Mengetahui

Ketua Jurusan Teknik Informatika



Achmad Basuki, S.T, M.MG, Ph.D
NIP: 19741118 200312 1 002 94



PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar referensi.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiarisme, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 22 Juli 2020

Kharisma Fadillah.

Kharisma Fadillah

NIM: 135150201111126



ABSTRAK

Kharisma Fadillah, ANALISIS KINERJA MEKANISME ROUTING FLOODING DAN CONVERGECAST PADA WIRELESS SENSOR NETWORK

Pembimbing: Ir. Heru Nurwarsito, M.Kom

Wireless Sensor Network (WSN) adalah jaringan sensor nirkabel yang memiliki tujuan tertentu, termasuk penggunaan untuk pemantauan. Untuk mengetahui paket data yang ditransfer dan jumlah total paket data yang hilang memerlukan protokol routing yang bertugas menentukan jalur terpendek dari node source ke node destination. Pada penelitian sebelumnya *routing flooding* dan *routing gossiping* yang diimplementasikan dengan simple yang dilakukan dengan proses simulasi untuk mengulas kinerja jaringan dan penggunaan daya pada jaringan WSN, dimana hasil yang didapat *routing gossiping* lebih efisien dalam masalah energi. Dari penelitian sebelumnya peneliti mengambil ide bahwa *routing convergecast* mempunyai salah satu kelebihan yang sama terhadap *routing flooding* yaitu sama-sama memiliki kinerja terbaik dalam pengiriman data. Penelitian ini melakukan analisis algoritma protokol routing flooding dan *convergecast* yang bertujuan untuk menilai performa kinerja routing berdasarkan ukuran latency, backoff time, power consumption, juga untuk menganalisis jenis routing protokol yang optimum dalam jaringan wireless sensor network. Perancangan topologi menggunakan 5 macam node yaitu 7, 14, 21, 28, dan 35 node. pengujian dilakukan dengan parameter latency, backoff, activity durations, dan power consumption. Hasil parameter activity durations dengan nilai 13 ms, lalu latency dengan nilai dibawah 0,005 ms, sedangkan untuk *routing flooding* mempunyai nilai yang hampir sama dengan *convergecast* di akhir node untuk parameter backoff (3,1 ms *convergecast* & 3,2 ms *flooding*) dan power consumption (56,3 mW kedua routing)

Kata kunci : *flooding, Convergecast, Wireless Sensor Network, OMNeT++, MiXiM, Routing*



ABSTRACT

Kharisma Fadillah , ANALYSIS OF MECANISM ROUTING FLOODING AND CONVERGECAST PERFORMANCE IN WIRELESS SENSOR NETWORK**Supervisors : Ir. Heru Nurwarsito, M.Kom**

Wireless Sensor Network (WSN) is a wireless sensor network that has a specific purpose, including use for monitoring. To find out the transferred data packets and the total number of data packets lost requires a routing protocol that is tasked with determining the shortest path from the source node to the destination node use of power in the WSN network, where the results obtained routing gossiping are more efficient in terms of energy. From previous research, the researcher took the idea that convergecast routing has one of the same advantages of flooding routing, namely the same as having the best performance in data transmission. This research analyzes the routing protocol algorithm flooding and convergecast which aims to assess the performance of the routing performance based on latency, backoff time, power consumption, as well as to analyze the optimum type of routing protocol in a wireless sensor network. Topology design uses 5 kinds of nodes, namely 7, 14, 21, 28, and 35 nodes. Testing is done with parameters of latency, backoff, activity durations, and power consumption. The result of the activity durations parameter is 13 ms, then latency is below 0.005 ms, while for routing flooding it has a value that is almost the same as the convergecast at the end of the node for backoff parameters (3.1 ms convergecast & 3.2 ms flooding) and power consumption (56.3 mW second routing)

Keywords : *flooding, Convergecast, Wireless Sensor Network, OMNeT++, MiXiM, Routing*



DAFTAR ISI

PENGESAHAN.....	ii
PERNYATAAN ORISINALITAS.....	iii
PRAKATA.....	iv
ABSTRAK.....	vi
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	xi
DAFTAR TABEL.....	xii
DAFTAR LAMPIRAN.....	xiii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan.....	2
1.4 Manfaat.....	3
1.5 Batasan Masalah.....	3
1.6 Sistematika Penulisan.....	3
BAB 2 LANDASAN KEPUSTAKAAN.....	5
2.1 Kajian Pustaka.....	5
2.2 Kajian Teori.....	8
2.2.1 Wireless Sensor Network.....	8
2.2.2 Protokol Routing.....	10
2.2.3 Routing Flooding.....	12
2.2.4 Convergecast.....	14
2.2.5 Parameter Analysis.....	15
2.2.6 Omnet++.....	17
BAB 3 METODOLOGI.....	19
3.1 Studi Literatur.....	20
3.2 Analisis Kebutuhan.....	20
3.2.1 Kebutuhan Fungsional.....	21
3.2.2 Kebutuhan Non Fungsional.....	21



3.3 Perancangan Sistem.....	21
3.4 Implementasi.....	21
3.5 Pengujian.....	22
3.5.1 Skenario Pengujian Pertama dengan Activity Duration.....	23
3.5.2 Skenario Pengujian Kedua dengan BackOff Durations.....	23
3.5.3 Skenario Pengujian Ketiga dengan Latency.....	23
3.5.4 Skenario Pengujian Kelima dengan Power Consumption.....	23
3.6 Analisis Hasil.....	23
3.7 Kesimpulan.....	23
BAB 4 PERANCANGAN DAN IMPLEMENTASI.....	25
4.1 Lingkungan Penelitian.....	25
4.1.1 Analisis Kebutuhan.....	26
4.2 Perancangan.....	28
4.2.1 Perancangan Topologi.....	28
4.2.2 Perancangan <i>Routing Flooding</i>	29
4.2.3 Perancangan <i>Routing Convergecast</i>	31
4.2.4 Skenario Pengujian.....	32
4.3 Implementasi.....	34
4.3.1 Desain implementasi.....	34
4.3.2 Alur Kerja Simulasi.....	35
4.3.3 Konfigurasi Script Simulasi.....	36
BAB 5 PENGUJIAN DAN ANALISIS HASIL.....	41
5.1 Skenario Pengujian.....	41
5.2 Hasil Pengujian.....	43
5.2.1 Hasil Pengujian Skenario Pertama (<i>Activity Durations</i>).....	43
5.2.2 Hasil Pengujian Skenario Kedua (<i>BackOff Durations</i>).....	46
5.2.3 Hasil Pengujian Skenario Ketiga (<i>Latency</i>).....	48
5.2.4 Hasil Pengujian Skenario Kelima (<i>Power Consumption</i>).....	51
BAB 6 PENUTUP.....	55
6.1 Kesimpulan.....	55
6.2 Saran.....	55
DAFTAR PUSTAKA.....	57



LAMPIRAN A BUKTI HASIL DATA PADA TOOLS OMNET.....	60
---	----

LAMPIRAN B HASIL PENGIRIMAN DATA PADA TOOLS OMNET.....	61
--	----

LAMPIRAN C HASIL DATA.....	63
----------------------------	----



DAFTAR GAMBAR

Gambar 2.1 Arsitektur WSN.....	8
Gambar 2.2 Routing <i>Flooding</i>	12
Gambar 2.3 Routing <i>Convergecast</i>	14
Gambar 3.1 Diagram Alir Metodologi.....	19
Gamabr 4.1 Flowchart Pembuatan Sistem.....	26
Gambar 4.2 Topologi 7 Node Sensor.....	29
Gambar 4.3 Flow Diagram Routing <i>Flooding</i>	30
Gambar 4.4 Flow Diagram Routing <i>Convergecast</i>	31
Gambar 4.5 Flow Diagram Desain Implementasi.....	34
Gambar 4.7 Aliran Pengiriman Data.....	38
Gambar 5.1 Pengujian pada Simulator.....	39
Gambar 5.2 Hasil Pengujian.....	42
Gambar 5.3 Grafik Hasil Rata Rata Kedua Routing.....	43
Gambar 5.4 Hasil Pengujian.....	45
Gambar 5.5 Grafik Hasil Rata Rata Kedua Routing.....	46
Gambar 5.6 Hasil Pengujian.....	48
Gambar 5.7 Grafik Hasil Rata Rata Kedua Routing.....	49
Gambar 5.8 Hasil Pengujian.....	51
Gambar 5.9 Grafik Hasil Rata Rata Kedua Routing.....	52



DAFTAR TABEL

Tabel 1.1 Kajian Pustaka.....	7
Tabel 4.1 Kebutuhan Perangkat Keras.....	27
Tabel 4.2 Kebutuhan Perangkat Lunak.....	27
Tabel 4.3 Kebutuhan Riset.....	28
Tabel 4.4 Skenario Pengujian Pertama.....	32
Tabel 4.5 Skenario Pengujian Kedua.....	32
Tabel 4.6 Skenario Pengujian Ketiga.....	33
Tabel 4.7 Skenario Pengujian Keempat.....	33
Tabel 4.8 Konfigurasi Topologi 7 Node Sensor.....	35
Tabel 4.9 Konfigurasi Topologi 14 Node Sensor.....	36
Tabel 4.10 Konfigurasi General SubModul.....	36
Tabel 4.11 Konfigurasi General sim-time-limit.....	37
Tabel 4.12 Konfigurasi Ukuran Area & Jumlah Host.....	37
Tabel 4.13 Konfigurasi <i>Flooding</i>	37
Tabel 4.14 Konfigurasi <i>Convergecast</i>	38
Tabel 5.1 Nilai Hasil Pengujian Pertama.....	41
Tabel 5.2 Nilai Hasil Pengujian Kedua.....	44
Tabel 5.3 Nilai Hasil Pengujian Ketiga.....	47
Tabel 5.4 Nilai Hasil Pengujian Keempat.....	50



DAFTAR LAMPIRAN

Lampiran Data Hasil Pengujian.....69



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Wireless Sensor Network (WSN) adalah jaringan sensor nirkabel yang memiliki tujuan tertentu, termasuk penggunaan untuk pemantauan lingkungan, bencana alam, atau bahkan pertahanan. Dalam WSN ada beberapa komponen dasar yang dapat diatur dalam jaringan, jaringan interkoneksi yang terhubung secara nirkabel, sink node sebagai titik pengumpulan data dan gateway internet serta server untuk menangani pengolahan data. WSN mengacu pada sekelompok sensor spasial tersebar yang berdedikasi untuk pemantauan dan rekaman keadaan fisik lingkungan serta mengatur data yang dikumpulkan di lokasi pusat. Jaringan sensor nirkabel sudah banyak digunakan dalam perkembangan IoT pada masyarakat modern yang diintegrasikan dengan banyak aplikasi-aplikasi yang salah satunya ada pada bidang kesehatan yang difungsikan sebagai pemantauan pasien jantung jarak jauh. Pada bidang pertahanan teknologi atau arsitektur jaringan sensor nirkabel juga sudah diimplementasikan secara langsung atau secara prototype untuk percobaan pada medan perang sebagai pengiriman informasi penting. Dalam lingkungan sehari-hari dalam bidang apapun *Wireless Sensor Network* sudah banyak digunakan sebagai alat untuk *monitoring* (Anwari, 2017).

Berbagai macam aplikasi sudah banyak diimplementasikan pada arsitektur jaringan sensor nirkabel, contohnya sistem pemantauan kegiatan yang ada di lapangan. Salah satu protokol *routing* yaitu *convergecast* yang dapat diimplementasikan pada beberapa aplikasi arsitektur jaringan sensor, karena protokol *routing convergecast* termasuk protokol *routing* yang baik dan efisien. Dengan waktu yang sangat singkat *convergecast* mampu menampilkan data *flow* dari banyaknya node sumber yang berbeda menuju node *sink* dan *convergecast* juga sebagai metode komunikasi umum pada *Wireless Sensor Network*. Untuk mencegah tingkat tabrakan data paket yang mengarah ke node *sink* memerlukan sinkronisasi yang tepat antar node pada jaringan *ad hoc* nirkabel (Jyoteesh, 2015).

Mekanisme *routing* adalah operasi atau proses perutean yang dilakukan dengan suatu dan beberapa cara atau alat. *Routing* dibagi menjadi dua macam yaitu *routing pro active* dan *routing reactive*. *Routing flooding* dan *convergecast* termasuk dalam jenis *routing reactive* karena sifat dari kedua *routing* yang beroperasi ketika sebuah node ingin mentransfer paket ke node lain untuk melacak dan menetapkan saluran yang efisien dan koneksi yang stabil. Dapat dikatakan bahwa *routing* jenis *reactive* cocok untuk jaringan skala kecil karena



tidak sering terjadi perubahan tabel routing, dan data yang terorganisir tidak diperbanyak kecuali ada perubahan dalam topologi. routing reactive juga tidak menimbulkan lalu lintas dan konsumsi daya yang besar. (Jyoteesh, 2015).

Broadcasting bekerja pada jenis jaringan yang berbeda tetapi biasanya digunakan dengan jenis jaringan routing convergecast bertipe wise route. Pada dasarnya mekanisme broadcasting selalu diikuti dengan convergecast pada wireless atau jaringan sensor network. Convergecast tipe wise route memiliki kemampuan meminimalisir loop saat paket data sebar. Wise route adalah algoritma perutean bebas loop sederhana yang membangun pohon perutean dari titik jaringan pusat. Berguna untuk jaringan sensor nirkabel dan traffic wireless sensor network. pada wise route. Node sink membroadcast pesan pembangunan rute, setiap node jaringan yang menerimanya, memiliki node sink sebagai induk pada topologi perutean, dan membroadcast ulang pesan pembuatan rute, prosedur ini memaksimalkan probabilitas bahwa semua node jaringan dapat bergabung dengan jaringan, dan menghindari loop. Mempertimbangkan manfaat seperti yang dijelaskan dalam literatur, pekerjaan pada penelitian ini terutama didasarkan pada perutean jaringan sensor network. Dalam melakukannya perbandingan rute wise dan efek jaringan tipe broadcasting dalam penelitian ini diambil berdasarkan lima parameter seperti activity duration, nilai backoff, latency, dan power consumption.

Dengan dilakukan analisis kinerja antara mekanisme *routing flooding* dan routing convergecast diharapkan mengetahui routing terbaik untuk jaringan sensor khususnya dalam wireless sensor network.

1.2 Rumusan Masalah

Berdasarkan penjelasan latar belakang diatas, maka rumusan masalah yang dapat diangkat yaitu :

1. Bagaimana penerapan mekanisme *routing flooding* dan *routing convergecast* pada *Wireless Sensor Network*?
2. Bagaimana hasil kinerja *routing flooding* dan *routing convergecast* pada *Wireless Sensor Network* ?

1.3 Tujuan

Tujuan umum :

Mengetahui analisis kinerja routing *Flooding* dan *Convergecast* yang diimplementasikan pada *wireless sensor network*

Tujuan Khusus :



1. Menerapkan mekanisme *routing flooding* dan *routing convergecast* pada *Wireless Sensor Network*
2. Mengetahui kinerja *routing flooding* dan *routing convergecast* pada *wireless sensor network*

1.4 Manfaat

Manfaat yang dapat diperoleh dari penelitian ini yaitu mendapatkan hasil analisis dari kedua protokol *routing* yang dapat digunakan sebagai acuan dalam penelitian selanjutnya sebagai pengembangan *Wireless Sensor Network* dengan pemilihan *routing* yang sesuai kebutuhan, dan juga untuk memilih protokol *routing* terbaik pada jaringan *Wireless Sensor Network*. Penulis juga berharap penelitian ini dapat digunakan sebagai pembelajaran terkait *routing* pada *Wireless Sensor Network* dan konfigurasi *routing* pada *Wireless Sensor Network*.

1.5 Batasan Masalah

Untuk tujuan agar kupasan inti sari lebih rinci dan tidak meluas maka diperlukan batasan masalah dalam riset ini. Berikut batasan masalah pada penelitian ini :

1. Perbandingan kinerja *rotting flooding* dan *convergecast*
2. Parameter yang diuji adalah :
 - Nilai *Activity Durations*
 - Nilai *back off*
 - Nilai *power consumption*
 - Nilai *latency*
 - Waktu dari aliran data yang menuju ke *sink* saat topology dijalankan
3. Solusi untuk *routing WSN* hanya digunakan dengan 2 *routing* yang ditentukan
4. Jumlah *node* maksimal yang digunakan untuk simulasi adalah 35 *node*
5. Tools dan *software* menggunakan network simulator OMNET++ dan library pendukung.
6. Penelitian ini menggunakan program / Environment pada library MiXiM
7. Jaringan yang digunakan adalah *Wireless Sensor Network* yang bersifat statis
8. Node menggunakan data device yang bekerja dengan standar IEEE 802.15.4 yaitu low-rate wireless personal area networks (LR-WPANs)

1.6 Sistematika Penulisan

Pada perancangan dan pembahasan penulisan sistematika laporan riset dapat dirinci secara garis besar sebagai berikut :



BAB I PENDAHULUAN

Bab I pendahuluan akan dijelaskan mengenai latar belakang penelitian, rumusan masalah hasil ekstraksi dari latar belakang, tujuan penelitian, manfaat penelitian yang dapat diambil, batasan penelitian supaya peneliti focus pada hal yang ingin dibahas, dan sistematika penulisan.

BAB II LANDASAN KEPUSTAKAAN

Bab II Landasan Kepustakaan berisi tentang teori dan referensi yang berkaitan dengan topik penelitian berdasarkan penelitian sebelumnya ataupun penelitian penunjang lainnya yang bersifat ilmiah sebagai dasar acuan.

BAB III METODELOGI PENELITIAN

Bab III Metodologi membahas langkah kerja yang akan dilakukan dalam penelitian analisis unjuk kerja. Tahapan berikut meliputi studi literatur, persiapan skenario, lingkungan penelitian, pengujian, pembahasan, dan hipotesis dibuat atas dasar penelitian yang dilakukan.

BAB IV PERANCANGAN DAN IMPLEMENTASI

Bab IV Hasil Pengujian akan membahas hasil pengujian dari setiap skenario dan menjelaskan proses perpindahan data pada saat pengujian.

BAB V PENGUJIAN DAN ANALISIS

Bab V Pembahasan akan menjelaskan dan menunjukkan hasil pengujian yang telah didapatkan, yang kemudian hasil tersebut akan dianalisis.

BAB VI PENUTUP

Bab VI Penutup terdapat kesimpulan yang diambil berdasarkan tahapan yang sudah dilakukan mulai dari persiapan skenario, pengujian, hingga pembahasan yang mengacu pada rumusan masalah penelitian.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Sebagai bahan pertimbangan dalam penelitian ini, dicatumkan beberapa hasil penelitian yang sudah ada sebagai berikut.

Tabel perbandingan penelitian terdahulu dengan penelitian yang dilakukan :

Tabel 1.1 Kajian Pustaka

No	Nama penulis, Tahun, dan Judul	Persamaan	Perbedaan	
			Penelitian Terdahulu	Rencana Penelitian
1	Alexandru Lavric. 2014." Street Lighting Control System Based On Large-Scale WSN: A Step Towards A Smart City	Wireless Sensor Network sebagai implementasi dan pengujian	Pengujian dan implementasi basis skala besar menggunakan wireless sensor network untuk membangun smart city	Implementasi wireless sensor network basis skala kecil dengan menggunakan simulator OMNET++ sebagai objek performa routing flooding dan convergecast
2	Raju Dutta.2013." <i>Comparison of Flooding and Gossiping Routing Protocols Using TOSSIM in Wireless Sensor Networks</i>	Melakukan penerapan komparasi protokol routing pada wireless sensor network	Mengkomparasi antara routing flooding dan gossiping pada Wireless Sensor Network menggunakan TOSSIM	Analisis hasil pengujian performa routing flooding dan convergecast pada wireless sensor network menggunakan MiXiM framework
3	Loverpreet Kaur and Jyoteesh Malhotra."Comp arasion of Wise Route and flooding Network Type of Convergecast Routing in WSN". International Journal of Sensor and Its	Pemakaian algoritma routing protocol Convergecast pada WSN	Komparasi jenis koneksi protocol Convergecast dan Broadcasting dalam WSN	Mensimulasikan atau mereplikasikan protocol Convergecast pada WSN diperlukan software OMNET selaku simulasi



	Applications for Control System, vol.3,no.2 (2015), pp.1-10			
4	Zhijing Qin, Di Wu.2018."Modeling and Analysis of Data Aggregation From Convergecast in Mobile Sensor Networks for Industrial IoT	Penggunaan convergecast dan penerapan pada node sensor	Bereksperimen dengan epoch model untuk menganalisis waktu ketersediaan link pada routing convergecast di jaringan sensor Mobile	Mensimulasikan protocol convergecast pada WSN dengan memerlukan software OMNET selaku simulasi
5	Dong-Won Kum.2010. "An Efficient On-Demand Routing Approach with Directional Flooding for Wireless Mesh Networks	Melakukan penerapan kinerja routing flooding pada node sensor	Mengajukan jenis pengembangan protokol routing DF (directional) sebagai efesensi on demand routing untuk Wireless Mesh Network	Mengimplementasikan protokol routing flooding pada WSN menggunakan software OMNET simulator untuk melakukan pengujian terhadap sistem

Penelitian sebelumnya dengan judul "Street Lighting Control System Based On Large-Scale WSN: A Step Towards A Smart City", ini menggunakan Wireless Sensor Network berskala besar untuk membangun smart city. Keuntungan utama dari sistem wireless sensor network terdiri dari biaya yang berkaitan dengan konsumsi energi, dan mengurangi intensitas peredupan cahaya dan menurunkan biaya perawatan sebagai hasil dari pemrosesan informasi yang diterima dari sensor, seperti deteksi kendaraan. Sistem pemantauan penerangan jalan memungkinkan tampilan laporan berkala mengenai konsumsi energi dan memberikan perincian tentang pekerjaan pemeliharaan yang dilakukan atau diprogram untuk tanggal selanjutnya (Lavric, 2014),

Penelitian sebelumnya dengan judul "Comparison of Flooding and Gossiping Routing Protocols Using TOSSIM in Wireless Sensor Networks", ini membandingkan dua routing protokol yaitu flooding dan convergecast pada



wireless sensor network dengan menggunakan TOSSIM. Protokol *flooding* dan *gossiping* diimplementasikan di TinyOS1.1.0 dengan beberapa tingkat keberhasilan yang berbeda beda. Parameter yang digunakan pada penelitian ini yaitu dengan mengukur time, energy, packet broadcasting, dan energi consumption. Evaluasi dasar dari protokol ini dilakukan perbandingan fase yang berbeda dan telah dibuat yang ditunjukkan oleh grafik yang berbeda. Setelah simulasi heuristik kedua protokol jelas bahwa keuntungan kinerja dengan *gossiping* lebih baik daripada *flooding*. (Dutta, 2013).

Pada routing protokol *convergecast* dipenelitian sebelumnya yaitu membandingkan routing protokol yang diterapkan pada *Wireless Sensor Network* yang menggunakan sumulator OMNeT++. Protokol routing yang digunakan ada dua yaitu *Flooding routing type convergecast* dan *Wise routing*. Tujuan dari penulis untuk menganalisis hasil dari kedua routing tersebut dengan parameter digunakan *BER (Bit Error Rate)*, *SNIR (signal to noise and interference ratio)*, *Back-off duration*, *SNR (Signal to noise Ratio)* dan *latency* (Malhotra, 2015).

Penelitian lain yang berjudul “Modeling and Analysis of Data Aggregation From Convergecast in Mobile Sensor Networks for Industrial IoT”, ini mengusulkan model durasi jalur waktu baru untuk pengumpulan data menggunakan convergecast dalam jaringan sensor seluler, dan dapat digunakan dalam aplikasi industri internet of things. Hasil teoretis dan eksperimental pada penelitian ini menunjukkan bahwa model mobile convergecast yang diterapkan dapat secara akurat mencerminkan waktu durasi jalur dalam simulasi (Qin, 2018).

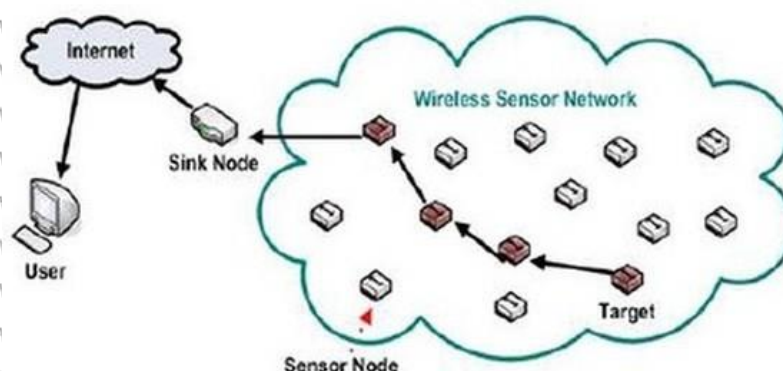
Penelitian selanjutnya dengan judul “An Efficient On-Demand Routing Approach with Directional Flooding for Wireless Mesh Networks”, ini mengusulkan directional flooding pada wireless mesh network untuk mengurangi redudansi dan overhead kontrol massage. Hasil pengujian penelitian ini dalam proses penemuan rute untuk mencapai gateway, skema *directional flooding* dapat mengurangi jumlah paket permintaan rute (RREQ) yang dibroadcast dengan menggunakan teknik *directional flooding*. Hasil simulasi juga menunjukkan bahwa vektor jarak tempuh berdasarkan permintaan ad-hoc dengan *directional flooding* dapat secara signifikan mengurangi biaya overhead perutean dengan paket RREQ dan meningkatkan kinerja keseluruhan dibanding dengan *flooding* atau AODV asli (Kum, 2010).



2.2 Kajian Teori

2.2.1 Wireless Sensor Network

Melalui kontrol dan komunikasi parameter fisik untuk mengontrol kondisi tempat dan berinteraksi dengan lingkungan dan juga terdiri dari sejumlah node sensor individu yang ditempatkan di tempat yang berlainan adalah sifat *Wireless Sensor Network* (WSN) atau sensor nirkabel.



Gambar 2.1 Arsitektur WSN

Gambar 2.1 menunjukkan gambaran WSN, sensor node yang berukuran kecil dapat dikerahkan pada suatu area sensor. Untuk membuat rute data yang dirangkaikan ke node yang berdekatan termasuk salah satu kemampuan sensor node. node sink yang akan dikirim Menuju pada base station transmisi radio dilanjutkan ke layer aplikasi (user) atau mediator antara node sensor dan user data. Beberapa platform seperti koneksi internet dapat diakses sebagai informasi atau melalui remote server yang memungkinkan pengguna dapat mengakses secara real time dengan satelit (Sonavane, 2008).

Untuk mengontrol lingkungan melalui hubungan orang atau peralatan dan lingkungan sekitarnya suatu kemungkinan sebagai jaringan node yang responsif lebih peka, hal tersebut sebagai contoh gambaran umum arsitektur *Wireless Sensor Network* (WSN). Jumlah node sensor *low energy* yang dipasang biasanya dalam pemantauan termasuk bagian jaringan sensor nirkabel. Node ini dapat menuai, memelihara (*maintenance*) dan menjalankan data yang dikumpulkan dari lingkungan sekitar. Terdiri dari jumlah node sensor yang berkolaborasi untuk memantau daerah untuk memperoleh data perihal lingkungan (Sonavane, 2008).

2.2.1.1 Fungsi dari Wireless Sensor Network (WSN)

Wireless Sensor Network (WSN) biasanya digunakan untuk fungsi sebagai berikut :



1. Monitoring

WSN mengukur kuantitas fisik seperti suhu, tekanan, kelembaban, dll, dan mengirimkan data ke konsentrator data. Data konsentrator adalah solusi perangkat lunak atau perangkat keras yang menghubungkan sejumlah saluran data dengan satu *destination*. Data konsentrator bertugas mengelola sumber utama dari berbagai sumber data yang berbeda. Berdasarkan data yang dikumpulkan, maka data dapat ditampilkan secara grafik, diambil keputusan tertentu berdasarkan *event-trigger*.

2. Control

Tergantung pada penggunaan WSN, tetapi pengelolaan kontrol ini jarang dilakukan, dan biasanya memanfaatkan fungsi kontrol dengan skala kecil, WSN biasanya terbatas. Fungsi kontrol dilakukan jika pin output dari mikrokontroler terhubung ke aktuator langsung diobjek sensor (Murdiyat, 2019).

2.2.1.2 Karakteristik dari Wireless Sensor Network (WSN)

Wireless Sensor Network memiliki fitur antara lain :

1. Daya atau power yang dapat disimpan terbatas juga device yang hemat energi
2. Kemampuan untuk menahan kondisi lingkungan yang keras. Kemampuan untuk mempertahankan sensor masih bisa digunakan meskipun kondisi lingkungan yang ekstrim, karena memungkinkan perangkat yang digunakan ditempatkan di daerah yang dapat menjadi ekstrim dalam suhu atau di zona bahaya.
3. Mobilitas node dari topologi jaringan yang dinamis. Sensor / perangkat memungkinkan bergerak pada lokasinya, misalnya untuk memantau posisi sensor ditempatkan di armada truk pengiriman.
4. Dari segi *hardware* (ukuran, sensor, *device* yang digunakan dan lain-lain) maupun *software*, memungkinkan adanya kegagalan komunikasi ataupun kesalahan operasi heterogenetis (keanekaragaman) dari node. Selain itu kemampuan yang dimiliki oleh *device* pun bisa beranekaragam.
5. Jumlah node dalam WSN bisa diperbanyak, yang membatasi jumlahnya adalah *bandwidth* dari gateway (Botan, 2015).

2.2.1.3 Kelebihan dan kekurangan Wireless Sensort Network (WSN)

Kelebihan dari penggunaan WSN ini diantaranya adalah :

- Sangat menguntungkan dalam kondisi geografis karena WSN itu simple/praktis/ringkas karena tidak perlu instalasi kabel yang rumit.



- Sensor yang bersifat mobile, yaitu tanpa harus khawatir tentang mengubah tata letak ruangan serta kabel yang tersedia ruangan pada suatu waktu memungkinkan untuk memindahkan sensor untuk pengukuran yang lebih akurat.

Kekurangan dari WSN diantaranya adalah :

- Data rate menjadi lebih rendah karena beban/load yang lebih tinggi dibandingkan wired sensor (sensor fisik).
- Sisi keamanan menjadi lebih rendah dan transfer rate semakin menurun karena semakin banyak sensor semakin besar *collision domain* (paket-paket data dapat saling bertabrakan/berbenturan).
- Adanya maintenance rutin untuk mengganti baterai yang telah habis (Murdiyat, 2019)

2.2.2 Protokol Routing

Pertukaran informasi tentang layanan adalah tanggung jawab protokol routing untuk menemukan jalan yang mungkin untuk mencapai node tujuan dengan mempertimbangkan panjang lompatan, energi yang diperlukan seminimal mungkin, dan lama jangka hidup dari koneksi nirkabel tersebut menggabungkan informasi tentang jalur yang rusak, *maintenance* jalur yang rusak, dan memfungsikan bandwidth minimum (Mitra, 2016).

Routing adalah informasi gerak algoritma melalui jaringan, dari node awal (sumber) ke node *destination*. Untuk mendefinisikan bagaimana node berhubungan dengan node lainnya merupakan karya komponen penting dalam routing protocol / algoritma routing dan menyebarkan informasi sumber memungkinkan simpul untuk menentukan rute yang optimal ke simpul tujuan pada jaringan komputer. Fungsi dari algoritma routing adalah untuk menghitung rute matematis atau rute yang efisien dan optimal atas informasi routing yang dimiliki oleh sebuah node.

Berikut beberapa hal penting pada protokol routing :

1. Pemilihan jalur terpendek yang akan di tunjukkan ke node *sink* (*destination*) harus efisien.
2. Ketika terjadi perubahan pada topologi harus terus up-to-date tabel routing (Mitra, 2016).

2.2.2.1 Static Routing

Berdasarkan yang telah dipastikan oleh administrator router melanjutkan paket data dari sebuah atau kumpulan jaringan ke jaringan yang lainnya. Kecuali



jika diganti secara manual dengan administrator rute pada *static routing* tidak berubah (Eril, 2012).

Berikut ini adalah karakteristik dari static routing :

1. Toleransi yang tidak ada jika terdapat beberapa kesalahan pada point point sambungan. Tidak akan dihubungkan oleh router jika terdapat perubahan pada jaringan arus lalu lintas paket data atau terdapat kegagalan sambungan antara dua atau lebih point – point yang tersambung secara langsung.
2. Pada komposisi konfigurasi jenis static routing adalah komposisi jaringan yang hanya memiliki beberapa router saja, yang pada umumnya tidak lebih dari dua atau tiga.
3. Oleh administrator ala manual informasi *routing*-nya dibagikan.
4. Hanya satu tabel routing yang dimiliki satu router
5. Biasanya difungsikan untuk jaringan dalam skala kecil dan konstan atau stabil.
 - A. Keuntungan Routing Static :
Karena tidak rentan atas serangan menipu (spoofing) jadi lebih aman dibanding dynamic routing.
 - B. Kelemahan Routing Static
Lebih merepotkan dibandingkan *dynamic routing* dan rentan atas kesalahan penulisan (Eril, 2012).

2.2.2.2 Dynamic Routing

Dynamic Routing adalah routing di mana router bisa menganalisa sendiri jalur atau rute terbaik yang telah terkirim meneruskan paket dari satu jaringan ke yang lain. Administrator tidak memastikan rute yang harus ditangkap pada paket ini. Bagaimana router menganalisis paket lalu router mempelajari sendiri sebagai satu kewajiban administrator. Sesuai dengan analisis yang telah dikaji oleh *router* maka rute pada *dynamic routing* berubah (Mitra, 2016).

Berikut ini adalah karakteristik dari dynamic routing :

1. Informasi routing tidak lagi disediakan secara manual, tetapi dibagikan oleh perangkat lunak.
2. Router secara otomatis akan menyelidiki kompensasi jalan yang tidak bisa digunakan jika salah satu jalur gangguan atau kerusakan peralatan.
3. Memproses jaringan yang lebih rumit dan ukuran atau wilayah yang lebih luas, dan perubahan konfigurasi jaringan yang selalu mengubah jaringan



(*connectionnot-stabil*). Serta jaringan cerdas (sudah menggunakan komputasi).

4. Membutuhkan protokol routing untuk membentuk tabel routing dan routing protokol dapat membuang-buang sumber energi atau daya komputasi.

A. Keuntungan Routing Dynamic :

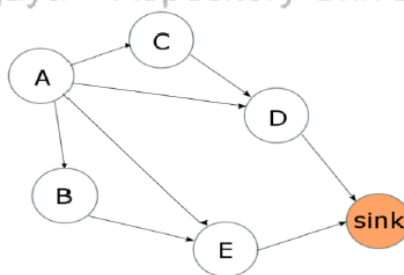
Karena router akan mengambil jalur lain yang siap atau sepiapabila ada suatu jalur yang rusak maka lebih mudah untuk mengatur jaringan.

B. Kelemahan Routing Dynamic :

Memperbarui tabel ARP dan didistribusikan ke semua komputer, berarti bahwa beban kerja menjadi lebih berat karena selalu router update tabel IP pada setiap waktu tertentu (Mitra, 2016).

2.2.3 Routing Flooding

Flooding adalah protokol routing yang menerapkan paket *random broadcast* dalam *networking*, yang membuat arus komunikasi dalam jaringan menjadi tidak teratur. Karena secara drastis mengurangi kinerja jaringan untuk simpul mengkonsumsi daya lebih untuk mentransfer data ke tujuan. *Flooding routing* koneksi termasuk jaringan mendasar yang primitif ke jaringan sensor nirkabel. Pembaruan kode dan transformasi parameter yang menghasilkan operasi dari seluruh node jaringan dapat membroadcast ke semua node (Bae Ko, 2004).



Gambar 2.2 Routing Flooding

Flooding adalah teknik routing sederhana dalam jaringan komputer di mana sumber atau node mengirimkan paket melalui setiap link atau jangkauan yang tersedia. *Flooding*, yang mirip dengan *broadcasting*, terjadi ketika paket sumber (tanpa routing data) yang ditransmisikan ke semua node jaringan terlampir. Karena *flooding* menggunakan setiap jalur dalam jaringan, jalur terpendek juga digunakan. Algoritma banjir mudah diimplementasikan (Lim, 2008).



Data *routing* jaringan awalnya tidak termasuk dalam paket data. Algoritma hop Count digunakan untuk melacak topologi jaringan, atau rute jaringan yang dikunjungi. Paket mencoba untuk mengakses semua rute jaringan yang tersedia dan akhirnya mencapai tujuannya, tetapi selalu ada potensi untuk paket akan terduplikasi. *Hop Count* dan beberapa teknik *flooding* selektif digunakan untuk menghindari keterlambatan komunikasi dan duplikasi. *Flooding* juga digunakan sebagai penyangkalan Layanan serangan oleh *traffic flood* jaringan untuk menurunkan layanan jaringan. Layanan *flooding* dengan banyak permintaan koneksi server yang tidak lengkap, karena jumlah permintaan *flooding*, server atau *host* tidak dapat memproses permintaan asli pada saat yang sama. Sebuah serangan *flood* mengisi server atau *host* memori buffer, setelah penuh, koneksi lebih lanjut tidak dapat dibuat, yang mengakibatkan penyangkalan layanan (Paruchuri, 2011).

Dalam penelitian, akan menunjukkan bahwa operasi penemuan rute dan penerusan dapat digabungkan, dan hasil dari penggabungan ini, dalam hal kinerja end-to-end, tidak perlu lebih rendah daripada skema perutean ad-hoc lainnya, misalnya, berdasarkan penerusan point-to-point. Kami memperkenalkan skema perutean ad-hoc di mana penerusan secara inheren disiarkan berbasis di bahwa node transmisi tidak pernah peduli tentang identitas tetangga berikutnya-hop. Apa yang penting, adalah identitas dari sumber dan tujuan, yaitu, paket yang ditujukan secara eksklusif dalam lapisan transportasi. Skema yang diusulkan adalah perbaikan *flood* langsung dibantu dengan beberapa heuristik yang mengurangi jangkauan ke garis sempit node sepanjang jalur terpendek antara sumber dan tujuan. Lebar garis ini dapat disesuaikan dengan parameter statis dan dinamis yang memperhitungkan pola mobilitas yang diharapkan atau dirasakan, kepadatan simpul, dan kualitas layanan yang diperlukan (QoS) (Paruchuri, 2011).

2.2.3.1 Karakteristik Flooding

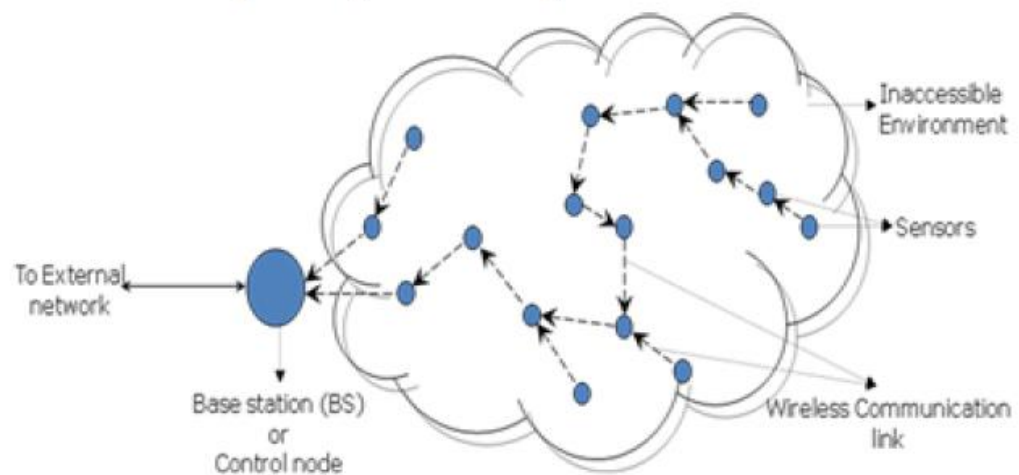
Beberapa karakteristik dari jaringan ini adalah :

1. Semua rute yang mungkin antara sumber dan tujuan dicoba. Paket akan selalu melewati jika ada jalur.
2. Seperti semua rute yang dicoba, akan ada setidaknya satu rute yang merupakan rute terpendek
3. Semua node terhubung langsung atau tidak langsung untuk dikunjungi (Bae Ko, 2004).

2.2.4 Convergecast

Algoritma convergecast adalah kebalikan dari *broadcasting* dalam sistem *message-passing*, bukannya pesan menyebarkan turun dari satu akar ke semua node, data yang dikumpulkan dari node terpencil melalui *tree* yang membentang langsung ke akar. Biasanya beberapa fungsi yang diterapkan ke data masuk di setiap node, dengan tujuan yang akhirnya akar memperoleh fungsi ini dari semua data di seluruh sistem (Veena, 2010).

Banyak aplikasi jaringan sensor nirkabel memerlukan baik *broadcast* dan *convergecasting* untuk penyebaran dan pengumpulan data. *Convergecast* adalah pola komunikasi di mana data mengalir dari banyak node sumber ke simpul wastafel tunggal atau pemancar. Ini adalah protokol *routing* yang berlawanan dengan *broadcast* seperti yang ditunjukkan pada teori sebelumnya. *Broadcast* diikuti dengan *convergecasting* di hampir setiap aplikasi dari WSNs. (Veena, 2010).



Gambar 2.3 Routing Convergecast

Hal ini membuktikan bahwa node memerlukan lebih banyak *power consumption* atau energi untuk mengirimkan data daripada *sensing*. Perhatian utama adalah untuk menghemat daya untuk meningkatkan kehidupan jaringan sensor. Jadi dalam *routing* energi WSN efisiensi sangat penting karena daya terbatas/*Battery*. Protokol *routing* yang digunakan dalam jaringan sensor berbeda dengan protokol *routing* jaringan lainnya. Karena seluruh node sensor adalah perangkat bertenaga baterai, konsumsi energi node. beberapa macam kelemahan convergecast yaitu:

- A. Konsumsi energi. Akan mengubah kinerja network dalam hal routing sebab salah satu node yang mati atau mengalami gangguan dan juga node sensor bawaan mempunyai tenaga baterai yang terbatas. Dan juga



akan menambah biaya serta mengganggu pada kinerja jaringan saat simulasi atau saat pengujian sistem sebab beberapa gangguan yang muncul yaitu menjalani konservasi energy yang berulang atau yang terlalu sering. Karena penggunaan energi adalah faktor yang paling penting dalam menentukan masa jaringan, energi yang digunakan harus efisien untuk menghasilkan performa maksimal.

- B. Terjadinya *latency* tinggi menciptakan kemacetan di jaringan komunikasi. Memperkenalkan mode tidur namun datang pada biaya tertentu. Solusi adalah dengan menggunakan *Time Division Multiple Access* (TDMA), tetapi hal ini membutuhkan node untuk melakukan sinkronisasi dengan satu sama lain cukup ketat, yang dapat menjadi tugas yang cukup kompleks di jaringan besar dengan lokasi node acak dan tidak sempurna (*Drifting*) jam. Membiarkan node mengatur bangun mereka dan tidur kali dalam mode desentralisasi mengurangi kompleksitas ini, tapi ini meningkatkan keterlambatan (juga disebut *latency*) untuk mentransfer informasi antara wastafel dan simpul jauh. Mendorong node pada sleep mode secara independen dari satu sama lain, yang merupakan solusi yang kita adopsi dalam makalah ini, tetapi pada saat yang sama menimbulkan kekhawatiran tentang meningkatkan *latency* jaringan. Lebih penting lagi, itu tidak hanya akan meningkatkan *latency* rata sendiri, tetapi juga akan meningkatkan macam *latency* ini. Untuk beberapa aplikasi, seperti pengumpulan data spasial untuk keperluan statistik, hal ini tentu dapat diterima, tetapi tidak bagi banyak orang lain yang jauh lebih banyak waktu-kritis. Sebuah contoh yang khas dan penting dari skenario seperti ini adalah penggunaan jaringan sensor untuk memantau suatu daerah dan mengirim alarm ketika kejadian abnormal merasakan apa yang terjadi (seperti intrusi, variabel yang berubah dengan cepat, dll) (Upadhyayula, 2003).

2.2.5 Parameter Analisis

2.2.5.1 Activity Durations

Nilai Durasi aktivitas adalah seluruh ukuran lama waktu yang dibutuhkan pada suatu kegiatan sampai selesai. Durasi aktivitas adalah kata benda yang mewakili jumlah waktu tertentu yang ada antara waktu mulai yang ditetapkan dan waktu berakhir dari aktivitas terjadwal. Durasi aktivitas harus diukur dengan cermat dari tanggal mulai hingga tanggal akhir dan dapat dihitung dalam unit kalender apa pun sesuai dengan jangka waktu proyek yang diantisipasi, seperti hari, Minggu, bulan, dll. Durasi kegiatan



dapat dibentuk ketika sebuah proyek dikandung, atau dapat dibentuk ketika sebuah proyek dimulai, dan dapat dimodifikasi jika manfaat situasi dengan kesepakatan semua pihak. Selain itu, jumlah total durasi aktivitas juga dapat dipecah menjadi bagian yang lebih kecil dalam pembentukan tonggak proyek, titik di mana baik keadaan saat ini dari proyek akan dinilai, dan berpotensi jumlah atau semua kiriman akan diserahkan baik dalam bentuk final atau draft (Artigues, 2012)

Pada penelitian ini proyek dengan nilai activity duration digunakan sebagai tahap untuk pengambilan analisis kinerja pada kedua routing dengan bantuan simulator dan sudah di atur dengan konfigurasi yang ditentukan yaitu waktu simulasi, ukuran, banyaknya node dan lain lain.

2.2.5.2 BackOff Durations

Carrier Sense Multiple Access with Collision Detection (CSMA/CD) adalah protokol jaringan untuk transmisi pembawa yang beroperasi di lapisan medium Access Control (MAC). Ini mendeteksi atau mengetahui Apakah saluran transmisi sibuk atau tidak, dan menunda transmisi sampai saluran bebas. Ketika lebih dari satu stasiun mengirim frame atau data mereka secara bersamaan, tabrakan terjadi. Terdapat algoritma sebagai mekanisme resolusi tabrakan yang umumnya digunakan untuk menjadwalkan retransmisi setelah tabrakan di Ethernet. Waktu tunggu Stasiun menunggu sebelum mencoba retransmission frame disebut sebagai BackOff Time (Misra, 2010).

Pada penelitian ini proyek dengan nilai waktu BackOff Durations di implementasikan sebagai tahap untuk pengambilan analisis kinerja pada kedua routing dengan bantuan simulator dan sudah di atur dengan konfigurasi konfigurasi yang ditentukan yaitu waktu simulasi, ukuran, banyaknya node dan lain lain.

2.2.5.3 Latency

Latency adalah interval waktu antara stimulasi dan respon, atau, dari sudut pandang yang lebih umum, penundaan waktu antara penyebab dan efek dari beberapa perubahan fisik dalam sistem yang diamati. Latency secara fisik merupakan konsekuensi dari kecepatan terbatas, besarnya kecepatan ini selalu kurang dari atau sama dengan kecepatan cahaya. Oleh karena itu, setiap sistem fisik dengan pemisahan fisik (jarak) antara sebab dan akibat akan mengalami semacam latensi, terlepas dari sifat stimulasi yang telah terpapar (Eril, 2020).



Definisi yang tepat dari latensi tergantung pada sistem yang sedang diamati dan sifat stimulasi. Dalam komunikasi, batas bawah latensi ditentukan oleh medium yang digunakan untuk komunikasi. Dalam sistem komunikasi dua arah yang andal, latensi membatasi tingkat maksimum informasi yang dapat ditransmisikan, karena sering ada batasan pada jumlah informasi yang "in-flight" pada satu saat. Dalam bidang manusia-mesin interaksi, persepsi latency memiliki efek yang kuat pada kepuasan pengguna dan kegunaan (Eril, 2020).

Game online sensitif terhadap latency atau lag, karena waktu respon yang cepat untuk peristiwa baru yang terjadi selama sesi permainan dihargai sementara waktu respon yang lambat dapat membawa penalti. Karena keterlambatan dalam transmisi gerakan permainan, pemain dengan koneksi internet latensi tinggi mungkin menunjukkan respon yang lambat meskipun waktu reaksi yang tepat. Ini memberi pemain koneksi latensi rendah sebagai keuntungan teknis (Fernando, 2019).

2.2.4.5 Power Consumption

Dalam teknik elektro, konsumsi daya mengacu pada energi listrik per satuan waktu, dipasok untuk mengoperasikan sesuatu, seperti alat rumah. Konsumsi daya biasanya diukur dalam satuan watt (W) atau kilowatt (kW). Energi yang digunakan oleh peralatan selalu lebih dari energi yang sangat dibutuhkan. Hal ini karena tidak ada peralatan yang 100% efisien. Daya yang terbuang sebagai panas, getaran dan/atau radiasi elektromagnetik. Sebagai contoh, bola lampu tidak hanya mengubah tenaga listrik menjadi terang itu juga dapat membuat panas (Tarkoma, 2014).

Power consumption adalah jumlah energi yang digunakan per satuan waktu. Konsumsi daya sangat penting dalam sistem digital. Masa pakai baterai sistem portabel seperti telepon seluler dan komputer laptop dibatasi oleh konsumsi daya. Daya juga signifikan untuk sistem yang dicolokkan, karena listrik biaya uang dan karena sistem akan terlalu panas jika menarik terlalu banyak kekuasaan. Sistem digital menarik daya dinamis dan statis. Daya dinamis adalah daya yang digunakan untuk mengisi kapasitansi sebagai perubahan sinyal antara 0 dan 1.

2.2.6 Omnet++

OMNeT ++ adalah aplikasisimulasi jaringan diskrit yang berorientasi pada objek. Simulator diskritadalah simulator yang bertidak /bereaksi terhadap peristiwa yang terjadi di dalamnya (event).Serangkaian event diskrit merupakan salah satu sifat daripada analitis jaringan komputer. Sistem komputer bekerja



dengan logic C++ dimana pada suatu program terdapat sesi start, sesi proses (pengiriman), dan sesi menutup. OMNeT++ merupakan simulator dari sebuah atau beberapa arsitektur jaringan bahwa setiap peristiwa yang terjadi dengan simulator OMNeT++ akan dikaitkan dengan objek tertentu yang berarti berorientasi pada objek (Virdis, 2019).

OMNeT ++ atau bisa disebut omnetpp adalah software simulasi jaringan yang berupa open source (kode sumber terbuka). Ada beberapa sesi pada omnetpp dipasang pada komputer yang pertama sesi simulasi, sesi pengiriman dan sesi penutupan. OMNeT ++ terkait dengan objek tertentu pada setiap simulasi yang terjadi di simulator yang berorientasi pada objek. Untuk simulasi itu sendiri OMNeT ++ juga menyediakan infrastruktur dan kebutuhan lain yang digunakan. OMNeT ++ berorientasi pada objek dan hirarkis. Untuk simulasi pemodelan OMNeT ++ menyediakan komponen arsitektur. Komponen (modul) menggunakan C ++ pemrograman ekstensi bahasa "H" dan "Cc". Karena arsitektur modular, simulasi ini dapat dengan mudah disusun ke dalam sistem karena memiliki fitur OMNeT ++ GUI (Graphical User Interface) yang luas (Virdis, 2019).

OMNeT ++ juga mendukung beberapa kerangka kerja adalah: Inet, Inetmanet, Mixim, Castalia, dan lain-lain. Bagi pengguna yang ingin mengembangkan kerangka simulasi jaringan yang akan membantu pengguna untuk dapat mengembangkan atau membuat simulasi jaringan baru. Kerangka Mixim digunakan untuk flooding routing dan untuk convergecast yang digunakan dalam penelitian ini (Virdis, 2019).

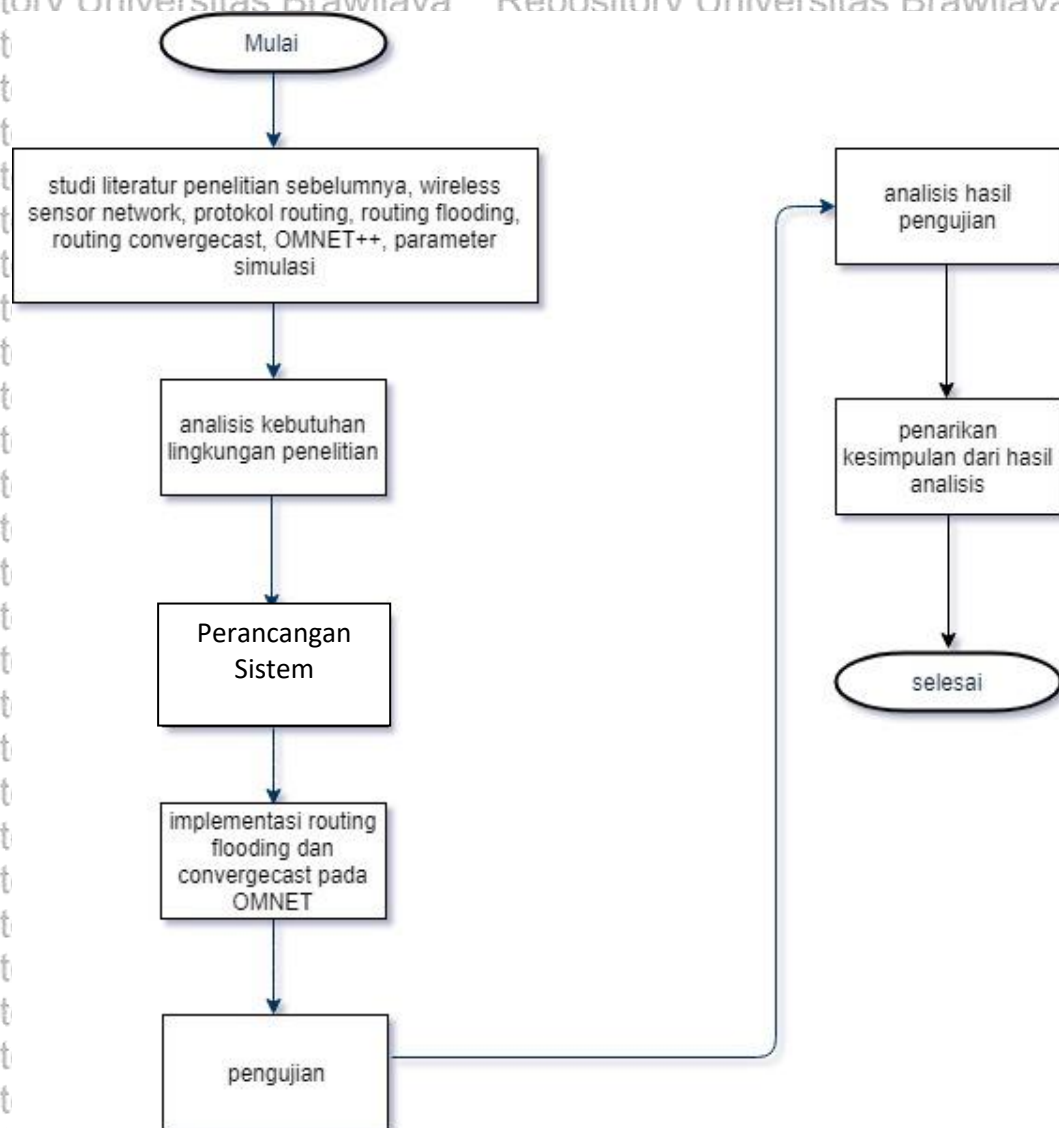
Windows, linux, dan MAC adalah multi platform OS yang dapat berkontributif dengan OMNeT++ karena pada sifat simulator ini adalah open source. Adapun beberapa bagian dari OMNeT++ adalah :

- a) Simulation kernel library
- b) NED (deskripsi topologi)
- c) OMNET++ IDE yaitu Eclipse
- d) GUI untuk simulator yang dieksekusi dengan coman Tkev
- e) Command-line user interface yang menggunakan Cindenv
- f) Utilities seperti make file pada tools
- g) Dokumentasi yaitu contoh simulasi



BAB 3 METODOLOGI

Bab ini menerangkan langkah yang dibuat dalam perancangan atau penyusunan dan pengujian simulasi yang selanjutnya akan dilakukan analisis untuk memperoleh hasil dari penelitian. Langkah pertama adalah untuk mengumpulkan landasan teori yang selanjutnya dijadikan untuk studi literatur dari penelitian ini. Kemudian dilanjutkan dengan perancangan dan penyusunan serta implementasi. Berikutnya, pengujian dan analisis simulasi yang telah dijalankan. Terakhir merupakan penarikan kesimpulan dan pemberian saran untuk penelitian yang akan datang.



Gambar 3.1 Diagram Alir Metodologi



3.1 Studi Literatur

Bagian ini memaparkan landasan teoritis untuk memenuhi analisis kebutuhan yang komprehensif untuk membentuk algoritma *flooding* dan *convergecast* pada jaringan *Wireless Sensor Network* (WSN). Untuk dapat menganalisis dan merancang sistem dibutuhkan teori dasar sebagai kebutuhan dalam penelitian ini, berikut sistem yang meliputi :

1. Teknologi jaringan *Wireless Sensor Network* (WSN)
2. Protokol routing pada *Wireless Sensor Network* (WSN)

- a. Protokol *routing flooding*

Kesemua node tetangganya paket dikirimkan dari node sumber. Kecuali kepada link yang digunakan untuk menerima paket keseluruhan link yang tergabung pada setiap paket yang masuk kembali di transmisikan pada setiap node.

- b. Protokol *routing convergecast*

Untuk disalurkan ke node *sink* dalam waktu yang relatif singkat yang berperan memvisualisasikan aliran data dari banyak node sumber yang berbeda adalah fungsi dari metode komunikasi yang dipakai pada banyak jaringan *Wireless sensor network*.

- c. Untuk memilih prioritas dari pemilihan rute pada suatu *routing* tabel adalah suatu nilai dari kinerja metrik yang meliputi :

1. *Letency* adalah waktu yang diperlukan dari awal transfer data hingga sampai ke server.
2. *Backoff Time* adalah waktu tunggu sebelum paket ditransfer ke *destination*.
3. *Power Consumption* adalah *energy* yang diperlukan pada waktu transfer paket data.
4. *Activity Durations* adalah lama waktu yang diperlukan selama simulasi berjalan

3.2 Analisis Kebutuhan

Analisis kebutuhan ditunjukkan untuk membuat analisis pada beberapa kebutuhan yang diharuskan sistem pada riset ini. Analisis kebutuhan pada riset ini dijelaskan sebagai berikut :



3.2.1 Kebutuhan Fungsional

Analisis kebutuhan fungsional dari sistem yang dibangun bertujuan untuk menentukan aliran informasi yang valid pada sistem untuk memahami alur kerja sistem. Adapun kebutuhan fungsional dari sistem adalah seperti berikut :

- Sistem berhasil mengimplementasikan algoritma *routing* protokol *Flooding* dan *Convergecast* dengan tepat sesuai topologi dan konfigurasi.
- Sistem dapat melakukan pendistribusian traffic berdasarkan *routing flooding* dan *routing convergecast*
- Nilai yang dihasilkan oleh sistem dapat diukur

3.2.2 Kebutuhan Non Fungsional

Analisis kebutuhan non-fungsional diharuskan untuk memastikan detail dari analisis kebutuhan sistem. Sebagai sistem yang hendak dibentuk sampai dengan sistem diimplementasikan yang diperlukan sebagai spesifikasi yang melingkupi komponen – komponen. Untuk sistem ini dibutuhkan *software* yang diuraikan sebagai berikut:

- Windows 7 pada laptop untuk mengoperasikan *software* yang akan digunakan untuk pengimplementasian adalah sistem operasinya.
- Simulator yang diterapkan adalah OMNET++
- Protokol *routing* yang diterapkan ialah protokol *flooding* dan protokol *convergecast*.

3.3 Perancangan Sistem

Bab ini berisi perancangan yang dibentuk dari kebutuhan fungsional dan non fungsional. Perancangan yang dibentuk bertujuan untuk membangun sistem yang akan diteliti. Perancangan yang dilakukan antara lain :

- Perancangan mekanisme *routing flooding*
- Perancangan mekanisme *routing convergecast*
- Perancangan topologi
- Dan perancangan konfigurasi

3.4 Implementasi

Implementasi atau penerapan riset adalah definisi umum dari simulasi desain, studi ini mensimulasikan perbandingan kinerja protokol *routing flooding* dan *convergecast*. simulator perangkat lunak yang diperlukan untuk penerapan dan mensimulasikan desain yang telah didesain adalah OMNeT ++. topologi Node yang akan diperlukan dalam OMNeT ++ ada 5 macam, yaitu dengan

7,14,21,28, dan 35 banyak node. Implementasi ini dibuat dengan menunjuk pada penerapan desain sistem melingkupi :

1. Sesuai dengan *range* yang dipilih, perancangan konfigurasi *routing flooding* dan *convergecast* yang diterapkan pada *Wireless Sensor Network* (WSN) akan dikonsepskan.
2. Untuk menyusun *Wireless Sensor Network* (WSN) perlu sistematika General sebagai fungsi keseluruhan dikedua routing.
3. Sistematika pada penentuan data yang dikupas pada jaringan *Wireless Sensor Network* (WSN).

3.5 Pengujian

Pengujian akan dilakukan untuk menguji kinerja dari metode yang diterapkan. Pengujian dilakukan dengan skenario yang telah ditentukan dengan parameter uji yang telah ditentukan. Pengujian yang akan dilakukan antara lain :

- Pengujian Fungsional
- Pengujian Kinerja

1. Pengujian *routing flooding* dengan parameter yang ditentukan.

2. Pengujian routing Convergecast dengan parameter yang ditentukan.

Pengujian akan dibuat secara bergilir antar *routing*, dengan memerlukan topologi *random*. Pada skenarionya nanti akan dibuat pengujian dengan 7 node sampai 35 node dengan masing-masing protokol, tujuannya untuk mengetahui hasil analisis yang lebih akurat dengan node yang semakin luas. Simulator yang digunakan untuk pengujian ini menggunakan simulator yang telah dikaji pada bab sebelumnya yaitu simulator OMNeT++. Jumlah node yang diuji maksimal 35 node dengan menggunakan algoritma yang dipilih pada setiap nodenya. Untuk komparasi atau perbandingan agregasi rata – rata antara *routing flooding* dan *convergecast* maka diperlukan parameter uji pada pengujiannya. Uraian di bawah ini adalah untuk parameter pengujiannya :

1. Backoff
2. Power Consumption
3. Latency
4. Nb Rx Frame
5. Activity Duration



3.5.1 Skenario Pengujian Pertama dengan Activity Duration

Activity duration adalah waktu atau masa suatu tindakan dari awal sampai yang ditentukan. Parameter pada penelitian ini untuk menganalisis nilai waktu simulasi pada mekanisme *flooding* dan *convergecast*.

3.5.2 Skenario Pengujian Kedua dengan BackOff Durations

Backoff duration adalah waktu tunggu suatu control acces di sebuah terminal jaringan sebelum melakukan pengiriman kembali sebuah frame/paket setelah terjadinya tumbukan. Parameter pada penelitian ini untuk menganalisis nilai waktu backoff saat terjadi tumbukan data pada traffic simulasi pada.

3.5.3 Skenario Pengujian Ketiga dengan Latency

Pada skenario pengujian ketiga akan dilakukan pengujian dengan menerapkan routing *Flooding* dengan 14 node menggunakan simulator Omnet++ dengan skema topologi random dan jenis topologi diatur *stationary* dalam rentang waktu 1 menit dan ukuran area dengan panjang 1000 meter dan lebar 1000 meter. Pada range 1,5 sampai 5 miliwath (mW) akan dikonfigurasi sebagai power yang akan digunakan tiap tiap node untuk pengiriman data.

3.5.4 Skenario Pengujian Kelima dengan Power Consumption

Pada skenario pengujian kelima akan dilakukan pengujian dengan menerapkan routing *Flooding* dan routing *convergecast* dengan 7 sampai 35 node menggunakan simulator Omnet++ dengan skema topologi random dan jenis topologi diatur *stationary* dalam rentang waktu 1 sampai 48 menit dan ukuran area dengan panjang 1000 meter dan lebar 1000 meter. Pada range 1,5 sampai 5 miliwath (mW) akan dikonfigurasi sebagai power yang akan digunakan tiap tiap node untuk pengiriman data.

3.6 Analisis Hasil

Analisis hasil yang telah dilakukan didasarkan pada parameter yang dites, untuk membuat analisis komparatif dari jenis routing protokol *flooding* dan *convergecast* dalam hal node dengan anggapan bahwa setiap indikator yang diperlukan untuk mengumpulkan konsumsi backoff, jumlah energi, *latency* dalam proses transmisi data antara node.

3.7 Kesimpulan

Kesimpulan dan saran menutup tahapan proses penelitian, dari hasil kesimpulan dapat diambil keuntungan dan kerugian dari sistem yang telah diuji, kesesuaian teori dan praktek dan merespon masalah penelitian formulasi telah dilakukan sebelumnya dan dapat disimpulkan tentang skenario yang telah



dibuat. Saran berfungsi untuk memberikan informasi tentang kekurangan dalam penelitian, yang kemudian di perhitungkan untuk penelitian masa depan.

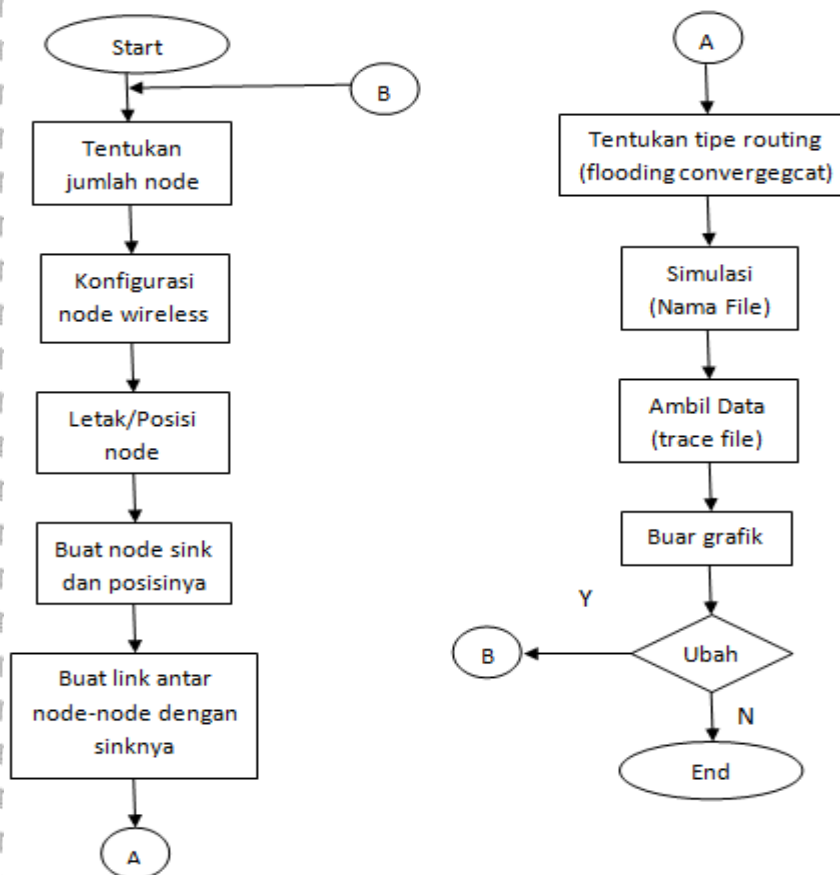


BAB 4 PERANCANGAN DAN IMPLEMENTASI

Bab ini membahas mengenai tahapan perancangan dan hasil implementasi pada simulator Omnet++ terhadap analisis protokol *routing Flooding* dan *Convergecast* yang disesuaikan dengan analisis kebutuhan serta perancangan simulasi. Implementasi dan pengujian mencakup penjelasan tentang spesifikasi sistem konfigurasi simulator dan prosedur simulasi.

4.1 Lingkungan Penelitian

Lingkungan penelitian didasarkan pada perencanaan area simulasi, jumlah *node* sensor, dan sistem komunikasi antar *node* dengan *sink*. Area simulasi ditetapkan sebagai luasan yang didefinisikan sebagai sumbu *x* dan sumbu *y*. Jumlah *node* yang telah ditentukan kemudian diletakkan atau diposisikan dengan aturan yang telah ditentukan, dimana dalam perencanaan ini sensor diposisikan dengan *random position*. Keseluruhan perencanaan parameter-parameter simulasi ini diinisialisasikan pada konfigurasi. Adapun keseluruhan cara kerja dari simulasi ini ditampilkan dengan *flowchart* 4.1 sebagai berikut :



Gambar 4.1 Flowchart Pembuatan Sistem

Pada Gambar 4.1 dapat dijelaskan kembali langkah-lagkah atau proses pembuatan yaitu pada proses awal adalah untuk menentukan jumlah node yang akan digunakan pada simulasi ini, dalam hal ini jumlah node ada 5 macam, kemudian langkah selanjutnya adalah menentukan posisi node yang telah di set pada konfigurasi node wireless. Kemudian proses selanjutnya adalah menentukan sink node, pada setiap node akan menentukan jarak dari node hingga menuju sink-nya. Kemudian user akan menentukan tipe routing yang akan dijalankan yaitu routing *flooding* dan *convergecast*, Ketika simulasi dijalankan di ambil data yang terdapat pada "Trance file" kemudian diambil grafik jika tidak memenuhi proses tersebut maka akan mengecek node-node yang lain.

4.1.1 Analisis Kebutuhan

Dalam rangka meneliti kinerja antara *routing flooding* dan *routing Convergecast* pada *Wireless Sensor Network* menggunakan simulator Omnet++, harus mempunyai kebutuhan yang dapat menunjang terciptanya lingkungan simulasi. Terdapat dua kebutuhan yaitu kebutuhan simulasi dan kebutuhan jaringan



4.1.1.1 Kebutuhan Simulasi

Kebutuhan simulasi yaitu kebutuhan yang digunakan sebagai tempat untuk merancang dan mensimulasikan penelitian ini. Kebutuhan simulasi dibagi menjadi dua yaitu perangkat keras dan perangkat lunak.

1. Kebutuhan Perangkat Keras

Kebutuhan perangkat keras dibutuhkan sebagai dasar utama dari pengembangan sistem ini berupa laptop. Spesifikasi laptop akan dijelaskan pada tabel 4.1

Tabel 4.1 Kebutuhan Perangkat Keras

Komponen	Spesifikasi
Processor	Intel Core i3 – 2365M CPU @ 1.40 Ghz
Memory	4GB
Storage	500 GB Harddisk Drive

2. Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak yang terdapat dalam penelitian ini akan dijelaskan pada tabel 4.2

Tabel 4.2 Kebutuhan Perangkat Lunak

Kebutuhan	Keterangan
Windows 10	Windows 10 merupakan sistem operasi yang terdapat pada laptop
MiXiM	Modeling framework yang dibuat untuk mobile dan jaringan nirkabel (Jaringan sensor nirkabel, Wireless Sensor Network)
Network Simulator Omnet++ 4.2.2	Network Simulator Omnet++ adalah aplikasi/tools yang digunakan untuk melakukan implementasi protokol

4.1.1.2 Kebutuhan Jaringan

1. Kebutuhan Node

Dalam Penelitian ini node yang digunakan terdapat 5 macam node yaitu berjumlah 5, 14, 21, 28 dan 35 pada tiap - tiap skenario.

2. Kebutuhan Node Sink/Node Tujuan



Dalam penelitian ini terdapat 1 node tujuan yang dimaksudkan agar semua node sumber ketika mengirimkan data akan terjadi peningkatan *traffic* data yang terpusat pada antrian yang menuju node tujuan sehingga akan menyebabkan *congestion* pada jaringan

3. Kebutuhan Link

Dalam penelitian ini *link* berperan sebagai kabel (*wired*) yang menghubungkan semua node yang ada dalam topologi jaringan. *Bandwith* yang diterapkan tercantum *default simulator* karena menggunakan *link Stationary*.

4.1.1.3 Kebutuhan Riset

Kebutuhan riset digunakan sebagai parameter pada penelitian dengan nilai yang sudah ditentukan. Beberapa kebutuhan tersebut dapat dilihat pada tabel 4.3

Tabel 4.3 Kebutuhan Riset

Parameter	Keterangan
Routing Protokol	1. <i>Convergecast</i> 2. <i>Flooding</i>
Jumlah Node	7 sampai 35
Ukuran paket data	** <i>.node[*].appl.trafficParam</i> = 30 detik
Model mobilitas	Stationary
Model propagasi	IEEE 802.15.4
Luas Area Jaringan	Y 1000 m dan X 1000 m
Waktu Simulasi	1 sampai 48 menit

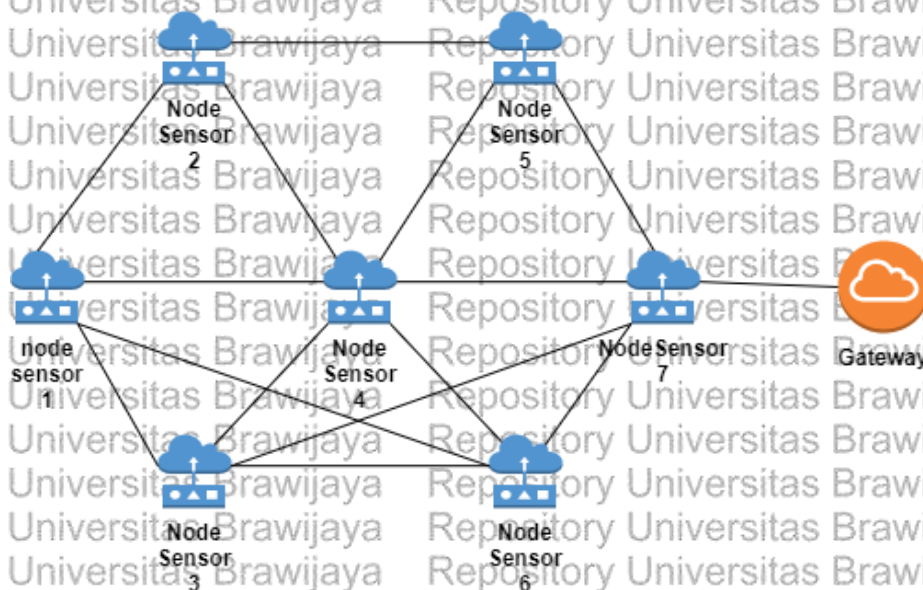
4.2 Perancangan

Pada bab perancangan ini akan dijelaskan mengenai perancangan topologi dan perancangan implementasi, dari mulai berapa topologi untuk pengimplementasian routing *flooding* dan *convergecast*, lalu perancangan implementasi dari nilai nilai tiap variabel, nilai nilai jarak yang ditentukan pada setiap node, dan nilai nilai waktu pengujian yang disiapkan.

4.2.1 Perancangan Topologi

Perancangan topologi yang digunakan untuk pengimplementasian routing *flooding* dan *convergecast* menggunakan 5 macam topologi yaitu topologi pertama menggunakan 7 sampai 35 node beserta sink nodenya, perbedaan

jumlah topologi yang akan dirancang dimaksudkan untuk menganalisis hasil kedua routing apabila kondisi node lebih banyak.

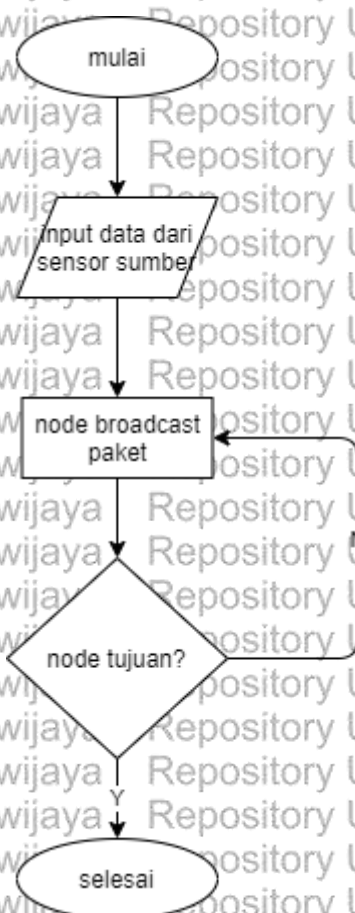


Gambar 4.2 Topologi 7 Node Sensor

Pada gambar 4.2, adalah topologi random yang dibuat di simulasi Omnet++ dari node [0] sampai node [6] yang berjumlah 7 node. Topologi random karena pada penelitian ini menggunakan node device default framework pada MIXIM. Terdiri dari sensor node sebagai node sumber, *connectionManager* dan *world* sebagai bagian dari simulasi jaringan, dan penetapan node [6] sebagai sink node. Tiap *node* akan menggunakan data *device* yang bekerja dengan standard IEEE 802.15.4 yaitu *low-rate wireless personal area networks* (LR-WPANs) yang bekerja pada frekuensi 2400 MHz. (Host802154_2400MHz). Karena *node* menggunakan *device* 802.15.4, maka sifat-sifat *device* tersebut juga akan berlaku, misalnya berkomunikasi menggunakan radio. Karena berkomunikasi menggunakan radio, maka tidak bisa diatur koneksi tiap node dan semua *node* dianggap saling terhubung satu dengan lainnya (stationary). Pengiriman node pada topologi ini saling terhubung serta memiliki kemampuan berkomunikasi dengan perangkat yang dituju dengan cepat.

4.2.2. Perancangan Routing Flooding

Routing flooding digunakan sebagai implementasi objek yang diterapkan pada jaringan wireless sensor network, juga sebagai arah jalur node node sensor mengirimkan paket data. Pada perancangan ini menjelaskan alur kerja *routing flooding* dimana node memiliki paket data yang telah di sensor disebarkan sumber tenaga yang kemudian akan di broadcast kepada node tetangga. Proses broadcast ke node tetangga ini bertujuan agar paket data terkirim sampai sink node (received node) dan gateway.



Gambar 4.3 Flow Diagram Routing Flooding

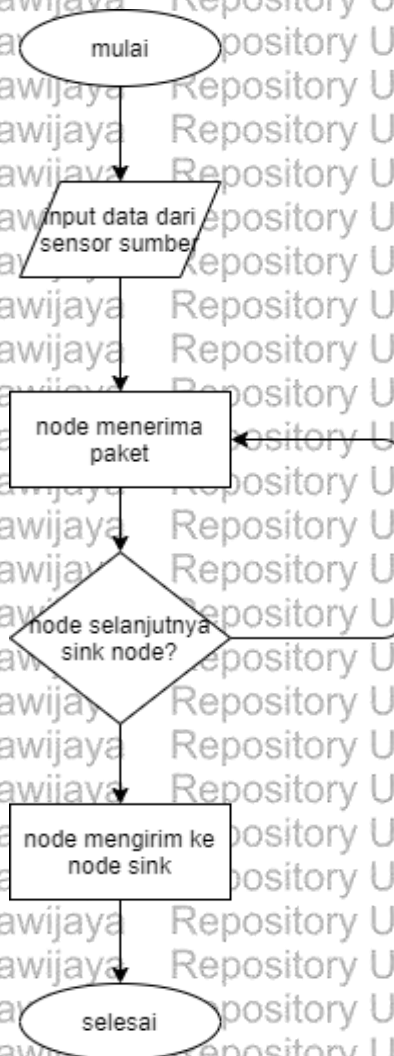
Tahap ini diawali dengan melakukan perancangan topologi dengan jumlah node yang berbeda beda. Hal dilakukan untuk memberikan variasi beban yang berbeda sehingga bisa dilakukan pengukuran hasil yang lebih baik.

Routing flooding ini akan diterapkan pada setiap node yang ada pada skenario pengujian dan diterapkan pada setiap skenario topologi yang sudah disiapkan. Tahap awal sebelum pengiriman data, node sensor sumber mendeteksi adanya energi yang akan di sensor dan di input ke sistem sensor lalu di broadcast ke seluruh node. Setelah proses pengiriman data dimulai, protokol *routing flooding* akan mendeteksi pada tiap tiap node yang terdekat dan terhubung oleh node sumber yang akan mengirimkan data untuk menuju ke sink node seperti gambar 4.4. *Flooding* merupakan *routing* protokol yang menggunakan random packet broadcast didalam sebuah jaringan. *Flooding* digunakan untuk menyebarkan update kode dan perubahan parameter, yang mempengaruhi operasi dari semua node dalam jaringan. Jika perubahan data atau penempatan data terjadi masalah pada salah satu route atau node yang dilewati, *flooding* terus mengirimkannya ke semua node sampai yang dibutuhkan

sudah terkirim. Ini termasuk salah satu kelebihan dari protokol *routing flooding* yang mana data akan sampai ke node sink tanpa request dari node source.

4.2.3 Perancangan Routing Convergecast

Routing convergecast digunakan sebagai implementasi objek yang diterapkan pada jaringan wireless sensor network. juga sebagai arah jalur node node sensor mengirimkan paket data. Pada perancangan ini menjelaskan alur kerja routing convergecast dimana node memiliki paket data yang telah di sensor disebuah sumber tenaga yang kemudian akan di kirimkan kepada node tetangga. Proses pengiriman ke node tetangga ini bertujuan agar paket data terkirim sampai sink node (received node) dan gateway.



Gambar 4.4 Flow Diagram Routing Convergecast

Tahap ini diawali dengan melakukan perancangan topologi dengan jumlah node yang berbeda beda. Hal dilakukan untuk memberikan variasi beban yang berbeda sehingga bisa dilakukan pengukuran hasil yang lebih baik.



Routing convergecast adalah mekanisme pengumpulan data dari satu set sensor menuju ke sebuah sink atau base station umum berdasarkan topologi. Perancangan *routing convergecast* hampir sama dengan *routing flooding*, hanya saja *routing convergecast* mengirimkan paket dari beberapa set sensor ke satu sensor dan dikirimkan lagi sesuai dengan tahap tree sampai menuju ke sink node. Perbedaan dengan *flooding* pada *convergecast* mekanismenya alur paket data menunjukkan dari banyak node ke satu node dengan Wireless Communication link, berbeda halnya dengan *routing flooding* yang mana mekanismenya dari satu ke banyak.

4.2.4 Skenario Pengujian

4.2.4.1 Skenario Pengujian Pertama dengan Activity Duration

Pada skenario pengujian pertama akan dilakukan pengujian dengan menerapkan *routing flooding* dan *routing convergecast* dengan 7 sampai 35 node menggunakan simulator Omnet++ dengan skema topologi random dan jenis topologi diatur *stationary* dalam rentang waktu 1 sampai 48 menit dan ukuran area dengan panjang 1000 meter dan lebar 1000 meter. Pada range 1,5 sampai 5 miliwatt (mW) akan dikonfigurasi sebagai power yang akan digunakan tiap tiap node untuk pengiriman data.

Tabel 4.4 Skenario Pengujian Pertama

Jenis Topologi	Jumlah node	Waktu (menit)	Daya (mW)
Stationary	7	1	1,5
	14	6	1,5
	21	12	5
	28	24	5
	35	48	5

4.2.4.2 Skenario Pengujian Kedua dengan BackOff Durations

Pada skenario pengujian kedua akan dilakukan pengujian dengan menerapkan *routing Convergecast* dan *routing flooding* dengan 7 sampai 35 node menggunakan simulator Omnet++ dengan skema topologi random dan jenis topologi diatur *stationary* dalam rentang waktu 1 sampai 48 menit dan ukuran area dengan panjang 1000 meter dan lebar 1000 meter. Pada range 1,5 sampai 5 miliwatt (mW) akan dikonfigurasi sebagai power yang akan digunakan tiap tiap node untuk pengiriman data.

**Tabel 4.5 Skenario Pengujian Kedua**

Jenis Topologi	Jumlah node	Waktu (menit)	Daya (mW)
Stationary	7	1	1,5
	14	6	1,5
	21	12	5
	28	24	5
	35	48	5

4.2.4.3 Skenario Pengujian Ketiga dengan Latency

Pada skenario pengujian ketiga akan dilakukan pengujian dengan menerapkan routing *Flooding* dengan 14 node menggunakan simulator Omnet++ dengan skema topologi random dan jenis topologi diatur *stationary* dalam rentang waktu 1 menit dan ukuran area dengan panjang 1000 meter dan lebar 1000 meter. Pada range 1,5 sampai 5 miliwath (mW) akan dikonfigurasi sebagai power yang akan digunakan tiap tiap node untuk pengiriman data.

Tabel 4.6 Skenario Pengujian Ketiga

Jenis Topologi	Jumlah node	Waktu (menit)	Daya (mW)
Stationary	7	1	1,5
	14	6	1,5
	21	12	5
	28	24	5
	35	48	5

4.2.4.4 Skenario Pengujian Kelima dengan Power Consumption

Pada skenario pengujian kelima akan dilakukan pengujian dengan menerapkan routing *Flooding* dan routing *convergecast* dengan 7 sampai 35 node menggunakan simulator Omnet++ dengan skema topologi random dan jenis topologi diatur *stationary* dalam rentang waktu 1 sampai 48 menit dan ukuran area dengan panjang 1000 meter dan lebar 1000 meter. Pada range 1,5 sampai 5 miliwath (mW) akan dikonfigurasi sebagai power yang akan digunakan tiap tiap node untuk pengiriman data.

**Tabel 4.7 Skenario Pengujian Keempat**

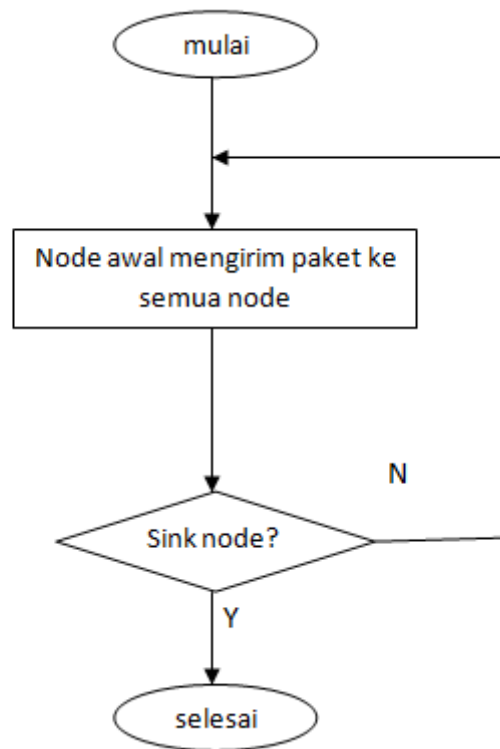
Jenis Topologi	Jumlah node	Waktu (menit)	Daya (mW)
Stationary	7	1	1,5
	14	1,6	1,5
	21	12	5
	28	24	5
	35	48	5

4.3 Implementasi

Pada sub bab ini akan dijelaskan secara detail tentang implementasi. Tahap implementasi dan dilanjutkan dengan konfigurasi script simulasi yang bertujuan untuk membangun lingkungan simulasi.

4.3.1 Desain implementasi

Flow diagram pada gambar 4.6 menunjukkan Implementasi penerapan arah node dari node satu ke node lainnya menuju node terakhir atau sink node. pada awalnya node transfer broadcast paket ke seluruh node. lalu semua node menerima paket dari node transfer awal. berikutnya sistem mengacak untuk memilih node yang mengirim atau meneruskan paket dari node sebelumnya begitu terus sampai semua node berstatus transfer, jika pemilihan random berada di sink node atau di node received terakhir maka proses selesai.



Gambar 4.5 Flow Diagram Desain Implementasi

4.3.2 Alur Kerja Simulasi

Alur Kerja yang terjadi mulai awal menentukan *routing* dan manajemen konfigurasi lalu proses pengiriman data sampai terjadi analisis hasil yaitu sebagai berikut :

1. Menentukan *routing* yang akan diuji, pada penelitian ini menggunakan *routing flooding* atau *routing convergecast*. kedua *routing* tersebut akan digunakan secara bergantian setiap kali pengujian
2. Menentukan jumlah topologi yang akan diuji, pada penelitian ini menggunakan 5 topologi dengan jumlah yang berbeda untuk tujuan mengetahui hasil analisis yang tepat. Kedua topologi tersebut akan digunakan secara bergantian setiap kali pengujian
3. Menentukan *trafficparam*, yaitu waktu pengalihan setiap node mengirim 1 packet setiap 30 detik
4. Selanjutnya pengiriman data dilakukan, dimana setiap node sumber akan mengirim paket data secara bertahap dengan selang waktu per 30 detik dengan lama waktu simulasi yaitu 60 detik.
5. Selanjutnya semua paket data dari setiap node sumber tersebut akan melewati beberapa node untuk diteruskan ke node tujuan atau disebut node sink



6. Selanjutnya ketika semua node sudah mengirimkan paket data secara berkala maka akan terjadi peningkatan *traffic* data sehingga terjadi penumpukan data pada delay yang menyebabkan salah satu node terjadi drop.
7. Selanjutnya setelah semua node sudah selesai mengirim ke node sink simulasi selesai dan dapat dilihat analisis dari tiap tiap routing dan setiap topologi.

4.3.3 Konfigurasi Script Simulasi

Pada sub bab konfigurasi script simulasi ini, ada tahapan tahapan yang perlu dilakukan agar simulasi dapat berjalan pada omnet++, yaitu sebagai berikut :

1. Konfigurasi topologi dengan 7 Node Sensor

Tabel 4.8 Konfigurasi Topologi 7 Node Sensor

Default setting.txt konfigurasi penempatan dengan 7 node	
1	<code>**.mobilityType = "StationaryMobility"</code>
2	<code>**.node[*].mobility.initFromDisplayString = false</code>
3	<code>**.node[*].mobility.initialZ = 0m</code>
4	<code>**.node[0].mobility.initialX = 250m</code>
5	<code>**.node[0].mobility.initialY = 70m</code>

Pada gambar diatas adalah konfigurasi topologi pertama dengan Omnet++, dijelaskan di node [0] sampai node [6] dengan sumbu X dan Y yang berbeda, koordinat Z yang tidak digunakan diset bernilai 0 pada semua node, seperti yang dijelaskan sebelumnya bahwa perancangan topologi yang akan diimplementasikan adalah topologi random. Jenis topologi diatur *stationary*, yaitu tiap node akan bersifat statis dan tidak berpindah lokasi.

2. Konfigurasi topologi II

Tabel 4.9 Konfigurasi Topologi 14 Node Sensor

Default setting.txt : konfigurasi penempatan dengan 14 node	
1	<code>**mobilityType = "StationaryMobility"</code>
2	<code>**node[*].mobility.initFromDisplayString = false</code>
3	<code>**node[*].mobility.initialZ = 0m</code>
4	<code>**node[0].mobility.initialX = 250m</code>
5	<code>**node[0].mobility.initialY = 70m</code>

Pada source code konfigurasi topologi dengan 14 node sensor adalah konfigurasi topologi Kedua dengan Omnet++, dijelaskan di node [0] sampai node [13] dengan sumbu X dan Y yang berbeda, koordinat Z yang tidak digunakan diset bernilai 0 pada semua node, seperti yang dijelaskan sebelumnya bahwa perancangan topologi yang akan diimplementasikan adalah topologi random. Jenis topologi diatur *stationary*, yaitu tiap node akan bersifat statis dan tidak berpindah lokasi, untuk penjelasan pada tiap tiap node sama seperti topologi pertama.

3. Mendefinisikan general submodules simulator dan pemberian numhost yang sudah tersimpan pada package MiXiM

Tabel 4.10 Konfigurasi General SubModul

Default setting.txt : konfigurasi general sub modul	
1	<code>#general sub modul</code>
2	<code>Package org.mixim.examples.WSNRouting;</code>
3	<code>Node[numHosts]:Host802154_2400MHz</code>

Pada konfigurasi sub modul diatas Template *network WSNRouting* menggunakan konfigurasi yang digunakan juga pada *BaseNetwork*. Lalu ada parameter *numHosts* yang digunakan untuk mengatur jumlah host yang ada di dalam *network*. Tiap node akan menggunakan data *device* yang bekerja dengan standard IEEE 802.15.4 yaitu *low-rate wireless personal area networks* (LR-WPANs) yang sudah dijelaskan pada sub bab sebelumnya.



4. Mendefinisikan durasi simulasi maksimal dikelas General untuk kedua *routing* yang diatur selama 1 menit

Tabel 4.11 Konfigurasi General sim-time-limit

Default setting.txt : konfigurasi general sim-time-limit	
1	#general sim-time-limit
2	Sim-time-limit = \${simDuration=1} min

Pada konfigurasi general sim-time-limit diatas menggunakan konfigurasi template yang ada pada setting package WSN. lalu ada parameter 'sim-time-limit' yang digunakan untuk mengatur waktu simulasi yang akan dijalankan pada saat pengujian *routing flooding* dan *routing convergecast* pada wireless sensor network. Tiap routing akan menggunakan waktu konfigurasi general yang sama agar analisa hasil lebih maksimal.

5. Konfiigurasi Ukuran area dan maksimal jumlah Host

Konfigurasi ukuran area digunakan untuk membuat ukuran area simulasi yang akan digunakan untuk kedua routing pada network simulator omnet++.

Tabel 4.12 Konfigurasi Ukuran Area& jumlah host

Default setting.txt : ukuran area & jumlah host	
1	#general ukuran area & jumlah host
2	**playgroundSizeX = 1000 m
3	**playgroundSizeY = 1000 m
4	**playgroundSizeZ = 0 m
5	**numHosts = \${numHosts=14,50}

Pada konfigurasi ukuran area dan jumlah host diatas menggunakan konfigurasi yang ada pada package mixim Wireless sensor network. Lalu ada parameter 'playgroundX' , Y, Z, dan numHost yang digunakan untuk mengatur ukuran area yang akan digunakan dalam simulasi pada routing flooding dan routing convergecast dengan wireless sensor network serta jumlah maksimal host yang akan ditetapkan pada simulasi. Tiap routing menggunakan ukuran dan maksimal host yang sama yaitu 50 maksimal.



6. Konfigurasi Routing Flooding

Tabel 4.13 Konfigurasi Flooding

Default_setting.txt : konfigurasi routing flooding	
1	#routing flooding
2	**node[*].networkType = "Flood"
3	**node[*].netw1.stats = true
4	**node[*].netw1.headerLength = 24 bit

Pada konfigurasi flooding diatas untuk simulasi menggunakan *flooding*, maka config yang dijalankan adalah config *flooding*. Dalam config ini *networkType* diatur menjadi *flood*. Lalu ada beberapa parameter penting yang berfokus pada routing flooding yaitu tipe jaringan protokol yaitu 'flood' yang berarti protokol flooding yang digunakan dengan inisialisasi panjang header 24 bit.

7. Konfigurasi Routing Convergecast

Tabel 4.14 Konfigurasi Convergecast

Default_setting.txt : konfigurasi routing convergecast	
1	#routing convergecast
2	**node[*].networkType = "WiseRoute"
3	**node[*].netw1.stats = true
4	**node[*].netw1.routeFloodsInterval = 1200 s

Pada konfigurasi convergecast diatas untuk simulasi pada *convergecast*, maka config yang dijalankan adalah config *convergecast*. Dalam config ini *networkType* diatur menjadi *Wise route* yang berarti algoritma routing bebas loop sederhana yang membangun pohon routing dari titik jaringan pusat, yang dirancang untuk jaringan sensor dan convergecast traffic. Lalu ada beberapa parameter penting yang berfokus pada routing convergecast yaitu pada tipe jaringan protokolnya 'wiseroute' yang berarti protokol convergecast traffic yang digunakan dengan inisialisasi selang route 1200 detik.



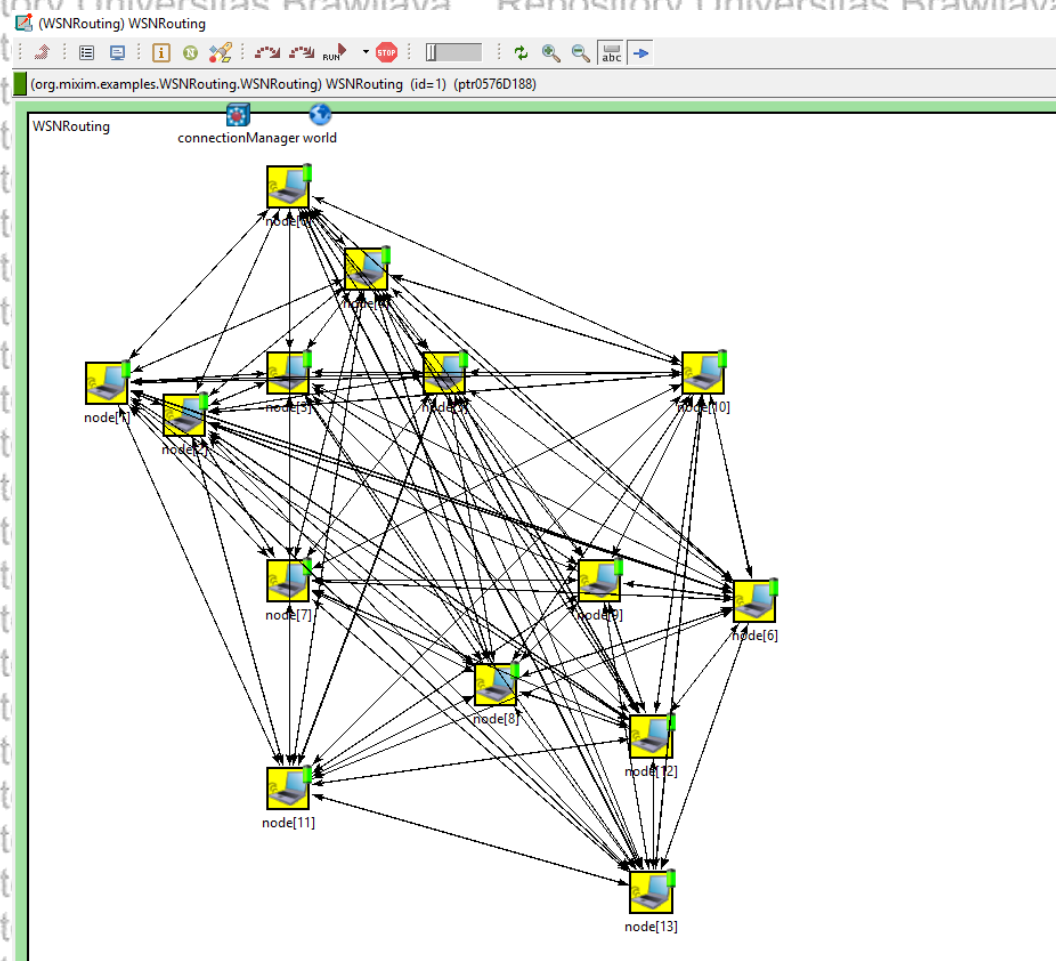
```
in encaps...
Info removed, netw addr=-1
netw 0 sending packet
sendDown: nHop=L3BROADCAST -> message has to be broadcasted -> set destMac=L2BROADCAST
pkt encapsulated
** Event #4 T=36.464405069127 WSNRouting.node[0].nic.mac (CSMA802154, id=26), on 'Data' (NetwPkt, id=127)
** Event #5 T=36.466645069127 WSNRouting.node[0].nic.mac (CSMA802154, id=26), on selfmsg 'timer-backoff' (cMessage, id=0)
** Event #6 T=36.468565069127 WSNRouting.node[0].nic.mac (CSMA802154, id=26), on selfmsg 'timer-cca' (cMessage, id=1)
** Event #7 T=36.468757069127 WSNRouting.node[0].nic.phy (PhyLayerBattery, id=27), on selfmsg 'radio switching over' (cMessage, id=4)
** Event #8 T=36.468757069127 WSNRouting.node[0].nic.phy (PhyLayerBattery, id=27), on 'Data' (MacPkt, id=129)
** Event #9 T=36.468757069127 WSNRouting.node[0].nic.mac (CSMA802154, id=26), on 'Radio switching over' (cMessage, id=130)
** Event #10 T=36.468757069127 WSNRouting.node[1].nic.phy (PhyLayerBattery, id=37), on 'Data' (MiximAirFrame, id=132)
** Event #11 T=36.468757069127 WSNRouting.node[2].nic.phy (PhyLayerBattery, id=47), on 'Data' (MiximAirFrame, id=133)
** Event #12 T=36.468757069127 WSNRouting.node[3].nic.phy (PhyLayerBattery, id=57), on 'Data' (MiximAirFrame, id=134)
** Event #13 T=36.468757069127 WSNRouting.node[4].nic.phy (PhyLayerBattery, id=67), on 'Data' (MiximAirFrame, id=135)
** Event #14 T=36.468757069127 WSNRouting.node[5].nic.phy (PhyLayerBattery, id=77), on 'Data' (MiximAirFrame, id=136)
** Event #15 T=36.468757069127 WSNRouting.node[6].nic.phy (PhyLayerBattery, id=87), on 'Data' (MiximAirFrame, id=137)
** Event #16 T=36.468757069127 WSNRouting.node[7].nic.phy (PhyLayerBattery, id=97), on 'Data' (MiximAirFrame, id=138)
```

Gambar 4.6 Aliran Pengiriman Data

Gambar 4.13 menjelaskan aliran data dari dimulai dari node 0 sebagai node awal menuju ke node selanjutnya/tatangga dengan cara broadcast yaitu mengirim ke semua node yang tersambung dengan node 0 sebagai pangalir data menuju node sink.

BAB 5 PENGUJIAN DAN ANALISIS HASIL

5.1 Skenario Pengujian



Gambar 5.1 Pengujian pada Simulator

Gambar 5.1 menunjukkan penerapan pengujian dengan 14 node pada network simulator, pada penjelasan node sebelumnya bahwa node bersifat stationary dimana semua node terhubung dan sifat dari kedua routing yang sama (broadcasting).

Pengujian dilakukan secara berkala dengan setiap macam node dari mulai 7 sampai 35 node. pada gambar 5.1 diambil contoh dengan 14 node dengan node transfer node ke [0] dan sink node ke [13]. Paket dikirimkan dengan broadcast di kedua routing, proses broadcast terus akan berjalan sesuai waktu dan penerimaan sampai ke node sink yang ditentukan pada setiap macam node. selanjutnya setelah semua node sudah selesai mengirim ke node sink simulasi selesai dan dapat dilihat hasil dari setiap routing.



5.2.1 Skenario Pengujian Pertama dengan nilai waktu Activity Durations

Pada skenario pengujian pertama akan dilakukan pengujian dengan menerapkan routing flooding dan routing convergecast dengan 7 sampai 35 node menggunakan simulator Omnet++ dengan skema topologi random dan jenis topologi diatur *stationary* dalam rentang waktu 1 sampai 48 menit dan ukuran area dengan panjang 1000 meter dan lebar 1000 meter. Power yang digunakan *node* untuk mengirim data dibatasi pada range 1,5 sampai 5 mW.

5.2.2 Skenario Pengujian Kedua dengan nilai waktu BackOff Durations

Pada skenario pengujian kedua akan dilakukan pengujian dengan menerapkan routing *Convergecast* dan routing flooding dengan 7 sampai 35 node menggunakan simulator Omnet++ dengan skema topologi random dan jenis topologi diatur *stationary* dalam rentang waktu 1 sampai 48 menit dan ukuran area dengan panjang 1000 meter dan lebar 1000 meter. Power yang digunakan *node* untuk mengirim data dibatasi pada range 1,5 sampai 5 mW.

5.2.3 Skenario Pengujian Ketiga dengan nilai waktu Latency

Pada skenario pengujian ketiga akan dilakukan pengujian dengan menerapkan routing *Flooding* dan routing convergecast dengan 7 sampai 35 node menggunakan simulator Omnet++ dengan skema topologi random dan jenis topologi diatur *stationary* dalam rentang waktu 1 sampai 48 menit dan ukuran area dengan panjang 1000 meter dan lebar 1000 meter. Power yang digunakan *node* untuk mengirim data dibatasi pada range 1,5 sampai 5 mW.

5.2.4 Skenario Pengujian keempat dengan nilai waktu Power Consumption

Pada skenario pengujian kelima akan dilakukan pengujian dengan menerapkan routing *flooding* dan routing convergecast dengan 7 sampai 35 node menggunakan simulator Omnet++ dengan skema topologi random dan jenis topologi diatur *stationary* dalam rentang waktu 1 sampai 48 menit, pada tiap skenario terdapat penambahan waktu simulasi karena proses pengiriman data yang lebih banyak sebab node yang bertambah banyak dan ukuran area dengan panjang 1000 meter dan lebar 1000 meter. Power yang digunakan *node* untuk mengirim data dibatasi pada range 1,5 sampai 5 mW.



5.2 Hasil Pengujian

5.2.1 Hasil Pengujian Skenario Pertama (*Activity Durations*)

5.2.1.1 Tujuan

Tujuan umum dalam penelitian simulasi ini adalah sebagai yang sudah dijelaskan di bab pertama yaitu mengetahui lingkungan apa saja yang korelatif terkait penelitian simulasi pada kedua routing, mengetahui perancangan pada tiap tiap routing protokol, dan juga mencari tau kinerja dan performansi dari kedua protokol routing serta diambil hipotesis baru. Untuk tujuan khusus pada hasil pengujian ini adalah sebagai menunjukkan hasil data rata-rata pada nilai waktu activity duration yang sudah disimulasikan pada implementasi pengujian dan juga hasil dari simulasi yang sudah diimplementasikan.

5.2.1.2 Prosedur

1. Menjalankan simulator OMNeT++ versi 4.2.2
2. Menetapkan konfigurasi jumlah node yang akan disimulasikan
3. Menetapkan konfigurasi ukuran simulator, waktu simulasi, dan konfigurasi sistem lainnya.
4. Pada file "omnetpp.ned" dijalankan dengan "run" konfigurasi flooding
5. Setelah proses run selesai, ambil data hasil simulasi dengan sesuai parameter yang ditentukan.
6. Setelah hasil waktu keluar lalu di *Average* (nilai rata-rata)

5.2.1.3 Hasil Pengujian

Pada skenario pertama merupakan pengujian simulasi dari routing protokol flooding dan routing convergecast dengan 5 macam node, pada penjelasan sebelumnya dikatakan bahwa power yang digunakan pada konfigurasi pengujian disetting 1,5 sampai 5 miliwatt, pada skenario pertama dilakukan dengan 1,5 watt dan ukuran simulasi 1000m X 1000m. Hasil pada tabel 5.1 adalah rata-rata dari setiap pernode, untuk hasil detail pernode dapat dilihat pada lampiran 3.1.

Tabel 5.1 nilai hasil pengujian

Flooding		Convergecast	
Nodes	Activity Dur (ms)	Nodes	Activity Dur (ms)
7	-0,27872	7	-0,85536
14	221,3732	14	-0,92515
21	496,7097	21	20,47677
28	941,7853	28	32,96598
35	1430,823	35	13,22263

Tabel 5.1 menunjukan nilai rata-rata pengujian dari routing flooding dan convergecast pada wireless sensor network dengan nilai waktu Activity Durations yang dihasilkan dari pengujian pertama yang terdiri dari 7 nodes komponen yang pada pengujian menggunakan waktu 1 menit dan power 1,5 mW.

/MiXiM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[0].appl	activity duration	-1.0	
/MiXiM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[1].appl	activity duration	0.014144	
/MiXiM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[2].appl	activity duration	0.009856	
/MiXiM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[3].appl	activity duration	0.009856	
/MiXiM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[4].appl	activity duration	0.005248	
/MiXiM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[5].appl	activity duration	0.009856	
/MiXiM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[6].appl	activity duration	-1.0	

Gambar 5.2 Hasil Pengujian

Pada gambar 5.1 menunjukan bukti hasil dari pengujian dengan skenario pertama yaitu dengan 7 nodes.

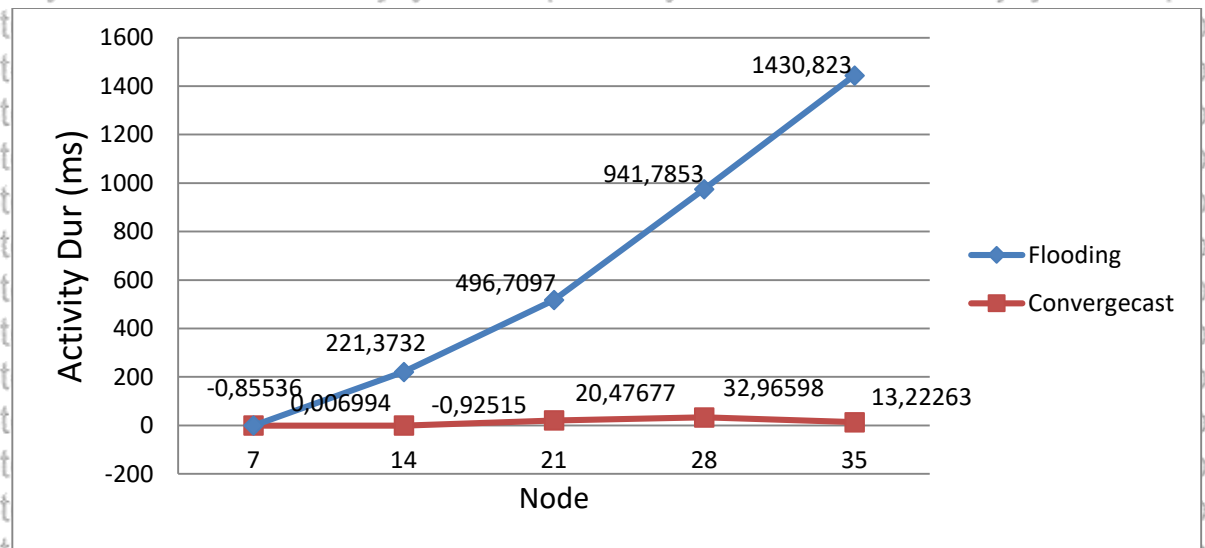
5.2.1.4 Analisis Hasil Pengujian

Perbandingan paket activity duration antara routing protokol flooding dan convergecast pada wireless sensor network dengan penambahan kapasitas node yaitu mulai 7 node paket ditampilkan dengan grafik yang dapat dilihat pada gambar 5.2 berikut dengan penjelasan sebagai berikut :

Routing protokol flooding ditunjukan pada grafik dengan simbol garis berwarna biru menunjukan garis yang cenderung naik, dimulai dari nilai 0 s dengan kapasitas 7 nodes, kemudian grafik naik ke angka 200 s ketika node dinaikan ke node 14 lalu selanjutnya sampai node ke 35 nilai waktu pada routing flooding naik sampai 1430 s mengalami penaikan angka semakn banyak node namun semakin naik angkanya.



Routing protokol convergecast ditunjukkan pada grafik dengan simbol garis berwarna merah menunjukkan garis yang cenderung jauh berbeda dari routing flooding, dimulai dari nilai 0 s dengan kapasitas 7 nodes, kemudian grafik tidak berubah pada node ke 14 dan naik di node berikutnya ketika node ke 21, hal ini dapat disebabkan karena konfigurasi node ke 21 berbeda dari sebelumnya yaitu penambahan waktu dan power pengujian dikarenakan tidak cukupnya lingkungan yang sama di node 21, 28, dan 35. lalu di node 28 mengalami kenaikan lagi tetapi di node 35 mengalami penurunan kembali.



Gambar 5.3 Grafik Hasil Kedua Routing

Dari dua hasil tersebut, penambahan kapasitas nodes mempengaruhi peningkatan kinerja di sisi packet activity duration pada routing flooding karena pada routing flooding kenaikannya stabil dan terus meningkat semakin banyaknya node, tetapi tidak berpengaruh pada routing convergecast dimana kenaikan grafik tidak stabil secara signifikan.

Hasil tersebut dapat dilihat bahwa saat menggunakan packet activity duration kedua routing mendapatkan kinerja yang sangat berbeda, sesuai dengan teori sebelumnya bahwasannya sifat dari convergecast yang menyimpan satu set data di setiap node sampai ke node sink dapat dibuktikan pada pengujian ini, bahwasannya pada grafik atau hasil dari routing convergecast waktu yang dibutuhkan tidak terlalu banyak, bahkan saat node berjumlah 35 waktu lebih menurun atau bisa disebut lebih cepat dan efisien sampai ke node sink. Maka dapat dibuktikan pula bahwa dari sifat convergecast dapat berhasil meningkatkan waktu pengiriman packet data sampai ke sink node.



5.2.2 Hasil Pengujian Skenario Kedua (*BackOff Durations*)

5.2.2.1 Tujuan

Tujuan umum dalam penelitian simulasi ini adalah sebagai yang sudah dijelaskan di bab pertama yaitu mengetahui lingkungan apa saja yang korelatif terkait penelitian simulasi pada kedua routing, mengetahui perancangan pada tiap tiap routing protokol, dan juga mencari tau kinerja dan performansi dari kedua protokol routing serta diambil hipotesis baru. Untuk tujuan khusus pada hasil pengujian ini adalah sebagai menunjukkan hasil data rata-rata dari nilai waktu *BackOff Durations* yang sudah disimulasikan pada implementasi pengujian dan juga hasil dari simulasi yang sudah diimplementasikan.

5.2.2.2 Prosedur

1. Menjalankan simulator OMNeT++ versi 4.2.2
2. Menetapkan konfigurasi jumlah node yang akan disimulasikan
3. Menetapkan konfigurasi ukuran simulator, waktu simulasi, dan konfigurasi sistem lainnya.
4. Pada file "omnetpp.ned" dijalankan dengan "run" konfigurasi Convergecast
5. Setelah proses run selesai, ambil data hasil simulasi dengan sesuai parameter yang ditentukan.
6. Setelah hasil waktu keluar lalu di *Average* (nilai rata-rata)

5.2.2.3 Hasil Pengujian

Pada skenario kedua merupakan pengujian simulasi dari routing protokol convergecast dan routing flooding dengan 5 macam node, pada penjelasan sebelumnya dikatakan bahwa power yang digunakan pada konfigurasi pengujian disetting 1,5 sampai 5 miliwatt, pada skenario kedua dilakukan dengan 1,5 watt dan ukuran simulasi 1000m X 1000m. Hasil pada tabel 5.2 adalah rata-rata dari setiap pernode, untuk hasil detail pernode dapat dilihat pada lampiran 3.2 .

Tabel 5.2 nilai hasil pengujian

Flooding		Convergecast	
Nodes	BackOff Dur (ms)	Nodes	BackOff Dur (ms)
7	3,001573	7	1,289334
14	0,228131	14	1,294271
21	1,548024	21	0,054467
28	2,403805	28	0,170325
35	3,277143	35	3,146869



Tabel 5.2 menunjukkan nilai rata rata hasil pengujian dari routing Convergecast dan routing flooding pada wireless sensor network dengan nilai waktu Backoff Duration yang dihasilkan dari pengujian kedua yang terdiri dari 5 macam node komponen yang pada pengujian menggunakan waktu 1 menit dan power 1,5 mW.

/MixIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[0].nic...	backoffDurati...	9.6E-4	
/MixIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[1].nic...	backoffDurati...	9.6E-4	
/MixIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[2].nic...	backoffDurati...	0.0	
/MixIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[3].nic...	backoffDurati...	0.00352	
/MixIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[4].nic...	backoffDurati...	3.2E-4	
/MixIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[5].nic...	backoffDurati...	0.00736	
/MixIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[6].nic...	backoffDurati...	0.0	

Gambar 5.4 Hasil Pengujian

Pada gambar 5.2 menunjukkan bukti hasil dari pengujian dengan skenario kedua yaitu dengan 7 nodes.

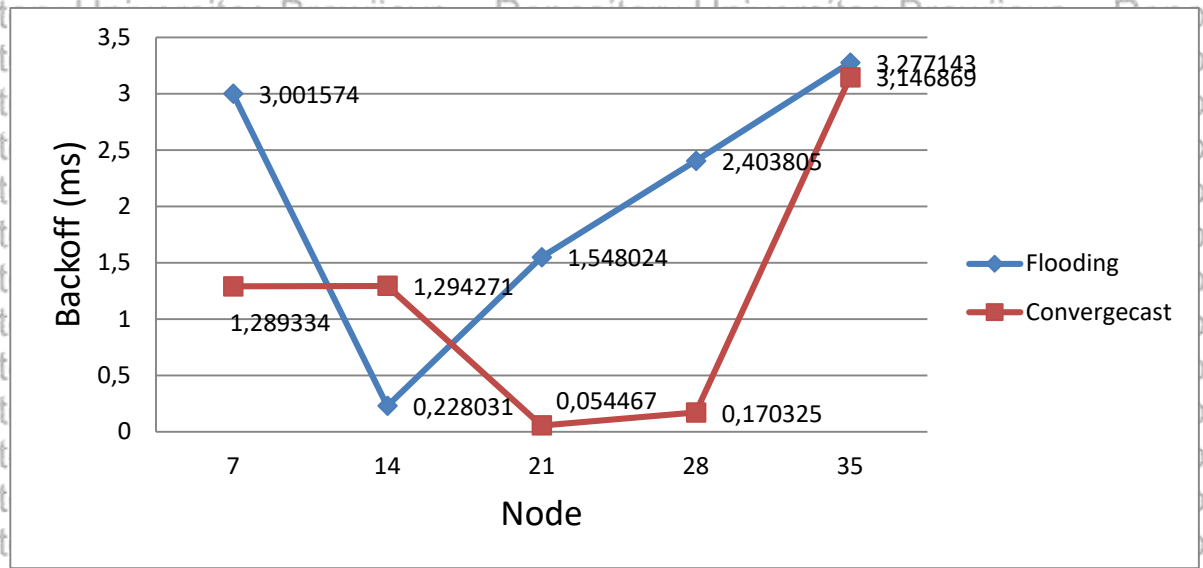
5.2.2.4 Analisis Hasil Pengujian

Perbandingan nilai backoff antara routing protokol flooding dan routing convergecast pada wireless sensor network dengan penambahan kapasitas node yaitu mulai dari 7 nodes hingga 35 nodes ditampilkan dengan grafik yang dapat dilihat pada gambar berikut dengan penjelasan sebagai berikut :

Routing flooding yang ditunjukan pada grafik dengan simbol garis berwarna biru menunjukkan grafik yang cenderung tidak stabil. Dimulai dari 7 node bernilai waktu 3 ms lalu mengalami penurunan di node 14 dan mengalami kenaikan di node 21 dan seterusnya.

Routing convergecast yang ditunjukan pada grafik dengan simbol berwarna merah menunjukkan grafik yang cenderung tidak stabil juga, hanya saja mengalami sedikit perbedaan hasil dari routing flooding Dimulai dari 7 node bernilai waktu 1,2 ms lalu mengalami penurunan waktu sampai 21 node, dan naik pesat sampai node ke 35.

Dari dua hasil tersebut, penambahan kapastias nodes mempengaruhi peningkatan kinerja disisi nilai backoff pada kedua routing karena kedua routing sama sama mengalami perubahan waktu sejalan dengan penambahan node, tetapi perubahan nilai pada parameter di kedua routing tidak stabil dengan kapasitas yang dimana node semakin bertambah.



Gambar 5.5 Grafik Hasil Kedua Routing

Hasil tersebut dapat dilihat bahwa saat menggunakan nilai backoff time kedua routing sama sama mengalami peningkatan tetapi tidak stabil. Pada kedua routing mengalami penurunan yang cukup signifikan terlebih dahulu sebelum sama sama mengalami kenaikan yang cukup signifikan juga. Pada parameter backoff Durations atau bisa disebut Backoff Periode sendiri definisinya yaitu pemeriksaan jalur/transmisi sebelum paket dikirimkan dari node satu ke node lain sampai menuju ke jalur sink node. Atau bisa disebut jeda waktu pencarian jalur sampai jalur paket pada node yang akan mengirim itu sudah jelas (tidak ada node lain yang transmisi pada jalur yang sama). Analisis yang dapat diambil bahwa routing convergecast lebih unggul dari pada flooding, terlihat pada grafik dan hasil pada node berjumlah 7, 21, 28 dan 35 itu lebih baik dari routing flooding.

5.2.3 Hasil Pengujian Skenario Ketiga (Latency)

5.2.3.1 Tujuan

Tujuan umum dalam penelitian simulasi ini adalah sebagai yang sudah dijelaskan di bab pertama yaitu mengetahui lingkungan apa saja yang korelatif terkait penelitian simulasi pada kedua routing, mengetahui perancangan pada tiap tiap routing protokol, dan juga mencari tau kinerja dan performansi dari kedua protokol routing serta diambil hipotesis baru. Untuk tujuan khusus pada hasil pengujian ini adalah sebagai menunjukkan hasil data rata-rata pada nilai Latency yang sudah disimulasikan pada implementasi pengujian dan juga hasil dari simulasi yang sudah diimplementasikan.

5.2.3.2 Prosedur

1. Menjalankan simulator OMNeT++ versi 4.2.2



2. Menetapkan konfigurasi jumlah node yang akan disimulasikan
3. Menetapkan konfigurasi ukuran simulator, waktu simulasi, dan konfigurasi sistem lainnya.
4. Pada file "omnetpp.ned" dijalankan dengan "run" konfigurasi Flooding
5. Setelah proses run selesai, ambil data hasil simulasi dengan sesuai parameter yang ditentukan.
6. Setelah hasil waktu keluar lalu di *Average* (nilai rata-rata)

5.2.3.3 Hasil Pengujian

Pada skenario ketiga merupakan pengujian simulasi dari routing protokol Flooding dengan 5 macam node, pada penjelasan sebelumnya dikatakan bahwa power yang digunakan pada konfigurasi pengujian disetting 1,5 sampai 5 miliwatt, pada skenario kedua dilakukan dengan 5 watt dan ukuran simulasi 1000m X 1000m. Hasil pada tabel 5.3 adalah rata-rata dari setiap pernode, untuk hasil detail pernode dapat dilihat pada lampiran 3.3 .

Tabel 5.3 nilai hasil pengujian

<i>Flooding</i>		<i>Convergecast</i>	
<i>Nodes</i>	<i>Latency (ms)</i>	<i>Nodes</i>	<i>Latency (ms)</i>
7	0,006993	7	0,001783
14	0,020052	14	0,003417
21	0,014662	21	0,000352
28	0,019124	28	0,001479
35	0,022462	35	0,00088

Tabel 5.3 menunjukan nilai rata-rata pengujian dari routing flooding dan routing convergecast pada wireless sensor network dengan nilai waktu latency yang dihasilkan dari pengujian ketiga yang terdiri dari 5 macam node komponen yang pada pengujian menggunakan waktu 6 menit dan power 5 mW.



/MiXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[0].appl	latency:mean	0.0
/MiXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[1].appl	latency:mean	0.014144
/MiXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[2].appl	latency:mean	0.009856
/MiXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[3].appl	latency:mean	0.009856
/MiXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[4].appl	latency:mean	0.005248
/MiXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[5].appl	latency:mean	0.009856
/MiXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[6].appl	latency:mean	0.0

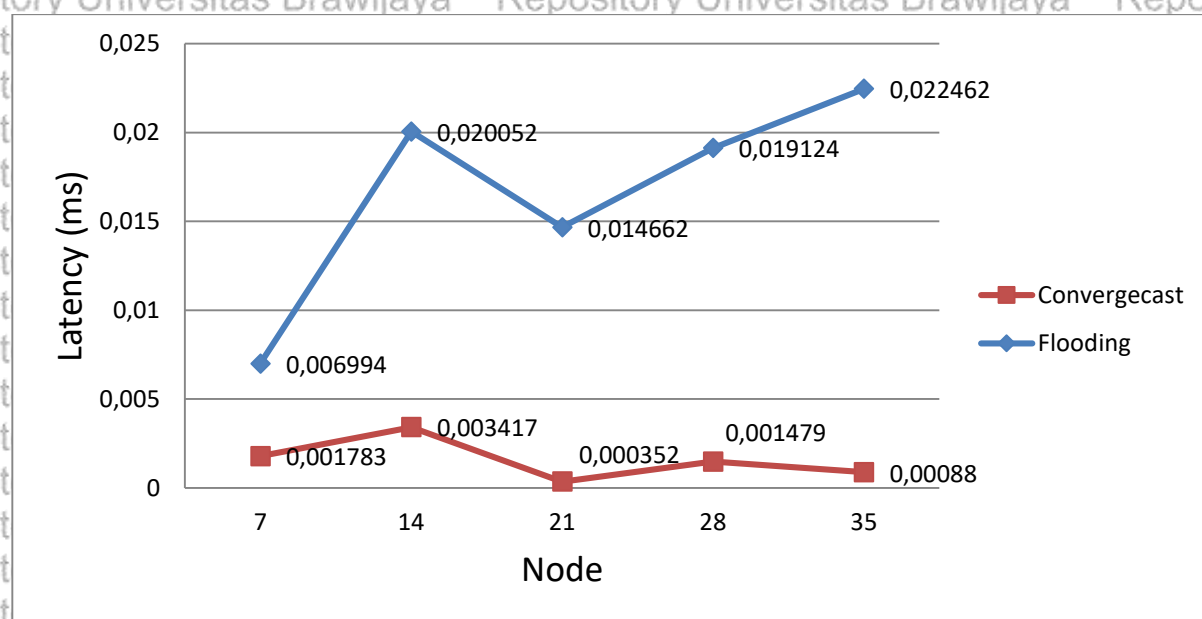
Gambar 5.6 Hasil Pengujian

Pada gambar 5.3 menunjukan bukti hasil dari pengujian dengan skenario ketiga yaitu dengan 14 nodes.

5.2.3.4 Analisis Hasil Pengujian

Perbandingan nilai latency antara routing protokol flooding dan routing convergecast pada wireless sensor network dengan penambahan kapasitas node yaitu mulai dari 7 nodes hingga 35 nodes ditampilkan dengan grafik yang dapat dilihat pada gambar berikut dengan penjelasan sebagai berikut:

Routing protokol flooding ditunjukan pada grafik dengan simbol garis berwarna biru menunjukan garis yang cenderung naik. dimulai dari nilai 0,006994 ms dengan kapasitas 7 nodes, kemudian grafik naik terus, tetapi di node ke 21 mengalami penurunan 0,5 ms lalu naik terus sampai node ke 35.



Gambar 5.7 Grafik Hasil Kedua Routing

Routing protokol convergecast ditunjukan pada grafik dengan simbol garis berwarna merah menunjukan garis yang cenderung jauh lebih rendah dari pada routing flooding, dimulai dari node 7 dengan waktu 0,0017 ms tidak jauh berbeda dari routing flooding dengan waktu 0,006 ms, tetapi pada routing ini tetap stabil untuk penurunan dan kenaikan waktu pada grafik tersebut. Dapat



dilihat pada akhir node 35 di node terbanyak dengan waktunya lebih kecil yaitu dengan waktu 0,0008 ms.

Dari dua hasil tersebut, penambahan kapasitas nodes mempengaruhi peningkatan dan penurunan kinerja disisi nilai latency pada kedua routing. Karena semakin besar kapasitas node, semakin banyak pula data yang dilayani, pada grafik tersebut terdapat perbedaan yang signifikan dari kedua routing.

Hasil tersebut dapat dilihat bahwa saat menggunakan nilai latency kedua routing sama sama mengalami perubahan waktu sejalan dengan bertambahnya node. Pada routing flooding mempunyai waktu yang lebih tinggi dari pada routing convergecast, walaupun di awal node 7 perbedaan waktu tidak begitu jauh, tapi seiring dengan bertambahnya node nilai waktu dari kedua routing sangat berbeda. Latency sendiri adalah interval waktu antara stimulasi dan respon atau dapat diambil pengertian yang lebih umum yaitu penundaan waktu antara penyebab dan efek dari beberapa perubahan fisik dalam sistem. Kaitannya antara teori dan hasil pengujian dengan parameter latency, pada kedua routing ini mempunyai nilai waktu interval atau latency yang sangat kecil atau bisa dibilang dalam pengiriman paket ke setiap node sampai ke node sink tidak mengalami gangguan atau efek negatif pada paket sehingga waktu interval atau penundaan waktu pengiriman paket sangat kecil. Untuk perbandingan dari kedua routing ini dapat dilihat routing convergecast lebih sedikit waktunya dalam artian lebih baik kinerja latency dari pada flooding. Maka dari itu penyebab dari nilai latency yang lebih kecil bisa disebabkan karena keunggulan dari sifat routing convergecast yang dijelaskan sebelumnya.

5.2.4 Hasil Pengujian Skenario Kelima (*Power Consumption*)

5.2.4.1 Tujuan

Tujuan umum dalam penelitian simulasi ini adalah sebagai yang sudah dijelaskan di bab pertama yaitu mengetahui lingkungan apa saja yang korelatif terkait penelitian simulasi pada kedua routing, mengetahui perancangan pada tiap tiap routing protokol, dan juga mencari tau kinerja dan performansi dari kedua protokol routing serta diambil hipotesis baru. Untuk tujuan khusus pada hasil pengujian ini adalah sebagai menunjukan hasil data rata-rata nilai waktu power consumption yang sudah disimulasikan pada implementasi pengujian dan juga hasil dari simulasi yang sudah diimplementasikan.

5.2.4.2 Prosedur

1. Menjalankan simulator OMNeT++ versi 4.2.2
2. Menetapkan konfigurasi jumlah node yang akan disimulasikan



3. Menetapkan konfigurasi ukuran simulator, waktu simulasi, dan konfigurasi sistem lainnya.
4. Pada file "omnetpp.ned" dijalankan dengan "run" konfigurasi Flooding
5. Setelah proses run selesai, ambil data hasil simulasi dengan sesuai parameter yang ditentukan.
6. Setelah hasil waktu keluar lalu di *Average* (nilai rata-rata)

5.2.4.3 Hasil Pengujian

Pada skenario kelima merupakan pengujian simulasi dari routing protokol Flooding dan routing convergecast dengan 5 macam node, pada penjelasan sebelumnya dikatakan bahwa power yang digunakan pada konfigurasi pengujian disetting 1,5 sampai 5 miliwattdan ukuran simulasi 1000m X 1000m serta waktu 1 sampai 48 menit simulasi. Hasil pada tabel 5.4 adalah rata-rata dari setiap pernode, untuk hasil detail pernode dapat dilihat pada lampiran 3.4 .

Tabel 5.4 nilai hasil pengujian

<i>Flooding</i>		<i>Convergecast</i>	
<i>Nodes</i>	<i>Power Consum (mW)</i>	<i>Nodes</i>	<i>Power Consum (mW)</i>
7	56,39986	7	56,39971
14	56,39987	14	56,39988
21	56,39771	21	56,39983
28	56,39838	28	56,39981
35	56,39899	35	56,3989

Tabel 5.5 menunjukan nilai rata-rata pengujian dari routing Flooding dan routing convergecast pada wireless sensor network dengan nilai waktu power consumption yang dihasilkan dari pengujian kelima yang terdiri dari 5 macam nodes komponen yang pada pengujian menggunakan waktu 1 sampai 48 menit dan power 1,5 mW sampai 5 mW. Karena pada node ke 21 dibutuhkan waktu dan tenaga yang lebih banyak dari pengujian sebelumnya. Pengujian ini dilakukan dua kali sebelum diambil hasil sebenarnya, pada pengujian pertama digunakan waktu dan tenaga yang sama, tetapi mendapatkan hasil yang tidak maksimal dalam arti proses pengiriman paket tidak sampe selesai menuju node sink.



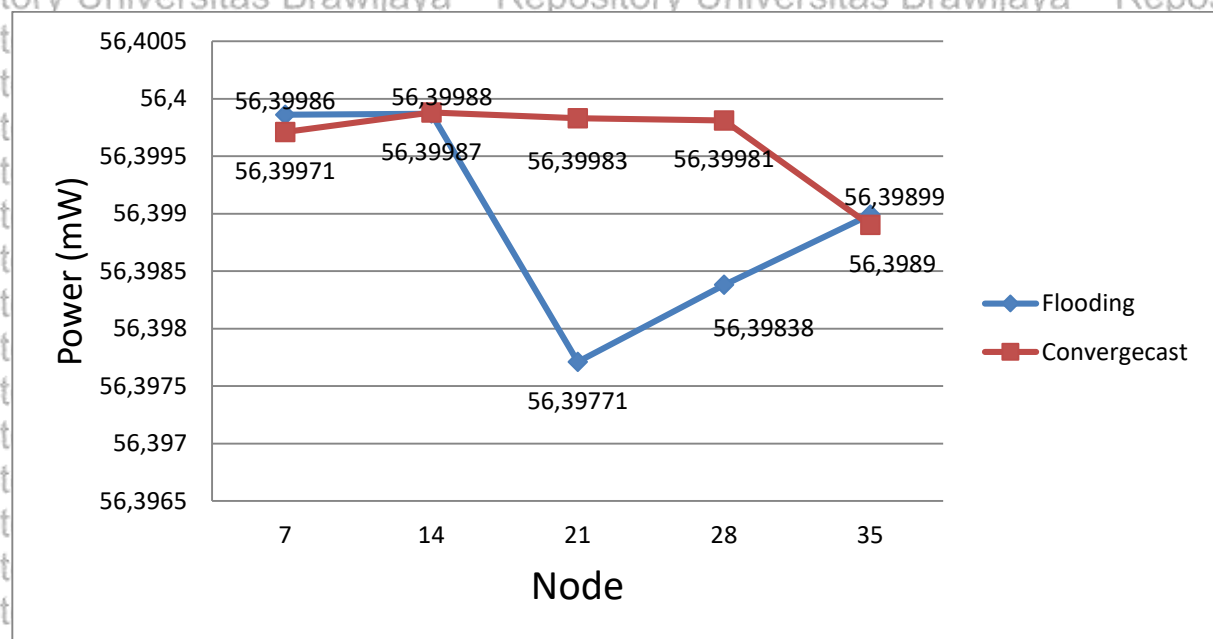
/MIXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[0].batt...	Mean power c...	56.3998476...
/MIXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[1].batt...	Mean power c...	56.3998476...
/MIXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[2].batt...	Mean power c...	56.3998476...
/MIXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[3].batt...	Mean power c...	56.3998476...
/MIXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[4].batt...	Mean power c...	56.3998476...
/MIXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[5].batt...	Mean power c...	56.3998476...
/MIXIM...	flooding-BC-1...	flooding-BC	1	flooding-BC-1-20200203-13:25:55-11708	WSNRouting.node[6].batt...	Mean power c...	56.4

Gambar 5.8 hasil dari rata-rata pengujian

Pada gambar 5.5 menunjukan bukti hasil dari pengujian dengan skenario kelima yaitu dengan 5 macam nodes.

5.2.4.4 Analisis Hasil Pengujian

Perbandingan nilai power consumption antara routing protokol flooding dan routing convergecast pada wireless sensor network dengan penambahan kapasitas node yaitu mulai dari 7 nodes hingga 35 nodes ditampilkan dengan grafik yang dapat dilihat pada gambar berikut dengan penjelasan sebagai berikut:



Gambar 5.9 Grafik Hasil Kedua Routing

Routing flooding yang ditunjukan pada grafik dengan simbol garis berwarna biru menunjukan grafik yang cenderung menurun. Dimulai dari nilai 56,3998 mW dengan kapasitas 7 nodes, kemudian di node 14 nilai tetap sama, di node 21 nilai power pada flooding menurun menjadi 56,3977 mW lalu di node 28 dan 35 mengalami sedikit kenaikan sampai 56,3989 mW

Routing convergecast yang ditunjukan pada grafik dengan simbol berwarna merah menunjukan grafik yang cenderung menurun sedikit sampai node terakhir. Dimulai dari 56,39971 mW dengan kapasitas 7 nodes, kemudian grafik



tetap stabil sampai node 28, dan di node terakhir yaitu node 35 mengalami penurunan dengan power 56,3989.

Dari dua hasil tersebut, penambahan kapasitas nodes mempengaruhi peningkatan kinerja, pada grafik menunjukkan tiap tiap node mengalami penurunan power sampai dengan 35 node. Menunjukkan bahwa routing ini mampu menampung power yang tidak begitu besar saat node bertambah banyak. Dari grafik diatas bahwa nilai power yang baik terlihat ada di routing flooding.

Hasil tersebut dapat dilihat bahwa saat menggunakan nilai power consumption kedua routing sama sama mengalami penurunan power yang berarti kinerja keduanya baik saat penambahan node. Karena routing flooding dalam mengirimkan paket data tidak berhenti pada setiap node, teori flooding langsung menyebarkan terus ke nodes tetangga terdekat hingga node sink, tidak peduli data rusak atau data yang hilang. Berbeda dengan routing convergecast yang mempunyai sifat dan teori pengumpulan data dari satu set sensor menuju ke sebuah sink atau base station umum jadi rentan waktu yang dibutuhkan dan nilai umpan balik atau backoff cenderung lebih baik dibandingkan routing flooding, sedangkan routing flooding dalam mengirimkan paket data secara cepat sehingga bisa terjadi pengumpulan data pada salah satu set data sensor. Namun ketika semua sumber mengirimkan paket data secara bersamaan akan terjadi lonjakan traffic data, dimana mekanisme ini akan menampung paket data yang datang hingga kapasitas nodes penuh. Jika kapasitas nodes penuh, paket data yang datang akan di drop sehingga mengakibatkan congestion atau kehilangan data pada routing flooding akan berkurang setengah dan kecepatan pengiriman akan kembali dari awal sehingga kepadatan antrian akan berkurang alasannya meningkatnya nilai backoff pada routing flooding ketika menggunakan penambahan kapasitas nodes, semakin bertambah pula data yang ditampung untuk dilayani. Dan meskipun akhirnya terjadi packet drop namun akan dilanjutkan ke fase fast recovery untuk mempertahankan nilai backoff tetap naik. Hal tersebut membuat nilai backoff routing flooding lebih baik daripada routing convergecast saat menggunakan nilai backoff. Pada grafik ini menunjukkan bahwa teori dan hasil sejalan dan sebanding karena pada routing convergecast sebenarnya memiliki power consumption yang lebih banyak karena sifat convergecast yang menyimpan satu set data sensor di setiap nodenya. Dalam hal ini routing flooding lebih baik karena power yang digunakan cenderung lebih rendah dibanding routing convergecast.



BAB 6 PENUTUP

6.1 Kesimpulan

Pada penelitian pengujian kinerja pada routing convergecast dan routing flooding pada wireless sensor network menggunakan simulator OMNet++ mendapatkan hasil pengujian dengan parameter nilai Activity Duration, nilai Backoff, Latency, Nb Rx Frames serta power consumption pada kedua routing, dan dapat diambil kesimpulan sebagai berikut :

1. Penerapan mekanisme routing flooding dan routing convergecast pada wireless sensor network sesuai dengan skenario pengujian mendapat hasil yang jelas. Dari pengujian yang terdapat pada skenario dapat dibuktikan bahwa konfigurasi algoritma routing convergecast dan routing flooding berjalan sesuai perancangan serta mendapatkan hasil analisis yang berbeda dari setiap penambahan node, serta kedua routing dapat mendistribusikan paket secara benar berdasarkan node transfer sampai sink node. Paket dapat terdistribusikan tanpa mengalami kesalahan.
2. Pada protokol routing flooding lebih tinggi hasilnya dari pada routing convergecast, hal disebabkan karena routing flooding terdapat nilai backoff dan latency yang cukup tinggi dari routing convergecast yaitu pada nilai latency di 7 nodes convergecast lebih baik 0,0052 ms dari flooding, di 21 node 0,0153 ms dan di 35 node 0,0216 ms lalu pada nilai backoff di 7 nodes convergecast juga lebih baik 1,7122 ms dari routing flooding, di 21 node 1,4936 ms, dan di 35 node 0,1303 ms, hasil tersebut adalah pengurangan dari jumlah rata-rata hasil nilai. Pada power yang tergunakan oleh routing flooding hampir sama dari routing convergecast yaitu 0,0015 mW lebih besar routing convergecast hal ini berbanding lurus dengan teori sebelumnya disebabkan karena routing convergecast mempunyai pola komunikasi penyimpanan satu set data node dari node lainnya yang membutuhkan energi atau kapasitas yang lebih besar dibandingkan routing flooding. Pada nilai parameter activity durations convergecast juga lebih baik dari pada routing flooding yaitu 476,233 ms perbandingannya dari 21 nodes dan 1417,601 ms pada 35 nodes.

6.2 Saran

Berdasarkan penelitian yang sudah dilakukan, peneliti akan memuat sebuah saran dengan harapan dapat digunakan pada penelitian selanjutnya, sebagai berikut :



1. routing dapat menggunakan selain routing flooding dan routing convergecast pada wireless sensor network, seperti routing protokol probabilistik
2. pengujian bisa dilakukan dengan menggunakan parameter uji selain dari, activity duration, latency, backoff, Nb Tx Frames dan Power consumption.
3. skenario pengujian dapat dilakkan dengan skenario skenario lain untuk lebih mendapatkan hasil yang lebih baik
4. simulator yang digunakan dapat simulator simulator lain untuk lebih mendapatkan perbandingan serta parameter yang lebih baik, seperti GNS3, CisCo, Dan lain lain



DAFTAR PUSTAKA

- Anwari, H. (2017). Implementasi Multi-Channel Pada Wireless Sensor Network. S1. Universitas Brawijaya Malang
- Arganata, G. (2017). Implementasi *Wireless Sensor Network* Publisher Menggunakan Protokol Message Queue Telemetry Transport – Sensor Network (MQTT-SN). S1. Universitas Brawijaya Malang
- Artigues, C. (2012). Robust optimization for resource-constrained project scheduling with uncertain activity durations. Leuven : Springer Science and Business Media
- Bae Ko, Y, Mu Choi, J, and Kim, JH. 2004. "A New Directional Flooding Protocol for Wireless Sensor Networks". H-K Kahng and S. Goto (Eds). LNCS 3090, pp 93-102. Berlin : Springer-Verlag
- Dhiman, D. (2000). High Speed LAN Technology Handbook. San Jose : Springer
- Dimas, H. (2016). *Implementasi Low Power Multi Sensor Node Pada Wireless Sensor Network*. S1. Institut Teknologi Bandung
- Eril. 2020. Mengenal Apa Itu Latency, Penyebab dan Cara Menghitungnya. [Internet]. Tersedia di : <https://qwords.com/blog/apa-itu-latency/>. [18 Mei 2020].
- Fernando, R. 2019. Network Latency dalam Online Gaming : Semua yang Kamu Perlu Tahu. [Internet]. Tersedia di : <https://media.skvegrid.id/network-latency-dalam-online-gaming/>. [4 Januari 2019]
- Ngu, A. H., & Gutierrez, M. (2016). IoT Middleware: A Survey on Issues and Enabling Technologies. IEEE Internet of Things Journal.
- Chodorow, K., & Dirolf, M. (2010). *Routing flooding: The Definitive Guide* (1sted.). O'Reilly Media
- Lim, H, and Kim, C. 2000. "Multicast Tree Constuction and Flooding in Wireless Ad Hoc Networks". Department of Computer Science and Engineering, Seoul National University . ACM 2000 1-58113-304-9/00/08.
- Mell, P., & Grance, T. (2011). The NIST Definition of Cloud Computing. Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, United States Department of Commerce.
- Misra, I, S. (1990). *Wireless Communications and Networks 3G and Beyond*. Kolkata : Tata McGraw-Hill Education
- Mitra, D, Sarkar, S, and Hati, D. 2016. "A Comparative Study of Routing Protocols". International Research Journal of Advanced Engineering and Science. Volume 2, Issue 1, pp. 46-50.



Paruchuri. V.K, Mu Duresi. A, Dash. D.S, and Jain. R. 2012. "Optimal Flooding Protocol for Routing in Ad-hoc Networks". Department of Computer and Information Science. LNCS 3090, pp 93-102. Ohio

R. Kumar, J. Vats, and A. Kumar, "A Comparative Study of Routing Protocols," vol. 2, no. 5, pp. 1962–1964, 2011.

Ramakrishnan, S. 2013. Wireless Sensor Networks: From Theory to Applications. London : CRC Press.

Sarma Upadhyayula, Valliappan Annamalai, Sandeep Gupta, A lowlatency and energy-efficient algorithm for convergecast in wireless sensor networks, IEEE Global Communications Conference, 2003.

Seguin, K. (2012). The Little MongoDB Book. San Fransisco: Git Hub Inc.

S. S. Sonavane, V. Kumar and B. P. Patil. Design Factors, Requirements and Research issues of wireless Sensor Networks. International journal of Engineering Research & Industrial Applications (IJERIA), Vol.1, No. III, pp 79-93, 2008.

Sukamaaji, Anjik. 2008. Jaringan Komputer: Konsep Dasar Pengembangan Jaringan & Keamanan Jaringan (Subnet, VLSM, Routing, DES, PGP, & Firewall). Yogyakarta : Andi.

Suakanto, Sinung. 2015. Wireless Sensor Network: Teori dan Praktek Berbasis Open Source. Surabaya : Informatika.

Tarkoma, S. 2014. ENERGY CONSUMPTION Modeling and Optimization. Cambridge : Cambridge University Press

Upadhyayula. S, Annamalai. V, and Gupta. S. K. S. 2003. "A Low-Latency and Energy-Efficient Algorithm for Cnvergecast in Wireless Sensor Networks". Dept. Of Computer Science and Engineering. 0-7801-7974-8. Tempe

Naheman, W., Wei, J (2013). Review of Routing Convergecast and Performance Testing on HBase. IEEE.

Lourenco, Joao Ricardo, dkk. 2015. *No SQL in Practice: A Write-Heavy Enterprise Application*. Coimbra: Portugal

Qing, Xue., & Xiaoming, Ren. 2012. "Research of routing protocols simulation for *wireless Sensor Network* Based on Omnet++". Department of equipment commanding and management Academy of Amored Force Engineering.

Rachmat E., dkk. 2013. Analisis *flooding* Routing dan Probabilistic Broadcast Routing *Wireless Sensor Network* menggunakan simulator Omnet++. Universitas Brawijaya Malang.

Kaur, L., & Jyoteesh, M. "Comparison of Wise Route and *flooding* Network Type of Convergecast Routing in *Wireless Sensor Network*". International jurnal of Sensor and Its Applications for Control System, Vol.3, No.2 (2015), pp.1-10



Veena K.N, and Kumar B.P. 2010. "Convergecast in Wireless Sensor Network: A Neural Network approach". JNN College of Engineering. 978-1-4244-7932-0. Shimoga

Virdis, A. 2019. Recent Advances in Network Simulation: The OMNeT++ Environment and its Ecosystem. Gewerbestrasse : Springer Nature

Wehrle, Klaus. 2010. Modeling and Tools for Network Simulation. Berlin : Springer Science.

Yash Sharma. 2015. What is wireless sensor network (WSN) technology? [Internet]. Tersedia di: <https://www.quora.com/What-is-wireless-sensor-network-WSN-technology>. [26 Oktober 2018].

Yang, Shuang-Hua. 2013. Wireless Sensor Networks: Principles, Design and Applications. Loughborough : Business Media.

Dini. 2015. 8 Kelemahan dan Kelebihan Static Routing. [Internet]. Tersedia di: <https://dosenit.com/jaringan-komputer/teknologi-jaringan/kelemahan-dan-kelebihan-static-routing>. [26 Oktober 2018].

Z. Tang, H. Wang, Q. Hu and L. Hai, "How Network Coding Benefits Converge-Cast in Wireless Sensor Network," IEEE, Dalian, 2012.

<https://omnetpp.org/>
"PENGERTIAN WIRELESS SENSOR NETWORK". wordpress.com. 15 Agustus 2015. 02 Juli 2020. Anak Botan.
<<https://botanmasure.wordpress.com/2015/08/19/pengertian-wireless-sensor-network/>>

Murdiyat, P. 2015. Mengenal Wireless Sensor Network, Solusi Alternatif Mendeteksi Bencana. [Internet]. Tersedia di: <<https://infokomputer.grid.id/read/121280286/mengenal-wireless-sensor-network-solusi-alternatif-mendeteksi-bencana?page=all>>. [02 Juli 2020].

Eril. 2020. Mengenal Perbedaan Routing Statis dan Routing Dinamis. [Internet]. Tersedia di: <[https://qwords.com/blog/perbedaan-routing-statis-dan-dinamis/#:~:text=Routing%20dinamis%20\(Dynamic%20Routing\)%20adalah,semuanya%20harus%20dilakukan%20secara%20manual.>](https://qwords.com/blog/perbedaan-routing-statis-dan-dinamis/#:~:text=Routing%20dinamis%20(Dynamic%20Routing)%20adalah,semuanya%20harus%20dilakukan%20secara%20manual.>)> [02 Juli 2020].



LAMPIRAN A BUKTI HASIL DATA PADA TOOLS OMNET

A.1 Hasil data Activity Duration pada routing Convergecast

Browse Data

Here you can see all data that come from the files specified in the Inputs page.

All (1322 / 1322) Vectors (0 / 0) Scalars (28 / 1322) Histograms (0 / 0)

runID filter module filter activity duration

Folder	File name	Config name	Run number	Run id	Module	Name	Value
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[0].appl	activity duration	26.1927324...
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[1].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[2].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[3].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[4].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[5].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[6].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[7].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[8].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[9].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[10].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[11].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[12].appl	activity duration	-1.0
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[13].appl	activity duration	-1.0

A.2 Hasil data BackOff Duration pada Convergecast

convergecast.anf flooding.ini flooding.anf omnetpp.ini config.xml SimpleHost.ned new.anf Host802

Browse Data

Here you can see all data that come from the files specified in the Inputs page.

All (1322 / 1322) Vectors (0 / 0) Scalars (28 / 1322) Histograms (0 / 0)

runID filter module filter backoffDurations

Folder	File name	Config name	R	Run id	Module	Name	Value
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[13].nic...	backoffDurati...	0.01632
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[12].nic...	backoffDurati...	0.00864
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[11].nic...	backoffDurati...	0.00256
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[10].nic...	backoffDurati...	0.0080
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[9].nic...	backoffDurati...	0.01568
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[8].nic...	backoffDurati...	0.00608
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[7].nic...	backoffDurati...	0.01184
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[6].nic...	backoffDurati...	0.0128
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[5].nic...	backoffDurati...	0.01792
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[4].nic...	backoffDurati...	0.0304
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[3].nic...	backoffDurati...	0.00832
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[2].nic...	backoffDurati...	0.0096
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[1].nic...	backoffDurati...	0.0224
/MiXiM...	convergecast-...	convergecast	2	convergecast-2-20191118-10:40:38-3324	WSNRouting.node[0].nic...	backoffDurati...	0.00224



LAMPIRAN B HASIL PENGIRIMAN DATA PADA TOOLS OMNET

B.1 Hasil Pengiriman data pengujian pada salah satu topologi node *Routing Convergecast*

```

** Event #335 T=0.958714902625 WSNRouting.node[2].nic.mac, on 'route-flood'
** Event #336 T=0.958714902625 WSNRouting.node[2].netwl, on 'route-flood'
** Event #337 T=0.958714902625 WSNRouting.node[3].nic.phy, on 'route-flood'
** Event #338 T=0.958714902625 WSNRouting.node[3].nic.mac, on 'route-flood'
** Event #339 T=0.958714902625 WSNRouting.node[3].netwl, on 'route-flood'
** Event #340 T=0.958906902625 WSNRouting.node[1].nic.phy, on '{radio switching over}'
** Event #341 T=0.958906902625 WSNRouting.node[1].nic.mac, on '{Radio switching over}'
** Event #342 T=31.296311395708 WSNRouting.node[2].appl, on 'appDelay'
** Event #343 T=31.296311395708 WSNRouting.node[2].tranl, on 'Data'
** Event #344 T=31.296311395708 WSNRouting.node[2].netwl, on 'Data'
WiseRoute: Clnfo removed, netw addr=0
** Event #345 T=31.296311395708 WSNRouting.node[2].nic.mac, on 'Data'
** Event #346 T=31.296951395708 WSNRouting.node[2].nic.mac, on 'timer-backoff'
** Event #347 T=31.298871395708 WSNRouting.node[2].nic.mac, on 'timer-cca'
** Event #348 T=31.299063395708 WSNRouting.node[2].nic.phy, on '{radio switching over}'
** Event #349 T=31.299063395708 WSNRouting.node[2].nic.phy, on 'Data'
** Event #350 T=31.299063395708 WSNRouting.node[2].nic.mac, on '{Radio switching over}'
** Event #351 T=31.299063395708 WSNRouting.node[0].nic.phy, on 'Data'
** Event #352 T=31.299063395708 WSNRouting.node[1].nic.phy, on 'Data'
** Event #353 T=31.299063395708 WSNRouting.node[3].nic.phy, on 'Data'
** Event #354 T=31.299063395708 WSNRouting.node[4].nic.phy, on 'Data'
** Event #355 T=31.299063395708 WSNRouting.node[5].nic.phy, on 'Data'
...
** Event #2971 T=57.512723796992 WSNRouting.node[7].nic.phy, on 'Data'
** Event #2972 T=57.512723796992 WSNRouting.node[8].nic.phy, on 'Data'
** Event #2973 T=57.512723796992 WSNRouting.node[9].nic.phy, on 'Data'
** Event #2974 T=57.512723796992 WSNRouting.node[10].nic.phy, on 'Data'
** Event #2975 T=57.512723796992 WSNRouting.node[11].nic.phy, on 'Data'
** Event #2976 T=57.512723796992 WSNRouting.node[12].nic.phy, on 'Data'
** Event #2977 T=57.512723796992 WSNRouting.node[13].nic.phy, on 'Data'
** Event #2978 T=57.512723796992 WSNRouting.node[0].nic.phy, on 'Data'
** Event #2979 T=57.512723796992 WSNRouting.node[0].nic.mac, on 'Data'
** Event #2980 T=57.512723796992 WSNRouting.node[0].netwl, on 'Data'
** Event #2981 T=57.512723796992 WSNRouting.node[2].nic.phy, on 'Data'
** Event #2982 T=57.512723796992 WSNRouting.node[2].nic.mac, on 'Data'
** Event #2983 T=57.512723796992 WSNRouting.node[2].netwl, on 'Data'
** Event #2984 T=57.512723796992 WSNRouting.node[4].nic.phy, on 'Data'
** Event #2985 T=57.512723796992 WSNRouting.node[4].nic.mac, on 'Data'
** Event #2986 T=57.512723796992 WSNRouting.node[4].netwl, on 'Data'
** Event #2987 T=57.512915796992 WSNRouting.node[1].nic.phy, on '{radio switching over}'
** Event #2988 T=57.512915796992 WSNRouting.node[1].nic.mac, on '{Radio switching over}'
** Event #2989 T=60 WSNRouting.node[0].battery, on 'auto-update'
<!-- Simulation time limit reached -- simulation stopped at event #2990, t=60.
** Calling finish() methods of modules

```

B.2 Hasil Pengiriman data pengujian pada salah satu topologi node *Routing Flooding*



in encaps...

Cinfo removed, netw addr=-1

netw 0 sending packet

sendDown: nHop=L3BROADCAST -> message has to be broadcasted -> set destMac=L2BROADCAST

pkt encapsulated

```

** Event #4 T=36.464405069127 WSNRouting.node[0].nic.mac (CSMA802154, id=26), on 'Data' (NetwPkt, id=127)
** Event #5 T=36.466445069127 WSNRouting.node[0].nic.mac (CSMA802154, id=26), on selfmsg 'timer-backoff' (cMessage, id=0)
** Event #6 T=36.468565069127 WSNRouting.node[0].nic.mac (CSMA802154, id=26), on selfmsg 'timer-cca' (cMessage, id=1)
** Event #7 T=36.468757069127 WSNRouting.node[0].nic.phy (PhyLayerBattery, id=27), on selfmsg 'radio switching over' (cMessage, id=4)
** Event #8 T=36.468757069127 WSNRouting.node[0].nic.phy (PhyLayerBattery, id=27), on 'Data' (MacPkt, id=129)
** Event #9 T=36.468757069127 WSNRouting.node[0].nic.mac (CSMA802154, id=26), on 'Radio switching over' (cMessage, id=130)
** Event #10 T=36.468757069127 WSNRouting.node[1].nic.phy (PhyLayerBattery, id=37), on 'Data' (MiximAirFrame, id=132)
** Event #11 T=36.468757069127 WSNRouting.node[2].nic.phy (PhyLayerBattery, id=47), on 'Data' (MiximAirFrame, id=133)
** Event #12 T=36.468757069127 WSNRouting.node[3].nic.phy (PhyLayerBattery, id=57), on 'Data' (MiximAirFrame, id=134)
** Event #13 T=36.468757069127 WSNRouting.node[4].nic.phy (PhyLayerBattery, id=67), on 'Data' (MiximAirFrame, id=135)
** Event #14 T=36.468757069127 WSNRouting.node[5].nic.phy (PhyLayerBattery, id=77), on 'Data' (MiximAirFrame, id=136)
** Event #15 T=36.468757069127 WSNRouting.node[6].nic.phy (PhyLayerBattery, id=87), on 'Data' (MiximAirFrame, id=137)
** Event #16 T=36.468757069127 WSNRouting.node[7].nic.phy (PhyLayerBattery, id=97), on 'Data' (MiximAirFrame, id=138)

data msg BROADCAST! ttl = 10 > 1 -> rebroadcast msg & send to upper
** Event #43 T=36.470933069127 WSNRouting.node[4].nic.mac (CSMA802154, id=66), on 'Data' (NetwPkt, id=149)
** Event #44 T=36.470933069127 WSNRouting.node[4].trn1 (Aggregation, id=62), on 'Data' (ApplPkt, id=150)
** Event #45 T=36.470933069127 WSNRouting.node[4].appl (SensorApplLayer, id=63), on 'Data' (ApplPkt, id=150)
** Event #46 T=36.470933069127 WSNRouting.node[5].nic.phy (PhyLayerBattery, id=77), on selfmsg 'Data' (MiximAirFrame, id=136)
** Event #47 T=36.470933069127 WSNRouting.node[5].nic.mac (CSMA802154, id=76), on 'Data' (MacPkt, id=129)
** Event #48 T=36.470933069127 WSNRouting.node[5].netw1 (Flood, id=71), on 'Data' (NetwPkt, id=127)

data msg BROADCAST! ttl = 10 > 1 -> rebroadcast msg & send to upper
** Event #49 T=36.470933069127 WSNRouting.node[5].nic.mac (CSMA802154, id=76), on 'Data' (NetwPkt, id=152)
** Event #50 T=36.470933069127 WSNRouting.node[5].trn1 (Aggregation, id=72), on 'Data' (ApplPkt, id=153)
** Event #51 T=36.470933069127 WSNRouting.node[5].appl (SensorApplLayer, id=73), on 'Data' (ApplPkt, id=153)
** Event #52 T=36.471125069127 WSNRouting.node[0].nic.phy (PhyLayerBattery, id=27), on selfmsg 'radio switching over' (cMessage, id=4)
** Event #53 T=36.471125069127 WSNRouting.node[0].nic.mac (CSMA802154, id=26), on 'Radio switching over' (cMessage, id=155)
** Event #54 T=36.471893069127 WSNRouting.node[4].nic.mac (CSMA802154, id=66), on selfmsg 'timer-backoff' (cMessage, id=36)
** Event #55 T=36.472533069127 WSNRouting.node[5].nic.mac (CSMA802154, id=76), on selfmsg 'timer-backoff' (cMessage, id=45)
** Event #56 T=36.473813069127 WSNRouting.node[4].nic.mac (CSMA802154, id=66), on selfmsg 'timer-cca' (cMessage, id=37)
** Event #57 T=36.474005069127 WSNRouting.node[4].nic.phy (PhyLayerBattery, id=67), on selfmsg 'radio switching over' (cMessage, id=40)
** Event #58 T=36.474005069127 WSNRouting.node[4].nic.phy (PhyLayerBattery, id=67), on 'Data' (MacPkt, id=156)
** Event #59 T=36.474005069127 WSNRouting.node[4].nic.mac (CSMA802154, id=66), on 'Radio switching over' (cMessage, id=157)
** Event #60 T=36.474005069127 WSNRouting.node[0].nic.phy (PhyLayerBattery, id=27), on 'Data' (MiximAirFrame, id=159)
** Event #61 T=36.474005069127 WSNRouting.node[1].nic.phy (PhyLayerBattery, id=37), on 'Data' (MiximAirFrame, id=160)
** Event #62 T=36.474005069127 WSNRouting.node[2].nic.phy (PhyLayerBattery, id=47), on 'Data' (MiximAirFrame, id=161)
** Event #63 T=36.474005069127 WSNRouting.node[3].nic.phy (PhyLayerBattery, id=57), on 'Data' (MiximAirFrame, id=162)
** Event #64 T=36.474005069127 WSNRouting.node[5].nic.phy (PhyLayerBattery, id=77), on 'Data' (MiximAirFrame, id=163)

data msg BROADCAST! ttl = 9 > 1 -> rebroadcast msg & send to upper
** Event #95 T=36.476181069127 WSNRouting.node[2].nic.mac (CSMA802154, id=46), on 'Data' (NetwPkt, id=177)
** Event #96 T=36.476181069127 WSNRouting.node[2].trn1 (Aggregation, id=42), on 'Data' (ApplPkt, id=178)
** Event #97 T=36.476181069127 WSNRouting.node[2].nic.mac (CSMA802154, id=46), on selfmsg 'timer-backoff' (cMessage, id=18)
** Event #98 T=36.476181069127 WSNRouting.node[2].appl (SensorApplLayer, id=43), on 'Data' (ApplPkt, id=178)
** Event #99 T=36.476181069127 WSNRouting.node[3].nic.phy (PhyLayerBattery, id=57), on selfmsg 'Data' (MiximAirFrame, id=162)
** Event #100 T=36.476181069127 WSNRouting.node[3].nic.mac (CSMA802154, id=56), on 'Data' (MacPkt, id=180)
** Event #101 T=36.476181069127 WSNRouting.node[3].netw1 (Flood, id=51), on 'Data' (NetwPkt, id=181)

data msg BROADCAST! ttl = 9 > 1 -> rebroadcast msg & send to upper
** Event #102 T=36.476181069127 WSNRouting.node[3].nic.mac (CSMA802154, id=56), on 'Data' (NetwPkt, id=182)
** Event #103 T=36.476181069127 WSNRouting.node[3].trn1 (Aggregation, id=52), on 'Data' (ApplPkt, id=183)
** Event #104 T=36.476181069127 WSNRouting.node[3].appl (SensorApplLayer, id=53), on 'Data' (ApplPkt, id=183)
** Event #105 T=36.476181069127 WSNRouting.node[5].nic.phy (PhyLayerBattery, id=77), on selfmsg 'Data' (MiximAirFrame, id=163)
** Event #106 T=36.476181069127 WSNRouting.node[5].nic.mac (CSMA802154, id=76), on 'Data' (MacPkt, id=156)
** Event #107 T=36.476181069127 WSNRouting.node[5].netw1 (Flood, id=71), on 'Data' (NetwPkt, id=149)

```



LAMPIRAN C HASIL DATA

C.1 Tabel hasil data pada parameter Activity Durations

node 1	-1
node 2	0,0141
node 3	0,0098
node 4	0,0098
node 5	0,0052
node 6	0,0098
node 7	-1
total (Rata-Rata)	0,27876

node 1	-1
node 2	-1
node 3	-1
node 4	0,01248
node 5	-1
node 6	-1
node 7	-1
total (Rata-Rata)	0,85536

node 1	-1
node 2	250,0206
node 3	300,0163
node 4	300,0098
node 5	300,0046
node 6	300,0046
node 7	250,0231
node 8	250,0163
node 9	250,0231
node 10	250,0124
node 11	250,0098
node 12	0,027392
node 13	250,0231
node 14	150,0341
Total	221,3732

node 1	-1
node 2	-1
node 3	-1
node 4	-1
node 5	-1
node 6	-1
node 7	-1
node 8	-1
node 9	-1
node 10	-1
node 11	-1
node 12	-1
node 13	-1
node 14	0,04784
total	-0,92515



node 1	496,7869
node 2	495,5176
node 3	496,7986
node 4	496,7869
node 5	496,7896
node 6	496,7869
node 7	496,808
node 8	496,7986
node 9	496,808
node 10	496,8027
node 11	496,8027
node 12	496,8281
node 13	496,808
node 14	496,808
node 15	496,7896
node 16	496,7869
node 17	496,7986
node 18	496,2234
node 19	496,7986
node 20	496,7896
node 21	496,7869
total	496,7097

node 1	-1
node 2	-1
node 3	-1
node 4	-1
node 5	-1
node 6	-1
node 7	-1
node 8	-1
node 9	-1
node 10	-1
node 11	-1
node 12	-1
node 13	-1
node 14	-1
node 15	-1
node 16	-1
node 17	-1
node 18	-1
node 19	450,0121
node 20	-1
node 21	-1
total	20,47677



node 1	496,7869
node 2	957,0724
node 3	958,3455
node 4	958,3455
node 5	958,3455
node 6	958,3455
node 7	958,357
node 8	958,3543
node 9	958,3595
node 10	958,357
node 11	957,0724
node 12	958,3595
node 13	958,3741
node 14	958,3741
node 15	958,3455
node 16	958,3505
node 17	958,3455
node 18	958,3455
node 19	958,3505
node 20	958,3455
node 21	958,3455
node 22	958,3864
node 23	958,3889
node 24	958,3889
node 25	958,3686
node 26	958,3945
node 27	958,3889
node 28	958,3945
total	941,7853

node 1	-1
node 2	-1
node 3	-1
node 4	-1
node 5	-1
node 6	-1
node 7	-1
node 8	-1
node 9	-1
node 10	-1
node 11	-1
node 12	-1
node 13	-1
node 14	-1
node 15	-1
node 16	-1
node 17	-1
node 18	-1
node 19	-1
node 20	-1
node 21	-1
node 22	-1
node 23	-1
node 24	-1
node 25	-1
node 26	-1
node 27	-1
node 28	950,0475
total	32,96598



node 1	496,7863
node 2	1457,073
node 3	1458,346
node 4	1458,363
node 5	1458,346
node 6	1458,366
node 7	1458,379
node 8	1458,37
node 9	1458,37
node 10	1458,357
node 11	1458,373
node 12	1457,085
node 13	1458,379
node 14	1458,385
node 15	1458,346
node 16	1458,346
node 17	1458,346
node 18	1458,346
node 19	1458,346
node 20	1458,346
node 21	1458,385
node 22	1458,393
node 23	1458,395
node 24	1458,393
node 25	1458,395
node 26	1458,385
node 27	1458,393
node 28	1458,393
node 29	1458,395
node 30	1458,395
node 31	1458,401
node 32	1458,354
node 33	1458,351
node 34	1458,346
node 35	1458,391
total	1430,823

node 1	-1
node 2	-1
node 3	-1
node 4	-1
node 5	-1
node 6	-1
node 7	-1
node 8	-1
node 9	-1
node 10	-1
node 11	-1
node 12	-1
node 13	-1
node 14	-1
node 15	-1
node 16	-1
node 17	-1
node 18	-1
node 19	-1
node 20	-1
node 21	-1
node 22	-1
node 23	-1
node 24	-1
node 25	-1
node 26	-1
node 27	496,7922
node 28	-1
node 29	-1
node 30	-1
node 31	-1
node 32	-1
node 33	-1
node 34	-1
node 35	-1
total	13,2226343



C.2 Tabel hasil data pada parameter Backoff Durations

node 1	9,00006	node 1	0,00224
node 2	9,00006	node 2	0,00128
node 3	0	node 3	3,00002
node 4	0,00352	node 4	6,00004
node 5	3,00002	node 5	0,01664
node 6	0,00736	node 6	0,00352
node 7	0	node 7	0,0016
total	3,001574	total	1,289334

node 1	0,00448	node 1	0,00896
node 2	0,02368	node 2	0,0112
node 3	0,00608	node 3	0,01856
node 4	0,01632	node 4	0,01504
node 5	0,01632	node 5	0,0112
node 6	0,03158	node 6	0,02944
node 7	0,01632	node 7	9,00006
node 8	0,01248	node 8	0,00224
node 9	0,01216	node 9	0,00256
node 10	0,01792	node 10	0,0112
node 11	0,024	node 11	0,00544
node 12	0,00128	node 12	0
node 13	0,0112	node 13	9,00006
node 14	3,00002	node 14	0,00384
total	0,228131	total	1,294271



node 1	1,8684
node 2	1,4976
node 3	1,7222
node 4	1,72
node 5	1,7644
node 6	1,6508
node 7	1,0876
node 8	1,5132
node 9	1,5632
node 10	1,3792
node 11	1,1952
node 12	1,2617
node 13	1,1126
node 14	1,0694
node 15	1,5024
node 16	1,5862
node 17	2,1123
node 18	1,6844
node 19	1,8185
node 20	1,8284
node 21	1,5708
total	1,548024

node 1	0,0147
node 2	0,0892
node 3	0,0848
node 4	0,1036
node 5	0,1267
node 6	0,078
node 7	0,0195
node 8	0,0672
node 9	0,0342
node 10	0,0502
node 11	0,031
node 12	0,0108
node 13	0,0224
node 14	0,0057
node 15	0,0633
node 16	0,0716
node 17	0,0736
node 18	0,0492
node 19	0,0851
node 20	0,063
node 21	0
total	0,054467



node 1	2,4838
node 2	2,5171
node 3	2,6108
node 4	2,3718
node 5	2,726
node 6	2,5392
node 7	2,079
node 8	2,6931
node 9	2,2787
node 10	2,0956
node 11	2,0819
node 12	2,2009
node 13	2,2966
node 14	2,4192
node 15	2,5472
node 16	2,4643
node 17	2,5596
node 18	2,6393
node 19	2,74414
node 20	2,5689
node 21	2,4547
node 22	2,0876
node 23	2,3145
node 24	2,3052
node 25	2,3536
node 26	2,1974
node 27	2,2316
node 28	2,4448
total	2,403805

node 1	0,0198
node 2	0,0164
node 3	0,2812
node 4	0,214
node 5	0,1785
node 6	0,1945
node 7	0,1689
node 8	0,1488
node 9	0,1625
node 10	0,1817
node 11	0,1587
node 12	0,2131
node 13	0,1584
node 14	0,2425
node 15	0,1817
node 16	0,2137
node 17	0,2556
node 18	0,1654
node 19	0,1676
node 20	0,2217
node 21	0,1641
node 22	0,1548
node 23	0,1728
node 24	0,1955
node 25	0,1888
node 26	0,184
node 27	0,1644
node 28	0
total	0,170325



node 1	3,5625
node 2	3,3395
node 3	3,3996
node 4	3,6585
node 5	3,7542
node 6	3,552
node 7	2,9737
node 8	3,1395
node 9	2,8976
node 10	3,04
node 11	2,6995
node 12	2,6803
node 13	3,1132
node 14	3,0905
node 15	3,3852
node 16	3,1702
node 17	3,5244
node 18	3,7123
node 19	3,4217
node 20	3,6848
node 21	3,3504
node 22	3,2662
node 23	3,4198
node 24	3,502
node 25	3,3929
node 26	3,248
node 27	3,3814
node 28	3,247
node 29	3,407
node 30	2,9314
node 31	3,0582
node 32	3,0265
node 33	3,1139
node 34	3,3945
node 35	3,1616
total	3,277143

node 1	3,6915
node 2	3,5769
node 3	3,2204
node 4	3,4499
node 5	3,5667
node 6	3,3801
node 7	2,9945
node 8	3,2323
node 9	3,2166
node 10	3,0294
node 11	2,5257
node 12	2,8755
node 13	3,1059
node 14	2,9897
node 15	3,407
node 16	3,2572
node 17	3,4963
node 18	3,366
node 19	3,5478
node 20	3,7654
node 21	3,5081
node 22	3,238
node 23	3,1168
node 24	3,2812
node 25	3,0217
node 26	3,3177
node 27	3,3481
node 28	3,422
node 29	3,4204
node 30	3,0979
node 31	3,1539
node 32	2,6188
node 33	2,9216
node 34	2,967
node 35	0,0124
total	3,146869



C. 3 Tabel hasil data pada parameter Latency

node 1	0
node 2	0,0141
node 3	0,0098
node 4	0,0098
node 5	0,0052
node 6	0,0098
node 7	0
total	0,006957

node 1	0
node 2	0,01856
node 3	0,014775
node 4	0,011181
node 5	0,006518
node 6	0,009499
node 7	0,03512
node 8	0,01649
node 9	0,024784
node 10	0,024944
node 11	0,020608
node 12	0,027392
node 13	0,033264
node 14	0,0376
total	0,020052

node 1	0
node 2	0
node 3	0
node 4	0
node 5	0,01248
node 6	0
node 7	0
total	0,001783

node 1	0
node 2	0
node 3	0
node 4	0
node 5	0
node 6	0
node 7	0
node 8	0
node 9	0
node 10	0
node 11	0
node 12	0
node 13	0
node 14	0,04784
total	0,003417



node 1	0,0122
node 2	0,0144
node 3	0,0116
node 4	0,0105
node 5	0,0116
node 6	0,011
node 7	0,0214
node 8	0,0129
node 9	0,0165
node 10	0,016
node 11	0,0185
node 12	0,0215
node 13	0,0208
node 14	0,0251
node 15	0,0126
node 16	0,0132
node 17	0,0118
node 18	0,0122
node 19	0,0103
node 20	0,0118
node 21	0,012
total	0,014662

node 1	0
node 2	0
node 3	0
node 4	0
node 5	0
node 6	0
node 7	0
node 8	0
node 9	0
node 10	0
node 11	0
node 12	0
node 13	0
node 14	0
node 15	0
node 16	0
node 17	0
node 18	0
node 19	0
node 20	0
node 21	0,0074
total	0,000352



node 1	0,01286
node 2	0,0189
node 3	0,017
node 4	0,0158
node 5	0,0163
node 6	0,0147
node 7	0,0202
node 8	0,0166
node 9	0,0185
node 10	0,0174
node 11	0,0218
node 12	0,024
node 13	0,018
node 14	0,0214
node 15	0,0174
node 16	0,0194
node 17	0,0169
node 18	0,0171
node 19	0,0148
node 20	0,0162
node 21	0,0164
node 22	0,0214
node 23	0,0229
node 24	0,0231
node 25	0,0238
node 26	0,0239
node 27	0,0247
node 28	0,024
total	0,019124

node 1	0
node 2	0
node 3	0
node 4	0
node 5	0
node 6	0
node 7	0
node 8	0
node 9	0
node 10	0
node 11	0
node 12	0
node 13	0
node 14	0
node 15	0
node 16	0
node 17	0
node 18	0
node 19	0
node 20	0
node 21	0
node 22	0
node 23	0
node 24	0
node 25	0
node 26	0
node 27	0
node 28	0,0414
total	0,001479



node 1	0,0209
node 2	0,0209
node 3	0,0193
node 4	0,0182
node 5	0,0191
node 6	0,0177
node 7	0,0228
node 8	0,0193
node 9	0,0202
node 10	0,01298
node 11	0,0259
node 12	0,0253
node 13	0,0207
node 14	0,0226
node 15	0,021
node 16	0,0185
node 17	0,0198
node 18	0,0175
node 19	0,0193
node 20	0,0192
node 21	0,0234
node 22	0,0238
node 23	0,0239
node 24	0,025
node 25	0,025
node 26	0,0256
node 27	0,025
node 28	0,0267
node 29	0,0319
node 30	0,0274
node 31	0,0329
node 32	0,0276
node 33	0,0255
node 34	0,0243
node 35	0,017
total	0,022462

node 1	0
node 2	0
node 3	0
node 4	0
node 5	0
node 6	0
node 7	0
node 8	0
node 9	0
node 10	0
node 11	0
node 12	0
node 13	0
node 14	0
node 15	0
node 16	0
node 17	0
node 18	0
node 19	0
node 20	0
node 21	0
node 22	0
node 23	0
node 24	0
node 25	0
node 26	0
node 27	0
node 28	0
node 29	0
node 30	0
node 31	0
node 32	0
node 33	0
node 34	0
node 35	0,0308
total	0,00088



C.4 Tabel Hasil Data pada Parameter Nilai *Power Consumption*

node 1	56,3998
node 2	56,3998
node 3	56,3998
node 4	56,3998
node 5	56,3998
node 6	56,3998
node 7	56,4
total	56,39983

node 1	56,39982
node 2	56,39984
node 3	56,39982
node 4	56,39982
node 5	56,39982
node 6	56,39982
node 7	56,39989
node 8	56,39984
node 9	56,39989
node 10	56,39992
node 11	56,39997
node 12	56,39989
node 13	56,39994
node 14	56,39989
total	56,39987

node 1	56,3999
node 2	56,3997
node 3	56,3997
node 4	56,3997
node 5	56,3995
node 6	56,3997
node 7	56,3998
total	56,39971

node 1	56,3998
node 2	56,39975
node 3	56,39981
node 4	56,39981
node 5	56,39978
node 6	56,39978
node 7	56,39981
node 8	56,39998
node 9	56,39995
node 10	56,39995
node 11	56,39995
node 12	56,39996
node 13	56,4
node 14	56,39995
total	56,39988



node 1	56,3978
node 2	56,3977
node 3	56,3978
node 4	56,3977
node 5	56,3978
node 6	56,3978
node 7	56,3976
node 8	56,3976
node 9	56,3976
node 10	56,3976
node 11	56,3977
node 12	56,3977
node 13	56,3975
node 14	56,3977
node 15	56,3977
node 16	56,3977
node 17	56,398
node 18	56,3977
node 19	56,3978
node 20	56,3977
node 21	56,3977
total	56,39771

node 1	56,3998
node 2	56,3998
node 3	56,3998
node 4	56,3998
node 5	56,3998
node 6	56,3998
node 7	56,3998
node 8	56,3998
node 9	56,3998
node 10	56,3999
node 11	56,3999
node 12	56,3999
node 13	56,3999
node 14	56,3999
node 15	56,3998
node 16	56,3998
node 17	56,3998
node 18	56,3998
node 19	56,3998
node 20	56,3998
node 21	56,3999
total	56,39983



node 1	56,3984
node 2	56,3983
node 3	56,3984
node 4	56,3983
node 5	56,3985
node 6	56,3984
node 7	56,3984
node 8	56,3984
node 9	56,3983
node 10	56,3983
node 11	56,3984
node 12	56,3985
node 13	56,3984
node 14	56,3984
node 15	56,3984
node 16	56,3983
node 17	56,3984
node 18	56,3985
node 19	56,3985
node 20	56,3984
node 21	56,3984
node 22	56,3983
node 23	56,3983
node 24	56,3984
node 25	56,3983
node 26	56,3983
node 27	56,3983
node 28	56,3984
total	56,39838

node 1	56,3998
node 2	56,3998
node 3	56,3998
node 4	56,3998
node 5	56,3998
node 6	56,3998
node 7	56,3998
node 8	56,3998
node 9	56,3998
node 10	56,3998
node 11	56,3998
node 12	56,3998
node 13	56,3998
node 14	56,3998
node 15	56,3998
node 16	56,3998
node 17	56,3999
node 18	56,3998
node 19	56,3998
node 20	56,3998
node 21	56,3998
node 22	56,3998
node 23	56,3998
node 24	56,3998
node 25	56,3998
node 26	56,3998
node 27	56,3998
node 28	56,4
total	56,39981



node 1	56,399
node 2	56,399
node 3	56,3991
node 4	56,399
node 5	56,399
node 6	56,399
node 7	56,399
node 8	56
node 9	56
node 10	56
node 11	56
node 12	56
node 13	56
node 14	56
node 15	56
node 16	56
node 17	56
node 18	56
node 19	56
node 20	56
node 21	56
node 22	56
node 23	56
node 24	56
node 25	56
node 26	56
node 27	56
node 28	56
node 29	56
node 30	56
node 31	56
node 32	56
node 33	56
node 34	56
node 35	56
total	56,39899

node 1	56,3989
node 2	56,3989
node 3	56,3989
node 4	56,3989
node 5	56,3989
node 6	56,3989
node 7	56,3989
node 8	56,3989
node 9	56,3989
node 10	56,3989
node 11	56,3989
node 12	56,399
node 13	56,3989
node 14	56,3988
node 15	56,3989
node 16	56,3988
node 17	56,3989
node 18	56,3989
node 19	56,3989
node 20	56,3989
node 21	56,3989
node 22	56,3989
node 23	56,3988
node 24	56,3988
node 25	56,3989
node 26	56,3988
node 27	56,3988
node 28	56,3989
node 29	56,3988
node 30	56,3988
node 31	56,3988
node 32	56,3988
node 33	56,3988
node 34	56,3988
node 35	56,3999
total	56,3989