

Tersedia online di [www.journal.unipdu.ac.id](http://www.journal.unipdu.ac.id)

Unipdu

Terakreditasi S2 – SK No. 34/E/KPT/2018

Halaman jurnal di [www.journal.unipdu.ac.id/index.php/register](http://www.journal.unipdu.ac.id/index.php/register)

## Pengembangan mekanisme *grid based clustering* untuk peningkatan kinerja LEACH pada lingkungan *Wireless Sensor Network*

### *Development of a grid-based clustering mechanism to improve LEACH performance in the Wireless Sensor Network environment*

Kristoforus Fallo <sup>a</sup>, Waskitho Wibisono <sup>b</sup>, Kun Nursyaful Priyo Pamungkas <sup>c</sup><sup>ab</sup> Teknik Informatika, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesiaemail: <sup>a</sup> [falloito@gmail.com](mailto:falloito@gmail.com), <sup>b</sup> [waswib@if.its.ac.id](mailto:waswib@if.its.ac.id), <sup>c</sup> [kun.18051@mhs.its.ac.id](mailto:kun.18051@mhs.its.ac.id)

#### INFO ARTIKEL

##### Sejarah artikel:

Menerima 19 Juni 2019  
Revisi 6 Agustus 2019  
Diterima 13 September 2019  
Online 13 September 2019

##### Kata kunci:

clustering  
energi  
grid-based LEACH  
merge Cluster Head  
sensor node  
Wireless Sensor Network

##### Keywords:

clustering  
energi  
grid-based LEACH  
merge Cluster Head  
sensor node  
Wireless Sensor Network

##### Style APA dalam menyitasi artikel ini:

Fallo, K., Wibisono, W., & Pamungkas, K. N. (2019). Pengembangan mekanisme *grid based clustering* untuk peningkatan kinerja LEACH pada lingkungan *Wireless Sensor Network*. *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, 5(2), 152-161.

#### ABSTRAK

*Low Energy Adaptive Clustering Hierarchy* (LEACH) merupakan algoritma *routing* pada *Wireless Sensor Network* (WSN) berbasis *cluster*. LEACH memilih sebuah *node* sebagai *cluster head* (CH) yang tugasnya untuk melakukan komunikasi dengan *sink* maupun guna mengumpulkan data dari *member node*. Persebaran CH pada LEACH yang dikatakan acak, kadang mengalami masalah mengingat rumus probabilitas pada tiap *round*. Hal ini akan menyebabkan CH yang terpilih bisa berada di tepi area, juga terjadinya pemborosan energi karena jalur yang terbentuk akan menjadi panjang. Oleh karena itu, kami ingin mengembangkan *routing protocol* G-LEACH menggunakan teknik *merge* CH dalam suatu area (*grid*) disertai beberapa parameter yang relevan, seperti posisi *node*, *node* dengan sisa energi terbesar, dan jarak yang dihitung dalam tiga jarak yaitu jarak *node* menuju *cluster center*, jarak *node* menuju *merge* CH, dan jarak *merge* CH menuju *sink*. Hasil pengujian menunjukkan bahwa dengan menggabungkan *cluster* (*merge* CH) pada transmisi data menuju *sink* pada protokol G-LEACH dapat menghasilkan masa hidup jaringan yang lebih lama pada seluruh operasi *node*, energi yang dibutuhkan pada semua *node* lebih rendah, dan lebih banyak paket data yang dikirim dan diterima oleh *sink*.

#### ABSTRACT

*Low Energy Adaptive Clustering Hierarchy* (LEACH) is a *routing algorithm* in a *cluster-based Wireless Sensor Network* (WSN). LEACH selects a *node* as a *cluster head* (CH) whose responsibility is for communicating with *sinks* and collect data from the *node members*. The distribution of CH on LEACH, which is basically random, sometimes has a problem in remembering the probability formula on each round. This may make the selected CH on the edge of the area as well as generate energy waste because the pathway formed will be lengthy. Therefore, we would like to develop the G-LEACH *routing protocol* using a *merge* CH technique in one area (*grid*) with several relevant parameters, such as the position of the *node*, the *node* with the largest remaining energy, and the distance calculated in three distances: the distance of the *node* to the *clustercenter*, the distance of the *node* to the *merge* CH, and the distance of the *merge* CH to the *sink*. The test result showed that combining *clusters* (*merge* CH) in the data transmission to the *sink* in the G-LEACH protocol could produce a longer network life on all *node* operations, lower energy required for all *nodes*, and more data package sent and received by the *sink*.

© 2019 Register: Jurnal Ilmiah Teknologi Sistem Informasi. Semua hak cipta dilindungi undang-undang.

## 1. Pendahuluan

Perkembangan teknologi sistem informasi saat ini sangat membantu manusia, salah satunya *Wireless Sensor Network* (WSN), yakni jaringan tanpa kabel (nirkabel) yang terdiri atas beberapa *sensor node*,

digunakan untuk mendukung kegiatan manusia dalam melakukan pengamatan di bidang militer, kesehatan, maupun fenomena/kejadian yang terjadi pada cakupan wilayah yang sangat luas. Teknologi ini menggunakan perangkat komputer berukuran ringkas (*sensor node*) dan metode komunikasi secara nirkabel dan terdistribusi yang terdiri atas *sink* dan kumpulan node dalam suatu area tertentu, serta memungkinkan untuk melakukan pengamatan guna memantau suatu sistem atau lingkungan seperti temperatur, tekanan, kelembapan pergesekan, dan lainnya yang didukung peralatan cerdas dan dikontrol secara *real-time* (Li & Shen, 2011). Akan tetapi, baterai yang digunakan dalam WSN tidak untuk diisi ulang atau diganti, sehingga perlu dilakukan peningkatan masa pakai jaringan untuk kinerja yang lebih baik (Sivakumara & Radhika, 2018). Hal ini juga dinyatakan Astuti dan Wibisono (2017) bahwa *sensor node* memiliki sumber daya yang terbatas, pengisian ulang baterai pada umumnya tidak dapat dilakukan, karena posisi *sensor node* berada pada wilayah cakupan yang sangat ekstrim, seperti bencana alam. Sementara, Manzoor, dkk., (2013) menyatakan bahwa waktu hidup jaringan selalu terkait dengan energi dari *node sensor* yang digunakan di daerah terpencil untuk pemantauan yang konstan dan toleran terhadap kesalahan. Oleh karena itu, masa hidup WSN akan mati ketika baterai pada semua *node* mati. Salah satu algoritma *routing protocol* yang dikembangkan untuk meningkatkan efisiensi energi pada WSN oleh Heinzelman, Chandrakasan, dan Balakrishnan (2000), yaitu *Low Energy Adaptive Clustering Hierarchy* (LEACH). LEACH merupakan algoritma *routing berbasis cluster*. LEACH memilih *node* sebagai *cluster head* (CH) yang berfungsi untuk mengumpulkan data dari *node* anggotanya agar selanjutnya komunikasi hanya melalui CH tersebut. Kepala *cluster* ini kemudian mengirim data ke simpul utama yang disebut *wastafel* (Mahajan, Reddy, & Rajput, 2016). Saat proses pemilihan CH, setiap *node* memutuskan apakah dirinya menjadi CH dengan perhitungan persentase target CH yang diinginkan dan juga berapa kali *node* itu menjadi CH. Pemilihan ulang CH bertujuan untuk merotasi jabatan CH agar tidak ada *node* yang mati, karena dengan menjadi CH, energi *node* tersebut menjadi tinggi (Palan, Barbadekar, & Patil, 2017), jadi energi sebagai faktor penting dalam mengambil keputusan dalam pemilihan CH (Adhisa & Wibisono, 2018). Pergantian CH pada LEACH dilakukan berdasarkan waktu dengan sebuah sesi yang disebut *round*. Pada *round* baru, setiap *node* akan memilih angka acak dan membandingkannya dengan nilai *threshold* yang digunakan Heinzelman, Chandrakasan, dan Balakrishnan (2000) dengan menghitung berdasarkan Persamaan 1.

$$T(n) = \begin{cases} \frac{p}{1-p \times (\text{rmod} \frac{1}{p})} & \text{if } n \in G \\ 0 & \end{cases} \quad (1)$$

Persamaan 1 digunakan untuk mencari CH dari beberapa penelitian. Sebagian peneliti melakukan penelitian dalam meningkatkan efisiensi *routing* LEACH pada WSN dan juga meminimalkan konsumsi energi. Permasalahan yang terjadi pada LEACH yaitu pada pemilihan CH yang acak pada setiap *round* yang akan menyebabkan CH terpilih bisa berada pada tepi area WSN. Hal ini akan menyebabkan jalur transmisi menuju *sink* menjadi panjang dan dapat menyebabkan terjadinya pemborosan energi. Berdasarkan permasalahan pada LEACH di atas, perlu dikembangkan suatu teknik *grid based clustering* dengan memodifikasi LEACH serta melakukan penggabungan *cluster* dalam satu area (*merge CH*) yang dikenal dengan *routing protocol* G-LEACH untuk meminimalkan konsumsi energi. Teknik *grid based* (*merge CH* dalam penelitian ini) belum pernah dilakukan oleh penelitian terdahulu. Modifikasi pada LEACH ini dilakukan dengan membagi area WSN menjadi beberapa *grid* serta beberapa parameter yang relevan dalam WSN, seperti posisi *merge CH*, jarak *node* menuju *merge CH*, dan jarak *node* menuju *sink*. Hasil pengembangan protokol G-LEACH ini kemudian akan dibandingkan dengan protokol pembandingnya, yakni LEACH.

## 2. State of the Art

Algoritma *routing protocol* LEACH dikembangkan oleh Heinzelman, Chandrakasan, dan Balakrishnan (2000). Terdapat dua fase yang berlangsung pada algoritma *routing protocol* LEACH, yaitu fase *setup* dan fase *steady state*. Fase *setup* dimulai dengan pemilihan *node* sebagai *cluster head* (CH) yang ditentukan secara acak berdasarkan level energi yang tersisa pada interval waktu tertentu. Pada interval berikutnya, sebuah *sensor node* (Hani & Ijeh, 2013), *node* yang akan dipilih sebagai CH berdasarkan sisa energi dengan menggunakan sisa energi untuk putaran berikutnya dan seterusnya sampai *node* terakhir mati (Kaur, Gaba, Miglani, & Pasricha, 2015). *Sensor node* mengambil nilai acak  $x$  sedemikian rupa sehingga  $0 < x < 1$  dan membandingkannya dengan *threshold* tertentu  $T(n)$  yang digunakan pada

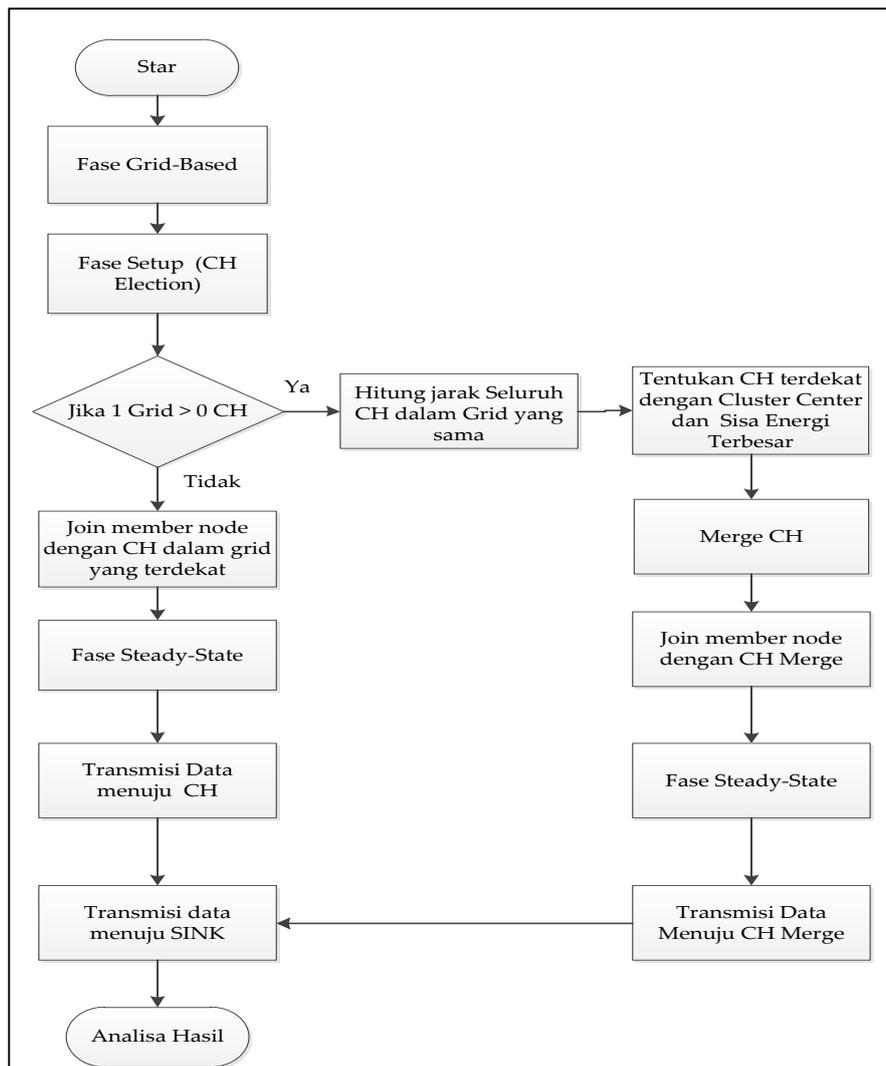
suatu *round*. Jika  $x < T(n)$ , *node* tersebut menjadi CH pada *round* tersebut, dan jika tidak, akan menjadi *member node* (Abushiba, Johnson, Alharthi, & Wright, 2017). Selanjutnya, CH akan mengirim pesan *broadcast* ke *sensor node*. Jika *sensor node* menerima pesan tersebut, akan dibalas dengan pesan yang menyatakan bahwa *sensor node* akan bergabung sebagai anggota *cluster*. Jika *node* yang bukan CH menerima pesan *broadcast* lebih dari satu, CH akan ditentukan berdasarkan tingkat kekuatan transmisi tertinggi (Hani & Ijeh, 2013), karena LEACH membentuk *cluster* dengan cara menentukan CH terlebih dahulu kemudian *node* lain bergabung sebagai *member* dengan bentuk *cluster* yang berbeda tiap *round*. Maka, LEACH dapat dikatakan membentuk *cluster* secara *bottom-up* (Abushiba, Johnson, Alharthi, & Wright, 2017).

Pada fase *steady-state* dilakukan pengiriman data hasil *sensing* dari *member node* berkurang dan menyebabkan *network lifetime* menjadi lebih tinggi. Karena tiap *round* pada tiap *node* akan menjalankan perhitungan probabilitas untuk menentukan dirinya menjadi CH apa tidak, bentuk *cluster* yang terbentuk akan berbeda tiap *round* pada LEACH. Permasalahan yang sering terjadi pada LEACH yaitu persebaran CH yang tidak merata. Hal ini akan menyebabkan CH yang terpilih akan berada di tepi area (Amirthalingam & Anuratha, 2016).

Demikian juga jarak antara setiap *node* dalam *cluster* juga memainkan peran penting dalam konsumsi energi. Ketika CH terpilih jauh dari *sink*, konsumsi energi akan tinggi, oleh karena itu jarak antara CH dengan *sink* harus diperhitungkan dalam pemilihan CH (Amirthalingam & Anuratha, 2016).

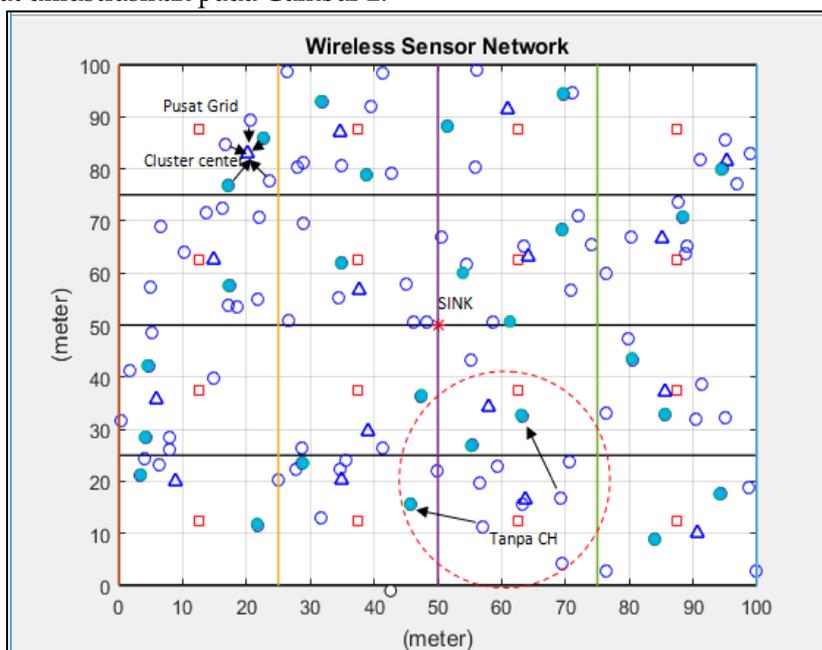
### 3. Metode Penelitian

Pengembangan *routing protocol* G-LEACH dilakukan dengan memodifikasi *routing protocol* LEACH seperti yang ditunjukkan pada *Flowchart* tahapan penelitian Gambar 1.



Gambar 1. *Flowchart* tahapan penelitian

Pengembangan *routing protocol* G-LEACH dilakukan dengan memodifikasi *routing protocol* LEACH. Modifikasi pada *routing protocol* G-LEACH ini dilakukan dalam beberapa fase, yaitu fase *grid based*, fase *setup*, dan fase *steady state*. Tujuan penambahan fase *grid based* pada *routing protocol* G-LEACH ini dilakukan untuk mengimplementasikan model *merge* pada *grid*. Awalnya, *node* disebarluaskan pada area *grid* yang telah ditentukan pada fase *grid base*, selanjutnya algoritma *routing protocol* G-LEACH akan melakukan pemilihan CH dan pembentukan *cluster* pada fase *setup*. Selanjutnya, setelah terbentuknya *cluster* dan pemilihan CH, protokol akan melakukan *merge* CH atau melakukan penggabungan *cluster*. Proses *merge* CH ini ditentukan pada fase *setup* dan akan digunakan pada fase *steady state* untuk melakukan transmisi data menuju *sink*. Tahapan-tahapan dalam *routing protocol* G-LEACH ini dapat diilustrasikan pada Gambar 2.



Gambar 2. Grid 16 formasi

Keterangan:

- = Cluster head
- △ = Cluster center
- = Pusat grid
- = Node normal
- \* = Sink

### 3.1. Fase grid-based

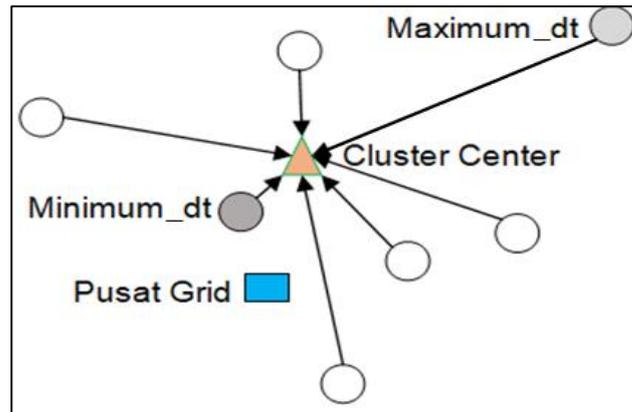
Fase *grid based* merupakan fase pembentukan *grid*. Fase ini terdiri atas beberapa teknik yang diimplementasikan dengan membagi area wsn menjadi sejumlah *grid*. Ada 2 teknik yang digunakan pada fase ini, yaitu tata letak dan jumlah *grid*, kemudian penentuan titik pusat *grid*. Tata letak dan jumlah *grid* pada fase *grid based* ini awalnya area WSN dibagi dalam beberapa *grid* sesuai dengan jumlah *grid*. Penentuan jumlah baris ( $x$ ) dan jumlah kolom ( $y$ ) dengan membagi panjang area WSN ( $Xm$ ) dan lebar WSN ( $Ym$ ) menjadi 4 wilayah atau 4 *grid*, pembagian 4 wilayah atau *grid*, sedangkan pada penentuan titik pusat tiap *grid*, pada panjang ( $x$ ) dan lebar ( $y$ ) setiap *grid* dibagi 2 ini berlaku pada semua area WSN seperti yang dilihat pada Gambar 2.

### 3.2. Fase Setup

Fase *setup* merupakan fase pemilihan CH dan pembentukan *cluster* berdasarkan *round*. Tiap *round* baru pada seluruh *node* akan mengambil secara acak angka 1 dan 0, kemudian menggunakan *threshold*

formula  $T(n)$  jika angka acak yang diambilnya kurang dari *threshold* yang dihitung. Maka *node* tersebut akan menjadi CH dan melakukan *broadcast* pesan bahwa dirinya menjadi CH. *Node* yang tidak menjadi CH yang mendapat pesan tersebut akan bergabung dengan CH dan membentuk *cluster* (Hani & Ijeh, 2013). Pembentukan *cluster* berubah setiap putaran (Rasyid, Lee, Syarif, & Arkham, 2018). Setelah pemilihan CH, setiap *node* akan menentukan jaraknya ke CH sesuai dengan kekuatan sinyal dari siaran informasi yang mereka terima (Li, Ding, & Liu, 2011). Jarak dan energi merupakan kekuatan besaran dalam WSN (Gupta & Marriwala, 2017).

Penetapan ambang batas yang mencakup energi dan jarak antara *node* dengan *base station* dan CH dengan *base station* sangat penting dalam pemilihan CH (Sharma, Mishra, & Srivastava, 2015). Pemilihan CH pada penelitian ini menggunakan perbandingan jarak dan antarsimpul dengan *cluster center* dalam masing-masing *grid* dan sisa energi terbesar masing-masing CH dalam *grid* tersebut. Seperti divisualisasikan pada Gambar 3.



Gambar 3. Pemilihan CH dalam satu *grid*

Berdasarkan gambar di atas, pemilihan CH dalam satu *grid* dengan menggunakan perbandingan jarak antarsimpul, kotak berwarna biru merupakan titik pusat *grid*. Kedua *node* yang berwarna abu-abu merupakan *node* yang memiliki jarak maksimum ( $dMax$ ), yaitu jarak maksimum *node* ke *cluster center* (segitiga) dan *node* dengan jarak minimum ( $dMin$ ) yang merupakan jarak minimum *node* ke *cluster center*, dengan asumsi bahwa *node* yang berada dekat *cluster center* memiliki sisa energi paling besar dan memiliki peluang besar untuk terpilih menjadi CH. Posisi merupakan posisi yang paling ideal untuk *merge*/penggabungan CH *cluster* dalam satu *grid* atau area dengan kepadatan rata-rata. Formula pemilihan CH yang digunakan dalam penelitian ini ditunjukkan pada Persamaan 2.

$$t = \frac{p}{1-p \times (r \bmod \frac{1}{p})} + (1-p) \frac{dMax-dC}{dMax-dMin} \left( \frac{eR}{eO} \right) \quad (2)$$

Pada persamaan tersebut,  $dC$  adalah jarak *node* menuju *cluster center*,  $eR$  merupakan sisa energi dari simpul untuk putaran saat ini, dan  $eO$  adalah energi awal yang diberikan pada setiap *node*. Pemilihan CH pada fase ini memperhitungkan sisa energi dan jarak dari *node* ke *cluster center* dan membandingkannya dengan *node* lain. Pada fase ini juga akan terbentuk beberapa *cluster* di dalam *grid*. Hal ini akan menyebabkan konsumsi energi tidak merata dengan *cluster* yang lainnya memiliki posisi CH berada di tepi *grid*. Oleh sebab itu, perlu adanya penggabungan *cluster* dalam satu *grid* dengan memilih CH yang memiliki sisa energi terbesar sebagai *merge* CH dan berada dekat *cluster center*.

Penerapan *merge* CH dilakukan dengan proses penggabungan CH yang lebih sedikit energi dan CH dengan energi terbesar dengan menambahkan parameter baru antarCH pada fase *setup*. Penggabungan CH ini terjadi dalam *grid* yang sama. Jika di dalam satu *grid* tidak terdapat CH, akan dilakukan penggabungan *node* dengan CH pada *grid* terdekat, seperti yang divisualisasikan pada Gambar 2 (lingkaran berwarna merah bulat putus-putus). Untuk melakukan *merge* CH dapat dihitung menggunakan beberapa parameter *function cost* dari  $ChExisting$  dan  $ChBaru$  dengan  $ChExisting$  menjadi parameter untuk setiap kandidat *merge* CH, sedangkan  $ChBaru$  merupakan parameter untuk

menentukan ulang *merge* CH pada *round* berikutnya. Seperti yang terdapat pada Persamaan 3 dan Persamaan 4.

$$ChExisting = w1 * \left(\frac{jarakLama}{gMaxMin}\right) + w2 * \left(\frac{Eavg}{eGrid}\right) \quad (3)$$

$$ChBaru = w1 * \left(\frac{jarakLama}{gMaxMin}\right) + w2 * \left(\frac{Eavg}{eGrid}\right) \quad (4)$$

Berdasarkan formula perhitungan Persamaan 3 dan Persamaan 4, untuk menentukan kandidat *merge* CH dalam setiap *grid* memenuhi parameter **ChExisting**, kandidat CH tersebut dipilih sebagai *merge* CH. Jika kandidat CH *node* setiap *grid* tidak memenuhi parameter, akan menjadi *node* biasa dalam *grid* itu. Hal ini berlaku juga pada pemilihan ulang *merge* CH selanjutnya pada **ChBaru**. Secara umum, alur metode penelitian *grid based* dalam makalah ini dapat dilihat pada *Pseudocode* 1 dan *Pseudocode* 2.

*Pseudocode* 1. Fase *setup* (CH Election)

1	Fase <i>setup</i> (CH election)
2	Hitung setiap <i>node</i> yang masih hidup per <i>round</i>
3	Generate initial random tiap <i>node</i>
4	Jika initial random <i>node</i> lebih kecil dari <i>treashold</i> , <i>node</i> menjadi CH
5	Menetapkan CH pada <i>round</i> per jalan ke dalam <i>table</i>
6	Menetapkan <i>round</i> dengan <i>node</i> yang tidak menjadi CH
7	Menghitung jarak antara <i>cluster head</i> dengan <i>sink</i>
8	//Menentukan jumlah CH yang telah dipilih
9	Menetapkan ID <i>node</i> CH yang terpilih
10	Mengelompokkan <i>node</i> ke dalam <i>cluster</i> & menghitung jarak <i>node member</i> ke CH
11	Generate initial random tiap CH dalam <i>grid</i> yang sama
12	//Menentukan <i>merge</i> CH
13	Menghitung jarak setiap <i>node</i> dengan CH dalam <i>grid</i> yang sama
14	//Menghitung <i>ChExisting</i> dan <i>costNodeChBaru</i> jika CH ada dalam <i>grid</i>
15	Jika CH tidak di dalam <i>grid</i> pilih CH dalam <i>grid</i> terdekat
16	//Menghitung jarak <i>node</i> dengan CH dari <i>grid</i> terdekat yang berbeda

### 3.3. Fase *Steady-State*

Fase *steady state* merupakan fase untuk melakukan transmisi atau pengiriman paket data ke *merge* CH dan *sink* juga untuk melakukan perhitungan biaya transmisi dan biaya penerimaan oleh *sink* yang dapat dihitung menggunakan persamaan yang diadopsi dari Heinzelman, Chandrakasan, dan Balakrishnan (2000) disajikan pada Persamaan 5, Persamaan 6, dan Persamaan 7.

$$ETx = Eelec \times b + Eamp \times b \times Dtchm^2 \quad (5)$$

$$ERx = (Eelec + EDA) \times b \quad (6)$$

$$ETx = (Eelec + EDA) \times b + Eamp \times b \times Dts^2 \quad (7)$$

Berdasarkan formula perhitungan Persamaan 5, Persamaan 6, dan Persamaan 7 dengan **ETx** merupakan biaya transmisi, **ERx** merupakan biaya penerimaan oleh *sink*, **Eamp** merupakan jumlah energi yang dihabiskan oleh *amplifier* untuk mengirimkan bit, **Eelec** merupakan elektronik radio energi, **EDA** merupakan energi data agregasi, **b** merupakan panjang data dalam bit, sedangkan

***Dtchm*** merupakan jarak *node* menuju *merge CH* dan ***Dts*** merupakan jarak *node* menuju *sink*. Secara umum, fase *steady state* dapat dijelaskan seperti pada *Pseudocode 2*.

*Pseudocode 3. Fase steady state*

1	Fase <i>steady state</i>
2	Untuk setiap <i>node</i> yang masih hidup hitung per <i>round</i> berjalan
3	<i>Generate initial random</i> tiap <i>node</i>
4	//Hitung biaya transmisi <i>node</i> ke <i>merge CH</i>
5	$ETx = (Eelec \times b) + (Eamp \times b \times Dtchm^2)$
6	//Menghitung biaya energi yang digunakan selama proses penerimaan oleh CH
7	$ERx = (Eelec + EDA) \times b$
8	//Menghitung biaya transmisi <i>merge CH</i> ke <i>sink</i>
9	$ETx = ((Eelec + EDA) \times b) + (Eamp \times b \times Dts^2)$
10	// menampilkan CH pada <i>round</i> berjalan
11	if <i>CLheads</i> == 0
12	<i>pesan</i> = ['no available ch at ', num2str( <i>rnd</i> )]
13	// menghitung semua <i>node</i> mati di akhir <i>round</i>

#### 4. Hasil dan Pembahasan

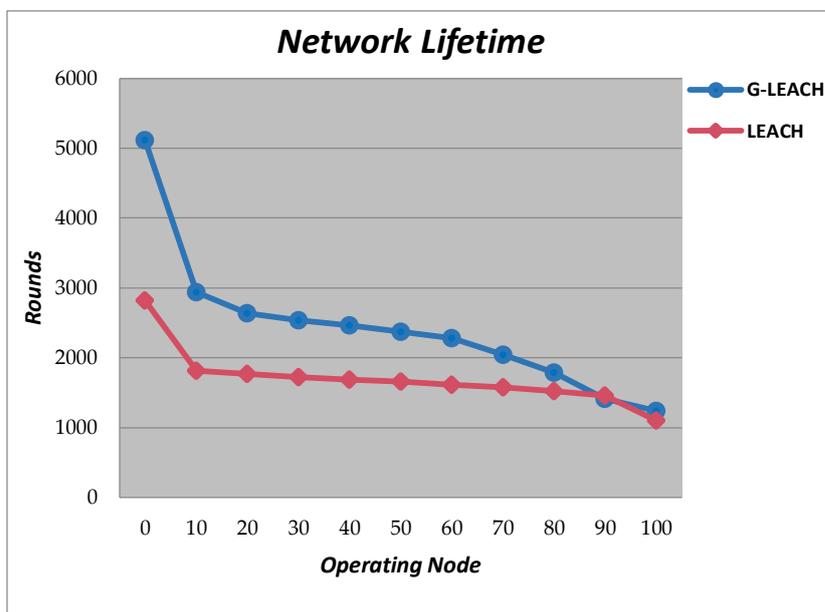
Eksperimen penelitian ini menggunakan Matlab dan beberapa parameter lainnya, terdiri atas 100 *node* yang ditempatkan secara acak dalam area 100 meter. Persentase CH yang didinginkan yaitu 0,1 dan total dari 4.000 bit paket dikirim oleh setiap *node* menuju *sink* melalui *merge CH* yang dipilih menggunakan parameter simulasi algoritma, seperti yang ditunjukkan pada Tabel 1. Setiap *node* akan menghabiskan energi per *round* dalam tingkat pertama model. Analisis kinerja masing-masing algoritma dilakukan berdasarkan total energi yang tersisa dan masa hidup *node* dan total energi konsumsi per *round*.

Tabel 1. Parameter model simulasi G-LEACH

Parameter	Kapasitas
Area simulasi	100 m × 100 m
Jumlah <i>grid</i>	16 <i>grid</i>
Jumlah <i>node</i>	100 <i>node</i>
<i>Eo</i>	1 <i>joule</i>
<i>Eelec</i>	50 × 10 <sup>-9</sup> <i>joule/byte</i>
<i>ETx</i>	50 × 10 <sup>-9</sup> <i>joule/byte</i>
<i>ERx</i>	50 × 10 <sup>-9</sup> <i>joule</i>
<i>Eamp</i>	100 × 10 <sup>-12</sup> <i>joule</i>
<i>EDA</i>	0 × 10 <sup>-9</sup> <i>joule</i>
Paket data ( <i>b</i> )	4.000 <i>byte</i>
<i>p</i>	0,1
<i>Sink</i>	50-100

##### 4.1. Hasil Perbandingan Network Lifetime protocol G-LEACH dan Protocol LEACH

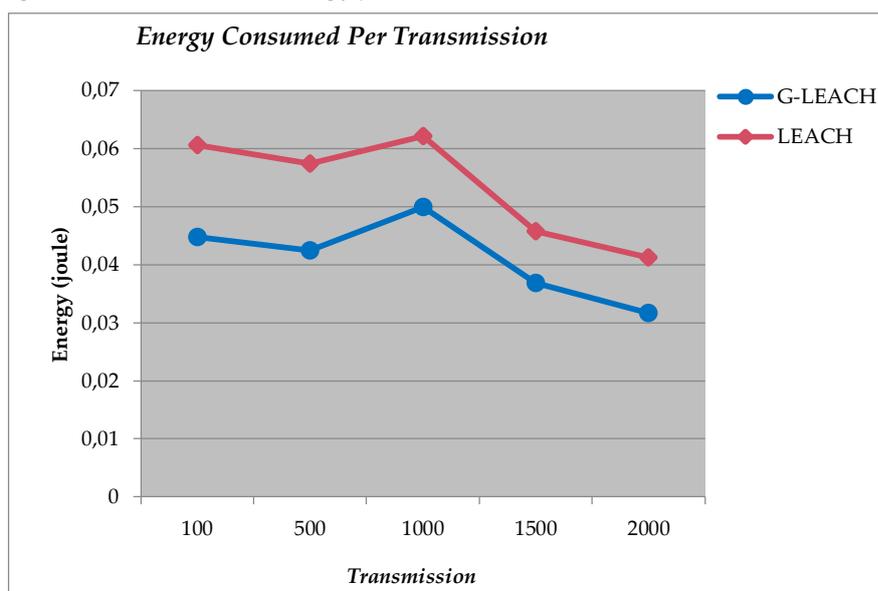
Berdasarkan hasil pengujian dalam membandingkan *network lifetime* pada protokol G-LEACH dan protokol LEACH, protokol G-LEACH memiliki *network lifetime* lebih tinggi dari protokol LEACH. Berdasarkan hasil uji coba masa hidup jaringan pada protokol G-LEACH mencapai 5.020, sedangkan masa hidup jaringan pada *routing protocol* LEACH *round* berkisar 2.941. Hasil simulasi perbandingan antara protokol G-LEACH dan LEACH terlihat seperti yang ditunjukkan pada Gambar 4.



Gambar 4. Network lifetime

Berdasarkan pengujian *network lifetime* pada protokol G-LEACH protokol LEACH, protokol G-LEACH memiliki *network lifetime* yang tinggi jika dibandingkan dengan protokol LEACH. Hal ini disebabkan karena pada protokol G-LEACH dalam setiap *grid* hanya CH yang melakukan pengiriman paket data. Semua paket data dikirim ke *merge* CH sehingga proses transmisi data lebih terpusat pada setiap area.

#### 4.2. Perbandingan Hasil Consumed Energy per Transmission



Gambar 5. Energy consumed per transmission

*Consumed energy per transmission* merupakan energi yang dikonsumsi semua *node* pada setiap transmisi antaranggota *node* dengan *merge* CH dan juga *merge* CH dengan *sink*. Hasil simulasi menunjukkan bahwa protokol G-LEACH memiliki *consumed energy* pada setiap transmisi yang lebih rendah dari protokol

LEACH. Pada hasil uji coba penelitian ini, tingginya *consumed energy per transmission* yang digunakan G-LEACH yaitu 0,05 joule per *transmission*, sedangkan pada protokol LEACH *tingginya consume energy per transmission* yang digunakan yaitu 0,06 joule per *transmission* seperti hasil pengujian yang ditunjukkan pada Gambar 5.

*Consumption energy* pada protokol G-LEACH lebih rendah dari protokol LEACH karena jarak antara anggota *node* dan *merge* CH lebih dekat, sehingga konsumsi energi setiap *node* dalam melakukan transmisi ke *merge* CH dan dari *merge* CH menuju *sink* lebih rendah.

## 5. Kesimpulan

Pengembangan *merge* CH pada *routing protocol* G-LEACH dengan menggunakan teknik *grid based clustering* dengan setiap *node* dalam setiap *grid* hanya melakukan transmisi melalui *merge* CH berdasarkan analisis hasil pengujian di atas, pada protokol G-LEACH *node* dengan *consumption energy* yang tinggi yaitu 0,049 pada saat melakukan transmisi data dapat menghemat energi 0,950156 joule dari energi yang diberikan yaitu 1 joule, sedangkan pada protokol LEACH dengan *node* yang tinggi konsumsi energi 0,0605208 joule dapat menghemat energi 0,939479 joule berdasarkan pengujian yang telah dilakukan, protokol G-LEACH dapat menghemat energi lebih tinggi daripada protokol LEACH. Sementara, pada selisih *consumption energy* antara protokol G-LEACH dan protokol LEACH yaitu pada *round* 100 selisih *consumption energy* yaitu 0,015761 joule, sedangkan pada *round* ke-500 selisih *consumption energy* 0,014934 joule, kemudian pada *round* 1.500 selisih *consumption energy* yaitu 0,008923 joule, dan pada *round* 2000 selisih *consumption energy* antara protokol G-LEACH dan protokol LEACH yaitu 0,009615. Berdasarkan pengujian yang telah dilakukan, protokol G-LEACH memiliki *network lifetime* yang panjang, *consumption energy per transmission* yang rendah, serta dapat meminimalkan konsumsi energi atau menghemat energi jika dibandingkan dengan protokol LEACH. Pada penelitian selanjutnya perlu dilakukan uji coba *merge* CH antar-grid atau *multi-hop* pada lingkungan WSN.

## 6. Ucapan Terima Kasih

Ucapan terima kasih kami sampaikan kepada seluruh pihak dosen Magister Teknik Informatika dan peneliti serta pengelola Laboratorium Institut Teknologi Sepuluh Nopember (ITS), Surabaya, atas bantuan hibah Penelitian yang diberikan dan dukungan terhadap penelitian kami, hingga kami dapat menjalankan penelitian ini dengan lancar tanpa kendala yang berarti.

## 7. Referensi

- Abushiba, W., Johnson, P., Alharthi, S., & Wright, C. (2017). An energy efficient and adaptive clustering for wireless sensor network (CH-leach) using leach protocol. *13th International Computer Engineering Conference (ICENCO)*. Cairo, Egypt : IEEE.
- Adhisa, R. R., & Wibisono, W. (2018). Energy Aware Multiple Cluster Head Selection and Stray Nodes Handling for LEACH Protocol in Wireless Sensor Network Environments. *2018 International Conference on Information Technology Systems and Innovation (ICITSI)*. Padang: IEEE.
- Amirthalingam, K., & Anuratha, A. (2016). Improved LEACH: A modified LEACH for Wireless Sensor Network. *International Conference on Advances in Computer Applications (ICACA)*. Coimbatore, India: IEEE .
- Astuti, L. D., & Wibisono, W. (2017). Peningkatan Networklifetimepada Wireless Sensor Network Menggunakan Clustered Shortest Geopath Routing (C-SGP) Protocol. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, 4(3), 148-15.
- Gupta, S., & Marriwala, N. (2017). Improved distance energy based LEACH protocol for cluster head election in wireless sensor networks. *4th International Conference on Signal Processing, Computing and Control (ISPCC)*. Solan, India: IEEE.
- Hani, R. M., & Ijeh, A. A. (2013). A Survey on LEACH-Based Energy Aware Protocols for Wireless Sensor Networks. *Journal of Communications*, 8(3), 192-206.
- Heinzelman, W., Chandrakasan, A., & Balakrishnan, H. (2000). Energy-efficient communication protocol for wireless microsensor networks. *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*. Maui, HI, USA, USA: IEEE.

- Kaur, J., Gaba, G. S., Miglani, R., & Pasricha, R. (2015). Energy Efficient and Reliable WSN based on Improved Leach-R Clustering Techniques. *Indian Journal of Science and Technology*, 8(16).
- Li, W., & Shen, W. (2011). Swarm behavior control of mobile multi-robots with wireless sensor networks. *Journal of Network and Computer Applications*, 34(4), 1398-1407.
- Li, Y., Ding, L., & Liu, F. (2011). The improvement of LEACH protocol in WSN. *International Conference on Computer Science and Network Technology*. Harbin, China: IEEE.
- Mahajan, M., Reddy, K., & Rajput, M. (2016). Design and Simulation of a Blacklisting Technique for Detection of Hello flood Attack on LEACH Protocol. *7th International Conference on Communication, Computing and Virtualization 2016* (pp. 675-682). Mumbai, India: Elsevier.
- Manzoor, B., Javaid, N., Rehman, O., Akbar, M., Nadeem, Q., Iqbal, A., & Ishfaq, M. (2013). Q-LEACH: A New Routing Protocol for WSNs. *International Workshop on Body Area Sensor Networks (BASNet-2013)* (pp. 926-931). Halifax, Nova Scotia, Canada: Elsevier.
- Palan, N. G., Barbadekar, B. V., & Patil, S. (2017). Low energy adaptive clustering hierarchy (LEACH) protocol: A retrospective analysis. *2017 International Conference on Inventive Systems and Control (ICISC)*. Coimbatore, India: IEEE.
- Rasyid, M. U., Lee, B.-H., Syarif, I., & Arkham, M. M. (2018). LEACH Partition Topology for Wireless Sensor Network. *International Conference on Consumer Electronics-Taiwan (ICCE-TW)*. Taichung, Taiwan: IEEE.
- Sharma, R., Mishra, N., & Srivastava, S. (2015). A proposed Energy Efficient Distance Based Cluster Head (DBCH) Algorithm: An Improvement over LEACH. *Procedia Computer Science*, 57, 807-814.
- Sivakumara, P., & Radhika, M. (2018). Performance Analysis of LEACH-GA over LEACH and LEACH-C in WSN. *6th International Conference on Smart Computing and Communications, ICSCC 2017*. 125, pp. 248–256. Kurukshetra, India: Elsevier.