**OPTIMASI LEACH *PROTOCOL* DENGAN METODE *K-MEANS++* *CLUSTERING* PADA *WIRELESS SENSOR NETWORK* (WSN)**

**SKRIPSI**

****

**PUTU MAS ANGGITA PUTRA**

**NIM. 1708561007**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS UDAYANA**

**JIMBARAN**

**2020**

**SURAT PERNYATAAN KEASLIAN KARYA ILMIAH**

Yang bertanda tangan di bawah ini menyatakan bahwa naskah Skripsi dengan judul:

**OPTIMASI LEACH *PROTOCOL* DENGAN METODE *K-MEANS++* *CLUSTERING* PADA *WIRELESS SENSOR NETWORK* (WSN)**

Nama : Putu Mas Anggita Putra

NIM : 1708561007

Program Studi : Teknik Informatika

E-mail : [anggitaputra13@gmail.com](mailto:anggitaputra13@gmail.com)

Nomor telp/HP : 0895370008969

Alamat : Jl. RY Sesetan Gang VI No 9.B, Br.Ambengan, Pedungan,

Denpasar Selatan, Bali

Belum pernah dipublikasikan dalam dokumen skripsi, jurnal nasional maupun internasional atau dalam prosiding manapun, dan tidak sedang atau akan diajukan untuk publikasi di jurnal atau prosiding manapun. Apabila di kemudian hari terbukti terdapat pelanggaran kaidah – kaidah akademik pada karya ilmiah saya, maka saya bersedia menanggung sanksi-sanksi yang dijatuhkan karena kesalahan tersebut, sebagaimana diatur oleh Peraturan Menteri Pendidikan Nasional Nomor 17 Tahun 2010 tentang Pencegahan dan Penanggulangan Plagiat di Perguruan Tinggi.

Demikian Surat Pernyataan ini saya buat dengan sesungguhnya untuk dapat dipergunakan bilamana diperlukan.

Denpasar, x Bulan 2021

Yang membuat pernyataan,

Putu Mas Anggita Putra

NIM. 1708561007

**LEMBAR PENGESAHAN**

Judul : Optimasi Leach *Protocol* Dengan Metode *K-Means++* *Clustering*

Pada *Wireless Sensor Network* (WSN)

Nama : Putu Mas Anggita Putra

Pembimbing : 1. I Gusti Agung Gede Arya Kadnyanan, S.Kom., M.Kom

2. Dr. Ngurah Agus Sanjaya ER, S.Kom., M.Kom

**ABSTRAK**

Title : Optimasi Leach *Protocol* Dengan Metode *K-Means++* *Clustering*

Pada *Wireless Sensor Network* (WSN)

Name : Putu Mas Anggita Putra

Supervisor : 1. I Gusti Agung Gede Arya Kadnyanan, S.Kom., M.Kom

2. Dr. Ngurah Agus Sanjaya ER, S.Kom., M.Kom

**ABSTRACT**

**KATA PENGANTAR**

Proposal penelitian ini dengan judul Optimasi Leach Protokol dengan Metode *K-Means++ Clustering* Pada *Wireless Sensor Network* (WSN) ini disusun dalam rangkaian kegiatan pelaksanaan Tugas Akhir di Program Studi Informatika FMIPA UNUD. Proposal ini disusun dengan harapan dapat menjadi pedoman dan arahan dalam melaksanakan penelitian di atas.

Sehubungan dengan telah terselesaikannya proposal ini, maka diucapkan terima kasih dan penghargaan kepada berbagai pihak yang telah membantu pengusul, antara lain:

1. Bapak I Gusti Agung Gede Arya Kadyanan, S.Kom., M.Kom sebagai Pembimbing I yang telah banyak membantu menyempurnakan proposal ini.
2. Bapak Dr. Ngurah Agus Sanjaya ER, S.Kom., M.Kom sebagai Pembimbing II yang telah banyak membantu menyempurnakan proposal ini.
3. Bapak I Komang Ari Mogi, S.Kom., M.Kom. sebagai pembimbing di penjurusan jaringan sensor nirkabel yang telah banyak membantu menyempurnakan proposal ini,
4. Bapak-bapak dan ibu-ibu dosen di Program Studi Informatika, yang telah meluangkan waktu turut memberikan saran dan masukan dalam penyempurnaan proposal ini;
5. Kawan-kawan di Program Studi Informatika yang telah memberikan dukungan moral dalam penyelesaian proposal ini.

Disadari pula bahwa sudah tentu proposal ini masih mengandung kelemahan dan kekurangan. Memperhatikan hal ini, maka masukan dan saran-saran penyempurnaan sangat diharapkan.

Jimbaran, xxx 2021

Penyusun

**DAFTAR ISI**

**DAFTAR TABEL**

**DAFTAR GAMBAR**

**DAFTAR LAMPIRAN**

**BAB I**

**PENDAHULUAN**

* 1. **Latar Belakang**

Perkembangan teknologi di bidang komunikasi dewasa ini sangatlah pesat. Salah satu teknologi yang banyak diterapkan saat ini adalah *Wireless Sensor Network* (WSN). *Wireless Sensor Network* dapat terdiri atas ratusan hingga ribuan *node*, dimana setiap *node* memiliki kemampuan untuk *sensing, processing data* dan *wireless communication.* Masing-masing *node* menggunakan baterai sebagai *supply node sensor* yang menyebabkan penggunaan energi menjadi salah satu point penting dalam membangun WSN. Penggantian baterai pada *node sensor* menjadi sulit untuk dilakukan mengingat *node sensor* diletakkan secara acak dalam jaringan dan dalam kondisi tertentu sulit untuk diketahui posisi dari semua *node sensor* yang ada pada WSN. Tentunya hal tersebutnya menyebabkan terganggunya stabilitas jaringan pada WSN dan mempengerahui kinerja jaringan dalam melakukan *sensing* atau *processing data* (Tonapo & Ratna W, 2016). Oleh sebab itu, diperlukan solusi lain untuk mengatasi keterbatasan energi pada WSN, yaitu dengan memilih *routing protocol* yang hemat energi.

*Routing protocol* adalah metode untuk menentukan jalur terdekat dalam proses *transfer dat*a dari *node* menuju *base station*. Banyak penelitian dilakukan untuk menganalisis penggunaan *routing protocol* yang tepat dalam mengoptimalkan penggunaan energi pada WSN. Algoritma pertama yang dikembangkan adalah *routing protocol* *Low-Energy Adaptive Clustering Hierarchy* (LEACH). Namun Algoritma LEACH memiliki beberapa kekurangan diantaranya seperti sisa energi *node* tidak dipertimbangkan dalam proses pemilihan *Cluster head* (CH). Oleh karena itu, kegagalan CH karena energi sisa yang rendah tidak dapat dihindari. Kemudian dalam pemilihan CH juga tidak mempertimbangkan jarak antara CH dengan *Base Station*. Sehingga konsumsi daya pada CH yang terletak jauh dari *Base Station* lebih tinggi dibandingkan yang di dekat *Base Station* (Ridwan dkk, 2020).

Pada penelitian sebelumnya oleh (Saheb & Sharma, 2017) menggunakan algoritma *K-means Clustering* untuk mengoptimalisasi pemilihan *cluster head* dalam sebuah klaster untuk membuat klaster yang seragam dan dapat mengurangi jarak antara *cluster head* dengan *node* lainnya. Hasil Penelitian ini menunjukan klaster yang dihasilkan simetris dan secara signifikan mengurangi jarak antara *node* dan *cluster head* yang mana memiliki konsumsi energi node yang seimbang dengan peningkatan masa hidup jaringan.

Oleh sebab itu, pada penelitian ini bertujuan untuk mengoptimasi protokol LEACH dengan menggunakan algoritma pengembangan dari *k-means*, yaitu dengan menggunakan algoritma *K-means++* untuk menjadi algoritma LEACH yang baru yaitu KM-LEACH. KM-LEACH merupakan optimasi dari protokol LEACH dimana mengoptimalkan algoritma *K-means++ Clustering* untuk lebih mengoptimalkan pemilihan *cluster head* yang akan dipilih berdasarkan energi yang tersisa dan jarak *node* dengan base station. Sehingga diharapkan dengan pengoptimalan pemilihan *cluster head*, dapat mengurangi masalah tidak meratanya distribusi *cluster head,* konsumsi energi yang tidak merata dan meningkatkan stabilitas energi dalam *Wireless Sensor Network* (WSN).

* 1. **Rumusan Masalah**

Rumusan masalah yang akan dijadikan acuan dalam penelitian ini yaitu:

1. Bagaimana pengaruh penerapan KM-LEACH terhadap efisiensi penggunaan energi (total konsumsi energi/Joule) pada WSN
2. Bagaimana kinerja perbandingan dari protokol LEACH optimasi dengan *k-means* dan KM-LEACH dioptimasi dengan *k-means++* yang sudah dioptimasi dalam segi total konsumsi energi, jumlah node yang mati dan jumlah node hidup per *round* pada WSN.
   1. **Batasan Masalah**

Agar penelitian tidak keluar dari topik penelitian, maka penelitian akan dibatasi sebagai berikut:

1. Simulasi dilakukan terbatas hanya dengan protokol *routing* LEACH untuk optimalisasi serta hanya menggunakan algoritma *K-Means++ Clustering*
2. Penelitian akan dilakukan secara simulasi menggunakan program Matlab 2015a.
3. Parameter kinerja yang diukur dan diamati adalah penggunaan total konsumsi energi (Joule), jumlah *node* yang mati dan jumlah *node* hidup per *round*.
4. Perancangan WSN hanya menggunakan 100 *node* sensor + 1 *base station* yang disimulasikan.
5. Pengujian dilakukan selama 100 *round.*
   1. **Tujuan Penelitian**

Adapun tujuan dari dilakukannya penelitian ini yaitu:

1. Menerapkan algoritma *K-means++ Clustering* sebagai optimalisasi dari Protokol LEACH serta mengetahui pengaruhnya terhadap efisiensi penggunaan energi (total konsumsi energi/Joule) pada WSN
2. Membandingkan kinerja protokol LEACH dioptimasi dengan *k-means* dengan KM-LEACH dioptimasi dengan *k-means++* dalam segi total konsumsi energi, jumlah *node* yang mati dan jumlah mode hidup per *round* pada WSN.
   1. **Manfaat Penelitian**

Manfaat yang diperoleh dari penelitian ini yaitu:

1. Bagi penulis

Hasil penelitian ini nantinya diharapkan dapat memberikan gambaran tentang kinerja dari protokol LEACH dioptimasi dengan *k-means* dan protokol LEACH dioptimasi dengan *k-means++* (KM-LEACH) pada *wireless sensor network.*

1. Bagi pihak lain

Hasil penelitian ini nantinya diharapkan dapat bermanfaat, menjadi referensi untuk menambah wawasan tentang *Wireless Sensor Network* khususnya mengenai penggunaan protokol LEACH terhadap serta bagaimana protokol ini bekerja.

* 1. **Metodelogi Penelitian**

Pada metodologi penelitian akan dijelaskan langkah-langkah dalam penyusunan penelitian yang akan dibuat mengenai Optimasi LEACH *Protocol* Dengan Metode *K-Means++ Clustering* Pada *Wireless Sensor Network* (WSN). Sub bab bahasan yang akan dijelaskan meliputi pengumpulan data, desain penelitian, dan metode yang digunakan.

* + 1. **Analisis Sistem**

Pada tahap ini dilakukan identifikasi perkiraan terkait kebutuhan yang diperlukan terkait penelitian yang akan dilakukan. Kebutuhan yang dimaksudkan terkait materi penunjang penelitian terkait penerapan routing protokol LEACH, penggunaan algoritma K-Means dan K-Means++ serta perancangan proses simulasi yang akan dilakukan dalam penelitian ini.

* + 1. **Pengumpulan Data**

Data yang akan digunakan dalam penelitian ini diperoleh dari hasil perbandingan simulasi pada aplikasi MATLAB 2015a dengan beberapa parameter uji untuk mendapatkan data:

* 1. Total konsumsi energi (Joule)
  2. Jumlah node hidup selama 100 round
  3. Jumlah death node selama 100 round

Dimana data yang didapat menggunakan skenario uji berdasarkan parameter dari penelitian sebelumnya oleh (Saheb & Sharma, 2017). Dimana nantinya parameter simulasi tersebut akan digunakan untuk menjadi skenario simulasi untuk mendapatkan data primer pada penelitian ini. Dimana skenario yang dirancang akan menggunakan 100 node yang disebar pada titik kordinat x dan y dengan luas daerah 100 m^2 x 100 m^2 untuk membuat skema topologi jaringan WSN.

* + 1. **Desain Penelitian**

Pada penelitian ini menggunakan proses simulasi yang merupakan suatu Teknik untuk meniru suatu proses-proses yang terjadi pada suatu sistem dengan bantuan perangkat komputer serta dilandasi dengan beberapa asumsi-asumsi tertentu sehingga sistem tersebut dapat dipelajari secara ilmiah (Kelton, 1991). Proses simulasi yang akan dilakukan dalam penelitian ini akan terdiri dari 2 proses utama yaitu proses *Setup Phase* atau pembentukan klaster dan proses *Steady State* atau transmisi data. Pada fase *Setup Phase* akan dimodifikasi dengan menggunakan algoritma yang ditentukan dalam penelitian itu yaitu algoritma *K-Means* dan *K-Means*++.

Kemudian setelah dilakukan proses simulasi akan didapatkan hasil dari penggunaan kedua metode yang nantinya akan dibandingkan hasilnya dengan menggunakan beberapa parameter uji untuk mengetahui metode mana yang lebih efisien dalam penggunaan energi diantara kedua metode LEACH yang dimodifikasi dengan algoritma *K-Means* dan *K-Means++* tersebut.

* + 1. **Metode Yang Digunakan**

Metode yang digunakan dalam penelitian ini adalah algoritma *K-Means* dan algoritma *K-Means++* dalam memodifikasi protokol LEACH. Kedua algoritma tersebut diterapkan pada proses *setup phase* dalam protokol LEACH. Dalam proses setup phase atau pembentukan klaster, algoritma *K-Means* dan *K-Means++* memiliki perbedaan yaitu dalam algoritma *K-Means* dalam penentuan klaster diawal dilakukan secara acak sedangkan pada algoritma *K-Means++* hanya penentuan klaster pertama yang dilakukan secara acak, kemudian klaster lainnya akan dipilih berdasarkan jarak terjauh dengan klaster sebelumnya dengan menggunakan rumus *ecludien distance.* Hasil klaster yang akan didapatkan antara 2 algoritma tentunya berbeda sehingga akan berdampak juga dalam proses transmisi energi pada proses steady state.

* + 1. **Skenario Pengujian**

Skenario pengujian merupakan salah satu hal yang penting dalam pelaksanaan sebuah penelitian. Dengan adanya skenario pengujian, penelitian dapat berjalan sesuai dengan apa yang direncanakan dan diharapkan akan mendapatkan hasil yang sesuai. Dalam penelitian ini seteleh dilakukan proses simulasi akan didapatkan hasil penelitian yaitu:

* Total Konsumsi Energi (Joule)
* Jumlah Node Hidup selama 100 round
* Jumlah Node Mati selama 100 round

Kemudian hasil tersebut akan dibandingkan antara keduan metode yaitu LEACH dengan algoritma *K-Means* dan LEACH dengan algoritma *K-Means++* untuk diketahui protokol LEACH mana yang lebih efisien dan hemat energi pada *Wireless Sensor Network* (WSN).

**BAB II**

**TINJAUAN PUSTAKA**

Pada bagian ini, akan dipaparkan sejumlah tinjauan empiris dan tinjauan teoritis yang dijadikan sebagai acuan dalam penelitian mengenai optimasi LEACH *Protocol* dengan metode *K-Means Clustering* pada *wireless sensor network* (WSN) yang dilakukan ini.

* 1. **Tinjauan Empiris**

Pada penelitian ini digunakan beberapa acuan yang bersumber dari penelitian terdahulu baik dari segi metode ataupun jenis penelitian yang dilakukan. Beberapa penelitian terkait yang penulis jadikan acuan antara lain:

1. ***Improved LEACH Protocol Based on K-Means Clustering Algorithm for Wireless Sensor Network*** (Saheb & Sharma, 2017)

Pada penelitian ini menggunakan algoritma K-means Clustering untuk mengoptimalisasi pemilihan cluster head dalam sebuah klaster untuk membuat klaster yang seragam dan dapat mengurangi jarak antara cluster head dengan node lainnya. Penelitian ini juga menambahkan parameter residual energy dalam penentuan cluster head. Penelitian ini menggunakan 100 node yang disimulasikan pada software MATLAB dimana node disebar pada titik kordinat dengan luas area 100 x 100 m^2. Dimana simulasi berdasarkan 2 parameter uji yaitu jumlah node yang hidup dan total konsumsi energi untuk membandingkan kinerja protokol LEACH dengan LEACH yang dioptimasi dengan algoritma K-Means.

Hasil Penelitian ini menunjukan cluster yang dihasilkan simetris dan secara signifikan mengurangi jarak antara node dan cluster head yang mana meliki konsumsi energi node yang seimbang dengan peningkatan masa hidup jaringan. Persamaan dengan penelitian yang akan dilakukan adalah sama-sama menggunakan algoritma k-means untuk mengatasi masalah penentuan cluster head yang optimal. Perbedaannya terletak pada pada penelitian yang akan dilakukan menggunakan algoritma pengembangan dari k-means yaitu algoritma k-means++.

1. **Optimasi Protokol LEACH Untuk Meningkatkan Stabilitas Pada *Wireless Sensor Network*** (Ridwan dkk, 2020)

Pada penelitian ini mencoba mengatasi kekurangan dari protokol LEACH yaitu pemilihan cluster head yang tidak mempertimbangkan energi sisa di setiap node, sehingga dapat menyebabkan kegagalan dalam proses pemilihan kepala klaster dan akan sangat mempengaruhi stabilitas pada jaringan. Solusi yang ditawarkan penulis yaitu metode Optimization-LEACH (O-LEACH) dimana proses pemilihan cluster head dikembangkan berdasarkan energi awal setiap node dan tingkat energi awal yang baru dihitung dari setiap node untuk setiap putaran.

Hasil penelitian menunjukan bahwa protokol LEACH biasa hanya mampu mempertahankan 2 % dan metode O-LEACH 40 % dari 200 node yang disimulasikan sehingga metode O-LEACH lebih optimal. Persamaan dengan penelitian yang akan dilakukan adalah sama-sama ingin mengoptimalisasi protokol LEACH dalam penentuan cluster head untuk meningkatkan masa hidup jaringan. Perbedaannya terletak pada metode yang digunakan, dimana dalam penelitian ini dalam pemilihan cluster head berdasarkan energi awal setiap node dan tingkat energi awal setiap node setiap putaran.

1. **Enhanced LEACH Protocol For Increasing a Lifetime Of WSNs** (Salem & Shudifat, 2019)

Pada penelitian ini ingin meningkatkan masa hidup jaringan dengan mengoptimalisasi pemilihan cluster head dalam protokol LEACH, dimana dalam pemilihan cluster head dengan protokol LEACH biasa dilakukan secara random namun dalam penelitian ini pemilihan cluster head dimodifikasi dengan mempertimbangkan beberapa parameter seperti jarak cluster head dengan base station dan energi yang ada pada node. Penelitian ini menggunakan software MATLAB untuk simulasi dengan menggunakan 45-85 node pada titik kordinat dengan luas area 100 x 100 m^2.

Hasil penelitian ini menunjukkan bahwa konsumsi daya diminimalkan, masa pakai jaringan dapat ditingkatkan, jumlah putaran dimaksimalkan, dan konsumsi daya yang menurun. Akhirnya, dibandingkan dengan protokol LEACH, pengembangan protokol LEACH ini dapat bekerja lebih baik. Persamaan dengan penelitian yang akan dilakukan terletak pada parameter yang digunakan sebagai acuan penentuan cluster head yaitu jarak antara base station dengan cluster head dan energi yang ada pada node. Perbedaannya pada penelitian ini tidak memperhitungkan jarak antara cluster head dengan node yang ada dalam klaster yang sama pada penentuan cluster head sedangkan pada penelitian yang akan dilakukan melakukannya dengan metode k-means++.

1. **Perbandingan Pengaruh Nilai Centroid Awal Pada Algoritma K-Means dan K-Means++ Terhadap Hasil Cluster Menggunakan Metode Confusion Matrix** (Nasir & Budiman, 2017)

Pada penelitian ini membandingkan tingkat akurasi antara 2 jenis algoritma, yaitu algoritma k-means dan k-means++ pada klaster. Penelitian ini dilakukan untuk mengetahui bagaimana pengaruh dari pemilihan centroid awal yang berbeda dari kedua algoritma tersebut terhadap hasil klaster. Evaluasi hasil klaster yang digunakan menggunakan metode confusion matrix untuk menghitung tingkat akurasi.

Dari hasil penelitian didapatkan bahwa kedua algoritma mempunyai hasil akhir klaster dapat berubah-ubah. Hal ini dibuktikan dari percobaan yang dilakukan 100 kali pada kedua algoritma dengan data set iris, algoritma k-means mempunyai 6 hasil klaster yang berbeda dan algoritma k-means++ mempunyai 5 hasil klaster berbeda. Algoritma k-means++ Mempunyai rata-rata akurasi yang lebih tinggi dari pada algoritma k-means. Rata-rata tingkat akurasi algoritma k-means yaitu 80,46 % dan pada algoritma k-means++ yaitu 83,66. Algoritma k-means++ mempunyai peluang lebih besar untuk mendapatkan hasil klaster terbaik, dari percobaan 100 kali, algoritma k-means++ terdapat 48 percobaan sedangkan algoritma k-means terdapat 19 percobaan.

Persamaan dengan penelitian yang akan dilakukan adalah metode yang digunakan yaitu k-means++, dimana hasil penelitian metode k-means++ memiliki peluang lebih baik dalam pembentukan klaster dibandingkan metode k-means biasa. Hal itu yang menjadi acuan dalam penelitian yang akan dilakukan yaitu dengan metode k-means++ untuk mengurangi kekurangan pada protokol LEACH. Perbedaanya terletak pada studi kasus permasalahan yang akan diteliti pada penelitian ini dengan penelitian yang akan dilakukan.

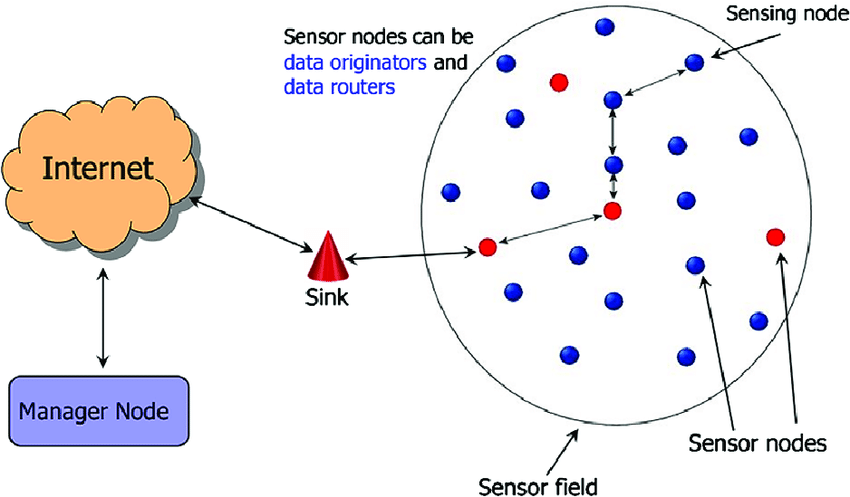
* 1. **Tinjauan Teoritis**

Pada penelitian ini digunakan beberapa teori yang digunakan serta dijadikan acuan dalam proses penelitian, antara lain:

1. Wireless Sensor Network

Wireless sensor network merupakan sekumpulan node yang dapat berupa sensor yang akan melakukan pengambilan data pada parameter ukur dan kemudian dikirimkan pada sebuah node sentral atau sebuah server untuk dilakukan pengolahan data. Setiap node sensor memiliki kemampuan untuk mengumpulkan data di sekitarnya dan merutekannya kembali menuju sink node melalui transmisi radio secara intensif. Data yang dapat dikumpulkan oleh sensor node bisa berupa tekanan suhu, pergerakan suatu benda atau object, kelembaban tanah dan sebagainya (Ridwan dkk, 2020).

Dimana pada umumnya wireless sensor network (WSN) terdiri dari dua bagian utama yaitu node sensor dan sink (Tonapa dkk, 2016). Node sensor merupakan suatu perangkat yang dapat menghasilkan informasi, biasanya merupakan sebuah sensor atau juga dapat berupa sebuah aktuator yang menghasilkan feedback pada keseluruhan operasi. Secara umum node sensor disebar dengan volume dan kerapatan yang tinggi. Sink merupakan kesatuan alat yang berfungsi pengumpulan informasi dari node sensor sehingga dapat dilakukan pengolahan informasi lebih lanjut sebelim data disimpan ke server/clound. Dengan adanya karakteristik tersebut, perlu adanya metodologi yang mampu melampaui dan mengembangkan karakteristik-karakteristik pada WSN serta tidak membatasi pengiriman informasi, jaringan, manajemen operasional, kerahasiaan, integritas, dan proses di dalam jaringan.

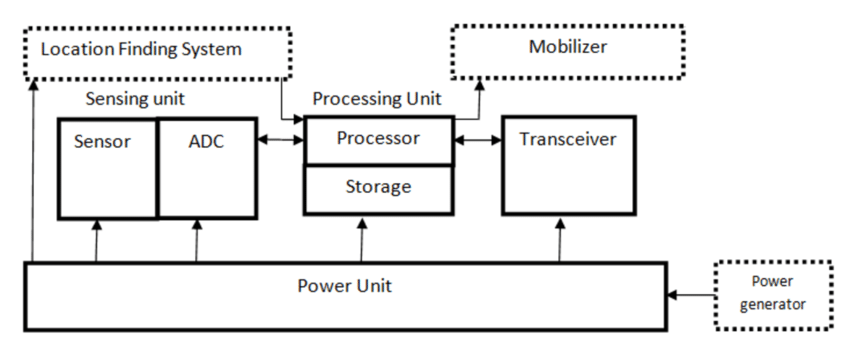


Gambar 2.1 Skema WSN

(Sumber: Mihobi dkk, 2017)

1. Node Sensor

Node dalam wireless sensor network dikenal dengan istilah “mote”. Dimana pada dasarnya node adalah sebuah komputer, walaupun memiliki bentuk dan kemampuannya tidak seperti umumnya komputer yang digunakan saat ini karena kemampuan yang masih terbatas dan ukurannya yang cukup kecil (smart dust), tetapi fungsi node sama seperti komputer pada umumnya dimana sampai saat ini terus mengalami perkembangan untuk meningkat kemampuan dari node (Sohraby dkk, 2011). Node dilengkapi alat pemroses (CPU), memori, sejumlah antarmuka Input/Output yang dapat diprogram ulang (terintegrasi pada mikrokontroler), transceiver untuk komunikasi radio, sumber daya energi yang menggunakan baterai, dan beberapa peralatan tambahan yang dapat disertakan sesuai kebutuhan penggunaan node itu sendiri.



Gambar 2.2 Struktur Pada Node

(Sumber: Sohraby dkk, 2011. Wireless Sensor Network Technology, Protocols, Applications Book. New Jersey:John Wiley & Sons, Inc)

Pada wireless sensor network, node memiliki fungsi dan kemampuan yang berbeda-beda, berikut beberapa jenis node dalam WSN (Sohraby dkk, 2011):

1. Sensor Node

Sensor node yaitu node yang berfungsi untuk membaca data lingkungan atau objek yang dipantau. Untuk keperluan pembacaan atau penginderaan, node ini dilengkapi dengan satu atau beberapa perangkat sensor. Dari kemampuannya, node ini dapat dibagi menjadi dua jenis. Pertama, Node dengan kemampuan standar dan kedua yaitu Node yang telah dilengkapi fasilitas yang lebih kaya seperti CCD camera, wireless LAN, logger, Web server dan lain-lain. Node jenis kedua ini juga mampu melakukan komputasi yang lebih kompleks dibanding jenis pertama.

1. Router

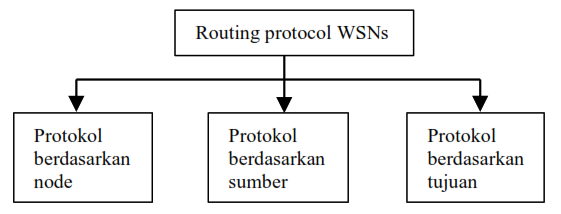
Router adalah mini komputer/node yang berfungsi untuk meneruskan paket data dari sebuah node ke node lain dan menghubungkan antara 2 jaringan yang berbeda. Node ini berguna untuk keperluan komunikasi multi-hop. Dalam aplikasi nyata, sebuah sensor node dapat deprogram untuk bertindak sebagai router.

1. Sink Node

Sink node adalah node yang berfungsi untuk mengumpulkan data penginderaan dari sensor node, kemudian meneruskannya ke perangkat lain, seperti ke database server untuk penyimpanan. Selain untuk mengumpulkan data dari sensor node, sink juga berfungsi sebagai penyebar paket dari perangkat atau sistem lain ke WSN, misalnya untuk keperluan pemrograman atau konfigurasi ulang sensor node secara remote.

1. Routing Protokol

*Routing* *protocol* merupakan suatu aturan-aturan mengenai *routing* atau aktifitas penting dalam suatu jaringan untuk memilih antara 2 *node* atau lebih agar bisa saling berkomunikasi satu sama lain. Aktifitas ini dikerjakan oleh *routing* *protocol* pada *router* dengan cara memilih jalur terbaik untuk menghubungkan *node-node* yang ada untuk berkomunikasi dan transfer data.



Gambar 2.3 Routing Protokol

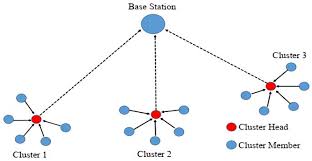
(Sumber: Novid dkk, 2019)

Pada WSN routing protocol dapat dibedakan menjadi 3 jenis yaitu routing protocol berdasarkan node, routing protocol berdasarkan sumber dan routing protocol berdasarkan tujuan (Novid dkk, 2019). Protokol berdasarkan node yang dimaksud adalah dalam penerapan routing protokol node-node yang ada dalam suatu WSN diklasifikasikan menjadi beberapa struktur hirarki node. Dimana setiap node akan terbagi menjadi beberapa cluster-cluster yang ada untuk membantu proses komunikasi antar node yang ada. Salah satu protokol yang menggunakan metode ini adalah LEACH (Low Energy Adaptive Clustering Hierarchy)

Protokol berdasarkan sumber hanya bekerja ketika protokol diminta untuk mengirimkan data. Routing protocol akan dibuat oleh node sumber hingga data yang dikirim mencapai node tujuan. Protokol yang menggunakan routing protocol jenis ini adalah SPIN (Sensor Protocol for Information via Negotiation). Sedangkan protokol yang menggunakan routing protocol berdasarkan tujuan ini adalah suatu protokol yang sudah memilih jalur yang tetap, sehingga tidak perlu mengirimkan sinyal pemberitahuan terlebih dahulu sebelum melakukan pengiriman data. Dengan itu proses pengiriman data akan menjadi lebih efisien dan sumber daya yang digunakan juga akan lebih sedikit. Contoh protokol dengan routing protocol berdasarkan tujuan yakni protokol Direct Diffusion (DD).

1. LEACH Protokol

LEACH merupakan salah satu routing protocol yang digunakan pada wireless sensor network (WSN). Protokol ini dimulai dengan pemilihan suatu node sebagai cluster head (CH) lalu dengan algoritma clustering memilih node non-CH sebagai anggota sehingga membentuk klaster. Mekanisme ini menghemat energi karena hanya CH yang transmisi data ke base station, sedangkan tiap node sensor cukup mengirim data ke CH masing-masing. Akibatnya, konsumsi energi berkurang. sehingga lifetime jaringan sensor menjadi maksimal (Permana dkk, 2015). Seperti yang direpresentasikan pada gambar 2.4.



Gambar 2.4 Skema LEACH Protokol

(Sumber: Ridwan dkk, 2020)

Pada awalnya node-node tersebar dalam jumlah besar pada suatu area dan proses pengiriman data masih terpusat pada base station. Namun dengan adanya algoritma LEACH, node-node tersebut dikelompokkan dalam beberapa klaster pada satu jaringan. Masing-masing klaster memiliki sebuah cluster head yang bertugas untuk mengkoordinasi pengiriman data dari node sensor ke base station.

1. Fitur-Fitur LEACH Protokol

Leach Protokol memiliki beberapa fitur diantaranya (Salem & Shudifat, 2019):

1. Data fusion, yaitu penggabungan data sehingga mengurangi disipasi energi dan menambah lifetime jaringan.
2. Adaptive, yaitu mudah menyesuaikan diri saat pembentukan formasi klaster.
3. Local Compression, yaitu mengkompres data agar ukuran data yang dikirim ke base station lebih kecil.
4. Randomization Rotation, yaitu perputaran kedudukan cluster head secara acak.
5. Self-Organizing, yaitu tiap node sensor memiliki sikap pengambilan keputusan sendiri untuk menjadi cluster head.
6. Algoritma LEACH

LEACH terbagi ke dalam beberapa sesi, tergantung dari jumlah cluster head yang diinginkan dan masa observasi. LEACH memastikan setiap node akan menjadi cluster head untuk satu sesi. Akibatnya, kedudukan cluster head menjadi tidak tetap atau bergantian sehingga suatu klaster memiliki formasi yang dinamis atau berubah-ubah setiap sesi. Algoritma LEACH dibagi menjadi 2 fase yaitu fase setup dan fase steady state (Heinzelman dkk, 2002). Proses algoritma LEACH dapat dijelaskan sebagai berikut :

1. Fase Setup

Pada *fase setup* terjadi penentuan *cluster head* dan proses pembentukan klaster atau sering disebut juga dengan algoritma *clustering*. Pada fase ini beberapa *node* secara *independent* akan memilih dirinya sendiri sebagai *cluster head* berdasarkan pada level energi terkininya dan nilai *threshold* T(n) yang dirumuskan dengan:

|  |  |
| --- | --- |
|  | (8) |

Dimana P adalah presentasi *cluster head* yang diinginkan dan G adalah sekumpulan *node* yang tidak menjadi *cluster head* dan r adalah putaran saat ini dan pada 1/P *round* terakhir (Ibrian dkk, 2020). Dimana dalam pemilihan *cluster head* berdasarkan residual energi yang dihitung dengan rumus:

|  |  |
| --- | --- |
|  | (9) |

Dimana:

**=** Residual energi setiap *node* selama *round*

***=*** Rata-rata energi setiap *node* selama *round*

***=*** Total Residual energi selama *round*

Untuk menghitung residual energi pada masing-masing *node* yang tersisa dengan rumus (Saheb & Sharma, 2017):

|  |  |
| --- | --- |
|  | (10) |

= *Round* saat ini

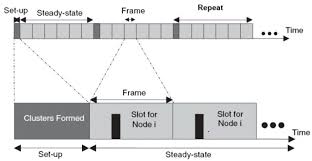
= Maksimum *round* pada simulasi

= Jumlah *node*

Pemilihan *cluster* tersebut meningkatkan jarak transmisi agregat antara *node* dan *cluster head* dan ini membuat lebih banyak energi yang dihabiskan dalam komunikasi *intra-cluster*. Untuk mengatasi masalah ini, mereka harus dipilih sebagai *cluster head* yang dekat dengan pusat *cluster*.

1. Fase Steady State

Pada fase ini terjadi proses transfer data antar node yang melibatkan aktivitas transmisi dan observasi. Proses steady state memakan waktu yang lebih lama dibandingkan dengan proses setup, karena proses transfer data terjadi melalui gelombang transmisi radio secara intensif. Sedangkan pada proses setup hanya terjadi proses penentuan cluster head dan pembentukan klaster. Pembagian fase terhadap waktu pada LEACH dapat direpresentasikan oleh gambar 6.5:

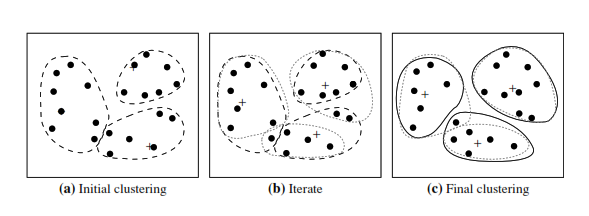


Gambar 2.5 Fase Pada LEACH

(Sumber: Ridwan dkk, 2020)

1. K-Means Clustering

Algoritma K-means Clustering adalah suatu metode penganalisaan data atau metode data mining yang melakukan proses pemodelan tanpa supervisi (unsupervised) dan merupakan salah satu metode yang melakukan pengelompokan data dengan sistem partisi (Jiawei dkk, 2011). Terdapat dua jenis data clustering yang sering dipergunakan dalam proses pengelompokan data yaitu hierarchical dan non-hierarchical, dan k-means merupakan salah satu metode data clustering non-hierarchical atau partitional clustering.



Gambar 2.6 Algoritma K-Means

(Sumber: Jiawei dkk, 2011. Data Mining Concept and Techniques. USA:Elsevier Inc)

Metode k-means berusaha mengelompokkan data yang ada ke dalam beberapa kelompok, dimana data dalam satu kelompok mempunyai karakteristik yang sama satu sama lainnya dan mempunyai karakteristik yang berbeda dengan data yang ada di dalam kelompok yang lain. Dengan kata lain, metode k-means bertujuan untuk meminimalisasikan objective function yang di set dalam proses clustering dengan cara meminimalkan variasi antar data yang ada di dalam suatu klaster dan memaksimalkan variasi dengan data yang ada di klaster lainnya.

Pada kasus ini, data berupa node yang dimana masing-masing node direpresentasikan sebagai titik kordinat yang berada pada posisi (x,y) sehingga untuk menghitung jarak pada masing-masing node harus mengurangi selisih antara nilai jarak x dan jarak y sehingga memodifikasi rumus Euclidean Distance menjadi seperti berikut (Fakhri M. Falahi, 2019):

|  |  |
| --- | --- |
|  | (11) |

Dimana:

J **=** jarak antara titik data node dengan *centroid*

= node ke-j pada klaster ke-i

= centroid ke-i pada klaster ke-i

Berikut adalah Langkah-langkah algoritma *k-means* (M. Ni’man Nasir & Irwan Budiman, 2017):

1. Tentukan jumlah klaster
2. Tentukan pusat awal K untuk setiap klaster secara *random*
3. Hitung jarak antara setiap titik data dari pusat mencakup mereka di titik pusat terdekat.
4. Ketika semua poin telah ditetapkan, hitung ulang posisi dari pusat K.
5. Ulangi Langkah C dan D hingga konvergensi. Ini menghasilkan pemisahan titik menjadi kelompok-kelompok yang matriksnya akan dibatasi dapat ditentukan.
6. Memperbarui pusat dengan rumus berikut

|  |  |
| --- | --- |
|  | (12) |

1. Konvergensi diperoleh ketika tidak ada variasi lagi di pusat *cluster*

Dimana dalam penelitian ini akan menggunakan data sebaran node sesuai dengan Table 4 Parameter Simulasi dimana node terdiri dari 100 node yang disebar pada titik koordinat 100 m^2 x 100 m^2 dengan jumlah klaster yang akan dibentuk yaitu 10 klaster. Dan berikut merupakan contoh proses perhitungan menggunakan persamaan (4) dan (5) untuk menghitung jarak antara node dan centroid serta bagaimana rumus perhitungan untuk memperbarui pusan klaster/centroid:

Table 3 Contoh Data Posisi Node

|  |  |
| --- | --- |
| Node ke (i) | Posisi node (x,y) meter |
| 1 | (4,3) |
| 2 | (5,4) |
| 3 | (6,7) |
| 4 | (7,8) |
| 5 | (8,6) |

Selanjutnya dari data diatas dipilih node ke 2 sebagai c1 dan 5 sebagai c2 sebagai titik centroid secara acak sesuai algoritma k-means, kemudian hitung jarak antara masing” node dengan *centroid* dengan rumus persamaan *(4)* seperti berikut:

m

m

m

605 m

m

m

m

m

m

Table 4 Perhitungan Centroid Terdekat Antar Node

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Node ke (i) | Jarak ke c1 | Jarak ke c2 | Jarak terdekat | Klaster |
| 1 |  | 5 |  | c1 |
| 2 | 0 | 605 | 0 | c1 |
| 3 |  |  |  | c2 |
| 4 |  |  |  | c2 |
| 5 |  | 0 | 0 | c2 |

Setelah node dikelompokkan dengan centroid terdekat, selanjutnya kita perbaharui pusat masing-masing klaster dengan persamaan *(5)* kemudian hitung ulang jarak setiap node dengan titik centroid baru dengan persamaan *(4)* untuk mengelumpokan node dengan klaster terdekat.

Table 5 Perhitungan Centroid Terdekat Antar Node Final

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Node ke (i) | Jarak ke c1 | Jarak ke c2 | Jarak terdekat | Klaster |
| 1 | 0.707 | 5 | 0.707 | c1 |
| 2 | 0.707 | 3.605 | 0.707 | c1 |
| 3 | 3.807 | 1 | 1 | c2 |
| 4 | 5.147 | 1 | 1 | c2 |
| 5 | 4.301 | 1.414 | 1.414 | c2 |

Karena posisi node pada klaster tidak mengalami perubahan maka iterasi dihentikan dan data node diatas merupakan hasil pengelompokan node dengan titik *centroid* atau klaster terdekat pada WSN.

1. K-Means++ Clustering

Algoritma *K-Means++ Clustering* adalah salah satu algoritma yang digunakan untuk klasifikasi atau pengelompokan data. Algoritma *k-means++* merupakan pengembangan dari algoritma *k-means* yang dikembangkan oleh Arthur dan Vassilvitskii pada tahun 2007 (Arthur & Vassilvitskii, 2007). Pada *k-means++* nilai *centroid* awal tidak dilakukan secara acak seperti pada algoritma *k-means* tetapi melalui tahap perhitungan. Setelah nilai *centroid* awal ditentukan maka tahapan selanjutnya sama persis seperti algoritma *k-means*.

Algoritma *k-means* dimulai dengan pemilihan K yang dipilih secara acak, K merupakan jumlah klaster yang ingin dibuat yang nilainya ditentukan secara acak dan nilai tersebut dijadikan sebagai pusat dari klaster atau *centroid*. Selanjutnya rumus *Euclidean Distance* untuk menghitung jarak dari setiap data terhadap masing-masing *centroid* sampai setiap data tersebut ditemukan jarak yang paling dekat dengan *centroid*. Setiap data diklasifikasikan berdasarkan jarak kedekatan dengan *centroid*. Langkah-langkah tersebut dilakukan berulang sampai diperoleh kestabilan dari nilai *centroid* (Falahi, 2019).

Sedangkan *k-means++* yaitu pengembangan dari algoritma *k-means*. Perbedaan dari algoritma k*-means* yaitu pada pemilihan nilai awal. *K-Means++* digunakan untuk meminimalisir dampak buruk dari algoritma *k-means* yang bergantung dari nilai awal. Rumus menentukan nilai awal pada *k-means++:*

|  |  |
| --- | --- |
|  | (6) |

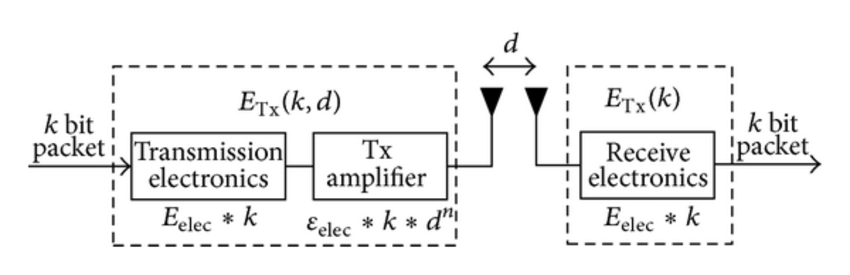
= Jarak *euclidean distance*

= Jumlah jarak *euclidean distance*

Adapun langkah-langkah pada algoritma *k-means++*adalah sebagai berikut (Fox & Emily, 2016):

1. Tentukan titik centroid pertama secara acak dari pada titik data node.
2. Untuk setiap node yang diamati, hitung jarak D(x) dengan centroid terdekat dengan rumus persamaan (4).
3. Pilih centroid baru dari antara titik node. dengan probabilitas x dipilih proporsional dengan rumus persamaan (6), dimana posisi node dengan nilai K terbesar dipilih sebagai centroid.
4. Ulangi langkah B dan C hingga jumlah centorid sama dengan jumlah klaster yang ditentukan.
5. Lanjut ke proses algoritma k-means biasa.
6. Energy Dissipation Model

Energy dissipation model adalah pemodelan bagaiamana pemancar menghilangkan energi untuk menjalankan elektronik radio dan power amplifier, dan penerima membuang energi untuk menjalankan elektronik radio. Dalam model disipasi energi radio yang efektif, baik ruang bebas (𝑑 kehilangan daya) dan model saluran multipath fading (𝑑 24 daya hilang) digunakan, tergantung pada jarak antara pemancar dan penerima. Jika jaraknya kurang dari ambang 𝑑 0, ruang model akan digunakan; jika tidak, model multipath digunakan. Oleh karena itu, konsumsi energi untuk mentransmisikan pesan bit lebih jauh dapat diformulasikan seperti gambar 2.7 (Qiang dkk, 2015).



Gambar 2.7 Energy Dissipation Model

(Saheb & Sharma, 2017)

1. MATLAB 2015 A

MATLAB merupakan singkatan dari (Matrix LABoratory) adalah sebuah lingkungan komputasi numerik dan bahasa pemrograman komputer generasi keempat. Dikembangkan oleh The MathWorks.Inc, MATLAB memungkinkan manipulasi matriks, pemplotan fungsi dan data, implementasi algoritma, pembuatan antarmuka pengguna, dan pengantarmukaan dengan program dalam bahasa lainnya. Meskipun hanya bernuansa numerik, sebuah kotak kakas (toolbox) yang menggunakan mesin simbolik MuPAD, memungkinkan akses terhadap kemampuan aljabar komputer. Sebuah paket tambahan, Simulink, menambahkan simulasi grafis multi ranah dan desain berdasar-model untuk sistem terlekat dan dinamik.



Gambar 2.8 Software MATLAB

(Sumber: ch.mathworks.com)

Matlab memiliki karakteristik yang berbeda dengan bahasa pemrograman lain yang sudah ada lebih dahulu seperti Delphi, Basic maupun C++. Matlab merupakan bahasa pemrograman level tinggi yang dikhususkan untuk kebutuhan komputasi teknis, visualisasi dan pemrograman seperti komputasi matematik, analisis data, pengembangan algoritma, simulasi dan pemodelan dan grafik-grafik perhitungan.

Matlab hadir dengan membawa warna yang berbeda. Hal ini karena matlab membawa keistimewaan dalam fungsi-fungsi matematika, fisika, statistik, dan visualisasi. Matlab dikembangkan oleh MathWorks, yang pada awalnya dibuat untuk memberikan kemudahan mengakses data matrik pada proyek LINPACK dan EISPACK. Saat ini matlab memiliki ratusan fungsi yang dapat digunakan sebagai problem solver mulai dari simpel sampai masalah-masalah yang kompleks dari berbagai disiplin ilmu.

**BAB III**

**ANALISIS DAN PERANCANGAN SISTEM**

* + 1. **Pengumpulan Data**

Data yang akan digunakan dalam penelitian ini diperoleh dari hasil perbandingan simulasi pada aplikasi MATLAB 2015a dengan beberapa parameter uji untuk mendapatkan data:

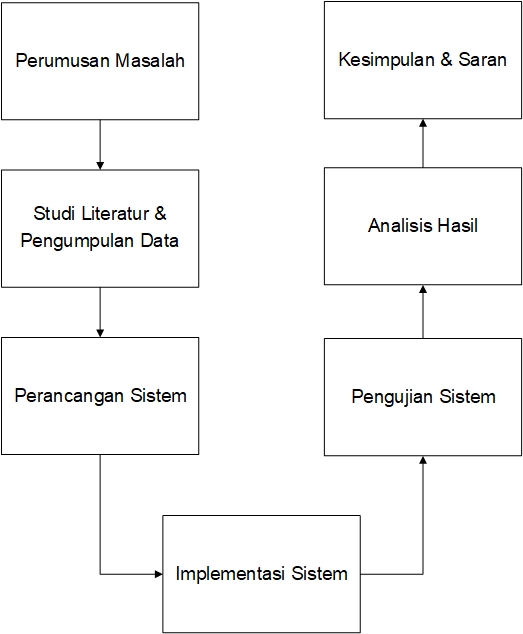
* 1. Total konsumsi energi (Joule)
  2. Jumlah node hidup selama 100 round
  3. Jumlah death node selama 100 round

Dimana data yang didapat menggunakan skenario uji berdasarkan parameter dari penelitian sebelumnya oleh (Saheb & Sharma, 2017). Dimana nantinya parameter simulasi tersebut akan digunakan untuk menjadi skenario simulasi untuk mendapatkan data primer pada penelitian ini. Dimana skenario yang dirancang akan menggunakan 100 node yang disebar pada titik kordinat x dan y dengan luas daerah 100 m^2 x 100 m^2 untuk membuat skema topologi jaringan WSN.

Kemudian setelah dilakukan proses simulasi, data yang didapatkan direpresentasikan dalam bentuk grafik pada software MATLAB yang menampilkan data total konsumsi energi (joule), jumlah node mati dan node hidup selama 100 round simulasi, setelah itu hasil dibandingkan dengan hasil penelitian sebelumnya untuk membuktikan apakah kinerja dari protokol KM-LEACH yang dioptimasi dengan algoritma k-means++ lebih baik daripada protokol LEACH yang dioptimasi dengan algoritma k-means saja. Selain itu dilakukan juga studi literatur untuk memberikan argumen yang logis, serta memperkuat hasil dari pengujian tersebut.

* + 1. **Desain Penelitian**

Tahapan penelitian pada perancangan simulasi ini melalui beberapa tahapan yang dilakukan seperti flowchart pada gambar 7.1:



Gambar **Error! No text of specified style in document.**.1.1 Desain Alur Sistem

(Sumber: Kadnyanan, 2018)

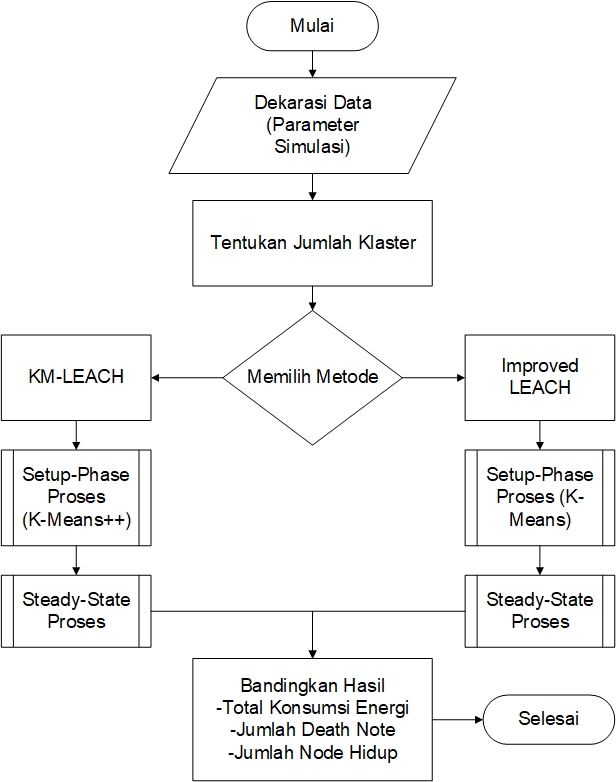
Sesuai gambar 1.1, Desain ini merupakan adaptasi dari sekema waterfall model yang diambil berdasarkan penelitian sebelumnya oleh (Kadnyanan, 2018). Dimana sesuai dengan desain metode pada gambar 1.1, pada tahap pertama yang akan dilakukan pada penelitian ini adalah perumusan masalah terkait permasalahan yang akan diangkat dalam penelitian ini. Setelah ditetapkan permasalahan yang akan diteliti, tahap kedua yang dilakukan adalah studi literatur dan pengumpulan data terkait permasalahan yang diangkat dan informasi-informasi atau kajian teori yang menunjang penelitian. Tahap ketiga dilakukan perancangan sistem untuk mengetahui alur dan segala keperluan sistem yang akan dibuat.

Barulah pada tahap keempat dilakukan implementasi sistem yang mana implementasi sistem akan dilakukan dalam bentuk simulasi dan kemudian dilakukan pengujian serta analisis sistem untuk mengetahui hasil akhir penelitian yang dilakukan, untuk mengetahui apakah penelitian yang dilakukan berhasil atau tidak. Dan di tahap akhir adalah kesimpulan & saran yang berisi hasil dari penelitian yang dilakukan dan saran atau arahan untuk pengembangan selanjutnya yang dapat dilakukan terkait penelitian yang diteliti.

Sistem ini dapat diimplementasikan secara nyata pada masing-masing node dan base station pada WSN. Dimana masing-masing node dapat terdiri dari perangkat Arduion uno, Xbee sensor ataupun jenis node sensor lainnya, dimana nantinya sistem optimasi protokol LEACH dengan k-means++ tersebut ditanamkan atau di program pada perangkat tersebut dalam jaringan WSN dengan menggunakan bahasa pemrograman C/C++.

* + 1. **Metode Yang Digunakan**

Pada penelitian ini akan menggunakan 2 metode yaitu algoritma LEACH dengan K-Means dan K-Means++. Setelah itu akan dibandingkan kinerja antara protokol LEACH dengan K-Means dan protokol LEACH dengan K-Means++ untuk mendapatkan hasil penelitian. Berikut adalah gambaran program yang akan dibuat secara umum yang direpresentasikan pada gambar 2.2:

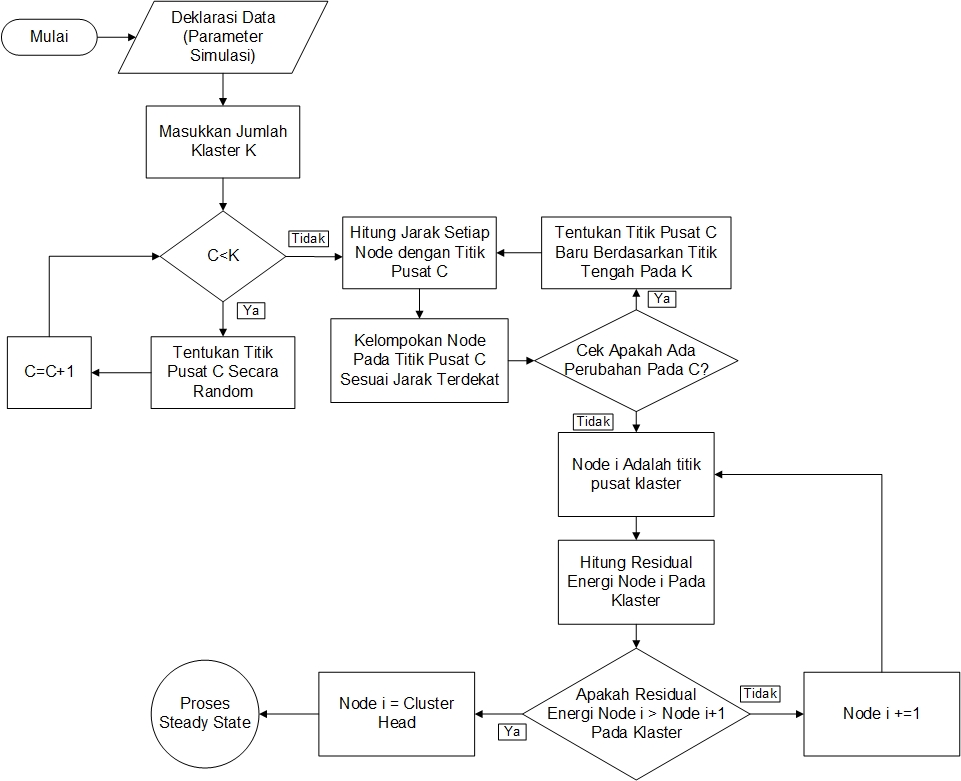


Gambar 1.**Error! No text of specified style in document.**.2 Flowchart Rancangan Program

* + 1. **LEACH Dengan K-Means**

Secara umum algoritma LEACH memiliki 2 proses utama yaitu Setup Phase dan Steady State. Pada proses Setup Phase inilah akan dimodifikasi dengan menggunakan algortima k-means. Algoritma k-means berperan dalam proses pembentukan klaster dimana dengan bantuan algoritma k-means dapat membantuk membentuk klaster yang lebih baik dibandingkan algoritma clustering biasa pada algoritma LEACH. Berikut adalah flowchart protokol LEACH yang sudah dioptimasi dengan K-Means pada gambar 1.3:

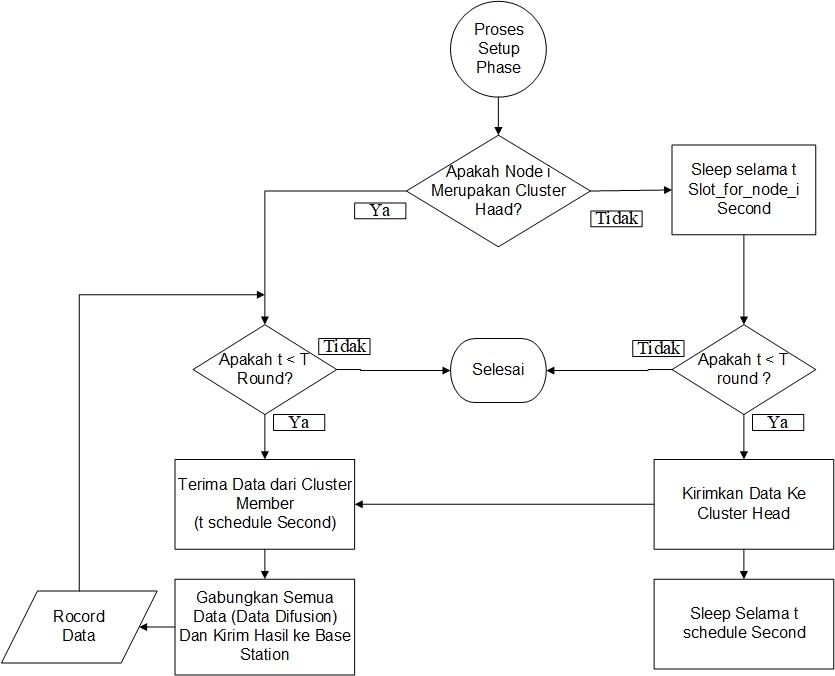
1. *Setup Phase*



Gambar 1.**Error! No text of specified style in document.**.3 Flowchart Setup Phase LEACH Dengan K-Means

Penjelasan *flowchart*:

1. Tahap pertama adalah membentuk topologi jaringan berdasarkan beberapa parameter simulasi yang sudah ditentukan
2. Kedua masukkan jumlah klaster
3. Kemudian lakukan iterasi untuk penentuan *centroid*/titik pusat klaster sebanyak C=Jumlah klaster yang ditentukan
4. Letakkan titik pusat klaster/*centroid* C secara *random*
5. Hitung jarak setiap node dengan *centroid* C dengan persamaan (4)
6. Kelompokan *node* pada titik pusat klaster/*centroid* C sesuai jarak terdekat
7. Kemudian cek apakah ada perubahan pusat klaster/*centroid* C
8. Jika iya maka tentukan titik pusat klaster/*centroid* C baru berdasarkan titik tengah pada k dengan persamaan (5), kemudian ulangi langkah 5-7 hingga tidak ada perubahan pusat klaster/*centroid* C
9. Jika tidak maka node i = titik pusat klaster/*centroid* C
10. Kemudian masuk pada penentuan *cluster* head berdasarkan persamaan (1) dimana dalam pemilihan memerhatikan parameter residual energi pada *node* tiap klaster yang dihitung dengan persamaan (2)
11. Bandingkan apakah residual energi *nod*e i > *node* i+1 dengan menggunakan persamaan (3)
12. Jika tidak maka *node* i = *node* i+1 dan ulangi langkah 9 hingga 11
13. Jika iya maka *node* i adalah *cluster head* dan lanjut ke proses *steady state*
14. Steady State

****

Gambar 1.**Error! No text of specified style in document.**.4 Flowchart Steady State LEACH

(Sumber: Sohraby dkk, 2011. Wireless Sensor Network Technology, Protocols, Applications Book. New Jersey:John Wiley & Sons, Inc)

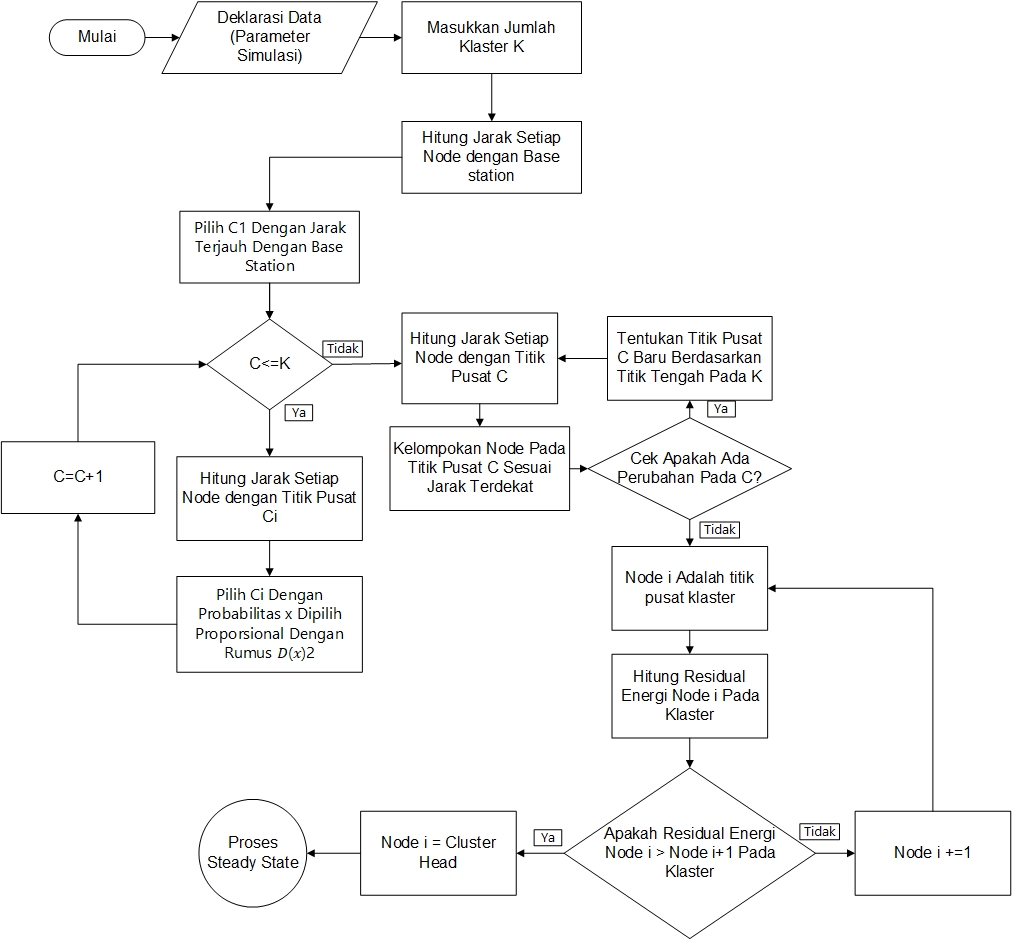
Penjelasan *Flowchart*:

1. Cek apakah *node* merupakan sebuah *cluster head*
2. Apabila merupakan *cluster head* maka, *cluster head* akan menerima data dari semua anggota klaster.
3. *Node* yang bukan *cluster head* hanya akan mengirim data sesuai dengan jadwal TDMA yang sudah diterima.
4. *Node* *cluster head* akan mengumpulkan dan melakukan kompresi terhadap data
5. Kirim semua data yang sudah dikompresi ke *Base Station*
6. Proses *steady-state* selesai.
   * 1. **LEACH Dengan K-Means++**

Algoritma *K-Means++ Clustering* adalah salah satu algoritma yang digunakan untuk klasifikasi atau pengelompokan data, algoritma ini merupakan pengembangan dari algoritma *K-Means*. Pada protokol LEACH dengan bantuan algoritma *K-Means++* akan memodifikasi *setup phase* dari protokol LEACH dimana dalam *setup phase* pemilihan *cluster head* menggunakan algoritma *KMeans*++.

Namun dalam penelitian ini dalam pemilihan Centroid awal tidak dilakukan secara acak seperti algoritma *K-Means++,* namun pemilihan dilakukan berdasarkan 1 parameter tambahan yaitu jarak terjauh antara posisi node dengan base station. Untuk mencegah penentuan centroid awal yang dilakukan secara acak, hal ini dilakukan sesuai prinsip kerja algoritma *k-means++* dalam penentuan centroid yaitu berdasarkan jarak node terjauh yang dimana jarak antara node dengan basestation dihitung dengan rumus *ecludien distance* pada persamaan (4). Untuk penjelasan lebih lanjut mengenai langkah-langkah yang dilakukan dapat dilihat pada *flowchart setup phase* dengan algoritma *K-Means++* yang direpresentasikan pada gambar 1.5:

1. Setup Phase



Gambar 1.**Error! No text of specified style in document.**.5 Flowchart Setup Phase KM-LEACH

Penjelasan Flowchart:

1. Tahap pertama buat bentuk *node* sesuai dengan parameter yang ditentukan
2. Input jumlah klaster
3. Hitung jarak semua node dengan base station dengan rumus dari persamaan (4)
4. Tentukan pusat klaster C1 berdasarkan jarak terjauh antara node dengan base station.
5. Pemilihan Ci selanjutnya hitung jarak semua node dengan pusat klaster C1 dengan rumus dari persamaan (4)
6. Tentukan Ci dengan probabilitas x dipilih proporsional dengan persamaan **.**
7. Ulangi langkah 4 dan 5 hingga semua klaster terbentuk
8. Hitung jarak setiap node dengan pusat klaster/*centroid* C dengan persamaan (4)
9. Kelompokan *node* pada titik pusat klaster/*centroid* C sesuai jarak terdekat
10. Kemudian cek apakah ada perubahan pusat klaster/*centroid* C
11. Jika iya maka tentukan titik pusat klaster/*centroid* C baru berdasarkan titik tengah pada k dengan persamaan (5), kemudian ulangi langkah 7-9 hingga tidak ada perubahan pusat klaster/*centroid* C
12. Jika tidak maka node i = titik pusat klaster/*centroid* C
13. Kemudian masuk pada penentuan *cluster* head berdasarkan persamaan (1) dimana dalam pemilihan memerhatikan parameter residual energi pada *node* tiap klaster yang dihitung dengan persamaan (2)
14. Bandingkan apakah residual energi *nod*e i > *node* i+1 dengan menggunakan persamaan (3)
15. Jika tidak maka *node* i = *node* i+1 dan ulangi langkah 11 hingga 13
16. Jika iya maka *node* i adalah *cluster head* dan lanjut ke proses *steady state*
    * 1. **Implementasi**

Untuk mengimplementasikan penelitian ini, terdapat komponen-komponen pendukung yaitu:

1. Kebutuhan Perangkat Keras (Hardware)

Dari segi kebutuhan perangkat keras dibutuhkan perangkat elektronik dengan spesifikasi yaitu:

* Processor intel(R) Core(TM) i7-7700HQ CPU @2.80GHz
* Memory 8192MB RAM
* VGA Card Nvidia Geforce 950M
* Harddisk 1000 GB HDD

1. Kebutuhan Perangkat Lunak (Software)

Dari segi kebutuhan perangkat lunak (software), sistem ini akan diimplementasikan pada sistem operasi Windows 10. Aplikasi yang digunakan dalam penelitian ini adalah:

* MATLAB 2015 A

Serta aplikasi tambahan yang dibutuhkan seperti berikut:

* Google Chrome
* Nitro Pro
  + 1. **Skenario Pengujian**

Pada penelitian ini Protokol LEACH dengan optimasi k-means dan KM-LEACH dengan optimasi k-means++ disimulasikan dengan software MATLAB 2015a dengan beberapa parameter berdasarkan hasil penelitian (Saheb & Sharma, 2017) untuk membuat skenario simulasi yang telah ditetapkan, diantaranya:

Table 1 Parameter Simulasi

(Sumber: Saheb & Sharma, 2017)

|  |  |
| --- | --- |
| Parameter | Nilai |
| Jumlah *node* | 100 |
| Luas area simulasi | 100 x 100 |
| Jumlah *round* (putaran) | 100 |
| Posisi *base station* | (50,50) |
| Probabilitas *cluster head %* | 10 |
| Energi awal *node* | 0.5 Joule |
| Data Length | 4000 bits |
| Eelec | 50nJ/bit |
| mp | 0.0013pJ/bit/ |
|  | 10pJ/bit/ |
| Data Aggregation (EDA) | 5nJ/bit/signal |

Dengan parameter diatas, skenario akan terus dijalankan hingga round 100. Simulasi akan menghasilkan data berupa grafik untuk mengukur kinerja dari protokol LEACH dengan metode k-means dan KM-LEACH dengan metode k-means++ dan pada masing-masing skenario, yang dimana nilai hasil simulasi yang dibahas lebih lanjut pada sub bab pengukuran dan hasil simulasi.

* + 1. **Pengukuran dan Hasil Simulasi**

Kualitas kinerja dari protokol KM-LEACH dengan protokol LEACH akan diukur dengan beberapa parameter seperti rata-rata konsumsi energi, banyaknya *death node* dan jumlah *node* yang hidup selama 100 round berlangsung. Dimana untuk menghitung total konsumsi energi yang dikeluarkan per-*bit* didapat dengan menggunakan konsumsi energi komunikasi radio yang sama yang diperlihatkan sebagai digunakan sebagai protokol LEACH. Model ini terdiri dari dua bagian: model konsumsi energi transmisi dan model konsumsi energi penerima. Sesuai dengan model disipasi energi komunikasi radio, didapatkan rumus untuk menghitung konsumsi energi dengan cara (Saheb & Sharma, 2017:

|  |  |
| --- | --- |
|  | (1) |
| jika | (2) |
| jika | (3) |

Dimana penjelas dari variabel rumus diatas yaitu:

= *Threshold Distance,*

**=** *Electronic Energy*

**=** *Parameter for Free Space Model*

**=** *Parameter for Multi Path Model*

Dimana untuk mendapatkan didapatkan dengan rumus:

|  |  |
| --- | --- |
|  | (4) |

Konsumsi energi untuk menerima pesan *k-bit* dapat dihitung dengan persamaan:

|  |  |
| --- | --- |
|  | (5) |

Sedangkan untuk mengetahui jumlah node mati dan node hidup dihitung dengan:

|  |  |
| --- | --- |
| ***Death node* = total *node - node* hidup** | (6) |
| ***Node* hidup = total *node – node* mati** | (7) |

Dimana setelah data didapatkan dari 2 skenario pada LEACH protokol dengan metode *k-means* dan KM-LEACH protokol dengan metode *k-means++*, kemudia data dibantingkan dengan tabel berikut untuk mendapatkan hasil akhir untuk mengetahui protokol yang lebih optimal diantara kedua protokol tersebut. Berikut adalah tabel parameter pengukuran kedua protokol:

Table 2 Parameter Uji Protokol

|  |  |  |  |
| --- | --- | --- | --- |
| Protokol | Jumlah *Death Node*/100 *Round* | Jumlah *Node* Hidup /100 *Round* | Total Konsumsi Energi/100 *Round* |
| LEACH dengan metode *k-means* |  |  |  |
| KM-LEACH dengan metode *k-means++* |  |  |  |

**BAB IV**

**HASIL DAN PEMBAHASAN**

**BAB V**

**PENUTUP**

**DAFTAR PUSTAKA**