

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

2.1.1 *Low Energy Adaptive Clustering Hierarchy (LEACH)*

Low Energy Adaptive Clustering Hierarchy (LEACH) merupakan salah satu protokol routing hierarkis paling awal dan banyak digunakan dalam jaringan sensor nirkabel (*Wireless Sensor Networks* - WSN) yang dirancang untuk meminimalkan konsumsi energi dan memperpanjang masa hidup jaringan. Protokol ini bekerja dengan membagi *node* sensor ke dalam beberapa klaster, di mana setiap klaster dipimpin oleh satu *cluster head* (CH) yang bertugas mengumpulkan, mengagregasi, dan meneruskan data dari *node* anggota ke *base station*. *LEACH* menggunakan mekanisme pemilihan CH secara periodik dan adaptif berdasarkan probabilitas tertentu, sehingga distribusi beban komunikasi menjadi lebih merata. Keunggulan utama dari *LEACH* terletak pada kemampuannya mengurangi jumlah transmisi langsung ke *base station*, yang secara signifikan menghemat energi pada skala jaringan besar. Oleh karena itu, *LEACH* sering dijadikan dasar dalam pengembangan berbagai protokol routing WSN yang lebih canggih dan kontekstual (Heinzelman, dkk, 2000).

A Comprehensive Survey on LEACH-Based Clustering Routing Protocols in Wireless Sensor Networks menyajikan tinjauan mendalam terhadap berbagai varian dan pengembangan dari protokol *LEACH* yang telah diusulkan dalam dua dekade terakhir. Studi ini mengklasifikasikan protokol-protokol tersebut berdasarkan karakteristik utama seperti pemilihan *cluster head*, skema komunikasi (single-hop dan multi-hop), serta strategi efisiensi energi yang digunakan. Selain itu, penelitian ini juga mengevaluasi kekuatan dan kelemahan dari masing-masing pendekatan dalam konteks efisiensi energi, skalabilitas, dan ketahanan jaringan terhadap dinamika topologi. Hasil dari survei ini menunjukkan bahwa meskipun *LEACH* telah menjadi dasar yang kuat dalam protokol routing WSN, masih banyak peluang pengembangan untuk menyesuaikannya dengan kebutuhan aplikasi WSN modern yang lebih kompleks dan dinamis (Daanoune, Abdennaceur, dkk, 2021).

Optimization of LEACH Protocol in Wireless Sensor Network Using Machine Learning penggabungan algoritma k-means yang dimodifikasi dengan protokol *LEACH* untuk meningkatkan efisiensi energi dalam jaringan sensor nirkabel (WSN). Dalam pendekatan ini, pemilihan *cluster head* tidak lagi dilakukan secara acak, melainkan berdasarkan kombinasi antara jarak *Euclidean* dan tingkat energi residu dari setiap *node*. Dengan demikian, hanya *node* yang memiliki energi cukup dan berada dalam posisi strategis yang dipilih sebagai *cluster head*, sehingga distribusi beban komunikasi menjadi lebih seimbang dan efisien. Hasil simulasi menunjukkan bahwa metode ini mampu meningkatkan efisiensi energi hingga 48,85% dibandingkan dengan protokol *LEACH* konvensional, serta memperpanjang masa hidup jaringan secara keseluruhan (El Khediri, dkk, 2024).

Modified Weighted Low Energy Adaptive Clustering Hierarchy (MW-LEACH) sebagai pengembangan dari protokol *W-LEACH* untuk meningkatkan *Quality of Service* (QoS) dalam jaringan sensor nirkabel (WSN) dengan *node* sink yang bergerak. Protokol ini mengintegrasikan mekanisme pemilihan *cluster head* berdasarkan sisa energi dan jarak *node*, serta menerapkan dua tingkat daya transmisi untuk komunikasi intra-klaster dan antar-klaster. Hasil simulasi menunjukkan bahwa *MW-LEACH* mampu meningkatkan throughput sebesar 63% dibandingkan dengan *W-LEACH*, serta mengurangi konsumsi energi dan rasio kehilangan paket secara signifikan (Coker, dkk, 2018).

A Comprehensive Review on Successors of LEACH Protocols in Wireless Sensor Networks mengkaji secara menyeluruh perkembangan dan penerus dari protokol *LEACH* yang telah banyak digunakan dalam jaringan sensor nirkabel (WSN). Kajian ini mengklasifikasikan berbagai varian *LEACH* berdasarkan pendekatan komunikasi yang digunakan, yaitu komunikasi single-hop maupun multi-hop, serta mengevaluasi performa masing-masing varian dari aspek efisiensi energi, skalabilitas, latency, dan umur jaringan. Studi ini memberikan wawasan mendalam mengenai bagaimana setiap varian berusaha mengatasi keterbatasan *LEACH* konvensional, terutama dalam hal distribusi beban komunikasi dan adaptasi terhadap topologi jaringan yang dinamis. Temuan dalam penelitian ini menunjukkan bahwa penerus *LEACH* mampu meningkatkan kualitas layanan (QoS) dan memperpanjang lifetime jaringan secara signifikan, sehingga menjadi

dasar penting dalam pengembangan protokol routing yang lebih adaptif dan hemat energi di masa mendatang (Al-Sodairi dan Ouni, 2018).

Penelitian mengenai *AE-LEACH* (*Advanced Efficient Low-Energy Adaptive Clustering Hierarchy*) merupakan pengembangan dari protokol *LEACH* yang bertujuan untuk meningkatkan efisiensi energi dalam jaringan sensor nirkabel (*WSN*). *AE-LEACH* mengintegrasikan pendekatan klasterisasi adaptif dengan pemilihan *cluster head* berdasarkan kriteria seperti energi residu dan jarak ke *base station*, sehingga distribusi beban komunikasi menjadi lebih seimbang dan efisien. Pendekatan ini dirancang untuk mengurangi konsumsi energi dan memperpanjang umur jaringan secara keseluruhan (Babu M, dkk, 2022).

NN_ILEACH, yaitu varian dari protokol *LEACH* yang dioptimalkan menggunakan pendekatan *neural network* untuk meningkatkan efisiensi energi dalam jaringan sensor nirkabel (*WSN*). Protokol ini mengintegrasikan *Energy Hole Removing Mechanism (EHORM)* untuk menghindari terjadinya “zona mati” di sekitar *base station*, yang sering terjadi pada *LEACH* konvensional karena konsumsi energi tidak merata. Dengan melibatkan *neural network* dalam proses pemilihan *cluster head*, *NN_ILEACH* secara adaptif dapat memilih *node* dengan kombinasi optimal antara posisi geografis dan tingkat energi tersisa. Hasil simulasi menunjukkan bahwa protokol ini secara signifikan memperpanjang masa hidup jaringan dan meningkatkan efisiensi transmisi data (El-Sayed, dkk, 2024).

MFG-LEACH (*Multi-Feature Game-Theoretic LEACH*) sebagai pengembangan lanjut dari protokol *LEACH* untuk meningkatkan efisiensi energi dalam jaringan sensor nirkabel (*WSN*). Protokol ini menggunakan pendekatan teori permainan multi-fitur untuk proses pemilihan *cluster head*, yang mempertimbangkan beberapa parameter penting seperti sisa energi *node*, jarak ke *base station*, dan kepadatan *node* dalam area tertentu. Dengan memodelkan interaksi antar *node* sebagai permainan strategis, *MFG-LEACH* mampu menentukan *cluster head* secara lebih adil dan efisien, sehingga beban komunikasi dapat didistribusikan secara merata. Hasil evaluasi menunjukkan bahwa protokol ini secara signifikan meningkatkan masa pakai jaringan dan stabilitas topologi dibandingkan *LEACH* klasik (Lyashenko, dkk, 2023).

DE-LEACH (Distance and Energy Aware LEACH) sebagai solusi untuk mengatasi kelemahan dalam mekanisme pemilihan *cluster head* pada protokol *LEACH* konvensional. Dalam pendekatan ini, pemilihan *cluster head* tidak hanya mempertimbangkan probabilitas acak, tetapi juga memperhitungkan jarak *node* ke base station dan energi residu pada setiap *node*. Tujuannya adalah untuk mencegah pemilihan *cluster head* dari *node* yang terletak jauh atau memiliki energi rendah, sehingga distribusi energi menjadi lebih merata dan umur jaringan lebih panjang. *DE-LEACH* terbukti meningkatkan efisiensi energi, memperkecil overhead komunikasi, dan memperpanjang *network lifetime* secara signifikan (Al-Sodairi dan Ouni, 2018).

Pada penelitian ini, protokol *Low Energy Adaptive Clustering Hierarchy (LEACH)* dikembangkan lebih lanjut dengan menyisipkan algoritma *clustering Hierarchical Agglomerative Clustering (HAC)* guna meningkatkan efisiensi dalam pembentukan kelompok *node* sensor yang lebih berdekatan secara spasial. Proses dimulai dengan penentuan kriteria data berdasarkan suhu dan kelembapan, di mana suhu diklasifikasikan ke dalam tujuh kategori, yaitu: sangat dingin, dingin, cukup dingin, sejuk, cukup panas, panas, dan sangat panas; sedangkan kelembapan dikategorikan menjadi: kering, cukup kering, sedang, lembap, dan basah. Berdasarkan klasifikasi tersebut, dilakukan proses klusterisasi *node* untuk mendukung pengambilan keputusan yang adaptif dalam sistem irigasi cerdas.

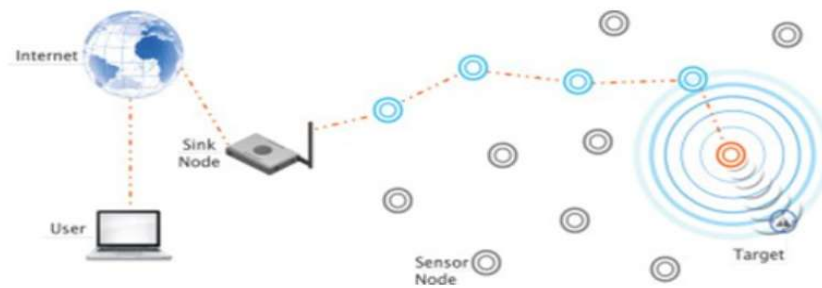
Cluster head (CH) ditentukan sebagai titik referensi utama dalam komunikasi data antar *node*. Untuk meningkatkan efisiensi energi dan mengurangi jarak komunikasi, penentuan *node* anggota dilakukan menggunakan pendekatan jarak *Euclidean*, yang kemudian dilanjutkan dengan proses pengelompokan menggunakan metode single linkage dari *HAC*. Integrasi *HAC* dalam pembentukan *cluster* memungkinkan sistem untuk mengidentifikasi *node-node* yang memiliki kedekatan spasial optimal terhadap CH, sehingga membentuk klaster yang lebih kompak dan hemat energi. Pengembangan ini diharapkan dapat meningkatkan stabilitas topologi jaringan serta memperpanjang masa pakai sistem secara keseluruhan.

Algoritma *LEACH-HAC* membentuk empat *channel* berupa saluran protokol pada *WSN* dengan proses pengelompokan *node* terdekat atau yang paling mirip. Empat *channel* pada *WSN* membagi *node* melalui proses *clustering* menggunakan algoritma *HAC*. Desain yang dilakukan mencakup pembentukan *cluster head* dan *node*, di mana *node* mengirimkan data kepada *cluster head*, dan *cluster head* selanjutnya akan mengirimkan data ke *base station* atau *server*. Dengan membagi menjadi empat *channel* atau saluran, proses ini akan mengurangi kepadatan lalu lintas protokol pada *WSN*, membagi saluran data, serta mengoptimalkan proses pengiriman data dan meningkatkan performa jaringan hasil penelitian menunjukkan bahwa *LEACH-HAC* mengembangkan model protokol dengan teknik *clustering* yang membentuk kedekatan antar *node*.

2.1.2 Wireless Sensor Network

Jaringan sensor nirkabel *WSN* terdiri atas banyak sensor yang didistribusikan secara spasial (Ali dan Kumar, 2022). Sensor-sensor tersebut digunakan untuk memantau berbagai kondisi lingkungan, seperti temperatur, kelembaban, cuaca, dan sebagainya (Thakur, dkk., 2020). Data hasil pengamatan ditransformasikan menjadi sinyal elektronik (Derdar, dkk., 2022). Sebuah sensor dilengkapi dengan *radio transceiver*, *microcontroller*, dan sumber energi. *Multi-channel* merupakan konsep dalam komunikasi atau sistem jaringan yang menggunakan lebih dari satu saluran transmisi untuk mengirim dan menerima data secara simultan, dengan tujuan meningkatkan kapasitas, efisiensi, dan keandalan komunikasi (Bakr dan Lilien, 2014).

Pada umumnya, sensor yang digunakan berukuran kecil dan tidak mahal (Almasri, dkk., 2022). Performa sensor kecil biasanya tidak sebaik sensor besar yang mahal. Namun, dengan ukuran yang kecil dan harga yang rendah, luas area yang diamati menjadi semakin besar karena jumlah sensor yang dapat dipasang lebih banyak (Kwong, dkk., 2007). Sensor pada jaringan *WSN* mengirim data ke *base station* atau dapat juga saling berkomunikasi satu sama lain. *Node* dalam *WSN* memiliki energi yang terbatas dan sering kali tidak memungkinkan untuk mengisi ulang baterai (Harb dan Desuky, 2011). *Wireless Sensor Network* ditunjukkan pada Gambar 2.1.



Gambar 2. 1 *Wireless Sensor Network*

Wireless Sensor Network WSN, di mana *node* sensor tersebar di suatu area. *Node* sensor tersebut memiliki kemampuan untuk meneruskan data yang dikumpulkan ke *node* lain yang berdekatan. Data dikirim melalui transmisi radio dan diteruskan menuju *base station* sebagai penghubung antara *node* dan pengguna (*user*). *Transceiver* berfungsi untuk menerima dan mengirim data dengan menggunakan protokol pada perangkat lain. *Mikrokontroler* berfungsi untuk melakukan perhitungan, mengontrol, dan memproses perangkat yang terhubung dengannya. *Power source* berfungsi sebagai sumber energi bagi sistem *WSN* secara keseluruhan. *Sensor* merupakan alat yang mampu mengubah suatu bentuk energi ke bentuk energi lain. Dalam hal ini, *sensor* mengubah besaran energi yang diukur menjadi energi listrik, yang kemudian dikonversi oleh *ADC (Analog to Digital Converter)* menjadi deretan pulsa terkuantisasi yang dapat dibaca oleh *mikrokontroler*.

Dalam ilustrasi, *node* berwarna merah berfungsi sebagai sensor yang memindai fenomena di sekitar target, seperti suhu dan kelembaban. *Node* berwarna abu-abu merepresentasikan bagian dari jaringan, sementara *node* berwarna biru menunjukkan jalur protokol yang mentransmisikan data menuju *base station* atau *server*. Sebagaimana umumnya pada jaringan komputer, *WSN* juga dapat diimplementasikan dalam beberapa topologi, yaitu:

1. *Point-to-Point Topology*
2. *Star Topology*
3. *Mesh Topology*
4. *Hybrid Topology*
5. *Tree Topology*

Jaringan *point-to-point topology* merupakan salah satu bentuk topologi jaringan yang paling sederhana. Dalam konfigurasi ini, dua perangkat ujung terhubung secara langsung tanpa perantara (Sivakumar dan Radhika, 2018). Topologi *point-to-point* ini secara fisik menghubungkan kedua *node* secara langsung melalui kabel atau media lainnya (Santhameena dan J., 2022). Topologi *point-to-point* digunakan untuk menentukan struktur jaringan telekomunikasi yang berbeda (Kavitha dan Ganapathy, 2022). Topologi logis *point-to-point* menggambarkan bagaimana data sebenarnya mengalir dalam jaringan (Cheng, dkk., 2020). Sedangkan topologi fisik *point-to-point* menggambarkan bagaimana *node* ditempatkan dalam jaringan (Alkalbani, dkk., 2016).

Topologi *star* merujuk pada metode atau mekanisme untuk menghubungkan dua atau lebih komputer dalam jaringan dengan pola berbentuk bintang. Konfigurasi ini mengkonsolidasikan koneksi dari *node* pusat ke setiap *node* atau pengguna, sehingga seluruh *node* terhubung langsung dengan *node* pusat tersebut (Zhang dan Mao, 2021). Topologi *star* memiliki prinsip kerja dengan kontrol atau kendali terpusat. Seluruh *link* melalui pusat, kemudian data disalurkan ke semua *node* atau hanya ke *node* tertentu yang ditentukan oleh *server* pusat (Taneja, dkk., 2020). Dalam istilah teknologi informasi, simpul pusat disebut sebagai stasiun primer, sedangkan *node-node* yang terhubung lainnya disebut sebagai stasiun sekunder atau *client* (Alkalbani, dkk., 2016).

Topologi *mesh* merujuk pada konfigurasi jaringan di mana sekelompok perangkat komputer terhubung secara langsung, memungkinkan setiap perangkat berkomunikasi satu sama lain tanpa perantara (Zeineldin, dkk., 2013). Topologi ini juga disebut sebagai topologi jala karena bentuk rangkaiannya menyerupai jala (Meshram, dkk., 2020). Topologi *mesh* banyak digunakan untuk jaringan berskala menengah yang memerlukan komunikasi antarperangkat dengan cepat (Meshram, dkk., 2020). Namun, karena menggunakan banyak kabel dan cukup sulit dikelola, topologi ini memiliki tantangan tersendiri dalam implementasinya (Sumiharto, dkk., 2019).

Topologi *hybrid* dapat diartikan sebagai model topologi baru dalam sebuah jaringan komputer yang tercipta dari gabungan dua atau lebih topologi jaringan dengan jenis yang berbeda (Rethfeldt, dkk., 2021). Karena merupakan gabungan

dari berbagai jenis topologi, desain topologi *hybrid* menjadi lebih rumit dan tidak menunjukkan ciri khas tertentu (Rethfeldt, dkk., 2021).

Dalam penelitian ini, peneliti menggunakan topologi *tree* (Santhameena dan J., 2022). Topologi ini memiliki sebuah perangkat yang menjadi *root*, yaitu perangkat yang menduduki level teratas serta berperan sebagai pusat utama komunikasi bagi semua *node* yang terhubung (Zuo, dkk., 2021). Kemudian, pada level di bawahnya, terdapat perangkat yang menjadi pusat koneksi bagi perangkat lain, membentuk topologi *star* (Al Rantisi, dkk., 2017). Dalam penelitian ini, level terbawah terdiri atas *node*, level tengah berfungsi sebagai *cluster head*, sedangkan level tertinggi berperan sebagai *sink* (Amir dan Kusri, n.d.).

2.1.3 Performa WSN

Performa merupakan hasil evaluasi dari proses kinerja pada *Wireless Sensor Network WSN*. Kriteria penilaian performa pada *WSN* meliputi *throughput*, *packet loss*, dan *delay*. Performa mengacu pada keberhasilan atau pencapaian dalam suatu proses. Dalam kasus ini, performa yang dievaluasi mencakup kinerja *WSN* berdasarkan *delay*, *packet loss*, dan *throughput*.

Secara performa, *WSN* dapat mengalami peningkatan atau penurunan. Indikator penurunan performa pada *WSN* terlihat dari tingginya *delay* dan *packet loss*, yang disebabkan oleh berbagai permasalahan dalam jaringan, seperti padatnya lalu lintas protokol dan *collision data* (Achroufene, dkk., 2021). Dalam penelitian ini, indikator performa sangat penting untuk mengoptimalkan kinerja *WSN*, yang akan diterapkan pada sistem irigasi agar sistem tersebut dapat bekerja secara optimal (Bristow, dkk., 2020).

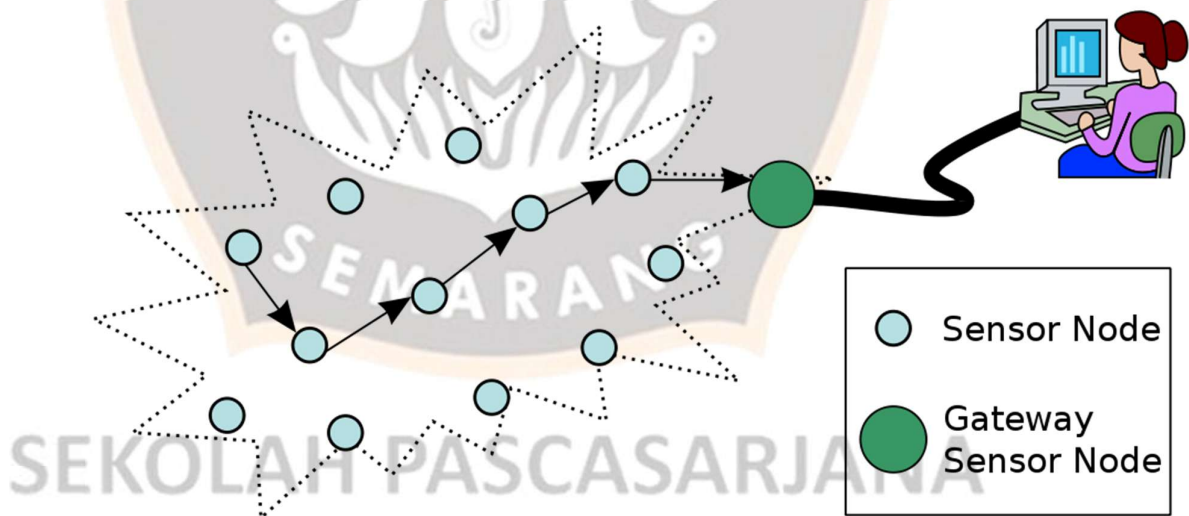
Salah satu metode yang diterapkan untuk meningkatkan performa pada *WSN* yakni Multi-Hop *LEACH* (M-*LEACH*). Dalam penelitian tersebut, digunakan parameter distance untuk memilih *cluster head* dengan jarak terdekat ke sink yang akan dipilih oleh member *node* (Fernando, dkk., 2020). Untuk mengatasi stray *node*, penelitian tersebut menggunakan model komunikasi *multi-hop* dengan Shortest Geopath Routing (SGR) atau Distance-Based Greedy Perimeter Stateless Routing (GPSR). Prinsip utama dari metode ini berfokus pada pemilihan *node* dengan jarak lebih dekat ke sink sebagai next-hop untuk meneruskan pesan, dengan

tujuan menemukan CH (Xia, dkk., 2020). Namun, dalam penelitian tersebut, faktor energi tidak diperhitungkan dalam penanganan masalah multiple CH maupun stray node dalam beberapa algoritma *routing WSN* yang telah ada. Jika faktor energi dipertimbangkan, maka nilai *throughput* dapat meningkat (Mahapatra dan Yadav, 2015).

2.1.4 Protokol WSN

Protokol merupakan media yang mengatur hubungan komunikasi serta perpindahan data antar komputer dan jaringan. Dalam protokol, terdapat aturan-aturan yang harus dipenuhi, baik oleh pengirim maupun penerima, agar komunikasi dapat terjalin dengan baik. Secara singkat, protokol berfungsi sebagai media perantara yang menghubungkan pengirim dengan penerima.

Beberapa protokol banyak digunakan untuk meningkatkan performa pada WSN. Dalam penelitian ini, dikembangkan protokol untuk meningkatkan performa pada WSN (Rizky, dkk., 2022). Protokol tersebut ditunjukkan pada Gambar 2.2.



Gambar 2. 2 Protokol

Protokol merupakan media yang mengatur hubungan komunikasi serta perpindahan data antar komputer dan jaringan. Dalam sistem, terdapat aturan-aturan yang harus dipenuhi, baik oleh pengirim maupun penerima, agar komunikasi dapat terjalin dengan baik. Secara singkat, protokol berfungsi sebagai media perantara yang menghubungkan pengirim dengan penerima (Mohinur Rahaman

dan Azharuddin, 2022). Beberapa protokol banyak digunakan untuk meningkatkan performa pada *WSN*. Dalam penelitian ini, dikembangkan protokol untuk meningkatkan performa pada *WSN* dengan menggunakan teknik *clustering*.

Protokol *routing SGP* merupakan salah satu protokol dalam kategori *location-based routing* (G. Wang dan Ma, 2011). Untuk menjalankan protokol ini, setiap *sensor node* harus memiliki penyedia layanan lokasi, seperti *GPS* (Liu, 2021). *Source node* menggunakan informasi geografis dari *node* tujuan untuk menemukan rute pengiriman (Chen dan P, 2021). Dalam *SGP*, *node* yang mengirim pesan akan meneruskan pesan ke tetangganya yang memiliki jarak paling dekat dengan tujuan (Sembiring, dkk., 2011). Jarak antara suatu *node* dengan tujuan dapat dihitung menggunakan rumus *Euclidean*, jika koordinat kedua *node* diketahui pada bidang dua dimensi. Protokol ini dapat meningkatkan performa *WSN* (Fraigniaud, 2005).

Syarat penggunaan protokol *SGP* mencakup hal-hal berikut:

1. Setiap *node* harus mengetahui posisi geografisnya.
2. Setiap *node* harus mengetahui posisi geografis tetangganya dalam satu langkah.
3. Tujuan *routing* didefinisikan sebagai sebuah *node* pada koordinat posisi dua dimensi atau sebagai sebuah area geografis yang direpresentasikan dengan poligon tertutup.
4. Setiap paket harus dapat membawa sejumlah kecil informasi *routing*.

Pada *WSN*, umumnya jangkauan radio dari suatu *node* terbatas. Oleh sebab itu, sangat mungkin terjadi penurunan performa dan *stray node* dalam algoritma *routing* berbasis *cluster*. *Stray node* merujuk pada kondisi di mana suatu *node* dalam jaringan tidak terintegrasi ke dalam *cluster* yang terbentuk (Li, dkk., 2017). Dalam *LEACH*, *stray node* terjadi jika, pada fase *setup*, *node* yang tidak menjadi *cluster head* tidak menerima pesan *CH advertisement* (Jasmine Lizy dan Chenthalir Indra, 2021).

Geographic Routing (*GR*) atau penentuan rute berdasarkan posisi geografis diperkenalkan untuk mengatasi permasalahan performa pada *WSN* serta keterbatasan protokol *routing* berbasis topologi (Kwong, dkk., 2007). Protokol *Geographic Routing* mengandalkan informasi lokasi fisik *node* dalam *WSN*, yang

diperoleh dari *location service* (misalnya, *GPS*). Dengan memanfaatkan informasi posisi geografis, *GR* tidak perlu melakukan pemeliharaan (*maintenance*) *routing table* dan bahkan dapat berjalan tanpa *routing table* (Tiglao, dkk., 2020).

Beberapa protokol *Geographic Routing (GR)* telah diperkenalkan, misalnya *Greedy Routing* yang dikembangkan oleh Takagi & Kleinrock (1980). Protokol ini menghitung semua jalur berdasarkan jarak *Euclidean* antara *sensor node* dan *sink node*, lalu memilih jalur yang sesuai (Thakur, dkk., 2020). Protokol-protokol terbaru umumnya merupakan pengembangan dari *Greedy Routing* dengan menambahkan modifikasi *clustering* pada protokol. Modifikasi ini dilakukan untuk mempertimbangkan faktor *delay*, *throughput*, *energy*, dan aspek lainnya agar performa *WSN* meningkat (Ilham, dkk., 2020).

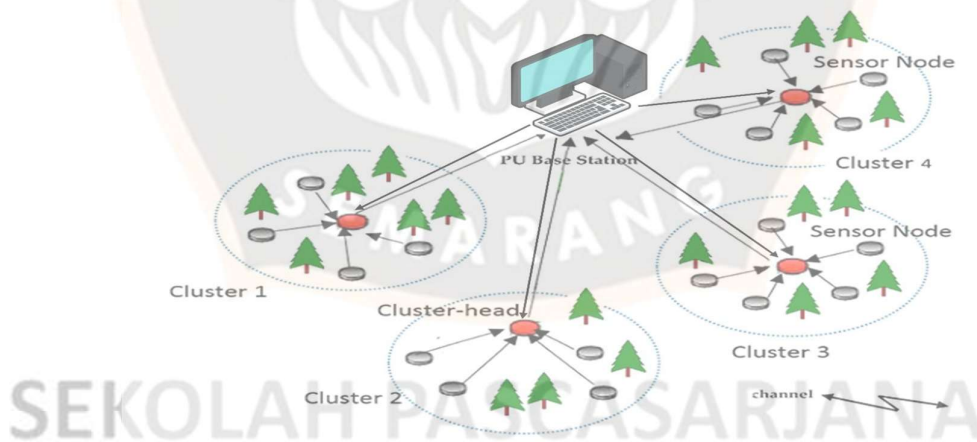
Sebuah metode protokol untuk menangani masalah performa dan *stray node* telah dikemukakan sebelumnya (Berthomier, dkk., 2021). *Energy-Efficient Multiple Cluster-Head Selection Routing Protocol (EEM-CHRP)* menangani rendahnya performa dan *stray node* pada *WSN* (Rehan, dkk., 2017). Hal tersebut dilakukan untuk mengurangi jumlah komunikasi langsung menuju *sink* yang boros energi. Namun, dalam *BW-CHRP*, model komunikasi *one-hop* tetap digunakan, dan luas area yang disimulasikan relatif kecil, yaitu 100×100 m, dengan kekuatan transmisi radio tiap *node* yang tidak terbatas (Perveen dan Medicine, 2017).

Protokol *Zigbee* merupakan protokol yang dikembangkan oleh *Zigbee Alliance*, di mana protokol ini dapat meningkatkan performa pada *WSN* dengan menggunakan komunikasi berbasis *MAC* dan *PHY layer* (Xia, dkk., 2020). Namun, protokol ini sudah dilengkapi dengan beberapa fitur; salah satunya, *node* yang menggunakan protokol ini dapat menjadi tiga jenis *node*, yakni *end device*, *at-router*, dan *coordinator* (Zhang dan Mao, 2021). Dengan demikian, topologi yang digunakan pun lebih beragam (Li, dkk., 2017).

Beberapa protokol *Geographic Routing (GR)* atau penentuan rute berdasarkan posisi geografis diperkenalkan untuk mengatasi keterbatasan protokol *routing* berbasis topologi (Hu, dkk., 2020). Protokol *Geographic Routing* mengandalkan informasi lokasi fisik *node* dalam *WSN* yang diperoleh dari *location service* (Xia, dkk., 2020). Dengan memanfaatkan informasi posisi

geografis, *GR* tidak perlu melakukan pemeliharaan *routing table* dan bahkan bisa berjalan tanpa *routing table* sama sekali (Zuo, dkk., 2021). Beberapa protokol *GR* telah diperkenalkan, misalnya *Greedy Routing* yang dikembangkan oleh Takagi, yang menghitung semua jalur berdasarkan jarak *Euclidean* dari *sensor node* menuju *sink node*, lalu memilih jalur yang paling sesuai.

Protokol-protokol terbaru pada umumnya merupakan pengembangan dari protokol *Greedy Routing* dengan cara menambahkan modifikasi tertentu sesuai dengan pertimbangan *delay*, *throughput*, *energy*, dan lain-lain (Novák, dkk., 2020). Dari beberapa protokol yang telah dikembangkan, *Shortest Geopath Routing (SGP)* merupakan salah satu protokol yang banyak diteliti pada *WSN* karena menjanjikan *packet delivery ratio* yang baik dengan waktu *delay* yang rendah (Beliakov, dkk., 2020). Selain itu, protokol *SGP* tidak memerlukan komputasi yang rumit, sehingga cocok untuk diterapkan pada *WSN* yang memiliki kemampuan komputasi terbatas (Astuti dan Wibisono, 2017). Visualisasi sistem irigasi cerdas ditunjukkan pada Gambar 2.3.



Gambar 2. 3 Visualisasi Sistem Irigasi Cerdas

Proses pertukaran data dalam jaringan *WSN* melibatkan pembacaan suhu dan kelembapan oleh *node sensor* menggunakan perangkat sensor lingkungan. Setiap *sensor* dalam jaringan akan membaca kondisi suhu dan kelembapan di sekitarnya, kemudian memproses data tersebut dan menyiapkannya untuk dikirimkan ke *cluster head* terdekat. *Cluster head* selanjutnya bertanggung jawab

untuk mengumpulkan, merangkum, dan mengirimkan data yang telah diproses ke *base station* atau *server* pusat untuk analisis lebih lanjut.

Permasalahan dalam proses pertukaran data pada jaringan sensor nirkabel (WSN). Dalam implementasinya, proses pertukaran data pada *Wireless Sensor Network* (WSN) kerap menghadapi sejumlah tantangan utama yang berdampak pada kinerja jaringan, antara lain:

- *Delay*, yaitu tingginya waktu latensi dalam pengiriman data yang disebabkan oleh kemacetan jaringan serta mekanisme routing yang tidak efisien;
- *Packet loss*, yakni hilangnya paket data selama proses transmisi yang umumnya dipicu oleh gangguan sinyal, interferensi elektromagnetik, atau antrean buffer yang melampaui kapasitas;
- *Jitter*, yaitu fluktuasi waktu kedatangan antar paket yang tidak konsisten, sebagai akibat dari perbedaan jalur transmisi atau ketidakstabilan dalam proses pengolahan data di setiap *node*;
- *Throughput* rendah, yaitu rendahnya laju transfer data yang dapat ditransmisikan secara efektif dalam satuan waktu, yang sering kali disebabkan oleh keterbatasan kapasitas jaringan dan alokasi channel komunikasi yang tidak optimal.

Untuk mengatasi permasalahan tersebut, penelitian ini bertujuan mengembangkan protokol komunikasi pada WSN berbasis integrasi algoritma *Low Energy Adaptive Clustering Hierarchy* (LEACH) dan *Hierarchical Agglomerative Clustering* (HAC). Pendekatan ini difokuskan pada peningkatan efisiensi dan performa jaringan melalui mekanisme pembentukan *cluster head* secara adaptif serta pengelompokan *node* berdasarkan kedekatan spasial.

Selanjutnya, algoritma *fuzzy logic* digunakan sebagai metode pendukung dalam menentukan struktur klaster yang optimal. Dalam konteks ini, parameter lingkungan seperti suhu dan kelembapan dijadikan sebagai input, yang kemudian di-fuzzifikasi ke dalam kategori linguistik (misalnya: panas, dingin, basah, dan kering) guna menghasilkan keputusan klasterisasi yang lebih adaptif dan kontekstual.

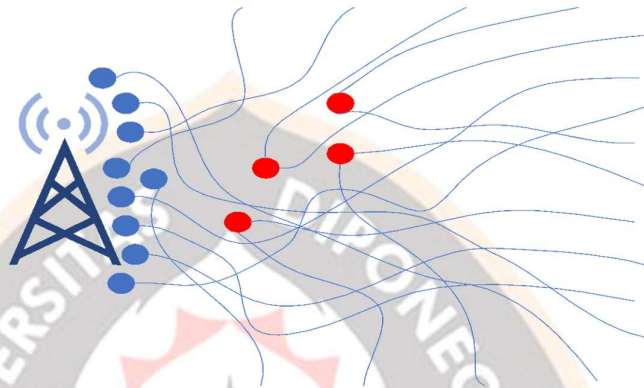
Proses pengelompokan *node* dilakukan melalui pendekatan *HAC* menggunakan metode *single linkage*, yang mampu membentuk *cluster* berbasis kedekatan antar *node*. Untuk mendukung efisiensi komunikasi dan menghindari gangguan antar *cluster*, diterapkan pula pendekatan *multi-channel*, di mana setiap *cluster* dapat menggunakan *channel* komunikasi berbeda secara paralel. Ini memungkinkan komunikasi simultan antar *node* dan *cluster head* tanpa terjadi interferensi sinyal, serta meningkatkan *Quality of Service (QoS)* pada parameter *delay*, *packet loss*, *jitter*, dan *throughput*. Dengan demikian, pengembangan struktur *cluster* yang adaptif berbasis *LEACH-HAC* dan *multi-channel* ini diharapkan dapat menyeimbangkan beban jaringan dan mengurangi kemacetan dalam proses pengiriman data.

Sistem cerdas merujuk pada sistem komputer canggih yang dapat mengumpulkan, menganalisis, dan merespons data yang dikumpulkan dari lingkungan sekitarnya (Ampatzidis, dkk., 2020). Sistem cerdas yang akan diterapkan pada irigasi dapat menyirami tanaman secara otomatis dengan mengumpulkan data berupa suhu dan kelembapan dari lingkungan sekitarnya (Pachayappan, dkk., 2020).

Sistem irigasi cerdas ini tidak lepas dari teknologi *Wireless Sensor Network* agar dapat terkoneksi satu dengan yang lainnya (Zhuo, dkk., 2018). Berbagai jenis *sensor* dapat membantu kinerja irigasi cerdas tersebut (Hadipour, dkk., 2020). *Sensor* tersebut bekerja dengan mendeskripsikan sistem yang telah dibangun melalui penerapan algoritma *fuzzy logic* untuk menjabarkan variabel suhu dan kelembapan agar pengairan menjadi lebih optimal (Syamsiyah, dkk., 2020). Permasalahan yang sering terjadi berupa *collision data* (Rethfeldt, dkk., 2021).

Collision merujuk pada segmen jaringan fisik di mana paket data dapat bertabrakan satu sama lain saat melewati media bersama (Al-Sodairi dan Ouni, 2018). Satu *sensor* kelembapan dan satu *sensor* suhu mengirimkan data menuju *server*. Oleh karena itu, banyaknya *sensor* dapat mengakibatkan terjadinya *collision data*. Ketika data sudah bertabrakan, alat tidak akan bekerja secara maksimal, bahkan dapat menyebabkan alat mengalami *blank data* dan mati seketika. Selain itu, sistem pengairan otomatis juga akan berhenti berfungsi (Krishnan, 2020). Tabel komparasi penelitian yang dilakukan sebelumnya

diperlihatkan pada Tabel 2.1. *Collision data* ditunjukkan pada Gambar 2.4.



Gambar 2. 4 *Collision Data*

Node yang mengalir dari *sensor* menuju *base station* dapat diamati. *Node* berwarna biru mengindikasikan bahwa data yang mengalir dari *sensor* menuju *base station* berhasil dikirimkan. Sementara itu, *node* berwarna merah menunjukkan *node* yang mengalami tabrakan (*collision data*).

Setiap *sensor* mengeluarkan frekuensi radio setiap detik, kemudian data dikirimkan ke *base station*. Apabila pengiriman data melebihi kapasitas tampung *base station*, maka kinerjanya akan semakin berat dan dapat mengalami gangguan. Karena setiap *sensor* memiliki frekuensi yang sama, sangat mungkin terjadi tabrakan data.

Penelitian ini menggunakan sensor DHT22 sebagai alat pengukur suhu dan kelembaban udara. Sensor DHT22 dipilih karena memiliki kemampuan pengukuran yang cukup akurat, dengan rentang suhu dari -40°C hingga $+80^{\circ}\text{C}$ dan kelembaban dari 0% hingga 100% RH. Sensor ini bekerja secara digital dan hanya memerlukan tiga kabel utama, yaitu VCC, GND, dan DATA, serta membutuhkan sebuah resistor pull-up sebesar $10\text{ k}\Omega$ pada jalur data untuk menjaga kestabilan sinyal. Data hasil pengukuran akan dikirimkan dalam bentuk sinyal digital melalui kabel data ke mikrokontroler, kemudian diolah untuk ditampilkan atau disimpan.

Untuk mendukung komunikasi nirkabel antar *node* sensor, sistem ini menggunakan protokol komunikasi sinyal radio seperti *LoRa* atau *ZigBee*, yang

memungkinkan perpindahan data dari sensor ke gateway atau pusat pengolahan data secara efisien dan hemat daya. Hal ini sangat penting dalam sistem jaringan sensor nirkabel (*Wireless Sensor Network/WSN*), terutama pada implementasi irigasi cerdas di area pertanian yang luas. Dengan sistem ini, pengukuran suhu dan kelembaban dapat dilakukan secara real-time, efisien, dan tanpa perlu koneksi kabel langsung ke pusat data.



Gambar 2. 5 Flowchart Sistem Irigasi Cerdas Berbasis IoT

Gambar menjelaskan proses otomatisasi sistem irigasi yang dimulai dari pembacaan data lingkungan hingga aktivasi sistem irigasi berdasarkan data yang dikirim melalui protokol sinyal radio. Berikut adalah uraiannya:

1. Start (Mulai)

Proses dimulai ketika sistem dinyalakan.

2. Pembacaan Suhu dan Kelembaban

Sensor DHT22 melakukan pengukuran suhu dan kelembaban udara di lingkungan sekitar. Sensor ini dipilih karena keakuratannya dan jangkauan pengukuran yang luas, yaitu dari -40°C hingga $+80^{\circ}\text{C}$ untuk suhu dan dari 0% hingga 100% RH untuk kelembaban.

3. Konversi Data ke Format Digital

Data analog yang dibaca oleh sensor dikonversi menjadi sinyal digital sehingga dapat diproses oleh sistem mikrokontroler.

4. Pengiriman Data Melalui Protokol Sinyal Radio

Data digital hasil pengukuran kemudian dikirim secara nirkabel menggunakan protokol sinyal radio ke pusat kendali, memungkinkan sistem bekerja tanpa kabel.

5. Aktivasi Irigasi Otomatis

Berdasarkan data suhu dan kelembaban yang diterima, sistem akan memutuskan untuk mengaktifkan irigasi otomatis jika memenuhi kondisi tertentu (misalnya, suhu tinggi dan kelembaban rendah).

6. Pengulangan Proses Aktivasi Irigasi (Duplikasi Tahap)

Terdapat pengulangan kotak "Automatic irrigation is activated" di akhir, kemungkinan merupakan duplikasi visual yang perlu diperbaiki. Satu kotak sudah cukup untuk menunjukkan bahwa irigasi telah aktif.

SEKOLAH PASCASARJANA

Tabel 2. 1 Tinjauan Penelitian performa WSN

Penulis, Tahun	Metode	Pembentukan Cluster Head	Tahapan	Hasil
Michael Rethfeldt, 2021	Adaptive Multi channel Wireless Mesh Network	Ya	Menggunakan <i>greedy algorithm</i> dengan mengalokasikan <i>channel</i> terbaik berdasarkan kualitas sinyal	<i>Graph Coloring Algorithm</i> memodelkan jaringan sebagai graf, di mana <i>node</i> merupakan titik <i>link</i>
Xun Chen, 2006	Multi channel MAC Protocol for Wireless Sensor Network	Ya	Setiap <i>node</i> atau <i>link</i> diberikan <i>channel</i> tertentu sebelum operasi jaringan dimulai	<i>node</i> atau <i>link</i> tetap pada <i>channel</i> yang sama tanpa perubahan dinamis
Kae-Hsiang Kwong, 2007	Self-Organizing Multi channel Medium Access Control (SMMAC)	Ya	<i>Node</i> mendeteksi kekuatan sinyal pada setiap <i>channel</i> untuk mengidentifikasi <i>channel</i> yang kosong	<i>Node</i> mengukur SNR pada setiap <i>channel</i> untuk memilih <i>channel</i> dengan kualitas terbaik
Md. Abdul Hamid, 2011	Multi channel MAC Protocol WSN	Ya	<i>Node</i> memilih <i>channel</i> terbaik yang tersedia berdasarkan pengukuran <i>real-time</i> kualitas sinyal	
Fraigniaud, 2005	Location-Based Routing (SGP)	Ya	Memanfaatkan informasi lokasi fisik untuk menentukan rute yang efisien	
Perveen dan Medicine, 2017	Energy-Efficient Multiple Cluster-Head Selection Routing Protocol (EEM-CHRP)	Ya	Menggunakan kriteria tertentu untuk memilih <i>node</i> yang sesuai sebagai kepala <i>cluster</i>	Pemilihan kepala <i>cluster</i> dilakukan secara berkala atau berdasarkan peristiwa tertentu dalam jaringan.

(Mahapatra, 2015)	Multi-Hop LEACH (M- LEACH)	Ya	Data yang dihasilkan oleh <i>node-node</i> sensor dalam <i>cluster</i> tersebut diteruskan ke <i>node</i> tetangga	Rute yang dipilih untuk mentransfer data melalui beberapa <i>hop</i> dengan memperhitungkan kekuatan sinyal Setelah jalur terpendek ditentukan, data dikirimkan melalui beberapa <i>hop</i> dengan melompati <i>node</i> sensor di sepanjang jalur terpendek
(Yadav, 2015)	Multi-Hop dengan <i>Shortest Geopath Routing</i> (SGR)	Ya	Menggunakan algoritma jalur terpendek untuk menentukan rute terbaik.	Paket data dikirim dari <i>node</i> pengirim ke <i>node</i> penerima menggunakan informasi lokasi untuk menentukan jalur terbaik <i>Node</i> pengirim ingin mengirimkan paket data ke <i>node</i> tujuan dengan memilih tetangga terdekat
Astuti dan Wibisono, 2017	Protokol <i>Geographic Routing</i> (GR)	Ya		Paket data kemudian diteruskan ke <i>node</i> dengan mengikuti jalur yang ditentukan. Meneruskan data ke <i>node</i> tetangga yang dipilih dengan harapan <i>node</i> tetangga tersebut lebih dekat.
Ilham dkk., 2020	Protokol <i>Greedy Routing</i>	Ya		
(Moorthi dan Thiagarajan, 2020)	<i>low-energy adaptive clustering hierarchy</i> (LEACH)	Ya	Sinyal terkuat menjadi <i>cluster head</i> (CH)	CH akan bergantian pada setiap <i>round</i> di setiap sesi

Tinjauan dan hasil penelitian

Dari beberapa tinjauan pustaka, secara teori, metode tersebut dapat meningkatkan performa pada *WSN*. Namun, berbagai model tersebut mengalami kenaikan *throughput* yang tidak begitu signifikan karena masih memiliki beberapa kekurangan, seperti pembentukan *node* yang saling berjauhan.

Penelitian yang akan dilakukan (Positioning penelitian)

Mengombinasikan pendekatan *LEACH*, *euclidean*, dan *HAC single linkage* sehingga mampu mengelompokkan *node* pada *WSN* agar *node* tersebut dapat dikelompokkan antara *cluster head* dan *node* secara berdekatan. Selain itu, dalam penelitian ini, jalur/*channel* pada *WSN* dapat dibagi untuk mengurai kepadatan lalu lintas yang ada pada jalur protokol *WSN*.

2.2 Keaslian (orisinalitas) penelitian

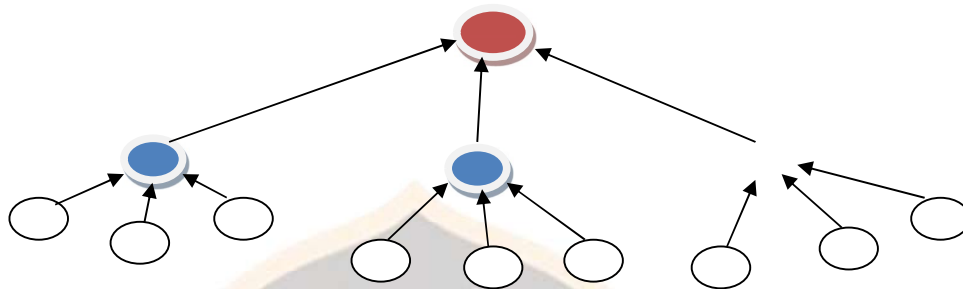
Keaslian (*orisinalitas*) penelitian ini terletak pada pengembangan *LEACH-HAC* sebagai solusi untuk meningkatkan performa *throughput* pada jaringan *WSN* yang digunakan dalam sistem irigasi cerdas. Metode ini menggabungkan pendekatan *multi-channel* dan *hierarchy clustering*, yang memungkinkan pembagian jaringan ke dalam beberapa *cluster* dengan efisiensi komunikasi yang lebih tinggi.

Dalam skema ini, setiap *node* dan *cluster head* ditempatkan secara adaptif dan berdekatan untuk meminimalkan jarak komunikasi. Pengelompokan data dilakukan secara terstruktur, sehingga mempercepat transmisi data. Penelitian ini secara khusus dirancang untuk mengatasi permasalahan keterbatasan *throughput* dan efisiensi komunikasi dalam skenario irigasi cerdas berbasis *WSN*, sekaligus memberikan kontribusi terhadap desain protokol routing yang lebih adaptif.

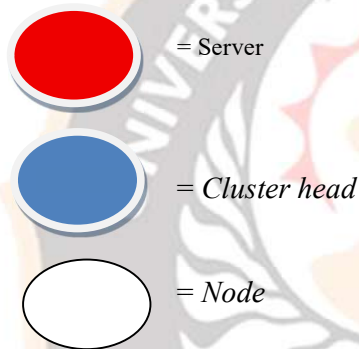
2.3 Landasan Teori

2.3.1 Clustering pada Wireless Sensor Network

Teknik *clustering* pada *WSN* umumnya diterapkan untuk mengoptimalkan penggunaan sumber daya energi. Salah satu protokol *clustering* yang banyak digunakan yakni *Low-Energy Adaptive Clustering Hierarchy*. Protokol routing tersebut bertujuan untuk meningkatkan konservasi energi (Derdar, dkk., 2022). Arsitektur protokol ditunjukkan pada Gambar 2.6.



Gambar 2. 6 Arsitektur Protokol



Node yang berwarna biru merupakan *cluster head* yang berperan sebagai acuan komunikasi antar *node* dalam jaringan, sementara *node* yang berwarna putih merupakan anggota *channel* dalam suatu *cluster* (Bakr dan Lilien, 2014). Ide utama dari *Low-Energy Adaptive Clustering Hierarchy* adalah membagi jaringan menjadi beberapa *cluster* berdasarkan kekuatan sinyal (*signal strength*), dengan menggunakan *cluster head* sebagai *router* untuk mengirim pesan ke *cluster* lain atau ke *base station* (N dan S, 2022).

Untuk meningkatkan efisiensi komunikasi dan mengurangi interferensi antar *cluster*, pendekatan ini diperluas dengan dukungan *multi-channel*, di mana setiap *cluster* dapat menggunakan *channel* komunikasi yang berbeda untuk transmisi data secara paralel dan simultan. Selain itu, pemrosesan data dilakukan di dalam *cluster head* sebelum dikirimkan, sehingga dapat mengurangi beban lalu lintas jaringan secara keseluruhan (Daanoune, dkk., 2021).

Sifat dinamis yang dimiliki oleh *Low-Energy Adaptive Clustering Hierarchy LEACH* membagi waktu menjadi beberapa interval (Arsana dan Kariada, 2019). Pada awal interval, *cluster head* ditentukan secara acak dari beberapa *node*

yang memiliki energi di atas level tertentu (Al-Sodairi dan Ouni, 2018). Setelah terpilih menjadi *cluster head*, *node* tersebut melakukan *broadcast* kepada *node* tetangga (Daanoun, dkk., 2021). Kemudian, *node* tetangga akan menisbatkan diri sebagai *member* kepada *cluster head* yang memiliki sinyal paling kuat (Mahapatra dan Yadav, 2015).

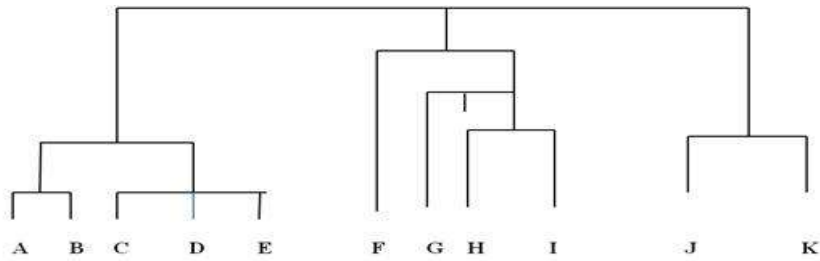
Selanjutnya, *cluster* terbentuk, dan *cluster head* akan menyusun jadwal *sampling* serta mengirimkannya ke seluruh *member node* (Santhameena dan J., 2022). *Member node* akan melakukan pemindaian sesuai jadwal dan mengirimkan hasil pemindaian kepada *cluster head* (Kavitha dan Ganapathy, 2022). Menjelang akhir interval, *cluster head* akan mengirim data yang telah dikumpulkan menuju *base station* (Bristow, dkk., 2020).

Meskipun dapat menghemat energi jaringan, *LEACH* juga memiliki kekurangan jika sebaran *sensor node* tidak merata (Babu M, dkk., 2022). *Cluster head* yang terpilih bisa berada di tepi *cluster*, sehingga tidak dapat menjangkau semua *member node* secara langsung (Geetha, dkk., 2012).

2.3.2 Metode *Clustering*

Clustering merupakan teknik yang digunakan dalam analisis data (Jasmine Lizy dan Chenthalir Indra, 2021). Hal ini terkait dengan proses pengelompokan data sedemikian rupa sehingga terbentuk kelompok yang memiliki derajat hubungan yang kuat antaranggota *cluster* yang sama, dengan mengelompokkan anggota lainnya ke dalam kelompok berbeda karena memiliki derajat hubungan yang lemah (Zhu, dkk., 2021).

Clustering dapat dikatakan sebagai alat penemuan karena metode ini mampu menemukan hubungan informasi melalui struktur yang ada (Salim, dkk., 2022). Data yang belum jelas dapat berubah menjadi sebuah hubungan baru dalam bentuk informasi yang bermanfaat serta masuk akal saat ditemukan (Moorthi dan Thiagarajan, 2020). Misalnya, dalam kebutuhan analisis pola tertentu, *clustering* digunakan untuk pengelompokan, pengambilan keputusan, *machine learning*, *document retrieval*, segmentasi citra, dan pengelompokan pola (Zuo, dkk., 2021). Representasi *cluster* hierarki ditunjukkan pada Gambar 2.8.



Gambar 2. 7 Representasi *Cluster Hierarki*

Representasi *cluster* hierarki merupakan teknis yang memiliki tujuan utama untuk mengelompokkan objek-objek berdasarkan karakteristik yang dimiliki (Lee, dkk., 2020). Analisis *cluster* mengklasifikasikan objek sehingga setiap objek yang memiliki sifat mirip atau memiliki kesamaan paling dekat dapat dikelompokkan (Hung, dkk., 2020).

Penelitian ini akan mengelompokkan variabel yang sesuai dengan metode *single linkage* untuk menentukan kedekatan antarvariabel (Wu, dkk., 2018). Huruf A sampai dengan K menggambarkan *output* berupa variabel yang akan dikelompokkan (Zuo, dkk., 2021).

Algoritma *clustering* lainnya merujuk pada *cluster* hierarki, yang mampu membentuk pengelompokan secara berjenjang (Rethfeldt, dkk., 2021). Dengan demikian, algoritma ini membentuk beberapa *cluster* pada tingkat di atasnya dan menghasilkan *subset* kelompok pada tingkat di bawahnya (Zuo, dkk., 2021).

Proses pengelompokan data dengan menggunakan pendekatan *HAC* dengan *single linkage*. Pada penelitian ini, data yang dikelompokkan berupa variabel dengan mendesain kesesuaian menggunakan teknik kesamaan pola (Ramos Emmendorfer dan de Paula Canuto, 2021). Dengan teknik pengelompokan ini, diyakini dapat membantu kinerja *Wireless Sensor Network (WSN)* dalam menerima dan mengirimkan data secara lebih optimal (Alrajeh, dkk., 2013).

Secara umum, tahapan *clustering* data adalah sebagai berikut (Basaran dan Günes, 2016).

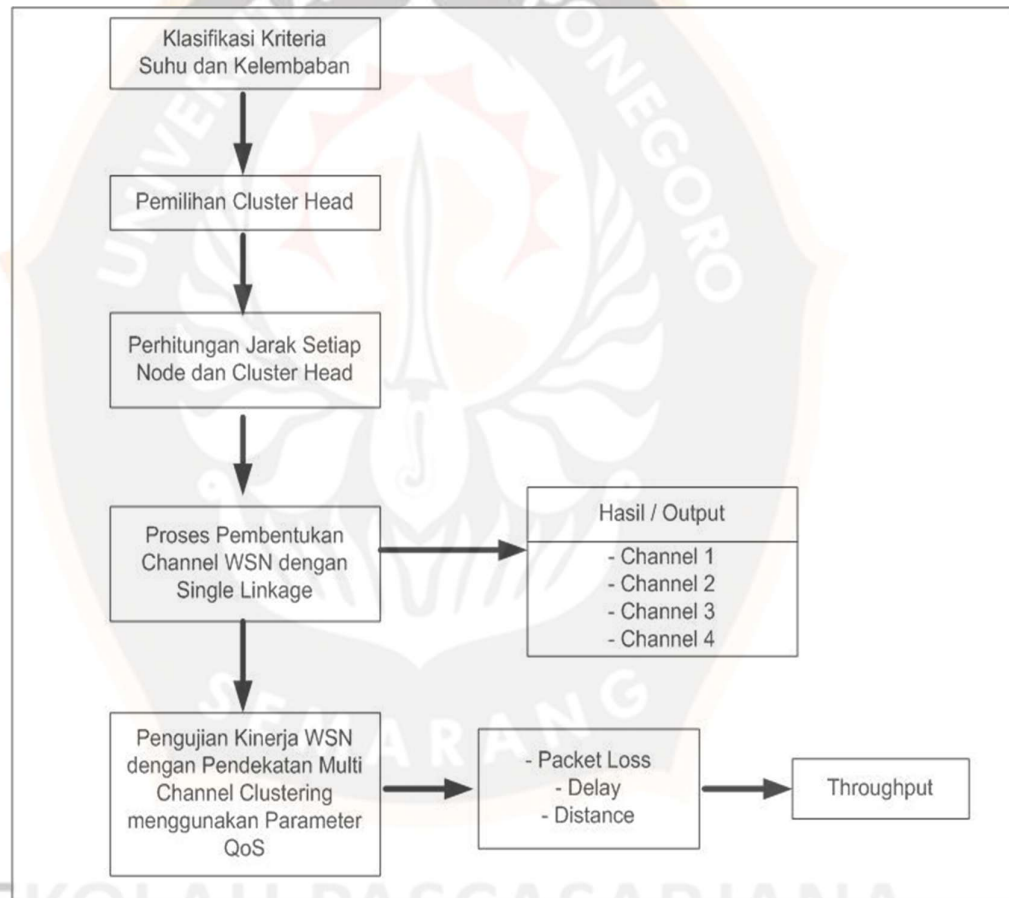
- a. *Representasi pola* (termasuk ekstraksi fitur dan/atau seleksi), mengacu pada penentuan jumlah kelas, jumlah pola yang tersedia, jenis, dan skala dari fitur yang tersedia untuk algoritma *clustering*;
- b. Definisikan ukuran pola kedekatan sesuai dengan domain data, biasanya diukur

dengan fungsi jarak yang ditentukan pada pasangan pola. Misalnya, menggunakan *Euclidean distance*;

c. Lakukan *clustering* atau pengelompokan;

d. *Abstraksi data* (jika diperlukan);

e. *Penilaian output* (jika diperlukan).



Gambar 2. 8 Alur *LEACH-HAC*

Metode yang digunakan pada penelitian ini memiliki empat langkah yang dirancang untuk memecahkan permasalahan *delay* dan *data loss* pada *WSN*. Hasil dari penerapan langkah-langkah tersebut berupa pembentukan *channel* atau saluran pada *WSN*, dan metode ini diuji menggunakan parameter *QoS*. Hasil pengujian menunjukkan adanya peningkatan performa pada *WSN* yang ditandai dengan meningkatnya nilai *throughput*.

2.3.4 Fuzzy Logic

Logika *fuzzy* merupakan suatu cara yang tepat untuk memetakan suatu ruang input ke dalam ruang output (Mikrokontroler, 2017). Secara umum, sistem *fuzzy* sangat cocok untuk penalaran pendekatan, terutama untuk sistem yang menangani masalah-masalah yang sulit didefinisikan dengan menggunakan model matematis, misalnya, nilai masukan dan parameter sebuah sistem yang bersifat kurang akurat atau kurang jelas, sehingga sulit mendefinisikan model matematikanya (Al-Ali, dkk, 2020). Sistem *fuzzy* memiliki beberapa keuntungan bila dibandingkan dengan sistem tradisional, misalnya pada jumlah aturan yang digunakan (Beliakov, dkk, 2020).

Pemrosesan awal sejumlah besar nilai menjadi sebuah nilai derajat keanggotaan pada sistem *fuzzy* mengurangi jumlah nilai yang harus dipergunakan pengontrol untuk membuat suatu keputusan (Sudharshan, dkk, 2019).

Ada beberapa metode untuk merepresentasikan hasil logika *fuzzy*, yaitu metode *Tsukamoto*, *Sugeno*, dan *Mamdani*. Pada metode *Tsukamoto*, setiap konsekuen direpresentasikan dengan himpunan *fuzzy* dengan fungsi keanggotaan monoton. Output hasil inferensi dari setiap aturan berupa z (Karyaningsih, 2020). Untuk metode *Sugeno*, mirip dengan metode *Mamdani*, hanya output (konsekuen) tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan linear (Camargos, dkk, 2020). Ada dua model metode *Sugeno*, yaitu model *Fuzzy Sugeno* orde nol dan model *Fuzzy Sugeno* orde satu (V́ictor, dkk, 2020).

Penelitian ini menggunakan *Fuzzy Sugeno* (Kapil Arasu dan Prakash, 2020). Metode ini bertujuan untuk memaksimalkan kinerja sensor kelembaban dengan mengubah data yang tidak pasti menjadi data yang lebih akurat setelah melalui proses pengolahan (Chang, dkk, 2011). *Fuzzy Sugeno* terbagi menjadi dua jenis, yaitu *Fuzzy Sugeno* orde nol dan *Fuzzy Sugeno* orde satu (Fale dan Abdulsalam, 2020).

Dalam penelitian ini, proses klasifikasi data suhu dan kelembaban menggunakan pendekatan *fuzzy*. Metode ini bertujuan untuk memaksimalkan kinerja sensor kelembaban dengan mengubah data yang tidak pasti menjadi data yang lebih akurat setelah diolah (Maalej, dkk, 2018). Fungsi keanggotaan merujuk

pada grafik yang mewakili besar derajat keanggotaan masing-masing variabel input, yang berada dalam interval antara 0 dan 1 (Cheng, dkk, 2020). Derajat keanggotaan sebuah variabel x dilambangkan dengan simbol $\mu(x)$ (Castillo dan Melin, 2020). Aturan-aturan (*Rules*) menggunakan nilai keanggotaan sebagai faktor bobot untuk menentukan pengaruhnya saat melakukan inferensi dalam menarik kesimpulan (LI, dkk, 2018). Ada beberapa fungsi yang dapat digunakan, namun dalam penelitian ini, peneliti menggunakan fungsi keanggotaan kurva bahu dan kurva segitiga (Jaiswal dan Ballal, 2020).

Derajat keanggotaan sebuah variabel x dilambangkan dengan simbol $\mu(x)$

a. Representasi kurva bahu

Fungsi keanggotaan yang merepresentasikan kurva bahu kiri (*left shoulder*) sebagai berikut:

$$\mu(x) = \begin{cases} 1, & x \leq a \\ \frac{b-x}{b-a} & a < x < b \\ 0, & x \geq b \end{cases}$$

Fungsi keanggotaan yang merepresentasikan kurva bahu kanan (*right shoulder*) sebagai berikut:

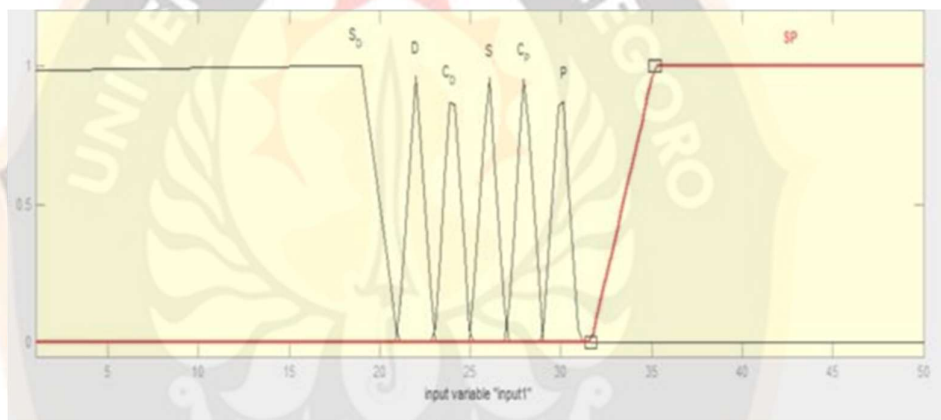
$$\mu(x) = \begin{cases} 0, & x \leq b \\ \frac{x-b}{c-b} & b < x < c \\ 1, & x \geq c \end{cases}$$

b. Representasi kurva segitiga

Fungsi keanggotaan yang merepresentasikan kurva segitiga sebagai berikut:

$$\mu(x) = \begin{cases} 0, & x \leq a \text{ atau } x \geq c \\ \frac{x-a}{b-a} & a < x < b \\ \frac{c-x}{c-b} & b < x < c \end{cases}$$

- a merujuk pada nilai domain terkecil dengan derajat keanggotaan nol.
- b merupakan nilai domain dengan derajat keanggotaan satu.
- c merupakan nilai domain terbesar dengan derajat keanggotaan nol.
- x merepresentasikan nilai input atau output yang akan dikonversi ke dalam bilangan fuzzy.
- μ menunjukkan fungsi keanggotaan dengan nilai $\{0,1\}$. Fungsi ini digunakan untuk memetakan kembali nilai fuzzy menjadi nilai crisp, yang berfungsi sebagai output atau solusi dari permasalahan. Variabel suhu ditunjukkan pada gambar 2.9.



Gambar 2. 9 Derajat Keanggotaan Variabel Suhu

Jika suhu kurang dari 21 derajat Celsius, maka masuk ke dalam kriteria sangat dingin. Jika suhu lebih dari 21 derajat Celsius dan kurang dari 23 derajat Celsius, maka masuk ke dalam kriteria dingin. Jika suhu lebih dari 23 derajat Celsius dan kurang dari 25 derajat Celsius, maka dikategorikan sebagai cukup dingin. Suhu lebih dari 25 derajat Celsius dan kurang dari 27 derajat Celsius termasuk dalam kategori sejuk.

Selanjutnya, jika suhu lebih dari 27 derajat Celsius dan kurang dari 29 derajat Celsius, maka diklasifikasikan sebagai cukup panas. Jika suhu lebih dari 29 derajat Celsius dan kurang dari 31 derajat Celsius, maka masuk ke dalam kategori panas. Sementara itu, jika suhu lebih dari 31 derajat Celsius, maka termasuk ke dalam kategori sangat panas.

Adapun kode atau singkatan yang digunakan adalah sebagai berikut:

- SD: sangat dingin
- D: dingin
- CD: cukup dingin
- S: sejuk
- CP: cukup panas
- P: panas
- SP: sangat panas

Berikut adalah representasi notasi *fuzzy* untuk tiap label suhu:

$$SD(x) = \begin{cases} 1, & 5 \leq x \leq 20 \\ \frac{21-x}{1}, & 20 \leq x \leq 21,1 \\ 0, & x \geq 35 \end{cases} \quad (2.1)$$

$$D(x) = \begin{cases} 0, & x \leq 21 \text{ atau } \geq 23 \\ \frac{x-21}{1}, & 21 < x \leq 22 \\ \frac{23-x}{1}, & 22 < x < 23 \end{cases} \quad (2.2)$$

$$CD(x) = \begin{cases} 0, & x \leq 23 \text{ atau } x \geq 25 \\ \frac{x-23}{1}, & 23 < x \leq 24 \\ \frac{25-x}{1}, & 24 < x < 25 \end{cases} \quad (2.3)$$

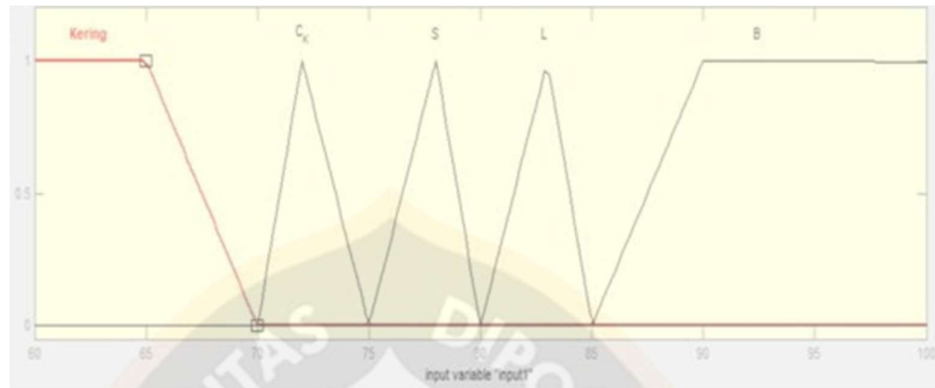
$$S(x) = \begin{cases} 0, & x \leq 25 \text{ atau } x \geq 27 \\ \frac{x-25}{1}, & 25 < x \leq 26 \\ \frac{27-x}{1}, & 26 < x < 27 \end{cases} \quad (2.4)$$

$$CP(x) = \begin{cases} 0, & x \leq 27 \text{ atau } x \geq 29 \\ \frac{x-27}{1}, & 27 < x \leq 28 \\ \frac{29-x}{1}, & 28 < x < 29 \end{cases} \quad (2.5)$$

$$P(x) = \begin{cases} 0, & x \leq 29 \text{ atau } x \geq 31 \\ \frac{x-29}{1}, & 29 < x \leq 30 \\ \frac{31-x}{1}, & 30 < x < 31 \end{cases} \quad (2.6)$$

$$SP(x) = \begin{cases} 0, & x \leq 31 \\ \frac{x-31}{1}, & 31 < x < 32 \\ 1, & x \geq 32 \end{cases} \quad (2.7)$$

SEKOLAH PASCASARJANA



Gambar 2. 10 Derajat Keanggotaan Variabel Kelembaban Tanah

Jika kelembapan kurang dari 70 persen, maka masuk ke dalam kriteria kering. Jika kelembapan lebih dari 70 persen dan kurang dari 75 persen, maka masuk ke dalam kriteria cukup kering. Jika kelembapan lebih dari 75 persen dan kurang dari 80 persen, maka dikategorikan sebagai sedang. Jika kelembapan lebih dari 80 persen dan kurang dari 85 persen, maka termasuk ke dalam kriteria lembap. Sementara itu, jika kelembapan lebih dari 85 persen, maka masuk ke dalam kategori basah.

Adapun kode atau singkatan yang digunakan adalah sebagai berikut:

- K: kering
- CK: cukup kering
- S: sedang
- L: lembap
- B: basah

Berikut adalah representasi notasi *fuzzy* untuk tiap label kelembaban:

$$K(x) = \begin{cases} 1, & x \leq 69 \\ \frac{70-x}{5} & 69 < x < 70 \\ 0, & x \geq 70 \end{cases} \quad (2.8)$$

$$C K(x) = \begin{cases} 0, & x \leq 70 \text{ atau } x \geq 75 \\ \frac{x-70}{2,5} & 70 < x \leq 72,5 \\ \frac{75-x}{2,5} & 72,5 < x < 75 \end{cases} \quad (2.9)$$

$$S(x) = \begin{cases} 0, & x \leq 75 \text{ atau } x \geq 80 \\ \frac{x-75}{2,5} & 75 < x \leq 77,5 \\ \frac{80-x}{2,5} & 77,5 < x < 80 \end{cases} \quad (2.10)$$

$$L(x) = \begin{cases} 0, & x \leq 80 \text{ atau } x \geq 85 \\ \frac{x-80}{2,5} & 80 < x \leq 82,5 \\ \frac{85-x}{2,5} & 82,5 < x < 85 \end{cases} \quad (2.11)$$

$$B(x) = \begin{cases} 0, & x \leq 85 \\ \frac{x-85}{1} & 85 < x < 86 \\ 1, & x \geq 86 \end{cases} \quad (2.12)$$

2.3.5 Random node

Simulasi merupakan proses representasi dari suatu sistem nyata beserta kondisi lingkungannya (*state of affairs*). Berbagai aspek dapat dirancang melalui pendekatan simulasi. Simulasi juga mencakup berbagai metode, salah satunya yang diterapkan dalam penelitian ini yaitu *Linear Congruent Method (LCM)*, yang dikenal sebagai metode sederhana (Daanoune, dkk, 2022).

Linear Congruent Method (LCM) sering digunakan untuk membangkitkan bilangan acak pada aplikasi-aplikasi permainan komputer, seperti permainan teka-teki, permainan menyusun huruf, dan aplikasi kuis. Pada permainan teka-teki, *Linear Congruent Method (LCM)* digunakan untuk mengacak posisi angka teka-teki yang akan disusun (Hung, dkk, 2020). Pada permainan menyusun huruf, *Linear Congruent Method (LCM)* digunakan untuk mengacak huruf-huruf, dan pada permainan kuis, penerapan *LCM* digunakan untuk mengacak pilihan jawaban. Pada penelitian ini, metode *LCM* digunakan untuk mengacak posisi *node* pada *Wireless Sensor Network (WSN)*, yang nantinya akan diolah menggunakan metode *LEACH-HAC* dengan tujuan untuk meningkatkan performa pada *WSN*.

$$X_n + 1 = (a X_n + C) \bmod m \quad (2.13)$$

X_n merepresentasikan node dengan nilai suhu dan kelembaban acak ke n , sementara A dan c mengacu pada konstanta *LCM*. dan M merupakan batas maksimum *node* suhu dan kelembaban acak. Algoritma *LCM* untuk *random node*, himpunan kandidat berisi elemen-elemen pembentuk solusi, sedangkan himpunan solusi berisi kandidat-kandidat yang terpilih sebagai solusi dari persoalan. Fungsi seleksi (*selection function*) memilih kandidat yang paling meyakinkan untuk mencapai solusi optimal. Fungsi kelayakan (*feasible function*) memeriksa apakah suatu kandidat yang terpilih dapat memberikan solusi yang layak, yaitu kandidat tersebut bersama-sama dengan himpunan solusi yang sudah terbentuk tidak

melanggar kendala yang ada. Fungsi objektif merujuk pada fungsi yang memaksimalkan atau meminimumkan nilai solusi.

2.3.6 Pemilihan *cluster head*

LEACH merupakan salah satu metode *hierarchy-based routing* yang dirancang untuk meningkatkan efisiensi energi dalam jaringan dengan membagi *node* ke dalam *cluster* (Almasri, dkk., 2022). Algoritma *LEACH* dimulai dengan memilih suatu *node* sebagai *cluster head*, kemudian menggunakan algoritma *clustering* untuk memilih *node* non-CH sebagai anggota, sehingga terbentuk *cluster* (Jasmine Lizy dan Chenthalir Indra, 2021).

Mekanisme ini menghemat energi karena hanya *cluster head* yang melakukan transmisi data ke *sink node*, sedangkan setiap *sensor node* cukup mengirim data ke *cluster head*-nya masing-masing. Akibatnya, konsumsi energi berkurang, sehingga waktu hidup jaringan menjadi lebih optimal (Singh, 2015).

Berikut adalah langkah-langkah algoritma penyelesaian *LEACH*. Rotasi acak dari *base station* atau *cluster head* serta pembentukan *cluster* yang sesuai. Kompresi lokal untuk mengurangi komunikasi global. Algoritma *LEACH* dibagi menjadi dua fase, yaitu fase *setup* dan fase *steady state*.

Algoritma dimulai dengan menentukan jumlah *cluster* beserta persentase *cluster head* yang diinginkan dan durasi masa aktif *node* tersebut selama menjadi *CH*. Setelah itu, setiap *node* memutuskan apakah akan menjadi *CH* pada sesi tersebut berdasarkan level energi yang tersisa. Pengambilan keputusan dilakukan oleh *node* dengan memilih angka acak antara 0 dan 1. Jika angka tersebut kurang dari batas *threshold*, maka *node* tersebut akan menjadi *CH* pada sesi tersebut (Fernandes, dkk., 2020).

Batas *threshold* dirumuskan sebagai berikut:

$$T(n) = \begin{cases} \frac{P}{1-P \times (r \bmod \frac{1}{P})} & \text{Jika } n \in G \\ 0 & \text{Jika tidak} \end{cases} \quad (2.14)$$

Ambang batas $T(n)$ pada setiap putaran menentukan apakah sebuah *node* akan menjadi *cluster head* berdasarkan suhu dan kelembaban. Sebuah *node* akan dipilih sebagai *cluster head* jika nilai acak m lebih besar dari $T(n)$ di mana m merepresentasikan bilangan acak dalam rentang 0 hingga 1.

Probabilitas suatu *node* menjadi *cluster head* dinyatakan dengan formula:

$$P(A) = \frac{A}{S}$$

Probabilitas $P(A)$ menunjukkan kemungkinan bahwa *node* dengan label A akan terpilih sebagai *Cluster head* dalam jaringan. Dalam konteks ini, A merepresentasikan identitas atau posisi dari *node* tertentu, sedangkan S mengacu pada jumlah total *node* yang terdapat dalam jaringan.

2.3.7 Perhitungan jarak setiap *node* dan *cluster head*

Pada tahap ini, proses penentuan jarak antar titik *node* yang tersebar pada *WSN* dilakukan. Selanjutnya, akan dicari titik yang terdekat dengan CH. Dalam menghitung jarak antara setiap sensor dan *cluster head*, perhitungan dilakukan menggunakan pendekatan *euclidean distance*. Dalam bidang matematika, *euclidean distance* merupakan jarak antara dua objek atau titik. *euclidean distance* dapat digunakan untuk mengukur kemiripan (*matching*) antar objek. Informasi yang dikumpulkan berupa jarak *euclidean* antara *node* dengan *neighbor* (Sembiring, dkk, 2011). *Euclidean distance* merepresentasikan panjang garis yang menghubungkan dua titik dalam suatu bidang koordinat. Jika koordinat berada pada bidang dua dimensi, maka jarak *euclidean* antara dua *node* dengan koordinat (x_1, y_1) dan (x_2, y_2) dapat dihitung menggunakan persamaan berikut:

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.15)$$

d adalah Jarak antara x dan y , x_i adalah *node* pada sumbu x posisi *cluster head* pada dimensi ke- i , y_i adalah *node* pada sumbu y pada dimensi ke- i , n adalah Jumlah dimensi dalam ruang vektor.

2.3.8 Proses pembentukan *channel WSN*

Setelah jarak antara *node* dan *cluster head* diperoleh, langkah berikutnya memasuki tahap analisis *clustering*. Dalam proses ini, setiap objek dikelompokkan sedemikian rupa sehingga memiliki karakteristik yang serupa atau tingkat kemiripan yang paling tinggi (Zuo, dkk, 2021).

Penelitian ini akan mengelompokkan variabel yang sesuai dengan metode *single linkage* untuk menentukan kedekatan (Delforge, dkk, 2021). *Single linkage* merupakan teknik *clustering* yang menghasilkan rangkaian pengurangan jumlah *cluster* pada setiap tahapannya. *Cluster* pada setiap tahap dibentuk dari tahap sebelumnya dengan menggabungkan dua *cluster* yang memiliki tingkat kemiripan tertinggi. Teknik ini dilakukan dengan dua pendekatan, yaitu teori *matriks* dan teori grafik (Yiğit, dkk, 2014).

Single linkage tidak memerlukan jumlah *cluster* yang spesifik dalam proses pengelompokannya, sehingga tidak perlu menentukan jumlah *cluster* yang akan dibentuk pada tahap awal. Meskipun demikian, dalam beberapa kasus tertentu, penentuan jumlah *cluster* dapat dilakukan dalam teknik ini. Dengan demikian, saat proses pengelompokan berlangsung, pemotongan *hierarki* dapat terjadi pada beberapa titik tertentu sesuai dengan kondisi awal pengelompokan (Du, dkk, 2015).

Tahapan Penyelesaian Menggunakan *Single Linkage*

- a. Tahap pertama, menentukan kriteria data berupa suhu dan kelembaban.
- b. Kriteria suhu: sangat dingin, dingin, cukup dingin, sejuk, cukup panas, panas, sangat panas. Kriteria kelembaban: kering, cukup kering, sedang, lembab, basah.

- c. Tahap kedua, membentuk *cluster head* yang menjadi acuan *channel* untuk digunakan oleh anggota *node*.
- d. Tahap ketiga, menentukan titik *node* terdekat menggunakan pendekatan *euclidean distance* saat mencari anggota *node* yang paling dekat dengan *cluster head*.
- e. Tahap keempat, menentukan anggota *node* dengan metode *single linkage clustering* berdasarkan kedekatan dengan *cluster head*. Proses dimulai dengan N *cluster*, di mana setiap *cluster* mengandung satu entitas tunggal dan sebuah *matriks* simetrik jarak (similarities) D berukuran N×N times N×N.

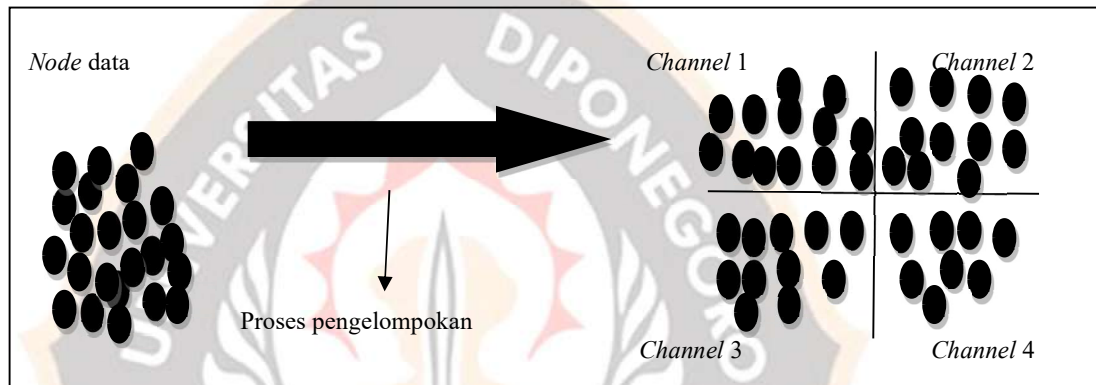
Proses Penggabungan Cluster

1. *Matriks* jarak *matriks* yang berisi jarak antara satu *node* dengan *node* lainnya. Dalam penelitian ini, proses dilakukan dengan menggabungkan pasangan *cluster* yang paling dekat (paling mirip). Misalnya, jika jarak antara *cluster* U dan V menjadi d UV, maka *cluster* tersebut dianggap paling mirip.
2. Penggabungan *cluster* U dan V dilakukan dengan memberi label *cluster* baru sebagai (UV). Setelah itu, update pada *matriks* jarak dilakukan dengan Langkah berikut:
 - a. Menghapus baris dan kolom yang bersesuaian dengan *cluster* U dan V.
 - b. Menambahkan baris dan kolom yang memberikan jarak antara *cluster* (UV) dengan *cluster-cluster* lainnya.
3. Ulangi langkah 2 hingga seluruh objek tergabung dalam satu *cluster* tunggal setelah algoritma selesai. Catat identitas *cluster* yang digabungkan serta tingkat-tingkat jarak atau similaritas saat penggabungan terjadi (Zuo dkk., 2021).

Selanjutnya proses *clustering* untuk mendesain *channel* sebagai saluran yang akan digunakan. Pendekatan yang digunakan yaitu pendekatan *single linkage*. Adapun caranya dengan mencari *matriks* jarak untuk pasangan *cluster* yang terdekat (paling mirip).

$$d_{(i,j),k} = \min(d_{ik}, d_{jk}). \quad (2.16)$$

$d_{(i,j),k}$ adalah jarak antara klaster gabungan i dan j dan klaster k yang dihitung berdasarkan prinsip *single linkage*, yaitu dengan memilih jarak terkecil antara dua kemungkinan d_{ik} , jarak antara elemen dari klaster i ke elemen dari klaster k , d_{jk} jarak antara elemen dari klaster j ke elemen dari klaster k .



Gambar 2. 11 Pembagian *Node* dengan Beberapa *Channel*

Pembagian *node* dengan beberapa *channel*. Adapun cara pengelompokan dilakukan dengan mencari *matriks* jarak untuk pasangan *cluster* yang terdekat (paling mirip), yang kemudian dikelompokkan menjadi beberapa *channel*. Berdasarkan model *Fuzzy* tersebut, terdapat beberapa tahapan algoritma yang harus dilakukan dalam implementasi metode sugeno dalam penelitian ini, yaitu sebagai berikut:

a. Pembentukan himpunan *fuzzy*

Pada tahap ini, variabel input suhu dan kelembaban digunakan dalam perhitungan nilai kebenaran dari premis pada setiap kriteria yang dimiliki oleh variabel tersebut. Kriteria pada variabel suhu terdiri dari, sangat dingin, dingin, cukup dingin, sejuk, cukup panas, panas, dan sangat panas. Kriteria pada variabel kelembaban terdiri dari, kering, cukup kering, sedang, lembab, dan basah. Dalam aturan basis pengetahuan, tahap ini bertujuan untuk mengambil nilai tegas (*crisp value*) dan menentukan derajat keanggotaan nilai tersebut dalam setiap himpunan *fuzzy* yang sesuai.

b. Pembentukan basis pengetahuan *fuzzy* (aturan)

Formula yang digunakan berupa kombinasi dari fungsi keanggotaan (derajat aktivasi) dan fungsi linier sugeno. Bentuk aturan yang digunakan dalam fungsi implikasi pada metode sugeno dalam penelitian ini sebagai berikut:

$$\text{IF } (X_1 \text{ is } A_1) * \dots * (X_n \text{ is } A_n) \text{ THEN } = Z \quad (2.17),$$

X_i menyatakan kriteria suhu, yang terdiri dari, sangat dingin, dingin, cukup dingin, sejuk, cukup panas, panas, dan sangat panas. A_i menyatakan variabel suhu. X_n menyatakan kriteria kelembaban, yang terdiri dari, kering, cukup kering, sedang, lembab, dan basah. A_n menyatakan variabel kelembaban. Z menyatakan kecepatan air untuk mengairi, yang terdiri dari, cepat, sedang, dan lambat. * Operator logika yang merepresentasikan hubungan antar himpunan fuzzy (biasanya berupa AND atau konjungsi) contoh if suhu adalah *cukup panas* and kelembaban adalah *lembab*, then kecepatan air adalah *sedang* (Chang dan Chang, 2004).

Aturan fuzzy dalam sistem pengairan yang Anda cantumkan didasarkan pada pengendalian dua input utama suhu variabel linguistik dengan nilai dingin, cukup dingin, normal, hangat, panas kelembaban variabel linguistik dengan nilai kering, normal, basah output (kecepatan sistem pengairan) output linguistik dengan nilai sangat cepat, cepat, sedang, lambat, sangat lambat, berhenti (Salamah, dkk, 2023).

Sistem fuzzy digunakan dalam pengambilan keputusan di bidang kontrol otomatis karena kemampuannya menangani ketidakpastian dan ketidaklinieran data input, seperti suhu dan kelembaban. Pada sistem irigasi otomatis, keputusan tentang kecepatan pengairan tidak selalu bisa dibuat secara logis dengan sistem biner, karena kondisi lingkungan bersifat kontinu dan subjektif (Zadeh, 1975).

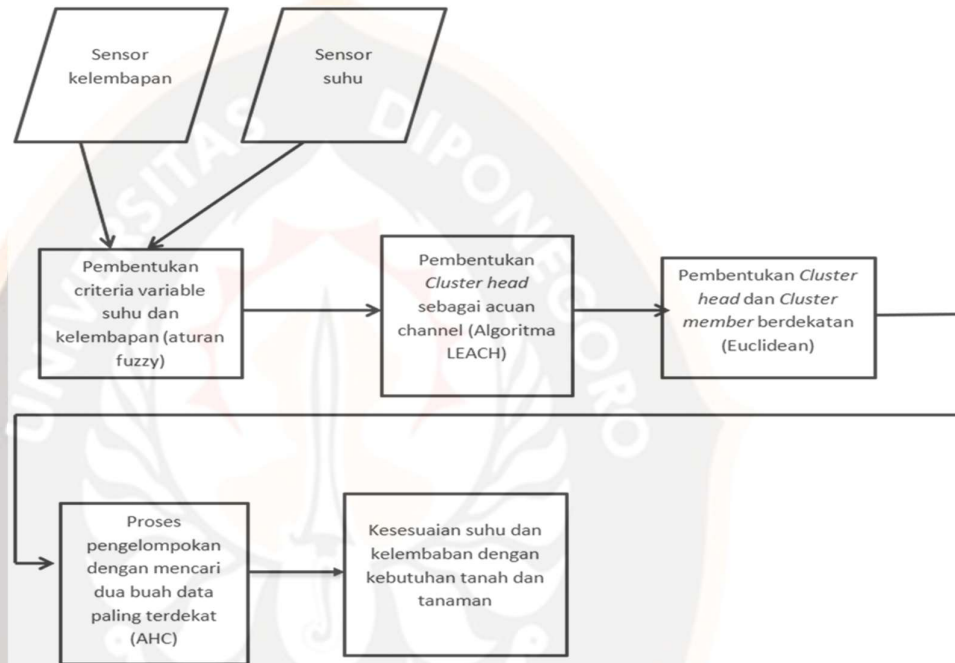
Aturan dalam sistem *fuzzy* yang digunakan dalam penelitian ini sebagai berikut:

1. Jika suhu dingin dan kelembaban kering, maka sistem pengairan berjalan dengan cepat.
2. Jika suhu cukup dingin dan kelembaban kering, maka sistem pengairan berjalan dengan cepat.
3. Jika suhu normal dan kelembaban kering, maka sistem pengairan berjalan dengan kecepatan sedang.

4. Jika suhu hangat dan kelembaban kering, maka sistem pengairan berjalan dengan lambat.
5. Jika suhu panas dan kelembaban kering, maka sistem pengairan berjalan dengan sangat lambat.
6. Jika suhu dingin dan kelembaban normal, maka sistem pengairan berjalan dengan cepat.
7. Jika suhu cukup dingin dan kelembaban normal, maka sistem pengairan berjalan dengan cepat.
8. Jika suhu normal dan kelembaban normal, maka sistem pengairan berjalan dengan kecepatan sedang.
9. Jika suhu hangat dan kelembaban normal, maka sistem pengairan berjalan dengan kecepatan sedang.
10. Jika suhu panas dan kelembaban normal, maka sistem pengairan berjalan dengan lambat.
11. Jika suhu dingin dan kelembaban basah, maka sistem pengairan berhenti mengalirkan air.
12. Jika suhu cukup dingin dan kelembaban basah, maka sistem pengairan berhenti beroperasi.
13. Jika suhu normal dan kelembaban basah, maka sistem pengairan berjalan dengan cepat.
14. Jika suhu hangat dan kelembaban basah, maka sistem pengairan berjalan dengan cepat.
15. Jika suhu panas dan kelembaban basah, maka sistem pengairan berjalan dengan pengairan sedang.

Aturan aturan ini dirancang untuk mencerminkan kebutuhan air yang optimal pada tanaman, di mana kelembaban kering menandakan kebutuhan irigasi yang tinggi, sementara kelembaban basah menunjukkan kondisi tanah yang sudah cukup air, bahkan pada suhu tinggi. Penyesuaian terhadap suhu ditujukan untuk menghindari efek stres termal pada tanaman akibat pemberian air yang terlalu cepat atau terlalu lambat (García, dkk, 2020).

Dalam basis pengetahuan *fuzzy*, tahap ini bertujuan untuk mengambil nilai tegas (*crisp value*) dan menentukan derajat keanggotaan nilai tersebut dalam setiap himpunan *fuzzy* yang sesuai. Alur *LEACH-HAC* ditunjukkan pada Gambar 2.12.

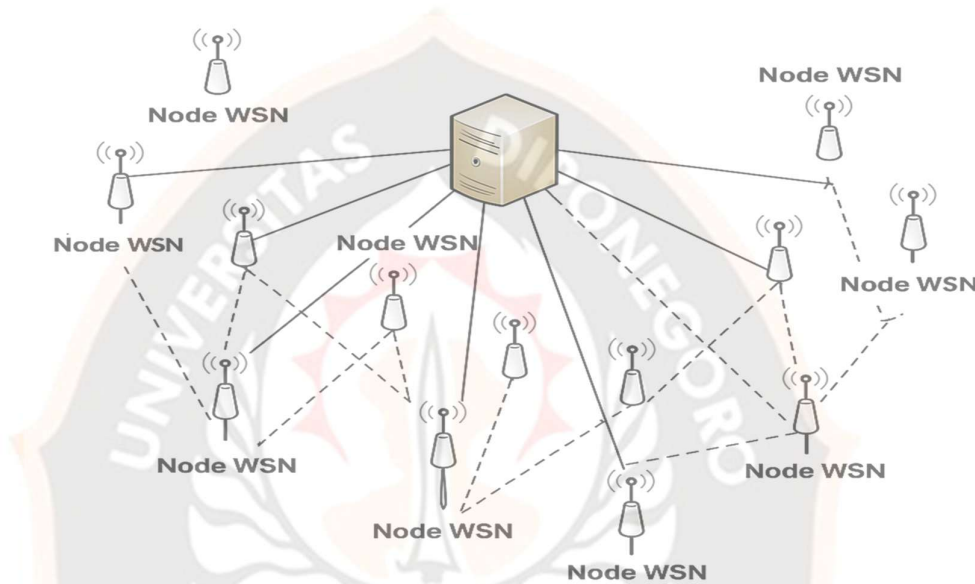


Gambar 2. 12 Alur Algoritma *LEACH-HAC*

Proses alur algoritma *LEACH-HAC*. Proses diawali dengan klasifikasi data suhu dan kelembapan menggunakan algoritma *fuzzy* untuk menentukan tingkat keanggotaan masing-masing variabel. Selanjutnya, dilakukan pembentukan *cluster head* menggunakan metode *LEACH*, yaitu metode hierarki adaptif berbasis *cluster* untuk mengatur komunikasi antara *node sensor* dan *base station*.

Setelah itu, *cluster head* dan anggota *cluster* didesain sedemikian rupa agar berdekatan guna meningkatkan efisiensi komunikasi dalam jaringan. Tahap berikutnya mencakup proses pengelompokan dengan mencari dua data yang paling berdekatan menggunakan pendekatan metode *Hierarchical Agglomerative Clustering (HAC)*, yang merupakan metode *unsupervised learning* dalam *clustering* yang bekerja dengan pendekatan *bottom-up*, yaitu setiap *data node* dianggap sebagai *cluster* tunggal, lalu secara bertahap digabungkan dengan *cluster* terdekat hingga terbentuk satu hierarki pohon menggunakan *single linkage*.

Tahap ini bertujuan untuk mengoptimalkan kinerja sistem irigasi berdasarkan kesesuaian suhu dan kelembapan yang diperlukan oleh tanah dan tanaman. Sementara itu, *single channel* diperlihatkan pada Gambar 2.13.



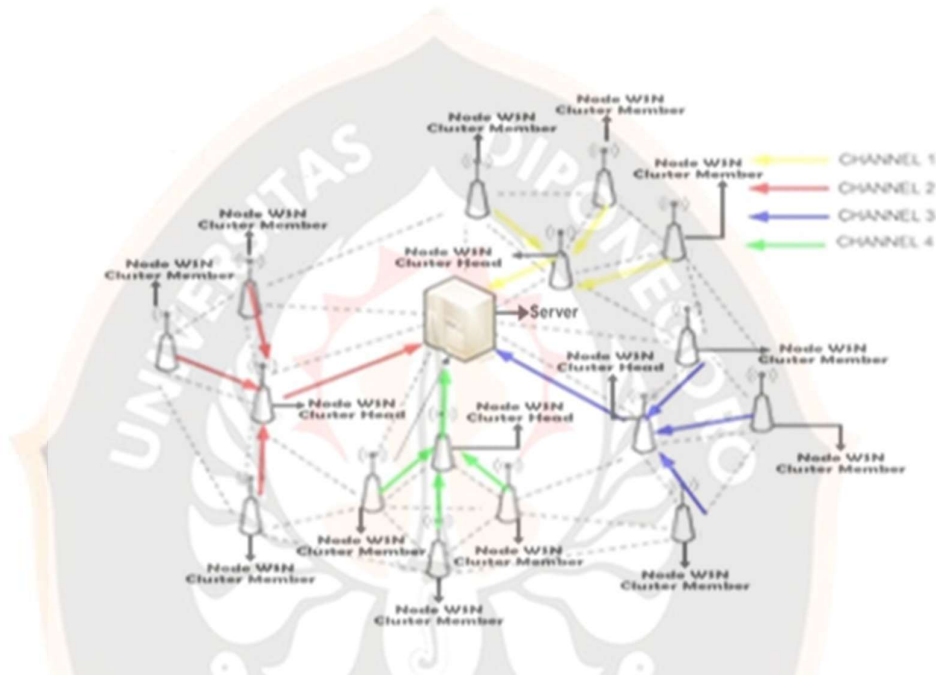
Gambar 2. 13 Single Channel

Model protokol *WSN* pada umumnya. Dalam model ini, setiap *node* mengirimkan data secara langsung ke *base station* atau *server*. Hal tersebut menyebabkan jumlah transaksi menjadi sangat banyak, sehingga memberikan beban besar pada *server*.

Seiring dengan bertambahnya jumlah *node WSN* yang melakukan transaksi, lalu lintas data dalam jaringan pun meningkat. Akibatnya, performa *WSN* dapat menurun karena tingginya beban komunikasi dan penggunaan energi yang tidak efisien.

Gambar 2.13 menggambarkan arsitektur dasar protokol komunikasi berbasis *single channel* yang masih memiliki keterbatasan dalam hal efisiensi dan *throughput* jaringan. Sementara itu, Gambar 2.14 menunjukkan pengembangan lebih lanjut dari model tersebut menjadi arsitektur *multi-channel*, yang dirancang untuk mengatasi permasalahan interferensi dan kolisi transmisi pada protokol sebelumnya.

Perubahan ini mencerminkan evolusi sistem komunikasi dalam jaringan sensor nirkabel yang lebih adaptif dan efisien, terutama dalam skenario aplikasi seperti sistem irigasi cerdas.



Gambar 2. 14 Model Protokol *LEACH-HAC*

Model protokol *LEACH-HAC* menggunakan empat *channel* komunikasi untuk pengelompokan *node* suhu dan kelembapan dalam jaringan *WSN*. Model ini mengusulkan mekanisme komunikasi berbasis *LEACH-HAC*, yang terdiri atas empat *channel* dengan peran spesifik dalam transmisi data. *Cluster Member* berfungsi sebagai simpul sensor yang mengumpulkan data suhu dan kelembapan, kemudian mengirimkannya melalui *channel* komunikasi berwarna merah, hijau, biru, dan kuning, yang merepresentasikan kategori data suhu dan kelembapan.

Sementara itu, *cluster head* berperan sebagai agregator data, menerima informasi dari masing-masing *cluster member* dan melakukan transmisi terintegrasi menuju *base station* atau *server*. Dengan pendekatan ini, efisiensi komunikasi dalam *WSN* dapat ditingkatkan melalui optimalisasi rute pengiriman data serta pengurangan konsumsi energi dalam jaringan.

Server berfungsi untuk menyimpan, mengolah, dan mendistribusikan data sehingga sistem irigasi cerdas dapat beroperasi secara otomatis. *Cluster head*

menampung data dari *cluster member* dan mengirimkannya ke *base station/server*. Selanjutnya, *server* mengolah data dengan algoritma *LEACH-HAC* dan mendistribusikannya kembali ke setiap *cluster head* agar proses pengairan pada sistem irigasi dapat berjalan secara otomatis.

Model protokol *LEACH-HAC* dapat menyelesaikan permasalahan ketika terlalu banyak *node* mengirimkan data secara bersamaan, atau ketika volume data besar ditransmisikan melalui satu saluran tunggal, yang dapat menyebabkan *delay*. Jika beban data berlebih, hal ini juga berpotensi menimbulkan *packet loss* dan *jitter* akibat kemacetan jaringan. Permasalahan-permasalahan tersebut mengakibatkan rendahnya nilai *throughput*, sehingga menurunkan performa *WSN* dan berdampak pada tidak optimalnya kinerja sistem irigasi.

Proses kerja algoritma *LEACH-HAC* dirancang untuk mengatasi rendahnya performa *WSN*, khususnya nilai *throughput*. Langkah-langkahnya adalah sebagai berikut:

1. Penentuan kriteria data suhu dan kelembapan:
 - Suhu diklasifikasikan ke dalam kategori: sangat dingin, dingin, cukup dingin, sejuk, cukup panas, panas, dan sangat panas.
 - Kelembapan dikategorikan menjadi: kering, cukup kering, sedang, lembap, dan basah.
2. Pembentukan *cluster head* sebagai acuan utama dalam menentukan *channel* yang akan digunakan oleh anggota *node*.
3. Penentuan titik *node* terdekat menggunakan pendekatan *Euclidean Distance* untuk mencari *node* yang paling dekat dengan *Cluster Head*.
4. Proses pengelompokan dengan *HAC*:
 - *Single Linkage* adalah teknik *Hierarchical Clustering* yang menghasilkan pengurangan jumlah kelompok secara bertahap.
 - Pada setiap tahapan, dua *Cluster* dengan kemiripan tertinggi (berdasarkan kedekatan terhadap *Cluster Head*) digabungkan.
5. Proses pengiriman data:
 - *Cluster Head* dan *Cluster Member* yang berdekatan mengirimkan data berupa variabel suhu dan kelembapan ke *Cluster Head*.
 - *Cluster Head* mengirimkan data tersebut ke *Base Station/Server*.

- o Data ini diklasifikasikan untuk menyesuaikan kebutuhan tanah dan tanaman.

Keunggulan algoritma *LEACH-HAC* adalah kemampuannya membentuk *Cluster Head* dan *Cluster Member* yang saling berdekatan, guna meningkatkan efisiensi energi dalam pengiriman data. Pengembangan protokol *WSN* berbasis pendekatan ini terbukti dapat mengatasi permasalahan rendahnya performa, meningkatkan nilai *throughput*, serta mengurangi *packet loss*, *delay*, dan *jitter*. Dengan demikian, sistem irigasi otomatis dapat beroperasi secara lebih optimal dan efisien.

2.3.9 Confusion Matrix

Confusion Matrix adalah alat evaluasi dalam pembelajaran mesin dan statistik yang digunakan untuk mengukur kinerja model klasifikasi. Matriks ini menyajikan gambaran menyeluruh mengenai akurasi klasifikasi model dengan menampilkan jumlah prediksi yang benar dan salah untuk setiap kelas dalam dataset. *Confusion Matrix* berperan penting dalam mengidentifikasi kesalahan yang terjadi serta menentukan area yang memerlukan perbaikan.

Akurasi adalah ukuran seberapa sering model atau sistem klasifikasi memberikan hasil yang benar terhadap seluruh data yang diuji. Dalam konteks *Wireless Sensor Network WSN*, akurasi menunjukkan seberapa efektif sistem mendeteksi kejadian atau kondisi lingkungan yang sebenarnya dengan menggunakan rumus akurasi:

$$\text{Akurasi} = \frac{TP+T}{TP+TN+FP+FN} \quad (2.18)$$

TP = True positive (Prediksi positif yang benar), TN = True negative (Prediksi negatif yang benar), FP = False Positive (Prediksi positif yang salah), FN = False negative (Prediksi negatif yang salah).

Contoh kasus sistem irigasi cerdas diuji pada 100 kali pengukuran dengan hasil sebagai berikut:

Tabel 2. 2 Contoh Akurasi

Kondisi sebenarnya	Prediksi System	Kategori
Kekeringan	Kekeringan	True positive (TP) =40
Tidak kekeringan	Tidak kekeringan	True negative (TN) = 45
Tidak kekeringan	Kekeringan	False negative (FN) = 10
Kekeringan	Tidak kekeringan	False negative (FN) = 10

$$\begin{aligned}
 \text{Akurasi} &= \frac{TP+TN}{TP+TN+FP+FN} \\
 &= \frac{40+45}{40+45+5+1} \\
 &= \frac{8}{100} \\
 &= 0,85 \text{ atau } 85\%
 \end{aligned}$$

Artinya model *WSN* memiliki akurasi sebesar 85% dalam mendeteksi kekeringan atau kelembaban dengan benar.

Presisi adalah ukuran performa model klasifikasi dalam memprediksi hasil positif yang benar dari seluruh prediksi positif yang dihasilkan. Dengan kata lain, presisi mengukur seberapa akurat model dalam memprediksi kelas positif. Contoh sistem irigasi cerdas diuji pada 100 kali pengukuran dengan hasil sebagai berikut:

Tabel 2. 3 Contoh Presisi

Kondisi sebenarnya	Prediksi system	Kategori
Kekeringan	Kekeringan	True positive (TP) =40
Tidak kekeringan	Tidak kekeringan	True negative (TN) = 45
Tidak kekeringan	Kekeringan	False negative (FN) = 5
Kekeringan	Tidak kekeringan	False negative (FN) = 10

$$\begin{aligned}
 \text{Presisi} &= \frac{TP}{TP+F} \\
 &= \frac{40}{40+5} \\
 &= \frac{40}{5} \\
 &= 0,888 \text{ atau } 88,8\%
 \end{aligned}
 \tag{2.19}$$

Dari semua prediksi kekeringan yang di buat sistem, 88,8% benar benar merupakan kekeringan.

Recall atau Sensitivitas adalah ukuran kemampuan model untuk mendeteksi semua kasus positif dengan benar dari seluruh kejadian positif yang ada dalam data. Contoh kasus sistem irigasi cerdas diuji pada 100 kali pengukuran dengan hasil sebagai berikut:

Tabel 2. 4 Contoh recall

Kondisi sebenarnya	Prediksi system	Kategori
Kekeringan	Kekeringan	True positive (TP) =40
Tidak kekeringan	Tidak kekeringan	True negative (TN) = 45
Tidak kekeringan	Kekeringan	False negative (FN) = 5
Kekeringan	Tidak kekeringan	False negative (FN) = 10

$$\begin{aligned}
 \text{Recall} &= \frac{TP}{TP+FP} \\
 &= \frac{40}{40+10} \\
 &= \frac{40}{50} \\
 &= 0,8 \text{ atau } 80\%
 \end{aligned}
 \tag{2.20}$$

Dari semua kejadian kekeringan yang benar benar terjadi, 80% berhasil dideteksi oleh sistem.

F1-Score adalah rata rata harmonis antara Presisi dan Recall. F1-Score digunakan untuk mengukur keseimbangan antara Presisi dan Recall dalam satu *metrik* yang terintegrasi.

Contoh kasus sistem irigasi cerdas diuji pada 100 kali pengukuran dengan hasil sebagai berikut:

Tabel 2. 5 Contoh F1-Score

Kondisi sebenarnya	Prediksi System	Kategori
Kekeringan	Kekeringan	True positive (TP) =40
Tidak kekeringan	Tidak kekeringan	True negative (TN) = 45
Tidak kekeringan	Kekeringan	False negative (FN) = 5
Kekeringan	Tidak kekeringan	False negative (FN) = 10

$$\begin{aligned}
 \text{F1-Score} &= \frac{2 \times (\text{presisi} \times \text{Recall})}{\text{Presisi} + \text{Rec}} & (2.21) \\
 &= \frac{2 \times (0,888 \times 0,8)}{0,888 + 0,8} \\
 &= \frac{2 \times (0,7104)}{1,688} \\
 &= \frac{1,4208}{1,688}
 \end{aligned}$$

F1-Score sebesar 84,1% berarti sistem memiliki keseimbangan yang baik antara kemampuan mendeteksi kekeringan (*Recall*) dan menghindari kesalahan prediksi (*Presisi*).

SEKOLAH PASCASARJANA