

BAB 1 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Sebagai bahan pertimbangan dalam penelitian ini, dicantumkan beberapa hasil penelitian terdahulu oleh beberapa penelitian sebagai berikut.

No	Nama Penulis, Tahun, dan Judul	Persamaan	Perbedaan	
			Penelitian Terdahulu	Rencana Penelitian
1	Gettu, Sonia Juneja 2012." Performance Analysis of SPIN and LEACH Routing Protocol in WSN: Deptt. Of Comp. Sc. & Engg HCE, Sonapat ,India	Melakukan analisis performa dengan menggunakan routing SPIN dan LEACH pada <i>Wireless Sensor Network</i> .	Menggunakan dua simulator jaringan nirkabel versi 2.34. dalam melakukan pengujian.	Mengimplementasikan routing SPIN-BC dan SPIN-RL dengan menggunakan simulator OMNET ++ dalam melakukan pengujian.
2	Kusum Lata, Kusum Dalal.2017." Performance Analysis of LEACH and M-GEAR Routing Protocols for WSN". *M.Tech Scholar, ** Assistant Professor, ECE Dept., DCRUST Murthal.	Melakukan analisis performa routing pada <i>Wireless Sensor Network</i> .	Menjelaskan hasil kinerja yang diimplementasikan terhadap algoritma <i>routing</i> dalam hal <i>throughput</i> dan konsumsi energy.	Menjelaskan hasil kinerja yang diimplementasikan terhadap algoritma <i>routing</i> dalam hal <i>Latency, Number data pakets forwarded, Tx Power, Number Tx.Frames dan Memori Node</i> untuk di analisa hasil performa dari routing tersebut.

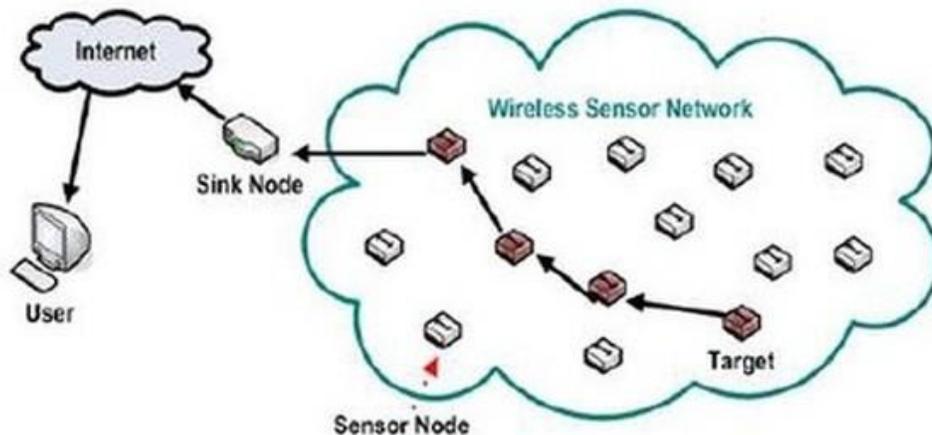
Tabel 1.1 Kajian Pustaka

Pada tabel 2.1 menjelaskan penelitian dari Gettu, Sonia Junej " *Performance Analysis of SPIN and LEACH Routing Protocol in WSN*" menggunakan routing SPIN dan LEACH pada *Wireless Sensor Network* serta menggunakan dua simulator jaringan nirkabel dalam melakukan pengujian. Sedangkan penelitian terakhir dari Kusum Lata, Kusum Dalal "*Performance Analysis of LEACH and M-GEAR Routing Protocols for WSN*" menjelaskan hasil kinerja yang diimplementasikan terhadap algoritma routing dalam hal *throughput* dan konsumsi energi, serta melakukan analisis performa routing yang ada pada *Wireless Sensor Network*.

2.2 Kajian Teori

2.2.1 *Wireless Sensor Network*

Wireless Sensor Network (WSN) atau jaringan sensor nirkabel merupakan suatu jaringan yang terdiri dari beberapa sensor *node* yang bersifat individu yang diletakkan ditempat yang berbeda untuk memonitoring kondisi suatu tempat dan berinteraksi dengan lingkungan dengan cara *controlling* dan *communication* terhadap parameter- parameter fisiknya.



Gambar 1.1 Arsitektur WSN

Sumber : (Giaz.Weebly, 2012)

Pada Gambar 2.1 menunjukkan gambaran umum WSN, dapat dilihat *node* sensor yang berukuran kecil tersebar dalam suatu area sensor. *Node* sensor tersebar dengan tujuan untuk menangkap adanya gejala atau fenomena yang hendak diteliti. Jumlah *node* yang tersebar dapat ditentukan sesuai dengan kebutuhan dan tergantung beberapa faktor seperti kemampuan *sensing node*. Tiap *node* sensor memiliki kemampuan untuk merutekan data yang dikumpulkan ke *node* yang lain yang berdekatan. Data dikirimkan melalui transmisi radio akan diteruskan menuju *Base station* atau *Sink node* yang merupakan penghubung antara *node* sensor dan *user*. Informasi tersebut dapat diakses melalui berbagai *platform* seperti koneksi internet atau satelit sehingga memungkinkan *user* untuk dapat mengakses secara *realtime* melalui *remote server* (E, Sugianto,&Sakti,2009).

Wireless Sensor Network (WSN) secara umum dapat dijelaskan sebagai jaringan *node* yang secara kooperatif merasakan dan memungkinkan untuk mengendalikan lingkungan melalui interaksi antara orang atau komputer dan lingkungan sekitarnya. *Wireless Sensor Network* biasanya terdiri dari sejumlah *node-node* sensor daya rendah yang dipasang dalam area pengawasan. *Node-node* ini dapat merasakan, mencoba dan memproses informasi yang terkumpul dari lingkungan sekitarnya. Sebuah *Wireless Sensor Network* memiliki sedikit ataupun tidak ada infrastruktur. Ini hanya terdiri dari sejumlah *node-node* sensor yang bekerja bersama untuk memonitor sebuah kawasan untuk mendapatkan data tentang suatu lingkungan.

2.2.2 *Protokol Routing*

Menurut C. Siva Ram (2004), tanggung jawab sebuah *routing protocol* adalah melakukan pertukaran informasi rute, menemukan jalur yang mungkin untuk sampai ke *node* tujuan dengan

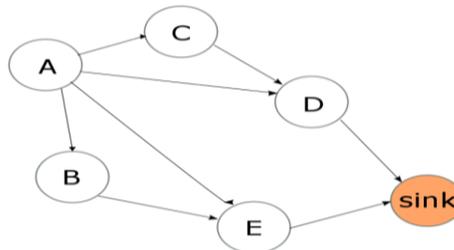
memperhatikan panjang lompatan, energi yang dibutuhkan seminimal mungkin, dan lama jangka hidup dari hubungan nirkabel tersebut, mengumpulkan informasi tentang jalur yang putus, memperbaiki jalur yang rusak, dan memanfaatkan *bandwidth* yang minimum.

Routing merupakan algoritma perpindahan informasi di seluruh jaringan dari *node* sumber ke *node* tujuan dengan minimal satu *node* yang berperan sebagai perantara. Komponen penting pada sebuah protokol *routing*/Algoritma *routing* berfungsi untuk menentukan bagaimana *node* berkomunikasi dengan *node* yang lainnya dan menyebarkan informasi yang memungkinkan *node* sumber untuk memilih rute optimal ke *node* tujuan dalam sebuah jaringan komputer. Sedangkan sebuah algoritma *routing* berfungsi untuk menghitung secara matematis jalur yang optimal berdasarkan informasi *routing* yang dimiliki oleh suatu *node*.

2.2.3 Routing Flooding

Flooding merupakan *routing* protokol yang menggunakan *random* paket *broadcast* didalam sebuah jaringan, yang menyebabkan komunikasi dalam sebuah jaringan tersebut kacau. Oleh karena itu *node* mengkonsumsi daya lebih untuk mengirim data ketujuan dan akibatnya *throughput* jaringan berkurang drastis. Jaringan flooding merupakan komunikasi mendasar primitif untuk Wireless Sensor Networks (WSN). Flooding digunakan untuk menyebarkan *update* kode dan perubahan parameter, yang mempengaruhi operasi dari semua *node* dalam jaringan.

Pengiriman paket dilakukan secara *broadcast* dari *source node* menuju *node-node* tetangga. *Node* yang menerima paket, akan melanjutkan *broadcast* ke tetangganya sampai paket diterima oleh *sink-node*. Kelebihan yang dimiliki oleh flooding *routing* adalah ketepatan paket untuk sampai pada *sink-node* tidak diragukan karena metode *broadcast* memastikan semua *node* pada jaringan menerima paket terkait. Sedangkan Kelemahannya adalah lalu lintas data akan padat dikarenakan metode *broadcast* yang digunakan serta kemungkinan terjadi tabrakan data cukup besar.



Gambar 1.2 Flooding

Sumber: (Pavan Bhat, 2011)

Pada topologi diatas, diasumsikan bahwa *node* A akan mengirim paket ke *sink*. Jika menggunakan flooding *routing*, maka *node* A akan melakukan *broadcast* dan mengirim paket melalui *node* manapun hingga data sampai ke *sink-node*. Flooding mengirim paket yang besar karena setiap *node* akan dikirimkan data yang sama dan meneruskan ke semua *node* yang ada tetapi flooding tidak memerlukan informasi apapun.

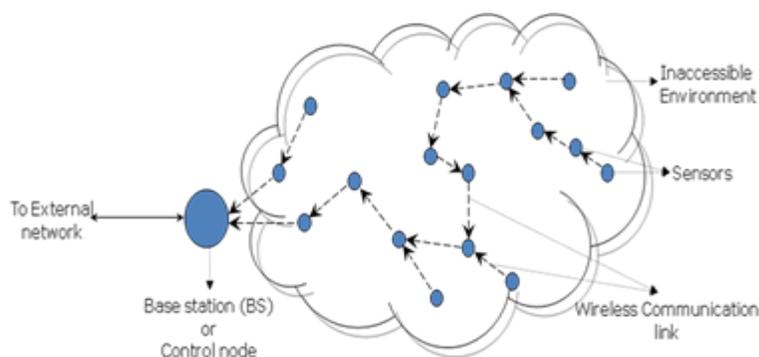
Dalam protokol flooding, pesan yang dikirim adalah perubahan dari basis data serta nomor urut pesan tersebut. Dengan hanya mengirimkan perubahan basis data, waktu yang diperlukan untuk pengiriman dan pemrosesan pesan tersebut lebih sedikit dibandingkan dengan mengirim seluruh isi basis data tersebut. Nomor urut pesan diperlukan untuk mengetahui apakah pesan yang diterima lebih baru daripada yang terdapat dalam basis data. Nomor urut ini berguna

pada kasus *link* yang putus menjadi tersambung kembali. Pada saat terdapat *link* putus dan jaringan menjadi terpisah, basis data kedua bagian jaringan tersebut menjadi berbeda. Ketika *link* yang putus tersebut hidup kembali, basis data di semua *router* harus disamakan. Basis data ini tidak akan kembali sama dengan mengirimkan satu pesan *link-state* saja.

2.2.4 Routing Convergecast

Convergecast adalah mekanisme pengumpulan data dari satu set sensor menuju ke sebuah *sink* atau *base station* umum berdasarkan topologi *tree*. Proses pengumpulan data dari semua atau satu set sensor di jaringan menuju *base station* (Banyak ke satu komunikasi).

Convergecast banyak digunakan untuk pengumpulan data. Biasanya pengumpulan data tersebut digunakan pada aplikasi jaringan sensor. Sebagai contohnya adalah untuk menentukan lokasi pada lingkungan perkotaan. Penentuan lokasi yang tidak diimplementasikan dengan *routing* Convergecast tidak bisa dieksekusi karena *node* sensor dibatasi pada kemampuan komputasinya (Dalian,2012)



Gambar 1.3 Convergecast

Sumber: (Valiappan, 2012)

Fokus utama dari convergecast adalah *energy* dan *latency*, energi meliputi batasan alami dari sensor. Dan *latency* meliputi kebutuhan dari seperangkat aplikasi monitoring yang digunakan. Konstruksi rute memainkan peran utama selama convergecasting.

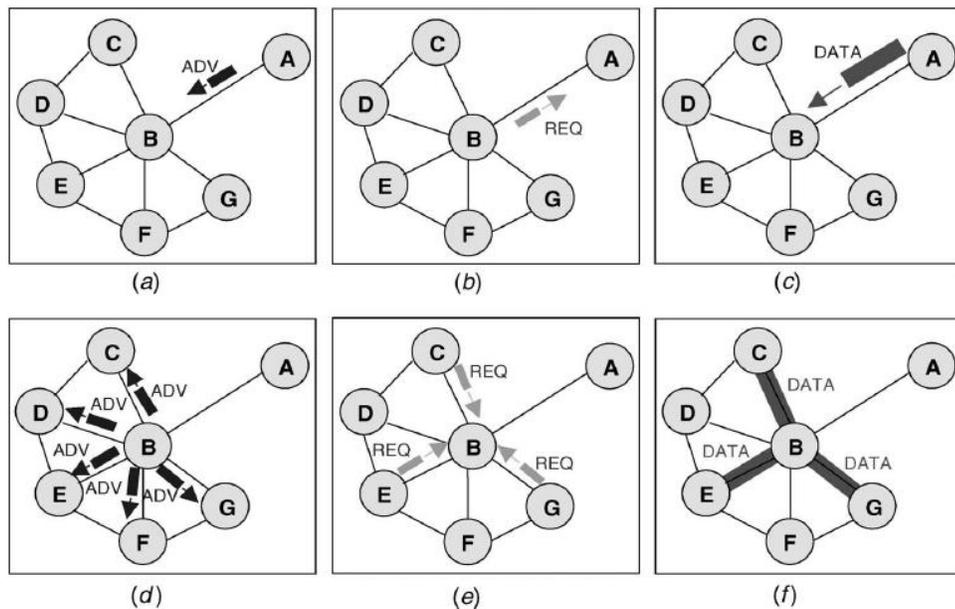
2.2.5 Routing SPIN

Sensor Protocol for Information via Negotiation (SPIN) adalah negosiasi berbasis data-centric yang termasuk kedalam jenis informasi protokol diseminasi untuk WSN. Tujuan utama dari protokol ini adalah untuk menyebarkan pengamatan yang dikumpulkan oleh node sensor individu untuk semua node sensor dalam jaringan secara efisien. Protokol sederhana seperti flooding dan gossiping biasanya diajukan untuk mencapai penyebaran informasi di WSN. Flooding mengharuskan setiap node mengirimkan salinan dari paket data ke semua server sampai informasi mencapai semua node dalam jaringan. Di sisi lain, gossiping menggunakan pengacakan untuk mengurangi jumlah paket duplikat dan mengharuskan hanya node yang akan diteruskan ke server lain yang dipilih secara acak. Flooding dan gossiping yang sederhana adalah menarik, karena kedua protokol tersebut menggunakan aturan forwarding sederhana dan tidak memerlukan perawatan topologi. Namun, kinerja algoritma dalam hal probabilitas penundaan paket dan pemanfaatan sumber daya cenderung memburuk dengan ukuran jaringan dan beban

lalu lintas. Kelemahan kinerja ini biasanya disebabkan oleh ledakan lalu lintas dan geografis yang tumpang tindih. Hasil implosi lalu lintas tersebut di beberapa salinan data yang sama dikirimkan ke node sensor yang sama. Di sisi lain, tumpang tindih geografis menyebabkan node menutupi wilayah geografis yang sama untuk menyebarkan informasi data yang sama ke node sensor jaringan. Protokol sederhana seperti flooding tidak mengubah perilaku mereka untuk beradaptasi pada komunikasi dan komputasi pada status terkini sumber energi mereka. Kurangnya kesadaran akan sumber daya dan adaptasi dapat mengurangi masa pakai jaringan yang menyebabkan node yang sangat aktif kemungkinan besar menguras sumber energi mereka. Tujuan utama dari SPIN dan jenis sejenisnya yang terkait adalah untuk mengatasi kekurangan dari penyebaran protokol informasi konvensional dan mengatasi kekurangan kinerja mereka. Prinsip dasar jenis protokol ini adalah negosiasi data dan adaptasi sumber daya. (J. Kulik, W.R. Heinzelman, H. Balakrishnan., 2002)

Negosiasi data berbasis semantik mengharuskan node-node yang menjalankan SPIN untuk "mempelajari" isi dari data sebelum data-data tersebut ditransmisikan antara node-node jaringan. SPIN memanfaatkan sistem penamaan data dimana node akan mengasosiasikan metadata dengan data yang dihasilkan dan menggunakan data-data deskriptif ini untuk melakukan negosiasi sebelum transmisi dilakukan pada data yang aktual. Sebuah receiver (penerima) yang menunjukkan ketertarikan pada isi dari data-data tersebut dapat mengirimkan permintaan untuk memperoleh data yang telah diiklankan. Bentuk negosiasi ini menjamin bahwa data dikirimkan dengan tujuan hanya untuk menarik node, sehingga implosi lalu lintas dapat dihilangkan dan transmisi data yang berlebihan diseluruh jaringan dapat dikurangi secara signifikan. Selain itu, pengguna deskriptor metadata dapat menghilangkan kemungkinan terjadinya overlap (tumpang tindih) karena node dapat membatasi permintaan penamaan yang dilakukan hanya untuk data yang ingin diperoleh. Adaptasi sumber daya memungkinkan sensor node menjalankan SPIN untuk menyesuaikan kegiatan mereka dengan kondisi terbaru dari sumber daya energi. Masing-masing node dalam jaringan dapat menjajaki manajer sumber daya yang terkait untuk memantau konsumsi sumber daya sebelum transmisi atau data proses dilakukan. Saat kondisi terkini dari energi menjadi rendah, pengurangan atau penghilangan sepenuhnya aktivitas tertentu dapat dilakukan oleh node, contohnya meneruskan third-party metadata dan paket data. Fitur adaptasi sumber daya dari SPIN memungkinkan node untuk memperpanjang masa jaringan. Untuk menjalankan negosiasi dan transmisi data, node akan menjalankan SPIN menggunakan 3 tipe jenis perintah. Perintah jenis pertama adalah ADV, yang biasanya digunakan untuk mengiklankan data baru antara node-node. Sebuah node jaringan yang memiliki data yang bisa dibagikan dengan node yang ada di sebuah jaringan dapat mengiklankan datanya sendiri dengan terlebih dahulu mengirimkan perintah ADV yang berisi metada yang menguraikan data-data tersebut. Jenis perintah kedua adalah REQ, yang biasanya digunakan untuk meminta data menarik yang telah diiklankan. Setelah menerima perintah ADV yang mengandung metadata, sebuah node jaringan biasanya tertarik untuk menerima data spesifik yang dikirimkan perintah REQ node iklan metadata, yang kemudian akan mengirimkan data yang diminta. Jenis perintah ketiga adalah DATA, yang berisi data aktual yang dikumpulkan oleh sensor bersama dengan header metadata. Perintah DATA biasanya lebih besar dari perintah ADV dan REQ. Perintah terakhir hanya berisi metada yang cenderung jauh lebih kecil daripada perintah data yang berkaitan. Membatasi transmisi berlebihan perintah data menggunakan

negosiasi berbasis semantik dapat menghasilkan penurunan dari konsumsi energi secara signifikan.



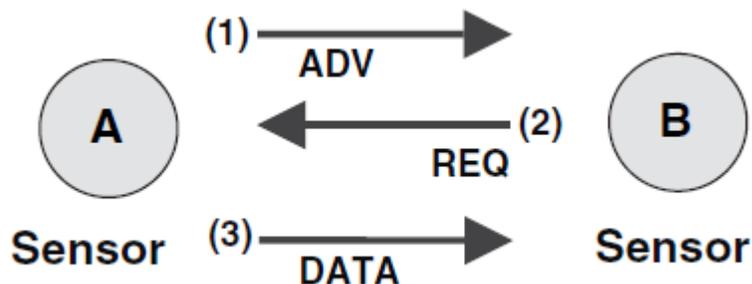
Gambar 1.4 SPIN DASAR

Sumber: (Sohraby, Minoli, & Znati, 2007)

Perilaku dasar dari SPIN digambarkan pada gambar 2.2, yang mana sumber datanya, sensor node A, menampilkan datanya pada server lainnya, sensor node B, dengan mengirimkan sebuah perintah ADV yang berisi metadata yang menggambarkan datanya. Node B mengungkapkan ketertarikan pada data yang telah ditampilkan dan mengirimkan sebuah perintah REQ untuk memperoleh data. Setelah menerima data, node B mengirimkan sebuah perintah ADV untuk menampilkan data yang baru saja diterima selver lainnya. Hanya 3 server inilah, node C,E, dan G inilah, yang memperlihatkan ketertarikan pada data. Node-node ini, mengeluarkan sebuah perintah REQ ke node B, yang akhirnya mengantarkan data kepada setiap node yang meminta. SPIN itu sendiri memiliki 4 macam sebagai berikut : (Sohraby et al., 2007).

2.2.5.1 Routing SPIN-PP

Versi yang paling sederhana dari SPIN, disebut sebagai SPIN-PP, di rancang untuk jaringan komunikasi dari poin ke poin. Protokol three-steps handshake yang digunakan oleh SPIN-PP digambarkan pada gambar dibawah ini :



Gambar 1.5 SPIN-PP

Sumber: (Sohraby et al., 2007)

Pada gambar 2.5. Pada langkah 1, node menahan data, node A mengeluarkan sebuah paket pemberitahuan (ADV). Pada langkah 2, node B memperlihatkan ketertarikan dalam menerima data, dengan mengeluarkan sebuah permintaan data (REQ). Pada langkah 3, node A menanggapi permintaan tersebut dan mengirimkan paket data kepada node B. Langkah ini melengkapi prosedur three-steps handshake. SPIN-PP menggunakan negosiasi untuk menanggulangi permasalahan ledakan dan tumpang tindih dari protokol traditional flooding dan gossiping. Sebuah penelitian kinerja berbasis simulasi pada SPIN-1 memperlihatkan bahwa protokolnya mengurangi pemakaian energi dengan faktor dari 3,5 dibandingkan dengan flooding. Protokolnya juga mencapai tingkat penyebaran data yang tinggi, mendekati optimum teoritis (Sohraby et al., 2007).

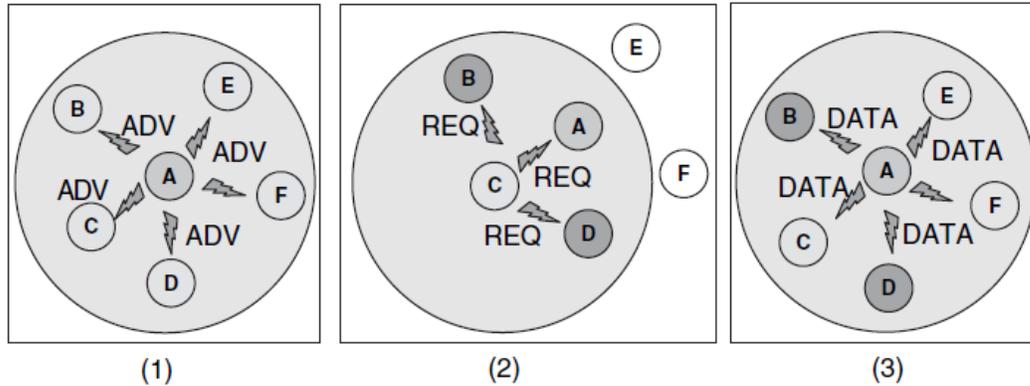
2.2.5.2 Routing SPIN-EC

SPIN-EC, menggabungkan mekanisme kesadaran akan sumber daya berbasis threshold untuk melengkapi negosiasi data. Ketika tingkat energinya mendekati threshold rendah, node yang menjalankan SPIN-EC mengurangi keterlibatannya dalam pengoperasian protokol. Secara khusus, sebuah node terlibat hanya jika diputuskan bahwa node tersebut dapat melengkapi semua tingkatan dari protokol pengoperasian tanpa menyebabkan tingkat energinya menurun dibawah threshold. Oleh karena itu, jika sebuah node menerima siaran (advertisement), node tersebut tidak akan mengirim perintah REQ jika ditentukan bahwa sumber tenaganya tidak cukup tinggi untuk mengirimkan sebuah perintah REQ dan menerima perintah DATA yang sesuai. Berbagai hasil simulasi dari protokol ini memperlihatkan bahwa SPIN-EC menyebarkan 60% lebih data per unit energi dari flooding.

Selain itu, data menunjukkan bahwa SPIN-EC sangat mendekati jumlah ideal data yang dapat disebar luaskan per unit energi. Baik SPIN-PP maupun SPIN-EC di desain sebagai komunikasi dari poin ke poin.

2.2.5.3 Routing SPIN-BC

SPIN-BC, di rancang sebagai jaringan penyiaran. Pada ketiga network ini, node membagi satu salurannya untuk komunikasi. Di kelas jaringan ini, ketika node mengirimkan paket data pada saluran penyiaran, paket yang dikirimkan akan diterima oleh semua node lain dalam jarak tertentu dari node pengirim. Protokol SPIN-BC menerima keuntungan dari kemampuan saluran penyiaran dan mengharuskan node yang telah menerima perintah ADV untuk tidak segera merespon dengan perintah REQ. Sebaliknya, node akan menunggu dalam waktu tertentu, selama kemampuan saluran penyiaran memonitor saluran komunikasi. Jika perintah nodehears REQ dikeluarkan oleh node lain yang tertarik untuk menerima data, permintaan tersebut akan ditolak secara otomatis, sehingga menghilangkan permintaan yang sama. Selanjutnya, setelah menerima perintah REQ, node iklan hanya akan mengirimkan pesan data sekali, bahkan saat menerima beberapa permintaan untuk pesan yang sama.



Gambar 1.4 SPIN-BC DASAR

Sumber: (Sohraby et al., 2007)

Pada gambar 2.4, dalam konfigurasi ini, node A yang menahan data mengirimkan sebuah paket ADV untuk menampilkan data ke server lainnya. Seluruh node akan mendengar iklan tersebut, tapi node C yang pertama mengeluarkan paket REQ untuk meminta data dari node A. Node B dan D membaca permintaan penyiaran dan tidak akan mengeluarkan paket REQ sendiri. Node E dan F tidak tertarik pada data yang ditampilkan, sehingga mereka sengaja menunda permintaan. Setelah membaca permintaan node C, node A akan membalas dengan mengirimkan paket data. Semua node dalam jangkauan transmisi dari A menerima paket data, termasuk node E dan F. Dalam jangkauan penyiaran, SPIN-BC mempunyai potensi untuk mengurangi konsumsi energi dengan meleminasi pertukaran data permintaan dan balasan yang berlebihan (Sohraby et al., 2007).

2.2.5.4 Routing SPIN-RL

Jenis protocol terakhir dari SPIN adalah SPIN-RL, yang memperluas kemampuan dari SPIN-BC untuk menambahkan kehandalan dan menanggulangi kesalahan transmisi perintah yang disebabkan oleh saluran lossy. Peningkatan kehandalan diperoleh dengan penyiaran ADV dan REQ secara berkala. Setiap node di dalam SPIN-BC melacak iklan yang tertangkap jaringan dan node dimana iklan tersebut berasal. Jika sebuah node meminta suatu spesifik data yang tidak diterima oleh data yang telah diminta dalam periode waktu tertentu, node tersebut akan mengirim permintaannya lagi. Selanjutnya, peningkatan kehandalan dapat dibuktikan dengan mengiklankan kembali metadata secara berkala. Akhirnya, jaringan SPIN-RL membatasi frekuensi saat mereka mengirim ulang pesan data. Setelah mengirimkan pesan data, sebuah node akan menunggu dalam selang waktu tertentu sebelum merespon permintaan lain untuk pesan data yang sama. Jenis protocol SPIN menunjukkan kelemahan utama pada flooding. Hasil akhir simulasi menunjukkan bahwa SPIN lebih efisien dalam hal energi daripada flooding. Selain itu, hasil akhirnya juga menunjukkan bahwa nilai SPIN dalam menyebar luaskan data lebih besar dari pada atau sama dengan tingkat protokol lainnya. SPIN mencapai kemajuan ini dengan cara melokalisasi perubahan topologi dan menghilangkan penyebaran informasi yang berlebih melalui negosiasi semantik. Perlu dicatat bahwa, pelokalisasian negosiasi mungkin tidak cukup untuk menutup seluruh jaringan dan memastikan bahwa seluruh node yang dibutuhkan mencapai iklan data dan data yang dibutuhkan. Situasi seperti ini terjadi jika perantara node tidak mengekspresikan ketertarikan pada data dan melepas perintah ADV kepada penerimanya.

Kekurangan ini memungkinkan pencegahan dari penggunaan SPIN pada aplikasi spesifik seperti memantau deteksi intrusi dan perlindungan infrastruktur yang penting (Sohraby et al., 2007).

2.2.6 Omnet++

Omnet++ atau omnetpp adalah *network simulation software discrete-event yang bersifat open source* (sumber *code* yang terbuka) *Discreate-event* berarti simulasinya bertindak atas kejadian langsung didalam event. Secara analitis jaringan komputer adalah rangkaian *discrete-event*. Komputer akan membuat sesi simulasi, sesi pengiriman, dan sesi penutup.

OMNET++ bersifat *object-oriented* berarti setiap peristiwa yang terjadi di dalam simulator ini berhubungan dengan objek-objek tertentu. OMNET++ juga menyediakan infrastruktur dan *tools* untuk memprogram simulasi sendiri. Pemrograman OMNET++ bersifat *object-oriented* dan bersifat hierarki. Objek-objek yang paling kecil disebut *simple module*, akan memutuskan algoritma yang akan digunakan dalam simulasi tersebut. OMNET++ menyediakan arsitektur komponen untuk permodelan simulasi. Komponen (modul) menggunakan bahasa pemrograman C++ yang berekstensi “.h” dan “.cc”. OMNET++ memiliki dukungan GUI (*Graphical User Interface*) yang luas, karena arsitektur yang modular, simulasi karnel yang dapat di *compile* dengan mudah di sistem.

Omnet juga mendukung beberapa *framework* yaitu: Inet, Inetmanet, Mixim, Castalia, dan lain-lain. *Framework* tersebut yang akan membantu *user* untuk mampu mengembangkan sebuah simulasi jaringan. Pada skripsi ini *Framework* yang digunakan adalah Mixim untuk protokol *routing* SPIN-BC dan Mixim untuk protokol *routing* Spin-RL (Varga,2014)

Karena bersifat *open source* maka OMNET++ mendukung *multi platform* OS seperti: Windows, Linux dan MAC. Adapun beberapa komponen dari OMNET++ adalah:

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

Simulator	Commercial /Open Source	Educational Support	Simulation Type	Protokol Supported
OPNET	Commercial	Yes	DES, Object Oriented	ATM, MANET, FDDI, WiFi, TCP, Wireless etc
GloMoSim	Open Source	Yes	Library Based Parallel	Wireless Network

QualNet	Commercial	No	Distributed and Parallel	Wired, Wireless, WLAN etc
OmNet++	Open Source	Yes	DES Modular Component Based	Wireless Network
NS-3	Open Source	YES	DES, Object Driven	Multicast routing, TCP, MANET, Wireless etc

Tabel 1.2 Daftar Simulator Jaringan dan Fungsinya
Sumber (Illahi, 2011)

2.2.7 MIXIM

MIXIM merupakan *libraries* pada omnet++ yang digunakan untuk membuat simulasi tentang *wireless sensor network*. Dalam *library* ini mencakup beberapa metode *routing* Spin-BC dan Spin-RL. Kemudian terdapat beberapa rincian yang ditawarkan pada MIXIM, misalnya perambatan gelombang radio, gangguan estimasi, radio *transceiver*, *power consumption*, *wireless MAC protocol* (Mochbatati, 2014)

MIXIM memiliki beberapa *protocol library*. Diantaranya adalah :

- a. *MAC protocol*
MAC *protocol* dirancang untuk membuat keputusan tentang berbagi media dalam komunikasi antara *node* dari sebuah sistem. Keputusan tersebut berupa kapan *node* harus mengirimkan pesan agar tidak mengganggu pengiriman pesan dari *node* lain.
- b. *Network layer protocol*
Pada *protocol library* ini bertugas untuk membuat keputusan mengenai tentang kapan untuk memperbarui rute untuk disinkronisasi.
- c. *Mobility models*
Protocol library ini telah ditambahkan beberapa modul untuk melengkapi kinerjanya. Modul tersebut adalah *Stationary mobility*.
- d. *Localization*
Merupakan *protocol library* yang dirancang untuk memberikan suatu layanan oleh lokalisasi lapisan opsional. Opsional tidak termasuk dalam protokol perangkat lunak yang standar, tetapi harus ditambahkan ke *node* modul eksplisit.

2.2.8 Parameter Simulasi

Parameter Simulasi digunakan untuk menentukan kinerja protokol *routing* pada saat pengujian, parameter-parameter simulasi ditentukan terlebih dahulu. Parameter-parameter tersebut adalah atribut-atribut tetap untuk skenario pengujian seperti yang terlihat pada Tabel 2.3.

Parameter Simulasi	
Protokol	SPIN-BC dan Spin-RL
Banyaknya <i>Node</i> SPIN-BC dan SPIN-RL	4,8,12,16,dan 20
Type <i>Node</i>	Statik
Simulasi Area	1000*1000m
Simulasi Waktu	30 s
Parameter yang digunakan	<i>Latency, Number data pakets forwarded,Tx Power,Number Tx.Frames dan Memori Node.</i>

Tabel 1.3 Parameter Simulasi

1. *Latency* merupakan waktu yang dibutuhkan dari awal pengiriman data hingga sampai ke server. Jenis *latency* ada dua, yaitu *Network Low Latency* dan *Network High Latency*. *Network Low Latency* adalah *Latency* yang terjadi pada jaringan sangat kecil. Sedangkan *Network High Latency* adalah *Latency* yang terjadi pada jaringan sangat besar. Terdapat beberapa penyebab terjadinya *Latency*, yaitu media transmisi yang bermasalah, perangkat lunak yang digunakan pengguna terdapat malfungsi, *router* dan *switch* terdapat kesalahan karena port membutuhkan waktu untuk mengubah header pada paket.
2. Number data packets forwarded merupakan jumlah data paket yang diteruskan oleh pesan REQ. pada routing SPIN tidak semua node menerima data yang di siarkan, hanya ada beberapa node yang dapat meneruskan pesan REQ. pada parameter ini akan dapat mengetahui data yang diteruskan oleh pesan REQ .
3. Tx power merupakan daya yang di butuhkan pada waktu pengiriman paket data. Salah satu kelebihan dari routing WSN adalah lebih hemat daya atau energy, pada parameter berikut ini akan membuktikan routing SPIN yang hemat daya atau energy.
4. Number Tx. Frames merupakan frame transfer data paket. Yang merupakan informasi pengiriman data paket dari layer data link, pada parameter ini akan mengetahui seberapa banyak informasi yang dibutuhkan pada saat pengiriman data paket.
5. Memori Node merupakan memori yang digunakan setiap node pada saat mengirim paket data. Setiap simulasi pasti akan membutuhkan memori dalam menyimpan data, pada parameter ini akan meneliti seberapa banyak memory yang digunakan pada saat node mengirim paket data.