

# Algoritmo de Detección y Notificación de Apuntamiento Entre Nodos de una Red Inalambrica Móvil de Sensores

Carlos René Suárez Suárez

Pontificia Universidad Javeriana  
Facultad de Ingeniería, Departamento de Ingeniería Electrónica  
Bogotá, Colombia  
2014

# Índice

<b>1. Introducción</b>	<b>6</b>
<b>2. Estudio de la localización de nodos en redes de sensores inalámbricas</b>	<b>8</b>
2.1. Modelo de canal . . . . .	8
2.2. Control de topología . . . . .	9
2.3. Métodos de estimación de distancia . . . . .	11
2.4. Métodos de cálculo de posición del nodo . . . . .	12
2.5. Algoritmo de localización . . . . .	15
2.6. Otras características . . . . .	16
2.6.1. Recolección de datos . . . . .	16
2.6.2. Tecnologías . . . . .	16
2.6.3. Conectividad y cobertura . . . . .	17
2.6.4. Protocolos . . . . .	17
2.6.5. Tiempo de vida de la red . . . . .	17
2.6.6. Seguridad . . . . .	18
2.6.7. Aplicaciones . . . . .	18
2.6.8. Tablas de enrutamiento . . . . .	18
2.7. Metodología de software . . . . .	18
<b>3. Especificaciones</b>	<b>20</b>
3.1. Entradas . . . . .	21
3.2. Salidas . . . . .	21
3.3. Descripción general y diagrama en bloques . . . . .	22
3.3.1. Diseño de algoritmos del sistema . . . . .	22
3.3.2. Pruebas de Software . . . . .	23
3.3.3. Simulación . . . . .	23
3.3.4. Pruebas de desempeño . . . . .	23
3.3.5. Análisis y conclusiones . . . . .	23
<b>4. Desarrollo</b>	<b>24</b>
4.1. Diseño de algoritmos . . . . .	24
4.1.1. Algoritmo de control de topología . . . . .	25
4.1.2. Mensajes de control de topología . . . . .	29
4.1.3. Algoritmo de enrutamiento . . . . .	29
4.1.4. Algoritmo de localización . . . . .	30
4.1.5. Algoritmo de Apuntamiento . . . . .	33
4.2. Pruebas de Software . . . . .	33
4.3. Simulación . . . . .	34
4.4. Configuración . . . . .	37
4.4.1. Escenarios . . . . .	38
4.4.2. Implementación . . . . .	38

4.5.	Pruebas de calidad y desempeño del sistema . . . . .	39
4.5.1.	Verificación y validación del modelo de simulación . . . . .	39
4.5.2.	Robustez . . . . .	40
4.5.3.	Tiempo de Convergencia . . . . .	40
4.5.4.	Precisión y Porcentaje de error . . . . .	41
4.5.5.	Tiempo de Vida de la red . . . . .	41
4.5.6.	Conectividad y cobertura . . . . .	41
4.6.	Recolección de información . . . . .	42
<b>5.</b>	<b>Análisis de resultados</b>	<b>43</b>
5.1.	Sensibilidad promedio . . . . .	43
5.2.	Tiempo de convergencia . . . . .	45
5.3.	Error de localización . . . . .	47
5.4.	Efectividad de apuntamiento . . . . .	48
<b>6.</b>	<b>Conclusiones</b>	<b>51</b>

# Lista de Tablas

- 1. Tabla de perdidas en entorno . . . . . 9
- 2. Métodos de Medición . . . . . 13
- 3. Ventajas de las plataformas . . . . . 34
- 4. Desventajas de las plataformas . . . . . 34
- 5. Parametros de configuración Inicial . . . . . 37
- 6. Matriz de confusión de apuntamiento . . . . . 48
- 7. Matriz elaborada de apuntamiento . . . . . 49

## Índice de figuras

1.	Clasificación de las técnicas de control de topología . . . . .	10
2.	Diagrama de flujo del proyecto . . . . .	22
3.	Diagrama de procesos de diseño de algoritmos . . . . .	24
4.	Algoritmo A3 . . . . .	27
5.	Diagrama de flujo del control de topología . . . . .	28
6.	Diagrama de flujo de calculo de posición . . . . .	31
7.	Diagrama de flujo de calculo de dirección . . . . .	32
8.	Escenario de apuntamiento . . . . .	33
9.	Escenario de configuración de inicio . . . . .	36
10.	Escenario de simulación gráfica . . . . .	38
11.	Escenario típico . . . . .	39
12.	Salida en el dominio del tiempo . . . . .	42
13.	Proceso de construcción del arbol . . . . .	43
14.	Sensibilidad . . . . .	44
15.	Desviación estándar entre A3 y APA . . . . .	45
16.	Comparación entre A3 y APA . . . . .	45
17.	Tiempo de convergencia típico . . . . .	46
18.	Error en distancia . . . . .	48
19.	Exactitud . . . . .	50
20.	Ejemplo de herencia en MiXiM . . . . .	55

# 1. Introducción

Las redes de sensores inalámbricas móviles (WSN) son sistemas flexibles, ambientalmente sostenibles y de rápida instalación que se componen de un grupo de sensores que recogen información de cualquier actividad en un punto geográfico, los sensores se instalan en un dispositivo que se denomina nodo, sin embargo para que la WSN sea funcional se utilizan diferentes algoritmos de conexión y de recolección de la información transmitida. La información en los nodos son datos que se almacenan se procesan y se transmiten en forma de mensajes a través de medios inalámbricos, lo cual genera que las WSN presenten pocas limitaciones con respecto al sitio donde se implementan.

Las WSN presentan actualmente una amplia variedad de implementaciones que son utilizadas como aplicaciones en las que un nodo sensor o varios sensores puede ser capaces de medir el estado de un entorno particular. Las aplicaciones van desde procesos sencillos como monitoreo, vigilancia y exploración hasta el control de la aplicación que se desarrolla. En la implementación de las WSN se utilizan un diferentes técnicas algoritmos y protocolos, que han sido diseñados para el correcto funcionamiento de los elementos del sistema que se deben tener controlados. Una de las técnicas más estudiadas es el control de topología, que junto con los algoritmos de control de topología se utilizan para ahorrar energía, para extender la vida útil de la red y para un óptimo envío de mensajes de información entre los nodos mediante un algoritmo de enrutamiento. Una de las técnicas más estudiadas es la topología de sensores, que es la conformación lógica de una estructura que forma una red troncal de conexiones inalámbricas, que permite la comunicación entre cualquier par de nodos en la red. Mediante los algoritmos de control de topología se busca tener un mejor rendimiento en términos de sobrecarga de mensajes y otros indicadores seleccionados, como lograr una mejor conectividad en el mantenimiento de topología y proporcionar una mejor cobertura y conectividad.

Las técnicas de la estimación de la distancia entre los nodos de la red es una función primordial para el funcionamiento de las WSN, con la estimación de esta variable las WSN pueden beneficiarse de esta información antes del despliegue de la red. Existen diferentes métodos que permiten calcular la distancia entre nodos, sin embargo cada uno de estos métodos presenta unas ventajas frente a los demás de acuerdo con la aplicación que se este desarrollando.

En cuanto a los algoritmos de localización se trata de uno de los principales aspectos de diseño y desarrollo en las WSN, debido a que la información de ubicación de cada nodo es típicamente útil para conocer el área de cobertura, el despliegue de los nodos, el enrutamiento de la información, servicios de localización y seguimiento de objetivos. Por lo tanto, la estimación de ubicación de un nodo es un desafío técnico importante. Existen diferentes métodos de estimación de localización, los más utilizados se basan en la potencia de transmisión que depende de la energía que tiene la batería del nodo,

el uso de los métodos de localización son diferentes en diversas aplicaciones y en este proyecto en especial hay varios retos que se deben superar.

Cuando se pretende construir una WSN se debe cumplir con unos requisitos que siguen una norma, así que para tener un desarrollo apropiado en un proyecto, es recomendable utilizar el estándar IEEE 802.15.4, debido a que es la norma internacional definida para la comunicación inalámbrica de sensores. El uso del estándar es muy flexible debido a que su concepción no se basa en una empresa o en una marca, como es una norma internacional, en la implementación de los prototipos se puede garantizar el suministro estable de productos, adicionalmente es ventajoso en la implementación por que se presenta competitividad en terminos de los precios. El 802.15.4 se ha especializado en los siguientes aspectos:

- La cantidad de nodos que pueden participar en los sistemas de redes de sensores, debido a que en muchos casos, se requiere la construcción de grandes redes.
- Los periodos de tiempo de establecimiento de la conexión entre los nodos debido a que la conexión a la red de sensores es posible en un tiempo muy corto y se reduce el consumo de energía innecesaria.
- Se ha optimizado la red de sensores inalámbricos (WSN) la velocidad de comunicación, mejorando así la sensibilidad de la comunicación y de ahorro de energía.

Este proyecto consiste en formular, diseñar, simular y realizar pruebas de desempeño a un algoritmo que permita recoger datos correspondientes a la distancia entre los nodos sensores, el proceso continua con el almacenamiento de los datos adquiridos y luego desde cada uno de los nodos se transmiten los datos a los demás integrantes de la red para que cada nodo receptor establezca el eje de coordenadas de su posición. A continuación se aplica un algoritmo que permite hallar la posición de los demás nodos sensores y con esta información cada nodo debe comprobar si se encuentra orientado hacia algún integrante de la red; todo el proceso anterior se requiere con el fin de conocer si existe línea de vista directa o apuntamiento entre nodos orientados desde un nodo hacia otros nodos, a partir de la ubicación entregada por los sensores de los nodos se determinará una región de apuntamiento en la que estarán comprendidas las trayectorias de la línea de vista directa. Finalmente, este proyecto es útil para aquellas aplicaciones en las que se requiera tener identificadas las posiciones y las direcciones de los integrantes de una WSN

## 2. Estudio de la localización de nodos en redes de sensores inalámbricas

En cierto tipo de WSN es importante obtener información precisa en la localización de los nodos sensores, por lo cual surgen varios factores que incluyen la identificación y correlación de la información recogida, esta información contiene el direccionamiento de los nodos, la posición en una área determinada así como la evaluación de la densidad de nodos y la energía de cobertura. Todos los factores anteriores utilizados en conjunto con diferentes métodos y algoritmos permiten conocer la localización de todos los nodos de sensores en una red . A continuación se describen en síntesis estos métodos.

### 2.1. Modelo de canal

El término canal se refiere al medio de transmisión entre las antenas transmisora y receptora. En el modelo de canal se estudian las características de los cambios de la señal a medida que viaja desde la antena del transmisor a la antena del receptor. Estas características dependen de la distancia entre las dos antenas, la ruta(s) tomada por la señal, y el medio ambiente (obstrucción y otros objetos) alrededor de la trayectoria. El perfil de la señal recibida puede ser obtenido a partir de la de la señal transmitida si tenemos un modelo del medio de transmisión entre los dos, este modelo del medio de transmisión se denomina modelo de canal.

En general, el perfil de potencia de la señal recibida se puede obtener mediante la convolución del perfil de potencia de la señal transmitida con la respuesta de impulso del canal. Convolución en el dominio del tiempo es equivalente a la multiplicación en el dominio de la frecuencia. Por lo tanto, la señal  $x$  de transmisión, después de la propagación a través del canal  $H$  se convierte en  $y$  [20] .

El desempeño de una WSN es determinado por las condiciones en que opera el canal de comunicaciones. En este trabajo se modela canal inalámbrico con un path Loss Model, el cuál es un modelo de pérdidas de trayecto a través de una combinación de métodos analíticos y empíricos que se describen a continuación:

$$Pr = PtPo\left(\frac{do}{d}\right)^\alpha \quad (1)$$

en la ecuación 1  $Pr$  es la potencia recibida,  $Pt$  es la potencia transmitida,  $d$  es la distancia,  $Po$  es la potencia a una distancia  $do$  y alfa es el factor de perdidas que se muestra en la tabla (1). [11]:

Tabla 1: Tabla de perdidas en entorno

Entorno	Factor de perdidas
Espacio Libre	2
Area urbana radio celular	2,7 a 3.5
radio celular con sombra	3 a 5
Edificios sin linea de vista	1.6 a 1.8
obstrucción por edificios	4 a 6
obstrucción por fabricas	2 a 3

Este modelo se utiliza para sistemas de cobertura grandes y pequeños, experimentalmente los estudios han demostrado que proporciona precisión en el análisis de trayectos múltiples. El canal opera a una frecuencia aproximada de 2.4 GHz y se tendra en cuenta un factor de perdidas que varia dependiendo de el terreno que se suman los efectos del canal de la señal transmitida.

Los efectos se aplican mediante la adición de una asignación de atenuación a la señal que define los factores de atenuación de la señal. La señal recibida (potencia RX) se obtiene multiplicando la señal de transmisión (potencia de transmisión) con cada asignación de atenuación (debido a esto los valores definidos por las asignaciones de atenuación se obtienen en realidad factores que están por debajo de uno y por lo tanto atenúan la señal cuando se multiplica con ella).

## 2.2. Control de topología

El control de la topología es el arte de coordinar las decisiones de los nodos con respecto a sus rangos de transmisión, generando un orden en las propiedades de la red de sensores, (como la conectividad) reduciendo el consumo de energía de los nodos e incrementando la capacidad de la red[17].

Un aspecto relevante a considerar es que el algoritmo de control de topología debe ser colocado dentro del stack de protocolos de las WSN, aunque aún es un tema de mucho estudio, en este documento se considerará que el algoritmo de control de topología se encuentra entre la capa de enrutamiento y la capa de acceso al medio. En la figura 1 muestra de una forma coherente y organizada la clasificación de los distintos enfoques con los que se puede abordar el problema de control de topología [16].

La técnica homogénea es la más sencilla por que se supone que los nodos usan la

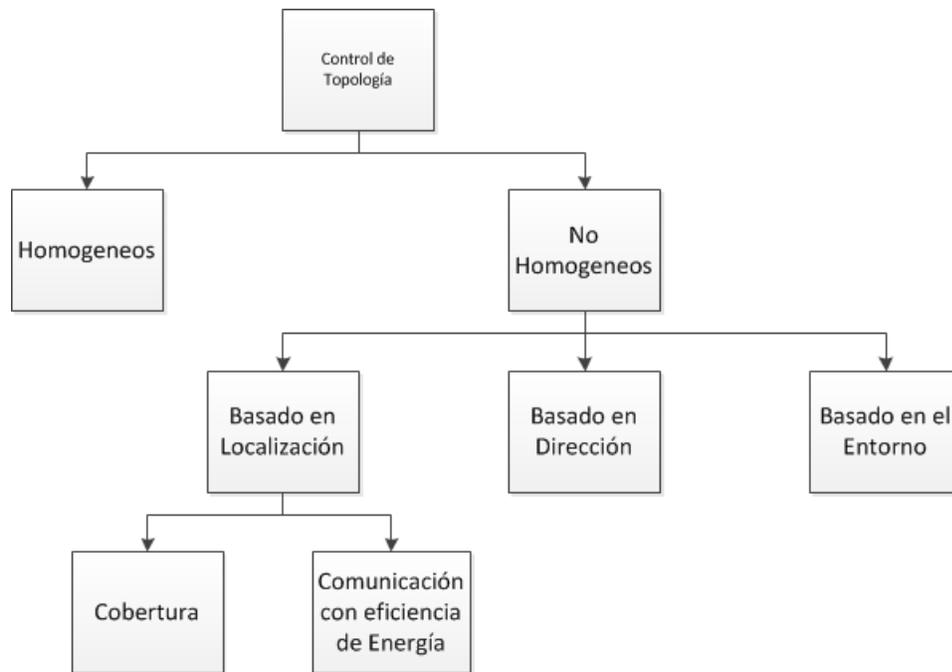


Figura 1: Clasificación de las técnicas de control de topología

misma potencia de transmisión por lo tanto presentan el mismo rango de cobertura todos los nodos, y el problema de control de topología se reduce a la de la determinación del mínimo valor de  $r$  de tal manera que el rango de transmisión Crítico es homogéneo. En la Técnica no homogénea los nodos eligen diferentes rangos de transmisión (siempre y cuando no excedan el rango máximo), esta técnica cuenta con tres subcategorías que dependen del tipo de transmisión para conformar la topología.

En la técnica basada en localización, se conocen las posiciones nodales exactas. Esta información se utiliza por una entidad centralizada para calcular un conjunto de asignaciones de intervalos de transmisión que optimiza una cierta medida (este es el caso del problema Asignación de intervalo y sus variantes), o que se intercambia entre los nodos y se utiliza para calcular una topología que es casi óptima de una manera completamente distribuida (este es el caso para los protocolos utilizados para la construcción de la topología basada en eficiencia energética por unidifusión o por difusión).

En la técnica basada en dirección se supone que los nodos no conocen su posición, pero se puede estimar la dirección relativa de cada uno de sus vecinos. Por último, en las técnicas a base de vecino a base de vecinos, los nodos conocen el ID de los vecinos y son capaces de ordenar de acuerdo con algún criterio (por ejemplo, la distancia, o la calidad del enlace. El algoritmo de control de topología tiene los siguientes aspectos:

**Relación con el Enrutamiento:** La capa de enrutamiento es la responsable de encontrar y mantener las rutas entre la fuente y el destino, cuando el nodo  $i$  tiene que enviar un mensaje al nodo  $j$ , el nodo  $i$  invoca el algoritmo de enrutamiento para revisar la posible ruta conocida de la transmisión de la información desde  $i$  hasta  $j$ ; si el algoritmo no encuentra una ruta conocida la transmisión de la información es descartada. Adicionalmente la capa de enrutamiento es la responsable de retransmitir la información en nodos intermedios.

El algoritmo de control de topología crea y mantiene la lista de nodos inmediatamente vecinos a cualquier nodo, este algoritmo puede activar la actualización de las tablas de enrutamiento en caso de detectar que la lista de vecinos ha cambiado. Así mismo muchos cambios en la lista de vecinos puede generar nuevas rutas y la modificación de rutas pueden generar la activación del algoritmo de control de topología.

**Relación con la capa de acceso al medio:** La capa de acceso al medio es la encargada de regular el acceso al canal inalámbrico compartido, es muy importante por que reduce los conflictos que se pueden presentar en el canal. De otra parte el nivel de potencia del transmisor y el nivel de sensibilidad del receptor son controlados por el algoritmo de control de topología, ya que este algoritmo es el encargado de calcular el correcto rango de transmisión del nodo. La capa de acceso al medio activa el algoritmo de control de topología ya que puede detectar nuevos vecinos mientras escucha los mensajes transmitidos por los nodos. La interacción entre la capa de acceso al medio y el algoritmo de control de topología asegura una rápida respuesta a los cambios que ocurren en la red.

**Rango de transmisión crítico (CTR):** Se refiere al mínimo valor de potencia que requieren los nodos para que se encuentren comunicados con los vecinos más cercanos, se requiere encontrar un valor para  $r$  (rango crítico) el cual garantiza que a lo largo y ancho de la WSN se tiene conectividad entre los nodos.

### 2.3. Métodos de estimación de distancia

La necesidad de abordar la localización de un nodo en tiempo real genera interés en el estudio de técnicas de procesamiento y sistemas de comunicación inalámbricos. Sin embargo no son sistemas generalizados que puedan ser utilizados fácilmente en las WSN, pero si constituyen una infraestructura ideal para explotarse de manera rentable para la localización de nodos. Hasta ahora, se han propuesto algunas soluciones para

la localización de nodos como la medición de distancia entre nodos sensores que se realiza basada en modelos estadísticos útiles para describir el tiempo de llegada time to arrival (TOA), ángulo de llegada angle-of-arrival (AOA), intensidad de señal recibida received-signal-strength (RSS) y pattern-matching based approaches, de las señales medidas en redes de sensores inalámbricos [12]; aunque en las mediciones también se utilizan sistemas de RF y medios acústicos para conocer la precisión de estimación de localización posible para un determinado conjunto de mediciones.

A continuación en la tabla (2) se presenta un cuadro conceptual de estos métodos de medición que son utilizados para localizar los nodos; resumido de la recopilación de varios autores.

## 2.4. Métodos de cálculo de posición del nodo

Varios métodos pueden ser utilizados para calcular la posición de un nodo, tales métodos incluyen la triangulación, la lateración, y multilateración [3], que son derivados de la triangulación [8], enfoques probabilísticos, bounding box, y posición central[3].

A continuación se presenta un cuadro conceptual de estos métodos de medición que son utilizados para localizar los nodos; resumido de la recopilación de varios autores [3].

**Triangulación:** Esta técnica usa propiedades geométricas de los triángulos para estimar la posición midiendo las distancias y el Angulo hacia tres nodos de referencia, utiliza TOA, TDOA, RSS. la transmisión puede ser en un sólo sentido y requiere una sincronización perfecta, mientras que si se utiliza La transmisión en dos sentidos marcas de tiempo deben ser insertadas en la señal de transmisión. Una debilidad es la imprecisión, es utilizada en Redes celulares, DSSS, UWB.

**Multilateración:**es una técnica que usa propiedades geométricas de los triángulos para estimar la posición midiendo las distancias a varios puntos de referencia, utiliza TOA, TDOA, RSS. La transmisión en un sólo sentido requiere una sincronización perfecta, mientras que La transmisión en dos sentidos no. Marcas de tiempo deben ser insertadas en la señal tx, es capaz de obtener posiciones precisas. Una debilidad es que utiliza demasiadas ecuaciones y sistema computacional de punto flotante. Está técnica es utilizada en redes celulares, DSSS, UWB.

**Angulación AOA:** Utiliza la información del ángulo y distancia recibidos para construir líneas entre nodos móviles y fijos y utilizar su intersección para encontrar la ubicación del nodo móvil. La posición puede determinarse con solo tres unidades de medición para el posicionamiento en 3-D o dos unidades de medición para el posicionamiento en 2-D, no se requiere ninguna sincronización de tiempo entre las unidades

Tabla 2: Métodos de Medición

Técnica de Localización	Características	Fortalezas y Debilidades	Usos y Aplicaciones
TOA, RTOF	Es el método más sencillo e intuitivo, calcula la distancia utilizando la información extraída del tiempo de retardo entre un nodo Tx móvil y otro nodo RX fijo	La transmisión en un sólo sentido requiere una sincronización perfecta. Marcas de tiempo deben ser insertadas en la señal tx	Redes celulares
TDOA	Calcula la distancia utilizando la información extraída del tiempo de retardo entre un nodo Tx móvil y varios nodos Rx fijos	Sincronización muy precisa entre nodos móviles, mientras que entre nodos móviles a fijos la sincronización no debe ser tan precisa.	Redes de sensores inalámbricas, Standar 802.11
AOA	Utiliza la información del ángulo recibido para construir líneas entre nodos móviles y fijos y utilizar su intersección para encontrar la ubicación del nodo Móvil	Requiere hardware adicional como arreglos de antenas que generan costos adicionales y tamaños de nodo más grandes. Altamente sensible a las múltiples trayectorias generadas por NLOS.	Aplicaciones que requieran nodos fijos en lugar de nodos móviles debido al tamaño.
RSSS	Se estima la distancia en base a la atenuación por propagación de la señal enviada desde un nodo fijo a un nodo móvil.	Se debe utilizar un modelo de propagación preciso. Su costo es bajo por ser capaz de estimar RSS.	Es utilizado en aplicaciones que requieren estimaciones que acepten un porcentaje alto de error.
Pattern Matching	Utiliza la información de la huella de la señal de radio medida señal en diferentes ubicaciones geográficas. El cálculo de la posición de un nodo consiste en comparar sus características de señal actuales con señales previamente recopiladas	Necesita una etapa de entrenamiento fuera de línea para obtener una base de datos, además, esta base de datos pueden no ser fiable si el canal y el medio ambiente cambian con el tiempo	Se utiliza en redes de área local utilizando RSS como la métrica obtenida en la base de datos. Utilizada en los sistemas celulares

de medición, Requiere de hardware complejo posiblemente reforzado con brújulas y acelerómetros, tienen la posibilidad de proporcionar un mejor posicionamiento, la precisión puede ser limitada debido al empobrecimiento, por las reflexiones de trayectoria múltiple.

**Enfoques Probabilísticos:** Es un técnica que estima la distancia suponiendo que hay  $n$  candidatos de ubicación  $L_1, L_2, L_3, \dots, L_n$ , y  $s$  es el vector de intensidad de señal obtenida, utiliza la fórmula de Bayes. Cuando un nodo desconocido recibe un paquete desde un nodo de referencia, se puede estar en cualquier lugar alrededor del nodo de referencia con la misma probabilidad. Cuando otro paquete se recibe de otro nodo de referencia, el nodo desconocido calcula su posición de nuevo. Cuando la información de posición nueva se recibe de otros nodos, se hace posible identificar la ubicación probable del nodo desconocido. Requiere sistema computacional robusto y alta capacidad de almacenamiento, la intensidad de la señal recibida es inexacta, el algoritmo es independiente del tipo de distribución. El algoritmo puede adaptarse a Diversas condiciones ambientales mediante el cambio de la probabilidad, presenta dificultades en la difusión de información sobre la distancia bajo condiciones no ideales, lo que lleva a una baja cobertura en la mayoría de los casos.

**Bounding Box:** Técnica que para cada nodo  $i$  de referencia, un límite se define como un cuadrado con su centro en el nodo la intersección de todas las dimensiones de las cajas determina la posición. Computacionalmente no requiere punto flotante, usa menos recursos de procesador que la trilateración. Un nodo desconocido sólo podría consultar algunos de sus vecinos, esto reduciría Los costos de comunicación, pero aumento en computación. El algoritmo puede ser iterado.

**Claves para el diseño del control de la Topología:** Como un primer aspecto del diseño del control de topología se tiene la capacidad de los nodos para tener su propia visión de la red, que se obtiene mediante el intercambio de mensajes con los nodos vecinos. Los mensajes se intercambian mediante enlaces bidireccionales y ayudan a preservar la propiedad de conectividad mientras la potencia de transmisión y la sensibilidad de recepción se encuentren dentro de un umbral determinado por la distancia entre los nodos.

El segundo aspecto a tener en cuenta en el diseño del control de topología es la calidad de la información emitida por los nodos; esta calidad se considera de un tipo de calidad media, ya que entregara la ubicación, el número de identidad de cada nodo y la dirección hacia la que se encuentra apuntando.

El último aspecto a tener en cuenta es el número de nodos que entregaran información a través del enlace inalámbrico, una menor cantidad de nodos conectados a cada rama del árbol no genera interferencia entre transmisiones recurrentes.

## 2.5. Algoritmo de localización

El algoritmo de localización es el componente principal en un sistema de WSN, este componente puede determinar las posiciones de los nodos en una WSN con la información relativa puede calcular las distancias y las posiciones de cada nodo [13].

Los algoritmos de localización se pueden clasificar en unas pocas categorías: distributed or centralized position computation [19]; with or without an infrastructure; relative or absolute positioning [6]; designed for indoor or outdoor scenarios [2]; and one hop or multihop [3]. En el algoritmo distributed or centralized position computation del mismo autor encontramos otros tres algoritmos que son Ad-hoc positioning, Robust positioning, and N-hop multilateration [9], aunque estos tres algoritmos fueron desarrollados de forma independiente, comparten una estructura común estructura.

**APS Ad Hoc positioning system:** Es un algoritmo con un número reducido de nodos baliza (tres o más) se comunica con nodos desconocidos, luego, cada nodo calcula su distancia a los nodos baliza mediante varios saltos; una vez que estas distancias se calculan, los nodos pueden calcular sus posiciones usando. Utiliza trilateración y puede tener tres métodos de propagación para calcular la distancia en cada salto: Dv-Hop, Dv-Distance, and Euclidean. Aunque requiere pocos nodos baliza, el error en el sistema puede ser alto debido a muchos saltos de la señal transmitida.

**Recursive Position Estimation:** Este algoritmo estima la posición en base a un conjunto de de nodos baliza (por ejemplo, 5 por ciento De los nodos), un nodo debe tener al menos tres vecinos de referencia con el fin de calcular su posición, mientras aumenta la información la localización se realiza de forma iterativa así que nodos recién ubicados o establecidos se convierten en nodos de referencia. El número de nodos de referencia aumenta rápidamente, de manera que la mayoría de los nodos puede calcular su posición por trilateración o multilateración y puede tener tres métodos de propagación para calcular la distancia en cada salto: Dv-Hop, Dv-Distance, and Euclidean. Adecuado para redes de gran tamaño debido a la propiedad de escalado lineal del algoritmo, eficiente en cualquier topología, requiere menor número de recursos computacionales. Tiene la Desventaja de propagación de errores de localización.

**Relative or absolute positioning algorithm:** Se basa en proximidad estocástica. Adecuado para redes de gran tamaño debido a la propiedad de escalado lineal del algoritmo, El método es eficiente en topologías irregulares o regulares, requiere menor número de recursos computacionales en la simulación.

**DV-Hop:** Utiliza encaminamiento por vector-distancia, una baliza difunde un mensaje que contiene las posiciones de todas las balizas con un contador de saltos (hop-count) inicializado a 1. Cada nodo mantiene el valor mínimo del contador de cada baliza

entre todos los mensajes que recibe, e ignora aquéllos cuyos contadores son más altos que los que tiene almacenados. Utiliza trilateración y puede tener tres Métodos de propagación para calcular la distancia en cada salto: Dv-Hop, Dv-Distance, and Euclidean. Es estable y previsible.

**Distributed position computation:** Se requiere La colocación de posiciones conocidas alrededor del perímetro de la red con tan sólo cuatro posiciones conocidas disminuyendo la incertidumbre Utiliza trilateración o Multilateración. Rápida solución para las redes con cientos de nodos, exige una significativa sobrecarga computacional realiza el seguimiento de un objeto a través de la red de sensores.

## 2.6. Otras características

A continuación se describen muy sencillamente otras características que aunque no se tenido en cuenta para este proyecto son muy importantes en otras aplicaciones.

### 2.6.1. Recolección de datos

En una WSN la información es presentada en forma de datos, los cuales son recogidos por los sensores, ya sea en todos o en algunos de los nodos, los datos se envían a través de comunicaciones inalámbricas a una estación base central para su posterior procesamiento, se debe tener en cuenta que la recolección de datos se puede dividir en la fase de despliegue del mensaje de control, la etapa de difusión y la fase de entrega de datos [10].

### 2.6.2. Tecnologías

Las soluciones adecuadas para adaptar en WSN son: ZigBee, Z-Wave, Insteon, Wave-nis y soluciones basadas en IP. Zig-Bee y 6LoWPAN fueron diseñados para uso general, el resto de las soluciones fueron desarrolladas para aplicaciones específicas [7]. Una vez identificados los principios de medición los algoritmos de posicionamiento y las métricas de rendimiento de los sistemas de localización de posicionamiento, hay dos enfoques básicos para el diseño de una red inalámbrica, el primer enfoque es el desarrollo de un sistema de señalización y una infraestructura de red de medición de ubicación unidades se centró principalmente en aplicaciones de localización inalámbrica. El segundo enfoque es utilizar una infraestructura existente de red inalámbrica para localizar un blanco [4] sin embargo, puede ser necesario utilizar algoritmos más inteligentes para compensar la baja precisión de localización.

En este documento se resumen de [8] la síntesis de la recomendación de aplicaciones de acuerdo con su cobertura de los actuales sistemas de posicionamiento basados en las tecnologías inalámbricas: GPS, RFID, Sistema Celular, UWB, WLAN (IEEE 802,11), Bluetooth (802.15), Sistemas UHF y bandas ISM

### **2.6.3. Conectividad y cobertura**

Para obtener la cobertura se debe integrar la información recopilada mediante el algoritmo de control de topología, a partir de esta etapa se puede realizar el seguimiento de objetos móviles en la red, para realizar este seguimiento el algoritmo sera implementado en un servidor de control centralizado donde los nodos están conectados inalámbricamente, o en sistemas de control distribuidos, así es posible comparar la cobertura en términos de poder consumo, costo y rendimiento; en este prototipo otros factores que influyen en la cobertura como son los obstáculos, las condiciones ambientales y el ruido no serán tenidos en cuenta.

### **2.6.4. Protocolos**

Se debe tener en cuenta los diferentes protocolos utilizados acorde con las capas del modelo OSI, la red debe considerarse como una sola entidad, donde los protocolos de comunicación colaboran eliminando redundancias en comunicación y computación, y mantienen una distribución uniforme de la energía. Sólo con esa atención cuidadosa a los detalles del consumo de energía en cada punto del proceso de diseño se espera ver una red cuyos sensores pueden ofrecer servicio continuo.

### **2.6.5. Tiempo de vida de la red**

Es el lapso de tiempo que existe entre el despliegue de los nodos de la red hasta el instante en que la red es considerada no funcional, es d anotar para considerar no funcional una red se debe tener en cuenta el tipo de aplicación, algunas WSN dejan de ser funcionales desde que el primer nodo muere, otras WSN dejan de ser funcionales cuando se generan particiones en la topología de la red y otras WSN dejan de ser funcionales en el momento que se crea una pérdida de cobertura.

### 2.6.6. Seguridad

Las redes de sensores inalámbricos requieren decisiones cuidadosas y tomar decisiones en cuanto a los diferentes aspectos de medidas de seguridad, los mecanismos actuales de seguridad deben garantizar la funcionalidad de la red.

### 2.6.7. Aplicaciones

Las Redes de sensores inalámbricos WSN han generado una especial atención durante los últimos años, una lista escalonada de aplicaciones civiles y militares pueden emplear redes de sensores inalámbricas para una mayor eficacia, sobre todo en zonas hostiles y remotas; Las ejemplos incluyen la prevención y gestión de desastres, protección de fronteras, vigilancia en el terreno de combate [5]. Se han convertido en una excelente herramienta para aplicaciones militares relacionadas con la detección de intrusos, vigilancia del perímetro, la recopilación de información para apoyo logístico en un área desconocida o para detección de localización por detección de movimiento [10].

### 2.6.8. Tablas de enrutamiento

Cada nodo en la red debe mantener una tabla de enrutamiento que es construida después de recibir los mensajes hello y retransmite la información al próximo salto. Las entradas de la tabla de Enrutamiento consisten en una dirección de destino al próximo salto y la distancia estimada a cada destino. Cada nodo es un destino y tiene de entrada una tabla de enrutamiento, por lo tanto todas las rutas son conocidas en su totalidad.

## 2.7. Metodología de software

En la práctica se realiza el análisis de OMNET y de MiXiM, para lo cuál se ha utilizado ingeniería inversa con varios propósitos como herramienta de aprendizaje para entender mejor la aplicación, como una manera de hacer nuevos frames compatibles con los que cuenta actualmente OMNET y MiXiM y, como una forma de comprobar que el código cumple con las especificaciones.

A continuación se describen los pasos que se han seguido para completar el proceso de ingeniería inversa, que se requiere para este proyecto:

- 1. Buscar el código de los programas OMNET y MiXiM.

- 2. Identificar las variables con las que cuenta y las variables que hacen falta.
- 3. Se han estudiado las rutinas que se requieren, descartando las que no se requieren.
- 4. Se modificaron rutinas y se construyeron nuevas.
- 5. Se repiten los pasos 3-4 a lo largo del desarrollo del proyecto.
- 6. Las rutinas “importantes” o componentes funcionales se recopilan.
- 7. Se asigna un significado a cada componente funcional, esto es (a) explicando qué hace cada componente funcional en el conjunto del sistema y (b) se explica qué hace el sistema, a partir de los diferentes componentes funcionales.

### 3. Especificaciones

En este proyecto se formuló un algoritmo que permite recoger datos correspondientes al eje de coordenadas de la posición de los nodos, almacenar los datos adquiridos y desde cada uno de los nodos transmitir los datos a los demás para que cada nodo receptor establezca las coordenadas de todos los dispositivos y compruebe si se encuentra orientado hacia algún integrante de la red.

En esta clase de proyectos es común que los resultados acerca de la posición de un nodo, sean imprecisos, así que se buscará la manera de explicar si en los resultados el error presentado puede ser previsto, para ello se debe tener en cuenta el error de localización que pueden manifestar los nodos. En este proyecto dependiendo de la aplicación, los elementos de la red deberán recalcular la nueva posición y orientación del nodo, lo cual permitirá una exactitud en el apuntamiento que cubrirá una variación alrededor del nodo que depende del tipo de aplicación.

En la simulación se espera visualizar el movimiento y rastreo de los nodos en tiempo real en una plataforma de simulación denominada OMNET++<sup>®</sup> y se espera que la red mantenga una conectividad total, lo anterior quiere decir que el 100 % de los nodos se han localizado y permanecen comunicados.

La simulación se realizará en OMNeT por que es modular y es un sistema extensivo de software basado en C++, se utiliza sobre todo para la construcción de simuladores de red que incluye redes cableadas e inalámbricas, adicionalmente proporciona frameworks modelo, que son desarrollados para utilizarlos en proyectos independientes como es el caso de este trabajo. OMNeT ofrece un IDE basado en Eclipse, un entorno de ejecución gráfica, y una serie de otras herramientas. Existen extensiones para la simulación en tiempo real, emulación de red, lenguajes de programación alternativos (Java, C), la integración de bases de datos, integración de SystemC, y varias otras funciones. Adicionalmente a OMNeT se utilizara MIXIM, que es un framework de modelamiento de OMNeT, creado para redes inalámbricas móviles y fijas (redes inalámbricas de sensores, redes de área corporal, redes ad-hoc, redes vehiculares, etc.) que ofrece modelos detallados de la propagación de ondas de radio, la estimación de la interferencia, el consumo de energía de radio transmisor-receptor inalámbrico y protocolos MAC (por ejemplo Zigbee). Es una fusión de varias frameworks de OMNeT diseñado para apoyar simulaciones móviles e inalámbricas.

### 3.1. Entradas

1. **Conjunto de Nodos:** Se refiere a todos los elementos de la que deben estar comunicados mediante enlaces Inalambricos, los nodos de red deben servir como repetidores inalambricos y cuentan con un sensor de orientación, el conjunto de todos los los nodos se distribuyen en una área determinada de forma uniforme.
2. **Campo de visión:** El campo visual es la porción del espacio que el nodo es capaz de cubrir con el apuntamiento. Este campo permite determinar los límites del cono de apertura hacia donde se encuentre direccionado el nodo, si los nodos se encuentran demasiado cerca pueden ocasionar disminución de la amplitud del cono de apertura generado en el campo visual.
3. **Constante de atenuación:**Esta constante prmite modificar las perdidas de propagación en el modelo de canal inalambrico, así se pueden considerar diferentes tipos de terreno en la simulación.
4. **Área de Cobertura:** Es la región en la que se ubican los nodos que serán parte de la topología de la red, siempre y cuando reciban algún tipo de señal que los asocie a la red. Una de las principales aplicaciones de redes de sensores inalambricos es proporcionar una cobertura adecuada de la región de implementación. Típicamente, el rango de detección de un sensor es un disco circular centrado en el sensor, cuyo radio se denominado radio de detección del sensor. Un punto se dice que está cubierto por una red de sensores, si cae en la detección de al menos un sensor, y una región se denomina cubierta si cada nodo en esta región está cubierto [14]. En este proyecto el área presenta una forma cuadrada cuyo lado  $L = 300$  m.
5. **Error:** Se refiere a la diferencia entre el valor real de la posición de cada uno de los nodos con el valor que resulta del cálculo hecho por el algoritmo de localización.

### 3.2. Salidas

1. **Distancia entre nodos:**Esta variable se medirá para todas las respuestas de medición de distancia, calculo de posición y calculo de apuntamiento.
2. **Alertas:** Señales que informan el comportamiento de los estados que se simulen en la red.
3. **Control de Topología:** reducción lógica de enlaces que se deben comunicar para mantener conectividad y una cobertura máxima manteniendo una estructura lógica en árbol.

4. **Tabla de Enrutamiento:** Es una base de datos electrónica basada en Hardware que almacena las rutas a los diferentes nodos en una red.
5. **Tabla de Apuntamiento:** Es una base de datos electrónica almacenada en el sink basada en Hardware que almacena las direcciones desde y hacia cada uno de los diferentes nodos en una red.
6. **Tiempo de Convergencia:** La convergencia es un termino general que se utiliza para conocer el tiempo de duración y determinar el número de nodos conectados.

Desarrollo de software: Se debe crear una aplicación a la medida que resuelva el algoritmo, utilizando e integrando frameworks existentes.

### 3.3. Descripción general y diagrama en bloques

El proyecto se aborda en orden, realizando un plan que tiene una serie de procesos y actividades, en el siguiente diagrama de bloques general se muestra cada uno de los pasos que se realizó en este trabajo.

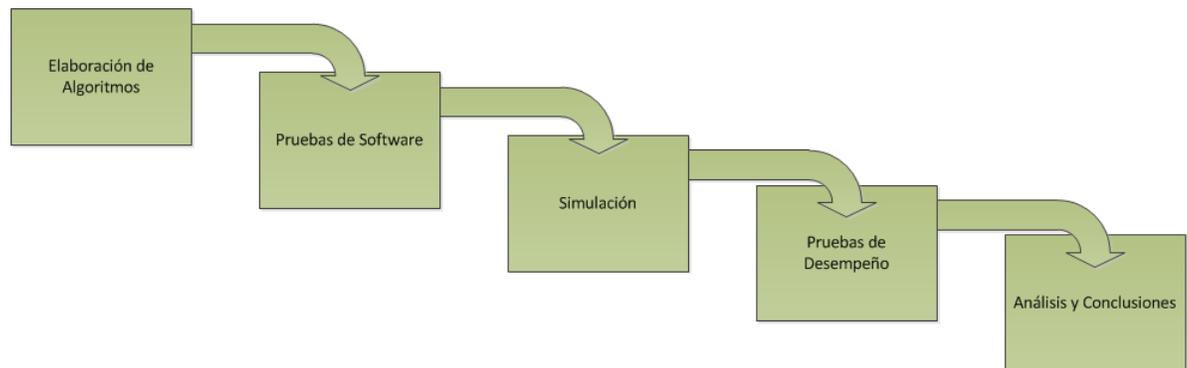


Figura 2: Diagrama de flujo del proyecto

#### 3.3.1. Diseño de algoritmos del sistema

Se diseñaran modelos de algoritmos de acuerdo a la técnicas de medición de distancia, de localización y de apuntamiento seleccionados. los algoritmos permiten crear un modelo analizando los datos que facilite cierto tipos de patrones o tendencias. En cuanto al control de topología se tiene definido el uso del algoritmo A3, ya que el algoritmo utiliza los resultados de este análisis para definir los parámetros óptimos para la creación del modelo. Estos parámetros se aplican luego a través de los datos de todo el conjunto para extraer patrones viables y estadísticos.

### **3.3.2. Pruebas de Software**

Se realizan las pruebas de la aplicación y de los frameworks a utilizar. La fase de pruebas del sistema tiene como objetivo verificar el software para comprobar si este cumple con los requisitos del proyecto. Dentro de esta fase se desarrollan varios tipos distintos de pruebas en función de los objetivos de las mismas. Algunos tipos son pruebas funcionales, pruebas de usabilidad, pruebas de rendimiento, pruebas de seguridad, etc. Este trabajo se centra en pruebas funcionales con interfaces gráficas. Estas pruebas verifican que el software ofrece a los actores humanos la funcionalidad recogida en la especificación.

### **3.3.3. Simulación**

En esta etapa se implementan los modelos de algoritmos que deben ser ejecutados en varias ocasiones, el algoritmo propuesto se implementa en un generador de eventos discretos con mecanismos estocásticos para observar el comportamiento resultante del sistema conforme pase el tiempo. Se implementará la simulación por medio de eventos discretos ya que de otra manera los datos de interés pueden ser difíciles de seguir en el dominio del tiempo.

### **3.3.4. Pruebas de desempeño**

Las pruebas de desempeño muestran la calidad de la implementación del modelo planteado, en este mismo documento en la sección 4.5 se explica en detalle cada una de las pruebas que se realizaron. La simulación permite evaluar el impacto de posibles cambios en las condiciones del sistema configurando diferentes escenarios con el fin de obtener distintos resultados que permiten comprender en qué grado afectan al sistema los cambios en las entradas.

### **3.3.5. Análisis y conclusiones**

Luego de cumplir con las etapas anteriores se analizarán los resultados de las simulaciones y se compararán con los modelos teóricos. La simulación produce unas salidas cuyos resultados son unos valores esperados y luego de varias ejecuciones se debe realizar un análisis estadístico de los datos simulados para poder concluir con la mejor aproximación.

## 4. Desarrollo

En este punto se describe cada una de las partes que conforman el trabajo realizado, inicialmente se explican en detalle los algoritmos diseñados para elaborar el respectivo código en C++ requerido para interactuar con los frames de MiXiM, a continuación se explica cómo se utilizaron los módulos de MiXiM tales como la capa de aplicación, la capa de red, la capa MAC y los módulos de la capa PHY para los nodos de la WSN con el fin de evaluar y disponer de datos exactos con los que se realizó la predicción de los modelos analíticos del estándar IEEE 802.15.4. Adicionalmente se explica cómo se obtienen los resultados de las simulaciones y predecir las métricas de desempeño y así comparar el rendimiento de MiXiM con el rendimiento derivado analíticamente.

### 4.1. Diseño de algoritmos

En la elaboración de este trabajo se requirió investigar y diseñar acerca de diferentes tipos de algoritmos requeridos en WSN, se explica un algoritmo de control de topología, algoritmo de enrutamiento, algoritmo de localización, y el algoritmo requerido para realizar el cálculo del apuntamiento entre los integrantes de la WSN.

Luego de realizar una revisión del estado del arte se establece que el desarrollo del proyecto se resolverá el problema de control de la topología utilizando el algoritmo A3: A Topology Construction Algorithm for Wireless Sensor Networks por su sencillez y por la demostración que realizan los autores de su funcionalidad, a continuación se resuelve el problema de localización realizando una modificación al método *Recursive position estimation* y finalmente se integrará la propuesta del algoritmo de apuntamiento y se completará la solución del problema. En la figura 3 se observa el proceso que se utilizó para este propósito.

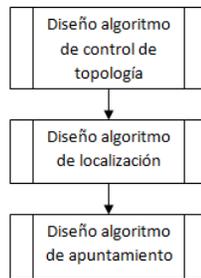


Figura 3: Diagrama de procesos de diseño de algoritmos

#### 4.1.1. Algoritmo de control de topología

El algoritmo A3 [21] consiste en dos componentes separados: está el algoritmo de construcción de la topología que descubre una topología reducida, construir una topología consiste en crear trayectorias de comunicación entre los nodos con el fin de evitar sobrecarga de mensajes de broadcast, esta sobrecarga se evita si se utilizan trayectorias de comunicación previamente definidas. Por otra parte se debe tener en cuenta el algoritmo de mantenimiento de la topología, que modifica la topología reducida cuando no se puede prestar el servicio solicitado por más tiempo. se espera que estos dos mecanismos funcionen de manera iterativa hasta que la energía de la red se agota.

**Reconocimiento del Entorno:** El proceso de construcción se inicia por un nodo predefinido como el nodo principal de toda la red que puede ser llamado Sink, después se despliegan los demás nodos integrantes de la red. El Sink, nodo A en la figura 4a, inicia el protocolo de descubrimiento de la topología mediante el envío de un mensaje de broadcast denominado Hello inicial, este mensaje permitirá a los vecinos de A conocer el nodo al que pueden conectarse directamente y que debido a esta condición será conocido como el nodo padre de estos vecinos. En la figura 4a, los nodos B, C, D, y E recibirán el mensaje Hello y calcularán la distancia de acuerdo con la ecuación (1), en donde evidencia la necesidad de la intensidad de potencia de la señal recibida.

$$d = \sqrt{\frac{Ct.Pt}{Pr}} \quad (2)$$

Si el nodo que recibe el mensaje no ha sido cubierto por otro nodo o no ha recibido otros mensajes Hello, se establece su estado como cubierto, adopta al nodo transmisor como su "Nodo primario", y le responde con un mensaje de Reconocimiento ACK, como se muestra en la figura 4b. Este mensaje incluye además una métrica de selección (explicado más adelante) que se calcula en base en la intensidad de la señal del mensaje Hello recibido.

$$M_{x,y} = W_e \frac{E_x}{E_{max}} + W_d \frac{RSSI_y}{RSSI} \quad (3)$$

La métrica descrita en (2) se utilizará más adelante por el nodo Sink para ordenar los candidatos. Si el nodo receptor ha sido cubierto por otro nodo, recibe este mensaje Hello para tomarlo como segunda opción en caso de enviar un mensaje al nodo padre principal y no recibir ninguna respuesta, este método le permite a los nodos tener rutas alternas en su transmisión.

Si un nodo padre no recibe ningún mensaje de Reconocimiento de Padre de sus

vecinos se apaga, como es el caso de los nodos E y B en la topología final, como se muestra en la figura 4f, dado que no tienen hijos.

### **Selección de nodos Hijo:**

El nodo que transmite el mensaje Hello establece un tiempo de espera para recibir las respuestas de sus vecinos, de acuerdo (2) el nodo sink construye una tabla con el orden de cada métrica en donde se almacena una lista de candidatos. Una vez este tiempo de espera expira, el nodo padre ordena la lista según la métrica seleccionada. El nodo primario emite un mensaje de reconocimiento de hijos que incluye la lista ordenada de todos sus candidatos. En la figura 4c, el nodo A envía la lista ordenada a los nodos B, C, D, y E.

Una vez que los nodos candidatos reciben la lista, establecen un período de tiempo de espera proporcional a la posición en la lista de candidatos. Durante ese Tiempo los nodos esperan mensajes sleeping de nodos que recibieron la lista y se denominan nodos hermanos. Si un nodo recibe un mensaje Sleeping durante el tiempo de espera el nodo que recibe el mensaje se apaga, lo que significa que uno de sus hermanos está mejor calificado para formar parte del árbol. Con base en este esquema, el mejor nodo de acuerdo con la métrica enviará primero un mensaje, bloqueando cualquier otro nodo que se encuentre en su rango. Por lo tanto, sólo los otros nodos candidatos fuera de su área de cobertura tienen la oportunidad de iniciar su propio proceso de generación hasta ahora descrito. Por ejemplo, en la figura 4d, el nodo D recibió un mensaje Dormir antes de que expirara su temporizador, por tanto se apaga. De lo contrario, como ocurre con los nodos C y E envían un nuevo mensaje sleeping a sus hermanos ya que que no reciben mensaje sleeping del nodo sink, es en ese momento que estos nodos se convierten en nuevos "nodo padre" comienza su proceso de búsqueda de candidatos.

**Segunda Oportunidad:** hay algunos casos en los que un nodo envía un mensaje sleeping y se convierte en un cuello de botella en el acceso a una sección de la red. Para evitar esta situación, cada nodo establece un temporizador una vez que se recibe el Mensaje sleeping para enviar un mensaje de Hello y comienza su propio proceso de construcción. Como se muestra en la figura 4e. El nodo D se despertará, enviará un mensaje Hello y se encontrará que el nodo G aún no está cubierto, así que el nodo D se activará como nodo padre de el nodo G. Esta operación aumenta la sobrecarga del algoritmo, pero garantiza la cobertura total de los nodos en el gráfico.

**Modificación del Algoritmo A3:** En este trabajo se propone una modificación al algoritmo A3 que se denomina APA, en él se estudia la probabilidad de que una región que está cubierta por sensores desplegados aleatoriamente presenten cambios en el radio de detección. En este proyecto se propone un método para garantizar que el 100 % de los nodos pertenezcan a la topología de la red y no se encuentren nodos desconectados de dicha topología, estos nodos que se encuentren fuera de la topología inicialmen-

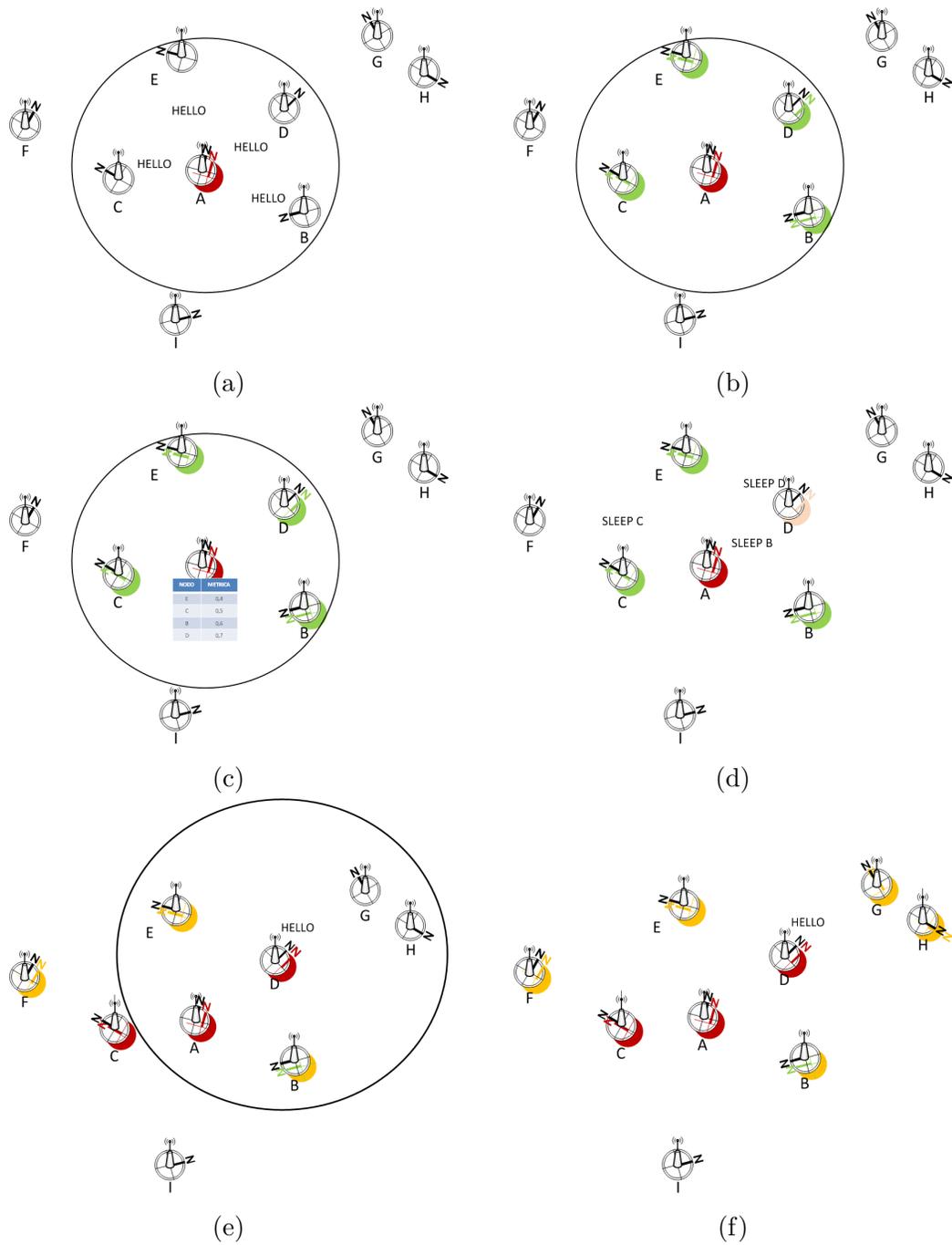


Figura 4: Algoritmo A3

te se denominaran huérfanos. Adicionalmente se propone que los nodos puedan tener alternativas para enviar sus mensajes utilizando una topología alternativa.

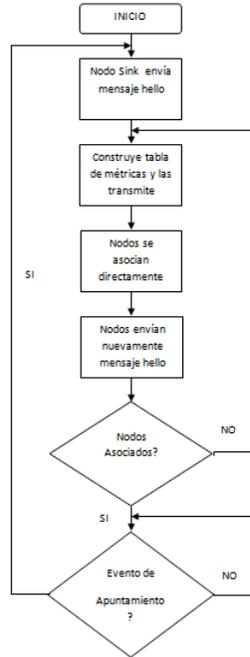


Figura 5: Diagrama de flujo del control de topología

Al iniciar la red todos los nodos establecen un temporizador que expira cuando se estime que la red ha llegado a converger, si al finalizar dicho temporizador uno o varios nodos no se han asociado a la topología de la red el nodo envía un mensaje de huérfano indicando a los demás nodos que quiere asociarse a la topología de la red, cualquier nodo de la red que ve ese mensaje responde con un reconocimiento de huérfano, así que debido a esto el nodo huérfano puede recibir varios reconocimientos y se adopta al nodo que tenga mayor nivel de señal recibida. Si no recibe ningún reconocimiento el nodo aumentará la potencia y repetirá el proceso, en la figura 5 se muestra el diagrama de flujo. La otra modificación del algoritmo se presenta cuando los nodos reciben los mensajes Hello ya que reciben el mensaje primario y toman este nodo que emite el mensaje como nodo padre, sin embargo si reciben otro mensaje Hello, guardarán en su tabla la conexión a dicho nodo como una trayectoria de comunicación secundaria, si al enviar por la trayectoria principal no se llegase a obtener respuesta.

**Solución al problema de asignación de rango:** En esta propuesta los nodos requieren que la potencia de transmisión y la sensibilidad de recepción sea ajustada, esta situación es un problema para cada uno de los nodos que debe ser resuelta satisfactoriamente, en esta propuesta se utiliza un algoritmo que realiza un incremento paso a paso de la potencia de transmisión y una reducción paso a paso de la sensibilidad.

### 4.1.2. Mensajes de control de topología

El algoritmo A3 utiliza cuatro tipos de mensajes: Mensaje Hello, Mensaje de Reconocimiento de nodo Padre, mensaje de Reconocimiento de nodos hijos, y mensaje Sleeping. Adicionalmente la modificación planteada en este proyecto se requiere un quinto mensaje denominado mensaje huérfano

- **Hello:** Mensaje que envía un nodo con el que los nodos vecinos que se encuentren dentro del CTR identifican al nodo que envió el mensaje como un nodo padre.
- **Reconocimiento de nodo padre:** Mensaje de respuesta a un mensaje Hello en el que el nodo confirma que reconoce al nodo que envió el mensaje Hello como nodo padre.
- **Reconocimiento de nodos hijos:** Mensaje que se envía luego de recibir el mensaje de reconocimiento como nodo padre, este mensaje es enviado a todos los nodos que respondieron el mensaje de reconocimiento de nodo padre.
- **Sleeping:** Mensaje que manda a nodos hermanos a desactivarse, el mensaje es enviado por los nodos que recibieron el mensaje de reconocimiento de hijo a sus hermanos con el fin de mostrar quien está más cerca al nodo padre.
- **Nodo huérfano:** Mensaje enviado por los nodos que no reciben mensaje Hello

### 4.1.3. Algoritmo de enrutamiento

En este trabajo se utilizó un algoritmo diseñado para redes de sensores denominado Wiseroute por los creadores de MiXiM, este algoritmo es considerado por los autores como un sencillo algoritmo de enrutamiento sin bucles que construye un árbol de enrutamiento desde un punto central de la red. El nodo Sink (el nodo en el centro de la red) difunde un mensaje de construcción de ruta. Cada nodo de la red que se encuentra dentro del alcance del mensaje lo recibe y selecciona como padre en el árbol de enrutamiento, luego el nodo que recibe este mensaje retransmite el mensaje de construcción de ruta. Según los autores este procedimiento maximiza la probabilidad de que todos los nodos de la red pueden unirse a la red, y evita bucles. El parámetro `rssThreshold` es un umbral que determina la potencia mínima que se requiere para evitar que el mensaje se transmita por enlaces de baja calidad.

#### 4.1.4. Algoritmo de localización

El algoritmo de localización propuesto será el APS Ad Hoc positioning system con una modificación que consiste en calcular la posición de todos los nodos partiendo de tres nodos baliza [23]. En este método modificado el nodo Sink puede o no ser uno de los nodos baliza, si conoce su posición se comunica inicialmente con nodos desconocidos mediante un mensaje de broadcast hacia los nodos que se encuentren a una distancia adecuada, los nodos que reciben este mensaje se denominarán nodos baliza secundarios luego de recibir el mensaje de broadcast los nodos baliza secundarios lo reenvían y posteriormente reciben el reconocimiento de los vecinos que se encuentren en su rango, con estos reconocimientos los potenciales nodos baliza secundarios construyen una tabla que contiene el número de vecinos que respondieron el reconocimiento, los potenciales nodos baliza secundarios envían la tabla al nodo Sink con el número de vecinos, el nodo Sink seleccionará como nodos baliza secundarios aquellos nodos que contengan el mayor número de vecinos. El nodo Sink, solicita a los nodos baliza secundarios que hallen la distancia entre ellos por medio de mensajes para este propósito, luego cuando los nodos baliza secundarios hallan esa distancia envían la información al nodo Sink, el nodo Sink con esta información calcula la posición de los nodos baliza secundarios referenciando su plano al nodo más cercano por medio de la ley de cosenos (4).

$$\angle\theta = \cos^{-1} \sqrt{\frac{a^2 - b^2 - c^2}{-2a.c}} \quad (4)$$

Posteriormente los nodos que no son baliza pueden calcular sus posiciones tomando como referencia los nodos baliza primarios y secundarios usando la técnica de trilateración, cada nodo calcula su distancia a los nodos baliza por medio de los mensajes recibidos de localización que son enviados por los nodos baliza, si un nodo recibe menos de tres mensajes baliza no puede calcular su posición y deberá esperar a recibir más información, cada vez que un nodo calcula su posición este nodo puede ser tomado como referencia por otros nodos que aun no lo han logrado así que emite mensajes de broadcast de su posición y de esta manera se llegará a un descubrimiento progresivo de la posición de cada nodo; cada vez que un nodo calcula su posición debe resolver un algoritmo de un sistema de ecuaciones mostrado en la ecuación (5), y deberá enviar su posición al nodo Sink por un mensaje de unidifusión utilizando las trayectorias que han sido configuradas con el Algoritmo A3 por medio de la técnica Dv-Hop [1].

$$\begin{aligned}
(x_i - x_1)^2 + (y_i - y_1)^2 &= r_1^2 \\
(x_i - x_2)^2 + (y_i - y_2)^2 &= r_2^2 \\
(x_i - x_3)^2 + (y_i - y_3)^2 &= r_3^2
\end{aligned}
\tag{5}$$

En la figura 6 se observa el proceso resumido que se utilizó para este propósito.

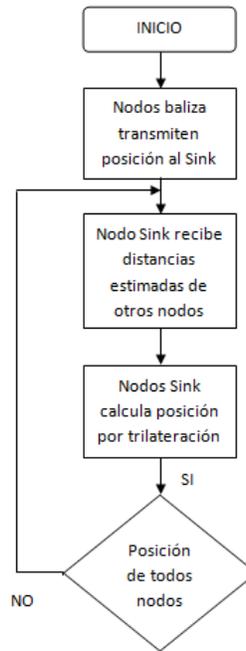


Figura 6: Diagrama de flujo de calculo de posición

El algoritmo se basa en el conocimiento que los nodos pueden tener acerca de la posición de los demás nodos, esta información para el apuntamiento es manejada y controlada por el nodo Sink. El nodo sink contiene una tabla en la que se encuentran las coordenadas de la posición y la dirección de la brújula de cada uno los nodos integrantes de la red; con esta tabla resuelve un algoritmo que utiliza la ecuación (6) y calcula cuales nodos se encuentran apuntando en dirección a otro nodo integrante de la red.

$$\angle\theta = \tan^{-1} \frac{Y_2 - Y_1}{X_2 - X_1}
\tag{6}$$

La ecuación (6) es una sencilla aplicación vectorial, en la tabla del nodo sink un puntero lógico se ubica en la posición del primer nodo en la tabla e inicia a recorrerla,

cada vez que avanza una posición en la tabla realiza la resta de la posición cartesiana del primer nodo con la posición de los demás nodos que muestra la tabla(método de vector resultante)halla el vector resultante entre la posición de los dos nodos, luego calcula el angulo de el vector resultante, este angulo resultante lo compara con la dirección que este indicando la brújula de cada nodo integrante en la red, si es igual quiere decir que esta apuntando a un nodo integrante de la red y el nodo sink emite un mensaje al nodo que esta apuntando y al nodo que está siendo apuntado

En el caso que un nodo se movilice o cambie su dirección el nodo emite un mensaje al nodo sink utilizando como ruta una trayectoria que se basa en la topología constituida, cuando el nodo sink recibe el nuevo mensaje vuelven a calcular el apuntamiento basado en la ecuación (6). En la figura 7 se observa el proceso resumido que se utilizó para este propósito.

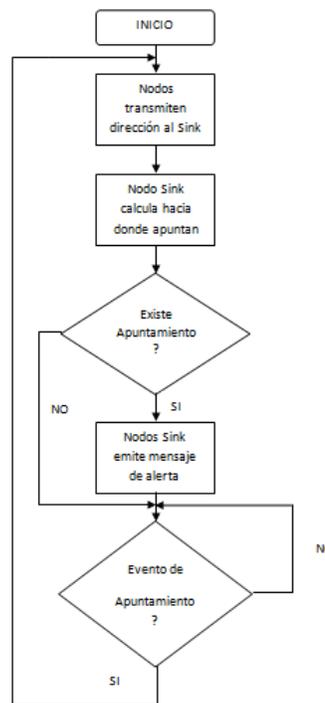


Figura 7: Diagrama de flujo de calculo de dirección

El aspecto del apuntamiento depende de la estimación de la dirección que cada nodo tiene de sus vecinos, en este proyecto se considera que el sensor cuenta con una brújula, así que si el nodo esta apuntando a un nodo en una dirección que coincide en 5 grados con la dirección de la brújula se le da aviso al nodo que tiene un apuntamiento y envía un mensaje al nodo sink. En este diseño se ha considerado que se obtendrá la dirección luego de obtener la localización.

#### 4.1.5. Algoritmo de Apuntamiento

El algoritmo de apuntamiento consiste en detectar si hay coincidencia entre la dirección de los nodos con la trayectoria de línea de vista directa sin obstáculos entre ellos, la orientación de cada nodo es indicada por un sensor que esta representado por una flecha en la figura 8, si en esa dirección se encuentra otro nodo en su trayectoria de línea de vista directa que esta representado por una línea punteada en azul en la figura 8 se dice que existe un apuntamiento. A partir de la información entregada por los sensores de los nodos y el error medio de distancia producido por el algoritmo de localización, se determina una zona de apuntamiento  $\Delta a$  en la que estarán comprendidas las trayectorias de la línea de vista directa como se muestra en la figura 8.

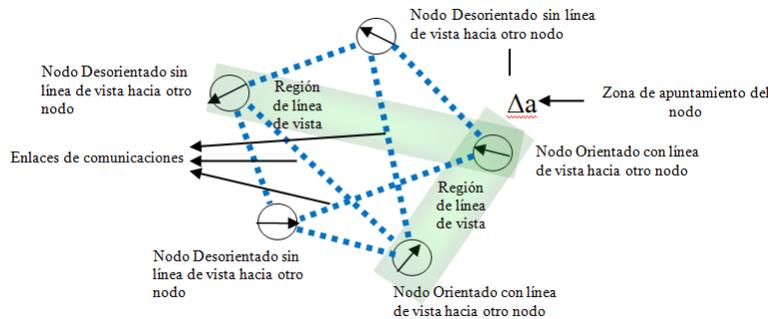


Figura 8: Escenario de apuntamiento

#### Mensajes de apuntamiento :

Los mensajes de apuntamiento son alarmas que se generan cuando un nodo se halla apuntado por otro nodo o cuando un nodo apunta a otro nodo.

- **Apuntamiento:** Cuando el sink detecta en su tabla de apuntamiento, este mensaje es enviado por el sink a un nodo que apunta a otro nodo.
- **Advertencia de apuntamiento:** Este mensaje es enviado por el sink al nodo que está siendo apuntado.

## 4.2. Pruebas de Software

Luego de seleccionar la plataforma OMNET [22] se identificó que cuenta con tres frameworks que se pueden utilizar para simular redes de sensores inalámbricas, los frameworks que se encontraron fueron instalados y con todos se realizaron pruebas durante

Imes, a continuación en la tablas se describen las ventajas y desventajas encontradas por el autor luego de las pruebas realizadas.

## Ventajas

Tabla 3: Ventajas de las plataformas

Característica	MiXiM	INET	NICTA Castalia
Capas del modelo OSI que opera	5	5	2
Capa física detallada	Si	No	No
Herencia y polimorfismo de otros Frameworks	Si	No	No
Soporte en grupos de estudio vía Web	Si	Si	Si
Existencia de información escrita suficiente	Si	Si	No
Constantemente actualizado	Si	Si	No

## Desventajas

Tabla 4: Desventajas de las plataformas

Característica	MiXiM	INET	NICTA Castalia
Requiere un alto nivel de conocimiento en C++	Si	Si	Si
Curva de aprendizaje lenta	No	Si	Si
Presenta Modo Gráfico	Si	Si	No

MiXiM fue seleccionado sobre INET y Castalia por que durante el mes de pruebas fue el framework que mostro resultados coherentes con lo esperado y se logró manejar con más facilidad que los otros dos, adicionalmente cuenta con una excelente información en línea en la que se muestran ejemplos de construcción de módulos clave en el funcionamiento integral del software. En opinión del autor se encontró que MiXiM:

1. Más modular, más fácil de utilizar el código
2. El código se pudo probar suficientemente y en las pruebas dio más confianza por que funcionaba como se esperaba.
3. Se encontraron errores que se podían solucionar más fácilmente que los otros dos.
4. Cuando se requirió construir bibliotecas se necesito menos esfuerzo para acostumbrarse al uso de pruebas,

## 4.3. Simulación

El método de simulación que utiliza OMNET es por medio de eventos discretos, en la realización de todas las simulaciones y para su posterior comprensión se mantuvo el

seguimiento a ciertas variables de conteo, variables de estado del sistema y variables de tiempo. Las principales variables consideradas fueron:

Variables de conteo: Contadores de Broadcast, Contadores de alarmas de apuntamientos Variables de estado del sistema: error de distancia, sensibilidad de los nodos, rssi de los nodos.

Variables de tiempo: Tiempo de duración de Broadcast, tiempo de triangulación, tiempo de convergencia. Los valores de estas variables se modificaron durante la simulación y fueron recopilados al finalizar o durante cada una de las corridas realizadas.

En esta etapa se utilizaron módulos de MiXiM de la capa de aplicación, capa red, capa MAC y de capa PHY para todos los Nodos integrantes de la WSN con el fin de evaluar si proporcionó datos precisos, se utilizaron los resultados para obtener el error de localización de cada nodo, el error de apuntamiento el tiempo de convergencia de la red, así como mediciones de rssi y sensibilidad de cada nodo, luego se comparó el valor entregado por las variables configuradas en MiXiM con el valor hallado analíticamente, para validar, si los resultados resultaron exactos.

Los módulos, clases, paquetes utilizados de MiXiM fueron los siguientes:

#### **Módulo WSN con 802.15.4**

El entorno gráfico de MiXiM permite de una manera sencilla crear un entorno de simulación basado en el estándar 802.15.4, presenta unos archivos de descripción de diferentes componentes de red que fue necesario utilizar y modificar en este trabajo, a continuación se describen brevemente cada uno de los componentes utilizados que son mostrados en la figura 9.

#### **Modulo CSMA 802.15.4:**

Modulo que no cuenta con beacon habilitado del protocolo CSMA, de acuerdo con la pruebas realizadas en otras simulaciones se obtienen resultados en puntualidad en tiempo real, soporta backoffs constantes, lineales y exponenciales, así como \* MAC ACKs.

**Appl:** Modulo que sirve de interfaz para la capa de aplicación, genera las señales necesarias del sensor considerado en este trabajo.

**Presl:** Modulo que sirve de interfaz para la capa de aplicación, en este trabajo no es utilizado.

**Sesl:** Modulo que sirve de interfaz para la capa de sesión, En este trabajo no es

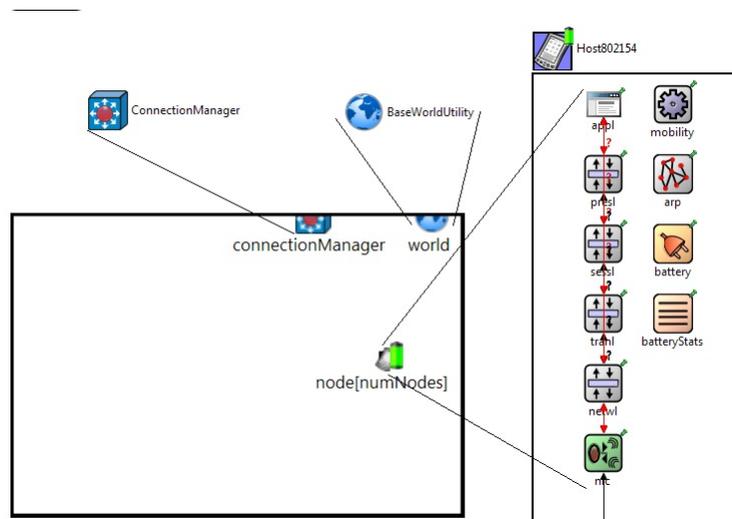


Figura 9: Escenario de configuración de inicio

utilizado.

**Tranl:** Modulo que sirve de interfaz para la capa de transporte, En este trabajo no es utilizado.

**Netwl:** Modulo que sirve de interfaz para la capa de red, junto con la clase Wiseroute crea un árbol de enrutamiento desde un punto central de la red.

**Nic:** Modulo que se configure para ser la interfaz entre capa física y el modulo de la capa Netwl.

**Mobility:** Módulo base de movilidad que otros módulos pueden heredar para ser configurado como estacionario o para cambiar su estado de movilidad en el tiempo de simulación. Es de utilidad para manejar algunas tareas comunes entre los módulos de movilidad, para actualizar el estado y la movilidad se debe programar la próxima actualización. Este modulo también proporciona la ubicación inicial aleatoriamente como un mecanismo por defecto.

**Arp:** Este modulo tiene una clase que tiene identificación de la capa de red así como la la identificación del módulo NIC y la dirección de la capa MAC.

**Battery:** Este modulo actualiza la capacidad residual (total actual de tensión \* Tiempo), publica el estado del agotamiento de la batería, y proporciona series de tiempo y la información resumen de estadísticas del módulo junto con el Battery Stats.

**Connection Manager:** Modulo base que realiza el cálculo de la distancia de in-

terferencia en base a la potencia de transmisión, recibe como parámetros la longitud de onda, el coeficiente de pérdida de propagación y un umbral mínimo de energía que podrá recibir cada nodo. Junto con el módulo central BaseConnectionManager coordina las conexiones entre todos los nodos, y se ocupa de la creación de enlaces dinámicos, adicionalmente BaseConnectionManager comunica periódicamente con el módulo ConnectionManagerAccess

**World Utilities:** Este modulo base determina el del área en la que los nodos se ubican, la magnitud está en (m).

#### 4.4. Configuración

La configuración inicial considera un modelo de canal del tipo SimplePathloss Model disponible en MiXiM como modelo de propagación, en este modelo utilizamos una variable de pérdida  $\alpha$  equivalente a 1.997, una frecuencia portadora de 2,4 GHz, en la tabla (5) se muestran los parámetros más importantes de la configuración inicial

Tabla 5: Parametros de configuración Inicial

Parámetro	Valor
Path Loss Coefficient	1.997
Path Loss Model	SimplePathLossModel
connectionManager.carrierFrequency	2,4 GHz
connectionManager.pMax	1.1 mW
connectionManager.sat	-91dBm
phy.sensitivity	-109,63 dBm
mobility.speed	uniform(0mps, 5 mps)
mobility.changeAngleBy	uniform(0deg, 5deg)
nic.phy.apuntamiento	20 deg
node[*].mobility.initialX	uniform (0m,150m)
node[*].mobility.initialY	uniform (0m,150m)

En este trabajo se propusieron unos objetivos a los que se dio cumplimiento debido a que con la simulación se obtuvieron resultados creíbles y lo más cercano posibles a la realidad, se compararon los modelos matemáticos propuestos con los resultados obtenidos, y se analizaron los resultados y determino el funcionamiento de la propuesta.

#### 4.4.1. Escenarios

Para lograr los resultados de la simulación que se requieren, se conformó un escenario en el que se consideraron nodos WSN ubicados con una distribución uniforme en un terreno cuadrado de 300m de lado, como se observa en la figura 10 los nodos se encuentran a una cierta distancia  $d$  entre ellos, esta distancia es aleatoria. Para cada uno de los escenarios se realizaron simulaciones independientes variando el número de nodos de cinco en cinco y también variando el rango del cono de apuntamiento, se estableció para los nodos un movimiento y una velocidad aleatorios, así mismo el cambio dirección de los nodos cambia aleatoriamente durante el transcurso de la simulación. Se debe aclarar que la velocidad de cambio de velocidad y de dirección depende de la aplicación a la que se requiera medir las métricas de desempeño.

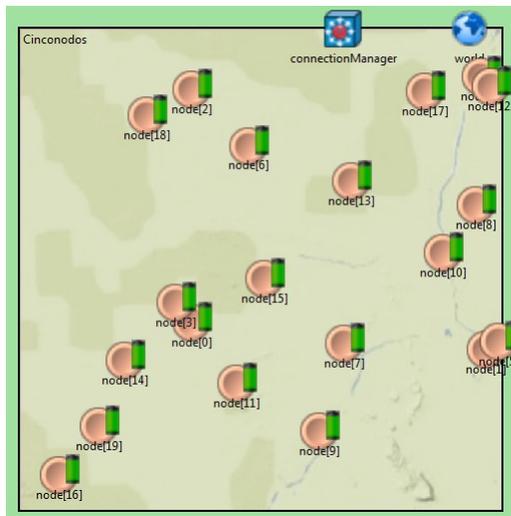


Figura 10: Escenario de simulación gráfica

#### 4.4.2. Implementación

Los nodos considerados en cada uno de los escenarios son Host802154, utilizan el módulo BurstApplLayer como capa de aplicación que genera las señales que se conforman con la información de una brújula electrónica, conectado en la capa inferior se utiliza el módulo Wiseroute como capa de red, con este módulo en esta capa se elabora el algoritmo de enrutamiento. Conectado en la capa inferior se encuentra el módulo CSMA802154 como capa MAC y PhyLayer como capa PHY. Por otra parte, se utiliza un elemento denominado Decider802154Narrow que permite decidir si la potencia de señal es suficiente para procesar la trama que viene del espacio libre.

Todas las capas seleccionadas son módulos básicos de MIXIM, y el nombre de

Host802154 proviene de un proceso automático de MiXiM. En la figura 11 se pueden ver las capas de los nodos utilizados en este trabajo. La elección de cada uno de los módulos y su código en c++ fue inducido por el propósito del experimento, se eligieron los módulos de la capa MAC y PHY porque tienen implementadas las normas del protocolo IEEE 802.15.4. En la capa de aplicación se requirió un generador de paquetes constante y con el módulo BurstApplLayer se logró este propósito, mientras que para la capa de red se eligió Wiseroute porque se requería realizar enrutamiento y no simplemente que sólo se propagara el paquete a la capa inferior.

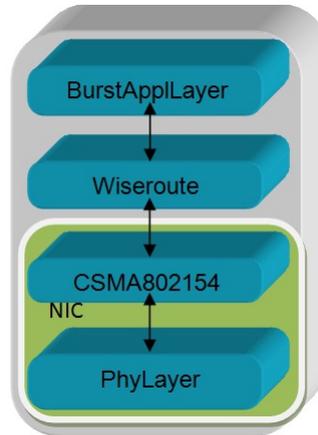


Figura 11: Escenario típico

## 4.5. Pruebas de calidad y desempeño del sistema

En esta sección se muestran los mecanismos y procedimientos de aceptación, y comprobación del desempeño que debe cumplir el sistema diseñado, y que son de utilidad para determinar si el sistema satisface los requerimientos y especificaciones planteadas. El modelo de red, el proceso de optimización y el algoritmo diseñado se evalúan y validan por medio de simulación en condiciones realistas. Los resultados obtenidos se analizan comparativamente, todos estos resultados de la simulación son comparados y analizados por separado; a continuación se determinan y describen los factores y características importantes relacionadas con el desempeño de la WSN.

### 4.5.1. Verificación y validación del modelo de simulación

Para verificar que el programa este libre de errores se siguen ciertas técnicas, la primera de ellas consiste en depurar por módulos o subrutinas y verificarlas por separado[15]. A continuación las técnicas para decidir que el modelo de simulación es válido o no se requiere realizar verificación y validación basado en diferentes técnicas como parte del

proceso de desarrollo del modelo[18], las técnicas escogidas para validar este modelo se analizarán a continuación:

**Animación:** El modo de funcionamiento del modelo está representado gráficamente como un esquema que se mueve a través tiempo. Por ejemplo, los movimientos de los nodos en el entorno y los mensajes transmitidos durante una simulación se mostrarán gráficamente.

**Validez de eventos:** Los eventos de generados se comparan con los del verdadero sistema para determinar si son similares. Un ejemplo de los eventos será el momento en que realiza la transmisión o el momento en que realiza un movimiento o un cambio de dirección un nodo.

**Valores fijos:** valores de uso (por ejemplo, las constantes) son utilizadas como variables internas o parámetros. Esto deberá permitir la verificación de los resultados del modelo contra los valores calculados.

**Análisis de sensibilidad:** Este técnica consiste en cambiar los valores de entrada y los parámetros internos de un modelo para determinar el comportamiento del modelo y su salida. la mismas relaciones deben ocurrir en el modelo como en el sistema real. Esos parámetros que son sensibles, es decir, que producen cambios significativos en el comportamiento del modelo o de salida, se deben precisar muy bien antes de utilizar el modelo.

#### 4.5.2. Robustez

En este proyecto se medirá la cantidad de nodos conectados mediante enlaces de comunicaciones y también se medirá la capacidad que tiene la red ante fallos. En este trabajo se aborda este problema y se proponen algunas técnicas para planear una WSN con una distribución uniforme.

#### 4.5.3. Tiempo de Convergencia

El período de convergencia se medirá teniendo en cuenta el tiempo en el que todos los nodos se encuentren conectados a la topología de la red y se realice el apuntamiento. De acuerdo a las técnicas utilizadas el desempeño se calculara teniendo en cuenta el enrutamiento y el envío de mensajes, en el enrutamiento encontraremos tablas que se crean dinámicamente mediante la obtención de información de los nodos. Los sensores de la red deben actualizarse constantemente a los cambios de topología en la red, rutas que pueden ser añadidas o eliminados, o las rutas pueden fallar debido a una interrupción

en la conexión.

Cuando una nueva conexión se añade o falla o existen cambios, las actualizaciones se envían por los nodos a través de la red que describen los cambios en la topología de la red, luego se ejecuta un algoritmo para recalcular las rutas y construir nuevas tablas de enrutamiento en función de la información de actualización. Después de cálculo, todas las tablas de enrutamiento llegan a una visión común de la red de topología. La convergencia de la red termina cuando se considere que todos los nodos están conectados y envían mensajes.

#### **4.5.4. Precisión y Porcentaje de error**

Cuando se integra el algoritmo de apuntamiento a las técnicas de localización se requiere conocer cuál es la precisión, para ello se considera un rango de apertura que se considera como el campo de visión que tiene el nodo hacia la dirección a la que está apuntando. Adicionalmente es imperativo tener la posibilidad de calcular cual será el error cuadrático medio que presenta el apuntamiento debido a que se utilizaran variables que propagan el error.

#### **4.5.5. Tiempo de Vida de la red**

En este trabajo se midió el tiempo de vida útil de una WSN, es una cuestión importante que se medirá en este proyecto. En este trabajo se aborda este problema y se sugiere una arquitectura de red jerárquica, similar a los escenarios realistas, donde los nodos con las fuentes de energía renovables (denotado como nodos primarios) llevan a cabo la mayoría de las tareas de entrega de mensajes, y los nodos equipados con baterías químicas convencionales (denotado como nodos secundarios) llevan a cabo menos tareas y por o tanto con menos demandas de comunicación. La cuestión clave en el diseño de esta arquitectura de red es el desarrollo de un nuevo marco de optimización para el cálculo de la asignación óptima de fuentes de energía renovables (asignación nodo primario) para maximizar la vida de la red, obteniendo el número mínimo de los suministros de energía y su asignación a cada nodo.

#### **4.5.6. Conectividad y cobertura**

Se considera la densidad de nodos y la potencia de transmisión para caracterizar la conectividad y la cobertura de la red, el escenario que se considera para la red presenta una distribución uniforme en un área limitada por el rango de transmisión.

## 4.6. Recolección de información

Omnet++ cuenta con la función de registro de eventos. Según los creadores de omnet su objetivo es ayudar en la comprensión de modelos complejos de simulación y para ayudar a implementar la simulación correctamente. Con el uso de esta herramienta se pudo examinar fácilmente cada detalle de la simulación hacia atrás y hacia adelante en términos de tiempo de simulación y paso a paso, fue útil en este trabajo porque se analizó el comportamiento en lugar de los resultados estadísticos. El archivo de registro de eventos se crea automáticamente durante la ejecución de una simulación al realizar la respectiva configuración en el archivo ini, el archivo resultante se puede ver con OMNeT++ IDE usando la Tabla de Secuencia y el registro de sucesos mostrados en la figura 12. Estas herramientas permiten examinar las relaciones de causalidad y proporcionan el filtrado basado en los tiempos de simulación, números de evento, los módulos y los mensajes.

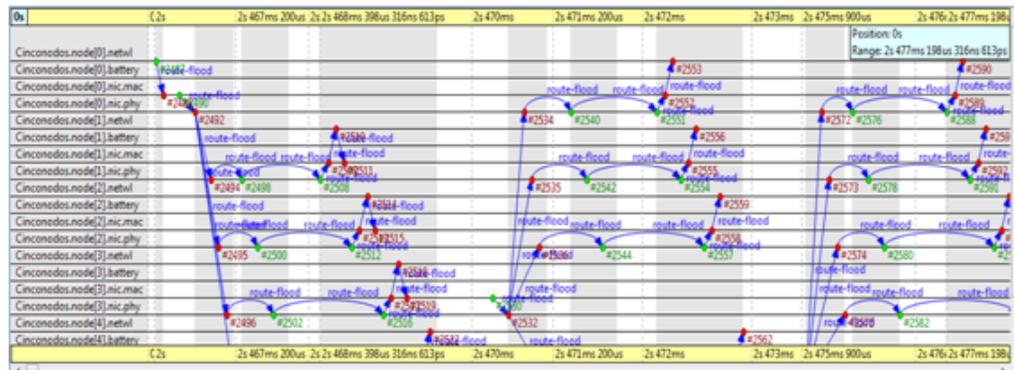


Figura 12: Salida en el dominio del tiempo

OMNeT++ permite registrar los resultados de la simulación, a través de vectores de salida y escalares de salida. Los vectores de salida son datos de series temporales. Los vectores de salida se utilizaron para calcular retrasos de extremo a extremo o tiempos de ida y vuelta de paquetes, cualquier cosa que es útil para obtener una visión completa de lo que sucedió en el modelo durante la ejecución de la simulación. Los escalares de salida se utilizaron como los resultados de resumen, computado durante la simulación y fueron guardados cuando la simulación se completó. Estos resultados fueron utilizados para construir el resumen estadístico compuesto por varios campos, tales como el recuento, media, desviación estándar, suma, mínimo, máximo, etc., y los datos del histograma opcionalmente. Los resultados fueron recogidos y registrados de dos maneras:

- 1. Basado en el mecanismo de señal, utilizando las estadísticas declaradas.
- 2. Directamente desde código C++, utilizando la biblioteca de simulación.

## 5. Análisis de resultados

### 5.1. Sensibilidad promedio

En este trabajo se utilizó un algoritmo de control de topología en el que todos los nodos mantienen la misma potencia de transmisión y cada uno de los nodos tendrá una sensibilidad diferente. La sensibilidad de cada nodo es el mínimo RSSI requerido para mantener conexión con el nodo más cercano, de esta manera los nodos administran su sensibilidad de recepción. A continuación se muestra una secuencia de conformación del árbol y la sensibilidad adquirida de cada uno de los nodos.

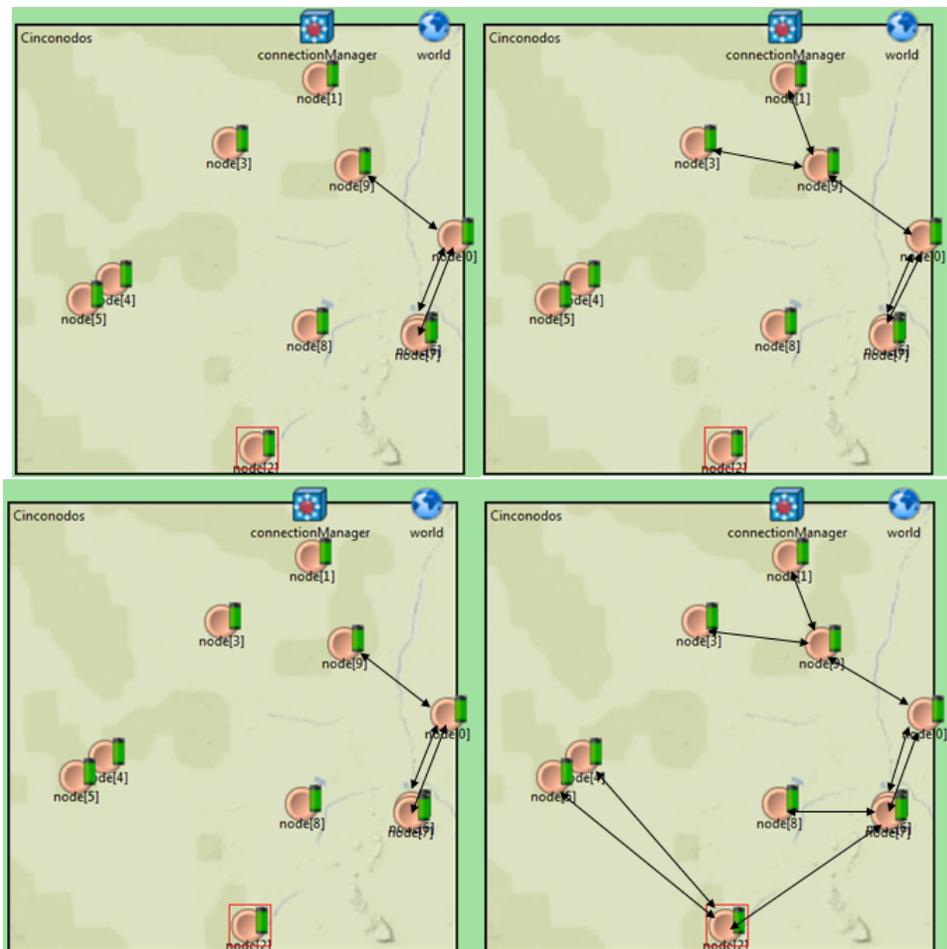


Figura 13: Proceso de construcción del árbol

En la fig 13 se evidencian los pasos del proceso que realiza el algoritmo de control de Topología, en la parte superior derecha se observa al inicio que el nodo 0 emite una señal y es recibida por tres nodos que se encuentran dentro del rango de cobertura,

después los nodos emiten broadcast y es al nodo 9 que los nodos 1 y 2 se conectan, en seguida de varios broadcast el nodo 7 conecta a los nodos 2 y 8, luego como los nodos 4 y 5 se encuentran fuera de cualquiera de los rangos entonces configuran su sensibilidad para finalmente conectarse al nodo 2.

La sensibilidad promedio se obtiene al hallar la relación entre la suma de la sensibilidad asignada a cada nodo con el número de nodos, como se muestra en la ecuación.

$$S_{prom} = \frac{\sum_{i=0}^N(S)}{N} \quad (7)$$

donde  $S_{prom}$  es la sensibilidad promedio de una ejecución de la WSN, n es el número de nodos y S es la sensibilidad de cada nodo que conforma la WSN. Para obtener un valor de sensibilidad media se realizaron 20 simulaciones independientes con variables aleatorias diferentes, con el fin de estimar una sensibilidad como se muestra en la ecuación (7):

$$S_m = \frac{\sum_{i=0}^N(S_{prom})}{N} \quad (8)$$

donde  $S_m$  es la sensibilidad media de la WSN,  $S_{prom}$  es la sensibilidad promedio de una ejecución y N es el número de ejecuciones que se realizaron, se muestran los resultados en la figura 14.

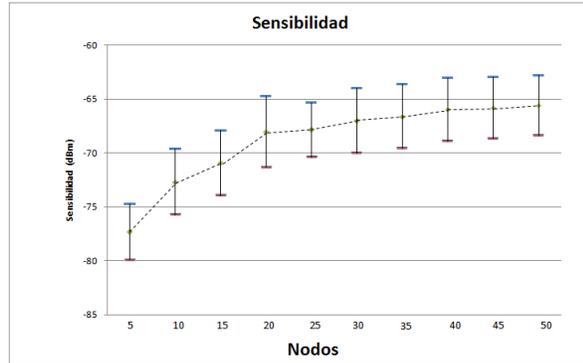


Figura 14: Sensibilidad

En la figura 14 se observa que si la densidad de nodos por unidad de área aumenta, la sensibilidad requerida por los nodos de la red requiere de un menor valor de energía, con una tendencia a estabilizarse si la densidad de nodos aumenta. Con el algoritmo de control de topología propuesto se garantiza la conexión del 100% de los nodos con la sensibilidad especificada por el fabricante del TI2402 y la reducción en el consumo de energía en un 2.4% al compararlo con el algoritmo A3.

En la siguiente figura 15 se realiza la comparación de sensibilidad requerida por nodo entre el algoritmo A3 y el algoritmo propuesto en este proyecto, en ella se observa que el algoritmo APA presenta menor energía promedio por nodo, adicionalmente que la desviación estándar de APA es menor que la desviación estándar del A3, lo cual evidencia que el algoritmo APA tiene un valor promedio más cercano a un consumo menor de energía.

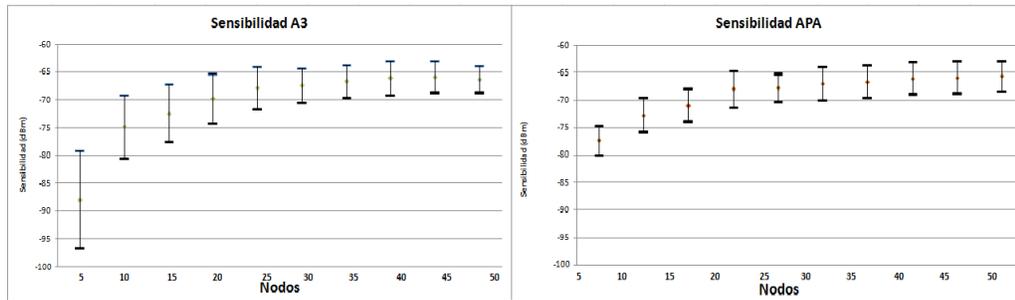


Figura 15: Desviación estándar entre A3 y APA

Con la intención de comprobar la relación de los valores promedio de sensibilidad entre APA vs A3 en la figura 16 se muestran dichos valores para un rango entre 5 a 50 nodos.

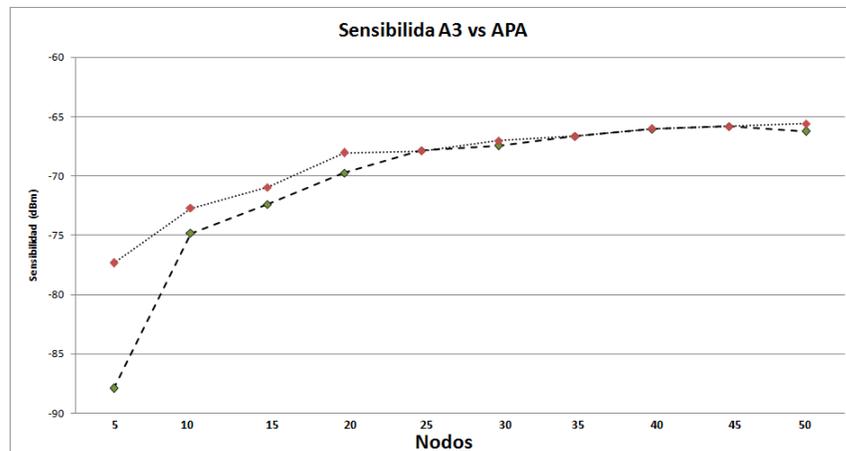


Figura 16: Comparación entre A3 y APA

## 5.2. Tiempo de convergencia

El tiempo de convergencia en este trabajo se refiere al intervalo de tiempo que existe entre el inicio y el momento en que todos los nodos de la WSN se encuentran conectados,

el inicio en  $t = 0$  se supuso que todos los nodos se encuentran encendidos. El tiempo de convergencia de la WSN se obtiene al hallar la diferencia entre el tiempo en el que el último nodo calcula el apuntamiento y el momento de inicio de la red, tal como se muestra en la ecuación.

$$t_{conv} = t_{rf} - t_{in} \quad (9)$$

Donde  $t_{conv}$  es el tiempo de convergencia de una ejecución de la WSN,  $t_{rf}$  es el tiempo en el que el último nodo apunta y  $t_{in}$  es el tiempo de inicio de la WSN.

Para obtener un valor de tiempo de convergencia medio se realizaron 20 simulaciones independientes con variables aleatorias diferentes, con el fin estimar una tiempo de convergencia como se muestra en la ecuación:

$$T_{conv} = \frac{\sum_{i=0}^N(t_{conv})}{N} \quad (10)$$

donde  $T_{con}$  es la media del tiempo de convergencia de la WSN,  $t_{conv}$  es el tiempo de convergencia una ejecución y N es el número de ejecuciones que se realizaron.

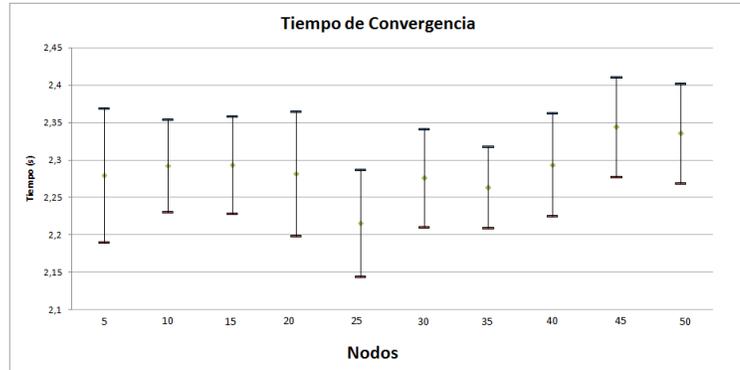


Figura 17: Tiempo de convergencia típico

En la figura 17 se observa que el tiempo de convergencia no supera los 2,4 s y presenta en promedio  $T_{con}=2,28s$ , cabe resaltar que así la densidad de nodos por unidad de área aumente, el tiempo de convergencia depende de la demora de los nodos en recibir el la trama de broadcast. El tiempo de broadcast es menor en redes más densas debido a que la señal se transmite entre nodos en un menor tiempo, y esa es la razón por la que el tiempo de convergencia presenta un valor promedio para cualquier número de nodos.

### 5.3. Error de localización

El error de localización en este trabajo describe la diferencia en distancia que se que se presenta entre los nodos cuya ubicación es entregada por el sistema y la ubicación entregada por el algoritmo de ubicación utilizado.

El error de localización de cada uno de los nodos de la WSN se obtiene al hallar la diferencia entre las coordenadas X y Y de la ubicación mostrada realmente y las coordenadas X y Y entregadas por el algoritmo de localización utilizado.

$$e_d = \sqrt{(X_i - X_d)^2 + (Y_i - Y_d)^2} \quad (11)$$

donde  $e_d$  es el error en distancia de cada uno de los nodos de la WSN,  $X_i$  es la coordenada en X conocida  $X_d$  es la coordenada en X hallada con el algoritmo de localización,  $Y_i$  es la coordenada en Y conocida y  $Y_d$  es la coordenada en Y hallada con el algoritmo de localización.

Para obtener el error promedio de distancia de todos los nodos de la red luego de una ejecución se suman todos los errores encontrados y se halla la relación con respecto al número de nodos como se muestra en:

$$e_{prom} = \frac{\sum_{i=0}^N (e_d)}{n} \quad (12)$$

Para obtener un error de distancia medio se realizaron 20 simulaciones independientes con variables aleatorias diferentes, todas las ejecuciones con la misma distribución, para las 20 ejecuciones se obtuvo una media como se muestra en la figura 18.

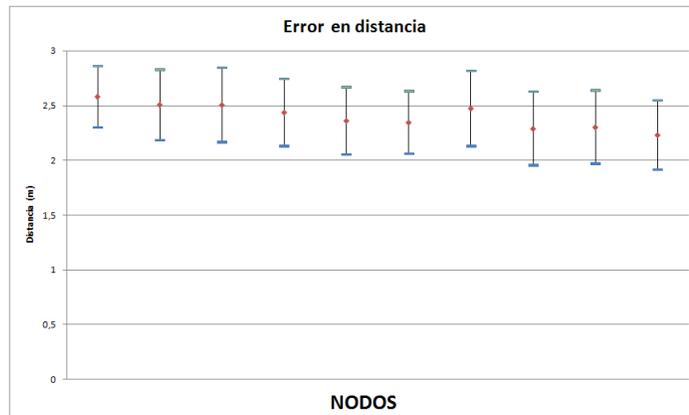


Figura 18: Error en distancia

En la figura 18 es evidente que se logra llegar a tener un error en distancia en promedio menor a 2.6 m para cualquier densidad de nodos. se observa que si la densidad de nodos por unidad de area aumenta, el error promedio de localización por nodo no se reduce, así mismo en las redes con escaso número de nodos la densidad es baja y se puede presentar un error que no se encuentra en el promedio y no existe más error que en las redes con una densidad alta, es evidente que el error cuadrático medio es aproximadamente constante por lo tanto se deduce que el algoritmo tendrá la misma probabilidad de error.

## 5.4. Efectividad de apuntamiento

En este trabajo la efectividad de apuntamiento se refiere al porcentaje de detecciones de apuntamientos dentro del rango del área de apertura que existe en la línea de vista entre nodos. el calculo se realiza teniendo en cuenta la relación de apuntamientos que pueden existir realmente con los apuntamientos que pueden ser predecibles en la simulación, para ello se utilizo una matriz de confusión que contiene información de los apuntamientos reales y los apuntamientos predecibles, la tabla (6) muestra como se realizó la matriz para clasificar los apuntamientos.

Tabla 6: Matriz de confusión de apuntamiento

	Apuntamientos Reales	
Apuntamientos Predecibles	Verdadero	Falso
Verdadero	a	b
Falso	c	d

Las entradas de la matriz tiene el siguiente significado:

- a es el número de apuntamientos reales que fueron predecibles en la simulación.
- b es el número de apuntamientos no existentes que fueron predecibles como apuntamientos en la simulación.
- c es el número de apuntamientos reales que no fueron predecibles en la simulación.
- d es el número de apuntamientos no existentes que fueron predecibles en la simulación.

La exactitud de apuntamiento se obtiene de la relación entre las predicciones que si corresponden con respecto a todas las predicciones tal como se muestra a continuación.

$$AC = \frac{(a + d)}{(a + b + c + d)} \quad (13)$$

En la tabla (7) se muestra a manera de ejemplo uno de los resultados obtenidos en una de las ejecuciones, de acuerdo con la ecuación 14 el resultado de efectividad equivale a 97.8 %

Tabla 7: Matriz elaborada de apuntamiento

Apuntamientos Simulados	Apuntamientos Reales	
	Verdadero	Falso
Verdadero	188	366
Falso	3293	163103

Para obtener un valor de exactitud media se realizaron 20 simulaciones independientes con variables aleatorias diferentes, con el fin de estimar una exactitud cómo se muestra en la ecuación:

$$ac = \frac{\sum_{i=0}^N (AC)}{N} \quad (14)$$

donde ac es la exactitud media de la WSN, AC es la exactitud de apuntamiento de una ejecución y N es el número de ejecuciones que se realizaron.

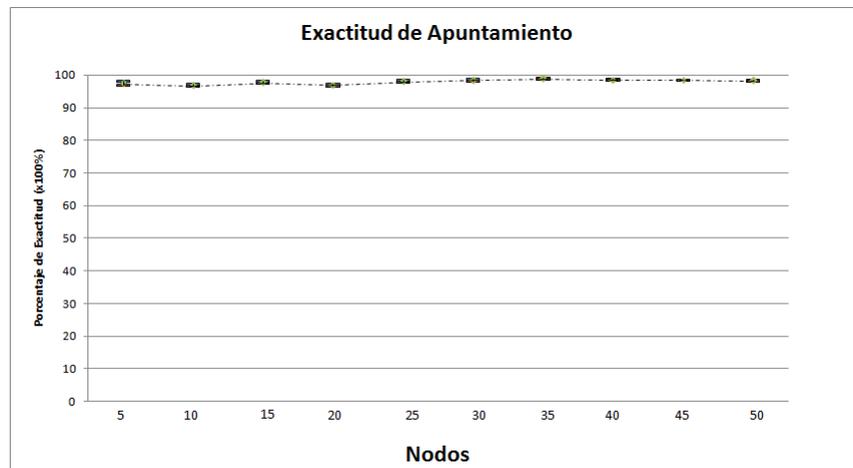


Figura 19: Exactitud

En la figura 19 se observa que la efectividad del apuntamiento se encuentra entre el 96 % y 99 % con un valor promedio de 97,9 % para redes conformadas entre 5 y 50 nodos, se debe anotar que si la densidad de nodos por unidad de área aumenta, el error promedio de apuntamiento por nodo se reduce. Este es un resultado esperado ya que la distancia entre nodos es menor y por lo tanto el rango de cobertura de apuntamiento se reduce y la precisión mejora.

## 6. Conclusiones

En este trabajo se ha tratado de interpretar con sumo cuidado la relación existente entre los resultados simulados obtenidos en MIXIM y los resultados de modelos analíticos existentes, en la simulación de diferentes escenarios se han teniendo en cuenta principalmente la variación del movimiento y la dirección aleatoria de cada uno de los nodos en la WSN. El interés principal del trabajo trató de analizar el apuntamiento y evaluar su exactitud con respecto a la predicción de un patrón de posición generada por un algoritmo que opera con la ubicación conocida de un mínimo número de dispositivos inalámbricos.

Para lograr los resultados más reales posibles se propuso una extensión en C++ con el fin de simular de una manera más realista la transferencia de información entre los dispositivos móviles cuando se encuentran en diferentes escenarios; con este mecanismo también se realiza una propuesta adicional por así decirlo, que consiste en variar parámetros que normalmente en MiXiM no se pueden modificar, mediante la implementación de un mapeo personalizado en lugar del utilizado por MIXIM. Estas nuevas definiciones jugaron un papel importante ya que facilitaron el manejo del Software.

Por otra parte se puede concluir que los resultados obtenidos en la simulación en MiXiM siguen un patrón similar al esperado, lo anterior se logró utilizando el módulo 802.15.4, a los resultados obtenidos en la simulación se les realizó un análisis estadístico básico en el que fueron evaluados diferentes tiempos de la red, el error en distancia y el error en exactitud ya son los parámetros de desempeño que más se distinguen. Se puede por lo tanto afirmar que los resultados de la simulación en MiXiM siguen la predicción de los modelos analíticos.

### **Temas que no se cubrieron**

El propósito de este trabajo fue evaluar en MiXiM los errores en exactitud que se pueden tener con diferentes aplicaciones que sean implementadas en escenarios reales. Por lo tanto no se trataron aspectos que tienen que ver con la capa física como el efecto de las colisiones o el manejo de errores en las tramas, en cuanto a la capa mac y de red no se trataron aspectos que tienen que ver con algoritmos de enrutamiento, adicionalmente no se trato el tiempo que debe tener cada retransmision de broadcast y de wiseroute por que son tiempos especificos que dependen de la aplicación.

### **Trabajos Futuros**

En el caso de trabajos futuros, este proyecto es un excelente punto de partida para trabajar en un algoritmo de apuntamiento adaptable o en un algoritmo de localización en el que se utilicen menos de tres nodos a los cuales se les conoce la posición para

realizar la localización del resto de los nodos en cualquier escenario. Por otra parte se pueden plantear trabajos futuros que expliquen los tiempos requeridos para que la red reenvie mensajes de acuerdo con la aplicación.

## Referencias

- [1] Isaac Amundson and Xenofon D. Koutsoukos. A survey on localization for mobile wireless sensor networks. In *Proceedings of the 2Nd International Conference on Mobile Entity Localization and Tracking in GPS-less Environments, MELT'09*, pages 235–254, Berlin, Heidelberg, 2009. Springer-Verlag.
- [2] Andrea Zanella Michele Zorzi. Andrea Bardella, Nicola Bui. An experimental study on ieee 802.15.4 multichannel transmission to improve rssi based service performance. *Real-World Wireless Sensor Networks*, pages 154–161, 2010.
- [3] Antonio A. F. Loureiro. Azzedine Boukerche; Horacio A. B. F. Oliveira, Eduardo F. Nakamura. Localization systems for wireless sensor networks. *IEEE Wireless Communications*, 14:6–12, 2007.
- [4] Romano Fantacci, Francesco Chiti, and Daniele Tarchi. An overview on wireless sensor networks technology. *Transport*, 12:24.
- [5] Carles Gomez and Josep Paradells. Wireless home automation networks: A survey of architectures and technologies. *Consumer Communications and Networking.*, pages 92–101, 2010.
- [6] Lillykutty Jacob Gopakumar Aloor. Distributed wireless sensor network localization using stochastic proximity embedding. *Computer Communications*, 33:745–755, 2010.
- [7] Carlos F. Garcia Hernandez. Wireless sensor networks and applications: a survey. *International Journal of Computer Science and Network Security*, 7:264–273, 2007.
- [8] Pat Banerjee Jing Liu. Hui Liu, Houshang Darabi. Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems Man and Cybernetics.*, 36:1067–1080, 2007.
- [9] Niels Reijers Koen Langendoen. Distributed localization in wireless sensor networks: a quantitative comparison. *Computer Networks*, 43:499–518, 2003.
- [10] Laurent El Ghaoui Lance Doherty, Kristofer S. J. Pister. Convex position estimation in wireless sensor networks. In *Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, 2001.
- [11] Jing Lu, Dianjie Lu, and XiaoXia Huang. Channel model for wireless sensor networks in forest scenario. In *Informatics in Control, Automation and Robotics (CAR), 2010 2nd International Asia Conference on*, volume 2, pages 476–479, 2010.

- [12] Matt Perkins Neiyer S. Correal Neal Patwari, Alfred O. Hero and Robert J. O Dea. Relative location estimation in wireless sensor networks. *IEEE Transactions on signal processing*, 51:2137–2148, 2003.
- [13] N. Patwari, J.N. Ash, S. Kyperountas, A.O. Hero, R.L. Moses, and N.S. Correal. Locating the nodes: cooperative localization in wireless sensor networks. *Signal Processing Magazine, IEEE*, 22(4):54–69, July 2005.
- [14] Chih Wei Yi Peng Jun Wan. Coverage by randomly deployed wireless sensor networks. *IEEE Transactions on Information Theory*, 52:2658–2669, 2006.
- [15] Sheldon M. Ross. *Simulation*. University of Southern California, 2006.
- [16] Paolo Santi. Topology control in wireless ad hoc and sensor networks. *ACM Comput. Surv.*, 37(2):164–194, June 2005.
- [17] Paolo Santi. *Topology Control in Wireless Ad Hoc and Sensor Networks*. WILEY, 2006.
- [18] Robert G. Sargent. Verification and validation of simulation models. *WSC '05 Proceedings of the 37th conference on Winter simulation*, pages 130–143, 2005.
- [19] Luca Schenato Damiano Varagnolo Saverio Bolognani, Simone del Favero. Consensus based distributed sensor calibration and least square parameter identification in wsns. *International Journal of Robust and Nonlinear Control*, 20:176–193, 2012.
- [20] Rahul P. Sawant, Qilian Liang, Dan O. Popa, and F.L. Lewis. Experimental path loss models for wireless sensor networks. In *Military Communications Conference, 2007. MILCOM 2007. IEEE*, pages 1–7, Oct 2007.
- [21] P.M. Wightman and M.A. Labrador. A3: A topology construction algorithm for wireless sensor networks. In *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, pages 1–6, Nov 2008.
- [22] Xiaodong Xian, Weiren Shi, and He Huang. Comparison of omnet++ and other simulator for wsn simulation. In *Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference on*, pages 1439–1443, June 2008.
- [23] Bin Xiao, Hekang Chen, and Shuigeng Zhou. Distributed localization using a moving beacon in wireless sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 19(5):587–600, May 2008.

## Anexo

En este aparte se muestra un aspecto que a consideración del autor es muy importante, ya que en la investigación realizada no se encontró evidencia el uso de este tipo de soluciones por parte de los usuarios de MiXiM. La plataforma MiXiM no permite heredar parámetros y variables entre clases que no sean friend como se observa en la figura 19.

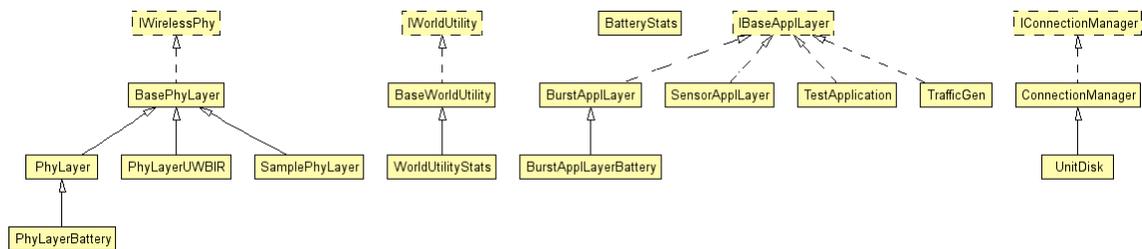


Figura 20: Ejemplo de herencia en MiXiM

En la figura 19 se muestra la con una línea continua la herencia entre las clase de MiXiM, este modelo muestra que la clase `BasePhyLayer` no puede heredar ni utilizar los parámetros de la clase `ConnectionManager`, para ello se planteó el siguiente código:

```
std::ofstream outputFile;
    char filename[100]="D:/omnetpp-
4.4/samples/cinconodos/Apuntamiento/angulonodo";
    char nombrenodo[30];
    itoa(nodo,nombrenodo,10);
    strcat(filename,nombrenodo);
    strcat(filename, ".txt");
    outputFile.open(filename);
    outputFile<< angulo;
    outputFile.close();

char filename[100]="D:/omnetpp-
4.4/samples/cinconodos/Apuntamiento/angulonodo";
    char nodoaux[30];
    char linea[30];
    itoa(pos,nodoaux,10); // convierte la fila de entero a
caracter en base que se quiera, para poder formar el nombre
    strcat(filename,nodoaux); //concatena fila a nodo
    strcat(filename, ".txt"); //asigna el nombre al archivo
como txt
    std::ifstream inputFile(filename); // abre el archive
```

```

char filename[100]="D:/omnetpp-
4.4/samples/cinconodos/VariableSalida/ultimaposicionxnodo";
    char nodoaux[30];
    char line[30];
    itoa(posicion,nodoaux,10); // convierte la fila de entero
a caracter en base que se quiera, para poder formar el nombre
    strcat(filename,nodoaux); //concatena fila a nodo
    strcat(filename, ".txt"); //asigna el nombre al archivo
como txt
    std::ifstream inputFile(filename); // abre el archivo
char filename1[100]="D:/omnetpp-
4.4/samples/cinconodos/VariableSalida/ultimaposicionynodo";
    char nodoaux1[30];
    char line1[30];
    itoa(posicion,nodoaux1,10); // convierte la fila de
entero a caracter en base que se quiera, para poder formar el nombre
    strcat(filename1,nodoaux1); //concatena fila a nodo
    strcat(filename1, ".txt"); //asigna el nombre al
archivo como txt
    std::ifstream  inputFile1(filename1); // abre el
archive

```

La solución consiste en manipular los parámetros actuales entregado por una clase, extraerlos al archivo correspondiente y luego ingresarlos a la clase que no es friend, tal como se muestra en el código mostrado en esta página.

```

char filename[100]="D:/omnetpp-
4.4/samples/cinconodos/VariableSalida/ultimaposicionxnodo";
    char namenodo[30];
    itoa(index,namenodo,10);
    strcat(filename,namenodo);
    strcat(filename, ".txt");
    std::ofstream outputFile;
    outputFile.open(filename);
    outputFile<< lastPosition.x;
    outputFile.close();
    //*****Grabando posiciones de los nodos en y
*****+
char filename1[100]="D:/omnetpp-
4.4/samples/cinconodos/VariableSalida/ultimaposicionynodo";
    char namenodo1[30];
    itoa(index,namenodo1,10);
    strcat(filename1,namenodo1);
    strcat(filename1, ".txt");
    std::ofstream outputFile1;
    outputFile1.open(filename1);
    outputFile1<< lastPosition.y;
    outputFile1.close();

```