



UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

La Universidad Católica de Loja

ÁREA TÉCNICA

TÍTULO DE INGENIERO EN ELECTRÓNICA Y

TELECOMUNICACIONES

Implementación de protocolos de retransmisión de paquetes para redes

FANET

TRABAJO DE TITULACIÓN.

AUTOR: Sánchez Moreno, Andrés Darío

DIRECTORA: Ludeña González, Patricia Jeanneth, Mgtr.

LOJA - ECUADOR

2020



Esta versión digital, ha sido acreditada bajo la licencia Creative Commons 4.0, CC BY-NC-SA: Reconocimiento-No comercial-Compartir igual; la cual permite copiar, distribuir y comunicar públicamente la obra, mientras se reconozca la autoría original, no se utilice con fines comerciales y se permiten obras derivadas, siempre que mantenga la misma licencia al ser divulgada. <http://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>

2020

APROBACIÓN DE LA DIRECTORA DEL TRABAJO DE TITULACIÓN

Magister.

Patricia Jeanneth Ludeña González.

DOCENTE DE LA TITULACIÓN

De mi consideración:

El presente trabajo de titulación **“Implementación de protocolos de retransmisión de paquetes para redes FANET”** realizado por Sánchez Moreno Andrés Darío, ha sido orientado y revisado durante su ejecución, por cuanto se aprueba la presentación del mismo.

Loja, enero de 2020

f)

DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS

“Yo *Sánchez Moreno Andrés Darío* declaro ser autor del presente trabajo de titulación: **Implementación de protocolos de retransmisión de paquetes para redes FANET**, de la Titulación de *Electrónica y Telecomunicaciones*, siendo *Patricia Jeanneth Ludeña González* directora del presente trabajo; y eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones legales. Además certifico que las ideas, conceptos, procedimientos y resultados vertidos en el presente trabajo investigativo, son de mi exclusiva responsabilidad.

Adicionalmente declaro conocer y aceptar la disposición del Art. 88 del Estatuto Orgánico de la Universidad Técnica Particular de Loja que en su parte pertinente textualmente dice: “Forman parte del patrimonio de la Universidad la propiedad intelectual de investigaciones, trabajos científicos o técnicos y tesis de grado o trabajos de titulación que se realicen con el apoyo financiero, académico o institucional (operativo) de la Universidad”

f.

Autor: Sánchez Moreno Andrés Darío

Cédula: 1104350820

DEDICATORIA

A mi madre, por todo el apoyo que me ha brindado para conseguir esta meta.

Andres Dario Sánchez Moreno

AGRADECIMIENTO

A mi directora de tesis, Mgtr. Patricia Ludeña, por su guía, apoyo y ayuda durante el desarrollo del presente trabajo. De igual manera a la ingeniera Katty Rohoden y al ingeniero Francisco Sandoval.

A mis amigos, por su paciencia, ayuda y apoyo durante todo nuestro trayecto universitario.

Andres Dario Sánchez Moreno

ÍNDICE DE CONTENIDOS

APROBACIÓN DE LA DIRECTORA DEL TRABAJO DE TITULACIÓN	ii
DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS	iii
DEDICATORIA	iv
AGRADECIMIENTO	v
ÍNDICE DE CONTENIDOS	viii
LISTA DE FIGURAS	x
LISTA DE TABLAS	xi
RESUMEN	1
ABSTRACT	2
LISTA DE ABREVIATURAS	3
INTRODUCCIÓN	4
1 LÍNEA BASE	6
1.1 Redes Ad-Hoc	7
1.1.1 Redes MANET	7
1.1.2 Redes VANET	8
1.1.3 Redes FANET	9
1.2 Broadcast en redes FANET	11
1.3 Broadcast Storm Problem (BSP)	12
1.4 Mecanismos para reducir el BSP.....	13
1.4.1 Esquema probabilístico	13
1.4.2 Esquema basado en contador	14
1.4.3 Esquema basado en distancia	15
1.4.4 Esquema basado en locación	15

1.4.5	Esquema basado en clusters	16
2	METODOLOGÍA	18
2.1	Selección de esquemas básicos	19
2.2	Selección de algoritmos híbridos.....	21
2.2.1	Algoritmo basado en contador con probabilidad arreglada	21
2.2.2	Algoritmo basado en contador con probabilidad dinámica.....	21
2.2.3	Retransmisión Vehicular Distribuida (<i>DV-Cast</i>).....	23
2.2.4	Difusión Sencilla y Robusta (<i>SRD</i>)	25
2.3	Selección del modelo de movilidad.....	26
2.3.1	Modelos de movilidad puramente aleatorios	27
2.3.2	Modelos de movilidad dependientes del tiempo	27
2.3.3	Modelos de movilidad de rutas planificadas	28
2.3.4	Modelos de movilidad grupales	29
2.4	Selección de métricas de evaluación.....	30
3	RESULTADOS	33
3.1	Implementación de esquemas básicos.....	34
3.1.1	Resultados para el esquema probabilístico	35
3.1.2	Resultados para el esquema basado en contador.....	37
3.1.3	Resultados para el esquema basado en distancia.....	38
3.2	Implementación de algoritmos híbridos	40
3.2.1	Implementación del algoritmo basado en esquema probabilístico y contador	41
3.2.2	Implementación del algoritmo <i>DV-Cast</i>	45
4	ANÁLISIS DE RESULTADOS	49
4.1	Comparación de esquemas básicos	50
4.2	Comparación de algoritmos híbridos	51
4.2.1	Cálculo del ahorro de potencia	54
	CONCLUSIONES	57
	RECOMENDACIONES	58

BIBLIOGRAFÍA	61
ANEXOS	62
Anexo A DATASHEET DRON INSPIRE 2	63
Anexo B DATASHEET SISTEMA DE TRANSMISIONLIGHTBRIDGE 2	64
Anexo C CÓDIGO PRINCIPAL PARA LA SIMULACIÓN DE LA RED	65
Anexo D CÓDIGO PARA LA RETRANSMISIÓN DELOS PAQUETES	70
Anexo E CÓDIGO ALGORITMO RANDOM WALK	72

LISTA DE FIGURAS

1.1	Redes <i>Ad-Hoc</i>	7
1.2	Red MANET.	8
1.3	Red VANET.	8
1.4	Red FANET.	9
1.5	<i>Broadcast</i> en redes FANET	12
1.6	Algoritmo del esquema probabilístico	14
1.7	Algoritmo del esquema basado en contador	14
1.8	Algoritmo del esquema basado en distancia.....	15
1.9	Algoritmo del esquema basado en locación.....	16
1.10	Algoritmo del esquema basado en <i>clusters</i>	17
2.1	Algoritmo del esquema basado en contador con probabilidad arreglada.	22
2.2	Algoritmo del esquema basado en contador con probabilidad dinámica.	23
2.3	Algoritmo <i>DV-Cast</i> con probabilidad arbitraria.....	24
2.4	Algoritmo <i>DV-Cast</i> con probabilidad dinámica	25
2.5	Algoritmo <i>SRD</i>	26
3.1	Rendimiento del esquema probabilístico	36
3.2	Rendimiento del esquema basado en contador	37
3.3	Rendimiento del esquema basado en distancia.....	39
3.4	Rendimiento del esquema basado en distancia.....	40
3.5	Rendimiento del algoritmo con probabilidad arreglada.	42
3.6	Rendimiento del algoritmo con probabilidad arreglada.	43
3.7	Rendimiento del algoritmo con probabilidad dinámica.	44
3.8	Rendimiento del algoritmo <i>DV-Cast</i>	46
3.9	Rendimiento del algoritmo <i>DV-Cast</i>	47
3.10	Rendimiento del algoritmo <i>DV-Cast</i> dinámico	48
4.1	Sobrecarga de la red con el método de Inundación.	51
4.2	Sobrecarga de la red aplicando esquema probabilístico	51
4.3	Sobrecarga de la red con esquema basado de contador	52
4.4	Sobrecarga de la red con esquema de distancia.	52
4.5	Sobrecarga con algoritmo de probabilidad dinámica.	53
4.6	Sobrecarga de la red con algoritmo <i>DV-Cast</i> dinámico	54
4.7	División del escenario.....	55
4.8	Ahorro de potencia de los algoritmos híbridos	55

LISTA DE TABLAS

1.1	Características redes <i>Ad-Hoc Ad-Hoc</i>	11
2.1	Algoritmos Híbridos	26
2.2	Métricas evaluadas	30
3.1	Características computador	34
3.2	Parámetros de Simulación	35
3.3	Escenarios de Simulación	35
4.1	Consumo de potencia de la red	56
A.1	Consumo de potencia de la red	63
B.1	Consumo de potencia de la red	64

RESUMEN

En el presente trabajo de titulación se analiza el rendimiento de los algoritmos híbridos de retransmisión de paquetes para mitigar el problema del BSP (*Broadcast Storm Problem*) en redes FANET (*Flying Ad-Hoc Network*), y el ahorro de potencia que se consigue al implementar estos algoritmos. La implementación de la red y los algoritmos híbridos se la realizó en una simulación en el software MATLAB. Se simularon seis escenarios de red con 100 nodos cada uno y se implementó el algoritmo *Random Walk* para simular la movilidad de los nodos. El rendimiento de los algoritmos fue evaluado en base a tres parámetros: el porcentaje de alcance de la red, la cantidad de retransmisiones ahorradas y el ahorro de potencia de los nodos. El algoritmo híbrido que presentó mejores resultados para mitigar el BSP es el algoritmo basado en probabilidad dinámica. Con este algoritmo se obtubieron porcentajes de alcance entre el 60% y el 100%, un ahorro de retransmisiones entre el 70% y el 90% y un ahorro de potencia del 70%.

PALABRAS CLAVES: MANET, VANET, *Broadcast Storm Problem*, FANET, MATLAB

ABSTRACT

In the present undergraduate degree work, it is analyzed the performance of the hybrid packet retransmission algorithms to mitigate the BSP (Broadcast Storm Problem) problem in FANET (Flying Ad-Hoc Network) networks and the power savings achieved by implementing these algorithms. The implementation of the network and the hybrid algorithms were performed in a simulation on MATLAB software. Six network scenarios with 100 nodes were simulated, and the Random Walk algorithm was implemented to simulate the nodes mobility. The performance of the algorithms was evaluated based on three parameters: the reachability, number of saved retransmissions, and the power saving of the nodes. The hybrid algorithm that presented better results to mitigate de BSP problem is the dynamic probability-based algorithm. With this algorithm were obtained percentages of reachability between 60% and 100%, saved retransmission between 70% and 90%, and a power saving of 70%.

KEYWORDS: MANET, VANET, *Broadcast Storm Problem*, FANET, MATLAB

LISTA DE ABREVIATURAS

BSP	<i>Broadcast Storm Problem</i>
MANET	<i>Mobile Ad-Hoc Network</i>
VANET	<i>Vehicular Ad-Hoc Network</i>
FANET	<i>Flying Ad-Hoc Network</i>
UAV	<i>Unmanned Aerial Vehicle</i>
DV-Cast	<i>Distributed Vehicular Broadcast</i>
SRD	<i>Simple and Robust Dissemination</i>

INTRODUCCIÓN

Gracias al gran avance tecnológico dentro de redes de sensores, electrónica y tecnologías de comunicación, actualmente es posible disponer de vehículos aéreos no tripulados (UAV por sus siglas en inglés) o drones que contienen sistemas de sensores y comunicaciones para realizar distintas misiones como monitoreo de variables en un área específica hasta aplicaciones militares. El conjunto de estos sistemas se conoce como FANET (*Flying Ad-Hoc Network*).

Una FANET, es una red conformada por nodos (UAVs o drones) que poseen movilidad y que pueden comunicarse entre sí. En estas redes no es necesario incorporar dispositivos que encaminen los paquetes, ya que los nodos, al poder comunicarse entre sí son los encargados de dirigir los paquetes hacia sus destinos. Esto permite que estas redes sean fáciles de desplegar e instalar, flexibles, versátiles y fáciles de operar. Eventos de emergencia, como desastres naturales, áreas militares o lugares en donde la infraestructura de telecomunicaciones esté comprometida o no se cuente con la misma, son los escenarios en donde se despliegan este tipo de redes. Con la implementación de este tipo de redes dentro de un área determinada se puede obtener datos como: estimación de la velocidad y calidad del aire; hasta monitorear eventos como incendios forestales, daños generados por desastres naturales o detección y seguimiento de objetos en tierra.

La importancia del estudio de estas redes radica en el problema que se genera en la retransmisión de paquetes. Al no contar con dispositivos como routers, estaciones base o puntos de acceso para encaminar los datos, son los nodos los encargados de retransmitir los paquetes entre ellos. Esto provoca la aparición de escenarios multisaltos, en donde un paquete tiene que ser retransmitido por todos los nodos que se encuentren entre el emisor y el receptor. A su vez, estas retransmisiones generan problemas a la red dando origen a 3 fenómenos: retransmisiones redundantes, contenciones y colisiones de paquetes; los cuales, en conjunto, se las conoce como Broadcast Storm Problem (BSP).

El presente trabajo tiene como propósito realizar un estudio del BSP, analizar diferentes algoritmos básicos e híbridos propuestos para evitar este problema, codificar los algoritmos en el software de simulación MATLAB y evaluar el desempeño de los mismos frente al

BSP. Todo esto con el fin de determinar qué algoritmos serían los más indicados para ser implementados en redes FANET en base al ahorro de paquetes.

En el primer capítulo, se presenta una línea base sobre retransmisión de paquetes en redes FANET, el problema del Broadcast Storm Problem y los algoritmos existentes para evitarlo. En el capítulo 2, se describe la metodología para la solución del problema, analizando los algoritmos de retransmisión de paquetes y las propuestas de algoritmos híbridos, se presenta las herramientas a utilizarse y las técnicas de análisis de resultados. La implementación de los algoritmos de retransmisión de paquetes en el software de simulación MATLAB se la realiza en el capítulo 3. En el cuarto capítulo, se realiza una discusión de resultados aplicando las técnicas de análisis de datos descritas en el segundo capítulo. Finalmente, se presentan las conclusiones y recomendaciones, así como líneas futuras de investigación que se surgieran para el presente trabajo.

CAPÍTULO 1

LÍNEA BASE

1.1. Redes Ad-Hoc

Una red *Ad-Hoc* es un tipo de red inalámbrica que no requiere de infraestructura de telecomunicaciones preexistente y de dispositivos encaminadores de paquetes como routers o puntos de acceso. Así, cada nodo presente en la red se encarga del ecaminamiento de los paquetes hacia su destino. Estas redes emplean CSMA/CA (Acceso Múltiple por Detección de Portadora y Prevención de Colisiones) como método de acceso al medio. Con este método, los nodos *escuchan* si existe una transmisión en la red antes de transmitir sus paquetes, esperando un tiempo aleatorio antes de transmitirlos. Como se muestra en la Figura 1.1, existen 3 tipos de redes *Ad-Hoc*: MANET (*Mobile Ad-Hoc Network*), VANET (*Vehicular Ad-Hoc Network*) y FANET (*Flying Ad-Hoc Network*).



Figure 1.1: Redes *Ad-Hoc*.

Fuente: [1].

Elaborado por: [1].

1.1.1. Redes MANET

Una MANET (*Mobile Ad-Hoc Network*) es una red conformada por nodos que poseen cierta movilidad arbitraria. Provocando cambios en la topología de la red de manera lenta y aleatoria. Estos nodos pueden ser dispositivos como: laptops, teléfonos celulares, sensores, etc, como se muestra en la Figura 1.2. Algunos de estos nodos pueden hacer uso de baterías u otros medios para obtener energía, requiriéndose optimizar el funcionamiento de éstos para conservar energía. Esta optimización produce una limitación en la potencia computacional de los nodos [1].

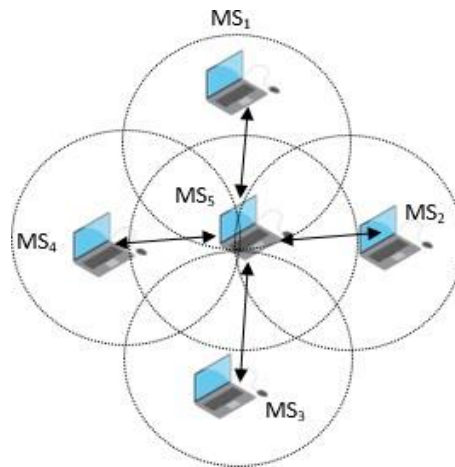


Figure 1.2: Red MANET.

Fuente: [1].

Elaborado por: Autor.

1.1.2. Redes VANET

Una VANET (Vehicular *Ad-Hoc* Network) es un subconjunto de red MANET en la cual, los vehículos (como, por ejemplo, autos, ambulancias, buses, etc.) son considerados como los nodos de la red. Es también conocida como red de alta movilidad ya que sus nodos, al ser vehículos, poseen gran libertad en movilidad y la velocidad con la que se desplazan es alta, provocando que los cambios en la topología de esta red se produzcan de manera rápida. Estas redes no presentan problemas de limitaciones de energía o espacio por obvias razones. Esto les permite tener una gran capacidad computacional [2]. En la Figura 1.3 se muestra un esquema de red VANET.

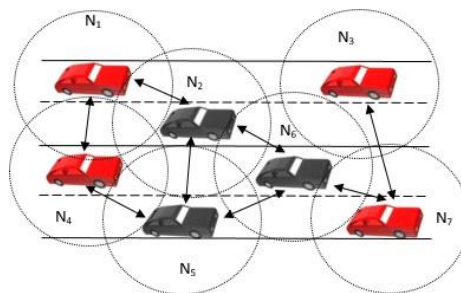


Figure 1.3: Red VANET.

Fuente: [2].

Elaborado por: Autor.

1.1.3. Redes FANET

Una red FANET (Flying Ad-Hoc Network) es un subgrupo de las redes MANET conformada por vehículos aéreos no tripulados (UAV) o Drones, los cuales son considerados como los nodos de esta red. En la Figura 1.4 se presenta un esquema de red FANET. Estos nodos pueden estar interconectados directamente o hacer uso de nodos intermediarios en escenarios multisaltos. Aunque existen algunas características que comparten con las redes MANET y VANET, existen algunas otras que las hacen diferentes como la movilidad, los cambios en la topología, el área de cobertura de los nodos, disponibilidad de energía y potencia computacional [3].

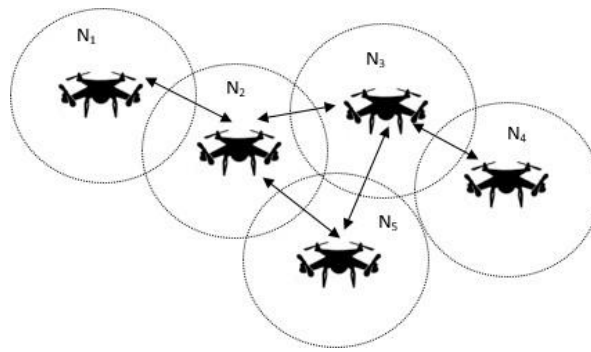


Figure 1.4: Red FANET.

Fuente: [3].

Elaborado por: Autor.

La movilidad de los nodos en una red FANET es mayor que en las redes VANET y MANET, debido a que la velocidad de los nodos es variable. Los nodos pueden ir desde encontrarse estáticos, realizando tareas de recolección de datos (en cuyo caso la velocidad del nodo sería nula); hasta realizar misiones de búsqueda y rescate (en las cuales se necesita una alta movilidad de los nodos). Adicionalmente, los nodos tienen total libertad de moverse en cualquier dirección y tienen la particularidad de poder desplazarse en tres dimensiones. Estas características dinámicas en la movilidad de las redes FANET hacen que los cambios en la topología de la red se presenten muy frecuentemente y de manera rápida, generando un impacto directo en los modelos de movilidad [3].

Se han propuesto varios modelos de movilidad para representar la movilidad y los cambios en la topología de las redes FANET. La existencia de varios modelos de movilidad se debe al dinamismo de este tipo de red. Cada modelo de movilidad puede ser requerido de

acuerdo al escenario de simulación, ya que los nodos pueden estar en reposo, o se puede requerir que los nodos estén en constante movimiento. Los modelos de movilidad que se han propuesto se detallan a continuación [3]:

- **Modelos de movilidad puramente aleatorios:** son los modelos más simples y los más comunes para implementar en este tipo de redes. Cada nodo elige aleatoriamente su velocidad y dirección independientemente de los otros nodos.
- **Modelos de movilidad dependientes de tiempo:** se basan en ecuaciones matemáticas que dependen tanto del estado actual de la simulación como la posición actual del nodo, esto con el fin de evitar cambios repentinos en la velocidad y la dirección.
- **Modelos de movilidad de rutas planificadas:** en estos modelos, los nodos siguen trayectorias pre-calculadas sin tomar ninguna dirección aleatoria.
- **Modelos de movilidad grupales:** se basan en movimientos coordinados por parte de todos los nodos de la red alrededor de un punto en común.
- **Modelos de movilidad híbridos:** se basan en la combinación de los modelos anteriormente descritos.

Los escenarios de aplicación de redes FANET están dentro del ámbito civil y militar. Un ejemplo de esto es el despliegue de una red FANET para manejo de desastres y seguridad civil como se lo explica en [4], en el cual, se despliega una red conformada por dos nodos los cuales tienen incorporados sensores para monitoreo de incendios de una estructura. Otro ejemplo se detalla en [5], en el cual, se despliega una red conformada por tres nodos para la detección y monitoreo de incendios forestales.

En [6], se propone una red FANET conformada por tres nodos para vigilancia y control de salud, en la cual el primer nodo se encarga de recoger los datos a transmitir, el segundo actúa como nodo de retransmisión entre el primer nodo y el nodo base; y el tercer nodo se encarga de recoger los datos y transmitirlos a la estación. En [7], se propone un escenario de monitoreo y vigilancia en un entorno semi urbano, además, se propone un algoritmo para calcular el número de nodos necesarios para cubrir una determinada área de trabajo. En [8], se describe una red FANET para misiones de búsqueda y persecución de objetivos, en este trabajo se describe como los nodos trabajan para comunicarse entre sí y dirigirse a los distintos puntos de monitoreo sin que dos o más nodos se encuentren en el mismo

punto. En [9], se detalla el despliegue de una red FANET para misiones de búsqueda y rescate, esta red está conformada por 4 nodos los cuales tienen incorporados sensores para detección de personas.

Generalmente los nodos de una red FANET tienen la energía suficiente para soportar sus equipos de comunicación y sus sistemas de sensores dentro de un periodo de tiempo, permitiéndoles disponer de una potencia computacional alta dentro de este periodo de tiempo. Los nodos de estas redes normalmente tienen áreas de cobertura extensas (en el orden de los kilómetros) por lo que no se requiere de una gran cantidad de nodos con respecto a las redes VANET y MANET [10].

En la tabla 1.1, se presenta un resumen de las principales características de las redes MANET, VANET y FANET, analizando movilidad, densidad, disponibilidad energética, cambio en la topología y la potencia computacional.

Tabla 1.1: Características redes *Ad-Hoc Ad-Hoc*

	MANET	VANET	FANET
Modelo de movilidad	Aleatorio	Constante	Usualmente aleatorio
Movilidad de los nodos	Baja	Alta	Muy alta
Densidad de nodos	Media	Alta	Baja
Disponibilidad de energía	Baja	Alta	Media
Cambios de la topología	Lenta	Rápida	Rápida
Potencia computacional	Limitada	Alta	Alta

Fuente: [1]

Elaborado por: Autor

1.2. Broadcast en redes FANET

Cuando uno de los nodos requiere enviar un paquete hacia otro, y debido a que en este tipo de redes no se hace uso de dispositivos encaminadores de paquetes, se presentan dos casos al realizar la transmisión del paquete. En el primer caso se presenta un escenario de comunicación directa, en el cual, el nodo receptor se encuentra dentro del área de cobertura del nodo emisor y la retransmisión se realiza directamente entre ellos, como se muestra en la Figura 1.5.a. En el segundo caso (Figura 1.5.b) se presenta un escenario multisalto, debido a que el nodo receptor no se encuentra dentro del área de cobertura del nodo emisor. En este caso, el paquete generado por el nodo emisor es retransmitido por todos los nodos que se encuentran entre el nodo emisor y receptor [1].

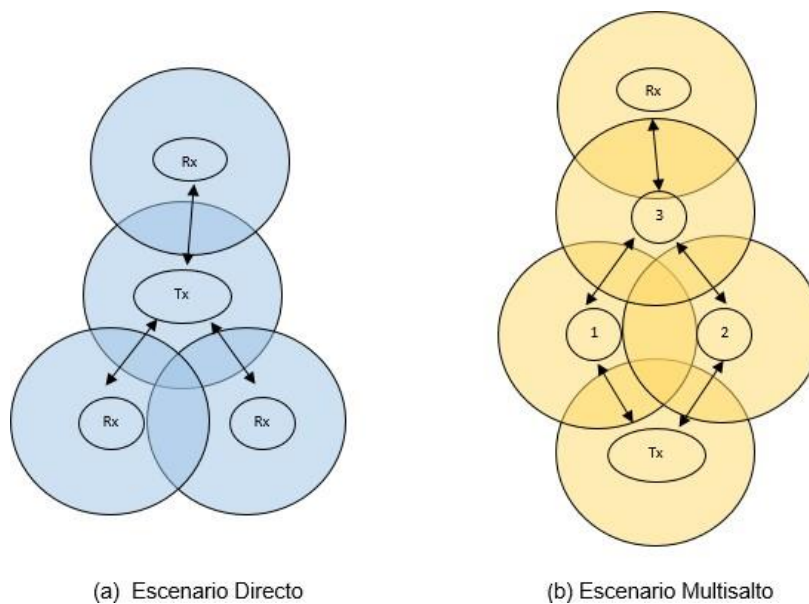


Figure 1.5: *Broadcast* en redes FANET.
Fuente: [1].
Elaborado por: Autor.

El método empleado para retransmitir paquetes dentro de estas redes se conoce como *Inundación (flooding)*. Con este método, cada vez que un nodo reciba un paquete, está obligado a retransmitirlo a todos sus nodos vecinos. Lo que implica un número de retransmisiones igual al número de nodos que conforman la red. Adicionalmente, se asume que todos los nodos poseen la capacidad de identificar paquetes duplicados (asociando cada paquete con un ID, un contador, etc.), esto con el fin de prevenir una *inundación* sin fin del mensaje, lo que provocaría un fallo completo de la red [11].

1.3. Broadcast Storm Problem (BSP)

En escenarios multisaltos, el método de *inundación* provoca el apareamiento de tres fenómenos en la red [11]:

- **Retransmisiones redundantes:** se producen debido a que múltiples nodos pueden contenerse entre sí dentro de sus respectivas áreas de cobertura, provocando que, al momento de retransmitir un paquete, este se replique múltiples veces y los nodos reciban el mismo paquete más de una vez.
- **Contención:** se produce luego de que los nodos vecinos reciben el paquete del nodo

emisor. Los nodos más cercanos generarán interferencia entre sí al momento de realizar la transmisión.

- **Colisiones:** son producto de las contenciones provocadas en la retransmisión del paquete entre los nodos cercanos. Cuando se produce una colisión, el nodo emisor del paquete está obligado a retransmitirlo nuevamente, provocando más retransmisiones redundantes.

Estos tres fenómenos, en conjunto, se los conoce como *Broadcast Storm Problem (BSP)*. El *BSP* ocasiona problemas a la red como: congestión, debido a la cantidad excesiva de paquetes que circulan por la red; aumento en la latencia y pérdida de paquetes; uso ineficiente del ancho de banda y principalmente un consumo innecesario de energía al realizar la retransmisión de los paquetes redundantes. Esto limita la disponibilidad de energía en los nodos y afecta directamente al tiempo de operación de los mismos y de la red.

1.4. Mecanismos para reducir el BSP

Se han propuesto distintos mecanismos para reducir los fenómenos que provocan el *BSP*. En el presente trabajo se describen cinco esquemas que pertenecen al grupo de los métodos heurísticos. Los cuales son llamados así porque requieren de la selección de un parámetro y la selección de valores umbrales para decidir si el nodo realiza o no la retransmisión del paquete. El rendimiento de estos métodos depende de los parámetros y los niveles de umbral seleccionados. Normalmente, los parámetros seleccionados están relacionados con el entorno de la red [12].

1.4.1. Esquema probabilístico

Una forma de reducir las retransmisiones redundantes que se generan por la *inundación* es empleando retransmisiones probabilísticas. Seleccionando un valor de probabilidad umbral para toda la red y asignándole un valor de probabilidad al paquete a retransmitir. Se define P_{umb} como el nivel umbral de probabilidad de la red (previamente seleccionado) y se define P_{asig} como el valor de probabilidad asignada al paquete (el cuál es un valor aleatorio entre 0 y 1 asignado por el nodo). Los nodos, al recibir un paquete de retransmisión, fijan el valor de P_{asig} al paquete y lo comparan con el valor de P_{umb} . Si el valor de P_{asig} es menor al valor

de P_{umb} el paquete se retransmite, caso contrario el nodo queda prohibido de retransmitir el paquete [11]. En la Figura 1.6 se detalla el algoritmo del esquema probabilístico.

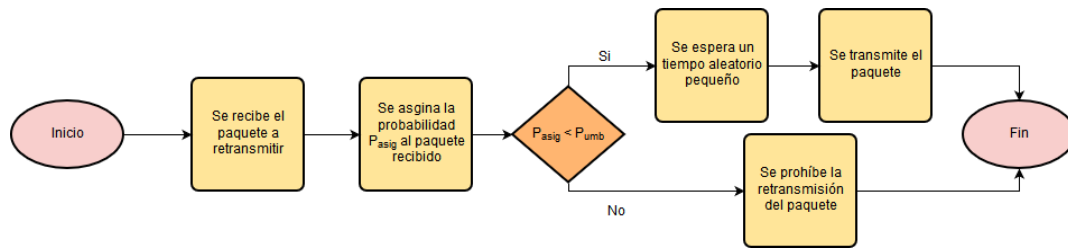


Figure 1.6: Algoritmo del esquema probabilístico.

Fuente: [11].

Elaborado por: Autor.

1.4.2. Esquema basado en contador

Otra forma de reducir las retransmisiones redundantes es usar un contador de paquete para rastrear el número de veces que el paquete ha sido recibido por el nodo. Se define C_{umb} como el nivel de contador umbral para toda la red y se define C_{msj} como el valor de contador del paquete por cada nodo. Al recibir el paquete por primera vez, se inicial el valor del contador del paquete en el nodo C_{msj} en uno, y se retransmite el paquete. En el caso de que el paquete ya se haya recibido anteriormente, se incrementa el valor de C_{msj} en uno y se compara con el valor de C_{umb} . Si C_{msj} es menor que el valor de C_{umb} se retransmite el paquete, caso contrario, se prohíbe al nodo de retransmitirlo [11]. En la Figura 1.7 se detalla el algoritmo del esquema basado en contador.

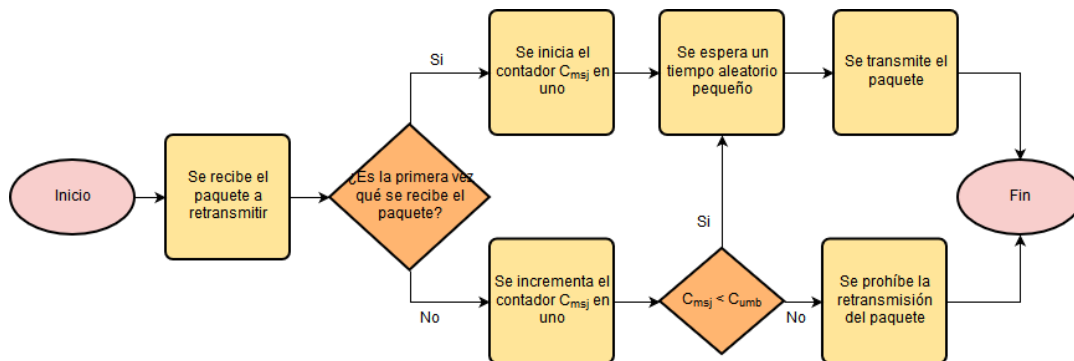


Figure 1.7: Algoritmo del esquema basado en contador.

Fuente: [11].

Elaborado por: Autor.

1.4.3. Esquema basado en distancia

El parámetro de selección que se emplea en este esquema es la distancia que existe entre el nodo emisor y receptor. Se define D_{min} como el valor de distancia umbral para toda la red, la cual es la distancia mínima a la cual se puede retransmitir el paquete, y se define D_{er} como la distancia que existe entre el nodo emisor y receptor. Cuando los nodos reciben por primera vez un paquete de retransmisión, se inicializa D_{er} con el valor de la distancia que existe entre el nodo emisor y receptor y se retransmite el paquete. Cuando el paquete ya ha sido recibido anteriormente se obtiene la distancia que existe entre el nodo emisor y receptor y se la compara con el valor de D_{er} , si el valor de D_{er} es menor que el valor de la nueva distancia, se actualiza el valor de D_{er} con el valor de la nueva distancia y se procede a comparar el valor de D_{min} y D_{er} . Si D_{er} es menor a D_{min} el paquete es retransmitido, caso contrario se prohíbe al nodo de retransmitirlo [11]. En la Figura 1.8 se detalla el algoritmo del esquema basado en distancia.

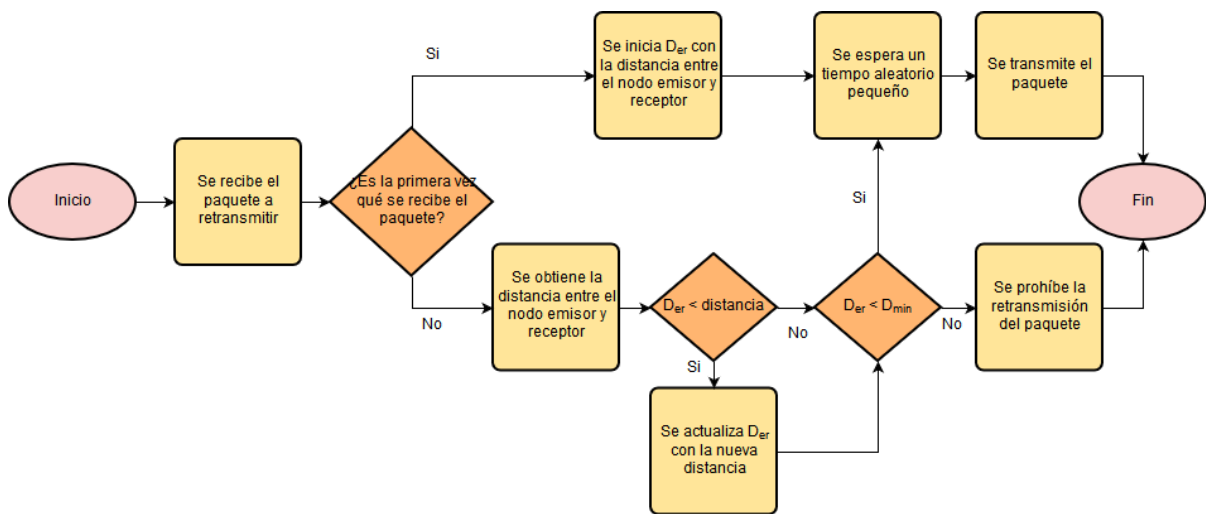


Figure 1.8: Algoritmo del esquema basado en distancia.

Fuente: [11].

Elaborado por: Autor.

1.4.4. Esquema basado en locación

Este esquema es similar al esquema basado en distancia, con la variante de que es necesario el uso de dispositivos que permitan obtener las locaciones de los nodos de manera más exacta. Si se adquiere las locaciones de los nodos transmisor y receptor, se puede cal-

cular su distancia de manera más precisa y, con ello, mejorar el rendimiento del algoritmo. Esto lo podemos conseguir haciendo uso de sistemas de posicionamiento con dispositivos GPS. Las variables de nivel de umbral y de comparación serán las mismas que el esquema basado en distancia [11]. En la Figura 1.9 se detalla el algoritmo del esquema basado en locación.

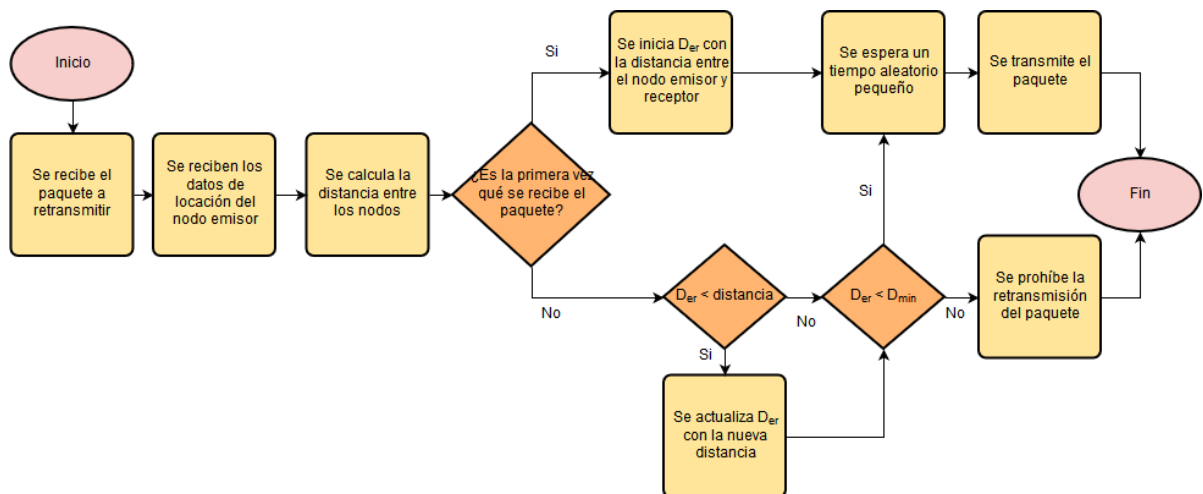


Figure 1.9: Algoritmo del esquema basado en locación.

Fuente: [11].

Elaborado por: Autor.

1.4.5. Esquema basado en clusters

Para este esquema, se asume que los nodos de la red se han agrupado en *clusters* y que se mantendrán bajo este modelo. En este esquema se definen dos nodos que serán los encargados de transmitir los paquetes: el nodo *head* y el nodo *gateway*. Dentro del *cluster*, el nodo *head* puede retransmitir un paquete hacia todos los demás nodos que se encuentran dentro del *cluster*. Para propagar mensajes a nodos que se encuentran en otros *clusters* se emplea al nodo *gateway*, siendo este el único nodo del *cluster* que puede enviar paquetes hacia otros *clusters*. Para retransmitir paquetes dentro de su mismo *cluster*, el nodo *head* puede utilizar cualquier esquema de los anteriormente descritos [11]. En la Figura 1.10 se detalla el algoritmo del esquema basado en *clusters*.

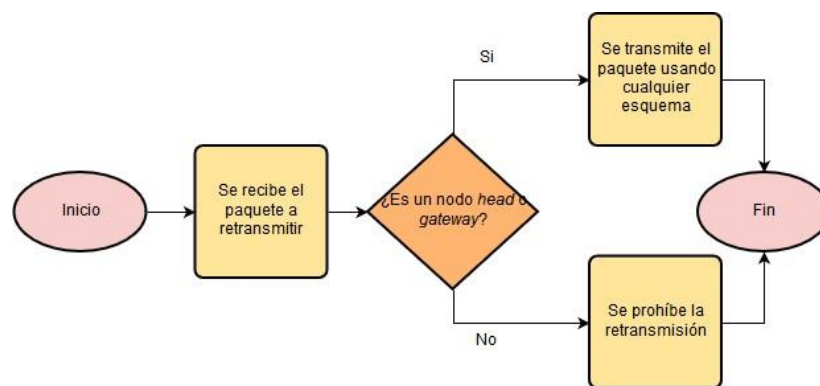


Figure 1.10: Algoritmo del esquema basado en *clusters*.

Fuente: [11].

Elaborado por: Autor.

CAPÍTULO 2
METODOLOGÍA

2.1. Selección de esquemas básicos

Los esquemas básicos que se implementaron en el presente trabajo fueron seleccionados en base a una clasificación previa de los mismos. En primer lugar, se consideró la factibilidad de implementar estos algoritmos en el software de simulación. Para esto se analizó los requerimientos de cada esquema (variables adicionales, dispositivos adicionales, etc.) necesarios para su correcto funcionamiento. Luego se analizó la adaptabilidad de los esquemas a las características de las redes FANET las cuales se encuentran en la Tabla 1.1, ya que los esquemas expuestos en la sección 1.4 han sido propuestos para reducir el problema que genera el *BSP* en redes MANET y VANET.

De esta clasificación se seleccionaron tres esquemas, los cuales presentaron adaptabilidad a un escenario de red FANET y factibilidad en su implementación. Los esquemas seleccionados se presentan a continuación:

- **Esquema Probabilístico**

Para la implementación del esquema probabilístico es necesario incorporar dos nuevas variables al entorno de simulación: una variable que contenga el valor umbral de probabilidad P_{umb} y otra variable que contenga el valor de probabilidad asignada a cada paquete de retransmisión P_{sig} . Ambas variables pueden ser incorporadas dentro de los parámetros generales de la simulación, incluyendo la actualización del valor que recibe la variable P_{sig} cada vez que se produzca una retransmisión del paquete, por lo que se lo consideró factible para su implementación.

- **Esquema basado en Contador**

Para implementar el esquema basado en contador es necesario incorporar dos nuevas variables al entorno de simulación: una que contenga el valor umbral de contador para toda la red C_{umb} y otra variable que contenga los valores de contadores del paquete para cada nodo de la red C_{msj} . Estas variables pueden ser incorporadas dentro de los parámetros generales de la simulación, incluyendo la actualización del valor de contador de mensaje para cada nodo, el cual requiere ser actualizado cada vez que el nodo reciba un paquete de retransmisión. Como ambas variables pueden ser incorporadas dentro de la simulación se consideró que el esquema es factible para su implementación.

- **Esquema basado en Distancia**

Para la implementación del esquema basado en distancia se requiere incorporar tres

nuevas variables al entorno de simulación: una variable que contenga el valor de distancia umbral D_{min} para toda la red, una segunda variable que contenga el valor de la distancia entre el nodo emisor y receptor de acuerdo al último paquete recibido por el nodo D_{er} y una tercera variable que contenga el valor actual de la distancia entre el nodo emisor y receptor para luego usar esta variable y actualizar el valor de D_{er} de acuerdo al algoritmo del esquema. Estas variables pueden ser incorporadas dentro de los parámetros generales de la simulación, incluyendo la obtención del valor de la distancia entre el nodo emisor y receptor y la actualización del valor de D_{er} , por lo que se lo consideró factible para su implementación.

De acuerdo a lo expuesto en la secciones 1.4.1, 1.4.2 y 1.4.3, ninguno de los esquemas seleccionados presentan restricciones que puedan limitar alguna de las características de una red FANET, las cuales se detallan en la Tabla 1.1. Por lo que se los consideró adaptables a un escenario de este tipo de redes.

El esquema basado en locación se lo consideró no factible para su implementación debido a que requiere el uso de dispositivos de locación para calcular la distancia real que existe entre los nodos. Como el presente trabajo se lo realizó en un entorno de red simulado, se puede considerar que la distancia calculada entre los nodos es exacta y la respuesta del esquema sería similar a los resultados que se obtengan de la implementación del esquema basado en distancia.

El esquema basado en *clusters* se lo consideró no adaptable a un escenario de red FANET y, por ende, no factible para su implementación debido a que los requerimientos del esquema limitan una de las características de las redes FANET. De acuerdo a lo expuesto en la sección 1.4.5, el esquema basado en *clusters* requiere que los nodos se agrupen en *clusters* y que se mantengan así durante todo el tiempo, pero esto no es posible debido a la alta movilidad que presentan los nodos de estas redes, como se muestra en la tabla 1.1. La movilidad de los nodos de una red FANET es mayor en comparación con las redes MANET y VANET y mantener los *clusters* inicialmente formados no es posible, con lo cuál el algoritmo del esquema fallaría, ocasionando problemas en la red.

2.2. Selección de algoritmos híbridos

Luego del proceso de selección de los esquemas básicos, se realizó una revisión de literatura sobre algoritmos híbridos propuestos para reducir el *BSP* que estén basados en los esquemas descritos en la sección 1.4. Luego se seleccionó los algoritmos híbridos a implementarse en base a los esquemas básicos que los conforman. Los algoritmos híbridos seleccionados fueron aquellos que están conformados por los esquemas básicos considerados a implementarse de acuerdo a la sección 2.1.

2.2.1. Algoritmo basado en contador con probabilidad arreglada

Este algoritmo busca mejorar el esquema basado en contador, incorporando el esquema probabilístico para determinar la retransmisión del paquete recibido [13]. Para este algoritmo se emplean las mismas variables para los valores de probabilidad umbral, contador umbral, probabilidad asignada al paquete y contador de paquete por nodo definidas en el esquema probabilístico y basado en contador.

Al recibir un paquete por primera vez se inicializa C_{msj} en uno y se retransmite el paquete. En el caso de que ya se haya recibido el paquete anteriormente, se incrementa el valor de C_{msj} y se compara con el valor de C_{umb} . Si C_{msj} es mayor o igual que C_{umb} el nodo queda prohibido de retransmitir el paquete. Caso contrario se asigna el valor de P_{sig} al paquete y se compara con el valor de P_{umb} . Si el valor de P_{sig} es menor que P_{umb} se retransmite el paquete, caso contrario se prohíbe al nodo de la retransmisión [13]. En la Figura 2.1 se presenta más a detalle el proceso del algoritmo.

2.2.2. Algoritmo basado en contador con probabilidad dinámica

Definir arbitrariamente el valor umbral de probabilidad para la retransmisión de los paquetes puede no ser óptimo a nivel global debido a que el rendimiento del algoritmo depende también de otros parámetros como la topología y el área de cobertura de la red, el rango de transmisión, cantidad de nodos, etc. En este algoritmo, se propone usar estos parámetros para calcular el valor umbral de probabilidad (P_{umb}) con el fin de que el algoritmo se adapte a cualquier escenario de red que se desee. Este algoritmo trabaja de igual manera que el algoritmo basado en probabilidad arreglada, con la diferencia de que el valor de P_{umb} debe

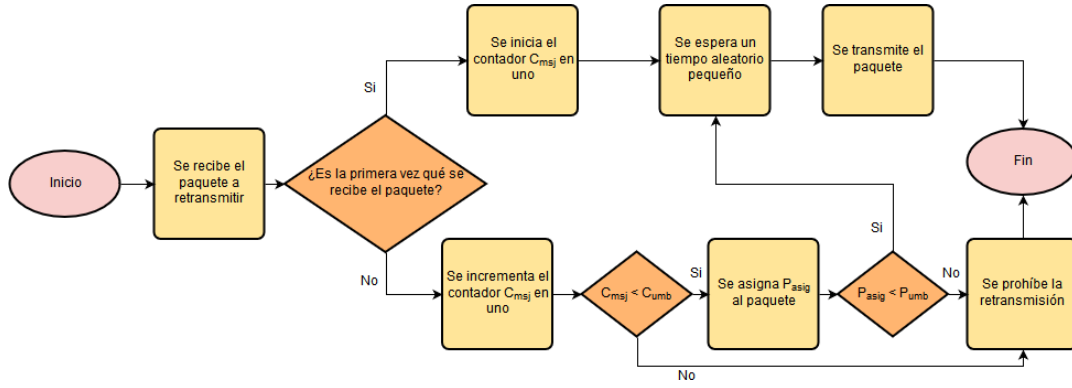


Figure 2.1: Algoritmo del esquema basado en contador con probabilidad arreglada.
Fuente: [13].
Elaborado por: Autor.

ser calculado a partir de las siguientes ecuaciones [13]:

$$\lambda = (N - 1) \frac{\pi R^2}{A} \quad (2.1)$$

$$P_{umb} = \exp - \frac{C_{msj}}{\lambda} \quad (2.2)$$

Donde:

N : número de nodos vecinos.

R : radio de transmisión (m).

A : área de cobertura de la red (m²).

P_{umb} : umbral de probabilidad.

C_{msj} : valor actual del contador del paquete en el nodo.

En el caso de que el paquete entrante ya se haya recibido con anterioridad, se incrementa el valor de C_{msj} y se compara con el valor de C_{umb} . Si C_{msj} es mayor o igual que C_{umb} el nodo queda prohibido de retransmitir el paquete. Caso contrario se procede a asignar el valor de P_{asig} al paquete y se calcula el valor de P_{umb} haciendo uso de las ecuaciones (2.1) y (2.2) y se compara ambos valores. Si P_{asig} es menor que P_{umb} se retransmite el paquete, caso contrario el nodo queda prohibido de retransmitirlo. En la Figura 2.2 se muestra más a detalle el proceso del algoritmo.

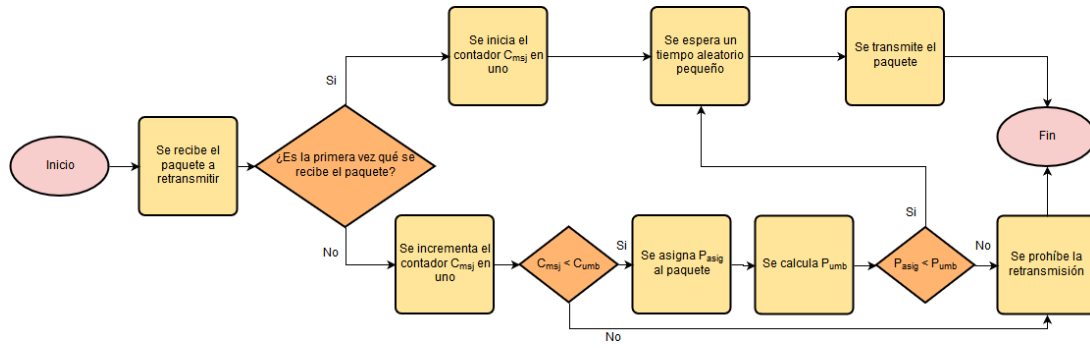


Figure 2.2: Algoritmo del esquema basado en contador con probabilidad dinámica.

Fuente: [13].

Elaborado por: Autor.

2.2.3. Retransmisión Vehicular Distribuida (DV-Cast)

Este algoritmo busca mejorar el esquema basado en distancia, incorporando el esquema probabilístico para determinar la retransmisión de los paquetes. Este algoritmo presenta dos variantes: en su primer variante el valor de probabilidad umbral es definido de manera arbitraria, y en la segunda el valor de probabilidad umbral es calculado de manera dinámica [14].

2.2.3.1. DV-Cast con probabilidad arbitraria

Para este algoritmo se emplean las mismas variables definidas en los esquemas probabilístico y basado en distancia, las cuales contendrán los valores de probabilidad umbral P_{umb} , probabilidad asignada al paquete P_{asig} , distancia umbral D_{min} , distancia entre los nodos D_{er} y una variable adicional que contenga el valor actual de la distancia entre los nodos. Este algoritmo realiza el mismo procedimiento que el esquema basado en distancia el cual se encuentra detallado en la Figura 1.8. Al momento de decidir la retransmisión del paquete por parte del esquema basado en distancia, se asigna el valor de probabilidad P_{asig} al paquete y se compara con el valor de probabilidad umbral P_{umb} . Si P_{asig} es menor que P_{umb} se retransmite el paquete, caso contrario se prohíbe su retransmisión [14]. En la Figura 2.3 se detalla el algoritmo.

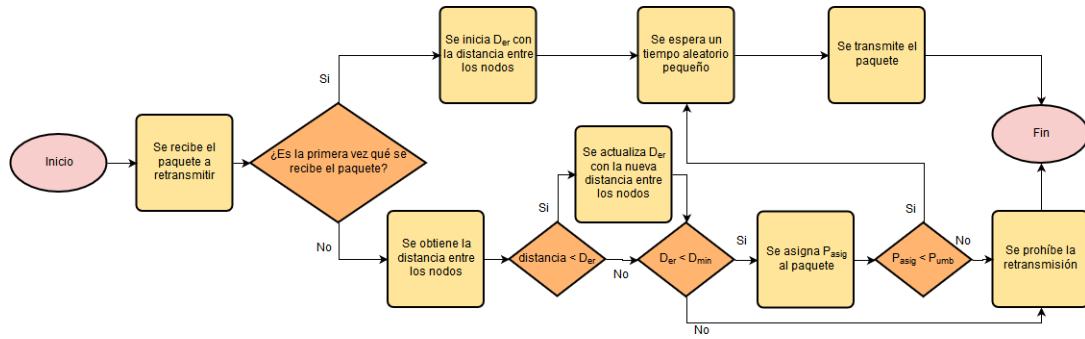


Figure 2.3: Algoritmo DV-Cast con probabilidad arbitraria.

Fuente: [14].

Elaborado por: Autor.

2.2.3.2. DV-Cast con probabilidad dinámica

Como se mencionó en la sección 2.2.2, definir de manera arbitraria el valor umbral de probabilidad puede no ser óptimo a nivel global. Por eso en este algoritmo se propone calcular el valor umbral de probabilidad en base a los parámetros de la red. Para este algoritmo, el valor umbral de probabilidad se lo calcula a partir de la siguiente ecuación [14]:

$$P_{umb} = \frac{D_{ij}}{R} \quad (2.3)$$

Donde:

P_{umb} : umbral de probabilidad.

D_{ij} : distancia entre los nodos (m).

R : radio de transmisión (m).

Este algoritmo realiza el mismo procedimiento que el esquema basado en distancia el cual se encuentra detallado en la Figura 1.8. Al momento de decidir la retransmisión del paquete por parte del esquema basado en distancia, se asigna el valor de probabilidad P_{asig} al paquete y se calcula el valor umbral de probabilidad P_{umb} de acuerdo a la ecuación (2.3). Si el valor de P_{asig} es menor que P_{umb} se retransmite el paquete, caso contrario se prohíbe la retransmisión del paquete [14]. En la Figura 2.4 se muestra el algoritmo más a detalle.

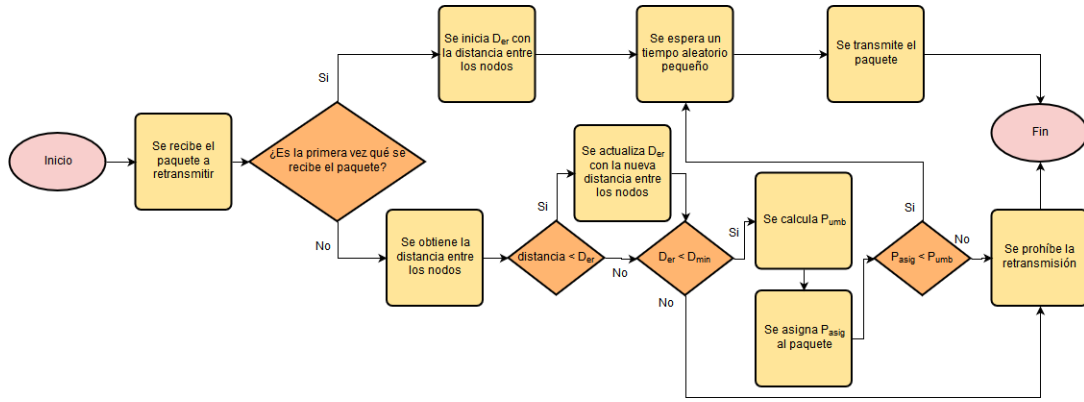


Figure 2.4: Algoritmo DV-Cast con probabilidad dinámica.

Fuente: [14].

Elaborado por: Autor.

2.2.4. Difusión Sencilla y Robusta (SRD)

Este algoritmo trabaja con los esquemas basados en distancia y *clusters*, buscando que la eficiencia de ambos mejore al trabajar en conjunto. Además, este algoritmo implementa una nueva técnica que se basa en adicionar un retraso o *delay* antes de realizar la retransmisión del paquete. Se asignan intervalos de tiempo para que cada nodo pueda retransmitir el paquete sin causar interferencia entre ellos. El valor de la distancia entre los nodos D_{er} y los intervalos de tiempo para cada nodo se calculan en base a las siguientes ecuaciones [15]:

$$D_{er} = \frac{\min(D_{ij}, R)}{R} \quad (2.4)$$

$$S_{ij} = [N_s * (1 - D_{er})] \quad (2.5)$$

Donde:

D_{er} : distancia entre los nodos (m).

D_{ij} : distancia actual entre los nodos (m).

R : radio de transmisión (m).

S_{ij} : número de intervalos de tiempo asignados al nodo.

N_s : número total de intervalos de tiempo disponibles.

En la Figura 2.5 se detalla el algoritmo.

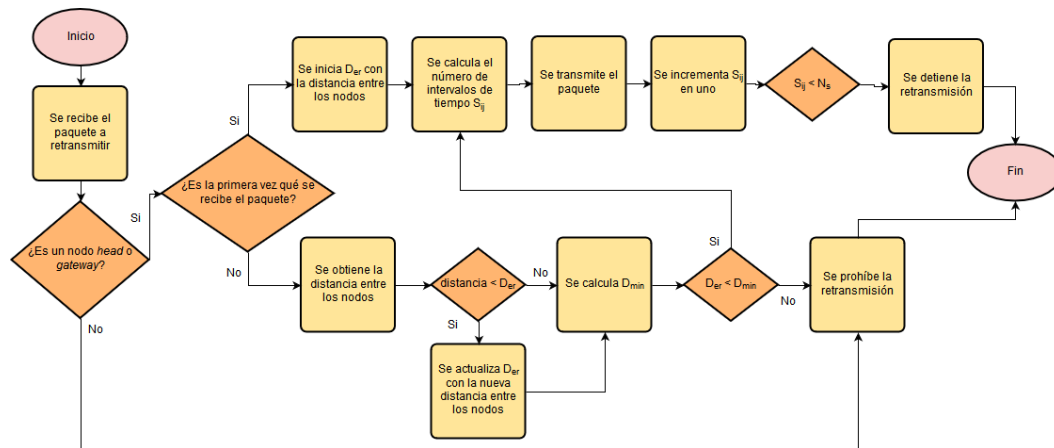


Figure 2.5: Algoritmo *SRD*.

Fuente: [15].

Elaborado por: Autor.

La Tabla 2.1 presenta un resumen de los algoritmos híbridos y los esquemas que los conforman.

Tabla 2.1: Algoritmos Híbridos

Algoritmo	Esquemas
Contador con probabilidad arreglada	Probabilístico-Contador
Contador con probabilidad dinámica	Probabilístico-Contador
<i>DV-Cast</i>	Probabilístico-Distancia
<i>SRD</i>	Distancia-Contador- <i>Delay</i>

Fuente: Autor

Elaborado por: Autor

Los algoritmos híbridos seleccionados para su implementación son: algoritmo basado en contador con probabilidad arreglada y dinámica y el algoritmo *DV-Cast*, debido a que se encuentran conformados por los esquemas seleccionados en la sección 2.1.

2.3. Selección del modelo de movilidad

Retomando lo expuesto en la sección 1.1.3, existen diversos modelos de movilidad propuestos para representar el movimiento de los nodos de una red FANET. Estos modelos varían de acuerdo a los requerimientos o el escenario de red a implementarse. A continuación, se describe más a detalle estos modelos.

2.3.1. Modelos de movilidad puramente aleatorios

Como se describió anteriormente, en estos modelos cada nodo elige aleatoriamente su dirección y velocidad independiente de los demás, lo que los convierte en los más simples y más comúnmente implementados para simular el movimiento de los nodos en estas redes. Dentro de los modelos de movilidad puramente aleatorios se encuentran [16]:

2.3.1.1. *Random Walk*

Este modelo de movilidad fue diseñado en base a lo impredecible que resulta el movimiento de los nodos en la red. Este modelo simula un movimiento irregular, eligiendo en cada instante de tiempo una dirección y una velocidad de manera aleatoria. Cada movimiento tiene lugar en un intervalo de tiempo y al final de este intervalo se calcula la nueva dirección y la velocidad en base a rangos previamente definidos para los mismos. Se lo considera un modelo de movilidad sin memoria ya que no almacena sus ubicaciones y sus velocidades pasadas [16].

2.3.1.2. *Random Waypoint*

Este modelo de movilidad parte del modelo Random Walk con algunas diferencias: en primer lugar, los nodos empiezan en estado estacionario durante un periodo de tiempo, luego de este tiempo los nodos eligen un destino aleatorio dentro del área de simulación. Su velocidad es uniformemente distribuida entre el punto inicial y el nuevo destino. Luego de haber llegado a su nuevo destino, el nodo hace una pausa durante un periodo de tiempo antes de repetir el proceso nuevamente [17].

2.3.2. Modelos de movilidad dependientes del tiempo

Estos modelos de movilidad tratan de evitar cambios bruscos en la velocidad y la dirección de los nodos haciendo uso de ecuaciones matemáticas. Dentro de estos modelos de movilidad se encuentran:

2.3.2.1. Gauss-Markov

Este modelo utiliza múltiples parámetros los cuales permiten ajustar el modelo para adaptarse a diferentes niveles de aleatoriedad. Al comienzo de la simulación, a cada nodo se le asigna una velocidad y una dirección. Luego, para cada instancia de tiempo, la nueva velocidad y la nueva dirección son calculadas en base a la dirección y velocidad anteriores [17].

2.3.2.2. Smooth Turn

Este modelo de movilidad permite que los nodos se muevan en trayectorias flexibles al correlacionar la aceleración del nodo a través de coordenadas espaciales y temporales. Cada nodo elige un punto en el área de simulación y luego circula alrededor de éste, hasta que el nodo elija otro punto de circulación [16].

2.3.3. Modelos de movilidad de rutas planificadas

Para estos modelos de movilidad es necesario predefinir las rutas que recorrerán los nodos para forzar al nodo a recorrerla. Cada nodo sigue un patrón específico de ruta hasta que llega al final de esta. Al llegar al final de la ruta, el nodo puede elegir entre cambiar de ruta o repetir la ruta anterior [16]. Algunos de los modelos de movilidad pertenecientes a este grupo son:

2.3.3.1. Movimiento Circular Semi-Aleatorio

Este modelo de movilidad fue diseñado para escenarios de movilidad curvos o para simular el desplazamiento de un nodo alrededor de un punto. Para este modelo se define un círculo alrededor del punto de monitoreo y el nodo parte de un punto en el círculo. Con una velocidad distribuida uniformemente entre la distancia del punto al círculo. Luego, para cada instancia de tiempo, el nodo se desplaza a un nuevo punto en el círculo repitiendo el proceso hasta haber recorrido todo el círculo. Luego el nodo elegirá aleatoriamente entre desplazarse a un nuevo círculo o repetir el proceso en el mismo círculo [17].

2.3.3.2. *Paparazzi*

Es un modelo de movilidad estocástico basado en una máquina de estado la cuál contiene seis rutas de movimiento. Al comienzo de la simulación, cada nodo elige el tipo de movimiento, su posición inicial y su velocidad. Luego de terminar su ruta, el nodo elige nuevamente la ruta, posición inicial y velocidad [16].

2.3.4. Modelos de movilidad grupales

Estos modelos de movilidad se basan en movimiento de nodos en forma grupal, en la que la dirección y velocidad de los nodos son iguales para todos los nodos que conforman la red. Algunos modelos de movilidad grupales son:

2.3.4.1. *Modelo de movilidad en columna*

En este modelo, los nodos se mueven alrededor de un punto de referencia o de una ruta predeterminada con la misma velocidad, dirección y a una distancia predeterminada entre ellos. Este modelo es propuesto para escenarios de exploración o búsqueda de objetivos [16].

2.3.4.2. *Modelo de movilidad de comunidad nómada*

En este modelo, los nodos de la red se mueven alrededor de un punto de referencia sin ninguna restricción, es decir, no es necesario una distancia predeterminada entre los nodos. Este modelo puede presentar algunos problemas al momento de llevarlo a un escenario real, ya que se pueden producir colisiones de los nodos al no haber un control en el trayecto de los nodos [17].

El modelo de movilidad elegido para el presente trabajo es el *Random Walk* ya que pertenece al grupo de modelos de movilidad comúnmente empleados en simulaciones de redes FANET además de permitir simular una red más dinámica, ya que los nodos tienen completa libertad de elegir su velocidad y dirección.

2.4. Selección de métricas de evaluación

Para seleccionar las métricas de evaluación primero se realizó una recopilación de las métricas que se han evaluado en las distintas implementaciones tanto de los esquemas base como de los algoritmos híbridos seleccionados. Las métricas que se evalúan en [10-15] se resumen en la Tabla 2.2.

Tabla 2.2: Métricas evaluadas

Algoritmos	Técnicas	Métricas
Esquemas básicos	Probabilístico Contador Distancia	Alcance Transmisiones/Retransmisiones ahorradas Latencia promedio
Contador con probabilidad arreglada Contador con probabilidad dinámica	Probabilístico Contador	Alcance Transmisiones/Retransmisiones ahorradas Mensajes Generados Memoria Usada
<i>DV-Cast</i>	Probabilístico Distancia	Alcance Transmisiones/Retransmisiones ahorradas Sobrecarga de la red Memoria usada

Fuente: Autor

Elaborado por: Autor

Para el presente trabajo, las métricas seleccionadas para su implementación fueron aquellas comunes entre las que se analizan en los esquemas básicos y los algoritmos híbridos. A continuación se descubren más a detalle y se explica su método de cálculo:

- *Alcance*: porcentaje de nodos de la red que han recibido el paquete. Se lo obtiene del cociente entre el número de nodos que han recibido el paquete y la cantidad total de nodos de la red.

$$\text{Alcance} = [100 * \text{nodos}_{\text{msj}} / \text{nodos}_{\text{total}}] \quad (2.6)$$

Donde:

$\text{nodos}_{\text{msj}}$: cantidad de nodos que han recibido el paquete al menos una vez.

$\text{nodos}_{\text{total}}$: total de nodos de la red.

- *Transmisiones*: porcentaje de nodos de la red que han retransmitido el paquete. Se lo obtiene del cociente entre la cantidad de nodos que han retransmitido el paquete y la

cantidad de nodos que conforman la red.

$$\text{Transmisiones} = [100 * (\text{nodos}_{\text{ret}}/\text{nodos}_{\text{total}})] \quad (2.7)$$

Donde:

$\text{nodos}_{\text{ret}}$: cantidad de nodos que han retransmitido el paquete.

- *Retransmisiones ahorradas*: porcentaje de nodos de la red que han recibido el paquete, pero no lo han retransmitido. Se lo obtiene de la diferencia entre el alcance y la cantidad de retransmisiones.

$$\text{Retransmisiones} = \text{Alcance} - \text{Retransmisiones} \quad (2.8)$$

- *Ahorro de potencia*: cantidad de potencia (mW) economizada por la red. Se la calcula en base a la potencia de transmisión necesaria para realizar la retransmisión del paquete hacia el nodo receptor. Para el cálculo de esta métrica se aplican las siguientes ecuaciones [18]:

$$M = P_{rx} - U_{rx} \quad (2.9)$$

$$L_b = 32.45 + 20\log(f) + 20\log(d) \quad (2.10)$$

$$P_{rx} = P_{tx} - L_{tt} - L_{tr} + G_{tx} + G_{rx} - L_b \quad (2.11)$$

Donde:

M : margen de desvanecimiento.

P_{rx} : potencia de recepción.

U_{rx} : umbral de recepción.

L_b : pérdidas en espacio libre.

f : frecuencia de operación (MHz).

d : distancia entre los nodos (Km).

P_{tx} : potencia de transmisión.

L_{tt} : pérdidas por cables y conectores en el transmisor.

L_{tr} : pérdidas por cables y conectores en el receptor.

G_{tx} : ganancia de la antena transmisora.

G_{rx} : ganancia de la antena receptora.

En cada ocasión en que se presente un ahorro de retransmisión, se procede a calcular la cantidad de potencia que se hubiese requerido para realizar la retransmisión. Este valor se irá adicionando en cada ocasión para así obtener la cantidad total de energía ahorrada en la red. Para la implementación de esta métrica, se decidió utilizar las características del dron Inspire 2 fabricado por la empresa DJI, del cual se dispone un modelo en el laboratorio de Telecomunicaciones de la UTPL. Por lo tanto, los datos necesarios para el reemplazo de variables en las ecuaciones 2.9, 2.10, y 2.11 se encuentran en las especificaciones técnicas del dron Inspire 2, disponibles en el Anexo A y las características del sistema de transmisión que incorpora este dron, las cuales se encuentran en el Anexo B.

CAPÍTULO 3

RESULTADOS

La implementación de los esquemas básicos y los algoritmos híbridos fué realizada en el software MATLAB (MATrix LABoratory), el cuál es un software de cómputo numérico que permite realizar cálculos complejos a partir de una representación de valores basados en matrices. Contiene librerías y comandos de alto nivel que lo hace adecuado para trabajar con simulaciones dentro de ámbitos científicos y tecnológicos. Presenta un entorno interactivo que permite tanto realizar la programación de algoritmos, entrada y salida de datos de forma versátil, así como la visualización de los mismos de forma sencilla [19]. La versión de MATLAB utilizada en este trabajo es la R2017a. Las características del computador en el cual se realizó el presente trabajo se detallan a continuación:

Tabla 3.1: Características computador

Dispositivo	Laptop
Marca	ACER ES-471
Sistema Operativo	Windows 8.1
Procesador	Intel Core i5-4210U
Memoria RAM	6.00 GB
Tipo de sistema	64 bits

Fuente: Autor.

Elaborado por: Autor.

3.1. Implementación de esquemas básicos

Para la implementación de los esquemas básicos se utilizaron los parámetros generales de simulación de acuerdo a [11]. Esto con el fin de realizar una validación del algoritmo y determinar su correcta implementación. Realizando una comparación entre los resultados obtenidos en [11] y los resultados obtenidos en el presente trabajo. Los esquemas básicos que se implementaron fueron los seleccionados en la sección 2.1. En la Tabla 3.2 se detallan los parámetros generales de simulación y los escenarios de simulación se detallan en la Tabla 3.3.

En el Anexo C se detalla el código principal desarrollado para simular los parámetros generales de la red de acuerdo a lo expuesto en las Tablas 3.2 y 3.3, En el anexo D se encuentra el código desarrollado para simular la retransmisión de los paquetes entre los nodos y el cálculo de la potencia ahorrada cada vez que se genere un ahorro de retransmisión. En el Anexo E se encuentra el código para la implementación del algoritmo de movilidad *Random Walk*, con este código se general el movimiento de los nodos de acuerdo al escenarios de

Tabla 3.2: Parámetros de Simulación

Cantidad de nodos	100
Radio de transmisión	500m
Distribución inicial	Aleatoria
Algoritmo de movilidad	Random Walk
Probabilidad umbral	0,2 - 0,4 - 0,6 - 0,8 - 1
Contador umbral	2 - 3 - 4 - 5 - 6
Distancia umbral	11m - 20m - 37m - 72m - 147m

Fuente: [11].

Elaborado por: Autor.

Tabla 3.3: Escenarios de Simulación

Escenarios	Área de simulación (u) 1u = 500m	Velocidad de los nodos (Km/h)
Escenario 1	1x1	0-10
Escenario 2	3x3	0-30
Escenario 3	5x5	0-50
Escenario 4	7x7	0-70
Escenario 5	9x9	0-90
Escenario 6	11x11	0-110

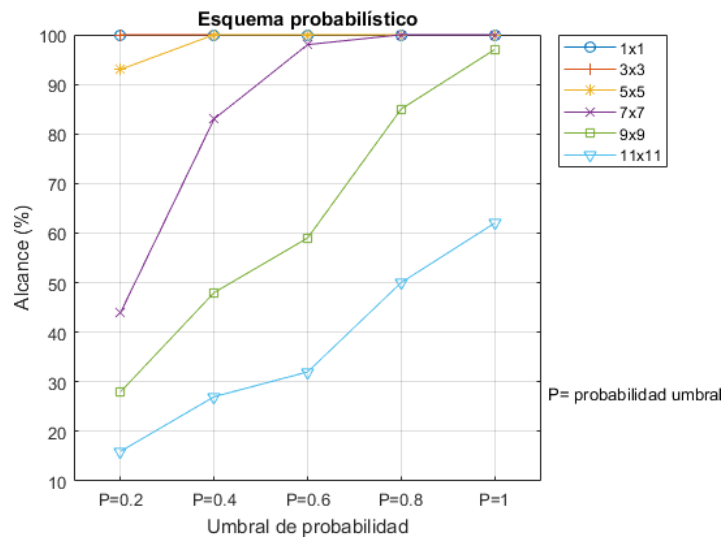
Fuente: [11].

Elaborado por: Autor.

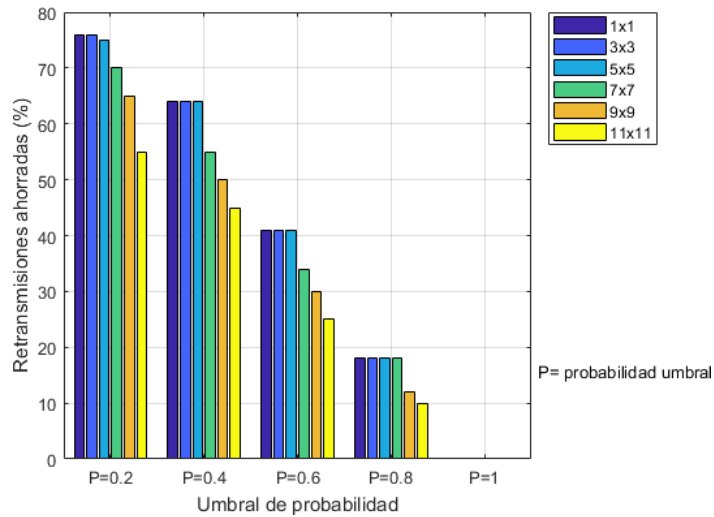
simulación y al rango de velocidad definido para cada escenario, y se actualiza la matriz que contiene la distancia entre los nodos cada vez que los nodos realicen su movimiento.

3.1.1. Resultados para el esquema probabilístico

El rendimiento del esquema probabilístico se presenta en la Figura 3.1. Los porcentajes de alcance de la red que se obtienen con este esquema se muestran en la Figura 3.1(a) y las retransmisiones ahorradas se muestran en la Figura 3.1(b) Si se fija el valor de probabilidad umbral P_{umb} en uno, el esquema se aproxima al método de inundación ya que la cantidad de retransmisiones ahorradas es igual a cero. A medida que el valor de probabilidad umbral se aproxima a cero, el porcentaje de retransmisiones ahorradas aumenta y el porcentaje de alcance disminuye. Para escenarios con una alta densidad de nodos (1x1, 3x3, 5x5), un valor umbral de probabilidad bajo (entre 0.2 y 0.5) es suficiente para garantizar un porcentaje alto de retransmisiones ahorradas y garantizar el máximo alcance en la red. En escenarios extensos, en donde los nodos se encuentran muy dispersos (7x7 9x9 11x11), es necesario aumentar el valor umbral de probabilidad para garantizar el máximo alcance



(a) Alcance.



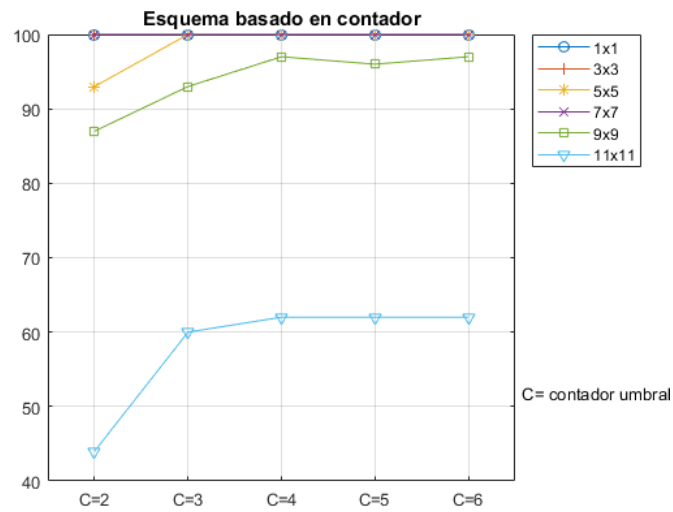
(b) Retransmisiones ahorradas.

Figure 3.1: Rendimiento del esquema probabilístico.

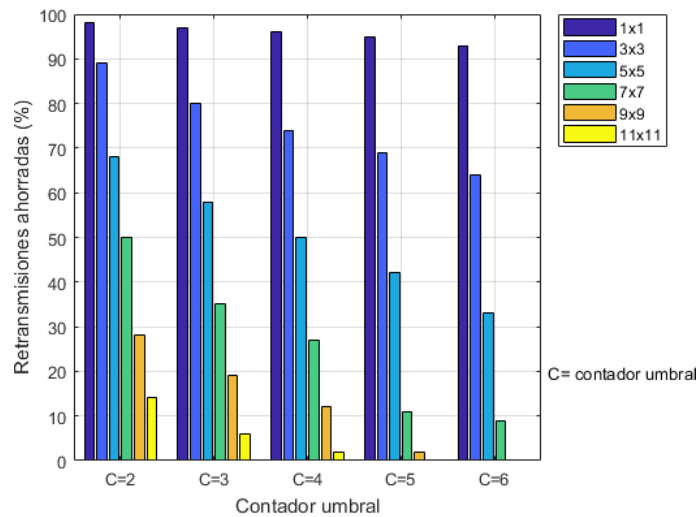
Fuente: Autor.

Elaborado por: Autor.

en la red. Con un valor de probabilidad umbral igual a 0.6 se puede garantizar entre el 30% y 40% de retransmisiones ahorradas con un alcance del 100% para escenarios de alta densidad y del 50% para escenarios en donde los nodos se encuentran dispersos.



(a)Alcance.



(b)Retransmisiones ahorradas.

Figure 3.2: Rendimiento del esquema basado en contador.

Fuente: Autor.

Elaborado por: Autor.

3.1.2. Resultados para el esquema basado en contador

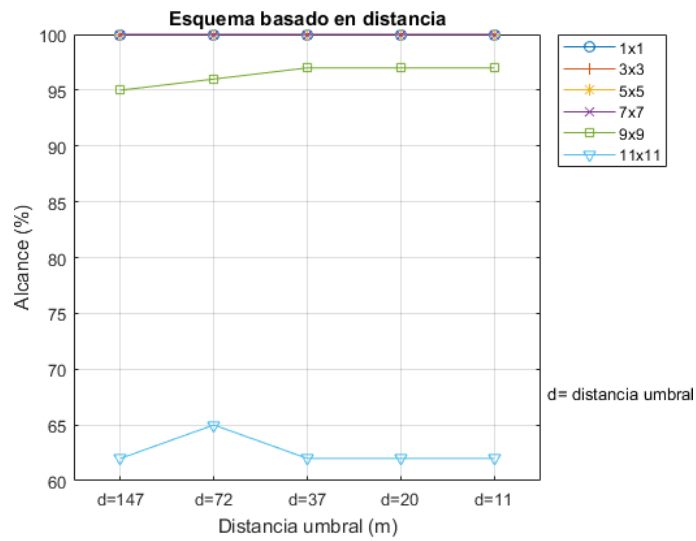
En la Figura 3.2 se presentan los resultados de la implementación del esquema basado en contador. Este esquema presenta un alto porcentaje de retransmisiones ahorradas en escenarios de alta densidad de nodos, pero disminuye drásticamente a medida que el esparcimiento de los nodos aumenta, tal como se muestra en la Figura 3.2(b). Esto se

debe a que la cantidad de nodos vecinos del nodo emisor disminuye al aumentar el área de simulación. Se puede observar que a medida que el valor de contador umbral aumenta, el esquema se aproxima al método de inundación. Con un valor de contador umbral igual a 2 se puede garantizar entre el 45% y 100% de alcance de la red (Figura 3.2(a)) con el máximo porcentaje de ahorro de retransmisiones en cada escenario.

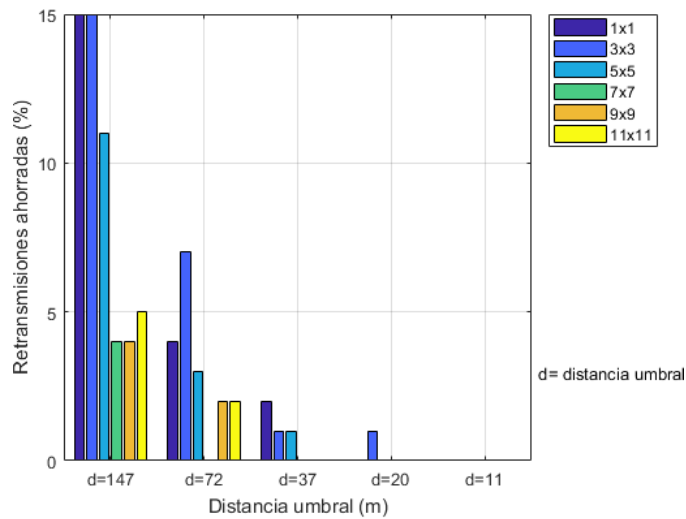
3.1.3. Resultados para el esquema basado en distancia

De los resultados que se muestran en la Figura 3.3 se observa que, con la implementación de este esquema, se puede obtener niveles de alcance constantes para todos los valores de distancia umbral (Figura 3.2(a)), en comparación a los niveles alcanzados con el esquema probabilístico y el esquema basado en contador. Los niveles de retransmisiones ahorradas obtenidos en este esquema son los más bajos de todos los esquemas básicos, como se observa en la Figura 3.2(b). Esto se debe a que los valores de distancia umbral propuestos son muy bajos y, de acuerdo al algoritmo del esquema, las retransmisiones seguirán realizándose hasta que la distancia entre los nodos sea mayor a la distancia umbral. Los niveles de retransmisiones ahorradas que se obtienen en base a los valores de distancia umbral propuestos en [11] son cercanos al método de inundación. Por esta razón, se analizó este esquema con valores de distancia umbral superiores a los propuestos.

Los valores de distancia umbral que se analizaron en esta implementación parten desde el valor máximo que se analiza en [11] hasta el valor del rango de transmisión de los nodos (500m). Los resultados que se obtuvieron se muestran en la Figura 3.4. Se puede observar que los niveles de alcance presentan una disminución drástica para valores umbrales superiores a los 350m de distancia, como se observa en la Figura 3.4(a). Esto se debe a que estos valores se aproximan al radio de transmisión de los nodos y, en estos casos, solo los nodos más lejanos entre sí están autorizados a retransmitir el paquete. Esto se puede observar más claramente en los niveles de retransmisiones ahorradas (Figura 3.4(b)). Para valores de distancia umbral superiores a 350m, se puede obtener niveles de retransmisiones ahorradas cercanas al 99% en escenarios de alta densidad de nodos (1x1 3x3, 5x5) ya que, en estos escenarios, solo es necesario que un nodo de la red retransmita el paquete. A diferencia de los escenarios en los que los nodos se encuentran muy dispersos (7x7, 9x9, 11x11). Al implementar este esquema con valores de distancia umbral



(a)Alcance.



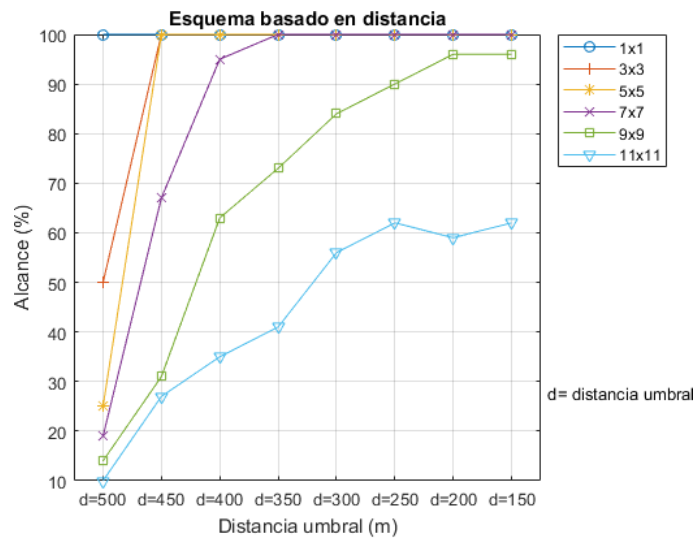
(b)Retransmisiones ahorradas.

Figure 3.3: Rendimiento del esquema basado en distancia.

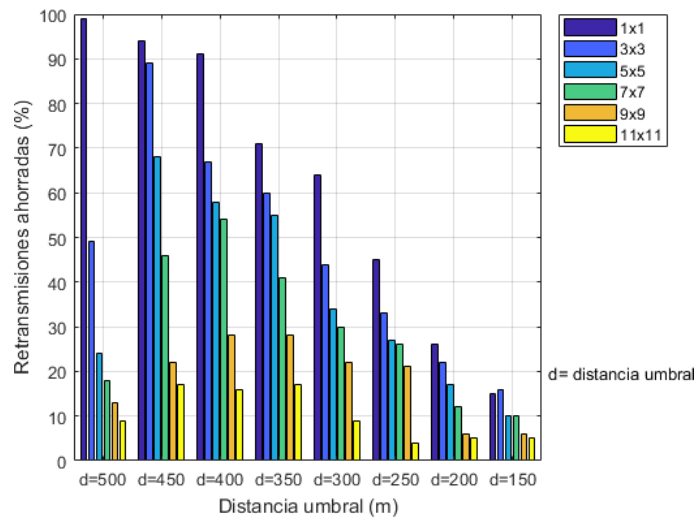
Fuente: Autor.

Elaborado por: Autor.

cercanos al rango de transmisión de los nodos provoca que las retransmisiones se reduzcan al mínimo, afectando al alcance que se logra en la red. Valores de distancia umbral superiores a 350 provocan que el algoritmo afecte a la red de forma negativa, por lo que no se recomienda emplear estos valores. Con valores entre 350m y 150m se puede conseguir entre el 70% y 100% de alcance en la red con el máximo porcentaje de retransmisiones ahorradas en cada escenario.



(a)Alcance.



(b)Retransmisiones ahorradas.

Figure 3.4: Rendimiento del esquema basado en distancia.

Fuente: Autor.

Elaborado por: Autor.

3.2. Implementación de algoritmos híbridos

Para la implementación de los algoritmos híbridos se utilizaron los parámetros generales y los escenarios de simulación expuestos en las Tablas 3.2 y 3.3, esto con el fin de realizar una comparación entre el rendimiento de los esquemas básicos y los algoritmos híbridos. Los resultados obtenidos se muestran a continuación.

3.2.1. Implementación del algoritmo basado en esquema probabilístico y contador

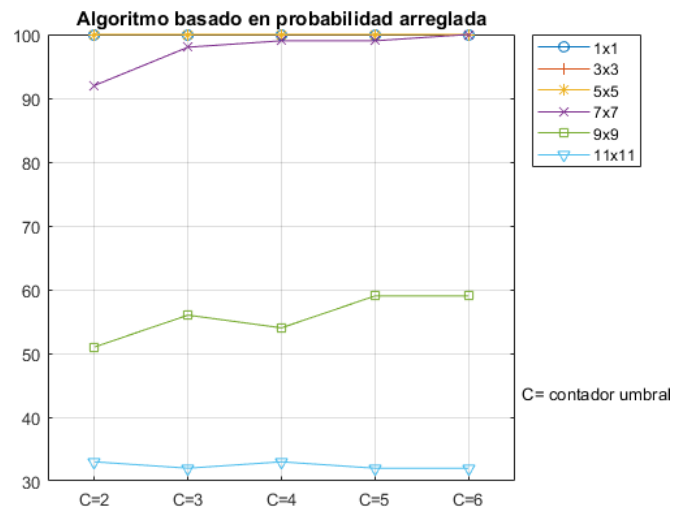
3.2.1.1. Resultados para el algoritmo basado en probabilidad arreglada

La implementación de este algoritmo se la realizó en dos partes: en la primera parte se seleccionó un valor de probabilidad umbral fija para toda la simulación ($P_{umb} = 0.6$) y se analizó el algoritmo para distintos valores de contador umbral C_{umb} de acuerdo a los valores que se analizan en la implementación del esquema basado en contador. Los resultados de este algoritmo se presentan en la Figura 3.5.

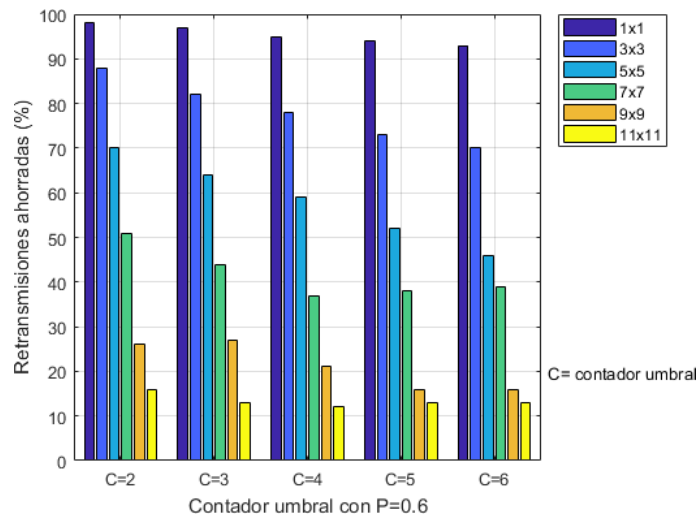
Se puede observar que al fijar un valor de probabilidad umbral para toda la simulación provoca que los niveles de alcance en la red se mantengan constantes para los diferentes valores de contador umbral que se evalúan, como se muestra en la Figura 3.5(a). Los resultados obtenidos de las retransmisiones ahorradas con este algoritmo (Figura 3.5(b)) son similares a los que se obtuvieron de la implementación del esquema basado en contador (Figura 3.2(b)), con un aumento en el porcentaje de retransmisiones ahorradas en escenarios amplios.

En la segunda parte (Figura 3.6), se seleccionó un valor de contador umbral fijo para toda la simulación $C_{umb}=2$. Con este valor se procedió a realizar la simulación en cada uno de los escenarios y para los distintos valores de probabilidad umbral P_{umb} de acuerdo a los valores analizados en el esquema probabilístico. Los resultados se presentan en la Figura 3.6.

Al fijar un valor de contador umbral para toda la red y evaluar el algoritmo para diferentes valores de probabilidad umbral provoca que los niveles de alcance en la red disminuyan cuando el valor de probabilidad umbral se aproxima a cero. Los niveles de retransmisiones ahorradas que se obtienen son similares a los resultados del esquema basado en contador. Dada la similitud de los resultados obtenidos de la implementación del esquema basado en contador (Figura 3.2) y del algoritmo basado en probabilidad arreglada (Figura 3.6), se puede concluir que, dentro de este algoritmo híbrido, el esquema basado en contador se encargar de reducir en mayor medida las retransmisiones redundantes, mientras que el esquema probabilístico trabaja como apoyo para el esquema basado en contador. Esto se puede observar en los niveles de retransmisiones ahorradas que se consiguen en los



(a)Alcance.



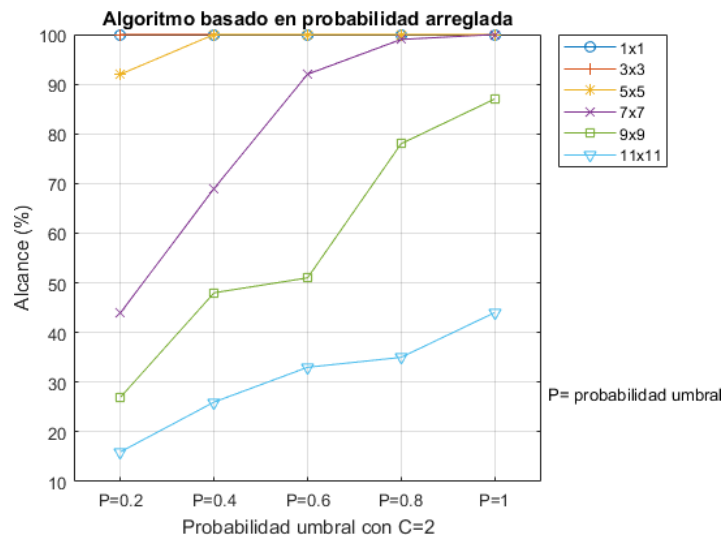
(b)Retransmisiones ahorradas.

Figure 3.5: Rendimiento del algoritmo con probabilidad arreglada.

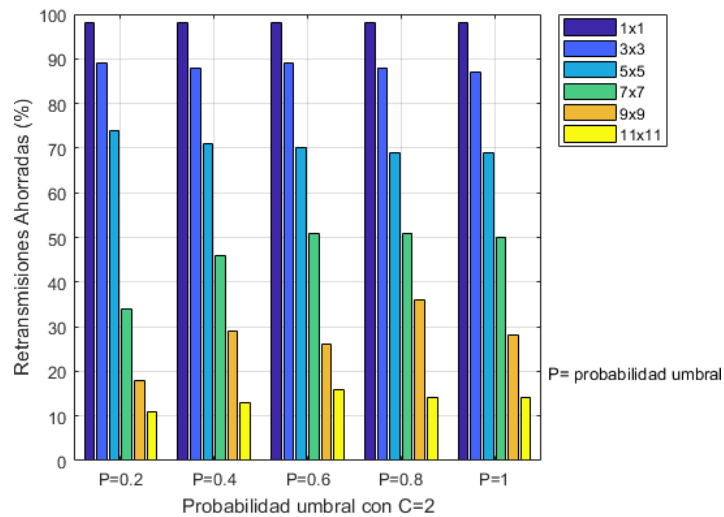
Fuente: Autor.

Elaborado por: Autor.

escenarios en los que los nodos se encuentran más dispersos (9x9, 11x11). De acuerdo a la Figura 3.2(b), las retransmisiones ahorradas conseguidas por el esquema basado en contador se aproxima al método de inundación (0%), mientras que en el algoritmo basado en probabilidad arreglada (Figura 3.6(b)) se pueden conseguir valores cercanos al 20%.



(a)Alcance.



(b)Retransmisiones ahorradas.

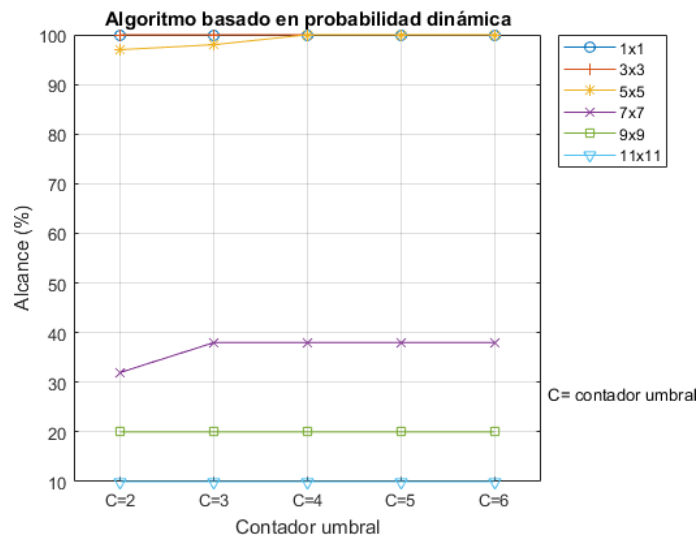
Figure 3.6: Rendimiento del algoritmo con probabilidad arreglada.

Fuente: Autor.

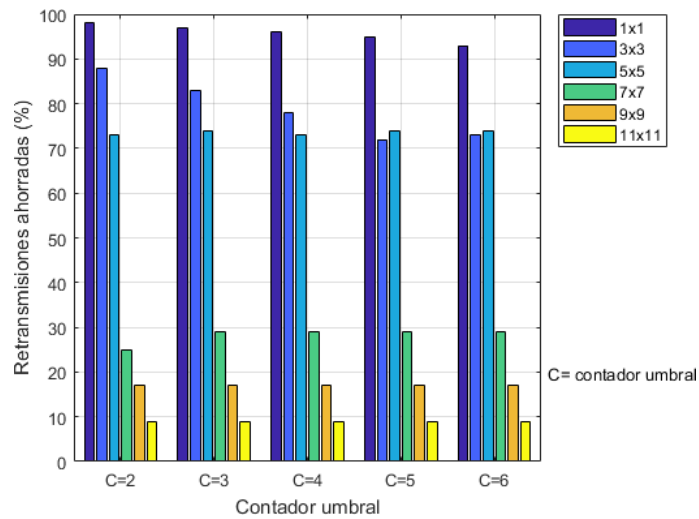
Elaborado por: Autor.

3.2.1.2. Resultados algoritmo basado en probabilidad dinámica

Para la implementación de este algoritmo se evaluaron los mismos valores de contador umbral que se analizaron para el esquema basado en contador. De los resultados que se muestran en la Figura 3.7 se puede concluir que el algoritmo basado en probabilidad dinámica presenta niveles de alcance constantes para los diferentes valores de contador umbral analizados al igual que los niveles de retransmisiones ahorradas. Para los escenar-



(a)Alcance.



(b)Retransmisiones ahorradas.

Figure 3.7: Rendimiento del algoritmo con probabilidad dinámica.

Fuente: Autor.

Elaborado por: Autor.

ios en los que existen una alta densidad de nodos (1x1, 3x3 y 5x5) los niveles de alcance en la red se mantienen cercanos al 100%, mientras que, para escenarios en los que los nodos se encuentran muy dispersos (7x7, 9x9, 11x11) el alcance disminuye drásticamente debido a que el valor de probabilidad umbral se calcula en base al número de nodos vecinos del nodo emisor. Los porcentajes de retransmisiones ahorradas que se alcanzan para escenarios de alta densidad de nodos se mantienen entre el 95% y el 70%.

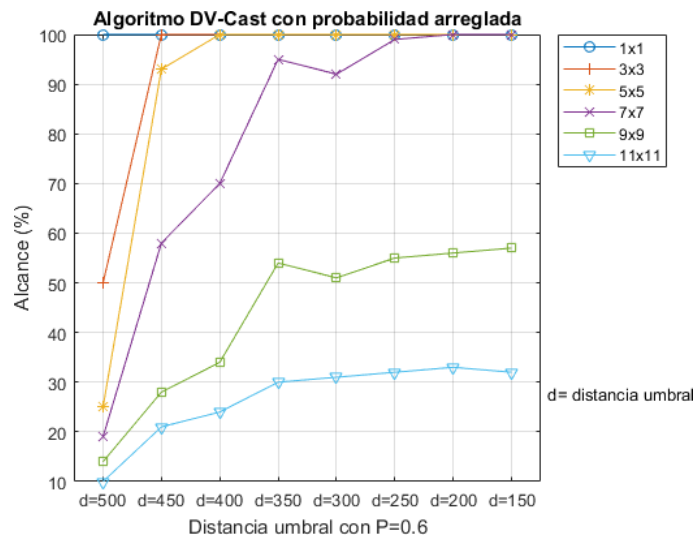
3.2.2. Implementación del algoritmo *DV-Cast*

3.2.2.1. Algoritmo *DV-Cast* con probabilidad fija

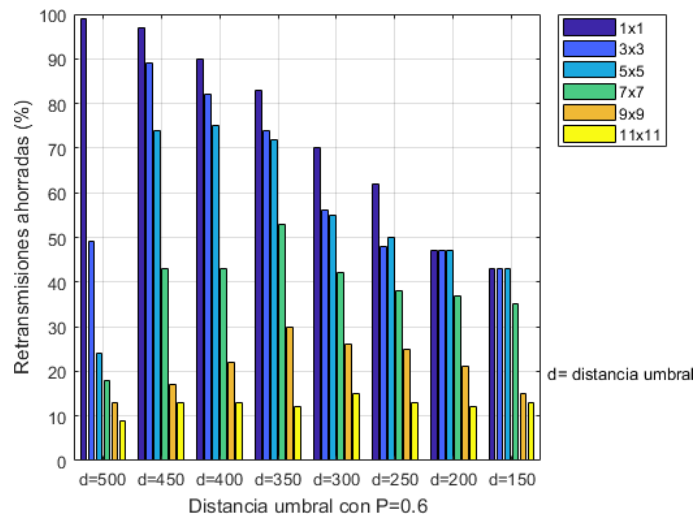
La implementación de este algoritmo se la dividió en dos partes: en la primera, se seleccionó un valor de probabilidad umbral fijo para toda la simulación $P_{umb}=0.6$. Con este valor se procedió a realizar la simulación en cada uno de los escenarios y para distintos valores de distancia umbral D_{min} de acuerdo a los valores analizados en la implementación del segundo caso del esquema basado en distancia. Los resultados se presentan en la Figura 3.8.

Se puede observar que para valores de distancia umbral superiores a 350m el porcentaje de alcance de la red disminuye. Como se explicó en la sección 3.1.3, esto se debe a que estos valores son cercanos al rango de transmisión de los nodos y en este caso, solo los nodos más alejados entre sí están permitidos de realizar retransmisiones en la red. Para valores de distancia umbral menores a 350, los porcentajes de alcance para los distintos escenarios se mantienen constantes.

En el segundo caso se seleccionó un valor de distancia umbral fija para toda la simulación ($D_{min}=150$). Con este valor se realizó la simulación en cada uno de los escenarios y para distintos valores de probabilidad umbral P_{umb} de acuerdo a los valores analizados en la implementación del esquema probabilístico. Los resultados se presentan en la Figura 3.9. De esta implementación se puede observar que los resultados obtenidos en el alcance de la red son similares a los obtenidos en la implementación del esquema basado en probabilidad (Figura 3.1). Mientras que los resultados obtenidos de los niveles de retransmisiones ahorradas son similares a los obtenidos de la implementación del esquema basado en distancia. Se puede concluir que en el algoritmo *DV-Cast* con probabilidad arreglada, el esquema basado en distancia domina al algoritmo y que el esquema probabilístico trabaja como apoyo para reducir la cantidad de retransmisiones redundantes que permite el esquema basado en distancia.



(a)Alcance.



(b)Retransmisiones ahorradas.

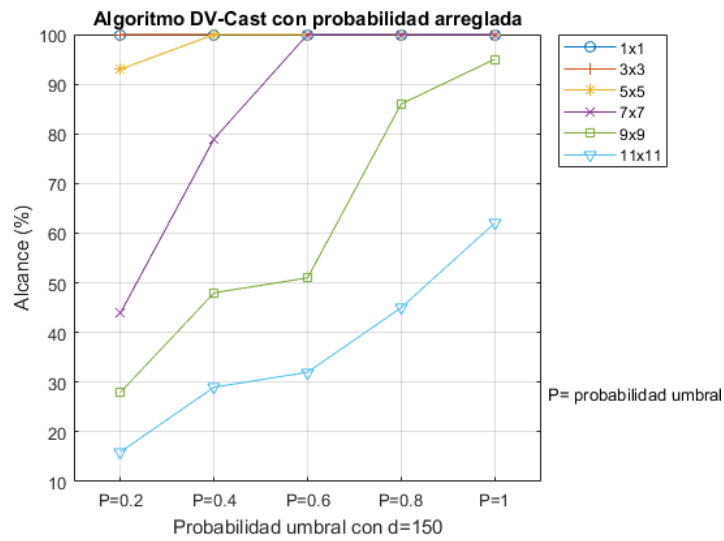
Figure 3.8: Rendimiento del algoritmo DV-Cast.

Fuente: Autor.

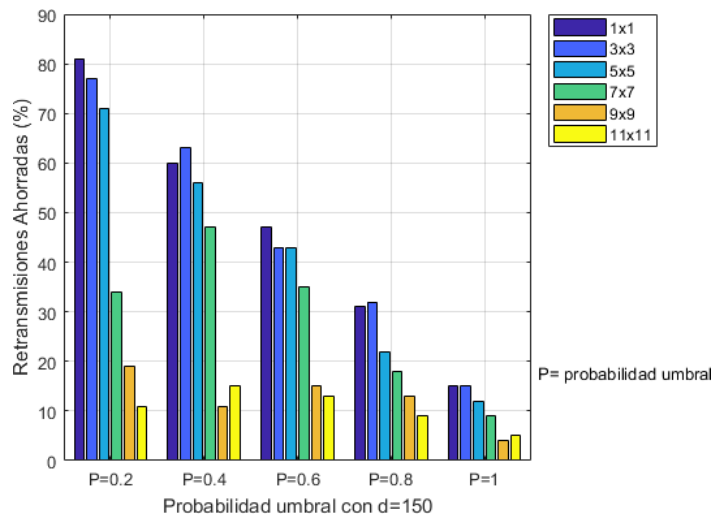
Elaborado por: Autor.

3.2.2.2. Algoritmo DV-Cast con probabilidad dinámica

De acuerdo con los resultados que se exponen en la Figura 3.10, con este algoritmo se pueden obtener valores máximos de alcance de la red hasta una distancia umbral de 400m. Para valores de distancia umbral mayores, los porcentajes de alcance disminuyen de manera drástica ya que estos valores son cercanos al radio de transmisión de los nodos. Los porcentajes de retransmisiones ahorradas pueden ir desde el 90% hasta el 50%. Este



(a) Alcance.



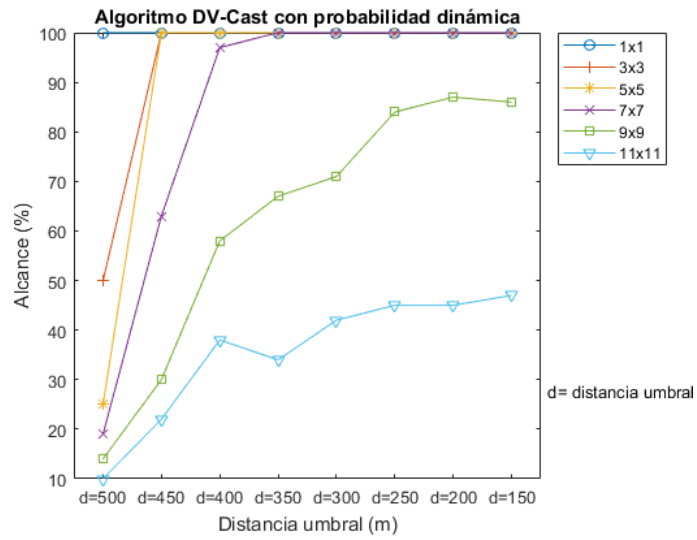
(b) Retransmisiones ahorradas.

Figure 3.9: Rendimiento del algoritmo DV-Cast.

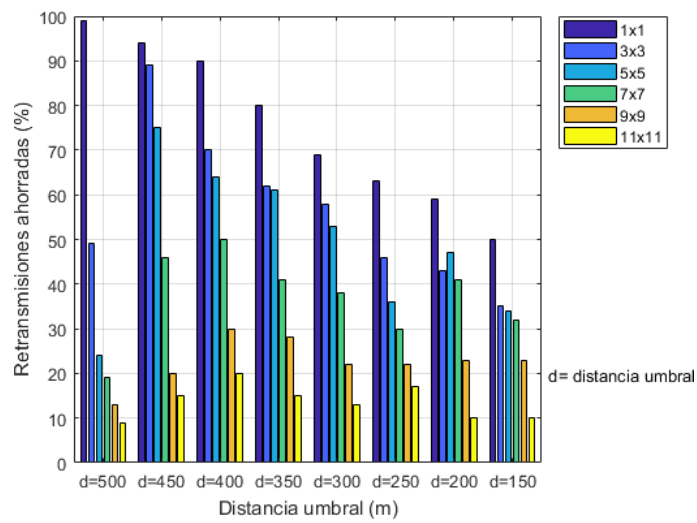
Fuente: Autor.

Elaborado por: Autor.

algoritmo permite tener mayor alcance en la red pero disminuye el porcentaje de retransmisiones ahorradas.



(a)Alcance.



(b)Retransmisiones ahorradas.

Figure 3.10: Rendimiento del algoritmo *DV-Cast* dinámico.

Fuente: Autor.

Elaborado por: Autor.

CAPÍTULO 4
ANÁLISIS DE RESULTADOS

4.1. Comparación de esquemas básicos

Analizando los resultados obtenidos de la implementación del esquema probabilístico (Figura 3.1), el esquema basado en contador (Figura 3.2) y el esquema basado en distancia (Figura 3.4), se puede determinar que con el esquema basado en contador se consigue un mayor porcentaje de retransmisiones ahorradas en escenarios donde la densidad de nodos es alta. En cambio, para escenarios en donde la densidad de nodos es baja, el porcentaje de retransmisiones ahorradas, con este algoritmo, es cercano al 0%. Los esquemas probabilístico y basado en distancia permiten ahorrar retransmisiones tanto en escenarios de alta y baja densidad de nodos pero en menor porcentaje en comparación al esquema basado en contador. Por esto se decidió implementar una nueva métrica de evaluación que permita determinar qué algoritmo reduce al máximo la cantidad de paquetes redundantes que se generan en la red. La nueva métrica se describe a continuación.

- **Sobrecarga de la red:** cantidad total de paquetes redundantes que se han generado en la red. Se la obtiene de la sumatoria de todos los contadores del mensaje de cada nodo.

Con esta nueva métrica se puede analizar el efecto del método de inundación dentro de los diferentes escenarios analizados en el presente trabajo. En la Figura 4.1 se muestra la sobrecarga de la red para cada uno de los escenarios de simulación al enviar un paquete de retransmisión. Se puede observar que el método de inundación provoca una excesiva cantidad de paquetes redundantes en escenarios donde la densidad de nodos es alta, lo que provoca una cantidad de retransmisiones redundantes en igual proporción. A medida que el área de simulación crece, la cantidad de paquetes redundantes disminuye, y de la misma manera la cantidad de retransmisiones redundantes.

En las Figuras 4.2, 4.3 y 4.4 se muestra la sobrecarga de la red para los esquemas probabilístico, basado en contador y basado en distancia respectivamente.

Con estos resultados se puede concluir que el esquema basado en contador es el esquema básico que reduce en mayor medida la cantidad de paquetes redundantes que se generan en la red. También se puede concluir que en el escenario 5x5 los algoritmos de los esquemas básicos trabajan de manera óptima en comparación a los demás escenarios. Esto se debe a que en este escenario se pueden obtener niveles altos de alcance de la

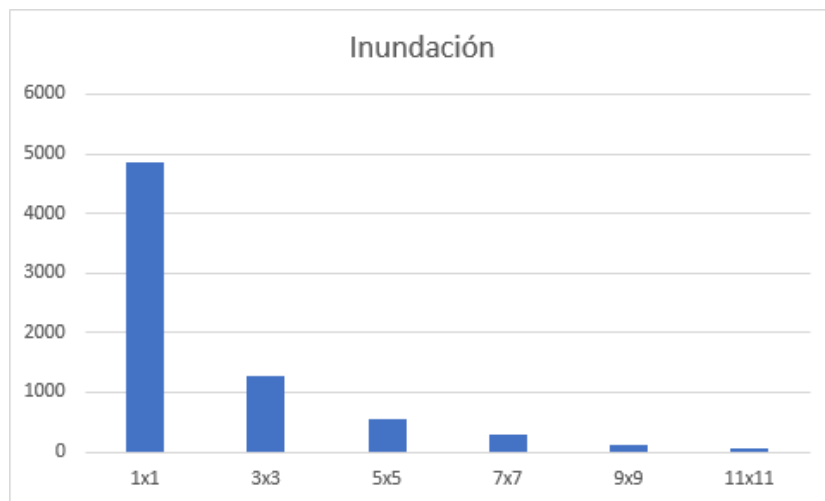


Figure 4.1: Sobrecarga de la red con el método de Inundación.

Fuente: Autor.

Elaborado por: Autor.

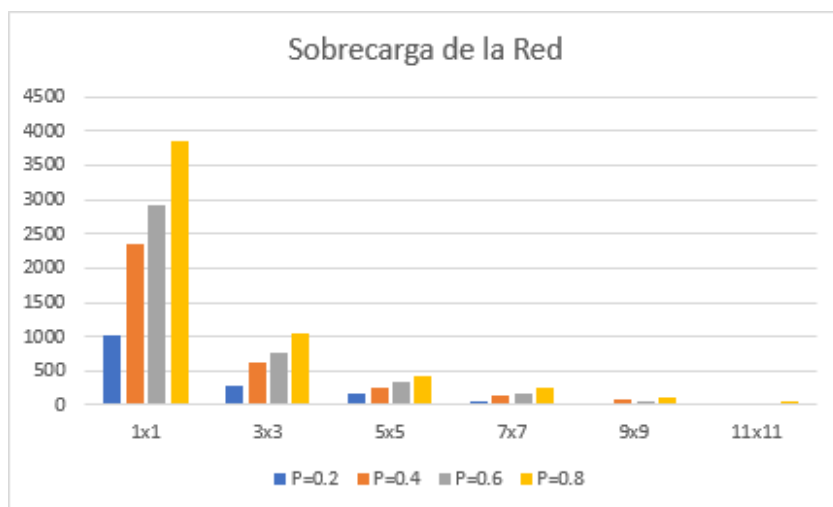


Figure 4.2: Sobrecarga de la red aplicando esquema probabilístico.

Fuente: Autor.

Elaborado por: Autor.

red, porcentajes de retransmisiones ahorradas cercanos al 70% y cantidad de paquetes redundantes mínimos para cada esquema.

4.2. Comparación de algoritmos híbridos

Luego de analizar los resultados obtenidos de la implementación de los algoritmos híbridos se puede determinar que:

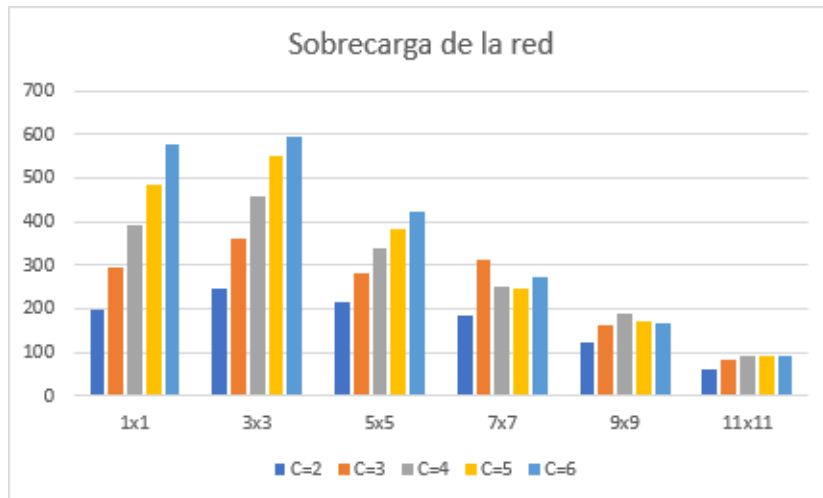


Figure 4.3: Sobrecarga de la red con esquema basado de contador.
Fuente: Autor.
Elaborado por: Autor.

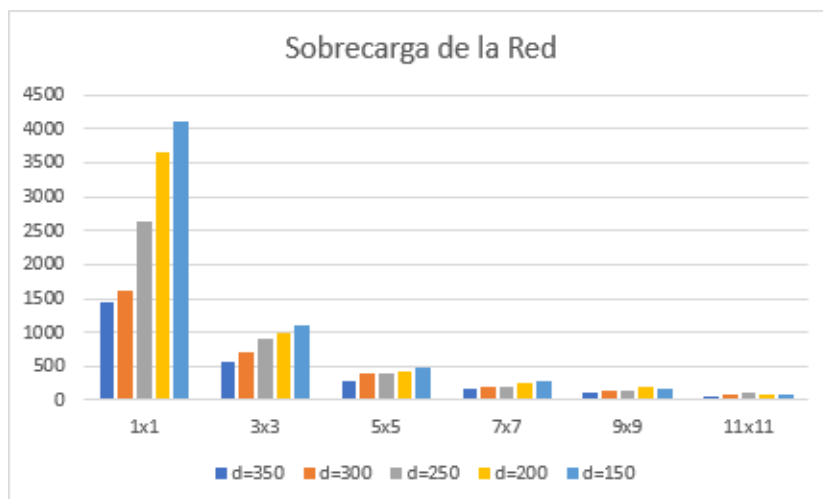


Figure 4.4: Sobrecarga de la red con esquema de distancia.
Fuente: Autor.
Elaborado por: Autor.

- De las dos variantes analizadas en la implementación del algoritmo basado en los esquemas probabilístico y contador, la variante que presenta mejores resultados es la que se basa en probabilidad dinámica. Con este algoritmo se pueden conseguir niveles de alcance constantes independientemente del valor de contador umbral que se utilice y el área de simulación, además de conseguir porcentajes de retransmisiones ahorradas más altos en comparación de los obtenidos en la implementación de los algoritmos basados en estos esquemas. Adicionalmente, este algoritmo puede adaptarse de manera au-

tomática a cualquier escenario de red que se esté analizando ya que el cálculo de la probabilidad umbral se basa en parámetros propios de la red.

- De las dos variantes analizadas en la implementación del algoritmo *DV-Cast* (basado en el esquema probabilístico y distancia), la variante que presenta mejores resultados es el algoritmo *DV-Cast* basado en probabilidad dinámica. Con este algoritmo se pueden conseguir niveles de alcance constantes independientemente del valor de distancia umbral que se utilice y el área de simulación, además de conseguir porcentajes de retransmisiones ahorradas más altos en comparación de los obtenidos en la implementación de los algoritmos basados en estos esquemas.

Al comparar los resultados tanto de alcance como retransmisiones ahorradas de la implementación del algoritmo basado en probabilidad dinámica y el algoritmo *DV-Cast* con probabilidad dinámica, son similares. Para determinar cuál de los dos algoritmos permite una mayor reducción de la cantidad de retransmisiones redundantes se implementó la métrica expuesta en la sección 4.1, cuyos resultados se muestran en las Figuras 4.5 y 4.6. Con

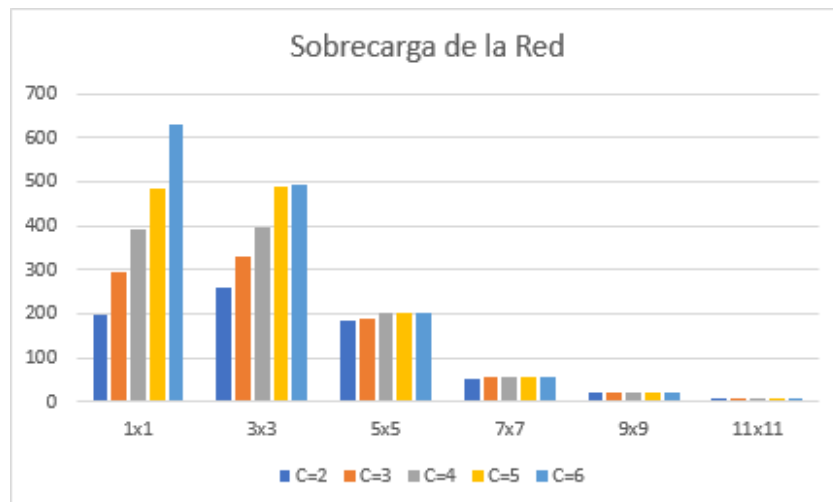


Figure 4.5: Sobrecarga con algoritmo de probabilidad dinámica.

Fuente: Autor.

Elaborado por: Autor.

estos resultados se puede concluir que el algoritmo basado en probabilidad dinámica es el algoritmo híbrido que permite reducir al mínimo la cantidad de paquetes redundantes que se generan en la red. Además, se puede observar que los algoritmos trabajan de manera óptima en el escenario 5x5, ya que, en escenarios de muy alta densidad de nodos (1x1 y

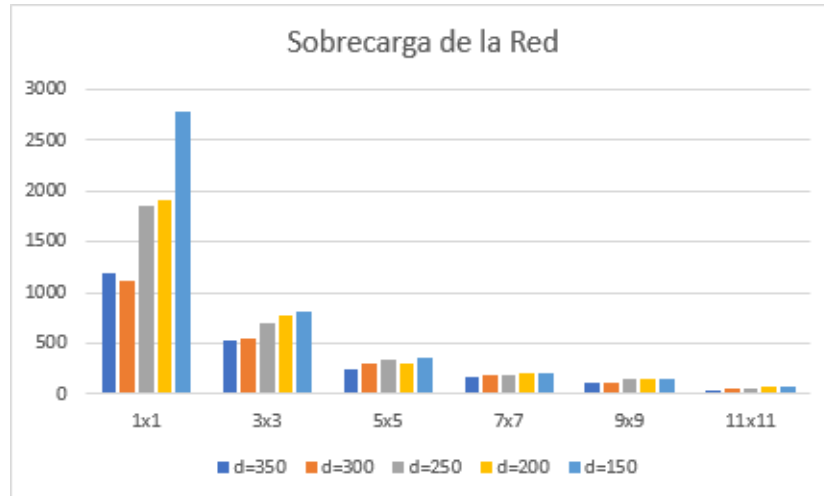


Figure 4.6: Sobrecarga de la red con algoritmo *DV-Cast* dinámico.

Fuente: Autor.

Elaborado por: Autor.

3x3), la cantidad de paquetes redundantes es muy alta y en escenarios en donde los nodos se encuentran muy dispersos (7x7, 9x9, 11x11) el alcance que se consigue en la red se reduce.

4.2.1. Cálculo del ahorro de potencia

Finalmente, se analizó el ahorro de potencia que se consigue con la implementación de estos algoritmos. Para esto, se dividió el área de simulación alrededor del nodo emisor del mensaje y obtener la cantidad de energía ahorrada por los nodos cercanos, intermedios y lejanos al nodo emisor del paquete. La división del mapa se encuentra ejemplificado en la Figura 4.7. El escenario en el cual se analizó el ahorro de potencia fue en el escenario 5x5, ya que en este escenario los algoritmos presentan los mejores resultados. El radio de simulación (R) es igual a la mitad de la longitud del mapa de simulación. Para el caso del escenario 5x5 y de acuerdo a la Tabla 3.3, el escenario tiene dimensiones de 2500x2500m, por lo que el radio de simulación es igual a 1250m. Todos los nodos que se encuentren dentro de un radio igual a un tercio del radio de simulación ($1/3 R$) serán considerados como los *nodos cercanos* al nodo emisor, los nodos que se encuentren entre un tercio ($1/3 R$) y dos tercios ($2/3 R$) del radio de simulación serán considerados como los nodos intermedios y todos aquellos nodos que se encuentren fuera del rango intermedio serán

considerados como los nodos lejanos al nodo emisor.

Los resultados del ahorro de potencia se presentan en la Figura 4.8. El algoritmo basado

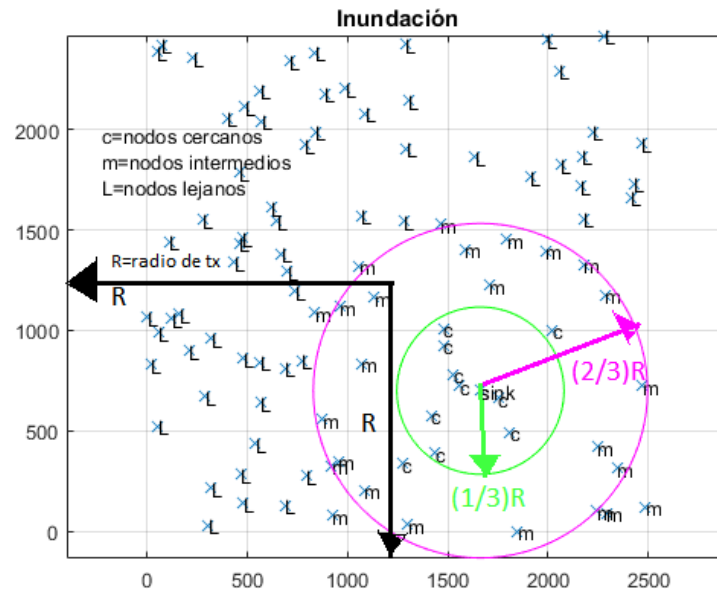


Figure 4.7: División del escenario.

Fuente: Autor.

Elaborado por: Autor.

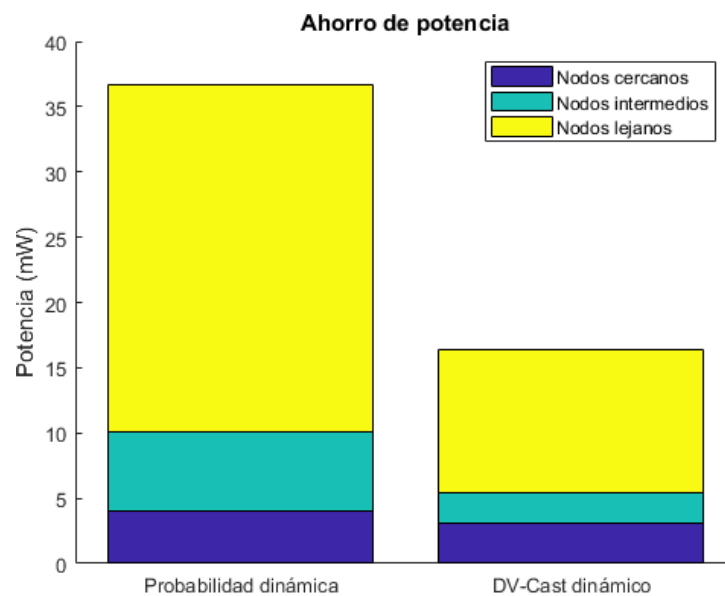


Figure 4.8: Ahorro de potencia de los algoritmos híbridos.

Fuente: Autor.

Elaborado por: Autor.

en probabilidad dinámica genera un ahorro de potencia a la red de cerca de 36mW mientras que el algoritmo DV-Cast con probabilidad dinámica genera un ahorro de cerca de 16mW. Estos resultados se obtuvieron en base a la metodología de cálculo de esta métrica, la cual se describe en la sección 2.4. Con esto se concluye que el algoritmo basado en probabilidad dinámica es el algoritmo que permite reducir en mayor medida los efectos del BSP en una red FANET garantizando el máximo alcance en la red y un mayor ahorro de potencia a los nodos. En la Tabla 4.1 se presenta una comparación de la potencia que se consume en la red con el método de inundación y con los diferentes algoritmos híbridos.

Tabla 4.1: Consumo de potencia de la red

	Inundación	Dv-Cast	Algoritmo probabilidad dinámica
Nodos cercanos	6,42mW	1,19mW	1,02mW
Nodos intermedios	10,01mW	8,74mW	2,49mW
Nodos lejanos	35,17mW	25,74mW	10,72mW
Total potencia consumida	51,6mW	35,67mW	14.23mW

Fuente: Autor.

Elaborado por: Autor.

CONCLUSIONES

- Se realizó una línea base sobre retransmisión de paquetes en redes FANET, en la cual se describe su concepto, características y el método de retransmisión de paquetes que se emplea por defecto. Además, se analiza el problema del BSP, efecto del método de inundación, los fenómenos que genera el BSP en la red y los métodos propuestos para mitigarlos.
- Se implementó una red FANET en el software de simulación MATLAB de acuerdo a los parámetros seleccionados y se incorporó un algoritmo de movilidad para la simulación del movimiento de los nodos.
- Se implementaron los esquemas básicos (probabilístico, basado en contador y distancia) propuestos para reducir el problema del BSP en la red y se analizó la eficiencia de cada esquema en base a las métricas seleccionadas.
- Se implementaron algoritmos híbridos de retransmisión de paquetes basados en los esquemas básicos propuestos para reducir el problema del BSP y se analizó la eficiencia de cada uno de acuerdo a las métricas seleccionadas.
- El mejor esquema básico para reducir el BSP es el esquema basado en contador, ya que se pueden ahorrar entre el 10% y el 90% de retransmisiones y mantener niveles constantes de alcance en la red.
- El algoritmo híbrido que presentó mejores resultados para mitigar los problemas que genera el BSP es el algoritmo basado en probabilidad dinámica, con la implementación de este algoritmo se obtuvieron porcentajes de alcance constantes, un ahorro de retransmisiones entre el 70% y el 90% y se consiguió un ahorro de potencia del 70%.

RECOMENDACIONES

- Para escenarios en donde los nodos se encuentren dispersos, y debido a que la distribución inicial de los nodos se realiza de manera aleatoria, se recomienda realizar múltiples simulaciones para distintas distribuciones iniciales con el fin de encontrar el escenario en donde los algoritmos presenten mejores resultados.
- Se recomienda determinar el nodo con el mayor número de nodos vecinos al comienzo de la simulación y designarlo como el nodo que inicia la transmisión del paquete. Esto con el fin de que el paquete se propague y obtener un mayor porcentaje de alcance.
- Debido a que cada escenario de simulación tiene límites de distancia, se recomienda controlar la movilidad de los nodos para evitar que salgan del área de simulación y se obtengan resultados erróneos.
- Se recomienda controlar las variables aleatorias de la simulación para evitar generar resultados diferentes cada vez que se simule la red.

Como trabajos futuros se propone:

- Como el presente trabajo se lo realizó en dos dimensiones, se propone realizar la simulación de la red en tres dimensiones para evaluar el efecto de la altura en los cálculos de la distancia y la movilidad de la red y tener un modelo más cercano a la realidad.
- Analizar el problema del BSP y el rendimiento de los algoritmos híbridos al implementar distintos algoritmos de movilidad.
- Desarrollar un algoritmo para calcular la cantidad de nodos necesaria de acuerdo al área de trabajo de la red. Para así evitar escenarios con alta densidad de nodos o escenarios en donde los nodos se encuentren dispersos.

BIBLIOGRAFÍA

- [1] I. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying ad-hoc networks (fanets): A survey," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013.
- [2] T. Saeed, Y. Mylonas, A. Pitsillides, V. Papadopoulou, and M. Lestas, "Modeling probabilistic flooding in vanets for optimal rebroadcast probabilities," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 2, pp. 556–570, 2018.
- [3] A. Guillen-Perez and M.-D. Cano, "Flying ad hoc networks: A new domain for network communications," *Sensors*, vol. 18, no. 10, p. 3571, 2018.
- [4] I. Maza, F. Caballero, J. Capitán, J. R. Martínez-de Dios, and A. Ollero, "Experimental results in multi-uav coordination for disaster management and civil security applications," *Journal of intelligent & robotic systems*, vol. 61, no. 1-4, pp. 563–585, 2011.
- [5] L. Merino, F. Caballero, J. R. Martínez-De-Dios, I. Maza, and A. Ollero, "An unmanned aircraft system for automatic forest fire monitoring and measurement," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1-4, pp. 533–548, 2012.
- [6] B. Bethke, J. How, and J. Vian, "Multi-uav persistent surveillance with communication constraints and health management," in *AIAA Guidance, Navigation, and Control Conference*, 2009, p. 5654.
- [7] P.-M. Olsson, J. Kvarnström, P. Doherty, O. Burdakov, and K. Holmberg, "Generating uav communication networks for monitoring and surveillance," in *2010 11th International Conference on Control Automation Robotics & Vision*. IEEE, 2010, pp. 1070–1077.
- [8] J. G. Manathara, P. Sujit, and R. W. Beard, "Multiple uav coalitions for a search and prosecute mission," *Journal of Intelligent & Robotic Systems*, vol. 62, no. 1, pp. 125–158, 2011.
- [9] J. Scherer, S. Yahyanejad, S. Hayat, E. Yanmaz, T. Andre, A. Khan, V. Vukadinovic, C. Bettstetter, H. Hellwagner, and B. Rinner, "An autonomous multi-uav system for

- search and rescue,” in *Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use*. ACM, 2015, pp. 33–38.
- [10] R. de Melo Pires, S. Z. Arnosti, A. S. R. Pinto, and K. R. Branco, “Experimenting broadcast storm mitigation techniques in fanets,” in *2016 49th Hawaii International Conference on System Sciences (HICSS)*. IEEE, 2016, pp. 5868–5877.
- [11] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, “The broadcast storm problem in a mobile ad hoc network,” *Wireless networks*, vol. 8, no. 2-3, pp. 153–167, 2002.
- [12] S. Najafzadeh, N. B. Ithnin, and S. Abd Razak, “Broadcasting in connected and fragmented vehicular ad hoc networks,” *International Journal of Vehicular Technology*, vol. 2014, 2014.
- [13] R. D. M. Pires, A. S. R. Pinto, and K. R. L. J. C. Branco, “The broadcast storm problem in fanets and the dynamic neighborhood-based algorithm as a countermeasure,” *IEEE Access*, vol. 7, pp. 59 737–59 757, 2019.
- [14] O. K. Tonguz, N. Wisitpongphan, and F. Bai, “Dv-cast: A distributed vehicular broadcast protocol for vehicular ad hoc networks,” *IEEE Wireless Communications*, vol. 17, no. 2, pp. 47–57, 2010.
- [15] R. S. Schwartz, R. R. Barbosa, N. Meratnia, G. Heijenk, and H. Scholten, “A simple and robust dissemination protocol for vanets,” in *2010 European Wireless Conference (EW)*. IEEE, 2010, pp. 214–222.
- [16] A. Bujari, C. T. Calafate, J.-C. Cano, P. Manzoni, C. E. Palazzi, and D. Ronzani, “Flying ad-hoc network application scenarios and mobility models,” *International Journal of Distributed Sensor Networks*, vol. 13, no. 10, p. 1550147717738192, 2017.
- [17] K. Kumari, B. Sah, and S. Maakar, “A survey: different mobility model for fanet,” *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 6, 2015.
- [18] J. M. H. Rábanos, L. M. Tomas, and J. M. R. Salis, *Transmisión por radio*. Editorial Universitaria Ramón Areces, 2013.

- [19] Ó. R. García, L. M. J. García, L. P. Castelló, A. G. Aparicio, and A. P. Vidal, *MATLAB: conceptos básicos y descripción gráfica*. Universidad Miguel Hernández, 2018.
- [20] DJI. (2017) Inspire 2 specs. <https://www.dji.com/inspire-2/info#specs>.
- [21] ——. (2017) Dji lightbridge 2 specs. <https://www.dji.com/lightbridge-2/info#specs>.

ANEXOS

ANEXO A. DATASHEET DRON INSPIRE 2

En la Tabla A.1 se detallan algunas de las especificaciones del dron Inspire 2. La lista completa de las características del dron se encuentran disponibles en [20]

Tabla A.1: Consumo de potencia de la red

Inspire 2 Specifications			
Modelo	T650A	Consumo	87,58Wh
Frecuencia de operación	2,400-2,483GHz 5,725-5,850GHz	Capacidad	4280mAh
Distancia máxima de transmisión	2,4GHz: 7Km 5,8GHz: 7Km	Sistema de comunicación	DJI Lightbridge 2
EIRP	2,4GHz: 26dBm 5,8GHz: 28dBm	Fuente de energía	Baterías

Fuente: [20].

Elaborado por: Autor.

ANEXO B. DATASHEET SISTEMA DE TRANSMISION LIGHTBRIDGE 2

En la Tabla B.1 se detallan algunas de las especificaciones del sistema de comunicación Lightbridge 2, el cual está incorporado en el dron Inspire 2. La lista completa de las características del sistema se encuentran disponibles en [21]

Tabla B.1: Consumo de potencia de la red

Especificaciones Lightbridge 2			
Ganancia de las antenas	2dBi a 2,450GHz	Conector de antena	MMCX Male
Frecuencia de operación	2,400-2,483GHz 5,725-5,850GHz	Tipo de antena	Omnidireccional
Distancia máxima de transmisión	2,4GHz: 7Km 5,8GHz: 7Km	Umbral de recepción	-110dB
EIRP	100mW a 2,4GHz	Fuente de energía	Baterías

Fuente: [21].

Elaborado por: Autor.

ANEXO C. CÓDIGO PRINCIPAL PARA LA SIMULACIÓN DE LA RED

```

1 %% Script principal para simulacion de red FANET%%
2 % function [] = programa_principal(p, w)
3 clear all;
4 close all;
5 clc
6 %%%%Variables generales de simulacion%%%%%%%%
7 randn('state', 80);
8 rand('state', 4); % permite generar resultados similares
9 %sinkNode = p;
10 global numOfNodes
11 numOfNodes = 100; % Numero de nodos de la red
12 global envSize
13 global txRange
14 txRange = 500; % Rango de transmision de los nodos
15 global escenario
16 escenario = 3; % Escenarios de prueba
17 switch escenario
18     case 1
19         envSize = 500;
20     case 2
21         envSize = 1500;
22     case 3
23         envSize = 2500;
24     case 4
25         envSize = 3500;
26     case 5
27         envSize = 4500;
28     case 6
29         envSize = 5500;
30     end
31 global floodProb; % probabilidad umbral de sistema
32 global savedTransmission; % porcentaje de retransmisiones ahorradas
33 global reTransmission; % porcentaje de retransmisiones realizadas
34
35 %%%%Generacion de los nodos de la red%%%%%%%%

```

```

36 saved Transmission = 0 ;
37 re Transmission = 0;
38 global xLocation;% Matriz que contiene la posicion en x de los nodos
39 global yLocation;% Matriz que contiene la posicion en y de los nodos
40 x Location = rand ( numOfNodes , 1 ) * envSize ;
41 y Location = rand ( numOfNodes , 1 ) * envSize;
42
43
44 % Generacion de la matriz de distancia entre los nodos
45 global distMatrix
46 distMatrix = zeros (numOfNodes ,numOfNodes) ;
47 for i=1:numOfNodes
48     for j=1:numOfNodes
49         distMatrix(i , j)=sqrt (( xLocation ( i)–xLocation ( j))^2 + (yLocation ( i)–
50             yLocation ( j))^2);
51     end
52 end
53 % Generacion de la matriz de adyacencia para determinar la
54 % conectividad entre los nodos
55 global connMatrix ;
56 connMatrix = ( distMatrix < txRange);
57
58 % Inicio de la transmision por parte de un nodo emisor
59 global sinkNode
60 sinkNode = 1; % Nodo emisor del paquete de retransmision
61
62 %%%Division del mapa para calculo de la energia ahorrada por nodo
63 centro = envSize / 2;
64 radio = centro / 3;
65 lejanos = zeros ;
66 intermedios = zeros;
67 cercanos = zeros;
68 for i = 1 : numOfNodes
69     if ( xLocation ( i) < ( xLocation ( sinkNode ) + radio ) ) && ( xLocation ( i ) > ( xLocation (
70         sinkNode ) – radio ) )
71         if ( yLocation ( i ) < ( yLocation ( sinkNode ) + radio ) ) && ( yLocation ( i ) > (
72             yLocation ( sinkNode ) – radio ) )
73             cercanos ( end+1 ) = i;

```

```

72         elseif (yLocation ( i ) < (yLocation (sinkNode) + (2*radio))) && (yLocation ( i )
73             > (yLocation (sinkNode) - (2*radio)))
74             intermedios ( end+1) = i ;
75         else
76             lejanos (end+1) = i ;
77         end
78     elseif (xLocation ( i ) < (xLocation (sinkNode) + (2*radio))) && (xLocation ( i ) > (
79         xLocation (sinkNode) - (2*radio)))
80         if (yLocation ( i ) < (yLocation (sinkNode) + (2*radio))) && (yLocation ( i ) > (
81             yLocation (sinkNode) - (2*radio)))
82             intermedios ( end+1) = i ;
83         else
84             lejanos ( end+1) = i ;
85         end
86     else
87         lejanos ( end+1) = i ;
88     end
89 % Matriz de nodos que han recibido el paquete
90 global visited ;
91 visited = zeros (1 , numOfNodes) ;
92 % Matriz de contador de mensaje por nodo
93 global contador
94 contador = zeros ( 1 , numOfNodes) ;
95 % Matriz de distancia entre nodos
96 global distancia
97 distancia = zeros (1 , numOfNodes) ;
98 % Matriz de energia ahorrada por nodo
99 global energia
100 energia = zeros ( 1 , numOfNodes) ;
101
102 % Topologia inicial de la red
103 figure (1) ;
104 plot (xLocation , yLocation , 'x') ;
105 title ( ' Flooding ' )
106 text (xLocation (sinkNode) , yLocation (sinkNode) , ' sink ' ) ;
107 for i = 2 : length (ceranos)
108     text (xLocation (ceranos ( i ) ) , yLocation (ceranos ( i ) ) , ' c ' )

```

```

109 end
110 for i = 2:length(intermedios)
111     text(xLocation(intermedios(i)), yLocation(intermedios(i)), 'm')
112 end
113 for i = 2:length(lejanos)
114     text(xLocation(lejanos(i)), yLocation(lejanos(i)), 'L')
115 end
116 pos = [(xLocation(sinkNode)-radio)(yLocation(sinkNode)-radio)(radio*2)(radio*2)];
117 rectangle('Position', pos, 'Curvature', [1 1], 'EdgeColor', 'g')
118 pos = [(xLocation(sinkNode)-(radio*2))(yLocation(sinkNode)-(radio*2))(radio*4)(radio
    *4)];
119 rectangle('Position', pos, 'Curvature', [1 1], 'EdgeColor', 'm')
120 axis equal
121
122 % inicializacion de las variables del nodo emisor
123 visited(sinkNode) = 1;
124 contador(sinkNode) = 1;
125 hold on
126 % Se llama al script de retransmision
127 DFS_movil(sinkNode);
128 hold off
129 % calculo del porcentaje de alcance de la red
130 global reachability
131 reachability = 100*sum(visited==1)/numOfNodes;
132 % calculo de la sobrecarga de la red
133 total_cont = sum(contador);
134 % calculo de la energia ahorrada de acuerdo a los nodos
135 % cercanos, intermedios y lejanos
136 ener_cercanos = 0;
137 ener_intermedios = 0;
138 ener_lejanos = 0;
139 for i = 2:length(cercanos)
140     ener_cercanos = ener_cercanos + energia(cercanos(i));
141 end
142 for i = 2:length(intermedios)
143     ener_intermedios = ener_intermedios + energia(intermedios(i));
144 end
145 for i = 2:length(lejanos)
146     ener_lejanos = ener_lejanos + energia(lejanos(i));
147 end

```

```
148 % total de energia ahorrada por la red
149 total_ener = sum( energia );
150
151 saved Transmission ;
152 re Transmission ;
153 return ;
154 % end
```


ANEXO D. CÓDIGO PARA LA RETRANSMISIÓN DE LOS PAQUETES

```
1 function [] = DFS_movil ( r )
2 global connMatrix ;
3 global visited ;
4 global xLocation ;
5 global yLocation ;
6 global floodProb ;
7 global savedTransmission ;
8 global reTransmission ;
9 global contador
10 global distancia
11 global energia
12 % Se obtienen todos los nodos vecinos del nodo emisor
13 neighborNodes = find ( conn Matrix ( r , : ) == 1 ) ;
14 % Se obtienen los nodos vecinos que no han
15 % recibido el paquete de retransmision
16 neighborNodes = intersect ( neighborNodes , find ( visited ( : ) == 0 ) ) ;
17 % Se calcula las distancias entre los nodos y se
18 % incrementa el valor de contador del mensaje por nodo
19 for k = 1 : length ( neighborNodes )
20     line ( [ xLocation ( r ) xLocation ( neighborNodes ( k ) ) ] , [ yLocation ( r ) yLocation (
        neighborNodes ( k ) ) ] ) ;
21     distancia ( neighborNodes ( k ) ) = sqrt ( ( xLocation ( r ) - xLocation ( neighborNodes ( k ) ) ) ^ 2
        + ( yLocation ( r ) - yLocation ( neighborNodes ( k ) ) ) ^ 2 ) ;
22     contador ( neighborNodes ( k ) ) = contador ( neighborNodes ( k ) ) + 1 ;
23 end
24 drawnow ;
25 % Se actualiza la matriz de nodos que han recibido el paquete
26 for k = 1 : length ( neighborNodes )
27     if ( visited ( neighborNodes ( k ) ) == 0 )
28         visited ( neighborNodes ( k ) ) = 1 ;
29         floodProb = 0.8 ; % probabilidad umbral
30         p = rand ; % probabilidad asignada al paquete
31         prx = 10 - 101 ; % potencia receptor
32         lc = 0.5 ; % perdidas por conectores rx y tx
33         gtr = 2 ; % ganancia rx y tx
```

```

34         if p <= flood Prob
35             % llamado a l s c r i p t para l a movilidad de l o s nodos movimiento (
36             );
37             % se retransmite
38             DFS_movil ( neighborNodes ( k ) );
39             re Transmission = re Transmission + 1 ;
40         else
41             % se a c t u a l i z a e l v a l o r de retransmsiones ahorradas saved
42             Transmission = saved Transmission + 1 ;
43             % se c a l c u l a l a energia ahorrada en l a r e t r a n s m i s i o n
44             lb = 32.45 + 20*(log10(2400)) + 20*(log10(distancia(
                    neighborNodes(k))/1000));
45             ptx = prx + lc - gtr - gtr + lc + lb;
46             ptx = 10^(ptx/10);
47             energia ( neighborNodes ( k ) ) = ptx + energia ( neighborNodes ( k ) );
48         end
49     end
50 end

```

ANEXO E. CÓDIGO ALGORITMO RANDOM WALK

```
1 function movimiento ()
2 global xLocation
3 global yLocation
4 global connMatrix
5 global escenario
6 global numOfNodes
7 global distMatrix
8 global txRange
9 % script para generar Random Walk para todos los nodos
10 switch escenario
11     case 1
12         for x = 1 : numOfNodes
13             signox = sign ( randn ) ;
14             pasox = signox * ( rand ( 1 , 1 ) * 3 ) ; mov
15             = x Location ( x ) + pasox ;
16             if mov < 0
17                 xLocation ( x ) = ( pasox * ( -1 ) ) + xLocation ( x ) ;
18             elseif mov > 500
19                 x Location ( x ) = ( pasox * ( -1 ) ) + x Location ( x ) ;
20             else
21                 x Location ( x ) = mov ;
22             end
23             signoy = sign ( randn ) ;
24             pasoy = signoy * ( rand ( 1 , 1 ) * 3 ) ; mov
25             = y Location ( x ) + pasoy ;
26             if mov < 0
27                 yLocation ( x ) = ( pasoy * ( -1 ) ) + yLocation ( x ) ;
28             elseif mov > 500
29                 y Location ( x ) = ( pasoy * ( -1 ) ) + y Location ( x ) ;
30             else
31                 y Location ( x ) = mov ;
32             end
33         end
34     case 2
35         for x = 1 : numOfNodes
```

```

36         signox = sign ( randn ) ;
37         pasox = signox * ( rand ( 1 , 1 ) * 8 ) ; mov
38         = x Location ( x ) + pasox ;
39         if mov < 0
40             xLocation ( x ) = ( pasox * ( -1 ) ) + xLocation ( x ) ;
41         elseif mov > 1500
42             x Location ( x ) = ( pasox * ( -1 ) ) + x Location ( x ) ;
43         else
44             x Location ( x ) = mov ;
45         end
46         signoy = sign ( randn ) ;
47         pasoy = signoy * ( rand ( 1 , 1 ) * 8 ) ; mov
48         = y Location ( x ) + pasoy ;
49         if mov < 0
50             yLocation ( x ) = ( pasoy * ( -1 ) ) + yLocation ( x ) ;
51         elseif mov > 1500
52             y Location ( x ) = ( pasoy * ( -1 ) ) + y Location ( x ) ;
53         else
54             y Location ( x ) = mov ;
55         end
56     end
57 case 3
58     for x = 1 : numOfNodes
59         signox = sign ( randn ) ;
60         pasox = signox * ( rand ( 1 , 1 ) * 14 ) ;
61         mov = x Location ( x ) + pasox ;
62         if mov < 0
63             xLocation ( x ) = ( pasox * ( -1 ) ) + xLocation ( x ) ;
64         elseif mov > 2500
65             xLocation ( x ) = ( pasox * ( -1 ) ) + xLocation ( x ) ;
66         else
67             xLocation ( x ) = mov ;
68         end
69         signoy = sign ( randn ) ;
70         pasoy = signoy * ( rand ( 1 , 1 ) * 14 ) ;
71         mov = y Location ( x ) + pasoy ;
72         if mov < 0
73             yLocation ( x ) = ( pasoy * ( -1 ) ) + yLocation ( x ) ;
74         elseif mov > 2500
75             yLocation ( x ) = ( pasoy * ( -1 ) ) + yLocation ( x ) ;

```

```

76         else
77             y Location ( x ) = mov ;
78         end
79     end
80 case 4
81     for x = 1 : numOfNodes
82         signox = sign ( randn ) ;
83         pasox = signox * ( rand ( 1 , 1 ) * 19 ) ; mov
84         = x Location ( x ) + pasox ;
85         if mov < 0
86             xLocation ( x ) = ( pasox * ( -1 ) ) + xLocation ( x ) ;
87         elseif mov > 3500
88             x Location ( x ) = ( pasox * ( -1 ) ) + x Location ( x ) ;
89         else
90             x Location ( x ) = mov ;
91         end
92         signoy = sign ( randn ) ;
93         pasoy = signoy * ( rand ( 1 , 1 ) * 19 ) ; mov
94         = y Location ( x ) + pasoy ;
95         if mov < 0
96             yLocation ( x ) = ( pasoy * ( -1 ) ) + yLocation ( x ) ;
97         elseif mov > 3500
98             y Location ( x ) = ( pasoy * ( -1 ) ) + y Location ( x ) ;
99         else
100             y Location ( x ) = mov ;
101         end
102     end
103 case 5
104     for x = 1 : numOfNodes
105         signox = sign ( randn ) ;
106         pasox = signox * ( rand ( 1 , 1 ) * 25 ) ; mov
107         = x Location ( x ) + pasox ;
108         if mov < 0
109             xLocation ( x ) = ( pasox * ( -1 ) ) + xLocation ( x ) ;
110         elseif mov > 4500
111             x Location ( x ) = ( pasox * ( -1 ) ) + x Location ( x ) ;
112         else
113             x Location ( x ) = mov ;
114         end
115         signoy = sign ( randn ) ;

```

```

116         pasoy = signoy * ( rand ( 1 , 1 ) * 25 ) ; mov
117         = y Location ( x ) + pasoy ;
118         if mov < 0
119             yLocation ( x ) = ( pasoy * ( -1 ) ) + yLocation ( x ) ;
120             elseif mov > 4500
121                 y Location ( x ) = ( pasoy * ( -1 ) ) + y Location ( x ) ;
122             else
123                 y Location ( x ) = mov ;
124             end
125         end
126     case 6
127         for x = 1 : numOfNodes
128             signox = sign ( randn ) ;
129             pasox = signox * ( rand ( 1 , 1 ) * 31 ) ; mov
130             = x Location ( x ) + pasox ;
131             if mov < 0
132                 xLocation ( x ) = ( pasox * ( -1 ) ) + xLocation ( x ) ;
133             elseif mov > 5500
134                 x Location ( x ) = ( pasox * ( -1 ) ) + x Location ( x ) ;
135             else
136                 x Location ( x ) = mov ;
137             end
138             signoy = sign ( randn ) ;
139             pasoy = signoy * ( rand ( 1 , 1 ) * 31 ) ; mov
140             = y Location ( x ) + pasoy ;
141             if mov < 0
142                 yLocation ( x ) = ( pasoy * ( -1 ) ) + yLocation ( x ) ;
143             elseif mov > 5500
144                 y Location ( x ) = ( pasoy * ( -1 ) ) + y Location ( x ) ;
145             else
146                 y Location ( x ) = mov ;
147             end
148         end
149     end
150 %se actualiza la matriz de distancias con los nuevos valores
151 %luego del movimiento de los nodos
152 for i = 1 : numOfNodes
153     for j = 1 : numOfNodes
154         distMatrix ( i , j ) = sqrt ( ( xLocation ( i ) - xLocation ( j ) ) ^ 2 + ( yLocation ( i ) -
            yLocation ( j ) ) ^ 2 ) ;

```

```
155155         end
156156     end
157157 se actualiza la matriz de adyacencia
158158 nnMatrix = ( distMatrix < txRange );
159159
160160 plot ( x Location , y Location , ' x ' );
161161 end
```