



UNIVERSIDAD NACIONAL DE COLOMBIA

Sistema inteligente de monitoreo para el consumo de energía de redes inalámbricas ad hoc

Ing. Jairo Orlando Maldonado Alarcón

Universidad Nacional de Colombia
Departamento de Ingeniería de Sistemas e Industrial
Bogotá, Colombia

2013

Sistema inteligente de monitoreo para el consumo de energía de redes inalámbricas ad hoc

Ing. Jairo Orlando Maldonado Alarcón

Tesis presentada como requisito parcial para optar al título de:
Magister en Ingeniería de Telecomunicaciones

Director:

PhD (c)., Luis Fernando Pedraza Martínez

Codirector:

PhD., Jorge Eduardo Ortiz Triviño

Línea de Investigación:

Redes y Sistemas de Telecomunicaciones

Grupo de Investigación:

Grupo de Investigación en Teleinformática de la Universidad Nacional de Colombia - GITUN

Universidad Nacional de Colombia

Departamento de Ingeniería de Sistemas e Industrial

Bogotá, Colombia

2013

Nota de Aceptación

Director

Codirector

Jurado

*Este trabajo se lo dedico a mis padres,
gracias a su esfuerzo pude alcanzar
mis objetivos académicos*

Agradecimientos

AL PhD (c), ingeniero Luís Fernando Pedraza Martínez por aportar sus conocimientos en TIC, redes de datos y Lógica difusa.]

Al PhD, ingeniero Jorge Eduardo Ortiz Triviño ya que con sus conocimientos en redes móviles *ad hoc* hizo un gran aporte al desarrollo del presente trabajo.

A la Ms, ingeniera Zoila Ramos de Flórez ya que puso a nuestra disposición sus conocimientos y la infraestructura del grupo GITUN para las pruebas y desarrollo esta tesis.

A la PhD, ingeniera Ingrid Patricia Páez Parra por sus aportes como profesora de la materia redes inalámbricas de la Maestría de Telecomunicaciones

A mis compañeros y profesores de la Universidad Nacional de Colombia de la Maestría de Telecomunicaciones, por su colaboración durante mi estadía en la universidad.

A todas las personas que de una u otra forma contribuyeron para la realización y feliz término de la presente investigación

Declaración

Me permito afirmar que he realizado la presente tesis de manera autónoma y con la única ayuda de los medios permitidos y no diferentes a los mencionados en la propia tesis. Todos los pasajes que se han tomado de manera textual o figurativa de textos publicados y no publicados los he reconocido en el presente trabajo. Ninguna parte del presente trabajo se ha empleado en ningún otro tipo de tesis.

Jairo Orlando Maldonado Alarcón

Bogotá, D.C., 17.12.2013

Resumen

En esta tesis se desarrolla un sistema inteligente por medio de la lógica difusa que permite el monitoreo del consumo de energía en redes móviles ad hoc (MANET), se determinan los criterios de rendimiento del consumo energético en este tipo de redes por medio de la simulación de métricas que permiten estimar el consumo energético.

Se realizó un análisis y selección de las métricas que determinan el consumo de energía de una MANET con lo que se buscó determinar las ecuaciones con las cuales se determina la mejor medida del consumo, esto con el objeto de llegar a un análisis de la eficiencia energética de la MANET y las métricas que permiten medir el consumo energético.

Las técnicas de inteligencia computacional estudiadas y analizadas son: la lógica difusa, las redes neuronales y algoritmos genéticos. Aquí se da una descripción de cada una de las técnicas, su funcionamiento y aplicaciones. Así mismo, se estableció la técnica de inteligencia computacional que mejor se adapta a los requerimientos del sistema, analizando y evaluando criterios como el tiempo de procesamiento y la complejidad en su implementación, para ello se analizaron los resultados simulando los parámetros por medio del NS3 y MATLAB FUZZY TOOLS, los cuales determinaron las métricas de consumo de energía.

Para el presente trabajo se dan las características de los simuladores para MANET. Se seleccionó el NS3 por sus diferentes aplicaciones en un variado tipo de protocolos inalámbricos, con diferentes tecnologías de redes como WiFi, WiMax, LTE, MANET y MESH; además tiene incorporados los módulos de protocolos, de criterios de rendimiento de la MANET como la movilidad, enrutamiento, propagación y consumo de energía, con código fuente disponible, por ser software libre, en un lenguaje comprensible como C++ y Python; su programación hace fácil modificar los módulos de protocolos de aplicación del simulador, desarrollando algoritmos nuevos o simplemente modificando los ya implementados.

Se simularon diferentes escenarios de comparación, donde se mide el consumo de energía, utilizando los criterios de rendimiento de consumo de energía en MANET, se hace un análisis estadístico y se calcula el porcentaje de error para determinar en qué medida se aproxima a un valor real del consumo de energía en MANET.

Las variables como la longitud y tiempo del paquete, el número de saltos, las pérdidas en el trayecto α y la localización en un sector Φ , son integradas y tenidas en cuenta en un conjunto de ecuaciones que representan la métrica del consumo, las ecuaciones que incluyen los criterios de eficiencia energética y son las utilizadas en el escenario del consumo de energía usando el método de lógica difusa.

Palabras clave: Monitoreo, redes móviles ad hoc, inteligencia artificial, eficiencia energética, consumo de energía.

Abstract

In this thesis an intelligent system is developed using fuzzy logic that allows monitoring of energy consumption in ad hoc mobile networks (MANET), the performance criteria of the energy consumption in such networks are determined by the simulation of metrics that allow to estimate the energy consumption.

An analysis and selection of metrics was made that determine the energy consumption of a MANET with what was intended to determine the equations from which the best measure of consumption is determined, this is in order to arrive at an analysis of the energy efficiency of the MANET and the metrics that allow to measure the energy consumption.

The studied and analyzed techniques of computational intelligence are: fuzzy logic, neural networks and genetic algorithms. Here is given a description of each of the techniques, its operation and its applications. Likewise, it was established the technique of computational intelligence that best fits the system requirements, analyzing and evaluating criteria such as processing time and complexity in implementation, for this the results were analyzed by simulating the walls of NS3 and MATLAB FUZZY TOOLS, which determined the metrics of power consumption.

For the present work the characteristics of the simulators are given for MANET, selecting the NS3 because of its different applications in varied kinds of wireless protocols, with various network technologies such as WiFi, WiMax, LTE, MANET and MESH; additionally it has incorporated the performance protocols modules criteria of MANET like the mobility, routing, propagation and energy consumption, with source code available, since it's a free software, in an understandable language like C++ and Python, its programming makes it easy to modify modules simulator application protocols, developing new algorithms or simply modifying the ones already deployed.

Different scenarios were compared, where the energy consumption is measured using the performance criteria of energy consumption in MANET, a statistical analysis is done and the error rate is calculated to determine the extent to which it approaches a real value of a simulated energy consumption in MANET.

Variables such as the packet's length and time, the number of hops, the losses in the path α and the location in a sector Φ are integrated and taken into account in a set of equations that represent the metric of consumption, the equations include energy efficiency criteria are the ones used in the scenario of energy consumption using the method of fuzzy logic. Variables such as packet length and time, the number of hops, the losses in the drive and location α in Φ sector are taken into account and integrated into a set of equations representing the metric of consumption, the equations include energy efficiency criteria and are used in the scenario of energy consumption using the method of fuzzy logic.

Keywords: Monitoring, mobile ad hoc networks, artificial intelligence, energy efficient, energy consumption.

Contenido

	Pág.
Resumen	XI
Lista de tablas	XVII
Lista de figuras	XVIII
Lista de Símbolos y abreviaturas	XIX
Introducción	1
1. Red móvil ad hoc (MANET)	5
1.1 Características de la MANET [10].....	5
1.2 Modelos de movilidad en la MANET	6
1.2.1 Modelo de movilidad RandomWalk	7
1.3 Criterios de rendimiento del consumo en MANET a través de modelos.....	8
1.3.1 Modelo del consumo energético para el enrutamiento en MANET [17].	8
1.3.2 Modelo del costo energético para un rango de transmisión [21].	11
1.3.3 Modelo de energía con tamaño de paquete y n nodos [32]	16
1.3.4 Modelo del consumo de energía utilizado NS3 [36, 37].....	19
1.3.5 Modelo del consumo de energía con n saltos entre nodos.....	20
1.4 Estimación de energía disipada en una MANET	21
1.5 Control del Consumo de energía.	22
1.6 Control de acceso al medio MAC.....	25
1.7 Gestión de energía en equipos	27
1.8 Definición de variables en MANET.....	28
1.9 Análisis de criterios de rendimiento de consumo en MANET.	30
2. Técnicas de inteligencia computacional	33
2.1 Lógica Difusa.....	33
2.2 Algoritmos genéticos	35
2.3 Redes neuronales.....	37
2.4 Selección de la técnica de inteligencia computacional.....	39
3. Análisis de simulación MANET	41
3.1 NS2	41
3.2 NS3	41
3.3 GloMoSim.....	42
3.4 GTNetS	42

3.5	SWANS.....	42
3.6	NCTUns	43
3.7	OMNET++	43
3.8	Resumen de características de los simuladores para MANET	44
4.	Resultados y análisis de la simulación en MANET.....	47
4.1	Escenario 1, consumo por número de saltos.....	47
4.2	Escenario 2, el modelo de energía en NS3 [73].....	48
4.3	Escenario 3, consumo variando el número de saltos en ecuación 1.1	49
4.4	Escenario 4, consumo incrementando el tiempo del paquete	51
4.5	Escenario del consumo de energía usando el método de lógica difusa.	52
4.5.1	Reglas del Modelo de energía en MANET con el método de lógica difusa.....	52
4.6	Escenarios de comparación entre modelos de consumo de energía.	56
5.	Conclusiones y recomendaciones	63
5.1	Conclusiones.....	63
5.2	Recomendaciones.....	64
A.	Anexo: código en MATLAB de la Interfaz de lógica difusa, para el modelo de energía en una MANET	65
B.	Anexo: Código del menú principal del programa análisis estadísticos, ecuaciones y graficas del consumo de energía en una interfaz en C#.	72
6.	Bibliografía.....	76

Lista de tablas

Pág.

Tabla 1-1: Pérdidas por trayectoria del medio en [31].....	29
Tabla 1-2: Análisis de criterios del consumo de energía de redes ad hoc.....	31
Tabla 3-1: Resumen de características de los simuladores para MANET.....	44
Tabla 4-1: Parámetros comunes de los escenarios de simulación para MANET	47
Tabla 4-2: Parámetros adicionales del escenario 2 de simulación para MANET	49
Tabla 4-3: Parámetros adicionales del escenario 3 de simulación para MANET	50
Tabla 4-4: Parámetros adicionales del escenario 4 de simulación para MANET	51
Tabla 4-5: Variables estadísticas de la comparación de consumos en una MANET	57
Tabla 4-6: Variables estadísticas de la comparación de los consumos en una MANET.	59

Lista de figuras

Figura 1-1: MANET redes dinámicas de múltiples saltos en [10].....	5
Figura 1-2: Modelos de movilidad [15].	7
Figura 1-3: Modelo Random Walk [16].....	7
Figura 1-4: Distribución de los nodos en una MANET en [17].	8
Figura 1-5: Cobertura de radio enlace (a) unidimensional, (b) bidimensional y tridimensional; (c) las redes. La región cubierta tiene un radio r y está centrada en la unidad. Fuente: [21].	11
Figura 1-6: Descomposición de la energía J_n para acceso básico con longitud de paquete = 1 KB. Fuente: [32].	16
Figura 1-7: Descomposición de la energía J_n para RTS/CTS con longitud de paquete de 1 KB. Fuente: [32].	17
Figura 1-8: Simulación en [37] del consumo de energía en NS3 ecuación 1.19 y 1.20. .	19
Figura 1-9: Los ciclos o periodos listen y sleep para S-MAC [3].	22
Figura 1-10: Función de coordinación distribuida. Fuente: [39].	25
Figura 2-1: La lógica difusa representada en diagrama de bloques [46].....	34
Figura 2-2: Diagrama de bloques algoritmos genéticos en [46].	37
Figura 2-3: Diagrama de una red neuronal. Fuente: [54].	38
Figura 4-1: Simulación del consumo en MATLAB utilizando la ecuación (1.1).	48
Figura 4-2: Simulación del consumo de energía en [73].	49
Figura 4-3: Simulación del consumo de energía con variación de número de saltos.....	50
Figura 4-4: Consumo incrementando de la duración del paquete, ecuación 1.13.....	51
Figura 4-5: Universo de entrada, longitud de paquete.....	53
Figura 4-6: Universo de entrada, número de saltos.....	54
Figura 4-7: Universo de salida, energía consumida.....	54
Figura 4-8: Interfaz en MATLAB, resultado del consumo con lógica difusa.	55
Figura 4-9: Comparación de datos Ecuación 1.1 y lógica difusa.	56
Figura 4-10: Comparación Ecuación 1.13 y lógica difusa MATLAB.....	57
Figura 4-11: Error absoluto de la comparación ecuación 1.13 y lógica difusa.	58
Figura 4-12: Error relativo de la comparación ecuación 1.13 y lógica difusa.	58
Figura 4-13: Error absoluto de la comparación ecuación 1.1 y lógica difusa.	59
Figura 4-14: Error relativo de la comparación de datos ecuación 1.1 y lógica difusa.....	60

Lista de Símbolos y abreviaturas

Abreviatura	Término
ACK	Ackow ledgement
ANN	Articial neural network
AOVD	Ad hoc on demand distance vector
AP	Acces Point
BS	Base station
CERA	Cluster based energy saving routing algorithm
CFP	Contention free period
CMMBCR	<i>Conditional max-min battery capacity routing</i>
CP	Contention period
CSMA/CA	Carriersense multiple acces with collision avoidance
CTS	Clear to send
DCF	Distributed coordination function
DR	Drain rate
DS	<i>Distribution Service</i>
DSR	<i>Dynamic Source Routing</i>
DSDV	Destination Sequeced Distance Vector
DFWMAC	<i>Distributed Foundation Wireless Medium Access Control</i>
E_c	Energía consumida
E_d	Energía disipada
E_o	Energía en un nodo
E_r	Energía residual
Ermax	Energía residual máxima
HCF	Hybrid Coordination Function
HMAC	Hybrid Medium Access Control
HEED	Hybrid, Energy-Efficient, Distributed Clustering
IA	INTELIGENCIA ARTIFICIAL
MAC	Medium Access Control
MANET	Mobiles ad hoc networks
MDR	Minimum drain rate
MMBCR	Min-max battery cost routing

MTPR	Minimum Total Transmission Power Routing
NACK	Negative acknowledgement
NS2	Simulador de redes versión dos
NS3	Simulador de redes versión tres
P _c	Potencia consumida
PCF	Point Coordinated Function
QoS	Quality of Service
RBP	Residual battery power
RSSI	Received Signal Strength Indicator)
RTS	Request To Send
RBP	Residual battery power
SMAC	Sensor Medium Access Control

Introducción

Actualmente el ahorro y manejo de sistemas de consumo de energía es un factor importante para los sistemas de monitoreo de la gestión de redes de telecomunicaciones. En [1] se ve el diagnóstico de la red como algo global donde todos sus equipos tienen que ser supervisados para identificar y resolver problemas, lo que implica disponer de tiempo y de personal.

Por ello existe la necesidad de un sistema que permita la toma de decisiones, un software de monitoreo que no solo muestre unos parámetros, sino que cree unas herramientas que minimicen el consumo de recursos energéticos de una red. Al pensar en un sistema general de monitoreo o gestión del consumo se deben tener en cuenta factores económicos: ¿cuánto le cuesta a una empresa mantener sus equipos en funcionamiento?, ¿cuánta de esa energía que consumen es malgastada?, ¿cuánto podrían ahorrarse con un sistema eficiente de consumo?.

Además, cada vez son más las personas que se conectan diariamente a Internet tanto desde su casa como desde el trabajo. A su vez, la red ha incrementado sustancialmente la velocidad de transmisión de datos. Todos estos factores hacen que los requerimientos tecnológicos sean mayores y que, por tanto, se incremente el consumo energético de las redes elevando los costos de operación.

En el presente documento se determinan los criterios de rendimiento del consumo de energía en una red móvil ad hoc (MANET), a partir de la evaluación de las variables más frecuentemente usadas como métricas del consumo comparando tanto ecuaciones teóricas como prácticas, para medir el desempeño de las MANET.

En cuanto al consumo de energía algunas de las ecuaciones son utilizadas en la práctica en el simulador NS3 y otras son teóricas, estudiadas como estrategia en el enrutamiento de redes.

Las ecuaciones se evalúan teniendo en cuenta varios criterios que establecen el rendimiento del consumo de la MANET como la cantidad de energía suministrada en los equipos, consumida en la transmisión de paquetes de información y por el número de saltos entre nodos en la llegada de datos, teniendo en cuenta el medio donde se localiza la MANET, con parámetros como pérdidas por trayectoria, localización, modelo de movilidad y área de cobertura.

También se tuvo en cuenta el tráfico de la red en cuanto a velocidad, tiempo y tamaño del paquete. Adicionalmente, se analizaron los modelos de consumo y cómo estos son afectados por las principales métricas tales como número de saltos y el tamaño del paquete; las ecuaciones del consumo se seleccionaron evaluando los criterios anteriormente mencionados. Las métricas de esta elección forman parte de la entrada del sistema inteligente, se elige la técnica de inteligencia computacional que se evalúa a través de la complejidad de implementación y tiempo de procesamiento, parámetros requeridos para obtener un resultado aceptable en la salida del sistema, el cual debe determinar el consumo de energía en el tiempo que sea requerido para poder ser monitoreado.

Existen varios trabajos en el ahorro de la energía a través del control de acceso al medio MAC. En [2-9] se han determinado variables a través de simulación que mejoran notablemente la eficiencia energética. También hay protocolos que solucionan los problemas de *overheating* y colisión, siendo estos dos factores los que más recursos energéticos consumen de una MANET. El aporte del presente trabajo consiste en determinar las métricas que afectan el consumo energético.

Así, un sistema de monitoreo es aquel que logre asegurar la existencia de procedimientos y controles orientados a la satisfacción de la administración, instalación y operación de una red [1]. Para ello se deben tener en cuenta la gestión de red, los equipos y su conectividad, la monitorización de las comunicaciones, la revisión de costos y la asignación formal de proveedores, reacción y aplicabilidad de estándares. Además debe cumplir con objetivos de control y tener una gerencia de comunicaciones con plena autoridad de voto y acción.

La gestión de redes es un proceso consecuente y necesario para las empresas e instituciones que buscan asegurar que todos sus activos sean protegidos en forma adecuada y que hacen del ahorro en el consumo de energía un factor importante. Llevar un registro actualizado de módems, controladores, terminales, líneas y todo equipo relacionado con las comunicaciones, mantener una vigilancia constante sobre cualquier acción en la red, registrar el costo por consumo de energía de comunicaciones son acciones necesarias para mejorar el rendimiento y la resolución de problemas presentados en la red.

OBJETIVO GENERAL Y OBJETIVOS ESPECÍFICOS:

OBJETIVO GENERAL

Monitorizar el consumo de energía para una red inalámbrica ad hoc a través de un algoritmo para el análisis de los recursos energéticos, por medio de inteligencia computacional.

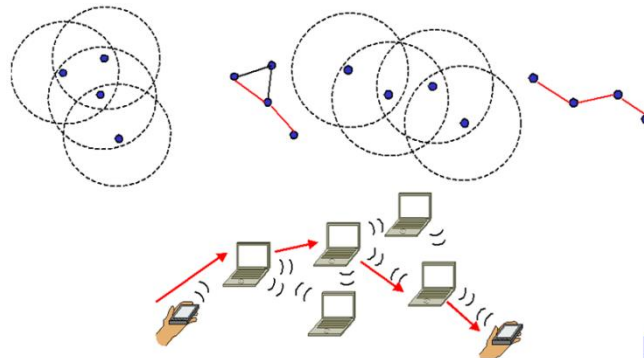
OBJETIVOS ESPECÍFICOS

- Determinar los criterios de rendimiento que permitan establecer el consumo de energía en una red inalámbrica ad hoc.
- Aplicar una técnica de inteligencia computacional para monitorizar una red inalámbrica ad hoc.
- Simular una métrica que permita estimar el consumo energético en una red inalámbrica ad hoc.

1.Red móvil ad hoc (MANET)

Una red móvil ad hoc (MANET) en [10] es una red formada por nodos móviles que se interconectan entre sí a través de enlaces inalámbricos de comunicación cuya duración, en general es corta y cuyo objetivo es soportar la comunicación solo un momento, dichos enlaces pueden ser de uno o de múltiples saltos. El conjunto de dispositivos móviles de una red ad hoc se pueden comunicar sin necesidad de un punto de acceso (AP) o sin una infraestructura preexistente. La red posee un conjunto autónomo y espontáneo de enrutadores móviles, conectados por enlaces inalámbricos cuya unión forma un grafo arbitrario, allí los nodos tienen libertad de movimiento y la topología multsalto cambia rápidamente de forma impredecible. Estas redes no requieren de un control centralizado. La forma básica como opera se ilustra mediante la figura 1-1.

Figura 1-1: MANET redes dinámicas de múltiples saltos en [10].



1.1 Características de la MANET [10]

La MANET se caracteriza por poseer una topología dinámica donde los nodos son libres de moverse arbitrariamente. Por ello su topología de red suele ser multsalto ya que puede cambiar al azar y rápidamente, a veces de manera impredecible, y puede consistir en dos enlaces bidireccionales y unidireccionales.

La MANET tiene un ancho de banda limitado, con capacidad variable en los enlaces inalámbricos. Además, cabe notar que el rendimiento de las comunicaciones inalámbricas después de considerar los efectos de acceso múltiple, *fading*, el ruido y la interferencia, etc., es, a menudo, menor a la tasa de transmisión máxima de un radio. Uno de los efectos de la relativamente baja a moderada capacidad del enlace es que la

congestión es generaliza, eso quiere decir que es probable que la demanda de la aplicación global se aproxime o exceda a la capacidad de la red.

La seguridad física limitada en las MANET es habitualmente más vulnerable que las amenazas de seguridad física de las redes de cable, por lo que la posibilidad de espionaje, falsificación y los ataques de denegación de servicio deben ser considerados cuidadosamente. Las técnicas existentes de seguridad de enlace se aplican a menudo en redes inalámbricas para reducir las amenazas de seguridad. Como beneficio, el carácter descentralizado de control de la red en MANET proporciona robustez adicional contra los puntos únicos de error de los enfoques centralizados.

En las MANET, uno de los parámetros más importantes es la seguridad ya que se requiere de la protección suficiente para evitar la interrupción del funcionamiento del protocolo ad hoc. Esto puede ser algo ortogonal a cualquiera de los protocolos de enrutamiento, por ejemplo, se pueden usar técnicas de seguridad IP.

Soporte a enlaces unidireccionales: los enlaces bidireccionales son usados en el diseño de protocolos de enrutamiento y muchos de ellos que no pueden trabajar sobre enlaces unidireccionales. Sin embargo, pueden existir enlaces unidireccionales en las redes inalámbricas. Si por ejemplo existen dos enlaces unidireccionales en sentidos opuestos para formar una conexión bidireccional en algunas regiones de la red ad-hoc, debe hacerse uso de ellos.

1.2 Modelos de movilidad en la MANET

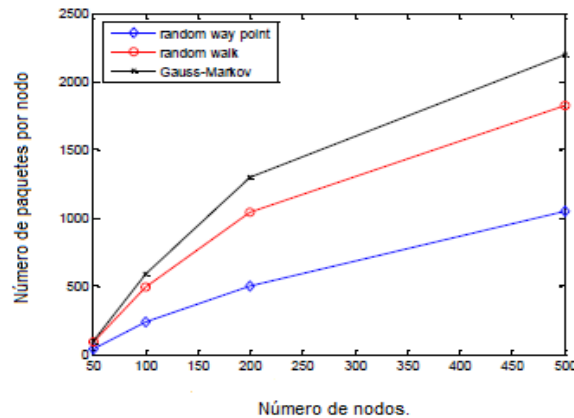
Para poder modelar el desplazamiento de los nodos en una determinada área de una red inalámbrica ad hoc y poder medir su desempeño se requiere de un modelo de movilidad en el que existe una cierta relación entre los nodos y sus movimientos en el escenario de simulación. Como se puede ver en la figura 1-2, dependiendo del modelo que se escoja hay una variación en el número de mensajes o paquetes por el número de saltos hasta la llegada de la información. Por eso es importante ver varios modelos de movilidad para tener un criterio de selección y poder escoger el que mejor represente las variables que afecten el consumo en una red ad hoc.

Este tipo de modelos de movilidad pueden ser estudiados a través de modelos de movilidad para entidades, en donde cada nodo tiene movimientos independientes. Las representaciones de estos modelos se encuentran en los modelos Gaus-Markov [11], RandonWalk [12], Random Waypoint [13] y el modelo en ciudades, otra representación es el modelo de movilidad de grupo con punto de referencia que describe el movimiento de grupos de entidades [14].

A continuación se hace una comparación de los modelos de movilidad, se muestra el trazo correspondiente al movimiento de los nodos para cada modelo. En la figura 1-2 se

representan los modelos de movilidad donde se determina el número de mensajes por nodo.

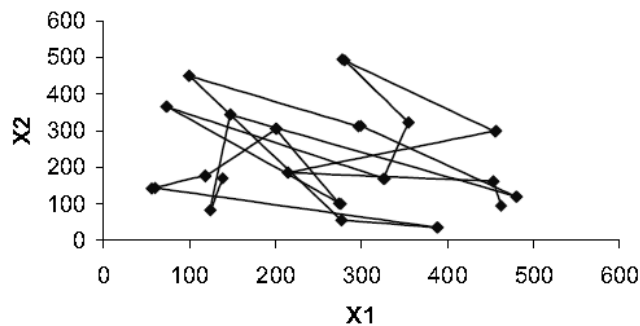
Figura 1-2: Modelos de movilidad [15].



1.2.1 Modelo de movilidad RandomWalk

Es un modelo de movilidad simple basado en direcciones y velocidades aleatorias. Un nodo móvil se mueve de su actual ubicación a una nueva ubicación seleccionando aleatoriamente una dirección y una velocidad para viajar. Es un modelo sin memoria en el que no se tiene conocimiento de localizaciones pasadas ni recuerdo de los valores de velocidad. Si un nodo móvil alcanza los límites del área de simulación, entonces este rebota con un ángulo igual al ángulo de incidencia y el movimiento continúa en la nueva dirección como se visualiza en la figura 1-3. Cuando un parámetro se cambia, ya sea la dirección o la velocidad en un nodo, el patrón de movimiento resulta en un movimiento restringido a una pequeña porción de zona dentro del área de simulación.

Figura 1-3: Modelo Random Walk [16].



1.3 Criterios de rendimiento del consumo en MANET a través de modelos

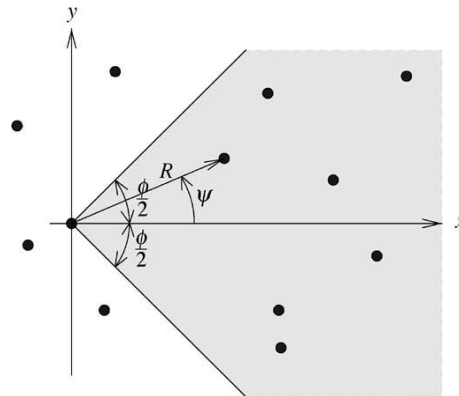
Este capítulo presenta los criterios de rendimiento del consumo de energía por medio del análisis de modelos. Estos van a definir las métricas que intervienen y afectan el consumo de una red inalámbrica ad hoc, ya que todas las métricas de rendimiento de redes de datos varían el consumo de energía. Esto se debe a que el envío de paquetes de información depende de factores como la potencia, la distancia y la cantidad de equipos que intervienen.

Aquí, los modelos del consumo son representados por ecuaciones las cuales están definidas por unas variables o métricas, relacionadas a su vez con el consumo de energía en cualquier red inalámbrica de datos. A continuación se presentan los modelos o ecuaciones básicas utilizadas con mayor frecuencia en simulación de redes y otras formuladas como conceptos físico teóricos de la disipación de energía y sistemas de potencia en transmisores de radiocomunicaciones.

1.3.1 Modelo del consumo energético para el enrutamiento en MANET [17].

La eficiencia en la conexión en un enlace de radiocomunicación es $X / R = \cos (\Psi)$, donde X es la posición del vecino más cercano en la coordenada x ya que (X, Y) y (R, Ψ) son las coordenadas cartesianas y polares respectivamente. Se pueden ver representadas en la figura 1-4 en [17] donde se determina la localización de los nodos distribuidos en un sector ϕ en el plano (X, Y) .

Figura 1-4: Distribución de los nodos en una MANET en [17].



El consumo de energía de una comunicación en una red móvil ad hoc (MANET) se especifica en la ecuación (1.1) donde se tienen en cuenta las pérdidas en el trayecto α debido a la atenuación del medio de propagación del lugar donde se encuentre la red. La localización en un sector ϕ , que representa la dirección del transmisor al receptor con respecto a la orientación en el plano en envío de paquetes a través de una red multisalto de n nodos, con una probabilidad de recepción $P_r = \sqrt[n]{P_{EE}}$, donde la probabilidad de entrega de paquetes de extremo a extremo deseada es P_{EE} , y n el número de saltos. La energía total que se espera para el consumo es $n^2 \mathbb{E}[R^\alpha]$, se expresa en la siguiente ecuación:

$$E = n^2 \left(\frac{2}{\phi}\right)^{\frac{\alpha}{2}} \Gamma\left(1 + \frac{\alpha}{2}\right) \quad (1.1)$$

Esta ecuación no solo es una métrica del consumo sino que también es una estrategia para el enrutamiento de redes dinámicas multisalto como las redes móviles ad hoc, para determinar la eficiencia en la ruta por donde los paquetes van a circular por n nodos de la red. La energía necesaria para la entrega de paquetes a través de un proceso aleatorio está dada por $\bar{X} = n\eta\phi \mathbb{E}[R(\phi)] = n\eta(\phi)\sqrt{\pi/(2\phi)}$ con una probabilidad P_{EE} , se determina normalizado $\Theta N / (-\ln P_{EE})$ donde N es la potencia de ruido y Θ es un umbral determinado por el hardware de comunicación, la modulación y la codificación en [18].

En las redes con n saltos aleatorios, el proceso de distribución Poisson corresponde a una distribución uniforme en [19], para redes con varias estaciones o nodos, este tipo distribuciones son equivalentes. Para efectos prácticos se presenta una red cuyos nodos constituyen un proceso Poisson bidimensional en una red aleatoria Poisson; se restringe para el caso $\lambda = 1$ (unidad de densidad), dado que el producto λA siempre se puede expandir tal que $\lambda = 1$.

Se establece que las distancias entre nodos vecinos se distribuyen como un proceso de distribución tipo Poisson homogéneo de densidad λ en el plano donde la probabilidad de encontrar k nodos en un área A está dada por:

$$P [k \text{ nodos } A] = \frac{(\lambda A)^k}{k!} \quad (1.2)$$

Para los esquemas de enrutamiento considerados, es necesario determinar la distancia desde un nodo a sus nodos vecinos los cuales se encuentran dentro de un sector ϕ , es decir, dentro de $\pm \phi/2$ al eje de la fuente al destino.

La representación de la distancia al vecino más cercano en una red aleatoria Poisson en un área con unidad de densidad: la distancia R entre un nodo y su vecino más cercano en un sector ϕ es una distribución Rayleigh con media $\sqrt{\pi / (2\phi)}$.

Como experimento se tiene la distancia R al vecino más cercano en un sector con ángulo de incidencia ϕ . La probabilidad de que no se encuentre un vecino en un sector ϕ hasta una distancia r es la distribución acumulativa complementaria:

$\mathbb{P}[R > r] = e^{-\frac{r^2 \phi}{2}}$, con lo que la densidad de probabilidad es:

$$p_R(r) = r\phi e^{-\frac{r^2 \phi}{2}} \quad (1.3)$$

Esta función de densidad es una distribución de Rayleigh con una media de $(\pi / (2\phi))^{1/2}$ y varianza $2 / \phi - \pi / (2\phi) = (4 - \pi) / (2\phi)$. El argumento ψ es una distribución uniforme entre $-\phi / 2$ y $\phi / 2$.

Una red de Rayleigh es una distribución aleatoria de Poisson donde el canal físico está sujeto a las pérdidas por trayectoria α . Para comparar diferentes esquemas de enrutamiento, es necesario definir la eficiencia en la ruta.

La eficiencia en la ruta es la relación de la distancia euclidiana entre los nodos finales y reales de la distancia recorrida. Como ejemplo, la eficiencia de ruta en una red en topología de malla cuadrada donde el encaminamiento o ruta al vecino más cercano corresponde a $\kappa(\phi) = \frac{1}{|\cos \phi| + |\sin \phi|}$

Donde ϕ denota la dirección del transmisor al receptor con respecto a la orientación en el plano. El máximo es 1, el mínimo es de $1/\sqrt{2}$ (a $\phi = \pi / 4$ $\kappa(\pi / 2)$), y el valor esperado es

$$\eta: = E[\kappa] = \frac{2\sqrt{2}}{\pi} \operatorname{arctanh} \frac{\sqrt{2}}{2} \approx 0,79 \quad (1.4)$$

Donde η denota la eficiencia de trayectoria esperada.

La media de la distancia al vecino más cercano, R_n^α está dada por:

$$E[R_n^\alpha] = \left(\frac{2}{\phi}\right)^{\frac{\alpha}{2}} \frac{\Gamma(n + \frac{\alpha}{2})}{\Gamma(n)} \quad (1.5)$$

Para obtener esta aproximación se selecciona el primer término del desarrollo en serie de $\Gamma(n + 1/2) / \Gamma(n) = \sqrt{n} (1 - (1/8n) + o(1/n^2))$ en [20]. Al calcular esta igualdad se obtienen expresiones con uno de los términos representados como Landau símbolo O . Este término indica el orden de truncamiento (orden de error) del desarrollo en serie calculada y observando que no es precisa para un n pequeño, ajustar (mediante la adición de $\pi / 4 - 1$ para n) tal que la aproximación es preciso para $n = 1$.

Para $n > 1$, se da un límite superior muy apretado. El segundo momento es $2n / \phi$, por lo tanto, la varianza es:

$$\operatorname{Var}[n] = \frac{2n}{\phi} - E[n]^2 \approx \frac{4 - \pi}{2\phi} \quad (1.6)$$

Utilizando la misma aproximación, desde el desarrollo en serie anterior se obtiene $\operatorname{Var}[R_n] = (1/2\phi) O(1/n)$. Se deben tener en cuenta los límites superiores e inferiores en la varianza.

$$\frac{4 - \pi}{2\phi} \leq \text{Var} [R_n] < \frac{1}{2\phi} \forall n \in \mathbb{N}. \quad (1.7)$$

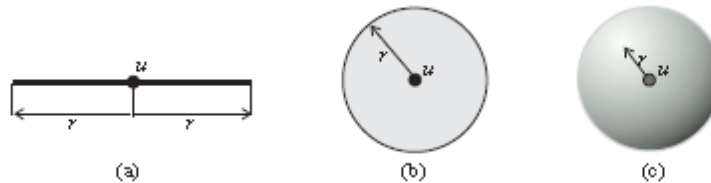
Desde el $4 - \pi \approx 1$, se concluye que la varianza es independiente.

1.3.2 Modelo del costo energético para un rango de transmisión [21].

Se analiza un modelo para el consumo de energía donde se determina como métrica el costo de energía. Este costo depende de la asignación del intervalo o rango de alcance de transmisión definido por la variable RA, correspondiente a la distancia de transmisión dada por el radio de cobertura de radio enlace r , el cual depende a su vez de la potencia de los equipos de radiocomunicación. Este modelo se define para una transmisión inalámbrica, utilizando el registro de la distancia en el trayecto sobre la topología y control de la MANET, lo que resulta de suma importancia para la investigación en [21].

En la figura 1-5 se representa la cobertura en la radiocomunicación de dos estaciones en una sola dimensión (segmento de longitud $2r$) y está centrado en la unidad u , que es la estación o nodo transmisor. La comunicación de dos dimensiones es un círculo de radio r centrado en u ; en redes de tres dimensiones, es la esfera de radio r con centro en u (Ver la figura 1-5).

Figura 1-5: Cobertura de radio enlace (a) unidimensional, (b) bidimensional y tridimensional; (c) las redes. La región cubierta tiene un radio r y está centrada en la unidad. Fuente: [21].



Dada una red $MD = (N, L)$, una asignación del rango para MD es una función que asigna un valor de RA ($u \in (0, \infty)$) a cada elemento de U de N , donde r_{max} es una representación de su rango de transmisión. El Parámetro r_{max} (Alcance Máximo de Transmisión) depende de las características de los equipamientos de los nodos transmisores-receptores de radio. Por lo general se asume que los nodos de la red son equipados con transmisores-receptores con características similares, es decir, r_{max} es el mismo para todos los nodos en la red. Dada una asignación del Rango RA para una determinada red $MD = (N, L)$, el costo energético de la RA se define como:

$$c(RA) = \sum_{u \in N} RA(u)^\alpha \quad (1.8)$$

Donde α es el gradiente de energía a distancia o factor de pérdidas por trayectoria. Cabe señalar que ésta definición de costo de energía es coherente con la hipótesis de trabajo que dice que la señal de radio se propaga según el modelo de ruta del registro de distancia.

El problema de asignación del intervalo o rango RA fue estudiada por primera vez en [22] y se definió de la siguiente manera: Sea N un conjunto de nodos en el espacio d -dimensional, con $d = 1, 2, 3$. Con esto se determina una función de asignación del rango RA de forma que la comunicación correspondiente está fuertemente vinculada, y $c(RA) = \max_{u \in N} (RA(u))^\alpha$ es mínima en todos los nodos conectados a las funciones de rango de trabajo.

Los algoritmos de enrutamiento calculan el camino más corto basados en el costo energético asociado con cada radio enlace utilizando una métrica de distancia. Existe para ello una aproximación muy en boga en los últimos tiempos y ampliamente diseminados en la literatura técnica en [23] y [24] que es el enrutamiento proactivo en el cual la capa de red *Distribution Service* (DS) periódicamente actualiza las rutas óptimas ya que la movilidad de los nodos hace del esquema de radio propagación un sistema aleatorio. Cuando un paquete de datos necesita ser transmitido el DS conoce la ruta óptima mediante un tráfico cursado por las constantes actualizaciones en la topología de la red.

DSDV (*Destination- Sequenced Distance-Vector*) es una variación del algoritmo de enrutamiento Bellmand-Ford para Internet clásico en [23] en el que cada nodo mantiene una tabla de enrutamiento con direcciones actualizadas en el formato (Destino, Próximo salto, Costo, Número de secuencia, Tiempo de vida), actualización y envío a los nodos vecinos son listados en marcos de datos más pequeños (Destino, Métrica, Número de secuencia). Este mecanismo permite a los nodos vecinos propagar información rápidamente del enlace al nodo fuente sobre el estado del nodo de destino.

La obtención del costo energético está determinada por la medición del RSSI (*Received Signal Strength Indicator*) como parámetro de aproximación. La diseminación del costo energético a través de la red con el protocolo DSDV asegura que todos los nodos de la red reciban un estado actualizado de la topología de diseminación de sus vecinos. Esto se debe al hecho de que los números de secuencia son incrementados en cada actualización de la simulación. El número de actualizaciones periódicas, a partir de cada simulación, dependerá de la cantidad de tráfico cursado en la red: El protocolo DSDV establecerá una ruta si los números de secuencia o la métrica de costo energético por actualización son mejores que la tabla de enrutamiento actual.

Por ello es posible determinar que la ruta óptima genere oscilaciones indeseables alrededor del punto óptimo debido a transientes inestables de energía en la topología de la red. Una forma de evitar esto es suavizar el esquema de la red; eso quiere decir que los nodos que fluctúen mucho en el esquema de movilidad no serán actualizados, ni los números de secuencia, ni su costo energético. Esto logrará que la red llegue a su estado

de régimen en más corto tiempo evitando así fluctuaciones indeseables sobre todo en los números de secuencia, pues son los más sensibles en las actualizaciones periódicas del algoritmo DSDV.

A continuación se reseñan una serie de documentos en los cuales se consideran las restricciones de energía de las estaciones en las MANET donde el objetivo en [25] es mejorar los procesos de enrutamiento utilizados y proponer nuevos mecanismos que integren las actividades de enrutamiento y control de consumo de energía sin degradar las prestaciones de la red. Para esto se ha propuesto una solución integral la cual reduce el consumo de energía y permite a las estaciones de la red desactivar dinámicamente su interfaz de red. Esta propuesta utiliza el mecanismo de control de energía, el cual se puede incorporar como parte del estándar IEEE 802.11 en [9].

El mecanismo de control de energía en [26], básicamente reduce el consumo de energía de las operaciones de overhearding, para integrar las actividades de enrutamiento y control de energía, el cual utiliza un protocolo basado en grupos denominados *Cluster-based Energy-saving Routing Algorithm* (CERA) en [26]. El protocolo CERA reduce el consumo de energía y permite a las estaciones permanecer en el modo de bajo consumo durante periodos de tiempo mayores. Un nuevo mecanismo de selección de rutas denominado tasa de consumo mínimo (MDR) introduce optimizaciones de consumo. Este mecanismo reduce la energía de transmisión utilizada por cada paquete enviado, prolongado así tanto la duración de la batería de las estaciones como la duración de los flujos de datos. Por último, se ha propuesto un mecanismo híbrido para reducir el problema de las tormentas de difusiones en la MANET. La evaluación de los mecanismos se ha realizado por medio de simulaciones, demostrando que los mecanismos propuestos se pueden utilizar de una forma integral y eficiente para reducir los problemas de energía en las redes MANET.

En [25, 26] los autores proponen como una nueva medida a MDR la cual se utiliza para pronosticar el tiempo de vida de los nodos de acuerdo con las condiciones de tráfico actuales. Esta métrica, cuando se combina con el valor de la potencia residual de la batería, permite establecer si un nodo puede ser parte de una ruta activa. El mecanismo MDR se puede utilizar en cualquiera de los protocolos de enrutamiento existentes como un establecimiento de la ruta crítica para lograr un doble objetivo: ampliar tanto la vida de las baterías nodales como la duración de la conexión. En [26] se propone la DR_i como la métrica que mide la tasa de disipación de energía en un nodo dado. Cada nodo n_i puede controlar su consumo de energía causada por la transmisión, recepción y actividades de overhearding y se calcula la tasa de pérdida de energía, que se denota por DR_i . Ahora bien, por cada T segundos de la toma de muestras promedio de la cantidad de energía consumida y la estimación de la disipación de energía por segundo durante los pasados T segundos en [24], T se establece en 6 segundos. El valor real se calcula utilizando el método de la media ponderada, procedimiento aplicado a la velocidad de consumo de energía de los valores anteriores DR_0 y DR_S que representan la actual muestra o los nuevos valores. La tasa de consumo de energía es calculado por: $DR_i = \alpha \times DR_0 + (\alpha - 1) \times DR_S$.

Usando $\alpha = 0.3$ estableciendo una mayor prioridad a la actual muestra de velocidad de consumo, a fin de reflejar el estado actual de gasto de energía de los nodos, se considera que la evaluación del desempeño, de acuerdo con valores diferentes de tiempo, es así mismo una vía prometedora para futuros estudios. La relación entre RBP_i y DR_i , donde RBP_i denota la potencia residual de batería, revela que el restante de la batería del nodo no se agota, es decir, que indica cuánto tiempo el nodo puede mantenerse al día con las operaciones de enrutamiento con las condiciones del tráfico actual. Basado en [19] la energía residual, la correspondiente función de costo se define como:

$$C_i = RBP_i / DR_i \quad (1.9)$$

En [26] se presenta una versión modificada llamada la Velocidad de Consumo Mínimo Condicional (CMDR) Ésta minimiza el total de potencia de transmisión consumida por los paquetes al utilizar el simulador NS2 y la fuente de protocolo de enrutamiento dinámico (DSR), comparando CMDR contra el enrutamiento con capacidad de batería máxima mínima condicional (CMMBCR). Las redes inalámbricas sin infraestructura fija son denominadas MANET. Los nodos que pertenecen a una MANET pueden ser los extremos de un conjunto de datos de intercambio o pueden actuar como enrutadores cuando los dos puntos finales no están directamente dentro de su rango de radio.

Un elemento crucial para la MANET es la actividad de los nodos con restricciones de energía. El desarrollo de protocolos de enrutamiento para la MANET ha sido un tema exhaustivo en esta área de investigación en los últimos años, durante los cuales se han propuesto varios protocolos de enrutamiento proactivos y reactivos en [23, 24], pero la mayoría de las propuestas no se han centrado en las limitaciones de potencia de nodos. Por otro lado, han aparecido varios protocolos del manejo de potencia en [25-27], pero solo algunos se han enfocado en el diseño de protocolos de enrutamiento con el fin de promover la utilización eficiente de la energía en [28-30].

Por ejemplo, K. Scott y Bambos N, en *Enrutamiento, Potencia de Transmisión Total Mínima (MTPR)* en [28], buscan minimizar el consumo de energía en la transmisión de los nodos que participan en la ruta adquirida. Dado que la potencia de transmisión requerida es proporcional a d^α fórmula presentada en [31], donde d es la distancia entre dos nodos y $2 \leq \alpha \leq 4$, MTPR selecciona las rutas con más saltos. Este acepta la participación de más nodos de reenvío de paquetes aumentando el retardo de extremo a extremo. Además MTPR no tiene en cuenta la energía restante de nodos.

En cuanto el enrutamiento y disponibilidad de energía de redes ad hoc en la búsqueda de la ruta óptima los algoritmos deben ser capaces de:

- Minimizar la potencia total necesaria para enviar los paquetes a través de una ruta.
- Maximizar el tiempo de vida de todos los nodos de la red.

Un algoritmo que cumple con estos parámetros es el Minimum Total Transmission Power Routing MTPR, el cual utiliza la métrica de la potencia de transmisión $P(n_i, n_{i+1})$ y utiliza n_i para transmitir a n_j para obtener la potencia mínima total entre estas terminales. Así, la potencia total de transmisión para la ruta L (P_L) sería: $P_L = \sum_{i=0}^{D-1} P(n_i, n_{i+1})$, donde los nodos intermedios $n_i \in$ ruta L ($i = 0 \dots D-1$) y donde n_o y n_d son los nodos origen y destino, respectivamente. Sea A el conjunto de las rutas posibles entonces la ruta óptima debe cumplir: $P_k = \min_{L \in A} P_L$.

Para lograr obtener la mínima potencia se debe disminuir el retardo de extremo a extremo lo cual depende del número de saltos y de la distancia.

En [29] se presenta el enrutamiento con Costo Batería Máxima y Mínima (MMBCR), que considera la potencia residual de las baterías y la capacidad de los nodos con las métricas de enrutamiento. MMBCR permite que los nodos con alta capacidad de batería participen en el proceso de enrutamiento con más frecuencia que los nodos con capacidad de batería más baja. En todos los caminos posibles, existe un nodo más débil que tiene la capacidad mínima de la batería. El enfoque MMBCR trata de elegir un camino cuyos nodos tienen la potencia máxima entre los nodos más débiles en otras rutas posibles para el mismo destino. MMBCR extiende un tiempo de vida útil de los nodos pero no garantiza que la potencia de la transmisión total se reduzca al mínimo en una ruta elegida.

En [29] se propone el mejoramiento del consumo de energía a través de la capa de enlace, lo que se busca es implementar el modo de apagado para evitar el agotamiento de las baterías y que sean capaces de alargar el tiempo de vida de las baterías de una red ad hoc. Por ello, presenta un protocolo que conserva la capacidad de batería de los nodos, apagando aquellos nodos que no se dedican a la transmisión y recepción activa de paquetes; dicho protocolo sería compatible con un modelo de capa de enlace utilizando el estado apagado para el manejo de energía. La respuesta es el DSR ya que este no requiere el intercambio periódico de mensajes de "hello" entre vecinos para comprobar la conectividad, haciendo posible que los nodos de la MANET puedan ponerse en el estado "sleep" para conservar su energía, ahorrándose además un ancho de banda considerable.

Por último, un enfoque híbrido ideado en [30], Enrutamiento con Capacidad de Batería Máxima y Mínima Condicional (CMMBCR), considera tanto la transmisión de energía total en el consumo de las rutas como la energía restante de los nodos. Sin embargo, no puede garantizar que los nodos con alta potencia restante puedan funcionar sin la potencia necesaria o cuando el tráfico pesado pasa a través del nodo.

En el trabajo previo [25], en una nueva métrica se ha propuesto la tasa de consumo, para ser utilizada con la carga residual de la batería y para predecir el curso de vida de los nodos de acuerdo con las actuales las condiciones del tráfico. Se describen los datos mínimos de consumo utilizando el mecanismo MDR que incorpora la métrica de

velocidad del consumo en el proceso de enrutamiento. Este mecanismo es básicamente una selección de la ruta, por ello, conocer el algoritmo de selección podría ayudar a aplicar al MANET, cualquier protocolo de enrutamiento, puede incluso ser utilizado para la selección del desempeño de la ruta [23-26].

1.3.3 Modelo de energía con tamaño de paquete y n nodos [32] .

En las figuras 1-6 y 1-7 se visualizan los datos que se derivan de las fórmulas para la energía $J(n)$ (ecuaciones 1.10 y 1.11) que la radio de una estación consume cuando se transmite con una energía consumida por Mega-Byte (JMB), con una longitud de paquetes de 1 Kilo-Byte de datos, en una red con n nodos o estaciones con el estándar IEEE 802.11. Los cálculos demuestran que $J(n)$ crece de forma proporcional o lineal con n , para $n > 4$. La energía útil consumida en la transmisión y la recepción de los datos con éxito es constante y la energía restante es desperdiciada. Cuando $n = 15$, las pérdidas de energía total ascienden al 80 %, y esta proporción crece con n . Más del 60 % de las pérdidas obedecen al *overgeding*, es decir, cuando hay recepción de paquetes dirigido a otra estación, el *overgeding* se elimina utilizando la información en paquetes RTS/CTS; se descompone la energía para el acceso básico, transmitiendo tx y recibiendo rx, ASK y RTS / CTS respectivamente.

Figura 1-6: Descomposición de la energía J_n para acceso básico con longitud de paquete = 1 KB. Fuente: [32].

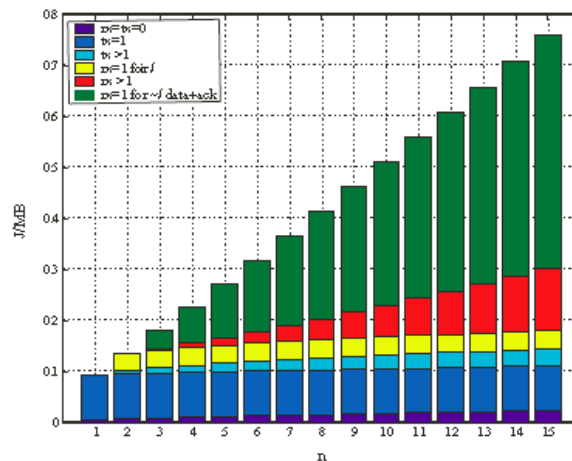
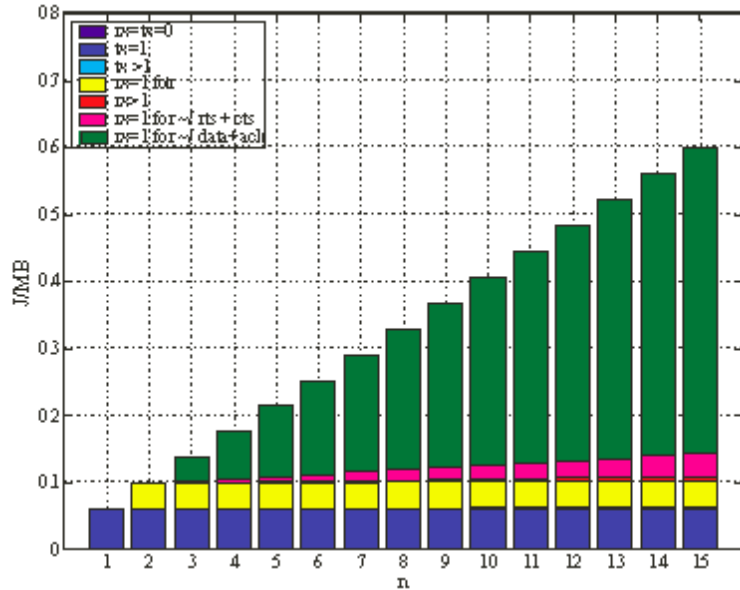


Figura 1-7: Descomposición de la energía J_n para RTS/CTS con longitud de paquete de 1 KB. Fuente: [32].



$J(n)$ corresponde a la energía total en Joules consumida por la estación l , con el fin de transmitir una longitud de paquete de 1 MB de datos. Debido a la simetría la energía $J(n)$ no depende de la estación o nodo l , pero sí depende de la energía consumida por l en un tiempo de duración del paquete. Con la probabilidad de una transmisión exitosa P_r , ecuación (1.22), la longitud del paquete transmitido L_p en un tiempo de duración t_p en el nodo l se calcula a continuación:

$$E [L_p \text{ transmitida por } l \text{ en un tiempo } t_p] = \frac{P_r}{n} L_p \tag{1.10}$$

Cuando se transmite con una longitud de paquete L_p en bytes por la estación l en un tiempo de operación o duración del paquete t_p , Se obtiene la energía:

$$J (n) = \frac{E [\text{energía consumida por } l \text{ en un } t_p]}{E [L_p \text{ transmitido por } l \text{ en un tiempo } t_p]} \tag{1.11}$$

$$J (n) = \frac{P_c * t_p}{\frac{P_r L_r}{n}} \tag{1.12}$$

En [33, 34] todas las estaciones móviles están equipadas con tarjetas de interfaz de red a una velocidad de transmisión R_p , con el protocolo inalámbrico IEEE 802.11. Los nodos tienen su energía inicial seleccionada al azar, ya que algunos nodos con un nivel de energía muy bajo y que no pueden iniciar la comunicación se les asigna una energía inicial a la fuente y a los nodos destino. El consumo de energía por estación es:

$$E = P_c * t_p \text{ Julios} \quad (1.13)$$

Donde P_c , es la potencia consumida si el nodo está transmitiendo un paquete y t_p es el tiempo necesario para transmitir un paquete de longitud $L_p = p_\eta + p_d$ a una velocidad R_p , es calculado por:

$$t_p = [(p_\eta + p_d) / R_p] \text{ segundos} \quad (1.14)$$

Donde p_η es la cabecera del paquete con longitud en bytes y el tamaño de los datos es p_d . Las corrientes necesarias para transmitir y recibir los paquetes son 280 mA y 240 mA respectivamente; 5 voltios son utilizados en las simulaciones en [33, 34], valores que son característicos de las tarjetas inalámbricas de PC LUCENT IEEE 802.11 WaveLAN.

En [35] se asume un modelo donde la radio disipa más energía durante la transmisión que en la recepción, donde cada nodo tiene una batería con energía residual limitada (E_r) cada nodo está equipado con antenas y pueden controlar los niveles de potencia, cuanto mayor sea el nivel de potencia y la distancia que cubran los nodos mayor ha de ser el consumo. El tiempo de vida útil t (s) en un nodo depende de la cantidad o carga de tráfico en el nodo de enrutamiento; t (s) también depende de la energía consumida durante la transmisión o recepción de la carga de tráfico. La energía residual E_r debe satisfacer la siguiente desigualdad:

$$\text{Velocidad de transmisión (b/s)} \times E_d \text{ (J/b)} \times t(\text{s}) \leq E_r \text{ (J)} \quad (1.16)$$

En donde la cantidad de tráfico es representada por la velocidad de transmisión que pasa a través de un nodo en bits por segundo (b/s), E_d es la cantidad de energía disipada en julios por bit (J/b), por el nodo durante la transmisión, recepción o ambos y el E_r es la energía residual o remanente en julios (J). Donde t es el tiempo de vida del nodo en segundos (s), entonces la desigualdad anterior se puede reescribir como:

$$t(\text{s}) \leq E_r \text{ (J)} / \text{Velocidad de Transmisión (b/s)} \times E_d \text{ (J/b)} \quad (1.17)$$

Un nodo se apaga cuando la energía consumida por comunicación excede su propia energía residual. La ecuación utilizada para calcular la energía consumida cuando un nodo está en comunicación está dada por:

$$E_c = \frac{P_c * L_p}{R_p} \quad (1.18)$$

Donde P_c es la potencia consumida si el nodo está transmitiendo un paquete, recibiendo o ambas cosas, L_p corresponde a la longitud o tamaño de paquete transmitido a una velocidad de transmisión R_p , con el fin de optimizar la potencia de transmisión en la interfaz del nodo con enrutamiento multi-salto de paquetes, ya que la energía consumida es proporcional a la distancia que se pretende alcanzar. Lógicamente, este tipo de enrutamiento incrementa el retardo total, siendo éste proporcional al número de saltos.

Por lo tanto, en este aspecto también se ha de encontrar un compromiso entre la energía consumida y el retardo promedio.

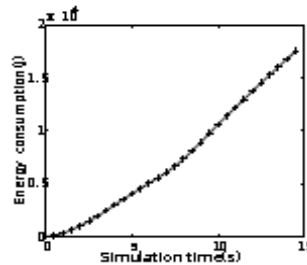
1.3.4 Modelo del consumo de energía utilizado NS3 [36, 37]

El simulador NS3 está desarrollado bajo programación de objetos, en [36] se define una clase que representa la fuente de energía básica denominada *Basic Energy Source* que almacena una cantidad inicial de energía E_0 en el usuario y este depósito de energía está continuamente utilizado por los objetos *Device Energy Model*. También almacena la muestra i de la corriente total en el nodo (I_i) y el marcador de tiempo asociada con ella (t_i). Cuando un dispositivo en un nodo cambia su estado operativo, la *Basic Energy Source* es notificada. El *Basic Energy Source* entonces calcula y almacena el valor de carga o corriente nuevo (I_{i+1}) y lo asocia al tiempo (t_{i+1}). La clase *Basic Energy Source* calcula el consumo de energía durante el período ($t_i, t_i + 1$) de la siguiente manera:

$$E_{i+1} = E_i + V \times (t_{i+1} - t_i) \times I_i \quad (1.19)$$

Donde E_i es el consumo de energía de la fuente en un tiempo inicial t_i y V es la tensión de alimentación. La actualización de la energía restante es accionada también desde dentro de la *Basic Energy Source*. Llamadas periódicas a la función *Update Energy Source* se programan con un intervalo definido por el usuario de manera que cuando se aplica una carga constante, *Basic Energy Source* todavía puede hacer un seguimiento de la reducción de la energía. Cuando las solicitudes de objetos externos para la información de energía desde el *Basic Energy Source*, por ejemplo, a través de *Get Remaining Energy*, la función *Update Energy Source* siempre se invoca de modo que la última información de energía será devuelta. En la figura 1-8 se simula la energía utilizando la ecuación 1.19 y 1.20 en un tiempo de simulación.

Figura 1-8: Simulación en [37] del consumo de energía en NS3 ecuación 1.19 y 1.20.



En el simulador NS3 también se tiene en cuenta la energía consumida por los nodos que están en un estado al llegar paquetes destinados a otro nodo (*overhearing*). El consumo de energía causado en “*overhearing*” por la transmisión de datos es la misma que la energía consumida por el hecho de recibir el paquete, de acuerdo con lo formulado en [33]. La cantidad total de energía, $E_{(n_i)}$, consumida en un nodo n_i se determina por:

$$E_{(n_i)} = E_{tx(n_i)} + E_{rx(n_i)} + (N-1) * E_{o(n_i)} \quad (1.20)$$

Donde E_{tx} , E_{rx} , y E_o denotan los consumos de energía por la transmisión, recepción y cuando hay recepción de paquetes dirigido a otra estación (*overhearing*) de un paquete, respectivamente. Cada uno de estos valores son calculados con la ecuación (1.13). El parámetro N representa el promedio de los nodos vecinos afectados por una transmisión de n_i del nodo, lo cual implica que cuando la red es más densa, el paquete “*overhearing*” causa más consumo de energía.

1.3.5 Modelo del consumo de energía con n saltos entre nodos

Para medir el consumo se establece la potencia de transmisión máxima de energía necesaria para la recepción de paquetes a un nodo n con un solo salto y con una fiabilidad de origen a destino P_{EE} . Para calcular cuál es el número de saltos en términos de consumo de energía, en primer lugar, se considera el caso de una cadena unidimensional de nodos equidistantes con distancia d . Considerando que E_0 es la energía requerida para una transmisión a través de la distancia d con fiabilidad de entrega de paquetes de extremo a extremo P_{EE} y una densidad de potencia del ruido N_0 y pérdidas por propagación en la trayectoria α , determinada por la distancia definida por el parámetro d^α , donde el consumo de energía está explícito en la ecuación:

$$E_0 = \frac{-d^\alpha \Theta N_0}{\ln P_{EE}} \quad (1.21)$$

La energía necesaria para cubrir una distancia total en n saltos entre nodos, con una fiabilidad de llegada de paquetes P_{EE} requiere de una energía en el primer salto calculado por: $E_1 = n^\alpha E_0$.

En el caso de saltos múltiples con n nodos, la probabilidad de transmisión de paquetes exitosa es:

$$P_r = \sqrt[n]{P_{EE}} \quad (1.22)$$

El rendimiento en la llegada de paquetes P_r requerida por cada salto, depende de la fiabilidad de extremo a extremo dada por:

$$\ln P_{EE} = n \ln P_r \quad (1.23)$$

El total de energía en este caso es: $n^2 E_0$, es reemplazando en la ecuación 1.20, se tiene:

$$E_n = n^2 \frac{-d^\alpha \Theta N_0}{\ln P_{EE}} \quad (1.24)$$

Para $\alpha = 2$, no hay ningún beneficio en absoluto en el enrutamiento multisalto.

Se asume un modelo para un canal con desvanecimiento de distribución Rayleigh con pérdidas por propagación. Una transmisión desde el nodo i al nodo j es satisfactoria si el

SINR i_j está por encima de un cierto umbral Θ que se determina por el hardware de comunicación, la modulación y la codificación en [18]. El SINR Y es un proceso aleatorio discreto:

$$Y = \frac{R}{N_o + I} \quad (1.25)$$

R es la potencia recibida que tiene una distribución exponencial con Media \bar{R} . Durante una transmisión de la distancia $d = ||x_i - x_j||$ con una pérdida de trayectoria dada por d^α en donde el promedio de la potencia recibida es $\bar{P} = P_0 d^{-\alpha}$ y donde P_0 es la potencia proporcional para transmitir; N_o denota la potencia de ruido e I es la potencia de interferencia que afecta la transmisión, es decir, la suma de las potencias recibidas de todos los transmisores no deseados; K corresponde a la constante de Boltzman que tiene un valor de $1,38 \times 10^{23}$ y T representa la temperatura absoluta, en grados Kelvin. La agitación térmica de los electrones está presente en cualquier dispositivo o conductor, denominada la cantidad de ruido térmico:

$$N_o = K * T \quad (1.26)$$

En una red con canal de desvanecimiento en banda estrecha Rayleigh, la probabilidad de recepción $p_r = \mathbb{P}[Y = \Theta]$ se puede factorizar en la probabilidad de llegada, en una red sin ruido y en la probabilidad de recepción sin interferencia. Sea R_o denota la potencia recibida de la fuente deseada y R_i , $i = 1, n$, la potencia recibida de n nodos. Todas las potencias recibidas se distribuyen de manera exponencial, es decir:

$$p_{R_i}(r_i) = \frac{1}{\bar{R}_i} \phi e^{-\frac{r_i}{\bar{R}_i}} \quad (1.27)$$

Donde \bar{R}_i denota el promedio de la potencia recibida $\bar{R}_i = P_i d^{-\alpha_i}$. La probabilidad de la señal de ruido $\text{SNR } Y_{N_o} = P_o / N_o$ es superior al umbral de Θ , es decir, la probabilidad de recepción sin interferencia.

1.4 Estimación de energía disipada en una MANET.

En [38] se da el estudio del consumo de energía de una red inalámbrica. Para la estimación de energía disipada, se asume una batería con máxima capacidad A_b de Amperio/hora y un voltaje promedio v_b . La máxima energía residual E_{rmax} puede ser calculada a partir de la siguiente expresión:

$$E_{rmax} = v_b \times A_b \times 3.6 \times 10^3 \text{ Joule} \quad (1.28)$$

Considerando y un periodo tiempo t_o , la energía consumida el nodo se expresa como:

$$E_o = v_b \times I_{eff} \times t_o \quad (1.29)$$

La energía consumida E_{tx} , para transmitir un paquete de tamaño k bytes y el tiempo de transmisión por bit es denotado por t_b . E_{tx} es calculado de la siguiente forma:

$$E_{tx} = v_b \times I_{tx} \times t_b \times k \times 8 \quad (1.30)$$

De igual manera, la energía consumida E_{rx} , para recibir un paquete puede ser calculada por:

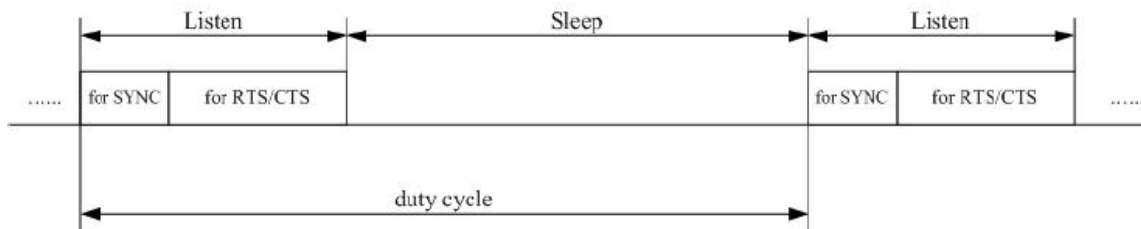
$$E_{rx} = v_b \times I_r \times t_b \times k \times 8 \quad (1.31)$$

1.5 Control del Consumo de energía.

En [3] y [5] se presenta el sensor-MAC (S-MAC), un nuevo protocolo MAC diseñado expresamente para redes de sensores inalámbricos, donde el objetivo principal es el consumo de energía. Este es un protocolo con buena escalabilidad y con la capacidad de evitar colisiones mediante el uso de una programación combinada y la utilización del esquema de contención. SMAC establece un ciclo de trabajo bajo para la operación de los nodos, además reduce el escuchar inactivamente mediante la desactivación periódica de los nodos, apagando completamente el transmisor/receptor. SMAC utiliza el mismo procedimiento tanto en RTS/CTS/DATA/ACK como en IEEE 802.11 [9]. Para lograr el objetivo principal de la eficiencia energética, se debe identificar cuáles son las principales fuentes que causan el uso ineficiente de la energía así como determinar qué ventajas y desventajas puede tener para reducir el consumo.

Muchas mediciones en [3] y [5] han demostrado que *escuchar inactivo* consume del 50 al 100 % de la energía requerida para recibir. Se denomina el retardo apagado (*sleep delay*), ya que es causado por el estado apagado del receptor. Se define el tiempo de ciclo completo del escuchar (T_{listen}) y el tiempo de la trama apagado (T_{sleep}) como se puede ver en la figura 1-9.

Figura 1-9: Los ciclos o periodos *listen* y *sleep* para S-MAC [3].



Se asume que un paquete llega al remitente con igual probabilidad en el tiempo dentro de una trama de datos (T_{Frame}). Así que el retardo promedio *sleep* en el remitente es:

$$D_s = T_{Frame} / 2 \quad (1.32)$$

$$\text{Donde, } T_{Frame} = T_{listen} + T_{sleep} \quad (1.33)$$

En comparación con los protocolos sin períodos de apagado, el ahorro de energía en S-MAC está definido por el ciclo de trabajo que puede indicar el porcentaje de ahorro de energía el cual se define como:

$$E_s = T_{sleep} / T_{Frame} = 1 - (T_{listen} / T_{Frame}) \quad (1.34)$$

El último elemento de la ecuación anterior es el ciclo de trabajo del nodo. Es conveniente que el tiempo de escucha sea lo más corto posible, de modo que para un ciclo de trabajo determinado, el retraso promedio de apagado es corto.

En [4], los nodos se orientan a la eficiencia de energía, arribo de paquetes y QoS, comparando su rendimiento contra el protocolo control de acceso al medio en sensores SMAC; los parámetros comparados son el tiempo que duran apagados los nodos, su consumo de energía y su retardo promedio. Este trabajo se realizó con la herramienta de simulación *OPNET Modeler* versión 11.5.A PL3, de la compañía OPNET Technologies Inc. Esta técnica mejora ampliamente el desempeño "PCF + Apagado" contra el SMAC.

La propuesta del trabajo [4], está basada en la modificación del protocolo PCF, de manera que les dé oportunidad a los nodos de ir apagándose después de ciertas condiciones. Con esto se consigue que ahorren energía y se tienen en cuenta requerimientos de calidad de servicio. En la propuesta llamada PCF + Apagado, el estado denominado apagado (*sleep*), dentro del modelo de procesos, corresponde a la subcapa MAC del estándar IEEE 802.11 en [9].

El estado *sleep* permite que los nodos manden sus radios a apagado para conservar energía. Los nodos, individualmente, controlan su estado de apagado mediante la formación de periodos apropiados de auto interrupciones de apagado y actividad. Lo siguiente que se tuvo en cuenta y se agregó en [4], fue que cada nodo fuera encuestado una sola vez dentro de un estado de actividad dada a aquellos nodos que explícitamente avisen el no tener datos por transmitir.

Cuando el nodo terminar de transmitir, se va a modo apagado. Si estando en el estado de transmitir se detecta que lo que queda de estado de actividad no es suficiente para transmitir una trama completa, además de apagar su transmisor y receptor, los nodos antes de pasar al estado de apagado tendrán que terminar lo que están haciendo. En caso de ser una transmisión no exitosa, es decir, dejar la trama en la cola de transmisión, deben volver a intentarlo después, pero sin incrementar el contador de intentos.

Todos los nodos de cada agrupamiento deben estar conscientes del tiempo de actividad y de apagado, de manera que todos los nodos lo primero que hacen al despertarse es activar una auto interrupción, necesaria para pasar al estado de apagado y posteriormente activar la segunda auto interrupción requerida para despertarse en la

siguiente trama. La PCF + Apagado, tiene una función que mide la energía consumida, la cual tiene en cuenta aquella que se está consumiendo en los diferentes estados en que puede estar un nodo.

Hay tres grupos de estados: activado, es cuando el nodo se encuentra en el estado *TRANSMIT* de *opnet*, transmitiendo alguna trama. Apagado, esto sucede cuando el nodo se encuentra en el estado nuevo de *SLEEP* de *opnet* y no está transmitiendo ni escuchando el medio. *No TRANSMIT*, ocurre cuando el nodo no está transmitiendo pero tiene al receptor encendido, se puede estar aquí si el nodo se encuentra en los estados *IDLE*, *DEFER*, *BACKOFF*, *BACKOFF_NEEDED*, *FRM_END*, *WAIT_FOR_RESPONSE* y *SCAN*.

Los resultados obtenidos en las simulaciones del modelo propuesto llamado PCF + Apagado en [4] y del SMAC en [3] y [5], son del tiempo dormido, del consumo de energía y del retardo promedio.

Una de las conclusiones del presente trabajo es que la implementación de un nuevo esquema de acceso al medio en el que los nodos vecinos que necesitan comunicarse directamente se coordinan de tal manera que ahorran energía mediante el apagado de su subsistema de comunicaciones durante los períodos en los cuales no tienen información por intercambiar.

Este mecanismo de acceso al medio está basado en el protocolo PCF (Función de Coordinación Puntual) descrito en el estándar de redes inalámbricas IEEE 802.11 en [9]. Se tuvieron en cuenta también los requerimientos de calidad de servicio en el sentido de que la frecuencia con que se presentan las ventanas de actividad coordinada y el tamaño de las mismas son suficientes para la transmisión oportuna del tráfico que se está generando. Como resultado del trabajo en [4] podemos decir que el nodo que más tiempo permaneció despierto es el que consumió mayor energía, los demás nodos consumieron menor energía y se fueron desgastando poco a poco. Se concluye también que el modelo propuesto, llamado PCF + Apagado, presenta un considerable ahorro de energía de todos los nodos sobre el modelo SMAC, además de tener menor retardo promedio.

En [7] y [8] se presenta un eficiente control de acceso al medio híbrido (HMAC), protocolo con una solución embebida de *cross-layer* de optimización, para ofrecer un transporte alternativo de extremo a extremo para la capa de enrutamiento y calidad de servicio (QoS), para tráfico sensible al retardo. Los esfuerzos de la presente investigación están encaminados hacia las redes inalámbricas de sensores, los cuales se centran principalmente en la exploración de la energía eficiente.

Los resultados de la simulación en NS2 muestran que HMAC supera el protocolo sensor de MAC (S-MAC) y al enrutamiento mejorado de MAC (RMAC) [6], en términos de retardo por salto y la relación de arribo de paquetes, manteniendo un rendimiento de consumo de energía superior.

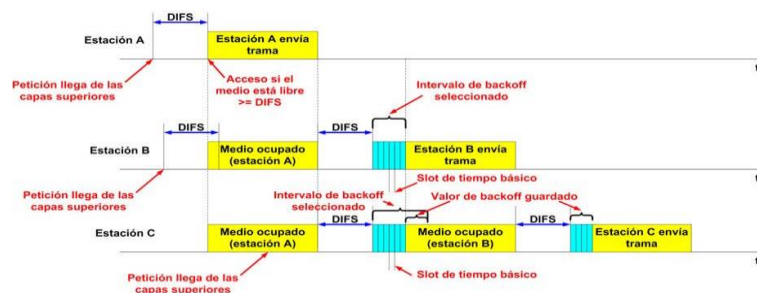
En [25] se propone el mejoramiento del consumo de energía a través de la capa de enlace. Lo que se busca es implementar el modo de dormido para evitar el agotamiento de las baterías, que sean capaces de alargar el tiempo de vida de las baterías de una red ad hoc. Se presenta un protocolo que conserva la capacidad de batería de los nodos, apagando aquéllos nodos que no se dedican a la transmisión y recepción activa de paquetes.

A la pregunta, qué protocolo sería compatible para ser usado con un modelo de capa de enlace utilizando el estado dormido para el manejo de energía, la respuesta es el DSR ya que DSR no requiere el intercambio periódico de mensajes de 'hello' entre vecinos para comprobar la conectividad, haciendo posible que los nodos de la red ad hoc puedan ponerse en el estado *sleep* para conservar su energía, ahorrándose además un ancho de banda considerable.

1.6 Control de acceso al medio MAC

El sistema de control de acceso al medio (N2 capa MAC) de IEEE 802.11 [9, 39], es la pila de protocolos DFWMAC (*Distributed Foundation Wireless Medium Access Control*), aunque este nombre es poco utilizado en la literatura al respecto. La base de DFWMAC es una técnica de coordinación distribuida llamada DCF (*Distributed Coordination Function*), su comportamiento se ve en la figura 1-5, donde en determinado tiempo el medio está ocupado mientras la estación envía la trama.

Figura 1-10: Función de coordinación distribuida. Fuente: [40].



IEEE 802.11 también define una técnica de coordinación centralizada llamada PCF (*Point Coordinated Function*) que solo está disponible en modo infraestructura. Esta técnica es opcional y además no se exige para los certificados de la alianza Wi-Fi, por lo que muy pocos aparatos lo implementan. El PCF alterna dos periodos de tiempo, periodos con conflictos (CP: *Contention Period*) y periodos libres de conflictos (CFP: *Contention Free Period*). Durante los CP las estaciones simplemente utilizan DCF. Durante los CFP el punto de coordinación el (AP) controla qué estación puede transmitir en cada momento de manera síncrona con un algoritmo Round-Robin. Esta coordinación centralizada permite ciertas gestiones de QoS, por ejemplo para conexiones sensibles al tiempo, como emisiones de vídeo. Además, se puede utilizar para minimizar el problema de los

nodos ocultos (si ningún nodo queda oculto al controlador). El principal inconveniente que tiene es que no define clases de tráfico.

La IEEE 802.11e define una nueva función de coordinación HCF (*Hybrid Coordination Function*) con el objetivo de incorporar garantías QoS y permitir aplicaciones de tiempo real. DCF utiliza un algoritmo CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*) con intercambio RTS/CTS opcional y acuse explícito de recibo. Usa comunicación asíncrona entre estaciones. La idea base de CSMA es que una estación que desea transmitir en un medio compartido, primero escucha el canal para verificar si tiene actividad. En caso de que el canal esté libre la estación comienza a emitir inmediatamente. Ethernet que utiliza CSMA/CD espera a que el canal quede libre (más la pausa requerida) y comienza a emitir. En caso de que otra estación estuviese también esperando para emitir se produce una colisión. Las estaciones emisoras detectan la colisión y esperan un tiempo aleatorio antes de volver a intentarlo. Sin embargo, dado que el rango dinámico de señales es muy amplio las NIC inalámbricas no son capaces detectar colisiones, por lo que no se puede usar CSMA/CD. En vez de ello se utiliza CSMA/CA.

CSMA/CA hace varias cosas:

- Verifica si el medio está libre u ocupado (CSMA);
- Cuando la línea está ocupada espera un tiempo aleatorio antes de volver a intentar enviar (CSMA);
- Incluye un mecanismo opcional de intercambio RTS/CTS antes de emitir el mensaje (CA);
- Incluye un mecanismo de acuse explícito de recibo a nivel de MAC (CA);

Es decir, cuando una estación inalámbrica desea transmitir, primero verifica que el canal esté libre durante un tiempo predeterminado. Si está libre comienza a emitir inmediatamente y si está ocupado espera primero a que quede libre y después un tiempo aleatorio antes de volver a verificarlo (CSMA).

Una vez la estación puede emitir utiliza un intercambio RTS/CTS (éste intercambio no se utiliza para mensajes muy cortos ya que la sobrecarga no vale la pena). Para este intercambio primero envía una trama corta de control de solicitud de transmisión RTS (*Request To Send*) indicando a las demás estaciones que no transmitan. Esta trama además especifica las estaciones origen y destino de la comunicación y el tamaño de la trama que se desea transmitir. Si la estación destinataria recibe correctamente esta señal devuelve una trama indicando que está ocupado, comunicándose con un nodo oculto (RxBUSY) o una trama indicando que todo está preparado para la emisión (CTS: *Clear To Send*) y el tamaño de trama que va a recibir.

Si una estación recibe el mensaje RTS pero no la respuesta (CTS o RxBUSY) sabe que es un nodo expuesto y que puede comunicarse con otro nodo a la vez.

Si una estación no ha recibido un RTS pero recibe un CTS sabe que es un nodo oculto y que debe esperar y además sabe cuánto tiempo debe hacerlo porque conoce el tamaño de trama que se va a enviar.

Tras emitir el mensaje, la estación emisora espera una respuesta del destinatario indicando una recepción correcta (ACK) o incorrecta (NACK). En el segundo caso o si no recibe respuesta el emisor volverá a emitir el mensaje, existiendo un límite para el número de posibles reenvíos. Sus principales inconvenientes son:

- Si muchas estaciones pretenden comunicar a la vez, ocurrirán muchas colisiones que disminuirán el ancho de banda disponible.
- No hay prioridades o clases de tráfico ni garantías de QoS.
- Cuando una estación gana el medio puede secuestrarlo. Por ejemplo una estación que transmita a 1 Mbps, puede tardar mucho tiempo en enviar un paquete, perjudicando al resto de estaciones.
- Para poder garantizar la usabilidad del canal se ha de sacrificar un porcentaje significativo de la capacidad incrementando el volumen de paquetes con tráfico de control.

1.7 Gestión de energía en equipos

La gestión de energía es especialmente importante en el caso de los equipos portátiles, pero también es un aspecto importante en otros sistemas. Hay disponibles dos tecnologías: APM (Gestión de Energía Avanzada, del inglés *Advanced Power Management*) y ACPI (Interfaz Avanzada de Configuración y Energía, del inglés *Advanced Configuration and Power Interface*). Además de estas tecnologías, también es posible ajustar la frecuencia de la CPU para ahorrar energía o reducir el ruido. Estas opciones se pueden configurar de forma manual o usando un módulo de YaST en [41]. A diferencia de APM, que se usaba anteriormente en los equipos portátiles solo para la gestión de energía, la información sobre el hardware y la herramienta de configuración de ACPI están disponibles en todos los equipos modernos (portátiles, equipos de sobremesa y servidores).

Todas las tecnologías de gestión de energía requieren hardware adecuado y rutinas de BIOS. La mayoría de los portátiles y muchos de los equipos de sobremesa y los servidores modernos se ajustan a estos requisitos. APM se utilizaba en muchos equipos antiguos. Dado que APM consiste, en gran medida, a un conjunto de funciones implementado en el BIOS, el grado de compatibilidad con APM puede variar en función del hardware. Este hecho es aún más evidente en el caso de ACPI, ya que presenta una mayor complejidad. Por esta razón no tiene mucho sentido recomendar uno u otro sistema. Lo mejor es probar los distintos procedimientos con el hardware y elegir la tecnología con la que se obtengan los mejores resultados.

En [38, 42] se presenta un estudio del estado del arte de los mecanismos para proveer seguridad relacionada con un algoritmo de distribución en grupos con energía eficiente, en las redes de sensores inalámbricas. Dentro de los principales esquemas de seguridad se encuentran los manejos de claves y criptografía, los protocolos de autenticación y los mecanismos de detección de intrusos o clones. En los mismos textos se enseñan los protocolos que solo trabajan con claves de grupo, claves públicas de superiores y claves de sesión con jefes de celda. Organizan la red en celdas mediante el algoritmo HEED (*Hybrid, Energy-Efficient, Distributed Clustering*), de esta manera existe una jerarquización de los nodos.

El protocolo busca la distribución de la clave para cada celda. Para esto, cada superior o cabeza de *clúster* genera una clave de grupo que envía a cada nodo de su celda. Esta clave de grupo se cifra con la clave que permite el enlace entre el superior y cada nodo, o mediante la clave pública del superior, las cuales son creadas en el despliegue. Cuando es necesario generar una nueva clave, el superior envía una petición de renovación de clave a la estación base (BS, *Base Station*) más cercana, luego de autenticarse mediante una firma digital, la cual contiene la clave privada y la identificación del superior.

1.8 Definición de variables en MANET

A continuación se definen las variables utilizadas para la aplicación de la lógica difusa y el uso de este método para medir el consumo de energía en redes ad hoc, se establece la relación que existen entre las diferentes ecuaciones representadas en el capítulo 1 como modelos del consumo para redes ad hoc, estudiando y analizando cada ecuación, determinando qué métricas permiten determinar el consumo energético en una red inalámbrica ad hoc, variables como el tamaño del paquete y número de saltos, utilizadas como entradas al motor de inferencia de la lógica difusa para determinar el consumo energético en una red.

El tamaño del paquete define la longitud de los paquetes que son cada uno de los bloques en que se divide la información. El tamaño hace referencia a la cantidad de datos en bits o bytes, donde el tiempo del paquete es el periodo que se tarda en transferir un paquete a una determinada velocidad de transmisión R_p que es un promedio del número de bits, caracteres o bloques, que se transfieren entre dos dispositivos, por una unidad de tiempo de duración en el paquete. Esto se presenta en la ecuación 1.14, donde p_h corresponde a la cabecera del paquete y p_d a los datos, entonces el tamaño del paquete corresponde a:

$$L = p_h + p_d \quad (1.35)$$

El número de nodos representan la cantidad de equipos que intervienen en la comunicación o en el transporte de paquetes de datos. Esta información puede viajar por diferentes rutas hasta llegar a su nodo destino. Las veces que viajan los paquetes de datos de equipo a equipo se denomina número de saltos.

Las pérdidas por trayectoria o propagación indican cómo la atenuación de la señal incrementa con la distancia d , son las pérdidas por obstáculos físicos más las pérdidas por la ruta que escoge la señal en el espacio libre que varía si se trata de un lugar rural o urbano. (Ver tabla 1-1) donde α es el exponente de pérdida de trayectoria.

Tabla 1-1: Pérdidas por trayectoria del medio en [31].

Medio	α
Área urbana de Radio Celular	2.7 a 3.5
Sombras de radio celular urbano	3 a 5
Línea de vista en Edificaciones	1.6 a 1.8
Obstruido en edificaciones	4 a 6

El área de cobertura está definida por parámetros como el ángulo de incidencia que indica la dirección del transmisor al receptor, que es el sector ϕ con respecto a la orientación en el plano.

La distancia hace referencia a la magnitud que mide la relación de lejanía o cercanía entre dos cuerpos, objetos o individuos, en este caso, a la separación entre nodos o equipos de radiocomunicación.

La energía de consumo se define como la rapidez con que se realiza un trabajo, expresando la potencia por el tiempo de trama de acuerdo con la ecuación 1.13. La energía consumida por bit se toma durante el tiempo de bit, una señal, digital o analógica, que contenga datos digitales binarios transmitidos a una determinada velocidad R_p en un tiempo de bit t_b . Teniendo en cuenta una potencia de $1 \text{ W} = 1 \text{ J/s}$, la energía por bit de la señal determinando el consumo es:

$$E_b = t_b \text{ vatios } \times \text{seg} \quad (1.36)$$

L es la longitud del paquete en bits, t_b es el tiempo necesario para transmitir un bit a una determinada velocidad de transmisión dada por:

$$R_p = L/t_b \quad (1.37)$$

El efecto de la potencia en el nodo transmisor dentro del tiempo de vida del enlace donde la degradación de la potencia de la señal se ajusta mediante el modelo de propagación en espacio libre formulado en [30], donde la potencia de la señal recibida es:

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \quad (1.38)$$

Donde P_t y P_r son las potencias de transmisión y recepción respectivamente en unidades de watts o vatios; G_t y G_r son las ganancias de las antenas en el transmisor y receptor respectivamente, siendo d la distancia o separación entre ambas. En las simulaciones el sistema de factor de pérdida $L = 1$, indica que no hay pérdidas en el sistema, donde λ es la longitud de onda portadora (en metros), que se relaciona con la frecuencia de la

portadora (f); donde c es la velocidad de la luz (3×10^8 m/s). Asumiendo una unidad de ganancia de la antena con una frecuencia portadora de 900 MHz.

Densidad de potencia del ruido: El ruido térmico se debe a la agitación térmica de los electrones. Está presente en cualquier dispositivo o conductor. La cantidad de ruido térmico presente en un ancho de banda B es:

$$N = N_o * B \quad (1.39)$$

El parámetro E_b/N_o es el cociente de la energía de la señal por bit entre la densidad de potencia del ruido por hercios. Donde la densidad de potencia del ruido N_o se mide en unidades de vatios por Hz; el ruido térmico está uniformemente distribuido. Por tanto, la energía de la señal sobre el ruido térmico es:

$$\frac{E_b}{N_o} = \frac{P_r B}{N * R_b} \quad (1.40)$$

Jitter: es la variabilidad temporal durante el envío de señales digitales, una ligera desviación de la exactitud de la señal de reloj. El *jitter* suele considerarse como una señal de ruido no deseada. En general se denomina *jitter* a un cambio indeseado y abrupto de la propiedad de una señal. Esto puede afectar tanto a la amplitud como a la frecuencia y la situación de fase. El *jitter* es la primera consecuencia de un retraso de la señal. La representación espectral de las variaciones temporales se denomina ruido de fase.

1.9 Análisis de criterios de rendimiento de consumo en MANET.

En la tabla 1-2 se hace una evaluación de las ecuaciones más frecuentemente usadas como métricas de consumo. En esta evaluación se tiene en cuenta si se cumple con los criterios o variables del consumo de energía para el rendimiento de una red inalámbrica ad hoc. En las ecuaciones (1.1) y (1.8) se analizan factores que afectan el consumo como las pérdidas por trayectoria, el número de nodos y el área de cobertura o sector de localización. Dichas ecuaciones son usadas para medir el consumo en una red multisalto con n saltos y son una buena representación de redes dinámicas como las MANET. Adicionalmente son una métrica para la eficiencia en el enrutamiento, sin embargo, en este caso, no son usadas como métrica del consumo de energía debido a que el aumento en el número de saltos entre nodos y la distancia variable en un sector Φ en un radio de rango RA, suministra un incremento en variable aleatoria del consumo de energía en MANET, por lo cual es un reto aplicarlas en la simulación de un sistema de monitoreo del consumo.

Por otro lado, las ecuaciones (1.13) y (1.18) son métricas del consumo que están relacionadas con la velocidad de transmisión, con el tamaño del paquete y con el tiempo que tarda en llegar la información en forma de paquetes; adicionalmente son las métricas utilizadas por defecto en el simulador NS3. La ecuación (1.24) es utilizada para medir el

consumo de acuerdo con el tráfico relacionado de forma aleatoria con la probabilidad de llegada de paquetes de extremo a extremo hasta el nodo n de recepción. La igualdad entre las ecuaciones (1.1), (1.13) y (1.15) en donde se suman los consumos transmitiendo, recibiendo y en modo de espera, permite que se relacionen todas las variables ya que son complementarias, pues se analizan diferentes parámetros en la MANET. Las ecuaciones (1.13), (1.18), (1.15) y (1.19) son ecuaciones utilizadas en los algoritmos de simuladores de redes, especialmente en NS3; éstas determinan el consumo en el equipo o nodo, pero su comportamiento tiende a ser lineal con valores proporcionales y adicionalmente están integradas en un sistema de simulación que tiene en cuenta otros modelos como el de propagación, movilidad y enrutamiento, pero tienen la desventaja de no tener en cuenta las pérdidas por trayectoria, que le da un comportamiento exponencial a la gráfica del consumo.

Las ecuaciones (1.1) y (1.8), con variables aleatorias, son teóricas. Sus parámetros comunes tienen en cuenta un grupo de n nodos en un sistema de enrutamiento y la localización en un sector Φ y pérdidas por trayectoria α , que son los criterios más importantes para la medida del consumo en una MANET. Las ecuaciones descritas en la tabla 1-2 son comparadas y seleccionadas para ser usadas en la técnica de inteligencia computacional. Las ecuaciones (1.1) y (1.13) son escogidas por relacionar las variables que afectan el consumo de energía ya que según lo anteriormente expuesto se cumple con los criterios de las métricas del consumo en una MANET de la tabla 1-2.

Tabla 1-2: Análisis de criterios del consumo de energía de redes ad hoc.

Criterios o métricas de Rendimiento	Ecuación (1.1) y (1.8) $E = n^2 \left(\frac{2}{\Phi} \right)^{\frac{\alpha}{2}} \Gamma \left(1 + \frac{\alpha}{2} \right)$ $= \sum_{u \in N} c(RA) RA(u)^\alpha$	Ecuación (1.13) ,(1.18) , (1.19) y (1,20) $E = P_c * t_p = \frac{P_c * L}{R_p}$ $E_{i+1} = E_i + V \times (t_{i+1} - t_i) \times I_i$ $E_{(ni)} = E_{tx(ni)} + E_{rx(ni)} + (N-1) * E_{o(ni)}$	Ecuación (1.12) $J(n) = \frac{P_c * t_p}{\frac{P_r}{n} L_r}$	Ecuación (1.24) $= n^\alpha \frac{E}{\ln P_{EE}} \frac{-d^\alpha \Theta N_0}{}$
Longitud de paquetes		*	*	*
Pérdidas de trayectoria α	*			*
Tiempo		*	*	
número de nodos	*		*	*
Área o sector de cobertura RA o Φ	*			

* Indica que la ecuación incluye el criterio o métrica de rendimiento del consumo de energía.

2. Técnicas de inteligencia computacional

La Inteligencia artificial es una nueva aplicación empleada en el monitoreo, gestión, auditoria de redes y en los sistemas actuales empleados en la industria y en el control automático en [43-46]. Los nuevos sistemas de gestión de red están basados en desarrollos de inteligencia computacional, de forma que el sistema de gestión permite descarga de trabajo al administrador de la red en [1].

2.1 Lógica Difusa

Las bases teóricas de la lógica difusa fueron enunciadas a principios de la década de 1960 por Lotfi Zadeh, quien la concibió como una generalización de la lógica tradicional de dos valores que permiten una escala en los niveles de verdad por lo que incorporan un continuo infinito de estados intermedios entre proposiciones “absolutamente falsas” y “absolutamente verdaderas”.

La lógica difusa se puede entender desde dos puntos de vistas, según lo expuesto en [47, 48], de manera que:

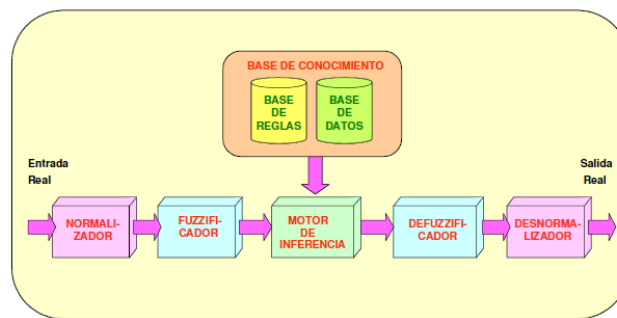
- a) En un sentido estricto, la lógica difusa es básicamente una lógica de razonamiento aproximado o una generalización y extensión de lógica multivaluada.
- b) En un entendimiento amplio, la lógica difusa consiste en clases de objetos en los que la pertenencia o no pertenencia es más gradual que abrupta.

Un Sistema Difuso, en sentido amplio, es un sistema basado en lógica difusa, donde ésta puede utilizarse como base para la representación de diferentes formas de conocimiento, o para modelar las interacciones existentes entre las variables de un sistema. Las principales potencialidades en torno a la lógica difusa que son relevantes son: (i) habilidades para controlar procesos no lineales: característica relevante de aplicación e identificación y control de sistemas dinámicos complejos con características no lineales y alto orden; (ii) utilización de términos y expresiones utilizados en el lenguaje natural en [49]; (iii) control de procesos cuando un modelo matemático no es evaluado dentro de una compleja aplicación de metodologías de la teoría del control clásico; (iv) posibilidades de implementación del conocimiento, aspectos intuitivos y experiencia de especialistas en control utilizando reglas lingüísticas en [50]: (conocimiento cualitativo y estructurado) y entradas imprecisas (lo contrario a las redes neuronales donde el conocimiento es configurado de forma no estructurada y cuantitativa).

El resultado de un control difuso basado en reglas presenta una superficie de control codificada en una base de conocimiento, son un conjunto de reglas, términos lingüísticos y factores de escala ejecutadas por un controlador dedicado, interpretador o compilador. El control difuso presenta un flujo de datos que pasan por las fases de (i) fusificación, (ii) análisis y ejecución de reglas y (iii) desfusificación [45, 50].

Una base de conocimiento agrega una base de reglas y una base de datos para el control difuso. La base de reglas es proyectada por la selección de las variables de entrada, variables de salida y reglas de control y una fase de desfusificación convierte las variables numéricas en variables lingüísticas.

Figura 2-1: La lógica difusa representada en diagrama de bloques [46].



A continuación se describen, de manera somera, las funciones de cada uno de los elementos de la figura 2-1.

a) Normalizador: este módulo realiza la transformación de los valores tomados por las variables de entrada del controlador al intervalo normalizado de valores [0,1] o [-1,1].

b) Fusificador: en este elemento se transforma un valor preciso tomado por las variables de entrada, en una información difusa, asignándole un grado de pertenencia a cada uno de los conjuntos difusos definidos en cada variable. El grado de pertenencia se define mediante una función característica asociada a cada conjunto difuso; de manera que para cada valor “x” que pueda tomar una variable de entrada, la función característica “ $\mu(x)$ ”, proporciona el grado de pertenencia de este valor “x” al conjunto difuso “A”. Para realizar la citada transformación, cuando el valor tomado por una variable es un valor preciso, se suele utilizar un operador de fusificación, también llamado de desfusificación, denominado “fusificador singleton” que, para un conjunto difuso “A” definido sobre un universo de discurso “U”, viene dado por la siguiente expresión:

$$\mu(x) = \begin{cases} 1, & \text{Si } x = x_0 \\ 0, & \text{Si } x \neq x_0 \end{cases} \quad x \in U \quad (2.1)$$

Tal como se ha indicado, este “desfusificador” es utilizado para describir formalmente una medida precisa mediante una función de pertenencia equivalente.

c) Motor de Inferencia: en este módulo tiene lugar el proceso de inferencia de acciones de control difuso utilizando implicaciones difusas y las reglas de inferencia de la lógica

difusa, es decir, el motor de inferencia obtiene un conjunto difuso para cada variable de salida, a partir de los conjuntos difusos de las variables de entrada, conforme a una relación definida tomando como referencia el conjunto de reglas difusas.

d) Base de Conocimiento. Este elemento funcional, contiene dos tipos diferenciados de información, almacenadas en las siguientes bases:

1) Base de Datos. Contiene información acerca de la forma de los conjuntos difusos, que es utilizada en el proceso de defusificación.

2) Base de Reglas. Almacena una colección de reglas lingüísticas, que serán utilizadas por el motor de inferencias.

e) Defusificador: como resultado del proceso de inferencia se obtiene un conjunto difuso. Teniendo en cuenta que las acciones de control difuso no son aplicables a ningún sistema a controlar, será necesaria la existencia de un elemento funcional que transforme los conjuntos difusos inferidos, para cada variable de salida, en un valor puntual, que pueda ser aplicado al sistema o proceso a controlar.

f) Desnormalizador: este módulo transforma los valores tomados por las variables de salida del controlador, que están comprendidos en el intervalo $[0,1]$ o en el intervalo $[-1,1]$, al rango de valores adecuado, para que dichas variables de salida, hagan actuar correctamente al sistema controlado.

En los casos de estudio se puede observar que la inteligencia computacional, en especial la lógica difusa, apunta hacia aplicaciones industriales [49] y al manejo de control automático y de procesos [44, 51] reevaluando los sistemas actuales de PID en [45] y abriendo campo a los PLC (Controlador Lógico Programable). Se puede ver así mismo, cómo los sistemas no lineales encuentran mejor respuesta en un entorno de lógica difusa o en cualquier otro concepto de inteligencia bioinspirada a través de, por ejemplo, algoritmos genéticos, redes neuronales, etc. Por ello, donde la ingeniería actual pretendía la linealidad de los procesos para hacer un manejo más fácil de los mismos, ha encontrado una forma de manejar estas técnicas que rompe con los paradigmas.

El presente texto, además de presentar el concepto de diferentes autores, pretende reevaluar la programación clásica y su lenguaje estructurado y la programación de objetos a partir de más hallazgos para la aplicación de la inteligencia computacional como la lógica difusa.

2.2 Algoritmos genéticos

Los algoritmos genéticos son procedimientos adaptativos para la búsqueda de soluciones en espacios complejos inspirados en la evolución biológica, con patrones de operaciones basados en el principio darwiniano de reproducción y supervivencia de los individuos que mejor se adaptan al entorno en el que viven.

Los principios básicos de los algoritmos genéticos fueron propuestos por Holland en [52]. La idea básica consiste en mantener una población de individuos también llamados cromosomas, que representan posibles soluciones al problema que evolucionan en el tiempo mediante un proceso controlado de variación y competición. Cada individuo tiene asociado un grado de bondad o de adaptación, que determinará qué cromosomas serán utilizados para obtener una nueva generación de cromosomas y cuáles se perderán.

Los nuevos cromosomas serán generados empleando operadores de entrecruzamiento y mutación, que modelan los procesos biológicos de reproducción que suceden en la naturaleza. Los algoritmos genéticos comenzarán con una población inicial de individuos (cromosomas), generados de forma aleatoria. Esta población evolucionará hacia otra población con mejores individuos que se obtendrán aplicando operadores genéticos, que simularán los procesos genéticos que se dan en la naturaleza. Durante sucesivas iteraciones denominadas generaciones, los individuos serán evaluados por su adaptación al objetivo, y con base en estas evaluaciones se formará una nueva población de individuos empleando un mecanismo de selección y operadores específicos para el entrecruzamiento y la mutación. Para realizar la evaluación se diseñará una función de evaluación específica de cada problema a resolver, función que devolverá un valor numérico, el cual será proporcional a la adaptación a la solución que presenta cada individuo.

Los algoritmos genéticos han sido aplicados con éxito a una amplia gama de problemas. Las razones a las que se debe este éxito son:

- a) Los Algoritmos Genéticos pueden encontrar solución a problemas difíciles de manera rápida y fiable.
- b) Los Algoritmos Genéticos son fácilmente integrables en modelos y simuladores existentes.
- c) Los Algoritmos Genéticos son fácilmente escalables.
- d) Los Algoritmos Genéticos son fácilmente hibridables.

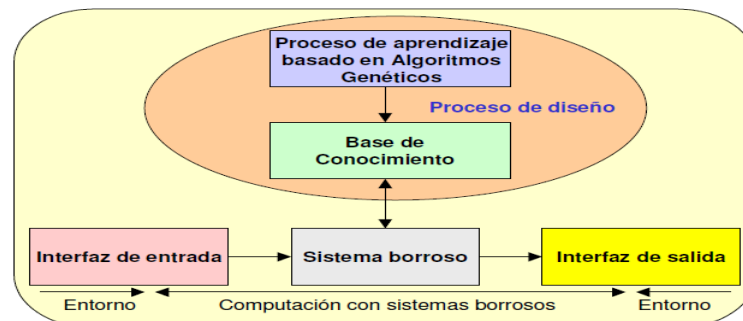
Todas estas razones se pueden resumir en una sola, los algoritmos genéticos son robustos, son más potentes en ambientes difíciles con espacios de búsqueda grande, discontinuos, complejos y pobremente conocidos. A pesar de no garantizar la obtención de la solución óptima al problema, los algoritmos genéticos generalmente aportan una solución buena y rápida a los problemas. Por esta razón, son ampliamente utilizados en multitud de campos. Los principios básicos de los Algoritmos Genéticos son los siguientes [52, 53]:

- a) Representación genética de las soluciones al problema. Es decir, un esquema de codificación de las posibles soluciones como una secuencia de bits. Donde cada cadena, será una posible solución (cromosoma) y cada bit un gen.
- b) Un conjunto inicial de soluciones al problema, es decir, una población inicial " $P(0)$ ".

- c) Una función de evaluación, que permita estimar la bondad de cada una de las posibles soluciones, de la población en cada generación.
- d) Operadores genéticos que permitan obtener una nueva generación de soluciones al problema, en función de la información genética contenida en la población.
- e) Valores para los parámetros utilizados por los algoritmos genéticos (tamaño de la población, probabilidad de aplicación de los distintos operadores genéticos, etc.).

La figura 2-2 por [46] ilustra la idea expuesta en estos párrafos. En el caso de los sistemas difusos basados en reglas, este conocimiento estará almacenado en forma de variables lingüísticas, parámetros que definen las funciones de pertenencia difusas, reglas difusas, número de reglas, etc. Desde el punto de vista de la optimización, la tarea de buscar una base de conocimiento adecuada, para resolver un problema específico, consiste en definir una base con un conjunto de parámetros que describan las reglas difusas y los conjuntos de pertenencia, para posteriormente obtener los valores de esos parámetros, de acuerdo con el criterio de optimización. Así, los parámetros de la base de conocimiento, que constituyen el espacio de optimización, forman el espacio fenotipo, que será transformado en una representación genética adecuada, el espacio genotipo. En los siguientes apartados se aborda el estudio de las características del proceso de aprendizaje con algoritmos genéticos en sistemas difusos basados en reglas.

Figura 2-2: Diagrama de bloques algoritmos genéticos en [46].



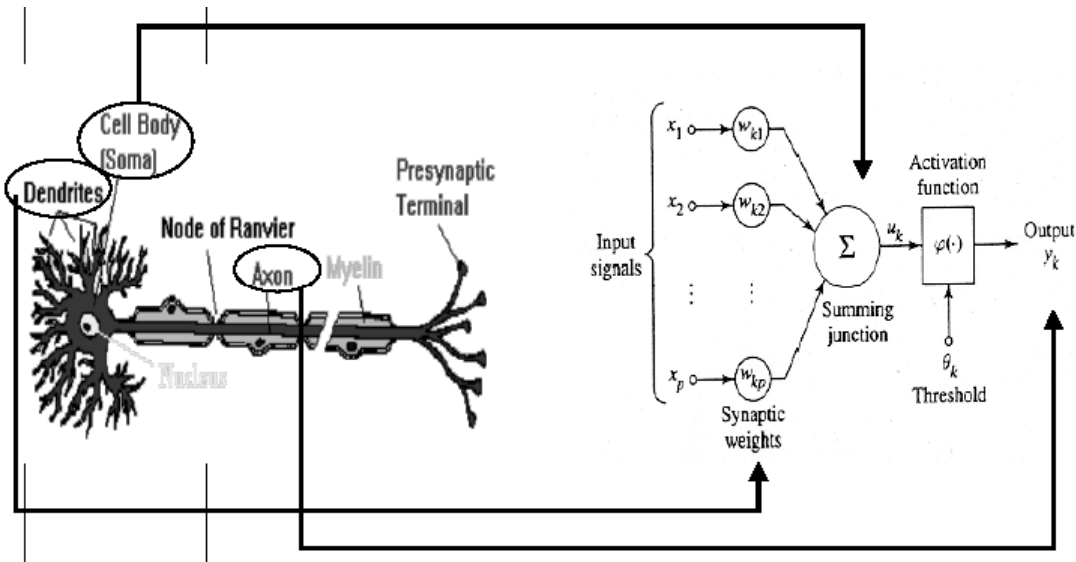
2.3 Redes neuronales

Las redes neuronales artificiales (ANN) son sistemas paralelos para el procesamiento de la información, inspirados en el modo en el que las redes de neuronas biológicas del cerebro procesan información [54]. Una red neuronal es el intento de poder realizar una simulación computacional del comportamiento de partes del cerebro humano mediante la réplica en pequeña escala de los patrones que éste desempeña para la formación de

resultados a partir de los sucesos percibidos. Concretamente, se trata de poder analizar y reproducir el mecanismo de aprendizaje y reconocimiento que posee el cerebro humano.

Las ANN tienen características similares a las del cerebro, son robustas y tolerantes a fallos. En el cerebro mueren todos los días gran cantidad de neuronas sin afectar sensiblemente el funcionamiento del mismo; el cerebro se adapta a nuevas circunstancias mediante el aprendizaje. Las ANN son flexibles, pueden trabajar con información borrosa, inconsistente, incompleta, probabilística o con ruido; son altamente paralelas. El cerebro está formado por muchas neuronas interconectadas entre sí y es precisamente el comportamiento colectivo de todas ellas lo que caracteriza su forma de procesar la información. El punto clave de las ANN es la nueva estructura de estos sistemas para el procesamiento de la información, estos están compuestos, al igual que el cerebro, por un número muy elevado de elementos básicos (las neuronas), altamente interconectados entre ellos y con modelo de respuesta para cada elemento en función de su entorno, muy parecido al comportamiento de las neuronas biológicas como se puede ver en la figura 2-3.

Figura 2-3: Diagrama de una red neuronal. Fuente: [54].



Con las funciones suma Σ y transferencia $f(x)$ se obtienen las salidas correspondientes de cada neurona:

$$f(w_{ij} x_j) = Y \quad (2.2)$$

Y : Salida de la neurona i

$F()$: Función de transferencia

w_{ij} : Ponderación de entrada j de la neurona i

x_j : El valor de entrada de j

2.4 Selección de la técnica de inteligencia computacional

En la tabla 2-1 se cuantifican y ponderan los datos cualitativos según lo presentado en [55] sobre aplicaciones de lógica difusa y otros métodos de inteligencia artificial.

Para evaluar qué tan favorable es el criterio de selección, se da un valor de cantidad a un valor atributivo de favorabilidad, donde el dato inferior es el desfavorable y el superior es el favorable. Se califica con criterios o variables de acuerdo con la siguiente asignación.

- 1 Desfavorable.
- 2 Poco favorable
- 3 Favorable
- 4 Muy favorable

Las técnicas de inteligencia computacional estudiadas y analizadas anteriormente son: la lógica difusa, las redes neuronales y algoritmos genéticos. Aquí se da una descripción de cada una de las técnicas, su funcionamiento y aplicaciones analizando diferentes criterios que serán evaluados como la complejidad de implementación, autoaprendizaje, tiempo de procesamiento y otros factores que determinan cuál técnica es la más favorable a los requerimientos del sistema de monitoreo del consumo de energía en la MANET.

En la tabla 2-1 se presenta la selección de tecnologías de un sistema de inteligencia artificial (IA). Por medio de algunas propiedades se espera saber qué tan propicio es un sistema inteligente con respecto a otro, con lo cual se determina qué ventajas se pueden obtener. Valorando y ponderando esas ventajas se puede obtener la tecnología IA que más rendimiento da en el sistema que se propone, el cual consiste en medir el consumo de energía en una MANET. De los datos analizados y del resultado la tecnología que obtuvo el mayor valor de ponderación es la lógica difusa.

Como criterio principal se tuvo en cuenta la complejidad de implementación, analizando el modelo matemático en todo lo que tiene que ver con el comportamiento de la función y gráfica del consumo de energía en cuanto a la no linealidad, además se revisó el desarrollo del algoritmo, qué tan complejo es y qué tanto se ajusta a la lógica matemática. Otro discernimiento es qué tanto representa los conocimientos humanos, qué tanto tiene de tolerancia al acierto, cuál es el tiempo de procesamiento, o sea el periodo de respuesta. Este último es uno de los factores que más importantes a lado de los demás criterios, los cuales son ponderados para obtener la tecnología que mejor se adapte al modelo.

El sistema debe representar la métrica de la energía de una MANET que es monitorizada en tiempo real, requiere que el comportamiento en el tiempo de procesamiento se ajuste a un lapso mínimo de respuesta requerido que depende de la velocidad de transmisión en la MANET, para una rapidez de 1 Mbps se requiere un tiempo en milisegundos, para 1Gbps un lapso en microsegundos ya sí sucesivamente incrementando el flujo de información.

Se espera que esta métrica de energía tenga una tolerancia al error o fallo mínimo para que pueda ser diagnosticado el gasto de energía de la MANET y que su aplicabilidad sea fácil de implementar. Las ecuaciones representan los criterios de rendimiento del consumo en un algoritmo de inteligencia computacional, adicionalmente el de ser multivariable ya que se utilizan diferentes métricas tanto a la entrada como a la salida del software de lógica difusa, teniendo como universos de entrada tanto las variables relacionadas con el gasto de energía y su desempeño en la MANET, como el número de saltos y la longitud del paquete donde el sistema tiene como respuesta el universo de salida el consumo de energía.

Tabla 2-1: Selección de tecnología de un sistema de inteligencia artificial IA.

Propiedades	Lógica Difusa	Redes Neuronales	Algoritmos genéticos
Complejidad de implementación	3	1	1
Autoaprendizaje	1	4	2
Representación del conocimiento humano	4	1	1
Lógica matemática	4	1	1
No linealidad	3	4	4
Tolerancia a la falla	4	4	4
Tolerancia al acierto	4	4	3
Tiempo de procesamiento	4	3	1
Promedio total	3,375	2,75	2,125

3.Análisis de simulación MANET

A continuación se valoran los diferentes simuladores de red, en función de su aplicación para redes inalámbricas. Los simuladores escogidos se enuncian en [56]: NS2, NS3, GloMoSim, J-Sim, GTNetS, JIST/SWANS, NCTUns y OMNeT++.

3.1 NS2

Su desarrollo empezó siendo un software Libre con Licencia GPL denominado Real Network Simulator en [57, 58]. Inicialmente fue diseñado para redes fijas, luego el Grupo Monarch de CMU implementó aplicaciones para la simulación de redes inalámbricas incluyendo a las redes ad hoc y WLAN (Wireless Local Área Networks). El NS2 se desarrolla en dos Lenguajes de programación, por un lado, utiliza versión extendida del tc, denominada Otcl para que el usuario codifique las especificaciones del escenario que desea simular y por otro lado, la implementación de los protocolos se encuentran en C++, este tipo de simulación permite obtener datos que son visualizados a través de una interfaz gráfica denominada NAM (Network Animation) que interactúa con lenguaje HTML.

3.2 NS3

NS3, siguiente versión del NS2 pero no compatible con esta. Su amplio uso se ve enunciado en [59]. Network Simulator 3 (NS-3) es un simulador de redes de eventos discretos, es un software libre licenciado bajo la licencia GNU GPLv2. En NS3 las librerías para la simulación están escritas en C ++ y Python y el código fuente disponible. Este simulador se ejecuta en entornos Linux y variantes de Unix, OS X y Windows a través de Cygwin. Con NS3 se simula MANET en [60], para recrear diferentes escenarios con diferentes modelos de propagación de señales en medios inalámbricos, llevar a cabo el proceso de control de acceso MAC, se pueden emplear diferentes modelos de movilidad enunciados en el capítulo 1; se pueden aplicar diferentes tipos de enrutamiento en MANET formulados en [61], como AODV, DSR, DSDV y TORA adicionalmente permite la simulación de diferentes protocolos de comunicación.

El poder manejar varias reglas hace de esta una herramienta muy robusta y útil para la simulación de redes de telecomunicaciones. Los módulos existentes permiten simular varios tipos de redes inalámbricas (por ejemplo MANET, WiFi, WiMax y LTE), de una manera sencilla, al escribir un programa (script) de simulación. Gracias a la disponibilidad del código fuente de los módulos es posible modificar el funcionamiento de cualquier modulo del simulador. También es posible crear nuevos módulos que implementan algoritmos o protocolos de aplicación que no están presentes en el simulador, el código fuente. Para el modelo de consumo de energía en WiFi está en [62] al igual que su antecesor NS2 se necesita de la interfaz gráfica NAM que se interconecta con un script en lenguaje Python e interactúa en HTML.

3.3 GloMoSim

GloMoSim en [63] se define como un entorno de simulación para redes inalámbricas, no es software libre, pero tiene licencia académica. Está programado en Parsec, un lenguaje basado en C especialmente diseñado para la simulación de eventos discretos en redes de comunicación de gran escala ya que soporta miles de nodos. Su desarrollo está inconcluso, la última versión estable data de diciembre de 2000. Su desarrollador Rajive Bagrodia en UCLA se ha inclinado por la versión propietaria de GloMoSim denominada QualNet a través de la empresa SNT (Scalable Network Technologies), programa del que sí se lanzan nuevas versiones periódicamente. En el punto en que quedó el desarrollo de GloMoSim tenía un soporte muy limitado de protocolos WiFi, únicamente código para 802.11b en modo ad-hoc, aunque existe documentación sobre la implementación en modo infraestructura en [64] y soporte para 802.11e en [65], no se ha encontrado código que implemente ningún protocolo WiMAX, aunque sí existe una investigación en [66], sin código disponible.

3.4 GTNetS

Georgia Tech Network Simulator definido en [67] es un simulador bajo licencia de software libre BSD, desarrollado en C++, usado para el diseño de topologías de gran escala y con gran número de nodos como las redes de sensores.

Se descarta por el limitado soporte de protocolos WiFi únicamente el 802.11b y la falta completa de soporte para WiMAX. No tiene acceso libre al repositorio de código.

3.5 SWANS

SWANS (Scalable Wireless Ad-hoc Network Simulator) en [68], es un simulador desarrollado en JIST (Java in Simulation Time). Fue diseñado principalmente para grandes redes de sensores y para la simulación de redes ad-hoc, sin embargo, no se ha

hecho ninguna implementación para el modo infraestructura. Solo soporta el estándar 802.11b, no se ha encontrado información sobre implementación del resto de estándares WiFi ni nada para WiMAX.

3.6 NCTUns

NCTUns en [69] es un simulador y emulador de redes cableadas e inalámbricas desarrollado en la University of NCTU (National Chiao Tung University) de Taiwan, comercializado y apoyado por Sim Real Inc. (empresa virtual fundada en 2002 con el fin de promover el uso de NCTUns). Su licencia es para usos no comerciales o lucrativos, por lo tanto no es software libre. Es desarrollado en código C++, instalado en distribuciones de Linux. Utilizado en los protocolos inalámbricos: 802.11a infraestructura AP/cliente, 802.11a ad-hoc, 802.11b infraestructura AP/cliente, 802.11b ad-hoc, 802.11e, 802.11p, 802.11d (Mesh y PMP), 802.16e, GPRS (General Packet Radio Service) y DVB-RCST (Digital Video Broadcast – Return Channel Satellite Terminal).

3.7 OMNET++

OMNeT++ en [70] es una plataforma de simulación con una arquitectura modular y extensible, tan flexible que se utiliza en ámbitos tan diversos como el modelado de redes, protocolos, sistemas de colas, multiprocesadores o arquitecturas hardware. En general, es un simulador útil para cualquier tipo de sistema en que pueda modelarse por la simulación, por eventos discretos y por intercambio de mensajes.

El simulador de red OMNeT++ tiene un diseño orientado a objetos programados en C++ simple y modular, lo que le permite escalar bien en la simulación de grandes redes. La entidad base en OMNeT++ es un módulo que puede ser atómico o compuesto por otros submódulos (hasta un nivel arbitrario de anidamiento). Los módulos se combinan para crear estructuras más complejas.

La topología de una red con el lenguaje propio NED.OMNeT++ fue diseñado para los protocolos inalámbricos ad hoc WiFi denominado INETMANET, el cual soporta los protocolos 802.11a/b/g en modos ad-hoc e infraestructura, con algunas limitaciones: no hay soporte para fragmentación, control de potencia ni PCF (Point CoordinationFunction). Para 802.11e existe una implementación en [71] muy limitada y aparentemente no ha sido validada por la comunidad de OMNeT (no se cita en ninguna otro sitio). En cuanto al soporte para larga distancia en 802.11, ACKTimeout y Slot Time son parámetros en el código. No existe soporte para WiMAX excepto el que provee el proyecto Numbat en [72], que implementa únicamente 802.16-2005(e) (nodos móviles).

3.8 Resumen de características de los simuladores para MANET

Tabla 3-1: Resumen de características de los simuladores para MANET

Simulador	Lenguajes	Protocolo inalámbrico	Modo operación red
Ns2	Octl y C++	802.11, 802.11a, 802.11b, 802.11e, 802.11g,802.16	Infraestructura ad-hoc
Ns3	C++ y Python	802.11a, 802.11b, 802.11e, 802.11g, 802.16, LTE	Infraestructura ad-hoc
GloMoSim	C y Parsec	802.11b, 802.11e	ad-hoc
GTNetS	C++	802.11b	ad-hoc
SWANS	JIST	802.11b	ad-hoc
NCTUns	C++	802.11a, 802.11b, 802.11d, 802.11e, 802.11p ,802.16e,	Infraestructura ad-hoc
OMNeT++	C++ y NED	802.11a, 802.11b, 802.11g, 802.11e, 802.16e,	Infraestructura ad-hoc

Según la tabla 3-1, para el presente trabajo se selecciona el NS3 por sus diferentes aplicaciones en un variado tipo de protocolos inalámbricos, con varias tecnologías de redes como WiFi, WiMax, LTE, MANET y MESH en modo ad hoc e infraestructura; adicionalmente ya tiene incorporados los módulos de protocolos de criterios de rendimiento de la MANET como la movilidad, enrutamiento, propagación y consumo de energía, con código fuente disponible, por ser software libre, en un lenguaje comprensible como C++ y Python; su programación hace fácil modificar los módulos de protocolos de aplicación del simulador, desarrollando algoritmos nuevos o simplemente modificando los ya implementados.

La estructura de su código en un lenguaje como C++ y Python permite que NS3 interactúe con otro tipo de software donde se desarrolle la técnica de inteligencia artificial; adicionalmente permite configurar su entorno con la ayuda de otro software como NAM, el cual, por medio de su interacción con el lenguaje Python y web HTML, permite la animación de las redes de datos.

Otra aplicación por medio de la interconexión con un script en Python es el de utilizar el software de radio enlace móvil en [56], con diferentes entornos como rural, movimiento vehicular y urbano. Para la interfaz de los gráficos se utiliza MATLAB, gnuplot y para las trazas pcap, como software intermedio a través de un script en Python para reportes o tablas en txt y una de las aplicaciones más importante en MATLAB para el procesamiento, análisis y adquisición de señales de entrada y salida del sistema, estadísticas, método de inteligencia computacional, los resultados de simulación y IA en graficas o trazas etc.

4. Resultados y análisis de la simulación en MANET.

En las siguientes simulaciones se presenta como resultado el consumo de energía en MANET. Se obtuvieron cuatro escenarios diferentes, en el primer escenario hay una variación en el número de saltos representando diferentes consumos de acuerdo a la ecuación 1.1; en el segundo escenario, se varía el número de saltos utilizando los modelos del NS3 y las ecuaciones 1.13, 1.19 y 1.24, aunque es una emulación que representa al sistema real, se tiene la desventaja de ser una aproximación lineal del consumo. En el tercer escenario se varía el número de saltos en la ecuación 1.1 de una forma aleatoria, obteniendo como resultado un consumo variable para dar en términos de tiempo; se reemplaza en la ecuación 1.13. Y en el cuarto escenario, se toman diferentes longitudes de paquetes en forma incremental utilizando la ecuación 1.18 que es la misma 1.13 pero en términos de tamaño de paquete, lo cual da como resultado una variación proporcional en el consumo.

Tabla 4-1: Parámetros comunes de los escenarios de simulación para MANET

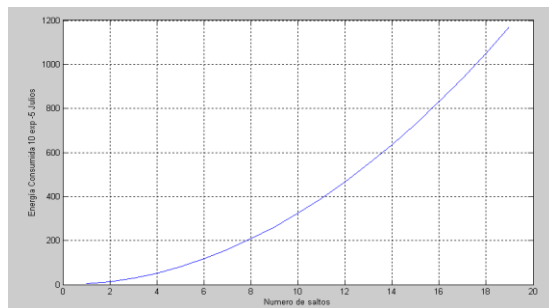
Parámetro	Cantidad
Número de saltos entre nodos	$1 \leq n \leq 20$
Pérdidas de trayectoria o propagación α	4
Sector ϕ	$\pi/2$ radianes
Protocolo inalámbrico	802.11b modo ad hoc
Modelo de propagación zona urbana	Distribución Rayleigh
Modelo de red o localización	Distribución Rayleigh
Modelo de Movilidad	Random Walk
Protocolo de enrutamiento	AOVD
Consumo de energía	Ec. 1.1, 1.13, 1.18, 1.19 y 1.24

4.1 Escenario 1, consumo por número de saltos

En el capítulo 1 se analiza la ecuación 1.1, donde se tienen en cuenta diferentes variables tabla 4-1, como pérdidas de trayectoria α que dependen de los obstáculos físicos y otros parámetros como la localización dada por el Sector ϕ . Estas medidas afectan el consumo de energía en una MANET, un mayor número de saltos entre nodos

representa un incremento exponencial del consumo de energía según datos representados en la figura 4-1, donde el valor del ángulo de incidencia ϕ es de $\pi/2$ radianes y las pérdidas de propagación α que corresponden a la constante debido al medio de zona urbana con un valor de 4 dado en las tablas 1-1 y 4-1.

Figura 4-1: Simulación del consumo en MATLAB utilizando la ecuación (1.1).



4.2 Escenario 2, el modelo de energía en NS3 [73]

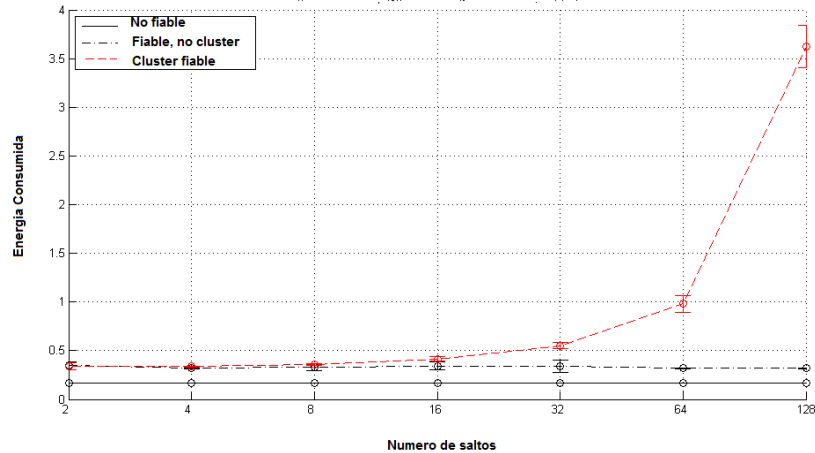
El modelo del consumo en una red WiFi es emulado por NS3 en el código de [62] y aplicado en [18, 37, 74], utilizando las ecuaciones 1.13, 1.19, 1.24 y diferentes parámetros como el modelo de energía en NS3, de una red inalámbrica 802.11b en modo ad hoc. En el modelo de movilidad posición constante, en el enrutamiento AOVD y en el modelo de pérdidas por propagación, factores tenidos en cuenta en la tabla 4-2 para la simulación, (Ver figura 4-2) donde se visualiza por intervalos un comportamiento lineal. Esto es debido a que en el simulador NS3 la representación del modelo de consumo de energía, es lineal como se puede observar en los diferentes rangos como por ejemplo el de 32 a 64 números de saltos, y en su conjunto tiende a la forma exponencial. Se puede ver graficando una mayor cantidad de nodos en una simulación más extendida,

En la figura 4-2 se observan 3 tipos de resultados, lo no fiable, representado por líneas continuas, donde no hay transmisión de paquetes entre nodos por lo tanto la energía es constante con un consumo de 0.1 julios de los nodos de la red, si todos los equipos estuvieran apagados la energía tendería a cero. El fiable, con trazas discontinuas, en una transmisión que no involucra un grupo de nodos, una comunicación que sigue una ruta óptima con un consumo continuo de 0.3 julios.

Por último, la línea roja donde hay un clúster fiable de nodos transmitiendo en una red aleatoria que sería el característico de MANET. Aquí clúster hace referencia a que se toma un grupo de nodos que intervienen en la comunicación de paquetes y fiable que constata una transmisión de tramas entre nodos con una fiabilidad en la llegada de paquetes, representada en las ecuaciones 1.22 y 1.23. La ruta óptima es la línea fiable, donde se consigue el ahorro en el consumo, por ser ese el camino más corto, que determina la métrica del trayecto óptimo en el enrutamiento de la información.

Tabla 4-2: Parámetros adicionales del escenario 2 de simulación para MANET

Parámetros	Cantidad
Número de saltos entre nodos	$1 \leq n \leq 128$
Modelo de propagación	pérdidas por propagación
Modelo de movilidad	posición constante

Figura 4-2: Simulación del consumo de energía en [73].

4.3 Escenario 3, consumo variando el número de saltos en ecuación 1.1

Al aplicar la ecuación 1.1 y variando el número de saltos entre nodos de forma aleatoria entre $1 \leq n \leq 20$, y con los parámetros dados en la tabla 4-3, manteniendo constante las pérdidas de trayectoria en una zona urbana con valor de cuatro y en un sector de $\pi/2$ radianes, y reemplazando 1.1 en 1.13, para obtener los términos en función del tiempo, se obtiene la simulación de la figura 4-3 que constituye variaciones en el consumo de energía en micro-julios que se presentan en la MANET, con diferente número de saltos, representado por una variación de las distancias entre los nodos, es un proceso de Poisson con una densidad de probabilidad Rayleigh en un modelo de radio enlace y localización con diferentes rutas que escogen los n saltos al azar en el envío de paquetes, para el enrutamiento de la información, lo anteriormente expuesto determina la eficiencia en la ruta, con la mejora en la métrica del trayecto óptimo o ruta más corta en los protocolos de enrutamiento en [61], donde hay una disminución en la métrica del consumo.

La localización de los nodos vecinos se constituye en un proceso de distribución tipo Poisson homogéneo, ecuación (1.2). La orientación de n nodos en un sector a $\phi/2$ radianes del eje x , un vector distancia r en las coordenadas cartesianas (X, Y) y Polares (r, Ψ) . El argumento ψ es una distribución uniforme entre $-\phi/2$ y $\phi/2$, como

experimento se tiene la distancia r al vecino más cercano en un sector con ángulo de incidencia ϕ . La densidad de probabilidad de la distancia al n -ésimo vecino más cercano en un sector ϕ es

$$p_{R_n}(r) = r^{2n-1} \left(\frac{2}{\phi}\right)^n e^{-\frac{r^2}{2} \phi} \quad (4.1)$$

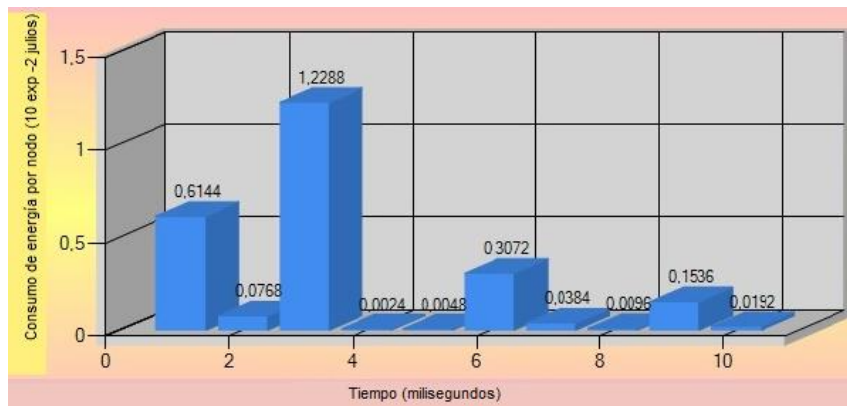
Esta función de densidad de probabilidad tiene una distribución Rayleigh con una distancia media dada en la ecuación (1.5) y la varianza proporcionada en la ecuación (1.7); el valor esperado de la energía consumida es la ecuación (1.1).

Tabla 4-3: Parámetros adicionales del escenario 3 de simulación para MANET

Parámetro	Cantidad
Tiempo de paquetes	$1 \leq t \leq 10$ mili-segundos
Distancia Media	Ec. (1.5)
Varianza	Ec. (1.7)
Valor esperado de energía consumida	Ec. (1.1)
Densidad de probabilidad de la distancia	Ec. (4.1)

En la ecuación (1.13), cada transmisión tarda un tiempo de 1 milisegundo por paquete, se presentan transmisiones durante un periodo $1 \leq t \leq 10$ mili-segundos; se despeja el consumo, luego se reemplaza en la ecuación (1.1) variando la cantidad de saltos de forma aleatoria entre $1 \leq n \leq 20$, con lo que se obtiene la figura 4-3. Así la variable aleatoria número de saltos es una métrica que afecta la cantidad de energía, por lo cual se elige como universo de entrada al sistema de lógica difusa.

Figura 4-3: Simulación del consumo de energía con variación de número de saltos.



4.4 Escenario 4, consumo incrementando el tiempo del paquete

Las ecuaciones 1.13 y 1.18 representan el incremento del consumo de energía conforme aumenta la longitud del paquete en una MANET. En ellas se presenta una relación proporcional entre las variables como se ve en la figura 4-4; por otra parte, en la ecuación 1.13 se tienen en cuenta los parámetros de la tabla 4-4, tales como el tiempo de los paquetes en $1 \leq t \leq 10$ mili-segundos, que es el tiempo que tardan las tramas en ser transmitidas, lo cual a su vez depende de la longitud del paquete entre $1 \leq L \leq 10000$ bytes, la velocidad de transmisión de 1 Mbps y la potencia en los equipos, que es la medida que más relación tiene con el consumo de energía en una MANET. Ahí intervienen un voltaje de 5 voltios y una corriente de 240 mA, dando una potencia de 1,2 vatios, valores que son característicos en las tarjetas inalámbricas PCLUCENT IEEE 802.11 WaveLAN, de los nodos de la red.

Tabla 4-4: Parámetros adicionales del escenario 4 de simulación para MANET

Parámetros	Cantidad
Voltaje del nodo	5 voltios
Corriente del nodo	240 mili-amperios
Potencia del nodo	1,2 vatios
Tiempo de paquetes	$1 \leq t \leq 10$ mili-segundos
Velocidad de transmisión	1 Mbps
Longitud de paquete	$1 \leq L \leq 10000$ bytes
Consumo de energía	Ec. 1.13 y 1.18

Figura 4-4: Consumo incrementando el tiempo de duración del paquete, ecuación 1.13.



4.5 Escenario del consumo de energía usando el método de lógica difusa.

Se utilizan dos métricas como universo de entrada al sistema de lógica difusa. La primera variable de ingreso es el número de saltos entre nodos que tienen un comportamiento exponencial calculado en la ecuación (1.1). Esta representa el consumo de energía que va aumentando su nivel, si para la comunicación entre nodos, la información que llega al nodo destino pasa por un mayor número de saltos, cuyos consumos de energía están determinados por un incremento exponencial en la medida que ocurran más saltos en la comunicación entre nodos,

La otra variable es la longitud del paquete la cual es lineal y proporcional con la energía, (Ver figura 4-4): El tiempo del paquete se determina de acuerdo con la ecuación (1.13), la cual depende de la longitud, de la velocidad de transmisión y del tiempo del paquete. En la ecuación (1.18), la variable que más relación tiene con el consumo de energía es la potencia en el nodo, la cual depende de la tarjeta inalámbrica.

4.5.1 Reglas del Modelo de energía en MANET con el método de lógica difusa

Los datos obtenidos en la simulación son representados en rangos de la función de membrecía o universo de entrada longitud de paquete de acuerdo con la figura 4-5. Se elige un tamaño máximo de paquete de 10000 bytes, unidades características de transmisión de datos en protocolos como IP o una LAN Ethernet. Para el caso de IP en [75], el máximo valor de la Unidad Máxima de Transferencia (*Maximum Transfer Unit*-MTU) es 65.536 bytes. No obstante, ese es un valor máximo teórico, pues en la práctica, la entidad IP determina el máximo tamaño de los datagramas IP en función de la tecnología de red por la que vaya a ser enviado el datagrama. Por defecto, la longitud del datagrama IP es de 576 bytes. Solo pueden enviarse datagramas más grandes si se tiene conocimiento fehaciente de que la red destinataria puede aceptar ese tamaño.

En la práctica, dado que la mayoría de máquinas están conectadas a redes Ethernet o derivados, el tamaño de datagrama que se envía es con frecuencia de 1500 bytes. Ejemplos de MTU para distintos protocolos usados en Internet: Ethernet: 1518 bytes, PPPoE: 1492 bytes, ATM (AAL5): 8190 bytes, FDDI: 4470 bytes, PPP: 576 bytes. Para esto se tiene en cuenta el tamaño en bytes de dichos protocolos con el fin de determinar unos rangos.

Para las reglas de la lógica difusa los paquetes son divididos en 8 fracciones de 1250 bytes, donde la longitud del paquete se fracciona por intervalos de longitudes cortas, medias y largas, en número de bytes representa la cantidad de paquetes, lo mismo se sucede con la energía que tiene un máximo de 0,1 julios divididos en 8 periodos de 0,0125 julios, y se determinan los niveles de energía en bajos medios y altos, que se

representaran, a su vez, en una función de membrecía o universo de salida del sistema, representados en la figura 4-7.

El consumo de energía por un número de saltos, obtenidos en la simulación y representados en intervalos de acuerdo con la función de membrecía, figura 4-6, se seleccionan de la figura 4-1, graficando los datos de la ecuación 1.1.

Si la información es transmitida entre 1 y 14 número de saltos entre nodos se da una cuantía esperada de energía en un nivel bajo; si los paquetes que llegan superan esta expectativa en un rango de 14 a 17 saltos de enrutamiento, el valor esperado de energía es intermedio. Una vez superado este intervalo se obtiene un consumo alto, lo cual determina los rangos donde haya pocos, medios y muchos saltos que representan el número de saltos, estos intervalos y su inferencia sobre su cuantía simbolizan las reglas de la lógica difusa.

En cuanto al universo de entrada, el número de saltos y el universo de salida de energía, son funciones exponenciales con variables aleatorias, como lo representan las ecuaciones teóricas formuladas para el análisis de criterios del consumo 1.1, 1.8, 1.21 y 1.24, siendo el sector o localización de los nodo y el factor exponente por las pérdidas por trayectoria $\alpha=4$, los de mayor afectación en el consumo energético; α es un valor constante que depende del lugar donde se encuentre ya sea en una área rural o urbana y del tamaño de las edificaciones, valor seleccionado de la tabla 1-1. Se escoge como entrada al sistema el número de saltos variable de la ecuación (1.1) que involucra el parámetro α y en un sector de $\pi/2$.

La lógica difusa es algo muy subjetivo para determinar qué valores están entre los medios y cuál entre los altos. Para ello no se vale solamente de la percepción sino que hace un análisis matemático, estadístico, integral en un intervalo de tiempo de la señal de energía dadas por las ecuaciones 1.1 y 1.13, o áreas bajo la curva de las gráficas 4-1 y 4-4, con lo cual se determina qué consumos de energía están en determinado intervalo de bajo, medio y alto. Además se tienen dos herramientas para hacer pruebas con diferentes escenarios por medio del simulador NS3 y Matlab.

Figura 4-5: Universo de entrada, longitud de paquete.

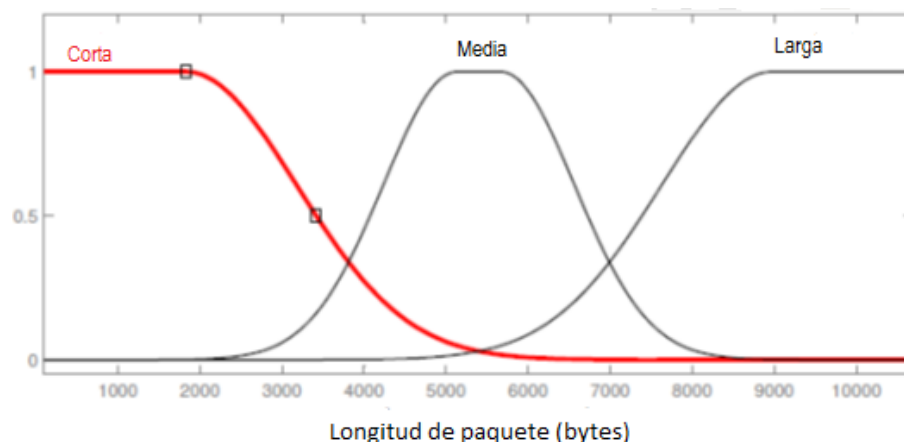
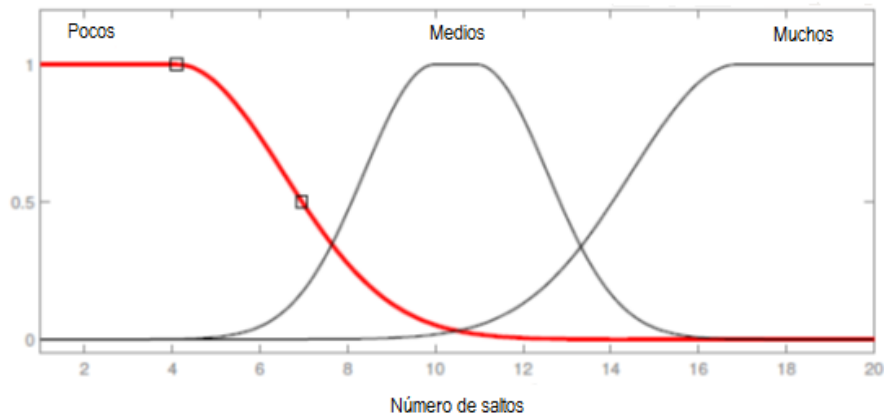
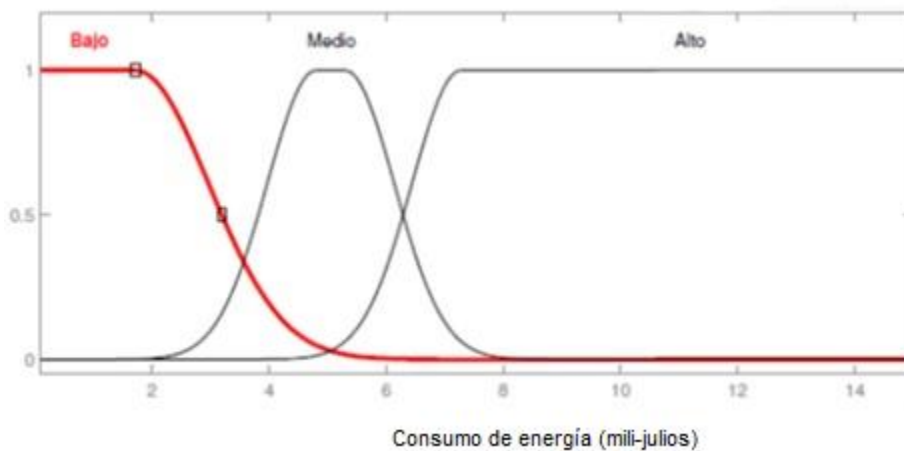


Figura 4-6: Universo de entrada, número de saltos.**Figura 4-7:** Universo de salida, consumo de energía.

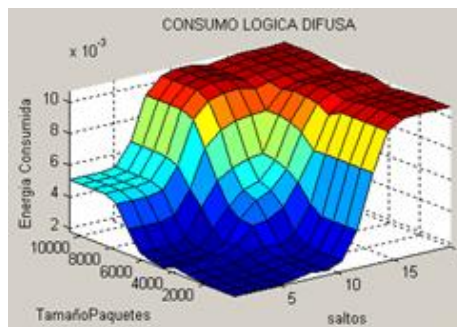
Los universos de entrada estarán representados por dos variables, longitud o tamaño del paquete y número de saltos. Las reglas que se determinan por medio del análisis de los datos arrojados por la simulación que se ingresaran al motor de inferencia están relacionadas por:

1. Si el número de saltos son pocos y la longitud del paquete es corta, entonces el consumo es bajo.
2. Si el número de saltos son pocos y la longitud del paquete es media, entonces el consumo es bajo.
3. Si el número de saltos son pocos y la longitud del paquete es larga, entonces el consumo es medio.

4. Si el número de saltos son medios y la longitud del paquete es corta, entonces el consumo es bajo.
5. Si el número de saltos son medios y la longitud del paquete es media, entonces el consumo es medio.
6. Si el número de saltos son medios y la longitud del paquete es larga, entonces el consumo es alto.
7. Si el número de saltos son muchos y la longitud del paquete es corta, entonces el consumo es alto
8. Si el número de saltos son muchos y la longitud del paquete es media, entonces el consumo es alto.
9. Si el número de saltos son muchos y la longitud del paquete es largo entonces el consumo es alto.

Los resultados obtenidos a través del método de la lógica difusa valida los datos adquiridos por medio de la simulación y la aplicación de las fórmulas. Con ese método se determina que las variables están relacionadas entre sí, que el número de saltos y el tamaño del paquete afecta el consumo energético, dado que el transporte de datos de información en forma de paquetes, a través de un determinada cantidad de nodos o saltos, da un incremento en la energía; que, entre mayor sea el número de eventos de saltos que ocurran en la ruta hasta la llegada de los paquetes, como se visualiza en la figura 4-8, mayores serán los resultados del consumo obtenidos. En la gráfica de superficie en tres coordenadas se determina la relación que hay entre el número de saltos, el tamaño de paquetes y el consumo de energía. Los valores bajos están representados en azul y los altos en color rojo.

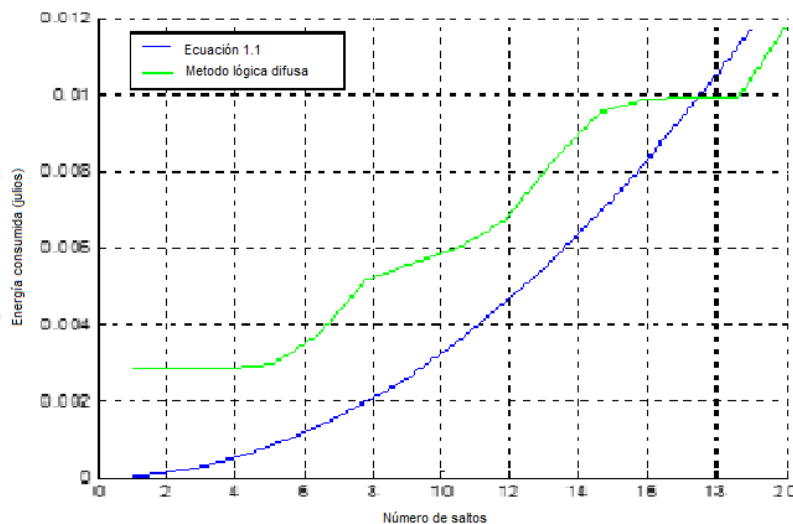
Figura 4-8: Interfaz en MATLAB, resultado del consumo con lógica difusa.



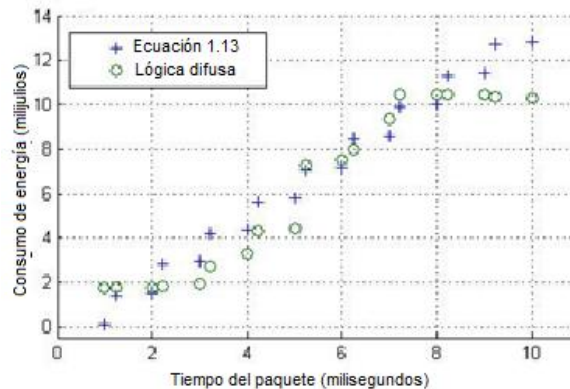
4.6 Escenarios de comparación entre modelos de consumo de energía.

En la figura 4-9 se ve la comparación entre la simulación del escenario 1 con la ecuación 1.1 y los resultados de la métrica de energía en lógica difusa, variando el número de saltos y manteniendo constante el tamaño del paquete. Se puede observar que de la comparación entre los dos se obtienen datos muy aproximados con un margen de error muy pequeño. Esto convierte a la lógica difusa en una herramienta muy útil para determinar el consumo de energía a través de otras métricas como el tamaño del paquete y el número de saltos. Además, permite comprobar cuándo un consumo es bajo, medio y alto, con lo cual se convierte en un método para monitorear una MANET y poder establecer así su eficiencia energética.

Figura 4-9: Comparación de datos Ecuación 1.1 y lógica difusa.



Ahora bien, al ingresar los datos en Matlab se obtiene la comparación entre los valores del escenario 4 en NS3 con la ecuación 1.13. Esto se puede evidenciar en la representación de la figura 4-10 con asteriscos y los datos de la lógica difusa con círculos de color verde. En la ecuación 1.13 la gráfica es lineal y la ecuación 1.1 es exponencial. El método denominado Consumo de Energía es un método de simulación que emula la realidad de un sistema físico, es una métrica de potencia por duración de paquete representada en las ecuaciones 1.13 y 1.18.

Figura 4-10: Comparación Ecuación 1.13 y lógica difusa MATLAB.

En la comparación con las simulaciones realizadas en MATLAB utilizando la lógica difusa y el escenario 4 en NS3, a través de la ecuación 1.13 se calcularon las medidas con incrementos desde valores bajos hasta finalizar con datos altos del consumo de energía en una MANET. La sumatoria total de consumos se calcula para la obtención de los datos.

En la Tabla 4-5 se dan las variables estadísticas de la media, varianza y desviación estándar con el fin de comparar los promedios de consumo en los modelos de energía en la ecuación 1.13 y contrastarlos con el método de la lógica difusa, tomando el universo de salida métrica del consumo de energía. Así el error absoluto establece la diferencia entre medidas, es decir, cuán equidistantes están los niveles de la señal de energía en tiempos discretos o discontinuos punto a punto. Adicionalmente la media es el promedio de estas medidas y el error relativo establece el porcentaje o error promedio entre medidas.

Tabla 4-5: Variables estadísticas de la comparación de consumos en una MANET

Variable estadística	Consumo Ecuación 1.13 Valor real	Consumo Lógica difusa Valor medido	Error Absoluto	Error Relativo
Media	0,006821286	0,006252702	0,0009	17,72 %
Varianza	1,50E-05	1,37E-05	4,22E-07	1,33 %
Desviación estándar	0,0038762	0,003701746	0,000649596	11,54 %

En la figura 4-11 se presentan los valores de error absoluto de la comparación de las medidas de simulación (ecuación 1.13) y el método de lógica difusa. Con estos se determinó que los datos son aproximados, el mínimo error absoluto fue de 0,00024 y el máximo error absoluto fue de 0,0025, con un promedio de error absoluto de 0,009. Por

tanto el porcentaje de error mínimo es del 4 % y el porcentaje de error máximo es del 36 %; donde hay más porcentaje de error es en los valores bajos del consumo de energía y al final de las simulaciones. Entre tanto en los valores intermedios se mantuvieron estables. Estadísticamente los consumos de ambos modelos contaron con promedios muy aproximados en cuanto a la energía total consumida en la MANET.

Figura 4-11: Error absoluto de la comparación ecuación 1.13 y lógica difusa.

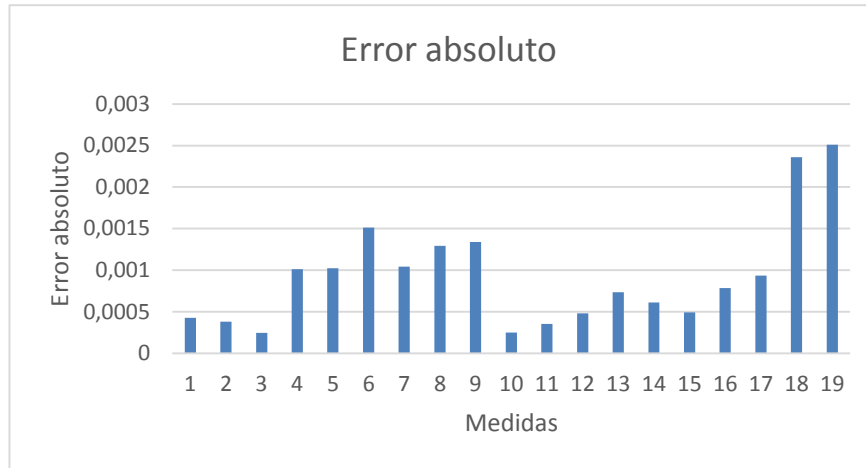
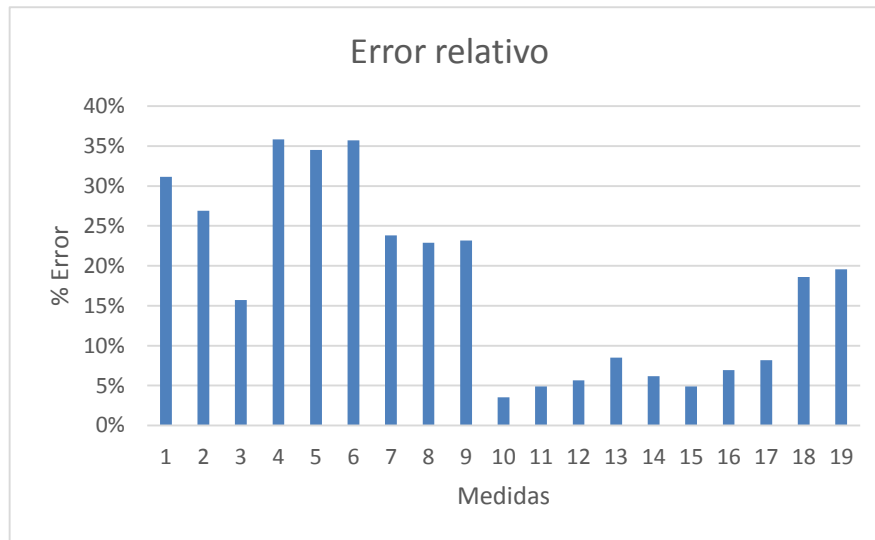


Figura 4-12: Error relativo de la comparación ecuación 1.13 y lógica difusa.



En la Tabla 4-6 se presentan las variables estadísticas media, varianza y desviación estándar de la comparación de los modelos del consumo de MANET, las simulaciones realizadas en MATLAB utilizando lógica difusa y el escenario 1 a través de la ecuación 1.1. Igualmente se calculan las medidas en incrementos desde valores bajos hasta

finalizar con datos de consumo de energía altos en una MANET; la sumatoria total de consumos se computan para la obtención de los datos estadísticos de la energía en los modelos propuestos, el porcentaje de error permite determinar qué tan aproximada es la medida de la energía entre el método de la lógica difusa y el modelo del consumo (Ver ecuación 1.1) que representa la medida real. Aquí, el error absoluto es la diferencia entre el valor real y el valor medido, y el error relativo es la razón entre el error absoluto y valor real.

Tabla 4-6: Variables estadísticas de la comparación de los consumos en una MANET.

Variable estadística	Consumo ecuación 1.1 Valor Real	Consumo Lógica difusa Valor Medido	Error absoluto	Error Relativo
Media	0,46527	0,006252702	0,2314	26,8028 %
Varianza	0,171973738	1,37E-05	0,0638	569,21 %
Desviación estándar	0,414697164	0,003701746	0,2526	23,8581%

En la figura 4-13 se proporcionan los valores de error absoluto de la comparación entre las medidas de simulación (Ver ecuación 1.1) y lógica difusa. Con esto se determina que los datos son aproximados, el mínimo error absoluto es de 0,004 y el máximo error absoluto es de 0,7177, con un promedio de error absoluto del 0,23. En la figura 4-14, el porcentaje de error mínimo es del 0,3 % y el porcentaje de error máximo es del 73 %. Donde hay más porcentaje de error es en los valores altos del consumo de energía; al final de las simulaciones se puede observar que el error es acumulativo, incremental en forma exponencial, toma valores altos en las medidas donde el nivel de energía es alto, esto es debido a que en la ecuación 1.1 entre más número de saltos transcurran tiende a un nivel elevado de energía.

Figura 4-13: Error absoluto de la comparación ecuación 1.1 y lógica difusa.

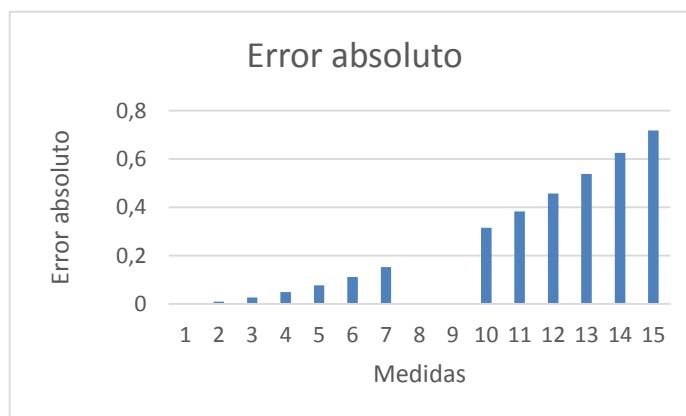
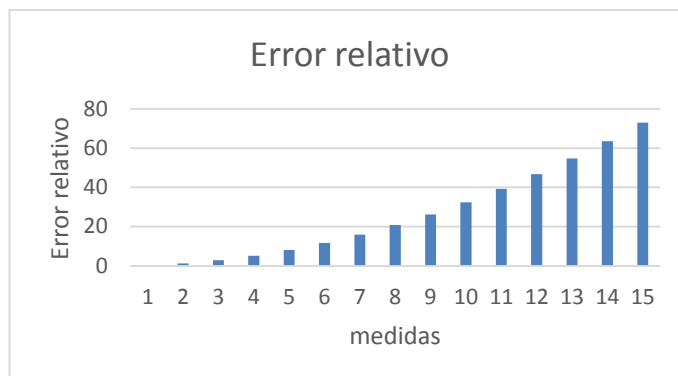


Figura 4-14: Error relativo de la comparación de datos ecuación 1.1 y lógica difusa.

Finalmente se puede concluir que los datos intermedios son muy aproximados (Ver figuras 4-11 y 4-12). El máximo alto llega a un límite de 14 mili-julios dando un cierto nivel constante de máximo en el universo de la función de membrecía consumo de energía, valor dado en figuras 4.9 y 4.10. El sistema, a partir de 10 mili-julios, es tomado como un consumo extremadamente alto fuera del alcance de la medida del consumo de energía por paquetes con 8 mili-segundos de duración. Por esta razón se debe definir en este modelo de energía, utilizando la metodología de la lógica difusa, en un escenario donde la MANET, con número máximo de paquetes de 10.000 Bytes, un número máximo de saltos entre nodos de 16 y un consumo máximo de energía de 10 mili-julios.

En la simulación del consumo de energía por el método de lógica empieza con un consumo inicial de 2 mili-julios, lo cual no fue tenido en cuenta en las ecuación 1.13, pero que sí es determinante en el cálculo de la ecuación 1.19 como la energía inicial. Esta se tomó en el modelo de energía con el método de lógica difusa, por lo que se determinó como error en la medida de energía en la figura 4.12.

Para no generar un error en la medida de la energía, se debe definir el nivel máximo de energía. La lógica difusa es una herramienta muy útil si se pueden definir muy bien sus reglas y funciones de membrecía, de lo contrario induciría a no determinar lo que realmente se busca. Si se utiliza correctamente la lógica difusa permite diagnosticar la energía en una MANET.

El error en la medida y en los valores estadísticos de la energía son de suma importancia para la retroalimentación por medio de un sistema de control en lógica difusa o controlador PID difuso en [45, 49]. El módulo propuesto de control de energía consumida por medio de MAC en [4-8], en la gestión de los estados o periodos de *sleep* y *listen overharing*, los nodos se ponen en estado *sleep* cuando no están en funcionamiento, el retardo promedio del tiempo *sleep* en el nodo receptor son las ecuaciones 1.32 y 1.33

que dependen a su vez de las ecuaciones 1.13 y 1.18 que determinan la longitud del paquete y duración en el tiempo.

Detectando el tiempo de *listen* y *sleep* en el MAC y asumiendo que un paquete llega al remitente con igual probabilidad en el tiempo dentro de una trama de datos, donde *overhearing* es el estado inactivo del nodo en espera para recibir el tráfico que no se envía, el estado *listen* y *sleep*, es decir, estado de reposo, el porcentaje de ahorro de energía en MAC está determinado en la ecuación 1.34, siendo el consumo en el transmisor la ecuación 1.30 y en el receptor la ecuación 1.31

5. Conclusiones y recomendaciones

5.1 Conclusiones

En el sistema propuesto se monitoreó el consumo de energía en una red inalámbrica ad hoc utilizando criterios de número de saltos y de longitud de paquete y a través de la técnica de inteligencia computacional llamada lógica difusa, con la que se encontró a partir de las simulaciones, una adecuada aproximación con los modelos existentes.

La lógica difusa determina la métrica del consumo de energía y adicionalmente la simula para monitorizar una MANET. Para este sistema inteligente de monitoreo es un lenguaje muy intuitivo que facilita su aplicación, admite el desarrollo de algoritmos lógicos fácilmente comprensibles, que permiten que sea gestionado con pocos conocimientos de programación que intuyan el uso de la herramienta FUZZY TOOLS DE MATLAB.

El sistema difuso es más automático, su fácil uso y tiempo de procesamiento permite al sistema de monitoreo inteligente tomar decisiones de forma mecánica en cuanto al diagnóstico de la energía en una MANET. Al sistema se le puede implementar un módulo para aplicar el control de fallos de acuerdo con las necesidades de energía de los equipos de la MANET.

Los modelos actuales del consumo de energía se han aplicado a la simulación de redes de datos, en el campo estadístico de procesos aleatorios algunos son solo teóricos y otros aplicados a métricas diferentes al consumo de energía como el enrutamiento; mientras que el modelo de consumo de energía a través de la lógica difusa no solo determina la métrica de la energía, sino que también permite tener un diagnóstico en cuanto a la eficiencia energética.

El sistema de lógica difusa es escalable, da la posibilidad de adicionar, en el futuro, nuevos módulos que relacionen varias métricas relacionadas con el consumo de energía tales como la eficiencia energética en la red y en cada uno de los equipos, monitorizar el tráfico de paquetes, la localización o distancia entre equipos, el enrutamiento en ruta eficiente a través de un consumo eficaz, la carga de batería en dispositivos y sensores remotos, la cantidad de equipos, el número de saltos en el enrutamiento y el estado de los nodos de la red.

El sistema puede interactuar con diferentes tipos de software, inclusive con diferentes lenguajes. Se puede interconectar con un software como MATLAB y de radio enlace

móvil, con el diseño de una aplicación web con interfaz en lenguajes HTML, MATLAB, gnplot, mapas digitalizados y de geo posicionamiento geográfico, con otros simuladores o con el mismo NS3 con la colaboración de software de radio enlace. Puede recrear escenarios con diferentes protocolos de propagación en escenarios rurales, urbanos, montañosos o con mucha vegetación.

Se han identificado las siguientes fuentes principales del consumo de energía, el primero es la colisión: cuando un paquete de transmisión circula por la red sin llegar a su destino tiene que ser desechado, ya que el seguimiento de las retransmisiones aumenta el consumo de energía. La colisión aumenta también el retardo. La segunda fuente es el *overheading*, es decir cuando un nodo recoge los paquetes que están destinados a otros nodos. La tercera fuente son los gastos de control de paquetes, el envío y recepción de paquetes de control consume demasiada energía y se pueden transmitir menos paquetes de datos útiles. La última fuente importante de ineficiencia está en el *overheading*, es decir cuando se está esperando para recibir el tráfico que no se envía. Esto es especialmente cierto en muchas aplicaciones de redes inalámbricas.

5.2 Recomendaciones

Se puede aplicar la variable consumo de energía como métrica del enrutamiento; ya que permite determinar a qué número de nodos y distancia entre equipos se dará la comunicación en la llegada de información de los paquetes de datos a su destino, siendo la ruta óptima de enrutamiento el camino más corto donde hay un consumo de energía bajo.

Para futuras simulaciones se podría estudiar el tráfico auto similar o fractal en las redes inalámbricas de sensores o ad hoc, ya que las simulaciones en varios escenarios mostraron que los resultados de los datos en tamaño de paquetes son proporcionales al consumo de energía, mientras que los saltos siguen un incremento exponencial en diferentes unidades de cantidad y de tiempo, tomadas con diferentes escalas y utilizando variables aleatorias con otros modelos estadísticos.

A. Anexo: código en MATLAB de la Interfaz de lógica difusa, para el modelo de energía en una MANET

```

function varargout = interfaz(varargin)
% INTERFAZ MATLAB code for interfaz.fig
%   INTERFAZ, by itself, creates a new INTERFAZ or raises the
existing
%   singleton*.
%
%   H = INTERFAZ returns the handle to a new INTERFAZ or the handle
to
%   the existing singleton*.
%
%   INTERFAZ('CALLBACK',hObject,eventData,handles,...) calls the
local
%   function named CALLBACK in INTERFAZ.M with the given input
arguments.
%
%   INTERFAZ('Property','Value',...) creates a new INTERFAZ or raises
the
%   existing singleton*. Starting from the left, property value
pairs are
%   applied to the GUI before interfaz_OpeningFcn gets called. An
%   unrecognized property name or invalid value makes property
application
%   stop. All inputs are passed to interfaz_OpeningFcn via varargin.
%
%   *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only
one
%   instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help interfaz

% Last Modified by GUIDE v2.5 10-Jun-2013 16:15:06

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @interfaz_OpeningFcn, ...
                  'gui_OutputFcn',  @interfaz_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})

```

```

        gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before interfaz is made visible.
function interfaz_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to interfaz (see VARARGIN)

% Choose default command line output for interfaz
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes interfaz wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = interfaz_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
global int1
global int2
global int3
global int4
global force2
phi=pi/2;
alpha=4;
p=1.2;
v=1000000;
load datos
load paquetes

```

```

phi=pi/2;
alpha=4;
p=1.2;
v=1000000;
for n=1:19
    resultado(n)=(n*n)*(2/phi)^(alpha/2)*gamma(1+(alpha/2));
    tamp(n)=(resultado(n)*v)/p;
    salto(n)=n;
end
hold(handles.axes1)
grid(handles.axes1)
plot(handles.axes1,resultado*10^(-5));
c=mean(datos(:, :));
plot(handles.axes1,paquetes(:,2),c,'g');
resultado2=[paquetes(:,2),c];
xlabel(handles.axes1,'Numero de Saltos')
ylabel(handles.axes1,'Consumo de Energía (Julios)')
title(handles.axes1,'ECUACION DE CONSUMO')
legend(handles.axes1,'Ecuación 1.1','Lógica difusa');

hold(handles.axes2)
grid(handles.axes2)
plot(handles.axes2,resultado*10^(-5));
c=mean(datos(:, :));
scatter(handles.axes2,paquetes(:,1),c,'g');
resultado2=[paquetes(:,1),c];
xlabel(handles.axes2,'Tiempo del paquete (mili-segundos)')
ylabel(handles.axes2,'Energía Consumida mJ')
title(handles.axes2,'ECUACION DE CONSUMO')

rema = load('Vreman.txt');
tamp = load('Vtam.txt');

for i=1:19
    inputs(i,1) = rema(i);
    inputs(i,2) = tamp(i);
end

ener = readfis('prueba1.fis');
econsu = evalfis(inputs, ener);

res_sof = load('consumo.txt');

t=[1 1.23 2 2.23 3 3.23 4 4.23 5 5.23 6 6.23 7 7.23 8 8.23 9 9.23 10];

scatter(handles.axes2,t*1000,res_sof,'b')
legend(handles.axes2,'Ecuación 1.13','Lógica difusa');

%er=abs(resultado(15)-resultado2(15))./resultado(15)
eal=abs(resultado(1)*10^(-3)-c(1));

```

```

ea2=abs(resultado(2)*10^(-3)-c(2));
ea3=abs(resultado(3)*10^(-3)-c(3));
ea4=abs(resultado(4)*10^(-3)-c(4));
ea5=abs(resultado(5)*10^(-3)-c(5));
ea6=abs(resultado(6)*10^(-3)-c(6));
ea7=abs(resultado(7)*10^(-3)-c(7));
ea8=abs(resultado(1)*10^(-3)-c(8));
ea9=abs(resultado(1)*10^(-3)-c(9));
ea10=abs(resultado(10)*10^(-3)-c(10));
ea11=abs(resultado(11)*10^(-3)-c(11));
ea12=abs(resultado(12)*10^(-3)-c(12));
ea13=abs(resultado(13)*10^(-3)-c(13));
ea14=abs(resultado(14)*10^(-3)-c(14));
ea15=abs(resultado(15)*10^(-3)-c(15));

media=[ea1,ea2,ea3,ea4,ea5,ea6,ea7,ea8,ea9,ea10,ea11,ea12,ea13,ea14,ea15
]
media_EA=mean(media)
desv_EA=std(media)
varianza_EA=var(media)
covarianza_EA=cov(media)
mediana_EA=median(media)

er1=(resultado(1)*10^(-3)- c(1)/resultado(1)*10^(-3))*100;
er2=(resultado(2)*10^(-3)- c(2)/resultado(2)*10^(-3))*100;
er3=(resultado(3)*10^(-3)- c(3)/resultado(3)*10^(-3))*100;
er4=(resultado(4)*10^(-3)- c(4)/resultado(4)*10^(-3))*100;
er5=(resultado(5)*10^(-3)- c(5)/resultado(5)*10^(-3))*100;
er6=(resultado(6)*10^(-3)- c(6)/resultado(6)*10^(-3))*100;
er7=(resultado(7)*10^(-3)- c(7)/resultado(7)*10^(-3))*100;
er8=(resultado(8)*10^(-3)- c(8)/resultado(8)*10^(-3))*100;
er9=(resultado(9)*10^(-3)- c(9)/resultado(9)*10^(-3))*100;
er10=(resultado(10)*10^(-3)- c(10)/resultado(10)*10^(-3))*100;
er11=(resultado(11)*10^(-3)- c(11)/resultado(11)*10^(-3))*100;
er12=(resultado(12)*10^(-3)- c(12)/resultado(12)*10^(-3))*100;
er13=(resultado(13)*10^(-3)- c(13)/resultado(13)*10^(-3))*100;
er14=(resultado(14)*10^(-3)- c(14)/resultado(14)*10^(-3))*100;
er15=(resultado(15)*10^(-3)- c(15)/resultado(15)*10^(-3))*100;

mediar=[er1,er2,er3,er4,er5,er6,er7,er8,er9,er10,er11,er12,er13,er14,er1
5]
media_ER=mean(mediar)
desv_ER=std(mediar)
varianza_ER=var(mediar)
covarianza_ER=cov(mediar)
mediana_ER=median(mediar)

a=readfis('prueba1.fis');
a.input(1).name='Numero de Saltos';
a.input(2).name='Longitud de Paquetes';
a.output(1).name='Energia Consumida';
gensurf(a)
title(handles.axes3, 'CONSUMO LOGICA DIFUSA')
force2=evalfis([int1 int2; int3 int4],a);

```

```

set(handles.text1,'string',force2(1));
set(handles.text2,'string',force2(2));

% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

function edit3_Callback(hObject, eventdata, handles)
global int3
int3=get(hObject,'String'); %Almacenar valor ingresado
int3 = str2double(int3); %Transformar a formato double
handles.edit2=int3; %Almacenar en puntero
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit3 as text
%        str2double(get(hObject,'String')) returns contents of edit3 as
a double

% --- Executes during object creation, after setting all properties.
function edit3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)
global int4
int4=get(hObject,'String'); %Almacenar valor ingresado
int4 = str2double(int4); %Transformar a formato double
handles.edit2=int4; %Almacenar en puntero
% hObject    handle to edit4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit4 as text
%        str2double(get(hObject,'String')) returns contents of edit4 as
a double

```

```

% --- Executes during object creation, after setting all properties.
function edit4_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit1_Callback(hObject, eventdata, handles)
global int1
int1=get(hObject,'String'); %Almacenar valor ingresado
int1 = str2double(int1); %Transformar a formato double
handles.edit2=int1; %Almacenar en puntero
guidata(hObject,handles); %Salvar datos de la aplicación
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
%         str2double(get(hObject,'String')) returns contents of edit1 as
a double

% --- Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit2_Callback(hObject, eventdata, handles)
global int2
int2=get(hObject,'String'); %Almacenar valor ingresado

```

```
int2 = str2double(int2); %Transformar a formato double
handles.edit2=int2; %Almacenar en puntero
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit2 as text
%        str2double(get(hObject,'String')) returns contents of edit2 as
a double

% --- Executes during object creation, after setting all properties.
function edit2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
clc
close all
clear all
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
nodo
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

B. Anexo: Código del menú principal del programa análisis estadísticos, ecuaciones y graficas del consumo de energía en una interfaz en C#.

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace AnalisisEstadisticos
{
    public partial class Menu : Form
    {
        private int childFormNumber = 0;

        public Menu()
        {
            InitializeComponent();
        }

        private void ShowNewForm(object sender, EventArgs e)
        {
            Form childForm = new Form();
            childForm.MdiParent = this;
            childForm.Text = "Window " + childFormNumber++;
            childForm.Show();
        }

        private void OpenFile(object sender, EventArgs e)
        {
            OpenFileDialog openFileDialog = new OpenFileDialog();
            openFileDialog.InitialDirectory =
Environment.GetFolderPath(Environment.SpecialFolder.Personal);
            openFileDialog.Filter = "Text Files (*.txt)|*.txt|All Files (*.*)|*.*";
            if (openFileDialog.ShowDialog(this) == DialogResult.OK)
            {
                string FileName = openFileDialog.FileName;
            }
        }

        private void SaveAsToolStripMenuItem_Click(object sender, EventArgs e)
        {

```

```
        SaveFileDialog saveFileDialog = new SaveFileDialog();
        saveFileDialog.InitialDirectory =
Environment.GetFolderPath(Environment.SpecialFolder.Personal);
        saveFileDialog.Filter = "Text Files (*.txt)|*.txt|All Files (*.*)|*.*";
        if (saveFileDialog.ShowDialog(this) == DialogResult.OK)
        {
            string FileName = saveFileDialog.FileName;
        }
    }

    private void ExitToolsStripMenuItem_Click(object sender, EventArgs e)
    {
        this.Close();
    }

    private void CutToolStripMenuItem_Click(object sender, EventArgs e)
    {
    }

    private void CopyToolStripMenuItem_Click(object sender, EventArgs e)
    {
    }

    private void PasteToolStripMenuItem_Click(object sender, EventArgs e)
    {
    }

    private void ToolBarToolStripMenuItem_Click(object sender, EventArgs e)
    {
        toolStrip.Visible = toolBarToolStripMenuItem.Checked;
    }

    private void StatusBarToolStripMenuItem_Click(object sender, EventArgs e)
    {
        statusStrip.Visible = statusBarToolStripMenuItem.Checked;
    }

    private void CascadeToolStripMenuItem_Click(object sender, EventArgs e)
    {
        LayoutMdi(MdiLayout.Cascade);
    }

    private void TileVerticalToolStripMenuItem_Click(object sender, EventArgs e)
    {
        LayoutMdi(MdiLayout.TileVertical);
    }

    private void TileHorizontalToolStripMenuItem_Click(object sender, EventArgs e)
    {
        LayoutMdi(MdiLayout.TileHorizontal);
    }

    private void ArrangeIconsToolStripMenuItem_Click(object sender, EventArgs e)
    {
        LayoutMdi(MdiLayout.ArrangeIcons);
    }
}
```

```
}

private void CloseAllToolStripMenuItem_Click(object sender, EventArgs
{
    foreach (Form childForm in MdiChildren)
    {
        childForm.Close();
    }
}

private void eToolStripMenuItem_Click(object sender, EventArgs
{
    EcVsPaquetesT EcPFT = new EcVsPaquetesT();
    EcPFT.MdiParent = this;
    EcPFT.Show();
}

private void contentsToolStripMenuItem_Click(object sender, EventArgs
}

private void indexToolStripMenuItem_Click(object sender, EventArgs
{
    EcPaquetesFijos EcPF = new EcPaquetesFijos();
    EcPF.MdiParent = this;
    EcPF.Show();
}

private void searchToolStripMenuItem_Click(object sender, EventArgs
{
}

private void ecVsPaquetesToolStripMenuItem_Click(object sender, EventArgs
{
    EcPaquetesFijos EcPF = new EcPaquetesFijos();
    EcPF.MdiParent = this;
    EcPF.Show();
}

private void consumoEcTiemposToolStripMenuItem_Click(object sender,
EventArgs e)
{
    GraficoUnido GraficoU =new GraficoUnido();
    GraficoU.MdiParent = this;
    GraficoU.Show();
}

private void consumosVsNodosToolStripMenuItem_Click(object sender,
EventArgs
{
```

```
        ConsumovsNodos ConsVsNodos = new ConsumovsNodos();
        ConsVsNodos.MdiParent = this;
        ConsVsNodos.Show();
    }

    private void consumoVsNodosTiemposToolStripMenuItem_Click(object sender,
EventArgs
    {
        ConsumovsNodosT ConsumoNodos = new ConsumovsNodosT();
        ConsumoNodos.MdiParent = this;
        ConsumoNodos.Show();
    }

    private void viewMenu_Click(object sender, EventArgs
    {

        //EcV ConsumoNodos = new Form1();
        // ConsumoNodos.MdiParent = this;
        // ConsumoNodos.Show();

    }

    private void pruebaToolStripMenuItem_Click(object sender, EventArgs e)
    {
        EcVsDuracion ConsumoNodos = new EcVsDuracion();
        ConsumoNodos.MdiParent = this;
        ConsumoNodos.Show();
    }

    }
}
.
```

6. Bibliografía

- [1] M. G. Piattini Velthuis y E. del Peso Navarro, Auditoría Informática: Un enfoque práctico, Madrid: RA-MA, 2001.
- [2] Y. Wei, J. Heidemann y D. Estrin, «An Energy-Efficient MAC Protocol for Wireless Sensor Networks,» *Proc. INFOCOM. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, pp. 1567-1576, 2002.
- [3] A. I. Jing , j. Kong y D. Turgut, «An adaptive coordinated medium access control for wireless sensor networks,» *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium* , vol. 1, pp. 214-219, 28-1 june-july 2004.
- [4] J. M. Hernández, E. Inzunza, J. E. Olgin, J. J. Soares, P. A. Rousseu y C. Camargo, «802.15.4 MAC protocol applied to efficient energy consumption,» *Revista INGENIERÍA E INVESTIGACIÓN*, vol. 30, nº 1, pp. 71-74, 2010.
- [5] S. Siddiqui y S. Ghani, «ES-MAC: Energy Efficient Sensor-MAC protocol for Wireless Sensor networks,» *Networking, Sensing and Control (ICNSC), 10th IEEE International Conference*, pp. 28-33, 10-12 April 2013.
- [6] S. Du, A. K. Saha y D. B. Johnson, «RMAC: A routing-enhanced dutycycle MAC protocol for wireless sensor networks,» *In Proc. INFOCOM*, pp. 1478-1486, 2007.
- [7] W. Heping , Z. Xiaobo , N.-A. Farid y K. Ashfaq , «Cross-Layer Optimized MAC to Support Multihop QoS Routing for Wireless Sensor Networks,» *IEEE Transaction on*

- Vehicular Technology*, vol. 59, nº 5, pp. 2556-2563, June 2010.
- [8] V. Karthikeyan, A. Vinod y P. Jeyakumar, «An Energy Efficient Neighbour Node Discovery Method for Wireless Sensor Networks,» *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, nº 6, pp. 1395-1399, 2013.
- [9] I. S. 802.11, «IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements Part 11: wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications,» *Amendment to IEEE Std 802.11, 1999 Edition Reaff 2003*, pp. 1-189, 2005.
- [10] M. Villapol, «Casos de Estudio Redes Ad Hoc y Mesh,» *Comisión de Telecomunicaciones Republica Bolivariana de Venezuela. Universidad de Caracas*, 2009.
- [11] «B. Liang and Z. Haas, "Predictive Distance-based Mobility Management for PCS Networks," , Vol. 3, New York, NY, USA, pp. 1377-1384, March 1999.,» *Proceedings of the IEEE INFOCOM 1999*, vol. 3, pp. 13377-1384, March 1999.
- [12] V. Davies, «Evaluating Mobility Models with Ad Hoc Network," Master's Thesis, Colorado School of Mines,» 2000.
- [13] D. Johnson y D. Maltz , «Dynamic Source Routing in Ad Hoc Wireless Network,» *Mobile Computing*, pp. 153-181, 1996.
- [14] X. Hong, M. Gerla, G. Pei y C. Chiang, «A Group Mobility Model for Ad Hoc Wireless Networks,» *Proceedings of the 2nd ACM International*.
- [15] M. A. Alzate, «Vehicular ad hoc network (VANET),» *IEEE CONFERENCIA SISTEMAS DE TRANSPORTE INTELIGENTE*, agosto 2012.
- [16] J. E. Ortiz Triviño y Hernandez, «Calculo de algunas medidas estadísticas para evaluar el desempeño de redes ad hoc,» *Ingeniería y Competividad*, vol. 8, 2006.
- [17] M. Haenggi, «On Routing in Random Rayleigh Fading Networks,» *IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS*, vol. 4, nº 4, July 2005.
- [18] A. Ephremides , «Energy Concerns in Wireless Networks,» *IEEE Wireless Communications*, vol. 9, pp. 48-59, August 2002.
- [19] E. S. Sousa y J. A. Silvester, «Optimum transmission ranges in a direct-sequence

- spread-spectrum multihop packet radio network,» *IEEE J. Sel.*
- [20] R. L. Graham, D. E. Knuth y O. Patashnik, *Concrete Mathematics: A Foundation for Computer Science*, 2nd ed. Reading, MA: Addison-Wesley, 1994.
- [21] P. Santi , *Topology Control in Wireless Ad Hoc and Sensor Networks*. Istituto di Informatica e Telematica del CNR, Italy: John Wiley & Sons, 2005.
- [22] L. Kirousis , E. Kranakis , D. Krizanc y Pelc , «Power consumption in packet radio networks,» *Theoretical Computer Science* 243, pp. 289-305, 2000.
- [23] C. Perkins y P. Bhagwat, «Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers,» *Computer Communications Review*, pp. 234-244, october 1994.
- [24] «C. Perkins and E. Royer, "Ad-hoc On-Demand Distance Vector Routing,» *Proceeding of the 2 nd IEEE Workshop, Mobile Comp, Sys and Apps*, pp. 90-100, February 1999.
- [25] J. C. Cano y A. Manzoni, «A clustering algorithm to provide interoperability to local area wireless networks. Department of Computer Engineering, Polytechnic University of Valencia,» *Proceedings of the 27th Annual IEEE Conference on Local Computer Networks*, pp. 742-1303, 2002.
- [26] D. K. Kim, J. J. Garcia, L. Aceves, K. Obraczka, J. C. Cano y P. Manzoni , «Power-Aware Routing Based on The Energy Drain Rate for Mobile Ad Hoc Networks,» *IEEE ICCCN 2002*, October 2002.
- [27] S. L. Wu , Y. C. Tseng y J. P. Sheu , «Intelligent Medium Access for Mobile Ad Hoc Networks with Busy Tones and Power Control,» *IEEE Journal On Selected Areas In Communications*, vol. 18, n° 9, September 2000.
- [28] K. Scott y N. Bambos , «Routing and channel assignment for low power transmission in PCS,» *Proceedings of IEEE International Conference on Universal Personal Communications (ICUPC '96)*, 29-2 September-October 1996.
- [29] S. Singh, M. Woo y C. S. Raghavendra ", «Power-aware with Routing in Mobile Ad Hoc Networks,» *Proceedings of Mobicom*, 1998.
- [30] C. K. Toh, «Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks,» *IEEE Communications Magazine*, June 2001.
- [31] T. S. Rappaport , *Wireless Communications: Principles and Practice*, Prentice Hall,

1999.

- [32] Ergen, M; Varaiya., P;, «Decomposition of Energy Consumption in IEEE 802.11,» de *Communications, 2007. ICC '07. IEEE International Conference on*, Glasgow., 2007.
- [33] «J. C. Cano, P. Manzoni. Evaluating the energy-consumption reduction in Manet a MANET by dynamically swithcing-off network interface,» *Computers and Communications. Proceedings. Sixth IEEE Symposium on* , pp. 186-191 , 2001.
- [34] L. Feeney y M. Nilsson, «Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment,» *Proceedings of IEEE INFOCOM, Anchorage, AK, 2001.*,» *Proceedings of IEEE INFOCOM, Anchorage, AK,*, 2001.
- [35] W. El-Hajj, D. Kountanis, A. Al-Fuqaha y M. Guizani, «A Fuzzy-Based Hierarchical Energy Efficient. Routing Protocol for Large Scale Mobile AdHoc Networks (FEER).,» *IEEE ICC, 2006.*
- [36] H. Wu, S. Nabar y R. Poovendran., «An Energy Framework for the Network Simulator 3 (ns-3),» Seattle. USA.
- [37] C. Zhao, B. Ge y X. Tian, «A generic simulation framework for energy consumption in data center networks,» *IEEE ICC 2013 - Ad-hoc and Sensor Networking Symposium*, pp. 1678 - 1682, 2013.
- [38] O. Younis y S. Fahmy, «Energy-Efficient Routing and Data Aggregation in Sensor Networks: An Experimental Study,» CSD TR-04-031, Purdue University,» 2004.
- [39] G.-R. d. c. n. 31, «Guime.net,» [En línea]. Available: http://guimi.net/monograficos/G-Redes_de_comunicaciones/G-RCnode31.html..
- [40] D. Remondo, «Tutorial on Wireless Ad Hoc Networks,» *n Proc. of the 2 nd International Working Conf. on Performance Modeling and Evaluation of Heterogeneous Networks HET-NETs '04*),, 26-28 July 2004.
- [41] J. Arndt, S. B. Behlert, F. Bodammer y J. Branam , «Suxe Linux,» Novell, 2005. [En línea]. Available: www.novell.com/es-es/documentation/suse93/.../admin93-screen.pdf.
- [42] O. Younis y S. Fahmy, «HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad-hoc Sensor Networks,» *IEEE Transactions on Mobile Computing*, vol. 4, nº 4, pp. 366-379, 2004.

- [43] V. Altroc, *Fuzzy Logic and Neuro-Fuzzy Applications Explained*, N J, Englewood Cliffs, NJ, 1995.: Prentice Hall, 1995.
- [44] C. V. Altrock, «Lógica Difusa dentro de la Ingeniería Automotriz, Publicado en Circuit Cellar INK, publicación 88, Noviembre 1997.,» *Publicado en Circuit Cellar INK*, nº 88, Noviembre 1997.
- [45] L. S. Coelho y A. R. Coelho, *Fuzzy PID controllers: structures, design principles and application for nonlinear practical process*, *Advances in soft computing - engineering design and manufacturing*, R. F. T. e. P. K. C. Roy, Ed., London: Springer-Verlag, 1999..
- [46] M. Gadeo, «Estrategias evolutivas para la adquisición de conocimiento en controladores difusos temporales difuminados, aplicadas al enrutamiento adaptativo en redes de comunicaciones,» Alcalá, 2009.
- [47] L. A. Zadeh, «Fuzzy sets,» *Information and Control*, vol. 8, 1965.
- [48] L. A. Zadeh, «Why the success of fuzzy logic is not paradoxical,» *IEEE Expert*, vol. 9, nº 4, 1994.
- [49] L. S. Coelho, A. R. Coelho y O. M. Almeida, «Proyecto y Estudio de un Caso de Implementación de un Sistema de Control Difuso,» *Revista Control y Automatización*, vol. 14, nº 1, 2003.
- [50] E. Mamdani y S. Assilian, «An experiment in linguistic synthesis with a fuzzy logic controller,» *International Journal of Man Machine Studies*, vol. 7, nº 1, 1975.
- [51] L. S. Cohelo, «Identificación y control de procesos multivariables vía metodologías de avanzada inteligencia computacional,» Santa Catarina, 2000.
- [52] J. H. Holland, «Adaptation in Natural and Artificial Systems,» The MIT Press, London, 1992.
- [53] Goldberg, D.E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, New York: Addison Wesley, 1989..
- [54] L. a. C. f. B. T. (CBIT), «SNU CSE Biointelligence,» 2002.
- [55] Y.-H. Song y A. T. Johns , «Application of fuzzy logic in power systems Part 2 Comparison And integration with expert systems, neural networks and genetic algorithms,» *POWER ENGINEERING JOURNAL*, August 1998.

- [56] P. Azpiroz De Pedro, «Integración de Radio Mobile y NS-3 para la planificación de redes rurales híbridas WiMAX+WiFi,» Universidad Carlos III de Madrid. Escuela Politécnica Superior, Departamento de Ingeniería Telemática Ingeniería de Telecomunicación, Madrid, 2010.
- [57] K. Fall y K. Varadhan, «Ns Notes and Documentation,» 2005.
- [58] A. Triviño, E. Casilari y A. Ariza , «IMPLEMENTACIÓN DE PROTOCOLOS EN EL NETWORK SIMULATOR (NS2).,» Málaga.
- [59] O. NS3, «network simulator and animator version 3,» 11 2013. [En línea]. Available: <http://www.nsnam.org/>.
- [60] D. Cavin, Y. Sasson y A. Schiper , «On the Accuracy of MANET simulators,» *ACM Workshop on Principles of Mobile Computing (POMC 02)*, pp. 38-43, October 2002.
- [61] J. Broch, D. B. Maltz, D. B. Johnson, y. Hu y J. Jetcheva, «A performance comparison of multi-hop wireless ad hoc network routing protocols,» de *In Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, 1998.
- [62] S. Nabar , «Código fuente del modelo de energia en NS3 en C++ bajo Licencia de software libre GNU,» Laboratorio de seguridad de la Universidad de Washington, 2010. [En línea]. Available: <http://www.nsnam.org/>.
- [63] L. Bajaj, M. Takai, R. Ahuja, K. Tang y M. Gerla, «GloMoSim: A Scalable Network Simulation Environment,» Los Angeles, 1999.
- [64] M. Cloran., «Simulation of ieee 802.11 pcf function in glomosim.,» DUBLIN CITY UNIVERSITY SCHOOL OF ELECTRONIC ENGINEERING. BEng in Telecommunications. Supervised by Dr. J. Murphy and DR. Sean Murphy., April 2004. [En línea]. Available: <http://www.eeng.dcu.ie/~murphys/projects/MichaelCloran-FinalReport.pdf>. [Último acceso: 7 Marzo 2013].
- [65] J. Farooq, «Implementation and evaluation of ieee 802.11e wireless lan in glomosim.,» [En línea]. Available: <http://www.jahanzeb.com/download/masterthesissummary.pdf>.
- [66] M. Carlberg y A. Dammander, «Wimax - a study of mobility and a mac-layer implementation in glomosim.,» [En línea]. Available: <http://www.cs.umu.se/education/examina/Rapporter/CalbergDammander.pdf>.

- [67] G. Tech, «Gtnets,» [En línea]. Available: [http://www.ece.gatech.edu/research/labs/MANIACS/GTNetS/..](http://www.ece.gatech.edu/research/labs/MANIACS/GTNetS/)
- [68] C. University., «Jist/swans.,» [En línea]. Available: [http://jist.ece.cornell.edu/..](http://jist.ece.cornell.edu/)
- [69] NCTU. , «Nctu network simulator,» [En línea]. Available: [http://nsl10.csie.nctu.edu.tw/.](http://nsl10.csie.nctu.edu.tw/)
- [70] OMNeT++ Org , «OMNet++.,» [En línea]. Available: [http://www.omnetpp.org/..](http://www.omnetpp.org/)
- [71] OMNet++, «IEEE 802.11 INET Patch.,» [En línea]. Available: [http://upload.ttnz.cz/readfile.php?file=horal/802.11e/ieee802.11-INETpatch-1.0.tar.bz2. .](http://upload.ttnz.cz/readfile.php?file=horal/802.11e/ieee802.11-INETpatch-1.0.tar.bz2.)
- [72] Numbat., «New Ubiquitous Mobility Basic Analysis Tools,» [En línea]. Available: <http://klub.com.pl/projects/numbat..>
- [73] A. Laya, «Multi-Radio Cooperative Strategy for Reliable Multicast,» March 2012. [En línea]. Available: <http://wikienergy.cttc.es/images/2/2d/Ns-3-tutorial-complete.pdf>.
- [74] K. Wang, «NS-3 Energy Model An Introduction,» March 2012. [En línea]. Available: <http://wikienergy.cttc.es/images/2/2d/Ns-3-tutorial-complete.pdf>.
- [75] J. Postel, «Request for Comments: 879. The TCP Maximum Segment Size and Related Topics. Network Working Group,» 1983.