



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA
INGENIERÍA ELÉCTRICA - TELECOMUNICACIONES

**MODELADO DEL COMPORTAMIENTO
DINÁMICO DE UNA RED IOT CON EL
PROTOCOLO DE LORA**

TESIS
QUE PARA OPTAR POR EL GRADO DE:
Maestro en Ingeniería Eléctrica
(Telecomunicaciones)

PRESENTA:
Luis Miguno Haro

COMITÉ TUTORIAL
Dr. Víctor Rangel Licea
FACULTAD DE INGENIERÍA, UNAM
Dr. José Jaime Camacho Escoto
FACULTAD DE INGENIERÍA, UNAM

CIUDAD UNIVERSITARIA CDMX, AGOSTO 2021



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

JURADO ASIGNADO:

Presidente: Dr. Sergiy Khotyaintsev
Secretario: Dr. Luis Francisco García Jiménez
Vocal: Dr. Víctor Rangel Licea
1er. Suplente: Dr. Ramón Gutiérrez Castrejón
2o. Suplente: Dr. José Jaime Camacho Escoto

La tesis se realizó en la Facultad de Ingeniería, UNAM.

TUTOR DE TESIS:

Dr. Víctor Rangel Licea

CO-TUTOR DE TESIS:

Dr. José Jaime Camacho Escoto

*Este trabajo está dedicado
a mi mamá, **Alicia Haro Sánchez**
y mi papá, **Lino Moreno Flores**
por su amor, educación y protección
que me han forjado como la persona que soy.*

Reconocimientos

A la Universidad Nacional Autónoma de México Por darme la oportunidad de ser parte de su comunidad y de esta manera haberme brindado una educación de calidad. Por los conocimientos y valores que me otorgó. Además del acceso a incontables fuentes de información científica.

Al CONACYT Organismo donde me encuentro con el CVU 1006140. Por el apoyo durante mis estudios de maestría mediante la beca que me fue otorgada.

Al proyecto EWIN Por mi participación en el proyecto EWIN con referencia EP/P029221/1-1-J15413/14/15. Este proyecto cuenta con número de registro UNAM: 49547-2363-12-IX-17. Red de información del agua para predicción de inundación en tiempo

Al Dr. Victor Rangel Licea Director de la tesis, por la supervisión de cada acción realizada durante el desarrollo del módulo de la red que se ha construido. Además de haberme presentado el funcionamiento de cada una de las partes del escenario. Mostrando su excelente capacidad y conocimientos en esta área.

Al Dr. José Jaime Camacho Escoto Co-director de la tesis, por la asesoría en la construcción y el desarrollo del módulo de este trabajo. El Dr. Camacho compartió conmigo las excelentes habilidades y conocimientos con las que cuenta, en cuanto al desarrollo de módulos de redes en el simulador Network Simulator 3.

Declaración de autenticidad

Por la presente declaro que, salvo cuando se haga referencia específica al trabajo de otras personas, el contenido de esta tesis es original y no se ha presentado total o parcialmente para su consideración para cualquier otro título o grado en esta o cualquier otra Universidad. Esta tesis es resultado de mi propio trabajo y no incluye nada que sea el resultado de algún trabajo realizado en colaboración, salvo que se indique específicamente en el texto.

Luis Miguel Moreno Haro. Ciudad Universitaria CDMX, Agosto 2021

Resumen

En años recientes debido a los cambios climatológicos se han incrementado el número de inundaciones en muchas ciudades de México y el mundo. Por ejemplo, en el país se ha observado que las lluvias intensas tienden a ser cada vez más regulares y severas. Esto provoca pérdidas materiales y humanas ya que los sistemas de alerta de estos fenómenos en el país, resultan ser muy costosos y complejos al implementarse.

Mediante un convenio que existe entre la Universidad Nacional Autónoma de México, la universidad de Colima y la universidad de Sheffield se pretende desarrollar un proyecto económico y eficiente para prevenir inundaciones a través del uso de sensores y redes de telecomunicaciones.

La solución planteada en este trabajo de investigación se obtuvo por medio de simulaciones por computadora, ya que se realizó un estudio del comportamiento dinámico de una red utilizando el protocolo *LoRa* en ríos como escenario principal. Al conocer mejor este comportamiento de la red, podrá ser posible obtener una mejor administración de alertas ante fenómenos naturales como las lluvias e inundaciones. Para la realización de este estudio se desarrolló e implementó un módulo en un potente software de simulación denominado *Network Simulator 3*.

Para el estudio del comportamiento dinámico se calculó el *throughput* máximo (a nivel de capa *MAC*) que la red *LoRa* puede alcanzar bajo ciertos parámetros y características, utilizando un determinado modelo de propagación. Posteriormente, se analizan los resultados a través de la simulación de diferentes escenarios, en función del número de boyas y su modelo de movilidad. Con este estudio, es posible conocer teóricamente el *throughput* de la red, dependiendo de dónde se monte el escenario, teniendo en cuenta el número de boyas o *end devices* y *gateways*.

Además se realizó el estudio de la tasa de envío a través de las gráficas de *PDR* (*Packet Delivery Rate*) frente al número de *end devices* por simulación. También el estudio del retardo medio de los paquetes se muestra gráficamente.

Índice general

Índice de figuras	XIII
Índice de tablas	XV
1. Introducción	1
1.1. Antecedentes	1
1.2. Planteamiento del problema	1
1.3. Motivación	2
1.4. Hipótesis	3
1.5. Objetivo	3
1.5.1. Objetivo general	3
1.5.2. Objetivos específicos	3
1.6. Metodología	4
1.6.1. Etapa 1: Compresión del módulo LoRaWAN	4
1.6.2. Etapa 2: Eliminación de la capa MAC	4
1.6.3. Etapa 3: Validación del módulo LoRa	4
1.6.4. Etapa 4: Integración de módulos que faltaban	5
1.6.5. Etapa 5: Diseño e implementación de la capa MAC	5
1.6.6. Etapa 6: Pruebas y de resultados del módulo	5
1.7. Contribuciones	6
1.8. Organización de la tesis	6
2. Conceptos básicos y estado del arte	9
2.1. Tecnologías IoT	9
2.2. Evolución de redes IoT	11
2.2.1. Internet de las cosas	11
2.2.2. IoT a través del tiempo	12
2.2.3. Tecnologías IoT actuales	13
2.3. Aportación de la tesis	15
2.4. Trabajos relacionados	16

ÍNDICE GENERAL

2.4.1.	Poyecto EWIN	16
2.4.2.	Estudio del rendimiento LoRaWAN	17
2.4.3.	Monitorización de inundaciones	17
2.4.4.	Bit Error Rate (BER) de la tecnología LoRa	18
2.4.5.	Estudio de propagación de LoRa en ríos con vegetación	20
3.	Tecnología LoRa	21
3.1.	Bandas de frecuencia que utiliza LoRa	21
3.2.	Ventajas del uso de LoRa	22
3.3.	Capa física	23
3.3.1.	Chirp	24
3.3.2.	Modulación de espectro extendido LoRa chirp	25
3.3.3.	Impacto del factor de esparcimiento SF	25
3.3.4.	Parámetros de la capa física	27
3.3.5.	Duraciones y rangos en LoRa	28
3.3.6.	Rendimiento a diferentes anchos de banda	29
3.3.7.	Tiempo en el aire	33
3.3.8.	Sensibilidad	35
3.3.9.	Símbolo en LoRa	35
3.4.	Modelo de propagación del proyecto EWIN	36
4.	Descripción del modelo de simulación de LoRa en Network Simulator 3 (NS-3)	39
4.1.	Sistema de eventos discretos	39
4.1.1.	Características del simulador NS-3	40
4.2.	Diseño del módulo LoRa	41
4.2.1.	Modelo de la capa física	42
4.3.	Resultados: Time On Air	43
4.3.1.	Características del escenario para paquetes de 24 bytes	43
4.3.2.	Resultados teóricos y simulados para tamaño de carga útil de 24 bytes	44
4.3.3.	Resultados teóricos de tiempo en el aire para paquete de 24 bytes	44
4.3.4.	Características del escenario para paquetes de 5 bytes	45
4.3.5.	Resultados teóricos y simulados para tamaño de carga útil de 5 bytes	46
4.3.6.	Resultados teóricos de tiempo en el aire para paquete de 5 bytes	47
4.4.	Modelo de la capa MAC	48
4.4.1.	Capacidad de calcular la distancia entre una boya y un river core	48

4.4.2.	Planificación del envío de paquetes	49
4.4.3.	Etiquetado de mensajes transmitidos	51
4.4.4.	Acuse de los paquetes enviados	51
4.4.5.	Retransmisión de paquetes	53
4.4.6.	Encabezados MAC	54
4.5.	Características de las boyas	55
4.5.1.	Modelado del efecto de variación de velocidad durante el trayecto	55
4.5.2.	Modelado del efecto de inversión de las boyas durante el trayecto	58
4.5.3.	Modelado del efecto de estancamiento de las boyas	58
4.6.	Consumos Energéticos del end device y GPS en las boyas	59
4.6.1.	Batería utilizada en el sistema	60
4.6.2.	Energía residual	61
4.6.3.	Experimento de duración de la energía	61
5.	Comportamiento dinámico	65
5.1.	Definición de diferentes escenarios	65
5.1.1.	Datos geográficos	66
5.1.2.	Selección de path loss utilizado	67
5.1.3.	Consumo energético de las boyas	68
5.1.4.	Velocidad en ríos	68
5.1.5.	Tasa de generación de paquetes	69
5.1.6.	Definición de los parámetros comunes de la simulación	69
5.2.	Escenario A: Modelado de la red con una boya	70
5.2.1.	Resultados	70
5.3.	Escenario B: Simulación de la red con diferentes valores de velocidad	74
5.3.1.	Resultados	75
5.4.	Escenario C: Probabilidad de éxito de recepción de paquetes para diferentes tasas de generación	77
5.4.1.	Resultados	77
5.5.	Escenario D: Modelado de la red con valores crecientes de velocidad dependiendo de su ubicación geográfica	79
5.5.1.	Resultados	79
5.6.	Escenario E: Modelado de la red con 100 boyas con velocidad constante	82
5.6.1.	Resultados	84
5.7.	Escenario F: Modelado de la red con 100 boyas con velocidad variable	87
5.7.1.	Resultados	87
5.8.	Relación de entrega de paquetes (PDR)	88
5.9.	Retardo promedio de los paquetes	89

ÍNDICE GENERAL

6. Conclusiones	91
6.1. Trabajos futuros	91
6.2. Conclusiones	91
6.2.1. Conclusiones en los escenarios de simulación	92
6.2.2. Conclusión del consumo energético	93
6.2.3. Conclusiones en retardo y relación de entrega de paquetes	93
A. Anexos	95
A.1. Tablas data rate	95
A.1.1. Tabla ancho de banda de 125kHz	96
A.1.2. Tabla ancho de banda de 250kHz	97
A.1.3. Tabla ancho de banda de 500kHz.	98
Bibliografía	99
Siglas	105

Índice de figuras

2.1. Velocidad de transferencia de datos contra alcance de comunicación.	10
2.2. Evolución de <i>IoT</i> [1].	13
2.3. Protocolos de comunicación inalámbrica para IoT [2].	14
2.4. LoRa BER.	19
3.1. Comparación de LoRa con otras tecnologías [3].	23
3.2. Ilustración de Chirp [4].	24
3.3. Símbolos de LoRa [5].	25
3.4. Comparación de SF7 hasta SF12 utilizando 125kHz [6].	26
3.5. Comparación del alcance del factor de esparcimiento y del ancho de banda para obtención de diferentes valores de rendimiento.	30
3.6. Rendimiento del protocolo LoRa utilizando un canal de 125 kHz.	31
3.7. Rendimiento del protocolo LoRa utilizando un canal de 250 kHz.	32
3.8. Rendimiento del protocolo LoRa utilizando un canal de 500 kHz.	33
3.9. Estructura del frame de LoRa [7].	34
3.10. Punto de la ruta del río donde el path loss cambia.	37
4.1. Pila de protocolos de LoRa.	41
4.2. Tiempo en el aire de paquetes de transmisión de 24 bytes en LoRa.	45
4.3. Tiempo en el aire de paquetes de confirmación de 5 bytes en LoRa.	47
4.4. Rango de transmisión de la boya.	49
4.5. Rango de planificación de la boya.	50
4.6. Diagrama de flujo del envío de paquetes.	52
4.7. Acuse de los paquetes en el área de transmisión.	53
4.8. Encabezados MAC de los paquetes.	54
4.9. Áreas y probabilidades de hundimiento [8].	55
4.10. Nodo y distancias a lo largo del río [8].	57
4.11. Consumo energético total de una boya que envía y recibe un solo paquete.	62
5.1. Posición geográfica del río, de los River Core y del dispensador.	66

ÍNDICE DE FIGURAS

5.2. Rendimiento de la red usando una sola boya a velocidad de 1 metro sobre segundo.	71
5.3. Consumo energético total del escenario con una boya a velocidad de 1 metro sobre segundo.	72
5.4. Consumo energético por boya del escenario A.	73
5.5. Relación de paquetes generados, recibidos y retransmitidos en el escenario B.	75
5.6. Consumo energético total de la boya para diferentes velocidades de movilidad.	76
5.7. Probabilidad de éxito de paquetes a diferentes tasas de generación de datos.	78
5.8. Paquetes recibidos, retransmitidos y generados en el escenario D. .	80
5.9. Consumo energético del escenario D.	81
5.10. Consumo energético por boya en el escenario D.	82
5.11. Relación de paquetes del escenario E.	84
5.12. Comparación de la relación de paquetes del escenario E usando varias semillas y sin usar el algoritmo de retroceso exponencial. . .	85
5.13. Comparación de la relación de paquetes del escenario E usando varias semillas y el algoritmo de retroceso exponencial.	86
5.14. Comparación de la relación de paquetes del escenario F usando varias semillas.	88
5.15. Relación de entrega de paquetes (PDR).	89
5.16. Retardo promedio de los paquetes en la cola para una simulación de 10 boyas.	90

Índice de tablas

3.1.	Bandas de frecuencia LoRa en el planeta [9].	22
3.2.	Chips en LoRa [4].	27
3.3.	Rendimiento.	29
3.4.	Sensibilidad del receptor Semtech SX1276 LoRa en dBm a diferentes anchos de banda y factores de esparcimiento [10].	35
3.5.	Path loss lado derecho e izquierdo [8].	37
4.1.	Características del escenario de simulación.	43
4.2.	Resultados teóricos para tamaño de carga útil de 24 bytes.	44
4.3.	Características del escenario de simulación.	46
4.4.	Resultados teóricos para tamaño de carga útil de 5 bytes.	46
4.5.	Consumo de los estados ED y GPS, valor técnico de ED [11].	60
4.6.	Consumo teórico y simulado de los estados ED y GPS.	63
5.1.	Coordenadas geográficas de los river core.	67
5.2.	Path loss lado derecho.	67
5.3.	Valores de consumo del end device.	68
5.4.	Parámetros de configuración.	69
5.5.	Parámetros de configuración del escenario A.	70
5.6.	Resultados de consumo de energía del escenario A.	73
5.7.	Parámetros de configuración del escenario B.	74
5.8.	Parámetros de configuración del escenario C.	77
5.9.	Parámetros de configuración del escenario D.	79
5.10.	Resultados de consumo de energía del escenario D.	81
5.11.	Parámetros de configuración del escenario E.	83
5.12.	Probabilidad de estancamiento de las boyas del escenario E.	85
A.1.	Data rate ancho de banda 125kHz.	96
A.2.	Data rate ancho de banda 250kHz.	97
A.3.	Data rate ancho de banda 500kHz	98

Introducción

1.1. Antecedentes

Esta tesis muestra el trabajo del desarrollo, la implementación y los resultados de la simulación de un escenario, con determinadas peculiaridades, de una red con tecnología del *Internet of Things (IoT)* en telecomunicaciones que utiliza tecnología *Long Range (LoRa)* para su cometido. La principal característica de este escenario, es que está montado en su totalidad en ríos. Todo esto para conocer mejor el comportamiento de un proyecto que es financiado por instituciones internacionales. Dicho proyecto tiene como objetivo principal conocer el comportamiento de los ríos en determinadas condiciones climáticas. Todo con el fin de monitorización para salvaguarda de las poblaciones en comunidades vecinas.

A partir de estas condiciones, se profundizará en el uso y manejo de un potente simulador de eventos discretos para sistemas de redes en telecomunicaciones, el simulador *Network Simulator 3 (NS-3)*. Mismo que ha permitido la simulación del escenario antes mencionado, tomando en cuenta todas las características necesarias para la implementación del proyecto.

Estando seguros de que esta obra será base y referencia para el conocimiento y aprendizaje del protocolo *LoRa* en escenarios tan complejos como ríos.

1.2. Planteamiento del problema

Las inundaciones debido a fuertes lluvias han sido un problema muy común en varias partes de la República Mexicana, en el Reino Unido y el mundo, en especial para zonas y ciudades cercanas a ríos ya que el desbordamiento de estos puede representar un problema para los habitantes. Como consecuencia del cam-

bio climático se pronostica que este problema empeore con el paso del tiempo. Haciendo que ésto sea un desafío mundial para la ciencia.

Las inundaciones y su gravedad, aumentan a medida que le los cambios climáticos se hacen presentes en la actualidad. Siendo las poblaciones más vulnerables como aquellas de países de bajos ingresos quienes enfrenten mayormente estas consecuencias negativas, lamentablemente la sociedad mexicana está incluida, [12]. Estudios realizados por todo el planeta, muestran que entre 1900 y 2010, las inundaciones tienen un número más significativo y creciente en comparación con otros desastres, tanto en términos de eventos registrados como de personas afectadas, [13]. Las inundaciones en México son un gran desafío, ya que las inundaciones aumentan en regularidad y severidad. Lamentablemente, la gestión de desastres relacionados con el clima en México trabaja más en las acciones de reparación, asistencia de emergencia que en la prevención, [14].

Actualmente no existe un sistema que realice un monitoreo de ríos, porque además de ser muy complejo puede resultar tener un costo demasiado elevado dependiendo de la tecnología que se use. Un ejemplo que representa una de las posibles soluciones con altos costos en cuanto a la implementación es el uso de satélites. Esta resultaría una excelente solución a este problema, pero requiere un alto capital de inversión y mantenimiento.

Es por ello que mediante este trabajo se ha buscado obtener resultados igual de precisos y en tiempo real, de una manera mucho más económica. Así, a lo largo de esta tesis se muestra el desarrollo, implementación y simulación de un escenario específicamente para la prevención de inundaciones, causadas por lluvias y desbordes de ríos, empleando la tecnología *LoRa*.

1.3. Motivación

Este trabajo se deriva de un proyecto con impacto internacional que busca realizar un sistema de detección de inundaciones rentable, que se pueda implementar fácilmente en países en vías de desarrollo: red de información sobre el agua de emergencia o Emergency Water Information Network (EWIN) en inglés. Éste es un proyecto financiado por el gobierno británico por medio de la institución *Engineering and Physical Sciences Research Council (EPSRC)* y apoyado por el *Global Challenges Research Fund (GCRF)* que tiene como objetivo promover el desarrollo económico y el bienestar de los países en desarrollo mediante la financiación de la investigación internacional.

Para este propósito, tres grupos de investigadores en México y el Reino Unido son respaldados por empresas de ambos países. Uniéndose así para implementar un trabajo capaz de pronosticar inundaciones utilizando teléfonos celulares y

tecnología *WiFi* / *LoRa*.

Además de la detección y prevención de inundaciones, *EWIN* engloba otras áreas de oportunidad: monitorización en el cuerpo de aguas dulces, optimización del uso de tecnologías celulares disponibles al implementar una red 3G/4G para aumentar la cobertura de comunicación de los sensores, transparencia en la disponibilidad de información, apoyo al uso de infraestructura digital y diseño de una red ad-hoc de sensores móviles.

1.4. Hipótesis

El añadir una capa *MAC* al protocolo *LoRa* y utilizarla para la comunicación entre dispositivos que empleen esta tecnología, permitirá la reducción de pérdidas de paquetes de información en redes de telecomunicaciones montadas completamente en escenarios de ríos. Además de obtener como resultado un bajo consumo energético por parte de los dispositivos de dicha red.

1.5. Objetivo

1.5.1. Objetivo general

Simular y caracterizar el comportamiento dinámico de una red *IoT* con el protocolo *LoRa* a partir del desarrollo e implementación de un red con características definidas para un sistema de monitorización de ríos.

1.5.2. Objetivos específicos

- Creación de un módulo de *LoRa* para el simulador de eventos discretos NS-3 que tenga un modelo de propagación para el escenario de los ríos.
- Modelado de una red *ad-hoc* basado en boyas móviles que incluye diferentes tipos de sensores y usando el protocolo *IoT LoRa*.
- Caracterización del modelo de propagación por medio de simulaciones.
- Definición de un modelo de movilidad similar a la de los ríos del estado de Colima para la red *ad-hoc*.

- Comparación del modelo de propagación propuesto con los resultados de simulación.
- Modelado del consumo energético de las boyas móviles en la red *ad-hoc* propuesta.

1.6. Metodología

En general, la metodología utilizada para el desarrollo de esta tesis es cuantitativa. Pues tiene un alcance experimental, que además representa la base de este trabajo. Es decir, los resultados obtenidos provienen completamente de simulaciones realizadas por medio de un programa de cómputo. En este caso, el simulador de eventos discretos *NS-3*, el cual es utilizado frecuentemente en la comunidad científica internacional. El principal tema evaluado fue el hecho de si los ríos pueden o no ser escenarios potenciales para el uso de tecnologías *IoT* como lo es *LoRa*, cuando de sensar y monitorear fenómenos naturales se trata.

1.6.1. Etapa 1: Compresión del módulo LoRaWAN

Es importante mencionar que el hecho de haber utilizado *NS-3* para este trabajo es gracias a que este simulador, cuenta con un módulo LoRaWAN que ha servido de base para este trabajo. En éste, se aprecia el diseño a nivel de capa física y también en la capa de control de acceso al medio.

1.6.2. Etapa 2: Eliminación de la capa MAC

Posteriormente, se diseñó e implementó un nuevo modelo de la capa *MAC* para dispositivos *LoRa*. Para esto, se tuvo que descartar la capa *MAC* del módulo LoRaWAN, con el fin de desarrollar una nueva que se apegará a las necesidades del proyecto de renombre internacional *EWIN*. Obteniendo así una capa desarrollada exclusivamente para el tipo de escenarios de esta tesis, donde *LoRa* es la tecnología utilizada.

1.6.3. Etapa 3: Validación del módulo LoRa

Más adelante se procedió a realizar la validación del módulo *LoRa* en el simulador. Esto por medio del experimento de un escenario simple con un dispositivo

LoRa emisor y otro receptor donde se obtenía el valor del tiempo en el aire del envío de un paquete de información y otro de confirmación *ACK*.

1.6.4. Etapa 4: Integración de módulos que faltaban

A continuación, se agregó al módulo *LoRa* la capacidad de representar la tasa de error de bit (*BER*) ya que antes carecía de este mecanismo. Además, un nuevo modelo de movilidad fue implementado en *NS-3* con la finalidad de utilizarlo en ríos, pues este trazara una ruta en la cual viajan dispositivos como las boyas de las simulaciones de este trabajo. También se agregó el *path loss* para ríos basado en investigaciones científicas en México y el Reino Unido.

1.6.5. Etapa 5: Diseño e implementación de la capa MAC

Esta etapa corresponde a la contribución más importante del trabajo durante esta tesis, pues se implementó un sistema de envío y recepción de paquetes de información cuya transmisión depende de ciertas características.

Además, de la implementación de confirmación de paquetes por medio de mensajes *ACK* en esta capa. El envío de paquetes se realiza solo a distancias cortas, por que cuando está muy lejos el paquete no va a llegar. Se toman varias condiciones para la transmisión y recepción de paquetes.

1.6.6. Etapa 6: Pruebas y de resultados del módulo

Después de todo el desarrollo realizado en las etapas anteriores, se procedió a la realización de pruebas del módulo para el análisis de los resultados obtenidos. Esto mediante la puesta en marcha de diversos escenarios con diferentes características entre sí. El escenario A tiene por objetivo conocer el comportamiento del sistema cuando se usa un receptor y transmisor. El B, realiza diferentes corridas con diferentes valores de velocidad para conocer el comportamiento del módulo en cada uno de estos casos. El objetivo del escenario C es obtener un promedio dependiente de diferentes simulaciones, en las cuales cada una de ellas tiene un valor diferente de velocidad de la boya. En el escenario D se cambia la velocidad constante por una variable para conocer la respuesta del módulo bajo estas condiciones. En el escenario E se usan ahora 100 nodos o boyas, además del algoritmo de retroceso exponencial. Con el fin de conocer la respuesta del módulo cuando se usa un número grande de dispositivos y un algoritmo de planificación de paquetes. Finalmente, en el escenario F se utiliza una velocidad diferente en cada una de las 100 boyas para tener un escenario mucho más realista.

1.7. Contribuciones

La principal contribución de este trabajo es el diseño e implementación de la capa *MAC* para los dispositivos finales, o boyas móviles (*end devices*). Durante el desarrollo de esta tesis se han tenido diversas contribuciones que se muestran a continuación:

- Implementación de un modelo de *path loss* para ríos.
- Implementación de las características *MAC* necesarias para el módulo, activando confirmación de recepción de paquetes por medio de *ACK's*.
- Diseño e implementación de las cabeceras necesarias para el protocolo de monitorización para ríos.
- Implementación del algoritmo de retroceso exponencial para un mejor control de flujo, que permite como máximo un número de 5 retransmisiones por paquete.
- Implementación de un modelo de movilidad para ríos donde la velocidad pueda ser configurable. Apegándose al comportamiento de la red *EWIN*.

Pero también se ha realizado una contribución de ingeniería a nivel de capa física:

- Implementación de un módulo de error que se encarga de asignar una probabilidad de error de paquete a partir de una aproximación matemática. Se obtiene así, el bit error rate.

1.8. Organización de la tesis

Este trabajo está dividido en VI capítulos. El segundo capítulo corresponde al estado del arte, de manera concisa se abordan las tecnologías *IoT*, además de la evolución de las redes en este ámbito, las que están presentes en nuestros tiempos y los trabajos relacionados que han sido base de esta tesis. Se proporciona también una concisa pero muy detallada explicación sobre las aportaciones del proyecto de red de emergencia de información sobre agua (*EWIN*) y sus alcances.

Posteriormente, en el capítulo tres se presentan a detalle los conceptos básicos de la tecnología *LoRa*. En este capítulo se profundizan los conceptos que involucran dicha tecnología. Además, se describe el modelo de propagación del proyecto

EWIN, mismo que ha sido clave para el conocimiento del rendimiento de nuestra red.

El cuarto capítulo muestra una descripción del modelo de simulación de *LoRa* en *Network Simulator 3*. Aquí, se abarcan temas básicos del simulador como lo son los eventos discretos, la pila de protocolos y el modelo de la capa física del módulo. También se analiza el diseño de la capa *MAC* de las boyas del módulo *LoRa* desarrollado para esta tesis, presentándose los encabezados y comandos *MAC* que fueron implementados.

En el capítulo cinco se hace un estudio a fondo del comportamiento dinámico de nuestro módulo LoRa que es utilizado en diferentes configuraciones de escenarios. Se muestran diferentes escenarios, desde el escenario A hasta el E, que van desde el uso de una única boya que tiene una velocidad constante hasta el uso de 100 de estas. Se presenta además información útil sobre velocidad en ríos y rápidos obtenida a partir de una publicación científica.

Finalmente se muestran los trabajos futuros que pudieran derivarse de esta tesis y las conclusiones finales en el capítulo seis.

Conceptos básicos y estado del arte

Internet of Things (IoT) se ha vuelto una herramienta vital en el desarrollo tecnológico de la sociedad humana, esto gracias a los beneficios que ofrece tanto a los humanos como a la información y a las cosas mismas. Como por ejemplo, el conocimiento del comportamiento del usuario, la mejora del servicio al cliente y la experiencia de usuario, la reducción de costos y la mejora en la toma de decisiones.

En este capítulo se abarcan los conceptos básicos y estado del arte de nuestro trabajo, desde sus inicios hasta la época actual; se describe a detalle *IoT*, diversas tecnologías abarcadas en este concepto, su evolución a través del tiempo y los trabajos relacionados que se desempeñan como base para el desarrollo del módulo diseñado en esta tesis.

2.1. Tecnologías IoT

IoT sigue siendo uno de los grandes avances tecnológicos. En términos generales, si se aplica correctamente, permite que los dispositivos no sólo sean generadores o actuadores de información controlables remotamente; sino que también permite la generación de valor en una operación. Un propósito del *IoT* es hacer que los dispositivos contribuyan a la generación de datos para una mejor toma de decisiones, generando así información compactada y extremadamente útil para la mejora continua de los procesos. Sin embargo, no todas las ubicaciones donde existen este tipo de procesos que deben ser mejorados están en cobertura de la señal de una red celular o inalámbrica con acceso a Internet [1].

Para el caso del escenario planteado en esta tesis, no todos los nodos (*end devices*) tienen acceso a Internet a través de la red 3G o 4G.

2. CONCEPTOS BÁSICOS Y ESTADO DEL ARTE

Por lo tanto, debe existir un dispositivo concentrador llamado *gateway* el cuál debe tener acceso a Internet y debe ser capaz de comunicarse con estos dispositivos de forma remota; integrando así a los dispositivos en ubicaciones remotas a sistemas conectados a Internet. En esta comunicación con los dispositivos hay varias tecnologías de radios elegibles, pero una de ellas ha destacado: *LoRa*. La tecnología *LoRa* permite la comunicación inalámbrica, con bajo consumo energético y con alcance de comunicación en el orden en kilómetros, siendo así muy adecuada para proyectos de *IoT*.

La figura (2.1) muestra una comparación entre el alcance de comunicación normalmente medido en metros y el *data rate* o velocidad de datos, de tecnologías inalámbricas en telecomunicaciones. Algunas tecnologías mencionadas anteriormente son mostradas y posicionadas en el gráfico.

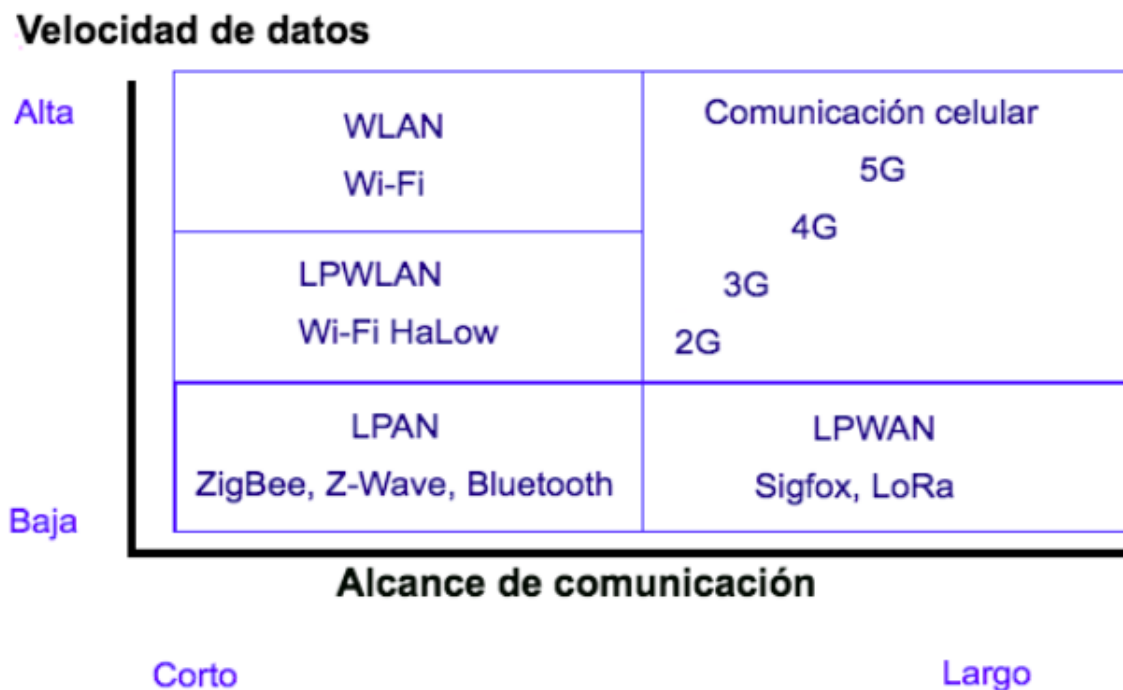


Figura 2.1: Velocidad de transferencia de datos contra alcance de comunicación.

Puede apreciarse que *LoRa* se encuentra dentro de las tecnologías de baja velocidad de datos pero muy alto alcance de comunicación, junto con la tecnologías clasificadas dentro de las redes de área amplia de baja potencia. Esta es una ventaja muy grande para el desarrollo de este proyecto, ya que existen otras tecnologías con alto alcance de comunicación como las de comunicación celular,

pero también con altos valores en la velocidad de datos; algo que resultaría muy costoso a nivel energético.

Por otro lado, las tecnologías *LPAN*, simplemente no serían opción ya que carecen de un amplio alcance de comunicación incluso siendo tecnologías de bajo consumo. En cuanto a las tecnologías como Wi-Fi se encuentran en el lado opuesto de nuestro gráfico, siendo así las tecnologías menos indicadas para este proyecto ya que además de consumir un alto nivel de energía, también tiene un corto alcance de comunicación.

Las redes de área amplia de baja potencia (*LPWAN*) están ganando impulso como una solución fácilmente disponible para la demanda de conectividad de *IoT*, principalmente gracias a los bajos costos de implementación y la ausencia actual de competidores basados en celulares para el uso de *IoT* de casos de baja demanda de conectividad [15].

Sobresalen dos tendencias que se utilizan en *IoT*: Las *LPAN* y *LPWAN*. Las *LPAN* suelen ser tecnologías con alcances de hasta 100 metros. Entre las cuales para *IoT* están *ZigBee*, *Z-wave* y *Bluetooth*. Por otra parte, las *LPWAN* tienen un alcance mucho mayor en comparación a las antes mencionadas alcanzando el orden en kilómetros. Entre estas tecnologías podemos mencionar a *LoRa*, *802.11ah* y *Sigfox*.

2.2. Evolución de redes IoT

2.2.1. Internet de las cosas

Internet de las cosas es un paradigma de comunicación reciente que vislumbra un futuro cercano, en el que los objetos de la vida cotidiana estarán equipados con microcontroladores para la comunicación digital y pilas de protocolos adecuados que les permitirán comunicarse entre sí. y con los usuarios, convirtiéndose en parte integral de Internet [16].

Este es un término que hace referencia a una interconexión digital entre objetos cotidianos con Internet. Los objetos de *IoT* tienen identificadores que son únicos. Tienen la capacidad de enviar la información a través de una red que interconecta cosas físicas y virtuales sin requerir interacción de persona a persona o de persona a sistema.

De manera más simple, es la conexión con Internet de objetos más que con personas. Esta conexión, tiene como objetivo un cambio radical y positivo en la calidad de vida de los humanos, ofrece una gran cantidad de nuevas oportunidades de acceso a datos, servicios específicos en la educación, seguridad, asistencia

sanitaria y en el transporte, entre otros campos. Es decir, *IoT* busca unir todo en el mundo bajo una infraestructura común.

El trabajo cotidiano de la sociedad humana ha dado cabida a un sin fin de aplicaciones en varios sectores sociales, tecnológicos y económicos. Gracias a todas estas necesidades, surge el nacimiento de una de las herramientas más importante de la actualidad: Internet. La evolución y revolución de esta herramienta fundamental resulta entonces en Internet de las cosas. *IoT* está en todas partes proporcionando un puente entre ricos y pobres, distribuyendo los recursos del mundo a quienes lo necesitan. De tal modo que estas oportunidades impacten en cada parte de la sociedad [1].

Así, las ciudades inteligentes, hogares inteligentes, monitores de desechos y aplicaciones agrícolas (como riego inteligente) pueden construirse donde haya conexión a Internet.

2.2.2. IoT a través del tiempo

Es importante hablar de la evolución de Internet para una mejor comprensión del paso de IoT a través de los años.

A principios de los años 60, fue posible la comunicación entre dos computadoras a través de una red de computadoras. Posteriormente, en los años 80 se introdujo la pila de protocolos TCP/IP. Y en 1991 la red informática mundial (WWW) se vuelve disponible en el mundo, haciendo más popular a Internet.

La figura (2.2) muestra como se crearon nuevos dispositivos a través de los años, como teléfonos que después fueron conectados a Internet. Las redes sociales y aplicaciones comenzaron a ganar popularidad en la última década. De esta manera, el siguiente paso en *IoT* es conectar entre sí a todos los objetos al rededor nuestro, comunicándose vía Internet [1].

Hoy en día, el consumo de energía y ancho de banda desafía el crecimiento de *IoT*. El deseo de compartir información y datos enriquecidos domina el uso de recursos, lo que supone un desafío para la duración de la batería y la carga de la red para dispositivos inalámbricos distribuidos [17].

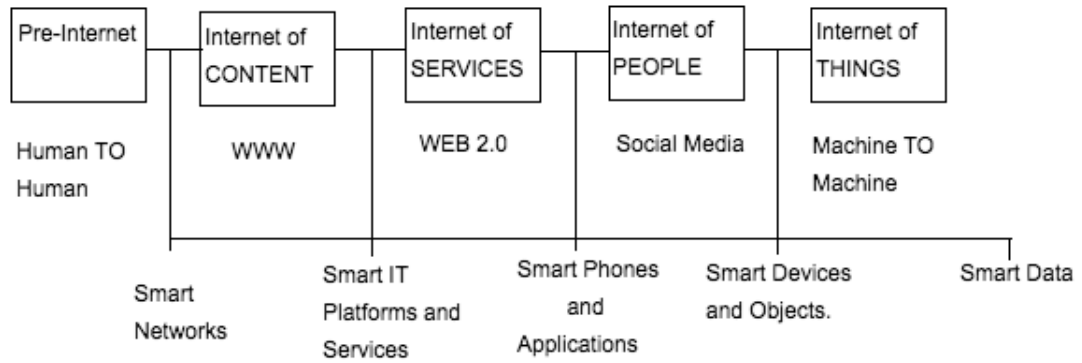


Figura 2.2: Evolución de *IoT* [1].

2.2.3. Tecnologías IoT actuales

Las tecnologías actuales de *IoT* cubren toda la pila de protocolos, desde la capa física hasta la capa de aplicación. Además, surgieron nuevas capas de datos diseñadas específicamente para *IoT*.

Para aplicaciones de corto alcance, estos incluyen, entre otros, identificación por radiofrecuencia (*RFID*), *Bluetooth Low Energy (BLE)*, comunicación de campo cercano (*NFC*), cuarta generación de sistemas celulares (4G y 5G), *IEEE 802.15.4* y el reciente *IEEE 802.11ah*. Para aplicaciones de largo alcance, las tecnologías Low Power Wide Area Networks (LPWAN) incluyen el protocolo LoRaWAN y el futuro *IoT* celular. Además, las tecnologías de web semántica también serán un habilitador clave para *IoT* [18].

El principal desafío del proyecto *EWIN* será encontrar los protocolos inalámbricos más apropiados que permitan la transmisión de los datos recopilados por cada uno de los *end device* y transmitir esta información en tiempo real hacia el servidor en la nube, teniendo en cuenta la alta restricción de consumo y el amplio alcance de comunicación entre los *end devices* y la red central, con el fin de reducir los costos de equipo y gestión. La batería de estos dispositivos debe durar varios días, a fin de contar con la información adecuada para la caracterización del modelo de inundación de los ríos de Colima.

Se han encontrado una gran lista de protocolos de comunicación inalámbrica para Internet de las cosas. Sin embargo, no todos estos protocolos inalámbricos se pueden utilizar para el proyecto *EWIN* debido a su alcance de cobertura reducido o su alto consumo de energía.

2. CONCEPTOS BÁSICOS Y ESTADO DEL ARTE

Como la figura (2.3) muestra, estos protocolos inalámbricos se clasifican en dos grupos principales:

- Tecnologías de red inalámbrica personal y de área local de comunicación de corto a medio alcance (*WPAN\LAN*)
- Tecnologías de red inalámbrica de área amplia de comunicación de largo alcance (*WWAN*)

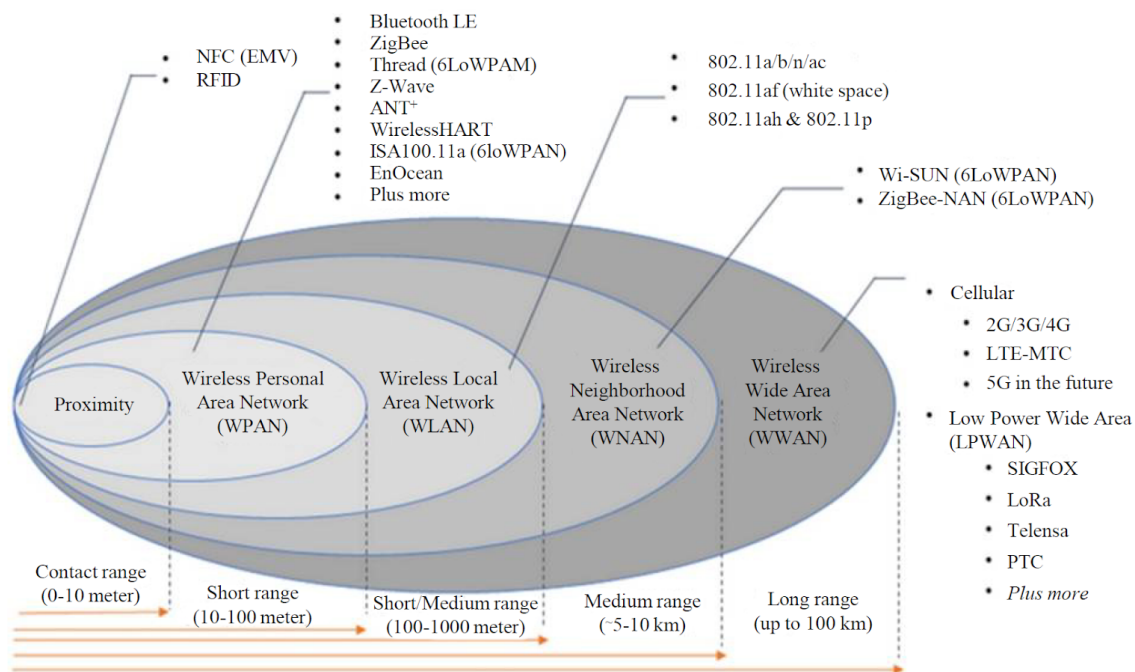


Figura 2.3: Protocolos de comunicación inalámbrica para IoT [2].

En el primer grupo *WPAN\LAN* podemos encontrar los protocolos inalámbricos *Bluetooth*, *ZigBee*, *6LoWPAN* y *Wi-Fi*, y en el segundo grupo *WWAN* tenemos;

- Los protocolos inalámbricos con licencia de espectro, como Redes Celulares: 2G, 3G, 4G-LTE y 5G (LTE A, versiones 10 a 14 en el futuro).
- Los protocolos inalámbricos exentos de licencia como *Low Power Wide Area Networks (LPWAN)*, *Long Range (LoRa)* y *SIGFOX*.

2.3. Aportación de la tesis

La principal aportación de esta tesis es comprobar por medio de simulaciones si los *end devices* o boyas, utilizados en la red *LoRa* que representa al proyecto EWIN, son capaces de cumplir con los objetivos planteados desde su desarrollo e implementación física. Los siguientes aspectos en los dispositivos que integran la red *LoRa* serán evaluados (mediante el uso del simulador *NS-3*):

- Capacidad de comunicarse y enviar todos los paquetes de información, recolectados a lo largo de la ruta, hacía los *gateway*. Tomando en cuenta que se tienen ciertos fenómenos físicos que propician la pérdida de paquetes como el *shadow fading* y las pérdidas por *path loss*. Además de ciertos comportamientos que se traducen en pérdidas de información debido al río que se recorre, como la inversión o estancamiento de las boyas.
- Conocer el retardo promedio de los paquetes de información, para que en el trabajo futuro se disminuya en caso de tener valores excesivamente altos.
- Validar el comportamiento de un único *end device* en la red.
- Verificar el comportamiento de la red cuando se tienen 100 o más dispositivos en el escenario de simulación.
- Ratificar el comportamiento de los dispositivos de la red cuando se tienen cambios muy severos en la velocidad.
- Comprobar el comportamiento de la red cuando se le asigna a los *end devices* una probabilidad de estancamiento y una probabilidad de inversión en el agua del dispositivo.
- Capacidad de las boyas de alcanzar sus objetivos sin que la batería se agote debido al consumo de los diferentes estados de las boyas y del GPS que tienen integrado.
- Obtener los valores exactos de consumo total de batería consumida de los *end devices*.
- Conocer el porcentaje de consumo de cada estado del *end device* y del GPS, sobre el total de consumo en cada simulación. Es decir asignar un porcentaje a cada uno de los estados del dispositivo para comprender mejor su comportamiento.

2.4. Trabajos relacionados

Para el desarrollo de esta tesis, se han consultado diversas publicaciones científicas que tienen una relación muy fuerte con el trabajo realizado en este módulo. El primero de ellos es un artículo derivado del proyecto *EWIN*, cuyas aportaciones se explican a detalle en esta sección. En segundo lugar, tenemos otra obra que también representa una de las bases principales de esta tesis, un módulo implementado en *NS-3* que cuenta con la capa física *LoRa* y la capa *MAC LoRa-Wan*. Posteriormente, se habla sobre estudios y trabajos para la monitorización de las inundaciones. Además de ello, se presenta una aproximación matemática que se implementó en el módulo para brindarle la capacidad de calcular el *Bit Error Ratio*. También, se hace referencia al estudio del modelo de propagación de *LoRa* que ha sido implementado en el módulo para el cálculo de pérdidas durante la propagación de las señales.

2.4.1. Proyecto EWIN

El proyecto *EWIN* (Red de emergencia de información sobre agua) nace de la necesidad de control que necesitan las naciones sobre las inundaciones. Este tipo de fenómeno crece por todo el planeta y con el pasar de los años se vuelven más peligrosos e intensos debido al cambio climático. Este proyecto es un sistema de detección de inundaciones. Para ello, utiliza métodos de predicción y detección hidrológica, combinados con las redes móviles que se encuentran en el planeta. Para ello, el proyecto utiliza diversos recursos como las estaciones de medición para pronosticar inundaciones del Reino Unido que además se tiene planeado aumentar en número. Con ayuda de esta tecnología y de la información que proporciona, en México se ha desarrollado un modelo matemático hidrológico para predecir los niveles de agua en áreas rurales y urbanas. En el futuro, se prevé incluir el análisis de inundaciones de aguas continentales, las inundaciones provenientes del mar y las inundaciones repentinas, las cuales podrían estudiarse utilizando técnicas similares. En cuanto a la composición electrónica del proyecto, podemos decir que este se integra de sensores colocados en cuerpos de agua. Además de que el sistema incluye soporte en radio y el uso de espectrogramas. Cabe mencionar que la infraestructura es altamente económica. [19].

Entre los beneficios de *EWIN* se tienen [20]:

- Proporcionar información a las autoridades y los servicios de emergencia y gestión de inundaciones pertinentes para reducir el riesgo.
- Salvaguarda de la vidas humanas.

- Impacto menor a los daños a la propiedad y al medio ambiente.

A partir de todas estas razones y tomando en cuenta el escaso desarrollo de tecnología en el monitorización de aguas continentales, sistemas obsoletos y nulo control de información, el proyecto *EWIN* se perfila como una solución eficiente a estos problemas.

2.4.2. Estudio del rendimiento LoRaWAN

Como primera base de este trabajo, se tomó como referencia el módulo *LoRaWAN* desarrollado en *NS-3* en [21]. Podemos notar que el autor del desarrollo de este módulo, evaluó varias métricas de rendimiento de una red *LoRa*. Realizó varias pruebas para estimar el rendimiento, la probabilidad de error de paquetes y la cobertura de la puerta de enlace *gateway*.

Los resultados de su simulación mostraron que el esquema de acceso *LoRaWAN* proporciona un mayor rendimiento con respecto a un esquema *ALOHA* básico, gracias a la ortogonalidad parcial entre sus factores de esparcimiento. Además, muestra que la arquitectura *LoRaWAN* puede escalar bien, principalmente debido al hecho de que un aumento en el número de *gateways* también mejora la cobertura y confiabilidad del enlace ascendente. Finalmente, una simulación que involucró una red con múltiples *gateways* y un modelo de tráfico realista resultó en una tasa de éxito de paquetes superior al 95% para un *gateway* que da servicio a aproximadamente 15,000 dispositivos finales *end device* [21].

Este módulo comprende dos modelos principales: uno para la capa física de *LoRa* que necesita representar los chips *LoRa* y el comportamiento de las transmisiones *LoRa*, y otro para la capa *LoRaWAN MAC*, que debe comportarse de acuerdo con las especificaciones oficiales.

Para representar estos dos modelos, el módulo presenta dos clases genéricas *LoraPhy* y *LorawanMac*. Estas clases se extienden luego por clases que modelan las peculiaridades de los dos dispositivos de red inalámbrica: el Dispositivo final (*end device*) y la Puerta de enlace (*gateway*). Por lo tanto, la capa física ha podido ser modelada mediante el uso de *EndDeviceLoraPhy* y *GatewayLoraPhy*, mientras que los objetos de la clase *EndDeviceLorawanMac*, *ClassAEndDeviceLorawanMac* y *GatewayLorawanMac* se utilizan para representar la capa *MAC*.

2.4.3. Monitorización de inundaciones

Los grupos sociales más vulnerables a las inundaciones generalmente radican los países en vías de desarrollo. Y es en estos donde las pérdidas de vidas son más grandes a comparación de países que tiene economías más estables. Un ejemplo de

ello, es el Reino Unido ya que éste tiene un sistema de defensa contra inundaciones mucho más desarrollado a comparación de otras naciones. Un hecho es que muchos países no cuentan con los recursos para la integración de una defensa a este problema. Además de que también carecen de experiencia en el uso de estas tecnologías debido a su alto costo. Afortunadamente la mayoría de las naciones cuentan con sistemas de redes de tecnología celular en sus territorios. Esto hace posible la implementación del proyecto *EWIN* en sus ciudades y pueblos, como un sistema de predicción y alerta de inundaciones [22].

A partir de los estudios realizados al proyecto *EWIN* se han obtenido resultados que validan que se trata de un sistema confiable, preciso y muy eficiente. Gracias al cual se tienen minutos de advertencia ante situaciones de riesgo por parte de fenómenos naturales como lluvias e inundaciones. Esto permite salvar muchas vidas en regiones donde es común las pérdidas humanas ocasionadas por este tipo de desastres naturales. La información generada por la red de información de agua de emergencia *EWIN* aminora el impacto de las inundaciones en la ciudad de Colima [23].

Sobre las bases mencionadas anteriormente, se han realizado diversas pruebas en campo gracias a las cuales podemos saber que se debe dar mantenimiento al sistema regularmente. Además se deben agregar más funcionalidades, en la elección del intervalo de muestreo y agregar un registro de la señal del GPS. [20]

2.4.4. Bit Error Rate (BER) de la tecnología LoRa

Para escenarios como el estudiado en esta tesis, LoRa ha demostrado estar a la vanguardia en comparación con las redes de área amplia de baja potencia. La modulación LoRa, que describe una técnica de modulación de espectro ensanchado chirp, admite una comunicación de largo alcance confiable y con eficiencia energética. A pesar de esto, el rendimiento de la tasa de error de bits subyacente (*BER*) de la modulación LoRa aún no se había implementado en el módulo para *NS-3* desarrollado por [21], mismo que se usa para el desarrollo de este proyecto. En la publicación científica [24], se deriva una aproximación de forma cerrada para el rendimiento del *BER* en *LoRa*. Y es justamente la aproximación matemática, la aportación de esta publicación científica [24] al módulo *LoRa* desarrollado para esta tesis.

Las curvas de la figura (2.4) muestran la probabilidad de Bit en error P_b cuando varía la relación señal a ruido (*SNR*) para los diferentes factores de esparcimiento que se utilizan en *LoRa*.

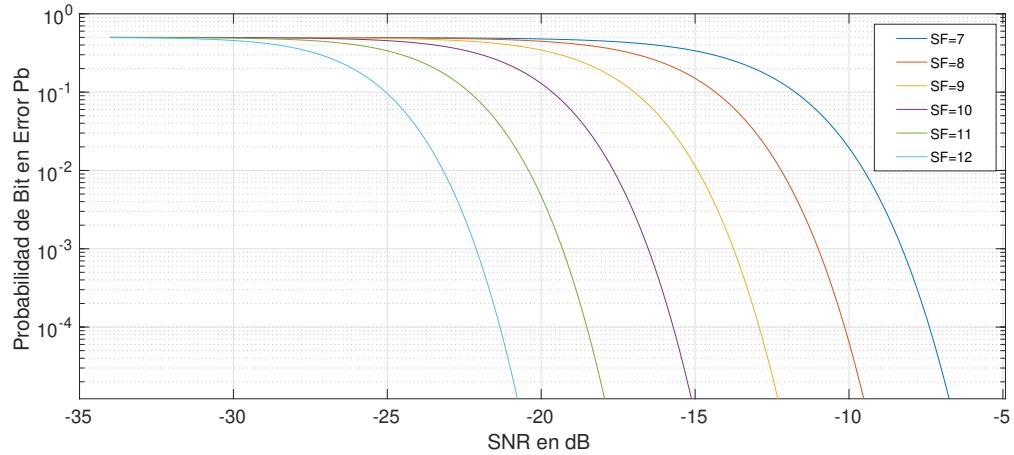


Figura 2.4: LoRa BER.

La ecuación (2.1) proporciona una aproximación del valor de la probabilidad de bit en error (P_b).

$$P_b \approx 0.5 * Q(\sqrt{\Gamma * 2^{SF+1}} - \sqrt{1.386 * SF + 1.154}) \quad (2.1)$$

Donde Q es la función $Q(z)$, SF es el factor de esparcimiento y Γ es el SNR para la comunicación LoRa. .

Se puede apreciar en la ecuación (2.1) que el factor de esparcimiento es una parte muy importante de la misma; pues aparece dos veces en esta aproximación. Es evidente que al aumentar valor del factor de esparcimiento, el valor del SNR disminuye notablemente.

La relación señal/ruido (SNR) es la relación entre la señal de potencia recibida y el nivel de potencia del ruido de fondo en todas la instancias (*noise floor*).

El ruido de fondo en todas la instancias es la suma de todas las fuentes de ruido y las señales no deseadas que pueden corromper la señal transmitida y por lo tanto provocar retransmisiones.

Si SNR es mayor que 0, la señal recibida opera por encima del ruido de fondo en todas la instancias. Si SNR es menor que 0, la señal recibida opera por debajo de éste. Normalmente, el ruido de fondo en todas la instancias es el límite físico de sensibilidad, sin embargo, *LoRa* funciona por debajo del nivel de ruido. Los valores típicos de *LoRa SNR* están entre: -20dB y + 10dB. Un valor más cercano a +10dB significa que la señal recibida está menos dañada.

LoRa puede demodular señales que están entre -7,5 dB y -20 dB por debajo del ruido de fondo en todas la instancias.

2.4.5. Estudio de propagación de LoRa en ríos con vegetación

Como hemos discutido anteriormente, recientemente se implementó un sistema de bajo costo para estudiar las condiciones de los ríos en el estado de Colima en México: el proyecto *EWIN*.

Esto ha sido posible gracias al trabajo de científicos mexicanos de la Universidad Nacional Autónoma de México, la Universidad de Colima y la universidad británica de Sheffield. Quienes se han encargado de realizar diversos estudios científicos para este objetivo. Uno de ellos, el estudio de propagación de *LoRa* en ríos con vegetación.

El trabajo de investigación antes mencionado se centra en la propagación de señales sobre ríos dentro de la vegetación utilizando un *end device* de *LoRa* con un enlace punto a punto con un *gateway* donde tanto la altura de la antena de transmisión como la altura de la antena de recepción se posicionan cerca del agua. Esta publicación proporciona un estudio de la propagación de señales en un entorno mixto de ríos (agua dulce) dentro de la vegetación [25].

El modelo de *path loss* desarrollado en esa publicación científica, fue implementado en el módulo *LoRa*.

Tecnología LoRa

LoRa es una tecnología de radiofrecuencia que permite comunicaciones a larga distancia (en el orden de magnitud de unos pocos kilómetros) con bajo consumo de energía. La palabra *LoRa* proviene de *Long Range*, que en español significa largo alcance. En este capítulo se proporcionan los conceptos y definiciones que abarca este protocolo. Se presentan las bandas que usa alrededor del mundo, las ventajas de esta tecnología, la descripción a fondo de su capa física y, finalmente, la descripción del modelo de propagación del proyecto que ha sido el principal impulso de esta tesis: el proyecto *EWIN*.

3.1. Bandas de frecuencia que utiliza LoRa

En LoRa, el nodo es un emisor de baja potencia que transmite, velocidades de datos que se encuentran en el rango de 0.183 bps a 27.34 kbps, dependiendo del factor de esparcimiento, la codificación y el ancho de banda de canal, a un receptor a través de una larga distancia [26].

En términos de frecuencias libres de funcionamiento, la tecnología LoRa utiliza frecuencias por debajo de 1 Ghz en bandas dedicadas, según las regiones del planeta [9].

La tabla (3.1) muestra las bandas y frecuencias, además de las regiones del planeta. También hace referencia a los planes de canales disponibles para quienes se encargan de implementar esta tecnología en cada región del mundo. En *LoRa*, el protocolo utilizado para acceder al canal es *ALOHA* [27], es decir, en cuanto se tiene el paquete, se transmite por el canal.

3. TECNOLOGÍA LORA

Tabla 3.1: Bandas de frecuencia LoRa en el planeta [9].

Región	Banda	Plan del canal
Australia	Desde 915 Mhz hasta 928 Mhz	AU915-928
América del norte	Desde 902 Mhz hasta 928 Mhz	US902-9284
Europa	Desde 863 Mhz hasta 870 Mhz	EU863-870
China	Desde 779 Mhz hasta 787 Mhz	CN779-787

3.2. Ventajas del uso de LoRa

LoRa presenta grandes ventajas de uso en la comunicación entre dispositivos [28]:

- Alcance de hasta 11 km, dependiendo de las características de cada escenario.
- Bajo consumo de energía eléctrica.
- Con *LoRa*, la interferencia puede ser hasta 19 dB mayor que la señal recibida.
- Bajo costo.

La figura (3.1) muestra una comparación entre energía consumida y rango de las tecnologías *Bluetooth*, *Celular*, *WiFi* y *LoRaWAN*.

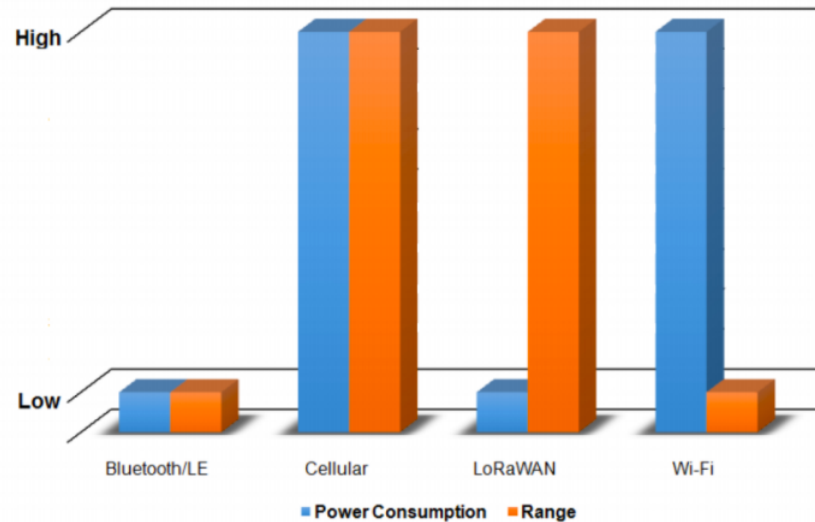


Figura 3.1: Comparación de LoRa con otras tecnologías [3].

El consumo de la tecnología *LoRa* es de los más bajos a pesar de que el rango de comunicación es muy alto. Lo que hace el protocolo, resulte adecuado para la implementación del trabajo de esta tesis. *LoRa* es una nueva tecnología que está tomando gran importancia en el campo de las telecomunicaciones rápidamente, esto ya que aborda las necesidades de un dispositivo integrado que funciona con un bajo consumo de batería.

3.3. Capa física

LoRa utiliza una modulación de espectro ensanchado de chirp, que usa chirridos de frecuencia con una variación lineal de la misma a lo largo del tiempo para codificar la información. Debido a la linealidad de los pulsos de chirrido, las compensaciones de frecuencia entre el receptor y el transmisor son equivalentes a las compensaciones de tiempo, que se eliminan fácilmente en el decodificador[29]. Esto también hace que esta modulación sea inmune al efecto *Doppler*, equivalente a un desplazamiento de frecuencia. El desplazamiento de frecuencia entre el transmisor y el receptor puede alcanzar el 20 % del ancho de banda sin afectar el rendimiento de la decodificación [10].

3.3.1. Chirp

En comunicaciones digitales, el espectro extendido de *chirp*, en inglés *Chirp Spread Spectrum (CSS)* es una técnica de espectro extendido que utiliza pulsos de chirrido modulados de frecuencia lineal de banda ancha para codificar información. Un *Chirp* es una señal sinusoidal de aumento o disminución de la frecuencia con el tiempo (a menudo con una expresión polinómica para la relación entre el tiempo y la frecuencia) [4]. En la figura (3.2) se muestra un ejemplo en el que la frecuencia aumenta linealmente con el tiempo. A veces, la frecuencia de los incrementos aumenta exponencialmente con el tiempo. Un pitido de subida puede derivarse de una expresión polinómica para frecuencia contra tiempo. Los *chirps* se desplazan cíclicamente, y son los saltos de frecuencia los que determinan cómo se codifican los datos en los *chirps*, también conocida como modulación LoRa.

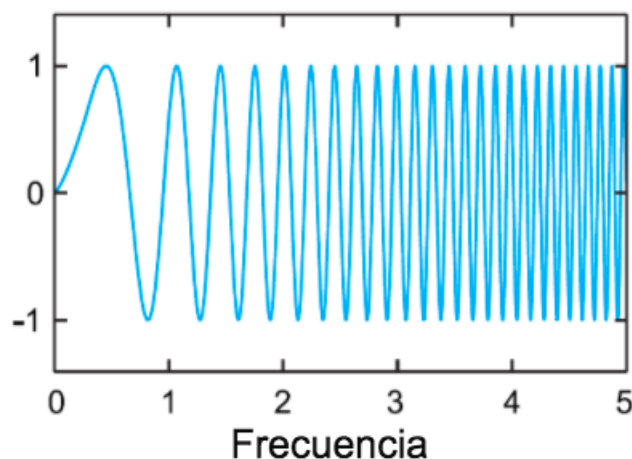


Figura 3.2: Ilustración de Chirp [4].

IEEE 802.15.4a, otro estándar de velocidad baja, lo especifica como técnica para su uso en redes de área personal inalámbricas. Esta técnica se ha utilizado durante muchos años para proporcionar comunicación sólida de largo alcance en las aplicaciones militares y espaciales, pero *LoRa* es la primera implementación comercial de bajo costo [30].

Al igual que con otros métodos de espectro ensanchado, el espectro ensanchado *chirp* utiliza todo su ancho de banda asignado para transmitir una señal, lo que lo hace robusto para el ruido del canal [4].

Además, debido a que los *chirps* utilizan una amplia banda del espectro, el espectro extendido de *chirp* también es resistente al desvanecimiento de múltiples rutas incluso cuando funciona a muy baja potencia. Sin embargo, a diferencia del

espectro ensanchado de secuencia directa (*DSSS*) o del espectro ensanchado de salto de frecuencia (*FHSS*), no agrega ningún elemento pseudoaleatorio a la señal para ayudar a disminuir el ruido en el canal, sino que confía en la naturaleza lineal del pulso de *chirp*. Además, el espectro extendido de *chirp* es resistente al efecto *Doppler*, que es típico en aplicaciones de radio móvil [31].

3.3.2. Modulación de espectro extendido LoRa chirp

LoRa usa tres anchos de banda diferentes: 125 kHz, 250 kHz y 500 kHz. Los símbolos *LoRa* se modulan sobre un *Chirp* ascendente con uno de estos tres diferentes anchos de banda [32].

La figura (3.3), por medio de un espectrograma, muestra que la capa física *LoRa* incluye 8 símbolos de preámbulo, 2 símbolos de sincronización, después 2 símbolos y cuarto de *chirp* hacia abajo y finalmente se tienen los encabezados, la carga útil física y *CRC* que es opcional [5].

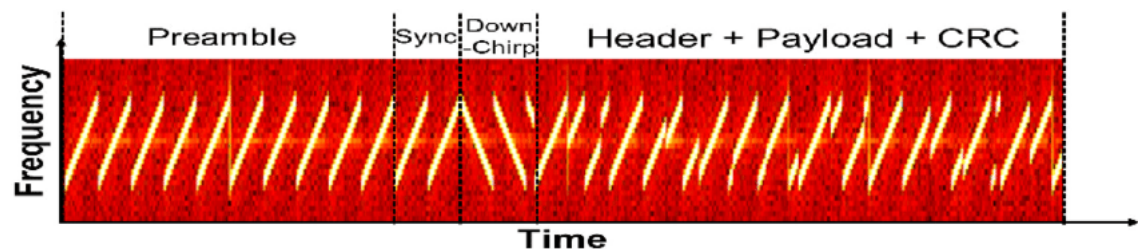


Figura 3.3: Símbolos de LoRa [5].

3.3.3. Impacto del factor de esparcimiento SF

Factor de esparcimiento: *LoRa* utiliza factores de esparcimiento que van desde SF7 hasta SF12 [6]. La figura (3.4) muestra un espectrograma de los diferentes factores de esparcimiento utilizando el ancho de banda de 125 kHz.

A mayor factor de esparcimiento también mayor será el tiempo de transmisión. Cuando baja el factor de esparcimiento, mayor será la transferencia. Además, un aumento de este factor con el mismo ancho de banda aumenta la *SNR* de la señal [32].

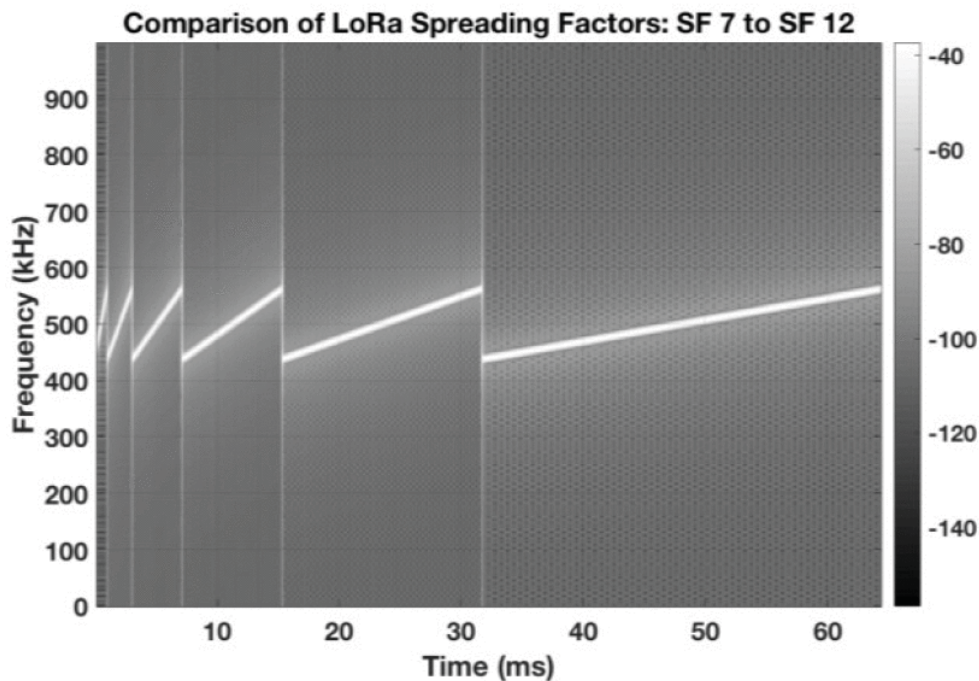


Figura 3.4: Comparación de SF7 hasta SF12 utilizando 125kHz [6].

Si aumenta el SF en un factor de 1, la duración del símbolo o el tiempo de barrido se duplica en comparación con el SF anterior.

SF8 toma exactamente el doble de tiempo de SF7 y SF9 toma exactamente el doble de tiempo de SF8. Esto sucede porque el tiempo de la duración del símbolo corresponde a una relación exponencial donde la base es siempre 2 con valor de exponente igual al factor de esparcimiento. Además, se reduce la tasa de bits aproximadamente a la mitad en comparación con el SF anterior. Esto quiere decir que el tiempo de transmisión del mensaje (ToA) aumenta, debido a que la distancia aumenta [6].

Para tener una mejor idea de cuál es el tiempo en el aire para una carga útil de 10 bytes, con un ancho de banda BW igual a 125kHz y un valor de factor de esparcimiento de 7 se tiene un tiempo de transmisión igual 41 ms.

Mientras que para la misma configuración, excepto por el valor del factor de esparcimiento de 12, se tiene un tiempo de transmisión igual a 991 ms.

Los dispositivos *LoRa* usan un factor de esparcimiento más alto cuando la señal es débil o hay mucha interferencia. El uso de un factor de esparcimiento más alto significa un mayor tiempo de transmisión (ToA).

Si un dispositivo final está más lejos de una puerta de enlace, la señal se debilita y, por lo tanto, necesita un factor de esparcimiento más alto.

3.3.4. Parámetros de la capa física

Hay varios parámetros disponibles para personalizar la modulación LoRa: ancho de banda (BW), factor de esparcimiento (SF) y tasa de codificación (CR) [3]. *LoRa* utiliza una definición poco convencional del factor de esparcimiento como el logaritmo, en base 2, del número de *chirp* por símbolo [4].

Estos parámetros influyen en el número de bits recibidos correctamente, su resistencia al ruido de interferencia y su facilidad de decodificación.

Un símbolo *LoRa* se compone de 2^{SF} *chirps*, que cubre toda la banda de frecuencia. Comienza con una serie de *chirps* ascendentes. Cuando se alcanza la frecuencia máxima de la banda, la frecuencia se envuelve y el aumento de frecuencia comienza de nuevo desde la frecuencia mínima. La figura (3.3) da un ejemplo de una transmisión *LoRa* en la variación de frecuencia a lo largo del tiempo. La posición de esta discontinuidad en frecuencia es lo que codifica la información transmitida. Como hay 2^{SF} chips en un símbolo, un símbolo puede codificar efectivamente $SF(7,8,9,10,11$ o $12)$ bits de información.

La tabla (3.2) muestra la relación entre el valor del factor de propagación y el número de chips.

Tabla 3.2: Chips en LoRa [4].

Factor de esparcimiento(SF)	Chips(2^{SF})
7	128
8	256
9	512
10	1024
11	2048
12	4096

Los primeros 8 símbolos de chirp ascendente son símbolos de preámbulo utilizados para detectar chirridos LoRa, los siguientes 2 símbolos de chirp descendente son símbolos de sincronización utilizados para la sincronización de temporización seguidos de los 5 símbolos modulados (carga útil) [32].

3.3.5. Duraciones y rangos en LoRa

En *LoRa*, la tasa de *chirp* depende solo del ancho de banda: la tasa de *chirp* es igual al ancho de banda (un *chirp* por segundo por Hertz de ancho de banda). Esto tiene varias consecuencias en la modulación: un aumento de uno de los factores de esparcimiento dividirá el intervalo de frecuencia de un chirrido por dos (ya que los chirridos 2^{SF} cubren todo el ancho de banda) y también multiplicará la duración de un símbolo por dos. Sin embargo, no dividirá la tasa de bits por dos, ya que se transmitirá un bit más en cada símbolo. Además, la tasa de símbolos y la tasa de bits en un factor de esparcimiento dado son proporcionales al ancho de banda de la frecuencia, por lo que una duplicación del ancho de banda duplicará efectivamente la tasa de transmisión [7].

Duración del símbolo: La duración de símbolo es el tiempo en segundos que le toma a un símbolo en ser barrido. Se calcula con la división de 2 elevado al valor del factor de esparcimiento SF sobre el valor del ancho de banda BW . Esto se traduce en la ecuación (3.1), que vincula la duración de un símbolo (T_s) al ancho de banda y al factor de esparcimiento [33].

$$T_s = \frac{2^{SF}}{BW} \quad (3.1)$$

Además, *LoRa* incluye un código de corrección de errores hacia adelante. La tasa de codificación (CR) es igual a $4/(4+n)$, con $n \in \{1, 2, 3, 4\}$.

Tasa del símbolo: La tasa de símbolo (R_s) se calcula utilizando la ecuación (3.2):

$$\begin{aligned} R_s(\text{simbolos/seg}) &= \frac{1}{T_s} \\ &= \frac{BW}{2^{SF}} \\ &= \frac{Rc}{2^{SF}} \end{aligned} \quad (3.2)$$

Donde el ancho de banda (BW) es en Hz y el factor de esparcimiento (SF) toma valores entre 7 y 12. Podemos observar que la tasa de símbolo (R_s) es inversamente proporcional a la duración del símbolo (T_s) [7].

Por ejemplo: Si tenemos un ancho de banda $BW=125$ kHz, y un factor de esparcimiento de $SF=7$, entonces la tasa del símbolo es :

$$R_s = \frac{125000}{2^7} = 977 \text{simbolos/seg.} \quad (3.3)$$

Rendimiento de la red: Teniendo esto en cuenta, además del hecho de que los bits de información SF se transmiten por símbolo, la ecuación (3.4) permite calcular la tasa de bits útil (R_b) [33].

$$R_b = SF \times \frac{BW}{2^{SF}} \times CR \quad (3.4)$$

Por ejemplo, una configuración con $BW = 125$ kHz, $SF = 7$, $CR = 4/5$ da una tasa de bits de $R_b = 5.5$ kbps. Estos parámetros también influyen en la sensibilidad del decodificador. En términos generales, un aumento del ancho de banda reduce la sensibilidad del receptor, mientras que un aumento del factor de esparcimiento aumenta la sensibilidad del receptor. Disminuir la tasa de codificación ayuda a reducir la tasa de error de paquete (PER) en presencia de ráfagas cortas de interferencia, es decir, un paquete transmitido con una tasa de codificación de $4/8$ será más tolerante a la interferencia que una señal transmitida con una tasa de codificación de $4/5$.

Por otro lado, definimos al ancho de banda (BW) con unidades en Hertz (Hz), como el número de vibraciones o ciclos de onda por segundo. Este ancho de banda es intercambiable con la velocidad de chip. Si el ancho de banda corresponde a los 125 kHz, eso quiere decir que se tienen 125000 chips / seg.

3.3.6. Rendimiento a diferentes anchos de banda

A partir de la ecuación (3.4), se puede obtener la información de la tabla (3.3) que proporciona los diferentes valores del rendimiento del protocolo LoRa para un ancho de banda de 125kHz.

Tabla 3.3: Rendimiento.

BW	125 kHz	125 kHz	125 kHz	125 kHz	125 kHz
CR \ SF	1	$\frac{4}{5}$	$\frac{4}{6}$	$\frac{4}{7}$	$\frac{1}{2}$
7	6836bps	5469bps	4557bps	3906bps	3418bps
8	3906bps	3125bps	2604bps	2232bps	1953bps
9	2197bps	1758bps	1465bps	1256bps	1099bps
10	1221bps	977bps	814bps	698bps	610bps
11	671bps	537bps	448bps	384bps	336bps
12	366bps	293bps	244bps	209bps	183bps

3. TECNOLOGÍA LORA

Se aprecia que el valor más grande del rendimiento es posible cuando se tiene el valor más alto del CR y el más bajo de SF . El valor del rendimiento decrecerá mientras se aumente el valor del factor de esparcimiento y se disminuya el valor de la tasa de codificación. El valor mínimo del rendimiento se obtiene al utilizar el CR más pequeño y el SF más alto. Los valores para los anchos de banda de 250 kHz y 500 kHz se encuentran en el apéndice A.

La figura (3.5) presenta un panorama de comparación entre diferentes valores del factor de esparcimiento (SF). Se puede apreciar que mientras mayor sea el valor del factor de esparcimiento, mayor también es el rango de cobertura de la señal transmitida.

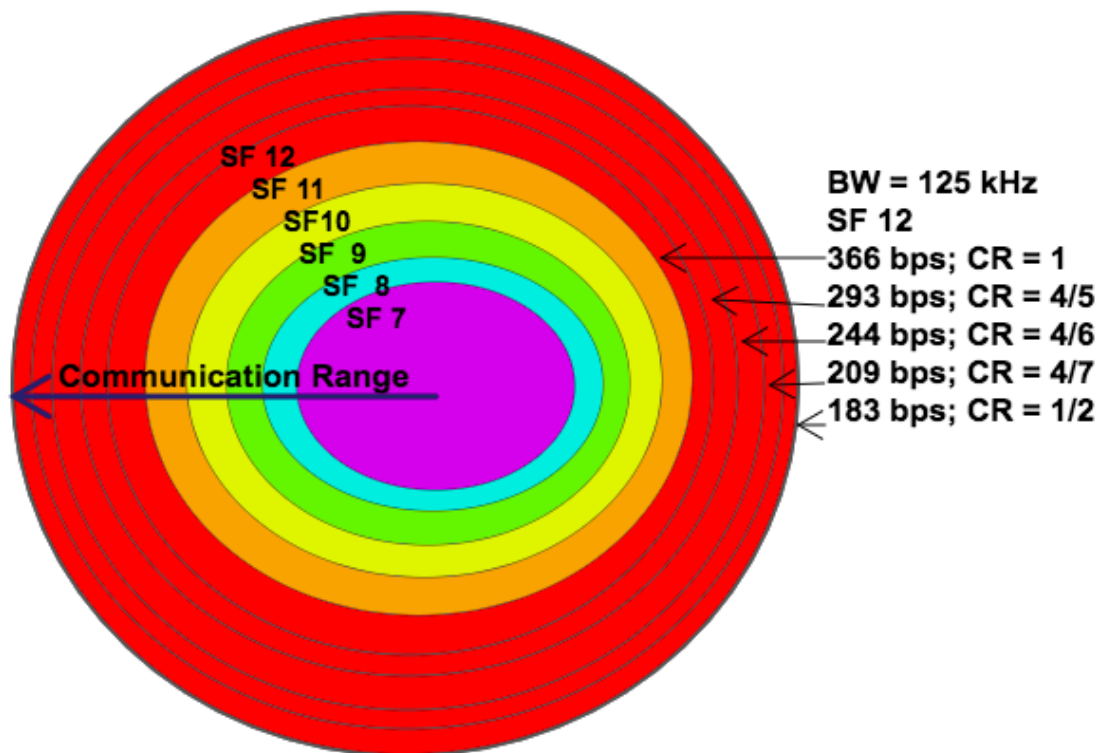


Figura 3.5: Comparación del alcance del factor de esparcimiento y del ancho de banda para obtención de diferentes valores de rendimiento.

Además, se incluyen los diferentes valores del rendimiento cuando el factor de esparcimiento tiene un valor de 12; es posible visualizar que mientras el valor de la tasa de codificación es más cercano a 1, el valor de los bits de información transmitidos es mayor.

La figura (3.6) corresponde a las curvas obtenidas a partir de la ecuación (3.4), donde se utiliza un valor de tasa de codificación y se hace variar el valor del factor de esparcimiento.

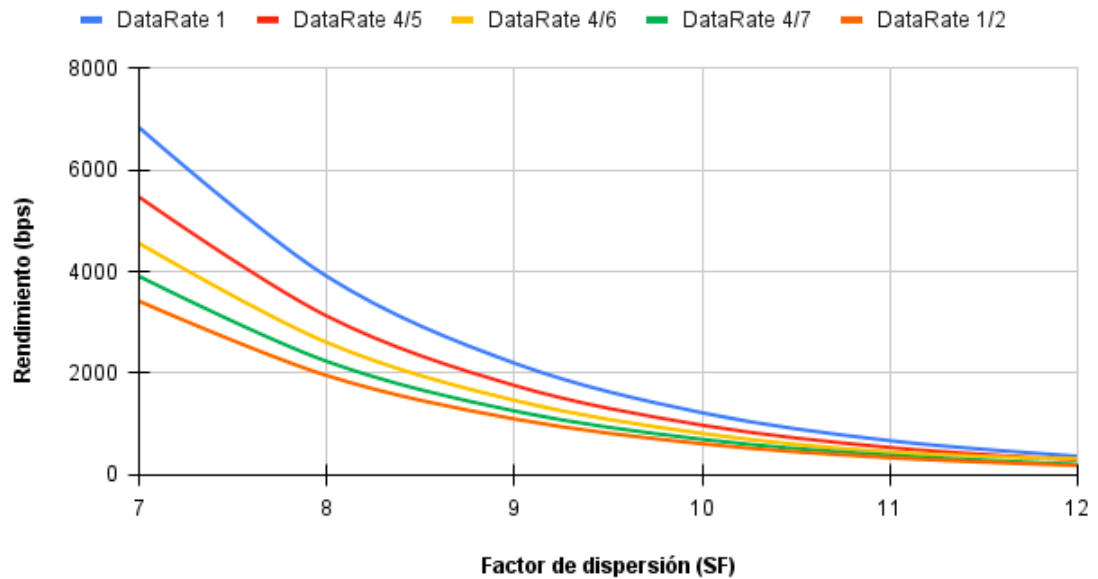


Figura 3.6: Rendimiento del protocolo LoRa utilizando un canal de 125 kHz.

Se aprecia que el máximo rendimiento, que es casi igual a los 7 kbps, se obtiene cuando el protocolo utiliza el valor de 7 en el factor de esparcimiento, el valor de 1 en la tasa de codificación y el ancho de banda igual a 125 kHz. Es importante subrayar que en la teoría podemos establecer este valor de 1 en la tasa de codificación, pero en la práctica el valor máximo a utilizar es igual a 4/5.

Posteriormente, cuanto más grande sea el valor del SF y el CR , el rendimiento disminuirá hasta llegar al menor de los casos cuando SF es igual a 12 y el CR 1/2.

La figura (3.7) presenta los diferentes comportamientos del rendimiento con los seis diferentes valores del factor de esparcimiento, los 5 de la tasa de codificación, pero para un ancho de banda de 250 kHz.

3. TECNOLOGÍA LORA

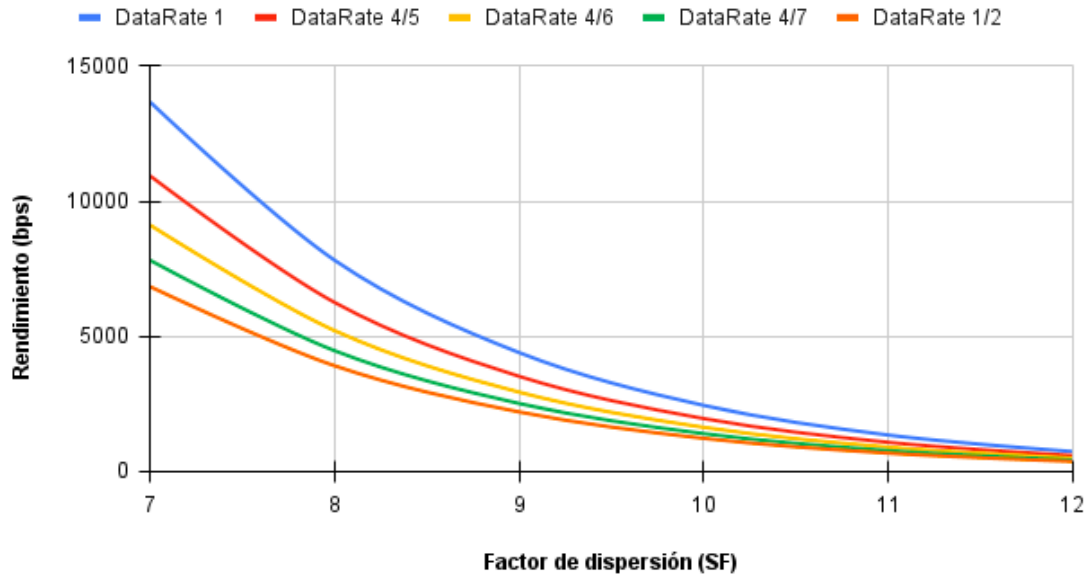


Figura 3.7: Rendimiento del protocolo LoRa utilizando un canal de 250 kHz.

Se visualiza que los máximos valores del rendimiento nuevamente se encuentran cuando el factor de esparcimiento equivale a 7 y tomando el máximo valor de la tasa de codificación.

El valor máximo obtenido utilizando este ancho de banda de 250 kHz, ahora es muy cercano a los 15 kbps. Se trata de un valor exacto de 13.6 kbps que corresponde al doble del valor del rendimiento del ancho de banda de 125 kHz cuando se usan los mismos parámetros.

La figura (3.7), proporciona el panorama de comparación de factor de dispersión contra rendimiento, pero ahora utilizando un ancho de banda de 500 kHz.

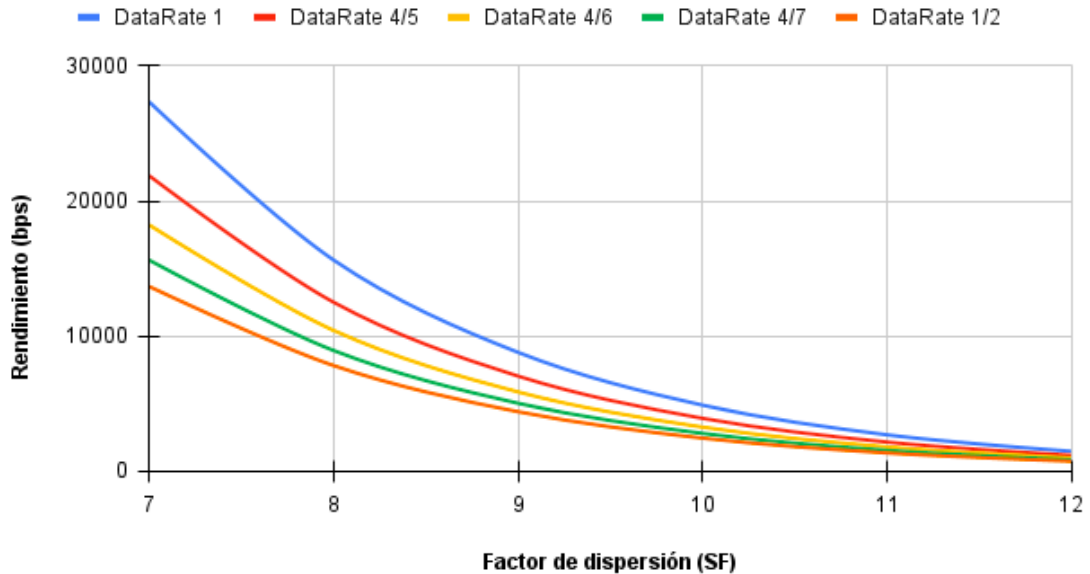


Figura 3.8: Rendimiento del protocolo LoRa utilizando un canal de 500 kHz.

Se aprecia que esta configuración proporciona los valores más altos en cuanto rendimiento se trata, otorgándonos un valor de 27.3 kbps. Recordemos que el máximo rendimiento en el caso del ancho de banda de 125 kHz (el menor de las tres configuraciones cambiando el ancho de banda) fue de 6.8 kbps, siendo la mitad del máximo rendimiento utilizando el ancho de banda de 250 kHz.

Como en las gráficas anteriores, el rendimiento presentará su valor máximo cuando se utilice un SF de 7 y un CR de 1.

3.3.7. Tiempo en el aire

El tiempo en el aire (ToA) en *LoRa* está en función de la carga útil para cada uno de los diferentes factores de esparcimiento (SF). En esta tecnología, el tiempo en el aire define el tiempo transcurrido en el aire para un paquete *LoRa* entre el dispositivo final *end device* y la puerta de enlace *gateway* [34]. El tiempo en el aire para diferentes configuraciones para cada paquete se puede calcular usando una fórmula proporcionada en las especificaciones de *LoRaWAN* [35].

Para realizar el cálculo del Tiempo en el aire comenzamos por definir la duración del símbolo T_S a partir de la ecuación (3.1).

3. TECNOLOGÍA LORA

En la figura (3.9) se muestra la estructura del paquete que utiliza la tecnología *LoRa*.

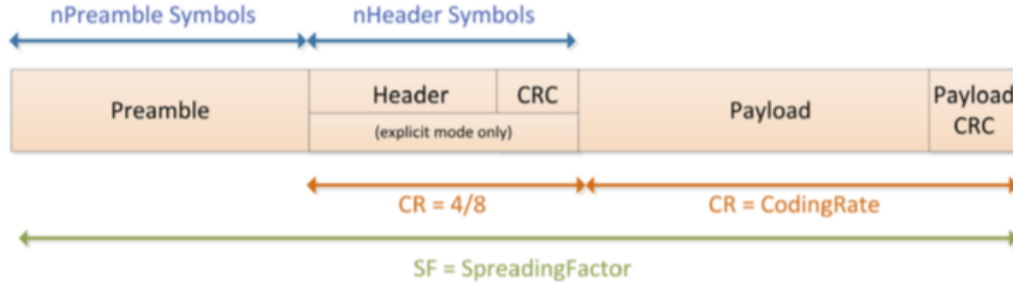


Figura 3.9: Estructura del frame de LoRa [7].

La duración del preámbulo $T_{preamble}$ depende del número de símbolos de preámbulo programado $n_{preamble}$, así como de la duración del símbolo T_S . La ecuación (3.5) muestra esta relación.

$$T_{preamble} = (n_{preamble} + 4.25)T_S \quad (3.5)$$

Por otro lado, el número de símbolos que conforman la carga útil del paquete y su cabecera $payloadSymbNb$ es igual a la ecuación (3.6).

$$payloadSymbNb = 8 + \max\left(\text{ceil}\left(\frac{8PL - 4SF + 28 + 16CRC - 20H}{4(SF - 2DE)}\right)(CR + 4), 0\right) \quad (3.6)$$

Donde PL es el número de carga útil en bytes, SF es el valor del factor de esparcimiento. H toma el valor de 0 cuando el encabezado está habilitado y de 1 cuando no lo está. DE vale 1 cuando la optimización de baja velocidad de datos está habilitada y vale 0 cuando está deshabilitada. CR es la tasa de codificación de 1 a 4. CRC hace referencia a la verificación de redundancia cíclica y solo está presente en transmisiones de enlace ascendente.

Ceil es una función que redondea un número al próximo múltiplo del argumento cifra significativa que en este caso vale 0, hacia arriba. La longitud del preámbulo es configurable; para el dispositivo utilizado en este estudio, este valor es $n = 8$ símbolos. De ello se deduce que si el tiempo en el aire requiere una reducción y la longitud del paquete se conoce de antemano, entonces se puede eliminar la información del encabezado H [7].

La duración de la carga útil $T_{payload}$ es entonces el período de símbolo T_S multiplicado por el número de símbolos de carga útil $payloadSymbNb$. La ecuación (3.7) nos muestra esta relación.

$$T_{payload} = payloadSymbNb \times T_S \quad (3.7)$$

La duración de la carga útil depende de la cantidad de bytes que tendrá, de la redundancia cíclica, de la cabecera y de si se habilita la optimización de baja velocidad de datos.

El tiempo en el aire, o la duración del paquete es simplemente la suma de la duración del preámbulo $T_{preamble}$ y la duración de la carga útil $T_{payload}$ como se muestra en la ecuación (3.8) [35].

$$ToA = T_{preamble} + T_{payload} \quad (3.8)$$

3.3.8. Sensibilidad

Las cifras de la tabla (3.4) que muestran la sensibilidad del receptor (tomadas de la hoja de datos SX1276), se presentan a continuación.

Tabla 3.4: Sensibilidad del receptor Semtech SX1276 LoRa en dBm a diferentes anchos de banda y factores de esparcimiento [10].

SF BW	7	8	9	10	11	12
125 kHz	-123	-126	-129	-132	-133	-136
250 kHz	-120	-123	-125	-128	-130	-133
500 kHz	-116	-119	-122	-125	-128	-130

3.3.9. Símbolo en LoRa

Un símbolo representa uno o más bits de datos, por ejemplo: Símbolo = 1011111 (decimal=95). En el ejemplo anterior, el número de bits sin procesar

que puede codificar el símbolo es 7. Esto es lo mismo que decir: Factor de esparcimiento $SF=7$ El símbolo tiene 2 a la SF valores. Si $SF = 7$, los valores van de 0 a 127. El valor del símbolo se codifica en una señal de barrido (*up-chirp*). La señal de barrido se divide en pasos o chips de 2 elevado al valor de SF [36].

Por ejemplo, si el símbolo es 1011111 (95 en decimal), el número de bits sin procesar que puede codificar este símbolo es 7 ($SF = 7$). La señal de barrido se divide en 2 elevado al valor de $SF = 27$ para un total de 128 chips

Otro ejemplo, supongamos que $SF = 12$ entonces cada símbolo puede transportar 12 bits de información sin procesar y hay $2^{12} = 4096$ valores de chip únicos que van de 0 a 4095.

Nota: Hay que tener en cuenta la diferencia entre un *chirp* y un chip. Un símbolo contiene un número de chips de 2 elevado al valor de SF. Los *chirps* son simplemente una rampa desde el flujo hacia arriba (*chirp* ascendente) o desde el flujo hacia arriba (*chirp* descendente).

3.4. Modelo de propagación del proyecto EWIN

El modelo de propagación para ríos del proyecto *EWIN* es una contribución científica atribuida a investigadores mexicanos que por medio de pruebas en campo pudieron obtener un modelo matemático que ahora es una de las bases de este proyecto.

Como se trata de un modelo de propagación recientemente desarrollado, fue necesario realizar la implementación del mismo en el software *NS-3*.

Es importante mencionar que el *Path Loss (PL)* cambia en un punto determinado a lo largo de la ruta que sigue el río. Es decir, tenemos una ecuación de *path loss* tanto para el lado derecho como para el izquierdo. La figura(3.10) muestra el punto exacto del cambio en la ecuación de *path loss* de este escenario. Las flechas indican la correspondencia de los sentidos que tiene este fenómeno.

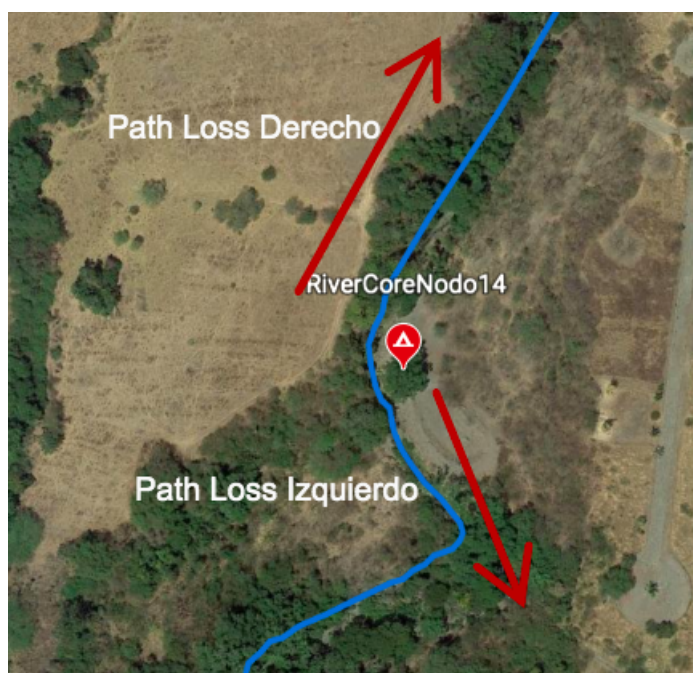


Figura 3.10: Punto de la ruta del río donde el path loss cambia.

Se corrobora que este punto se encuentra en la misma posición que uno de los *gateway* de LoRa que se han montado en el escenario real y que lleva por nombre *RiverCore 14*.

La tabla (3.5) muestra los valores de *path loss*, tanto del lado derecho como del izquierdo. Estas ecuaciones fueron obtenidas por científicos que utilizaron señales que se propagaban a lo largo del río. Se procedió a la implementación de este nuevo modelo de *path loss* para ríos, en el módulo *LoRa*, tomando en cuenta las ecuaciones de la tabla anterior.

Tabla 3.5: Path loss lado derecho e izquierdo [8].

Lado	D(m)	Pathloss model
Derecho	24.2-194	$50 + 42 \cdot \log_{10}(d/24.2) + 4.3$
Izquierdo	17.2 - 124	$52 + 31 \cdot \log_{10}(d/17.2) + 3.2$

Además de ello, se agregó el efecto *shadow fading* al módulo para tener resultados de un escenario más realista. Es decir, la degradación de la intensidad de la señal tiende a aumentar debido a obstáculos entre el transmisor y el receptor a lo largo de su propagación.

3. TECNOLOGÍA LORA

La ecuación (3.9) muestra el valor de *shadow fading* (σ) que se tomó en cuenta para el desarrollo del módulo.

$$\sigma = \mu_{\sigma} + z\sigma_{\sigma} \quad (3.9)$$

Donde μ_{σ} es la media, z es una variable aleatoria con media en cero y σ_{σ} es la desviación estándar.

Este valor de degradación depende de las condiciones del terreno donde se encuentre montado cada escenario. Para nuestro caso, el valor de la media fue de 8.2 y la desviación estándar de 3. Estos valores fueron obtenidos a partir de la publicación [37], ya que se trata de una categoría de terreno donde se ubica una densidad considerable de vegetación como árboles y plantas.

En cuanto al valor de z , se generó una variable aleatoria. De esta manera se tiene un modelo más realista.

Descripción del modelo de simulación de LoRa en Network Simulator 3 (NS-3)

La simulación de red emplea modelos representativos basados en conceptos y paradigmas de redes de comunicación del mundo real para reproducir comportamientos y estadísticas realistas. Su objetivo es lograr esta tarea sin los costos requeridos de construir bancos de pruebas equivalentes con hardware o componentes muy costosos.

Por consiguiente, en este capítulo se muestra la definición de sistemas de eventos discretos, una descripción del simulador utilizado donde se abordan sus características y la estructura básica del simulador. Además, se describe por completo el diseño del módulo *LoRa*, que es la principal aportación de esta tesis. Finalmente, se da una explicación de las características probabilísticas de las boyas o *end devices*.

4.1. Sistema de eventos discretos

Un sistema es un conjunto de componentes que interactúan y que se comportan juntos para realizar una función y esta función no puede ser realizada por ninguna de las partes individuales [38]. En un momento específico, el comportamiento de un sistema se puede describir de una manera mensurable, es decir, un estado. El espacio de estados de un sistema es un conjunto de todos los valores posibles que puede tomar un estado. Según el tipo de estados de un modelo, un sistema puede clasificarse en un sistema de estado continuo o un sistema de estado discreto [39].

4. DESCRIPCIÓN DEL MODELO DE SIMULACIÓN DE LORA EN NETWORK SIMULATOR 3 (NS-3)

Ahora bien, un Sistema de Eventos Discretos (DES) es un sistema dinámico que evoluciona de acuerdo con la ocurrencia de eventos físicos, a intervalos irregulares posiblemente desconocidos. Estos sistemas surgen en una variedad de contextos que van desde sistemas operativos de computadora hasta el control de procesos complejos. En un sistema de estado discreto, los estados solo pueden cambiar de un valor de estado discreto a otro. Un evento puede ocurrir instantáneamente y causar transiciones de estado [40].

NS-3 es un simulador de red de eventos discretos, dirigido principalmente para investigación y uso educativo. *NS3* es software libre, con licencia GNU GPLv2, y está disponible públicamente para investigación, desarrollo y uso. Gracias a esto se puede desarrollar un entorno de simulación de código abierto, gratuito y adecuado para la investigación de redes, en nuestro caso usando un módulo *LoRa* que fue adaptado pensando en nuestro escenario en específico.

4.1.1. Características del simulador NS-3

NS-3 emplea un planificador de eventos discretos para permitir a los investigadores y educadores construir topologías de red, generar tráfico para atravesar la red simulada y recopilar estadísticas basadas en los efectos de manipular las características de la red. Esta herramienta opera por medio del envío y recepción de paquetes, procesándolos individualmente como entidades distintas a medida que se mueven a través de una red [41].

El núcleo del software está diseñado para mejorar la escalabilidad, la modularidad, el estilo de codificación y la documentación, el núcleo está escrito en C++ pero con una interfaz de script Python. Se aprovechan varios patrones de diseño de C++, como punteros inteligentes, plantillas, devoluciones de llamada y copia en escritura. Las capacidades de agregación de objetos permiten extensiones de paquetes y modelos más fáciles.

Otra característica importante es que el software admite la incorporación de más software de redes de código abierto, como pilas de protocolos de kernel y analizadores de seguimiento de paquetes, lo que reduce la necesidad de transferir o reescribir modelos y herramientas para la simulación.

NS-3 proporciona un banco de pruebas experimental con pilas de protocolos para emitir paquetes de red a través de controladores de dispositivos reales. La representación interna de los paquetes está ordenada por bytes de red para facilitar la serialización.

El simulador además cuenta con métodos para rastreo de paquetes y recopilación de estadísticas. Para esto se utiliza un diseño basado en devolución de llamada, y esto permite el rastreo de los paquetes para proporcionar estadísticas de un determinado escenario [42].

4.2. Diseño del módulo LoRa

Este módulo comprende dos modelos principales: uno para la capa física de LoRa que necesita representar los *chips LoRa* y el comportamiento de las transmisiones *LoRa*, y otro para la capa *LoRa river MAC*, que debe comportarse de acuerdo con las especificaciones oficiales y las demandadas por el proyecto. La figura (4.1) muestra la pila de *LoRa* de este módulo.

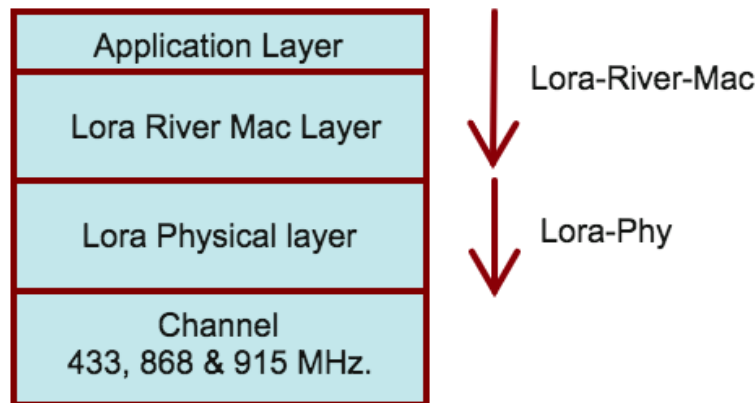


Figura 4.1: Pila de protocolos de LoRa.

Para representar estos dos modelos, el módulo presenta dos clases genéricas *LoraPhy* y *LoraRiverMac*. Estas clases se extienden luego por clases que modelan las peculiaridades de los dos dispositivos de red inalámbrica: el dispositivo final que hemos llamado *end device mac* y la puerta de enlace que en esta tesis es nombrado *river core mac*. Por lo tanto, la capa física puede ser modelada mediante el uso de la clase *LoraPhy*, mientras que los objetos de la clase *EndDeviceMac*, se utilizan para representar la capa *MAC*.

4.2.1. Modelo de la capa física

El modelo para la capa física debe tener en cuenta los dos factores clave de LoRa:

- La sensibilidad
- La ortogonalidad

Esto para decidir si una transmisión se recibe correctamente o no. Además, también debe ser consciente de cómo funcionan los chips que implementan la modulación y de su arquitectura.

En esta parte se realizó una de las más importantes contribuciones, ya que se determina la probabilidad del error de cada paquete que recibe la boya o *end device*. El valor del error del paquete se ha obtenido a partir de un módulo de error que se ha diseñado e implementado. El módulo, como primera fase obtiene justamente la probabilidad de error de cada bit de los paquetes que se envían. Para este cometido, se ha utilizando una matriz donde se encuentran los valores de la probabilidad antes mencionada. Dichos valores han sido calculados a partir de la aproximación de la publicación [24], misma que para su realización toma en cuenta al valor del factor de esparcimiento y la relación señal-ruido.

Más adelante, cuando se tiene la probabilidad de bit en error (P_{EB}), se utiliza la ecuación (4.1) para calcular la probabilidad de error del paquete (P_{EP}).

$$P_{EP} = 1 - (1 - P_{EB})^{PS} \quad (4.1)$$

Donde P_{EB} es la probabilidad de bit en error y PS es el tamaño del paquete en bits.

Posteriormente con el fin de tener un resultado mucho más realista del módulo, cuando se tiene el valor de la probabilidad de error del paquete (P_{EP}) se compara con un número aleatorio obtenido de una variable aleatoria con distribución uniforme. Si el valor del número aleatorio es menor o igual a de la probabilidad de error del paquete se asume que el paquete no se ha recibido correctamente, es decir, el paquete tiene un error.

4.3. Resultados: Time On Air

Se han obtenido diversos resultados a partir del módulo *LoRa*. Como primera validación, haremos una prueba del trabajo desarrollado. Se han tomado paquetes de transmisión de 24 bytes y paquetes de confirmación *ACK* de 5 bytes. Para conocer el tiempo en el aire, se recurre a la ecuación (3.8) que proporcionará este cálculo tanto de los paquetes de transmisión así como los paquetes de confirmación.

El escenario utilizado de esta simulación se conforma de una sola boya y un *river core* separados a 300 metros de distancia. La boya tiene movilidad y sigue una línea recta de tal manera que se acerca al *river core* con una velocidad de un metro sobre segundo. Como se aprecia, es un escenario muy sencillo pues la finalidad de este primer experimento es conocer el tiempo en el aire de un solo paquete de información.

4.3.1. Características del escenario para paquetes de 24 bytes

La tabla (4.1) nos muestra los valores de la estructura del *frame* utilizados del escenario de simulación para un paquete de información de tamaño de 24 bytes.

Tabla 4.1: Características del escenario de simulación.

Parámetro	Valor
n_{pream}	8
$CRC[0,1]$	1
Header H	0
$DE [0,1]$	0
SF	7
CR	1
Bw	125000 Hertz

4.3.2. Resultados teóricos y simulados para tamaño de carga útil de 24 bytes

Los resultados mostrados en la tabla (4.2) corresponden a los obtenidos después de la simulación y además después de utilizar las ecuaciones de la duración del símbolo T_S a partir de la tabla (3.1), la duración del preámbulo T_{pream} (3.5) y la duración de la carga útil $T_{payload}$ (3.7).

Tabla 4.2: Resultados teóricos para tamaño de carga útil de 24 bytes.

Parámetro	Teoría	Simulación
T_S [ms]	1.024	1.024
T_{pream} [ms]	12.544	12.544
$T_{payload}$ [ms]	49.152	49.152
T_{oA} [ms]	61.696	61.696

Se puede constatar a partir de todos los cálculos que el tiempo en el aire del paquete de transmisión de tamaño de 24 bytes es igual a 61.696 ms tanto en la simulación como en la teoría.

4.3.3. Resultados teóricos de tiempo en el aire para paquete de 24 bytes

La figura (4.2) muestra el comportamiento del tiempo en el aire de los paquetes de 24 bytes con las características de la tabla (4.2). Se aprecia que el valor del factor de esparcimiento varía entre sus posibles valores.

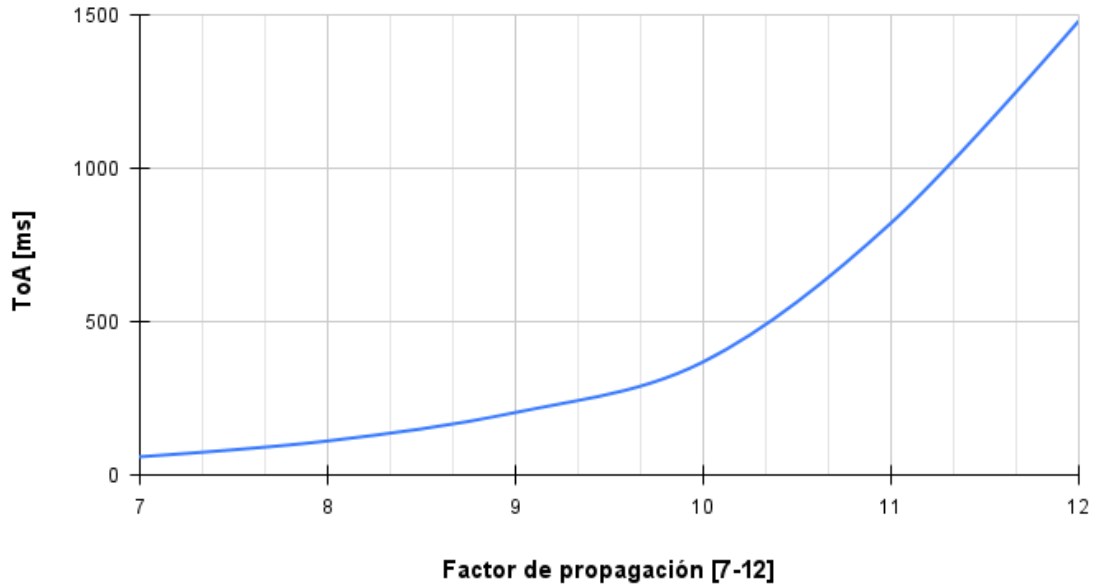


Figura 4.2: Tiempo en el aire de paquetes de transmisión de 24 bytes en LoRa.

Se visualiza que el tiempo en el aire del paquete de información crece a medida que el valor del factor de esparcimiento aumenta también. Se percibe aquí una de las características más importantes del factor de esparcimiento: que es el hecho de que cada que se aumenta en una unidad, el tiempo en el aire tiende a doblar su valor. El valor máximo de ToA obtenido para un escenario con estas características se presenta cuando el valor del SF es igual a 12, y el resultado es casi igual a los 1500ms.

4.3.4. Características del escenario para paquetes de 5 bytes

La tabla (4.3) muestra los valores de los parámetros del *frame* utilizados del escenario de simulación para un *ACK* de tamaño de 5 bytes.

4. DESCRIPCIÓN DEL MODELO DE SIMULACIÓN DE LORA EN NETWORK SIMULATOR 3 (NS-3)

Tabla 4.3: Características del escenario de simulación.

Parámetro	Valor
n_{pream}	8
$CRC[0,1]$	1
Header H	0
$DE [0,1]$	0
SF	7
CR	1
Bw	125000

4.3.5. Resultados teóricos y simulados para tamaño de carga útil de 5 bytes

La tabla (4.4) muestra el resultado del uso de las ecuaciones de la duración del preámbulo $T_{preamble}$ (3.5) y la duración de la carga útil $T_{payload}$ (3.7). En este caso se usaron paquetes de confirmación, es decir, la boya los recibió.

Tabla 4.4: Resultados teóricos para tamaño de carga útil de 5 bytes.

Parámetro	Teoría	Simulación
T_S [ms]	1.024	1.024
T_{pream} [ms]	12.544	12.544
$T_{payload}$ [ms]	18.423	18.423
ToA [ms]	30.976	30.976

Se nota que ahora el tiempo en el aire es menor comparado con el del caso de los paquetes de 24 bytes, ya que el tamaño de la carga útil del paquete también lo es y en esta ocasión es igual 5 bytes. A partir de toda la información se obtiene un valor de 30.976 ms en el ToA .

El valor del tiempo en el aire en la tecnología *LoRa* depende por lo tanto del tamaño de la carga útil del paquete, pues mientras mayor sea este valor mayor también será el valor del ToA .

4.3.6. Resultados teóricos de tiempo en el aire para paquete de 5 bytes

La figura (4.3) presenta los diferentes valores obtenidos al variar el valor del factor de esparcimiento entre 7 y 12 cuando se hace el cálculo del ToA .

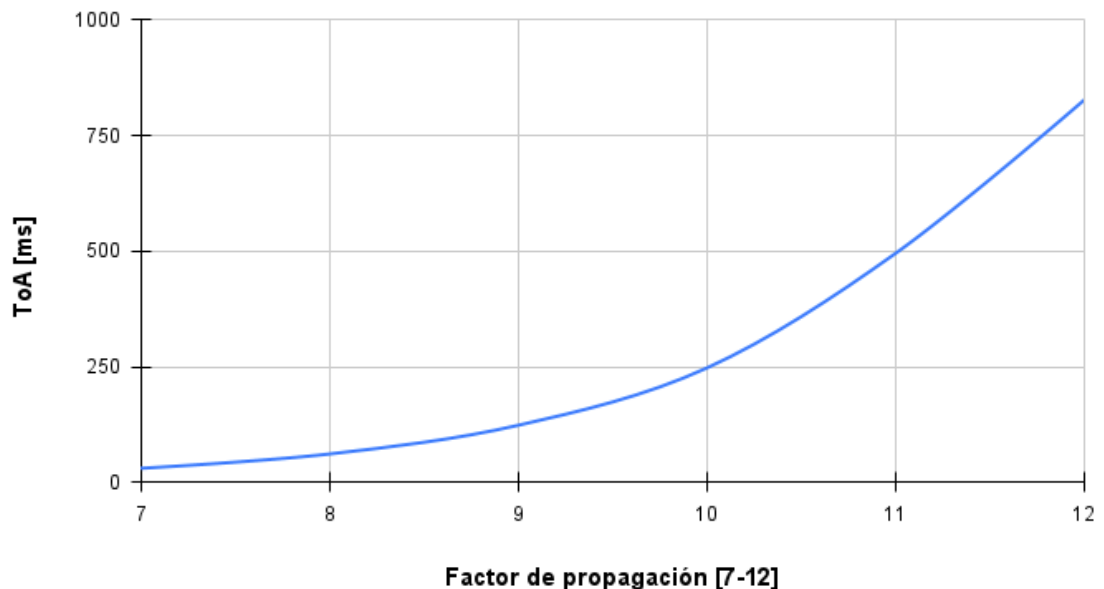


Figura 4.3: Tiempo en el aire de paquetes de confirmación de 5 bytes en LoRa.

En esta ocasión podemos visualizar que el ToA del paquete de tamaño de 5 bytes es menor a comparación de un paquete de 24 bytes, esto cuando se comparan los escenarios con valores idénticos de factor de esparcimiento. Como en la figura anterior, podemos notar el efecto del factor de esparcimiento sobre el tiempo en el aire del paquete: se dobla el valor del tiempo en el aire cuando se aumenta el valor del factor de esparcimiento una unidad. La figura (4.3) nos proporciona un mínimo valor de ToA de 30.976 ms para el menor de los valores de SF y de 827.3 ms para el mayor que corresponde a 12.

4.4. Modelo de la capa MAC

Los modelos de control de acceso al medio contenidos en este módulo tienen como objetivo implementar una versión *MAC* diseñada para ríos. El desarrollo y la implementación de la capa de control de acceso al medio es la contribución más importante de este trabajo de tesis, ya que se trata de un algoritmo diseñado exclusivamente para escenarios como este que están montados completamente en ríos.

La capa *MAC* tiene diferentes objetivos, entre los cuales se encuentran el hecho de asignar un ID a cada paquete generado y enviado de un dispositivo a otro. Esto con el fin de tener conocimiento de los paquetes recibidos correctamente.

Además, en esta capa se puede identificar el tipo de mensaje. Este puede ser de uno de los siguientes tipos:

- Información (DATA).
- Confirmación (ACK).

También, la capa *MAC* se encarga de la identificación del tipo de dispositivo. El cual puede tratarse de:

- Boya o *end device*.
- Puerta de enlace o *river core*.

4.4.1. Capacidad de calcular la distancia entre una boya y un river core

Una de las funciones de la capa *MAC*, es el hecho de que es capaz de calcular la distancia entre una boya y un *river core*. Para poder calcular la distancia con cada *river core*, es necesario que la boya conozca previamente las coordenadas geográficas de cada uno de ellos. La figura (4.4) muestra el rango de transmisión de cada una de las boyas de las simulaciones de esta tesis. Se aprecia que se tiene un rango de transmisión de 65 metros, esto con el objetivo de tener una alta probabilidad de transmisiones exitosas de paquetes entre la boya y el *river core*.

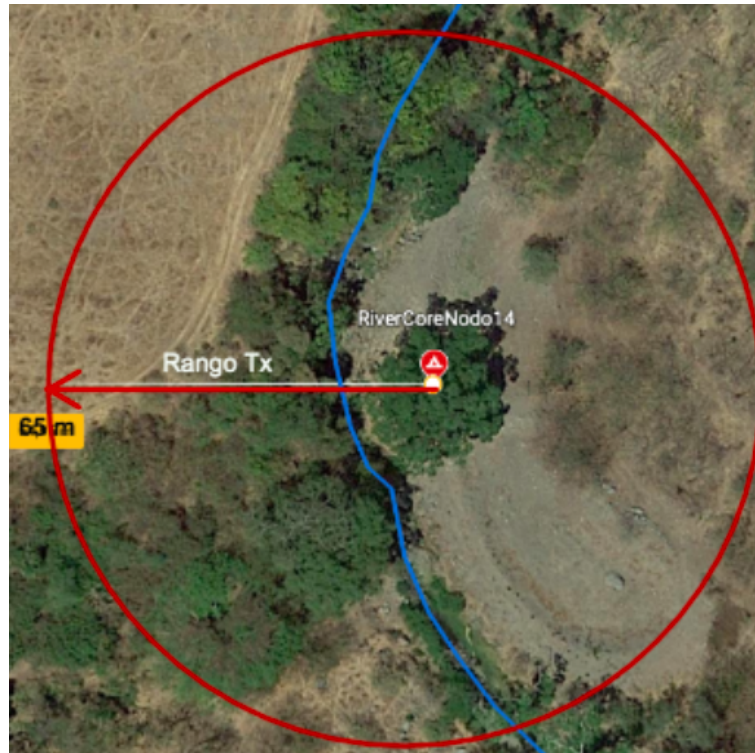


Figura 4.4: Rango de transmisión de la boya.

Primero se valida la posición geográfica de la boya (proceso que en la implementación física se realiza por medio del GPS) y posteriormente se realiza el cálculo de la distancia entre ella y cada uno de los *river core*.

Esto debido a que las posiciones geográficas de las compuertas de enlace son conocidas ya que se trata de dispositivos con posiciones geográficas invariables. Este proceso se realiza en la generación de cada paquete.

4.4.2. Planificación del envío de paquetes

Esta función está desarrollada e implementada a la medida para escenarios como el nuestro donde una planificación resulta vital para el envío exitoso de paquetes entre dispositivos. Se tiene como objetivo que la boya, al estar cada vez más cerca de un *river core*, tenga la capacidad de despertarse en periodos cada vez más cortos de tiempo para realizar de manera oportuna la transmisión de información.

4. DESCRIPCIÓN DEL MODELO DE SIMULACIÓN DE LORA EN NETWORK SIMULATOR 3 (NS-3)

La figura (4.5) muestra tanto el área de transmisión Tx como el área de planificación que las boyas conocen por medio del GPS.



Figura 4.5: Rango de planificación de la boya.

Ambas áreas están delimitadas por una circunferencia, la mayor corresponde al área de planificación y la menor a la de transmisión. Se visualiza una boya A que se encuentra atravesando el área de transmisión y por lo tanto enviando paquetes y recibiendo acuses del *river core*. La boya B conforme avanza, se encuentra despertando en periodos cada vez más cortos de tiempo pues esta se encuentra recorriendo el área de planificación. La boya C que se encuentra fuera de ambas áreas cambia de un estado a otro normalmente.

La planificación de transmisión depende completamente de la distancia entre dispositivo receptor y emisor. Ya que cuando se tiene una distancia muy alta, el módulo calculará un tiempo futuro en el cual volverá a verificar su posición. Este tiempo de planificación es muy grande cuando las distancias entre boya y *river core* es muy amplia, y tiene un valor cada vez más pequeño a medida que la boya se acerca al *river core*.

Este tiene el objetivo de despertar a la boya y realizar la transmisión de paquetes cuando se encuentre muy cerca de la puerta de enlace, para así evitar que la boya se despierte cuando haya pasado de largo y dejado atrás al *river core* y al área de transmisión. De esta manera aseguramos una transmisión más exitosa entre dispositivos. Posteriormente al envío de los paquetes, cuando la boya sale del rango de comunicación o se queda sin paquetes a enviar, ésta pasa automáticamente a un estado de reposo o *standby*.

4.4.3. Etiquetado de mensajes transmitidos

En esta capa *MAC*, se han definido dos tipos de mensaje; tanto los de información (*DATA*) como los de confirmación (*ACK*). El módulo *LoRa* es capaz de asignar a cada paquete de información el tipo de mensaje del que se trata entre estas dos opciones. Esto lo realiza antes de la transmisión, etiquetando a cada paquete con un número único. El objetivo es conocer cuales paquetes de información han sido recibidos correctamente durante la transmisión, por medio del etiquetado.

4.4.4. Acuse de los paquetes enviados

El acuse de los paquetes enviados entre dispositivos ha sido una aplicación implementada también en el módulo *LoRa*. Cuando el *river core* recibe un paquete, envía otro a la boya pero ahora de acuse (este paquete es de menor tamaño).

La figura (4.6) muestra el proceso de la boya al enviar paquetes de información hacia el *river core*. Cuando la boya recibe los paquetes que la puerta de enlace envía, valida cual es el número de ID que éste tiene. Si encuentra que se trata de un paquete cuyo mensaje de información ha sido enviado anteriormente, el dispositivo cesa de enviar el paquete con ese ID y lo elimina de su memoria. En caso contrario, se realiza una retransmisión del paquete con el ID correspondiente. Esta retransmisión ocurre un máximo de 5 intentos y cuando no se haya recibido acuse del paquete con el ID deseado. Es importante resaltar que la retransmisión también depende de la distancia entre el dispositivo emisor y receptor, debido a que este proceso solo ocurre dentro del área de transmisión.

4. DESCRIPCIÓN DEL MODELO DE SIMULACIÓN DE LORA EN NETWORK SIMULATOR 3 (NS-3)

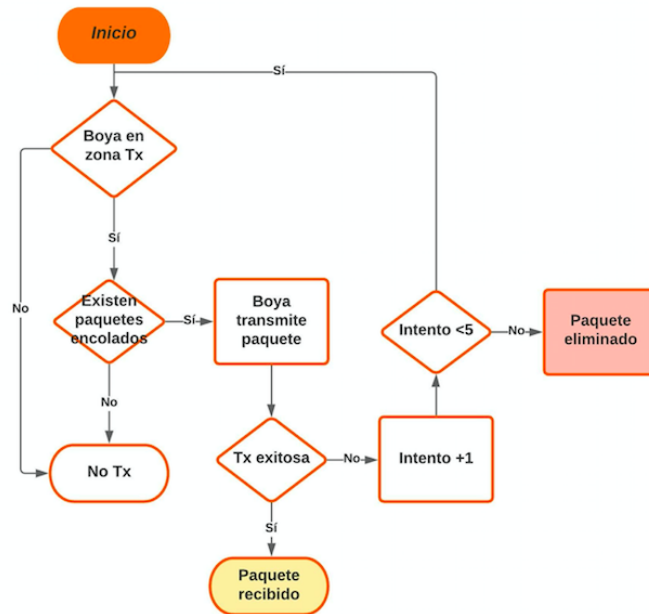


Figura 4.6: Diagrama de flujo del envío de paquetes.

Se observa a partir del diagrama de flujo de la figura (4.6) que una transmisión exitosa implica la recepción del paquete al *river core*. Por otra parte, si se exceden de los cinco intentos de retransmisión, el dispositivo deja de retransmitir el paquete y lo elimina.

La figura (4.7) muestra la transmisión de paquetes de información de la boya al *river core*, y la respuesta del *river core* quien envía un paquete de acuse *ACK* hacia la boya que sigue dentro de la zona de transmisión.

Además, es en esta parte donde se encuentra la implementación de la retroalimentación del algoritmo de retroceso exponencial (*Exponential backoff*). Este algoritmo depende tanto de la media como del límite. Y estos números dependen del número de retransmisiones del paquete. Cuando se tiene una retransmisión, se tiene un valor de media de 4 y un límite de 8. En cambio cuando se tiene un número de retransmisiones igual a dos, la media vale 8 y el límite 16. Hasta tener un valor máximo de la media de 64 y un límite de 128.

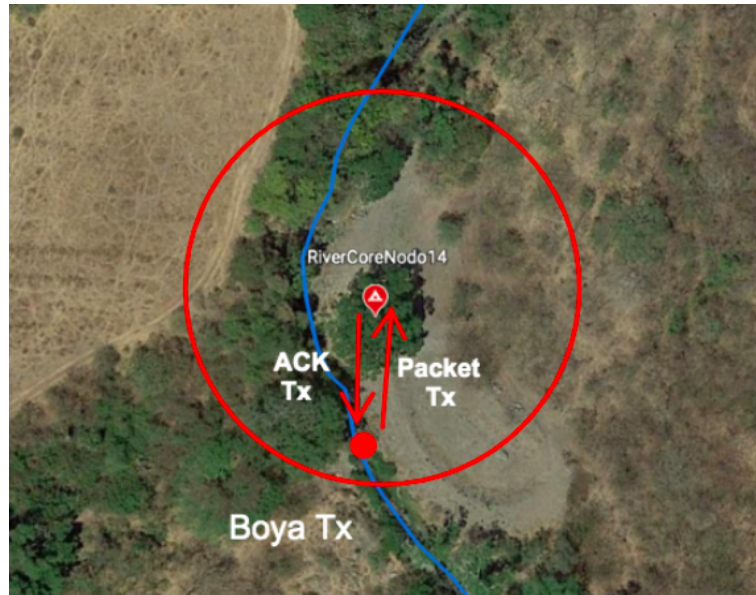


Figura 4.7: Acuse de los paquetes en el área de transmisión.

A partir de estos dos datos, se obtiene un número aleatorio de una distribución exponencial con un valor de entre 0 y el límite. El número aleatorio se multiplica por el tiempo que tarda el módulo *LoRa* en completar todo un ciclo de trabajo y es este el tiempo en el que vuelve a realizar la retransmisión. Esto con el objetivo de evitar colisiones entre paquetes.

4.4.5. Retransmisión de paquetes

El módulo *LoRa* tiene un límite de hasta 5 retransmisiones cuando el paquete no se acusa por medio de un *ACK*. En el caso contrario, cuando se acusa el paquete, el módulo borra de su memoria al mismo y continua con la transmisión o retransmisión de los restantes en caso de ser necesario.

El módulo siempre verifica la distancia entre el receptor y transmisor, y cuando es mayor del límite establecido por el usuario, simplemente no realiza ninguna retransmisión. Cuando las retransmisiones terminan de efectuarse, la boya pasa a un estado de reposo o *standby*.

4.4.6. Encabezados MAC

La estructura de paquetes está definida por las características de la red, y esta se implementa a través de la clase *Header* que deriva para un mejor control a su subclase *LoraRiverCoreMacHeader*. En particular, esta se encarga de serializar y deserializar los identificadores de cada paquete que han sido configurados. En la figura (4.8) se muestran los encabezados *MAC* diseñados para nuestra aplicación.

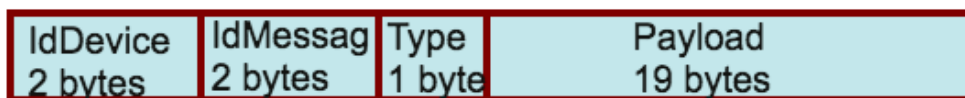


Figura 4.8: Encabezados MAC de los paquetes.

Se definen 5 bytes en el *header* que corresponden a los siguientes conceptos que se muestran en la figura (4.8); el formato de un mensaje *MAC* comprende a los siguientes campos:

- *IdDevice*: Identificador único de cada dispositivo. Así cada dispositivo de la red, ya sea un *river core mac* o un *end device mac*, debe tener uno diferente. Se pueden tener hasta $2^8 * 2^8 = 65536$ dispositivos en una red de este tipo.
- *IdMessage*: Asigna un identificador diferente a cada paquete. Este es utilizado para el control de flujo de paquetes. Su tarea es validar qué mensajes ya se han recibido y acusar los mensajes. Una vez que el contador de mensajes llegue a su máximo valor, regresará a cero.
- *Type*: determina si los paquetes son de información o corresponden a confirmaciones *ACK*'s.
- La carga útil: Transporta datos de la aplicación o un mensaje de incorporación.

4.5. Características de las boyas

Además del diseño de la capa *MAC* en los dispositivos *LoRa*, se tomaron en cuenta ciertas características para hacer más realista el módulo desarrollado en esta tesis y así tener un escenario muy similar al del proyecto *EWIN*. A continuación se detallan las implementaciones del modelado del efecto de variación de velocidad durante el trayecto, el modelado del efecto de inversión de las boyas durante el trayecto y el modelado del efecto de estancamiento de las boyas.

4.5.1. Modelado del efecto de variación de velocidad durante el trayecto

El mapa de la figura (4.9) representa las diferentes áreas del río Colima en tres diferentes colores con diferentes características en cuanto a profundidad en metros de las secciones, y como consecuencia se tiene una variación de velocidad de la corriente del río.

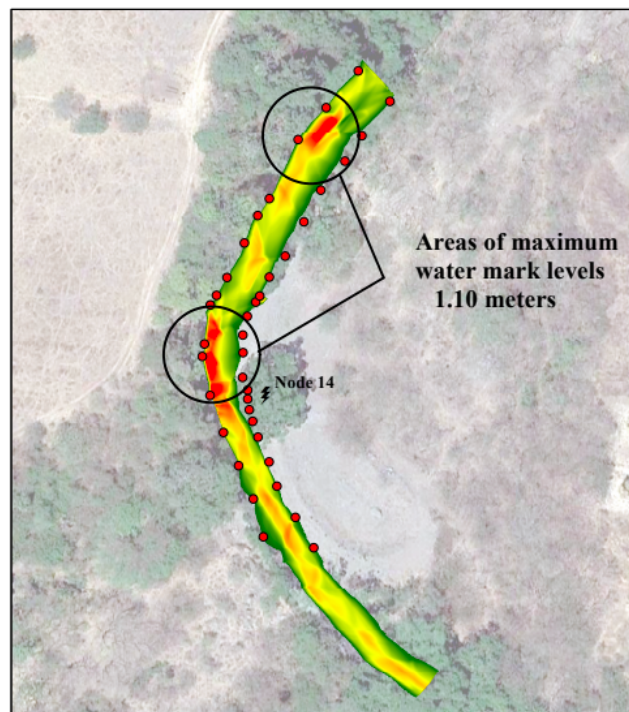


Figura 4.9: Áreas y probabilidades de hundimiento [8].

4. DESCRIPCIÓN DEL MODELO DE SIMULACIÓN DE LORA EN NETWORK SIMULATOR 3 (NS-3)

Esta variación dependiente de la profundidad y puede interpretarse de la siguiente manera:

- En verde se representan las áreas del río donde la corriente de agua tiene una velocidad muy baja debido a la profundidad muy pequeña .
- Las secciones amarillas, que cubren la mayoría del espacio, corresponde a la corriente con velocidades más grandes; esto debido a que la profundidad es más grande.
- Por último, en las secciones rojas del río se tienen profundidades de hasta 1.10 metros. Esto provoca que la corriente aumente su velocidad en estos tramos.

De esta manera, el mapa muestra la profundidad del río y como consecuencia la velocidad de la corriente en cada una de estas partes. A partir de esta información, se hizo la transición del movimiento a una probabilidad de hundimiento.

Para realizar el traslado del fenómeno físico al simulador primero se revisan los parámetros y características de la posición geográfica del dispositivo. Pues cuando las velocidades de las corrientes son más grandes, se tiene una probabilidad alta de pérdida de paquetes. De esta manera se le asigna una probabilidad de pérdida de paquetes a la boya dependiendo de su posición.

Como podemos visualizar se tienen tres áreas con diferentes probabilidades de error de *frame* dependientes del color del mapa. Es decir, en la parte roja se tienen niveles de profundidad de hasta 1.10 metros, en cambio las áreas de color verde del río tienen la mínima profundidad del río siendo esta de 0.55 metros. Además de ello, se visualizan puntos rojos que representan las marcas de los diferentes niveles de agua. El símbolo color negro en forma de rayo es el indicador de existencia de un nodo o *gateway*, en este caso se trata del *river core* número 14.

Es importante mencionar que este mapa es una parte del escenario real del río Colima en el cual se implementó el proyecto *EWIN* y que este fenómeno solo se modela en áreas alrededor de los *river core*, ya que es en estas donde se realiza el envío y recepción de los paquetes.

Las áreas del mapa de la figura (4.9) fueron designadas con ayuda del mapa de la figura (4.10) que representa el aumento de distancia, entre el *river core* y ciertos puntos específicos del río, a lo largo del mismo.

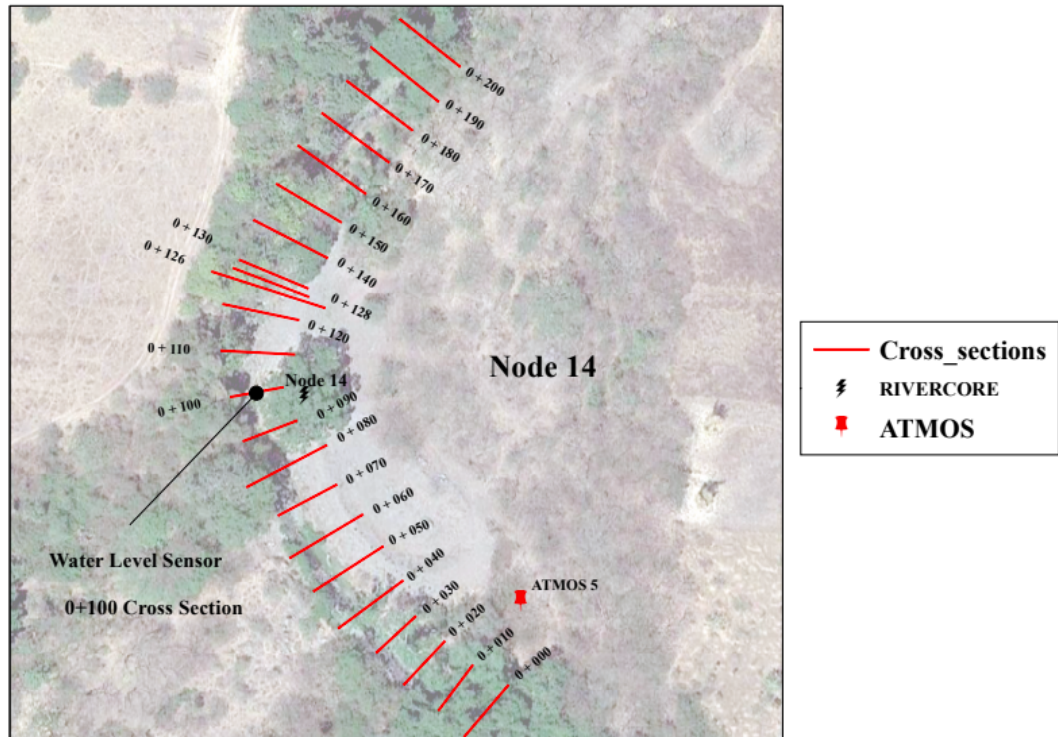


Figura 4.10: Nodo y distancias a lo largo del río [8].

Para poder realizar el modelado, se recurrió a la discretización de los puntos del río. Es decir, se tomaron los puntos geográficos de la ruta a lo largo del río para obtener la ruta completa de la trayectoria de la boya. Estos puntos tienen distancias aproximadamente de entre 1 y 10 metros de separación. Se tienen así, los segmentos discretos de la ruta de la boya a lo largo del río.

Se tiene además un punto de referencia que en la figura(4.9) se muestra como el Sensor del nivel de agua en un punto negro. Posteriormente, se visualiza el crecimiento de la distancia, este aspecto ha sido tomado como una herramienta muy importante para la asignación de los valores de probabilidad de error de *frame*.

4.5.2. Modelado del efecto de inversión de las boyas durante el trayecto

Como se discutió anteriormente, las áreas circunvecinas de la posición de cada *river core* tienen ciertas características que determinan una probabilidad de error en la recepción de los paquetes. Sabemos que existen secciones donde la corriente de agua es más rápida que en otras, esto provoca que cuando una boya que circula por la ruta del río tienda a girarse sobre su propio eje horizontal e incluso sobre su eje vertical durante su trayecto y por lo tanto perder muchos más paquetes de información durante la transmisión. Se decide entonces una probabilidad de error de *frame*.

A partir de esta probabilidad se pudo calcular la probabilidad de hundimiento de las boyas que se ha asignado a cada *end device*. El modulo tiene la capacidad de verificar su ubicación geográfica, y partir de eso, puede determinar si se encuentra en una zona verde, amarilla o roja. Dependiendo de esto, la boya es capaz de determinar una probabilidad de error. De esta manera, se genera un valor aleatorio con distribución uniforme, se realiza una comparación entre el valor aleatorio y el valor de la probabilidad de error. Si el valor aleatorio generado es mayor al valor de probabilidad de error obtenido anteriormente se da por hecho que la boya está invertida.

4.5.3. Modelado del efecto de estancamiento de las boyas

En esta parte del trabajo de este proyecto de tesis, se desarrolló e implementó un nuevo modelo de movilidad para ríos ya que *NS-3* no cuenta con uno que cumpla con las características necesarias para la simulación del escenario del proyecto *EWIN*. Para ello, se tomó como base uno de los que están incluidos en el simulador: el modelo llamado *waypoint mobility model*, que es un modelo de movilidad donde se toman como argumentos la posición (en coordenadas geográficas en nuestro caso) y la velocidad. Así, se agregó una probabilidad de estancamiento a cada dispositivo, representada por la ecuación (4.2).

$$P_{stuck} = 1 - P_{success} \quad (4.2)$$

Se ha otorgado el valor de éxito del 70%, eso quiere decir que la P_{stuck} es igual al 30%. y que se espera que el 70 por ciento de los dispositivos recorran sin problemas todo su modelo de movilidad.

La variable x de la $P_{success}$ se calcula con ayuda de la ecuación (4.3).

$$P_{success} = (1 - x)^{4070} \quad (4.3)$$

La probabilidad de éxito $P_{success}$ de las boyas será de 0.7 y se tiene además un número de puntos igual a 4074, este número de puntos fue obtenido gracias al uso de la herramienta de *Google Earth*.

Así, el valor de x de la ecuación (4.3) es igual a 0.000088. Cabe mencionar que este valor es configurable y se puede cambiar en el escenario principal dependiendo de los requerimientos del usuario.

Además, esta relación depende de una variable aleatoria. Ya que dicha variable aleatoria hace una comparación entre cada punto de nuestro modelo de movilidad y la variable aleatoria, y cuando este valor es mayor al valor de probabilidad de estancamiento x , el dispositivo se estanca y no se mueve en el futuro pues todos los puntos que quedan por ser recorridos se eliminan de la ruta a seguir de su modelo de movilidad.

4.6. Consumos Energéticos del end device y GPS en las boyas

Además de las características mencionadas anteriormente, se realizó la implementación del consumo energético por parte del GPS esto para tener el consumo total, tanto de los diferentes estados del dispositivo *end device* como del GPS.

Por una parte se tiene los cuatro estados del dispositivo.

- Transmisión Tx
- Recepción Rx
- Sleep
- Idle

Por otra parte, los estados del GPS solo son dos.

- Transmisión Tx
- No transmisión

El consumo energético en cada uno de los estados mencionados fue definido a partir de una medición realizada de manera física [11], utilizando dispositivos y escenarios implementados en el mundo real. Esta información se presenta en la tabla (4.5).

4. DESCRIPCIÓN DEL MODELO DE SIMULACIÓN DE LORA EN NETWORK SIMULATOR 3 (NS-3)

Tabla 4.5: Consumo de los estados ED y GPS, valor técnico de ED [11].

Parámetro	Valor medido	Valor hoja técnica SX1272
Fuente voltaje	3.3V	No aplica
End device		
Consumo Tx	50.3 mA	90 mA
Consumo Rx	31 mA	10.5 mA
Consumo Sleep	0.88 mA	0.1 μ A
Consumo Iddle	20.3 mA	1.5 μ A
GPS		
Consumo sensado	158.21 mA	No aplica
Consumo no sensado	0 mA	No aplica

4.6.1. Batería utilizada en el sistema

La batería utilizada en cada boya es siempre la misma para los escenarios simulados en este trabajo. Esto debido a que en las boyas físicas utilizan una batería de 4600 mAh con un voltaje de 3.3 volts.

A partir de estos datos, se procedió a la conversión de mAh a Joules. Se utiliza esta unidad de energía ya que el simulador realiza sus cálculos con ella. Para la primera conversión se toman los miliamperios hora (mAh) y el voltaje (V) para obtener vatios hora (Wh). La fórmula es la siguiente:

$$(mAh) * (V)/1000 = (Wh) \quad (4.4)$$

De esta manera se tiene un valor de 15.18 Wh . A partir de esta información se obtiene la energía en Joules, que es la relación de Wh multiplicado por 3600.

$$15.18Wh * 3600 = 54648Joules \quad (4.5)$$

Tenemos entonces un valor de 54648 Joules para la batería que la boya utiliza en nuestros escenarios.

4.6.2. Energía residual

La energía residual E_R hace referencia al consumo total energético del *end device* y del GPS dependiendo de sus diferentes estados. Esta se calculó con ayuda de la ecuación (4.6)

$$E_R = D \times C_E \times V_S \quad (4.6)$$

Donde D es la duración de cada estado en segundos, C_E es el consumo de energía de dicho estado que puede consultarse en la tabla (4.5) y V_S es el voltaje de suministro para las baterías de las boyas que en este caso es igual a 3.3 V.

4.6.3. Experimento de duración de la energía

A partir de la teoría presentada anteriormente y de todos los datos recavados hasta este momento, se diseñó un primer escenario para validar el consumo de energía de la teoría con el del simulador.

Comenzaremos por realizar el cálculo teórico del consumo energético total del envío de un paquete de 24 bytes y la recepción de un paquete de confirmación de 5 bytes.

Como podemos apreciar en la tabla (4.6), el valor del tiempo de transmisión T_x es de 61.696 milisegundos y el de recepción R_x del paquete *ACK* de confirmación es igual a 30.976 milisegundos; esto a partir de la teoría del tiempo en el aire ToA , utilizando los mismos parámetros para el cálculo de las tablas (4.2) (4.4).

Los valores del consumo de cada estado de la tabla (4.6) se obtuvieron a partir de la teoría, pero a continuación corroboramos esta información con la que el simulador proporciona.

Se colocó una boya y un *river core* con una distancia de separación de 300 metros con una velocidad de 1 metro por segundo, esto porque el generador de paquetes realiza su función cada 300 segundos. De esta manera cuando el paquete sea generado la transmisión ocurrirá y podremos validar los cálculos teóricos obtenidos con los resultados que el simulador proporciona.

La figura (4.11) muestra los resultados obtenidos en la simulación del módulo.

4. DESCRIPCIÓN DEL MODELO DE SIMULACIÓN DE LORA EN NETWORK SIMULATOR 3 (NS-3)

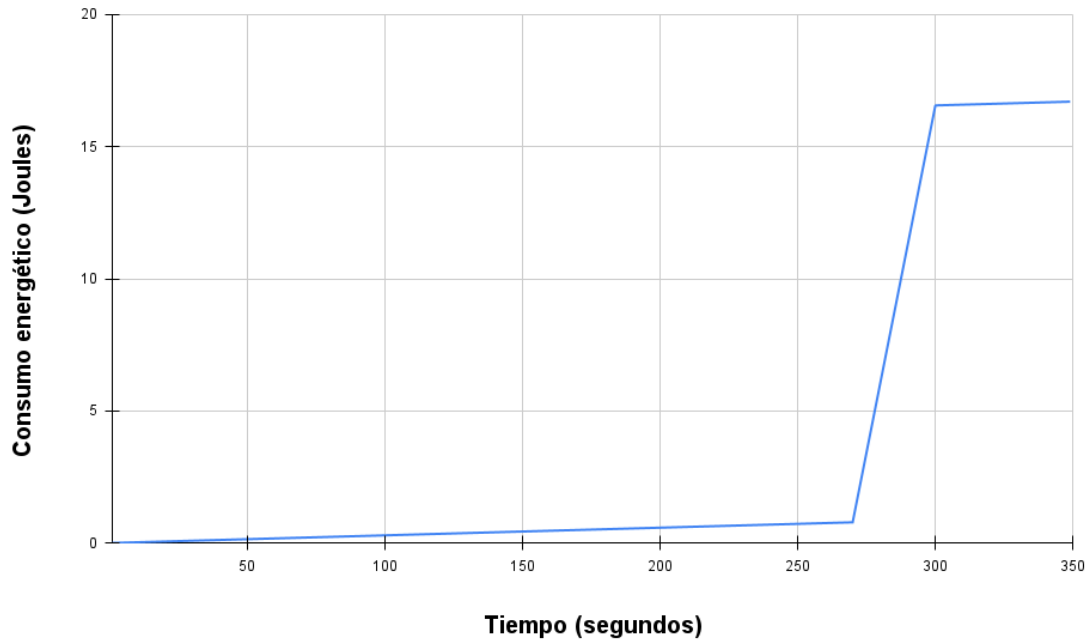


Figura 4.11: Consumo energético total de una boya que envía y recibe un solo paquete.

Podemos apreciar que el consumo de energía es mínimo durante los primeros 270 segundos de simulación, esto porque la boya se encuentra en un estado durmiente *Sleep* y es en este estado que se consume muy poca energía. A partir de ese momento, el consumo crece de forma significativa, pues el GPS despierta 30 segundos antes de la generación de cada paquete, en este caso esa generación sucede cada 300 segundos.

Cuando los 270 segundos son alcanzados, comienza el consumo de energía por parte del GPS. Posteriormente, a los 300 segundos comienza el consumo energético del estado de transmisión *Tx*, recepción *Rx* e inactivo *Iddle*. Estados por los que pasa la boya, para cambiar nuevamente al estado durmiente al final del ciclo de trabajo.

El simulador también proporciona los valores de cada consumo que dependen de los cuatro estados de la boya y del estado activo del GPS. Mediante la tabla (4.6) podemos corroborar que se tratan de los mismos valores que la teoría nos proporciona.

Tabla 4.6: Consumo teórico y simulado de los estados ED y GPS.

Estado	Consumo teórico	Consumo simulado
End device		
Consumo Tx	0.01024091904J	0.01024091904J
Consumo Rx	0.0031688448J	0.0031688448J
Consumo Sleep	0.8712J	0.8712J
Consumo Iddle	0.00049090272J	0.00049090272J
GPS		
Consumo sensado	15.66279	15.66279J
Totales		
-	16.54789067J	16.54789067J

Se verifica entonces, que el consumo de cada estado es el mismo cuando se trata del teórico como del simulado.

Comportamiento dinámico

El comportamiento dinámico de un determinado escenario se refiere al cambio o evolución de estados de un sistema cuando avanza el tiempo. Es importante para nosotros conocer estos cambios porque nos proporcionan una visión general o específica de los resultados en determinado escenario de estudio. Para el trabajo desarrollado en esta tesis se utilizaron diferentes escenarios, los cuales tienen diferentes características siendo una de las más importantes el uso de boyas que corresponden a los dispositivos llamados *end devices* en una red *LoRa*.

Por lo tanto, en este capítulo se presentan una serie de escenarios que se han simulado y analizado con el objetivo de conocer el comportamiento dinámico de la red que se ha establecido en este escenario, tomando como base al módulo desarrollado durante el trabajo de esta tesis.

5.1. Definición de diferentes escenarios

Para el estudio del comportamiento dinámico de la red que utiliza nuestro modelo de simulación de *LoRa* se definieron seis diferentes escenarios:

- Escenario A, en el cual la velocidad es constante y se utiliza solo un *end device*.
- Escenario B, en donde se realizan diferentes corridas con distintos valores (pero constantes) de velocidad, manteniéndose el uso de un solo *end device*.
- Escenario C, que toma diferentes valores de velocidad para el *end device*.
- Escenario D, donde se utiliza un único *end device* con velocidad variable dependiendo de su posición dado el tiempo transcurrido de simulación.

5. COMPORTAMIENTO DINÁMICO

- Escenario E, con 100 *end devices*, que tienen velocidad constante y la implementación del algoritmo de retroceso exponencial.
- Escenario F, con velocidad variable diferente en cada una de las 100 boyas.

5.1.1. Datos geográficos

La figura (5.1) presenta la ubicación geográfica del río Colima. Este nace en las faldas del volcán de Colima, cerca de la comunidad El Naranjal, y se va alimentando de pequeños manantiales que conforman su caudal. Al mismo tiempo, atraviesa la capital del estado de Colima hasta unirse al Río Armería que lleva el mismo nombre de la ciudad donde se ubica. Sus corrientes de agua recorren más de 50 kilómetros de distancia para unirse al antes mencionado río Armería. Durante su recorrido cruza la comunidad de El Chanal y las ciudades de Colima y Villa de Álvarez.

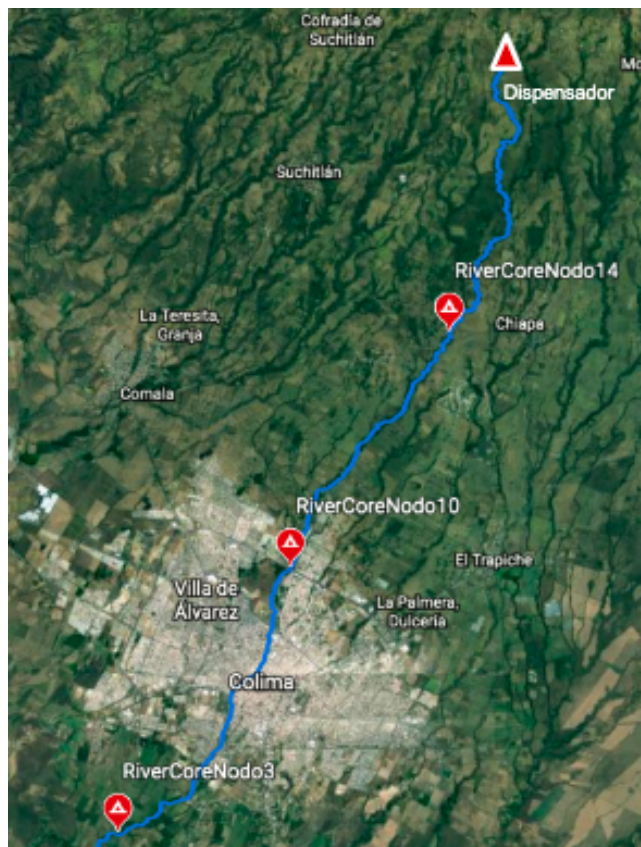


Figura 5.1: Posición geográfica del río, de los River Core y del dispensador.

Como se aprecia en la figura (5.1), la ruta que siguieron las boyas sobre el río fue trazada en el mapa en color azul. Esta ruta tiene una longitud mayor a los 30 kilómetros de distancia. Además, podemos observar indicadores en forma de globo a lo largo de la ruta, que llevan el nombre de *river core*. Estos corresponden al *river core* 3, 10 y 14 cuyas coordenadas geográficas se presentan en la tabla (5.1). Además de la posición del dispensador que se encargará de liberar las boyas o *end devices* al río.

Tabla 5.1: Coordenadas geográficas de los river core.

Descripción	Coordenadas geográficas
Dispensador	-103.65976 19.40567 1412.56506
River core 3	-103.7628 19.20628 388.64356
River core 10	-103.71591 19.27455 578.87799
River core 14	-103.67315 19.33446 851.23817

5.1.2. Selección de path loss utilizado

En la tabla (5.2) se muestra el modelo de *path loss* utilizado en todos estos escenarios. Es importante mencionar, que nuestro módulo *LoRa* también toma en cuenta los valores de la media y desviación estándar que también son mostrados en la tabla.

Tabla 5.2: Path loss lado derecho.

Lado	D(m)	Pathloss model
Derecho	24.2-194	$50 + 42 \cdot \log_{10}(d/24.2) + 4.3$
	Parámetro	Valor
	Media	8.2
	Desviación estándar	3

Durante las simulaciones utilizamos el valor del lado derecho del modelo de *path loss* presentado en la tabla (5.2).

5.1.3. Consumo energético de las boyas

La tabla (5.3) muestra los valores de consumo de este dispositivo utilizados en la simulación, tomando en cuenta el valor de energía de consumo por parte del GPS.

Tabla 5.3: Valores de consumo del end device.

Parámetro	Valor
Fuente voltaje	3.3V
End device	
Consumo Tx	50.3 mA
Consumo Rx	31 mA
Consumo Sleep	0.88 mA
Consumo Iddle	20.3 mA
GPS	
Consumo sensado	158.21 mA
Consumo no sensado	0 mA

5.1.4. Velocidad en ríos

Las mediciones de caudal en rápidos y ríos de gran pendiente son poco comunes debido a las dificultades técnicas asociadas con el posicionamiento y el funcionamiento de instrumentos suficientemente robustos. Se recopilaron datos detallados de velocidad de los rápidos del río Colorado en el este de Utah. Las velocidades de flujo se midieron con un *ADCP* y un tubo pitot-estático electrónico. La velocidad media máxima medida con el *ADCP* fue de $3,7 \text{ m/s}$. El tubo pitot-estático, aunque solo puede realizar mediciones puntuales, cuantificó la velocidad a 0,39 m por debajo de la superficie.

La velocidad media máxima medida con el tubo pitot-estático, fue de $5,2 \text{ m/s}$, con velocidades instantáneas de hasta $6,5 \text{ m/s}$ [43]. A partir de estos datos, se utilizarán diferentes velocidades para las simulaciones de este proyecto. Las velocidades van desde un metro por segundo y hasta 20 metros sobre segundo.

5.1.5. Tasa de generación de paquetes

La tasa de generación de paquetes de cada boya es la misma en todos los escenarios de la simulación. Es decir, para todas las simulaciones se tiene un valor de 300 segundos como valor de tiempo de generación entre cada paquete nuevo. Además, cada que se genera un paquete nuevo se tiene un tiempo de sincronización por parte del GPS entre la boya y los satélites con los cuales se calcula la ubicación geográfica. El tiempo de sincronía de la boya con los satélites es de aproximadamente 30 segundos, este valor fue obtenido de forma práctica en distintas pruebas de campo sobre el Río Colima. Por lo tanto, el GPS estará activo durante 30 segundos cada que un paquete sea generado.

5.1.6. Definición de los parámetros comunes de la simulación

Para la realización de las simulaciones de cada escenario se han definido parámetros específicos a cada uno de ellos. Estos parámetros se definen en la tabla (5.4).

Tabla 5.4: Parámetros de configuración.

Parámetro	Valor
Spreading Factor	7
Frecuencia	915 Mhz
Ancho de Banda	125 Mhz
Potencia Tx	10 dBm
Coding Rate (CR)	$\frac{4}{4+(CR=1)}$
Ganancia antena EDM	6 dB
Capacidad de la batería(J)	54648 J
Capacidad de la batería(mAh)	4600 mAh

5.2. Escenario A: Modelado de la red con una boya

Como primer escenario, se definió como característica exclusiva el hecho de tener una única boya en el escenario con una velocidad constante de 1 metro por segundo. Se montaron tres *river core* al escenario y también se colocó el dispensador que se encargó de liberar a la única boya. Para trazar la ruta en nuestro escenario, se convirtieron las coordenadas geográficas que entrega *Google Earth* en coordenadas cartesianas que pueda entender el simulador. Esta información se agregó al modelo de movilidad para que siguiera exactamente dicha ruta. El objetivo de esta primera simulación es conocer el comportamiento dinámico del módulo cuando se tiene una única boya y la ruta trazada a lo largo del río.

Las posiciones, tanto del dispensador como de los *river core*, se encuentran descritas en la tabla (5.1).

Por otro lado, la tabla (5.5) muestra los parámetros de configuración exclusivos de este escenario.

Tabla 5.5: Parámetros de configuración del escenario A.

Parámetro	Valor
Tasa de generación de paquetes	1 cada 300 seg
Tiempo de simulación	31500 seg
Número de boyas	1 única boya
Velocidad de la boya	1 metro por segundo

5.2.1. Resultados

La figura (5.2) muestra los primeros resultados. En el eje de las abscisas se encuentra el tiempo en segundos de la duración de la simulación, mientras que en el eje de las ordenadas se encuentra el número de paquetes que fueron cuantificados en tres diferentes categorías:

- Generados
- Recibidos
- Retransmitidos

Se aprecia que la recta de los paquetes generados crece de forma uniforme, ya que esta generación de paquetes ocurre en intervalos idénticos de tiempo (cada 300 segundos). Por otro lado, la representación de los paquetes recibidos y retransmitidos tienen forma de escalera con tres ascensos que corresponden a cada uno de los tres *river core*. Se puede notar que la línea de los paquetes recibidos alcanza siempre a la línea de los paquetes generados, sin descartar que existen retransmisiones que llegan a tener valores elevados.

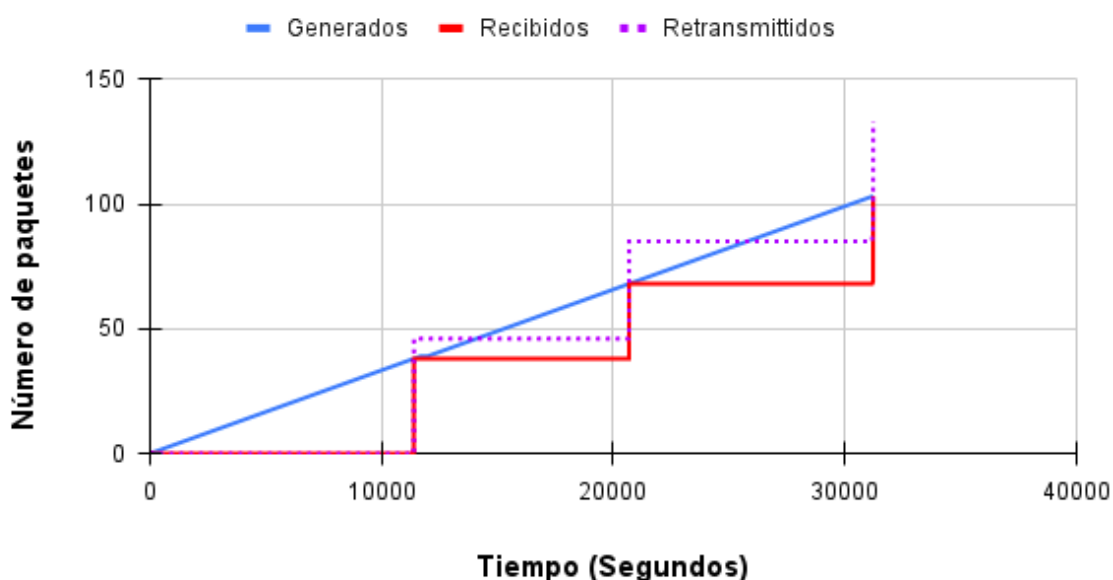


Figura 5.2: Rendimiento de la red usando una sola boya a velocidad de 1 metro sobre segundo.

Se recorrió un total de 31.25km . Este valor pudo ser medido ya que la boya avanza a una velocidad de un metro por segundo. En este periodo de tiempo, se generaron 104 paquetes de los cuales se tuvieron 133 retransmisiones en total. Por ello, podemos ver valores más altos en la gráfica de las retransmisiones en comparación con las llegadas.

5. COMPORTAMIENTO DINÁMICO

Todos los paquetes de información fueron recibidos en intervalos de tiempo pequeños. Como se comprobó en los cálculos anteriores, el tiempo de transmisión de cada paquete de 24 bytes es de 61.696 milisegundos y el del envío de confirmación *ACK* es de 30.976 milisegundos, esto quiere decir que el módulo tardaba menos de 100 milisegundos en realizar el envío, recepción y confirmación de cada paquete cuando no había retransmisiones de por medio. Es evidente que cuando se presentaba una retransmisión, este tiempo aumentaba.

La figura (5.3) muestra el consumo energético de la boya a lo largo del tiempo, pues como en el caso anterior el eje x representa el tiempo en segundos de la simulación pero con la diferencia de que el eje y ahora es el valor del nivel de batería en joules utilizado por la boya.

Se nota una tendencia a disminuir en forma de rizo, este es principalmente ocasionado por el GPS ya que es él quien consume la mayor cantidad de energía de la batería. Debido a que cada rizo corresponde a la activación del GPS, que como lo definimos, sucede 30 segundos antes de la generación de un nuevo paquete.

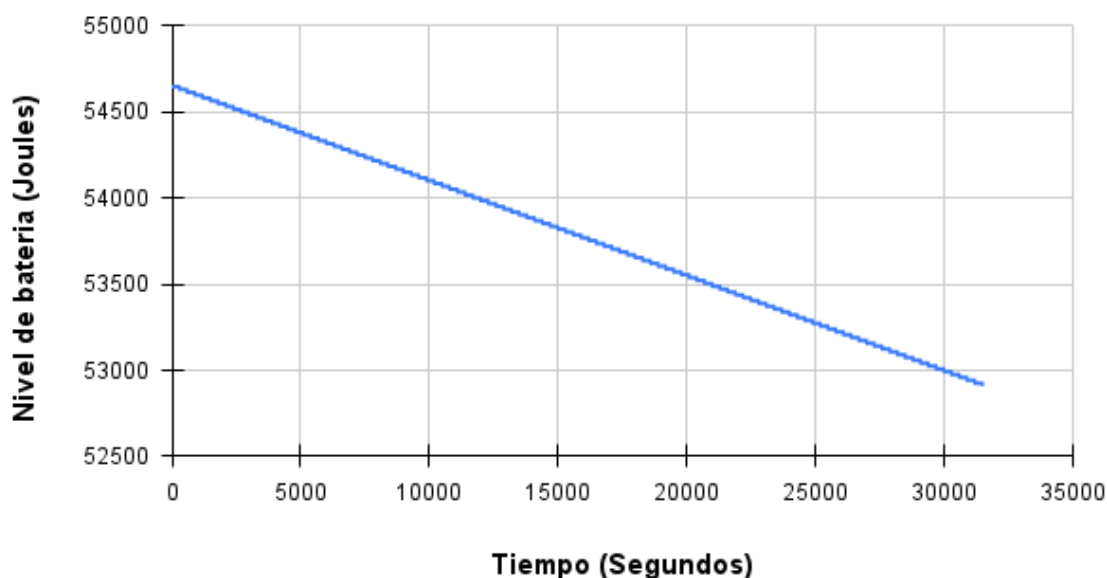


Figura 5.3: Consumo energético total del escenario con una boya a velocidad de 1 metro sobre segundo.

El valor de la batería es igual a 54648 Joules en su totalidad, a partir de ahí el modulo se encarga de calcular la energía residual dependiente del tiempo y de los estados por los cuales la boya y el GPS pasen.

La tabla (5.6) muestra los valores obtenidos de la simulación de consumo de los cuatro estados de la boya como del estado activo del GPS. Los valores representan el total de cada uno de esos cinco estados y la suma de estos proporciona un valor de consumo energético total de 1722.7 Joules.

Tabla 5.6: Resultados de consumo de energía del escenario A.

Estado end device	Valor de consumo
Consumo Tx	2.43734 J
Consumo Rx	0.32956 J
Consumo Sleep	90.6859 J
Consumo Iddle	0.394896 J
Consumo sensado GPS	1628.93 J
Consumo Total	1722.777696 J

La figura (5.4) presenta la relación del consumo de los cinco estados comparándose con el 100% de la energía total consumida.

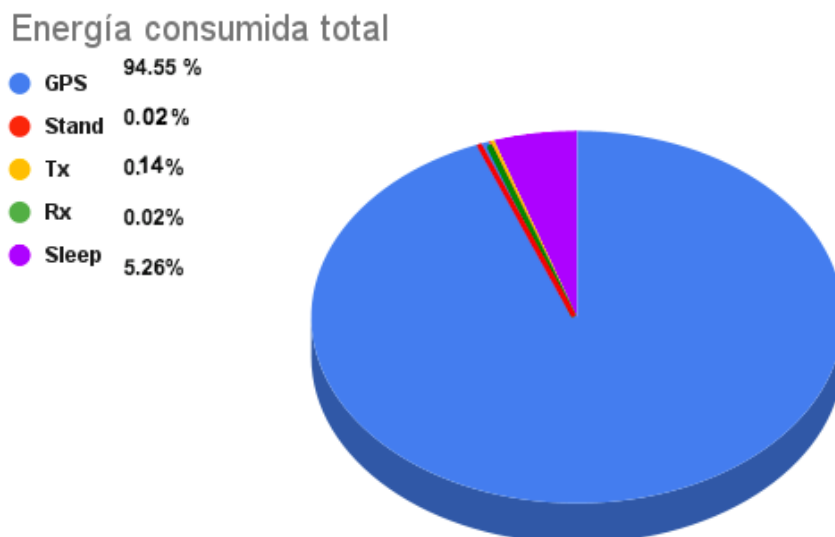


Figura 5.4: Consumo energético por boya del escenario A.

Podemos observar que más del 94 % de la energía total consumida es por parte del GPS, el resto corresponde a los cuatro estados de la boya siendo la etapa de dispositivo durmiente la que predomina. Esto se esperaba ya que la mayoría del tiempo, la boya se encuentra en modo *Sleep*. El consumo total de la batería del sistema es de 1722.7 joules.

5.3. Escenario B: Simulación de la red con diferentes valores de velocidad

Para conocer mejor el rendimiento de la red, se continuó trabajando con velocidades constantes en cada simulación pero ahora con diferentes valores que iban desde los 0.5 metros por segundo hasta los 20 metros por segundo en cada simulación.

Se realizó así, una serie de siete simulaciones. Cada una con un valor distinto pero constante de velocidad como parámetro de la boya. Los valores utilizados fueron de 0.5, 1, 3, 5, 10, 15 y 20 metros por segundo.

Para este escenario también se contó con una única boya. El escenario fue montado con las mismas condiciones mencionadas anteriormente pero con la diferencia de abordar diferentes velocidades en cada corrida de simulación.

La tabla (5.7) muestra los parámetros usados en esta simulación. Además, tanto la tabla (5.4) con los parámetros comunes, como la tabla (5.3) con los valores de consumo de los diferentes estados, han sido igualmente utilizadas en este escenario de manera similar a la simulación del escenario A.

Tabla 5.7: Parámetros de configuración del escenario B.

Parámetro	Valor
Tasa de generación de paquetes	1 cada 300 seg
Tiempo de simulación	63000 seg
Número de boyas	1 única boya
Velocidades de la boya en cada corrida	0.5,1,3,5,10,15,20 metros por segundo

5.3.1. Resultados

La figura (5.5) muestra el comportamiento de los escenarios con diferentes valores de velocidad en la boya. Se presentan los paquetes generados, los recibidos y los retransmitidos.

Se puede ver que el escenario con la menor velocidad, que corresponde a los 0.5 metros por segundo en la boya, es en el que se generaron más paquetes. Esto se debe a que por su velocidad toma mucho más tiempo recorrer todos los puntos del modelo de movilidad y en consecuencia la generación de paquetes aumenta y con ello el número de retransmisiones.

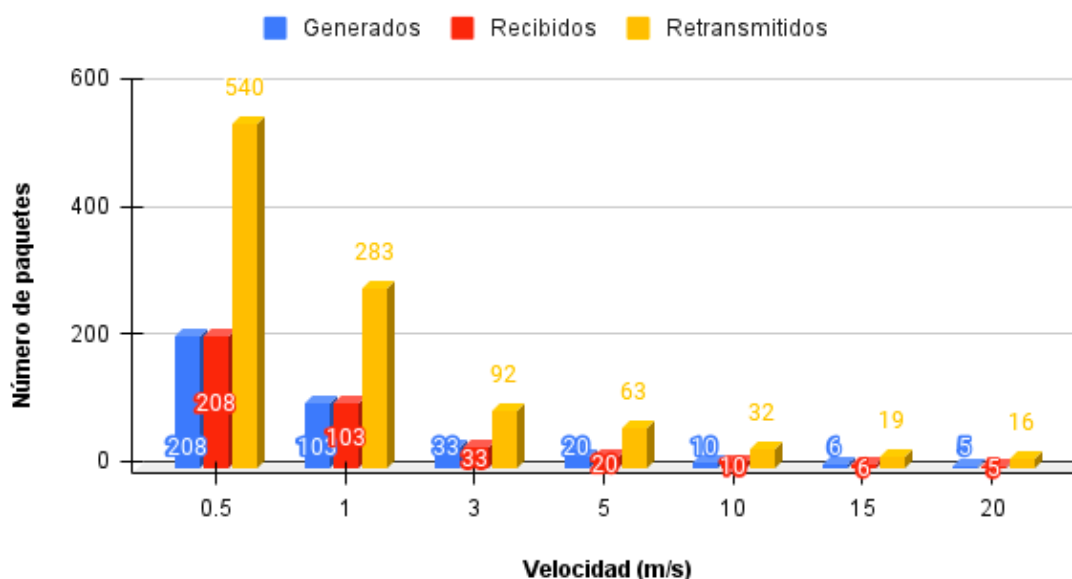


Figura 5.5: Relación de paquetes generados, recibidos y retransmitidos en el escenario B.

Por otra parte, al analizar el escenario de la boya con velocidad 1 metro por segundo, se observa que se trata de la mitad de paquetes generados. Esto es algo que se esperaba, pues la boya realiza el mismo recorrido en exactamente la mitad de tiempo. La generación de paquetes se ve afectada en este último comportamiento. Así como la generación de paquetes disminuyó, también lo hizo el número de retransmisiones.

5. COMPORTAMIENTO DINÁMICO

Se aprecia además como los valores tanto de paquetes generados, como de recibidos y retransmitidos disminuye a medida que la velocidad de la boya aumenta. Esto debido a que cada vez le tomará menos tiempo recorrer el mismo número de puntos en el modelo de movilidad y como consecuencia se tendrán menos paquetes generados por esta reducción de tiempo.

Un caso muy extremo sería el de la boya con velocidad de 20 metros por segundo, ya que solo se generarían 5 paquetes. Tomando en cuenta que cada paquete se genera en 300 segundos, tendríamos una distancia de 31.25 kilómetros recorridos en tan solo 1500 segundos, es decir 25 minutos.

A continuación se presenta la figura (5.6) que proporciona los diferentes valores de consumo energético total en cada uno de los casos anteriormente analizados.

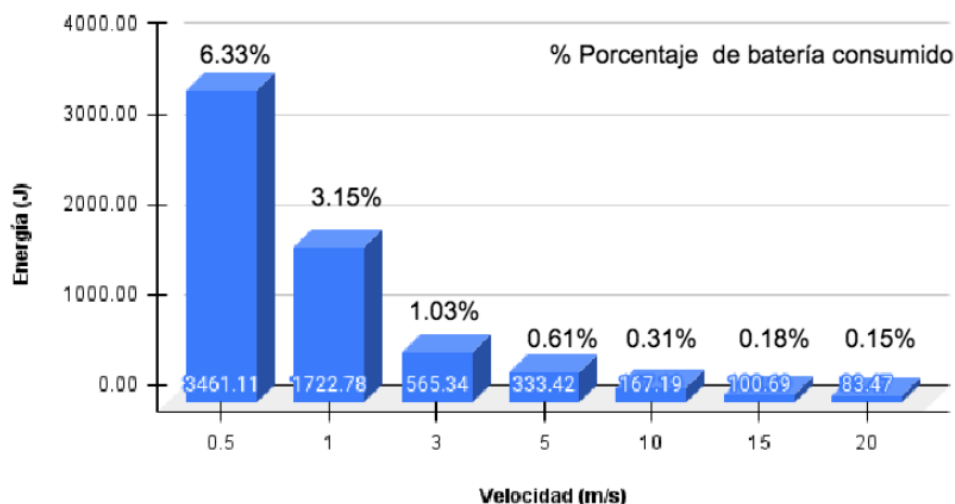


Figura 5.6: Consumo energético total de la boya para diferentes velocidades de movilidad.

Es evidente que mientras más largo sea el tiempo de la boya en recorrer el escenario completo, más grande será el consumo energético del dispositivo. Esto debido principalmente a que el GPS se activará un mayor número de veces, haciendo que la batería se drene cuando el tiempo de recorrido sea muy grande o la velocidad de la boya demasiado pequeña.

Cuando la boya se mueve a una velocidad de medio metro por segundo, tendrá un consumo total de 3461.11 Joules que es el doble de la energía total consumida del caso de 1 metro sobre segundo.

5.4. Escenario C: Probabilidad de éxito de recepción de paquetes para diferentes tasas de generación

Hasta este momento solo hemos utilizado una única tasa de generación de paquetes para las realizadas, que ha sido de 1 paquete generado cada 300 segundos. En este escenario vamos a conocer el comportamiento del escenario dependiendo de la tasa de generación de paquetes que cambiaremos junto con la velocidad de la boya. Esto proporcionará diferentes valores de probabilidad de éxito de los paquetes.

La tabla (5.8) muestra los parámetros de simulación para este escenario. La tabla muestra los valores de las diferentes tasas de generación.

Tabla 5.8: Parámetros de configuración del escenario C.

Parámetro	Valor
Tasa de generación de paquetes	1 cada 60,120,180 y 300 seg
Tiempo de simulación	90000 seg
Número de boyas	1 única boya
Velocidades de la boya en cada corrida	0.5,1,3,5,10,15,20 metros por segundo

5.4.1. Resultados

La figura (5.7) proporciona la probabilidad de éxito de paquete por cada tasa de generación. En el eje de las abscisas tenemos los diferentes valores de la velocidad que la boya tomó durante las simulaciones, y en el de las ordenadas tenemos la probabilidad de éxito de los paquetes generados en cada simulación.

Esta probabilidad se muestra en la ecuación (5.1).

$$P_{SucesP} = \frac{P_{Retrans}}{P_{Rec}} \quad (5.1)$$

5. COMPORTAMIENTO DINÁMICO

Donde P_{SucesP} es la probabilidad de éxito de los paquetes, $P_{Retrans}$ es el número total de paquetes retransmitidos durante la simulación y P_{Rec} es el número total de paquetes recibidos a lo largo del tiempo de simulación.

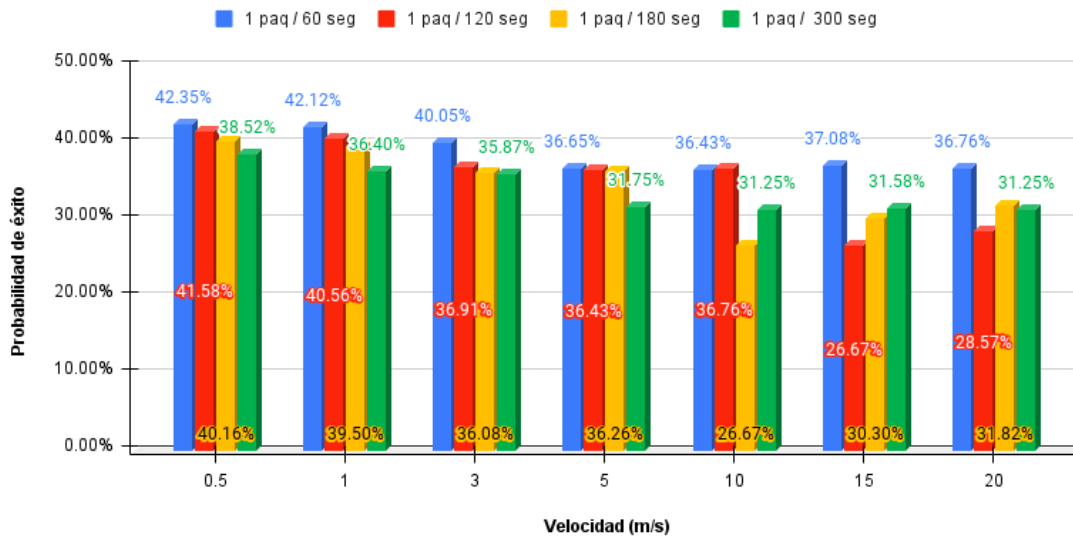


Figura 5.7: Probabilidad de éxito de paquetes a diferentes tasas de generación de datos.

Podemos visualizar que la probabilidad de éxito de los paquetes tiende a ser mayor cuando la velocidad de la boya es menor, es decir, se tiene una mayor probabilidad de éxito. Por ejemplo, en la simulación con velocidad de la boya de 1 metro por segundo en comparación con aquella de 20 metros por segundo, usando la misma tasa de generación de 1 paquete generado cada 60 segundos.

Por otra parte, el impacto de la tasa de generación se ve menos afectado en velocidades pequeñas. En este caso, para la boya con velocidad de medio metro por segundo se tiene un probabilidad de éxito de los paquetes con diferencias pequeñas en los porcentajes. Para este ejemplo en particular va de los 38.5% hasta los 42.3%.

Sin embargo, existe la posibilidad de que las probabilidades de éxito tengan umbrales más grandes entre sí. Vemos en la velocidad de la boya de 15 metros por segundo que las probabilidades no son tan uniformes como en otros casos.

Este efecto se aprecia en simulaciones donde la velocidad de la boya es muy grande.

5.5. Escenario D: Modelado de la red con valores crecientes de velocidad dependiendo de su ubicación geográfica

Las simulaciones anteriores muestran el comportamiento del escenario utilizando siempre a velocidades constantes. Es decir, durante cada una de las simulaciones la velocidad de la boya nunca tuvo variaciones a lo largo del trayecto. Por ello, en el escenario D se ha montado la boya con una característica diferente: la velocidad ya no es constante, ahora es variable. La variación de esta velocidad ha sido tomada a partir de las características geográficas del río Colima, como se muestra en la figura (4.9). Ya que cada afluente nuevo, encontrado a lo largo de la ruta, agrega corriente de agua en el río y por consecuencia se aumenta la velocidad de las boyas. Todo esto dependiendo del tamaño de dicho brazo del río. Y los valores de la variación se muestran en la tabla (5.9) que además muestra los valores de los parámetros de simulación de este escenario.

Además de los parámetros de la tabla (5.9), tanto la tabla (5.4) con los parámetros en común, como la tabla (5.3) con los valores de consumo de los diferentes estados, han sido igualmente utilizadas en este escenario.

Tabla 5.9: Parámetros de configuración del escenario D.

Parámetro	Valor
Tasa de generación de paquetes	1 cada 300 seg
Tiempo de simulación	5770 seg
Número de boyas	1 única boya
Velocidad de la boya	2 m/s hasta 8.7 m/s

5.5.1. Resultados

La figura (5.8) muestra el comportamiento de los paquetes a lo largo de la simulación. Como anteriormente lo hicimos, se contabilizaron a lo largo del tiempo la cantidad de paquetes recibidos, retransmitidos y generados.

5. COMPORTAMIENTO DINÁMICO

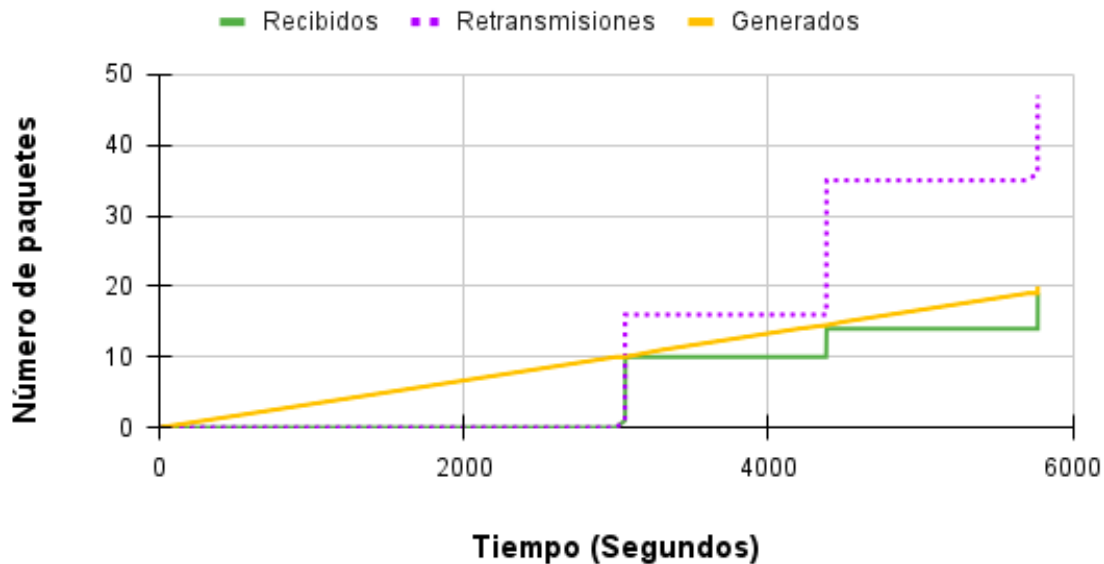


Figura 5.8: Paquetes recibidos, retransmitidos y generados en el escenario D.

Se observa ese comportamiento en forma escalonada con tres aumentos debido a la presencia de los tres *river core*. Pero a diferencia del escenario A, ahora los escalones tienen diferentes posiciones en el tiempo, esto debido a los cambios de velocidad en el escenario. Además, el número de retransmisiones crece demasiado, debido a que la velocidad también lo hace; como observamos en simulaciones pasadas, la velocidad tiene un papel crucial en cuanto al número de retransmisiones se trata.

En esta ocasión, la simulación dura 5770 segundos.

La figura (5.9) muestra el compartimiento a lo largo del tiempo del consumo energético de la boya. En este escenario también se analizó el consumo de energía de la boya a través del tiempo. Y como se esperaba, los rizados se hacen presentes, pero ahora en cantidad menor con respecto a las otras simulaciones.

Esto debido a que la simulación tiene una duración menor en tiempo ya que su aumento de velocidad hace que la boya llegue a su punto final en menos tiempo.

5.5 Escenario D: Modelado de la red con valores crecientes de velocidad dependiendo de su ubicación geográfica

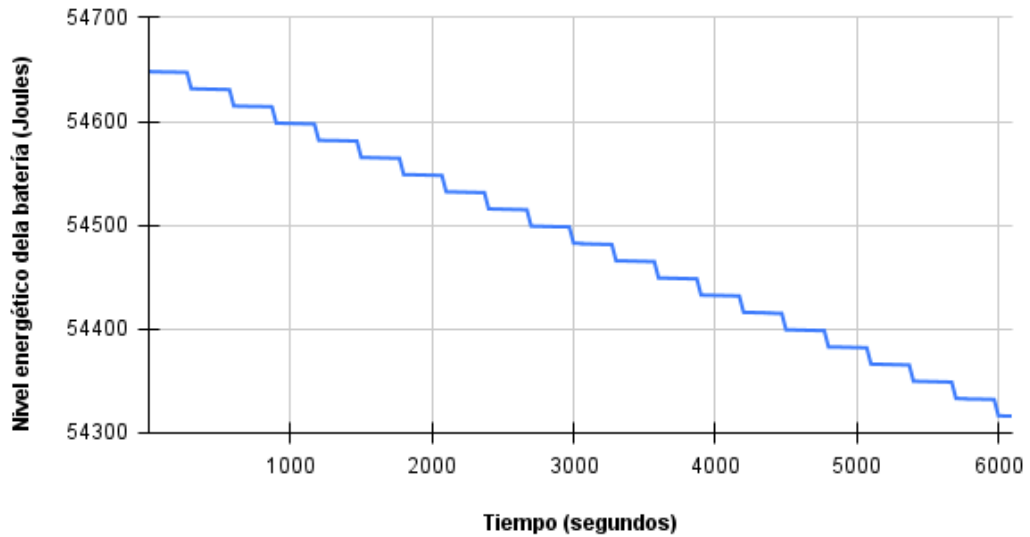


Figura 5.9: Consumo energético del escenario D.

El consumo total de cada estado tanto de la boya como del GPS se muestra en la tabla (5.10).

Tabla 5.10: Resultados de consumo de energía del escenario D.

Estado end device	Valor de consumo
Consumo Tx	0.696382 J
Consumo Rx	0.0602081 J
Consumo Sleep	16.7358 J
Consumo Iddle	0.470194 J
Consumo sensado GPS	313.256 J
Consumo Total	331.2185841 J

Finalmente, la figura (5.10) muestra la energía consumida total por la boya y el GPS en este escenario. Es un gráfico que muestra los valores porcentuales de cada estado de la energía total consumida.

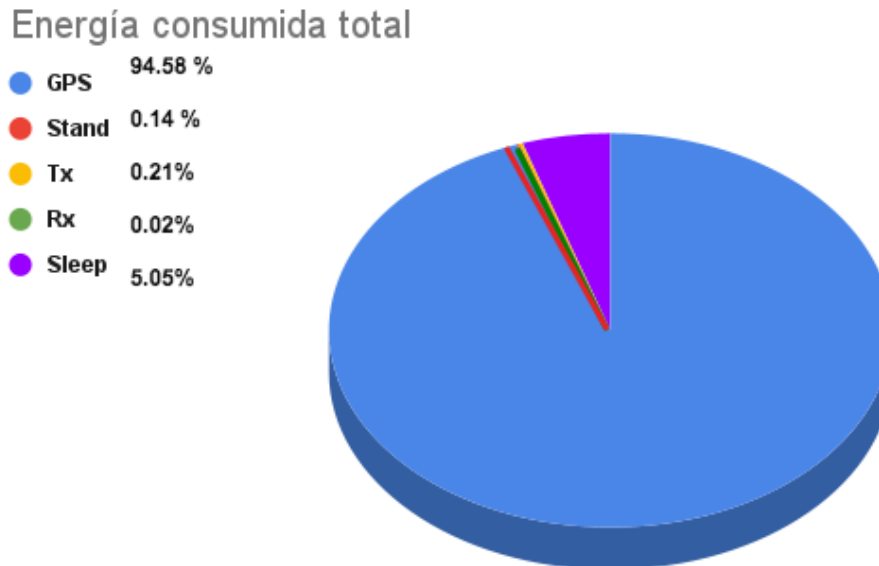


Figura 5.10: Consumo energético por boya en el escenario D.

Nuevamente predomina el consumo energético del GPS con un valor de 94.5 %. En cuanto a los cuatro estados de la boya, el que consumió más fue el de modo durmiente. Esto debido a que el dispositivo se encuentra la mayoría del tiempo durmiendo.

5.6. Escenario E: Modelado de la red con 100 boyas con velocidad constante

En el escenario E se continuó trabajando con una velocidad constante de 1 m/s . Y como característica distintiva, se montaron 100 boyas en el dispensador [44], mismo que las liberaba con una diferencia de 10 minutos entre cada una. Como podemos recordar de las simulaciones pasadas, la boya bajo la condición de tener una velocidad de 1 m/s , la ruta de este escenario y tasa de generación de 1 paquete cada 300 segundos, genera un total de 104 paquetes. Como ahora se tienen 100 de estos dispositivos, se esperan un total de 1040 paquetes generados.

La tabla (5.11) muestra el valor de los parámetros de simulación utilizados en este escenario.

Tabla 5.11: Parámetros de configuración del escenario E.

Parámetro	Valor
Tasa de generación de paquetes	1 cada 300 segundos
Número de boyas	100 boyas
Velocidad de la boya	1 metro por segundo
Valor de las semillas utilizadas	1,2,3,4,5

Otra característica importante a resaltar, es el hecho de que ahora se toma en cuenta a la semilla de los generadores de números aleatorios para conocer el comportamiento de nuestro escenario bajo diferentes parámetros en las partes del módulo donde hay una dependencia de aleatoriedad. A partir de este escenario, se tomara en cuenta el modelado del efecto de inversión de las boyas durante el trayecto descrito en el capítulo 4. Es decir, ya existe una probabilidad de inversión de los dispositivos en el río que recorren. Como primera corroboración, se hará el experimento de tal manera que todas las boyas recorran por completo la ruta definida en el modelo de movilidad. Es decir, se espera un total de 1040 paquetes generados. En este ejemplo, la semilla usada tuvo el valor de uno.

Se decidió además implementar el algoritmo de retroceso exponencial (*exponential back off*).

El cual es un protocolo de resolución de colisiones ampliamente utilizado. Este es empleado para espaciar la retransmisión repetida del mismo paquete de datos [45]. En redes de telecomunicaciones, este algoritmo se usa comúnmente para programar la retransmisión después de una colisión entre paquetes.

Este proceso fue implementado en la capa *MAC* de nuestros dispositivos, esto con el objetivo de comparar el número de colisiones obtenidas anteriormente y esperar una reducción de ese número en este escenario. Además de ello, se limitó a que los dispositivos solo hicieran un máximo de 5 intentos de retransmisión por paquete.

5.6.1. Resultados

La figura (5.11) muestra la cantidad de paquetes generados, recibidos, retransmitidos, las colisiones y el total de paquetes enviados (transmisiones y retransmisiones) del primer experimento en el escenario D.

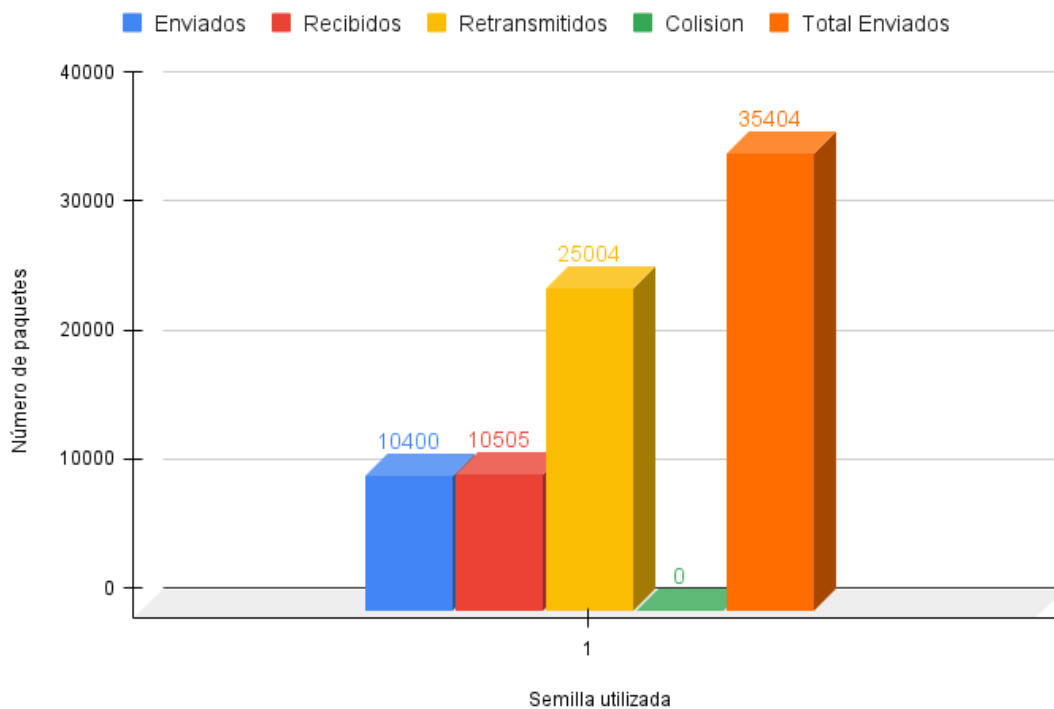


Figura 5.11: Relación de paquetes del escenario E.

Uno de los resultados más notables es que el número de paquetes colisionados es igual a cero. Esto debido a que las boyas siguen siempre la ruta trazada por el modelo de movilidad sin detenerse ni atorarse. Así se tienen los 10400 paquetes enviados que se esperaban; el número de retransmisiones sumado al número de paquetes enviados es igual al de paquetes enviados en total. Posteriormente, a cada dispositivo se le agregó la probabilidad de atorarse en algún punto de la ruta a seguir de tal manera que un 30 % de las boyas totales se atorará en algún punto del trayecto.

5.6 Escenario E: Modelado de la red con 100 boyas con velocidad constante

Esto debido a que en las pruebas de campo que se realizaron en los ríos de Colima, había áreas con muy altas probabilidades de nodos atorados, como se muestra en la figura (4.9). Se implementó entonces un nuevo modelo de movilidad para ríos que toma en cuenta dicha característica.

La tabla (5.12) muestra la probabilidad de estancamiento de las boyas del escenario E.

Tabla 5.12: Probabilidad de estancamiento de las boyas del escenario E.

Parámetro	Valor
Probabilidad de que la boya se atore	30%

La figura (5.12) muestra el comportamiento de los diferentes escenarios que tiene por diferencia el cambio de semilla utilizada y además en cada uno de ellos no se utilizó el algoritmo de retroceso exponencial. El número excesivo de colisiones de debe al estancamiento de boyas en cobertura de los *river core* que provocan colisiones con las demás boyas que pasan por el mismo *river core*.

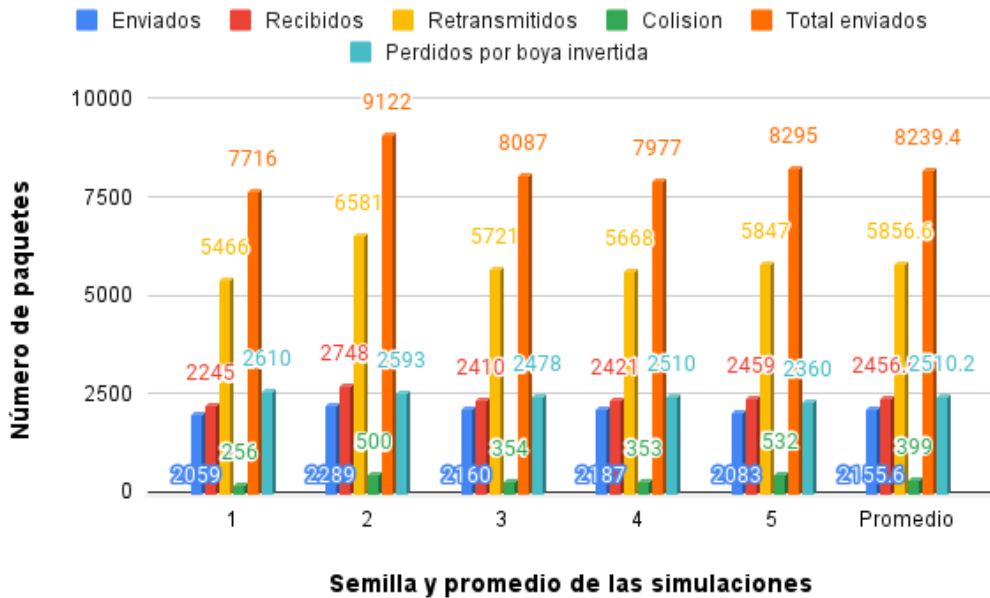


Figura 5.12: Comparación de la relación de paquetes del escenario E usando varias semillas y sin usar el algoritmo de retroceso exponencial.

5. COMPORTAMIENTO DINÁMICO

Una contribución a este proyecto de tesis, se incorporó el algoritmo de retroceso exponencial que reduce significativamente las colisiones. Como se aprecia en la figura (5.13), el comportamiento del escenario es muy parecido entre cada una de las corridas de la simulación. Ahora, se usaron semillas con valor de 1,2,3,4 y 5. Además, se muestra el promedio de las cinco simulaciones.

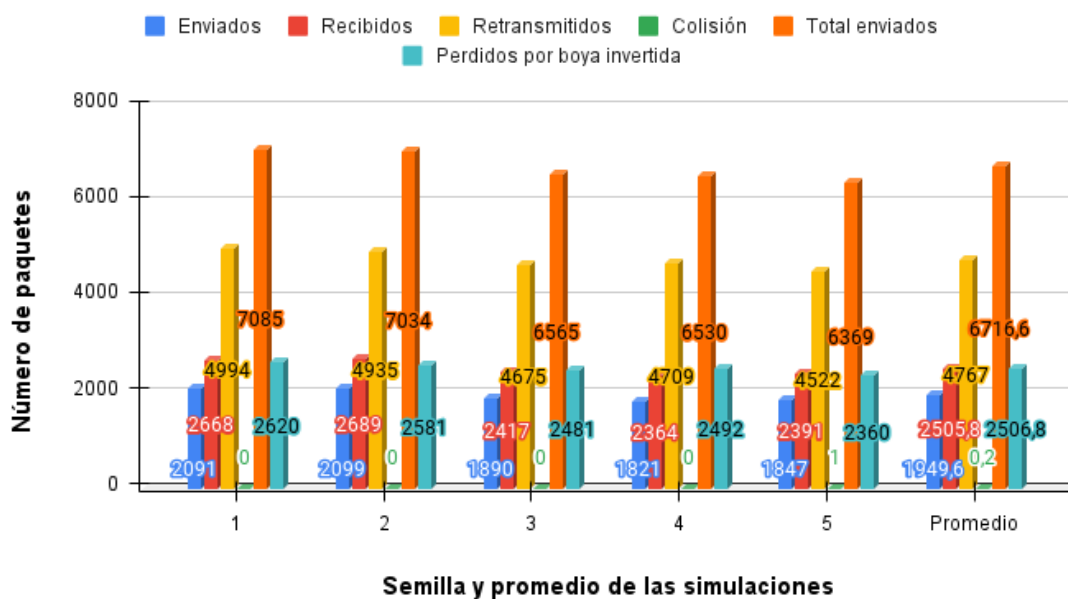


Figura 5.13: Comparación de la relación de paquetes del escenario E usando varias semillas y el algoritmo de retroceso exponencial.

Podemos visualizar que se reciben en promedio 2505 paquetes y se envían en total 6716 paquetes de los cuales 4767 corresponden a los retransmitidos y 2506 se pierden durante el transcurso del camino debido a que la boya se invierte.

5.7. Escenario F: Modelado de la red con 100 boyas con velocidad variable

Es importante tomar en cuenta que las boyas en el escenario E nunca se alcanzarán o rebasarán entre sí, ya que cada una sigue el mismo modelo de movilidad que comparten las mismas características. La única diferencia es el hecho de que cada una de ellas tiene una probabilidad de estancamiento. Entonces, con el objetivo de que las boyas tengan cada una un modelo de movilidad diferente (no solo en cuanto a la característica de estancamiento), se agregó una probabilidad a cada una de ellas que hacía que su velocidad variara dependiendo de una variable distribuida normalmente y aleatoria.

La probabilidad de variación de velocidad cambia en tramos que van desde uno hasta los 10 metros. Es decir, la velocidad de la boya cambia durante su trayecto cada determinado número de metros. Así, cada una de ellas tenía una velocidad diferente que variaba conforme la boya recorría las diferentes secciones del río.

La configuración del escenario F sigue siendo la misma que la del escenario E. Estos valores y parámetros se muestran en la tabla (5.11) y el valor de la probabilidad de estancamiento se muestra en (5.12).

5.7.1. Resultados

La figura (5.14) muestra los resultados obtenidos en la simulación del escenario F. Se aprecia la relación del número de paquetes contra los valores de semilla utilizados en cada simulación independiente.

Como se esperaba, el número de colisiones aumentó de una manera muy ligera ya que ahora las boyas se alcanzan y rebasan entre sí pero con una probabilidad pequeña de ocurrencia. Además, por tener variaciones de velocidades grandes en cada boya, las retransmisiones y el número de paquetes enviados aumentó. A pesar de estos resultados, podemos notar que el número de paquetes perdidos debido a que la boya se invierte es muy similar a los valores del escenario E.

5. COMPORTAMIENTO DINÁMICO

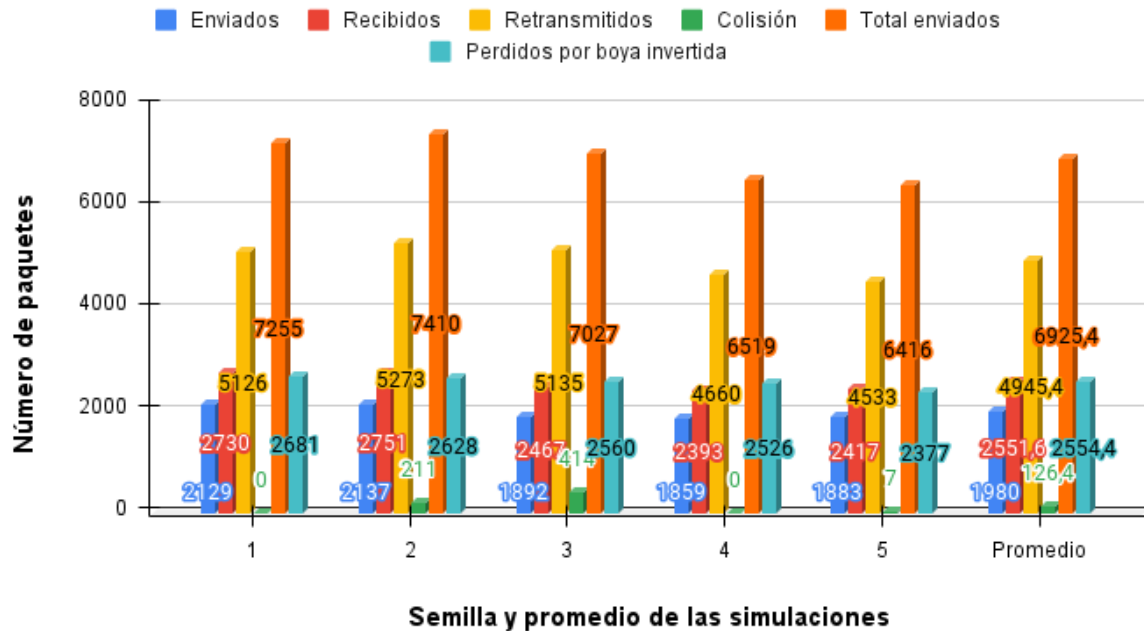


Figura 5.14: Comparación de la relación de paquetes del escenario F usando varias semillas.

5.8. Relación de entrega de paquetes (PDR)

La figura (5.15) muestra la relación de entrega de paquetes (PDR) de un escenario de simulación donde se hace variar el número de boyas, comenzando por 5 y aumentándose cada vez más hasta llegar a 100 unidades.

Se puede comprobar que esta relación tiende a mantener el mismo valor, es decir no sube ni baja bruscamente de un punto a otro en la gráfica.

Esto debido a diferentes factores que el módulo toma en cuenta en la simulación, como lo son el hecho de si la boya está invertida o no y también el efecto de pérdidas por interferencia. Recordemos que el número de paquetes perdidos tiende a ser un número alto, además de el valor del *shadow fading* donde se toma en cuenta el valor de la desviación estándar y de la media que se multiplica por un valor z que depende de una variable aleatoria.

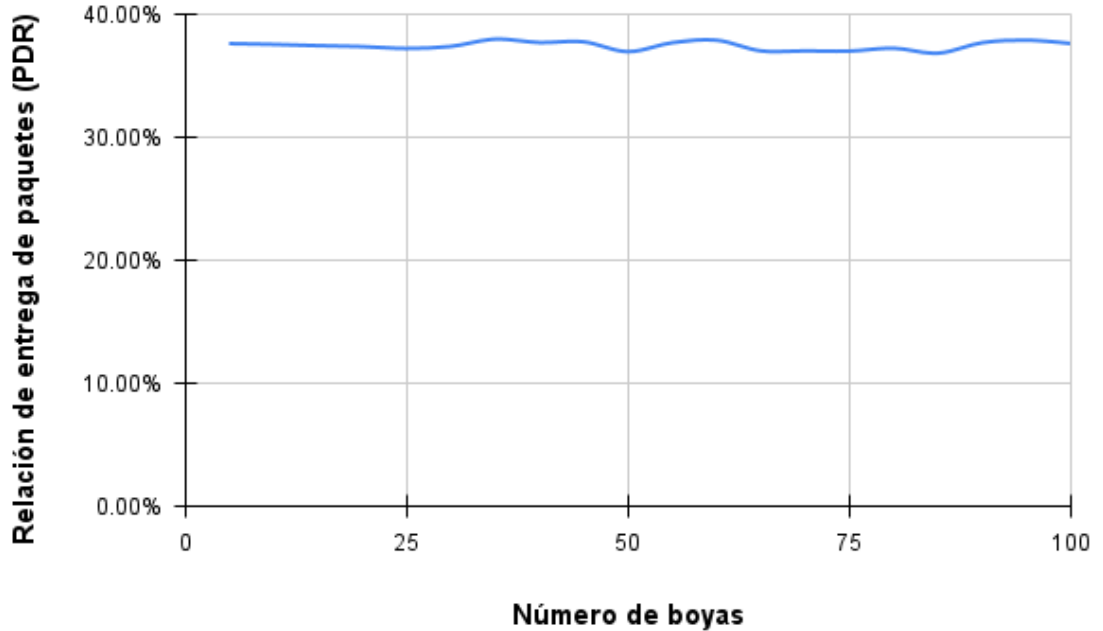


Figura 5.15: Relación de entrega de paquetes (PDR).

5.9. Retardo promedio de los paquetes

El retardo de un paquete hace referencia al tiempo consumido desde la creación del paquete hasta el arribo a su destino. La figura (5.16) proporciona la relación del promedio de los retardos de los paquetes recibidos a través del tiempo. En este caso, se trata de todos los paquetes generados, enviados y recibidos en un escenario de 10 boyas.

La gráfica comienza a registrar retardos a partir de los 12169 segundos de simulación, esto debido a que los tiempos de generación de paquetes y los de envío dependen de tanto de todo el trayecto de cada boya como de la velocidad del dispositivo. Imaginemos que se ha generado el primer paquete a los 300 segundos después de la liberación de la primera boya, este se almacenará en la memoria del dispositivo que recorre el río. Lo mismo sucede para los siguientes paquetes generados.

5. COMPORTAMIENTO DINÁMICO

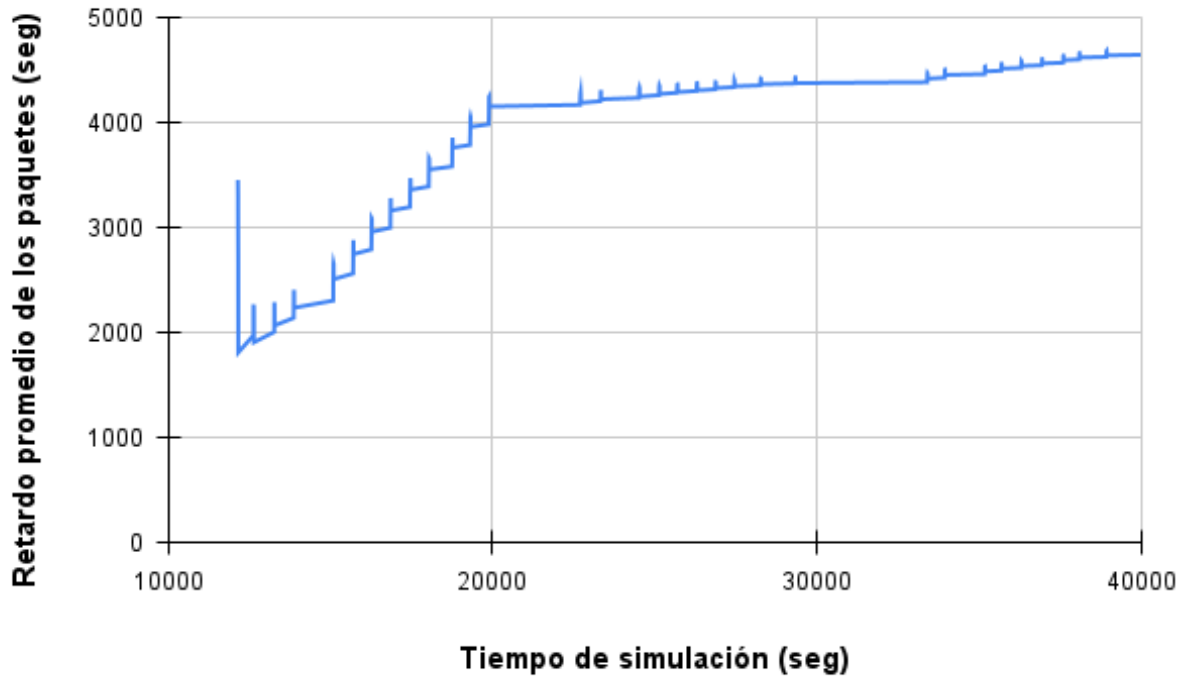


Figura 5.16: Retardo promedio de los paquetes en la cola para una simulación de 10 boyas.

La liberación de los paquetes sucede hasta el encuentro de la boya (que debe estar en la zona de transmisión) con el primer *river core* encontrado en la ruta. Es por esto que se tienen tiempos de retardos tan grandes, pues los paquetes que han sido generados desde la liberación de la boya deben esperar mucho tiempo para ser enviados y recibidos. En un escenario más realista se deben colocar más estaciones *river core*, para tener retardo de entrega de paquetes más reducidos. Sin embargo, esto podría aumentar el costo del proyecto de monitorización de ríos, dado que cada estación *river core* tendría un costo asociado.

Conclusiones

En este capítulo se presentan las conclusiones del trabajo desarrollado en esta tesis y los trabajos futuros o complementarios que pueden derivarse con base en los resultados obtenidos en la obra.

6.1. Trabajos futuros

Se espera que en el futuro, las mejoras a este módulo continúen principalmente en la implementación de *LoRaWAN* en la capa *MAC* de las boyas. De esta manera, ahora el valor del factor de esparcimiento *SF* pueda ser automáticamente ajustado dependiendo de las características del escenario de la simulación.

Además, de una mejora en la relación del promedio del retardo promedio de paquetes pues como hemos visto, estos tienen valores muy grandes debido a que el sistema solo cuenta con tres *river core* montados a lo largo de la trayectoria de las boyas. Una opción para reducir este efecto, sería el de colocar más de estos dispositivos para que las boyas no demoren tanto tiempo en encontrarse con ellos y poder realizar transferencia de información de una mejor manera.

6.2. Conclusiones

Los resultados muestran que tanto el proyecto *EWIN* como el módulo desarrollado de *LoRa* y los escenarios presentados en esta tesis son completamente viables para su implementación, a pesar de utilizar altos valores de media y desviación estándar en el *shadow fading*. Incluso cuando utilizamos valores muy volátiles de velocidades. Las boyas cumplen con sus objetivos en cada una de las pruebas planteadas.

6. CONCLUSIONES

Mediante el trabajo realizado a partir del simulador *NS-3* en este estudio, se logró cumplir con todos los objetivos definidos en el primer capítulo de la tesis. Ya que se desarrolló e implementó un módulo *LoRa* al cual posteriormente se le asignó un modelo de propagación de ríos. Este modelo es la base que define diferentes escenarios basados en boyas móviles que usan el protocolo *LoRa*. La ruta siempre es la misma, y las boyas siguen el camino que contiene las coordenadas geográficas exactas del río Colima, haciendo de esta obra una aportación mucho más realista. Por medio de las diferentes simulaciones realizadas en este estudio, se tiene un modelado del consumo energético de las boyas bajo diferentes escenarios y características. Por lo tanto, esta tesis sirve como referencia para comparar y evaluar los resultados de simulación del modelo de propagación con el de la publicación científica [25], modelo que ha sido desarrollado a partir de pruebas físicas en campo.

6.2.1. Conclusiones en los escenarios de simulación

En el caso específico del escenario A donde solo tenemos una boya con velocidad constante de un metro por segundo se concluye que se puede recibir sin ningún inconveniente toda la información sensada a lo largo del río. Además de que el consumo energético total durante el trayecto fue de 1722.7 joules y representan un 3.15 % del total de energía almacenada en la batería. De los diferentes estados, tanto del GPS como de la boya, el que más consumo representa es el de sensado de GPS ya que en términos generales corresponde al 94.55 % del total.

En cuanto al escenario B, podemos constatar que la velocidad de la boya es un factor que impacta mucho en cuanto al consumo total de la energía. Ya que cuando se tienen valores menores de velocidad, el consumo tiende a ser mayor a comparación de las velocidades altas. Esto debido a que siendo la misma ruta, a velocidades menores el dispositivo se demora más tiempo en recorrerla por completo y por lo tanto repitiendo más veces cada estado de los dispositivos.

Además de validar que mientras más alta sea la velocidad de desplazamiento de la boya, la probabilidad de éxito de recepción de los paquetes tenderá a ser menor. Pues como podemos ver en el escenario C, a velocidades más grandes del dispositivo, la pérdida de paquetes crece.

Específicamente en el escenario D, donde la velocidad deja de ser constante se concluye que mientras más crece la velocidad el número de pérdida de paquetes también crece y como consecuencia la cantidad de retransmisiones de los paquetes. El consumo de la energía en este escenario baja considerablemente dependiendo de la proporción de crecimiento de la velocidad de la boya.

En el escenario E, al no tener el algoritmo de retroceso exponencial se visualizan hasta un máximo de colisiones del 6 % del total de los paquetes enviados.

En cambio, después de la implementación del algoritmo se tiene un 0% de paquetes colisionados.

Para el caso de los escenarios donde la boya tiene cierta probabilidad de estancamiento, podemos concluir que la mayor pérdida de paquetes es causada por:

- Los efectos de la boya invertida.
- Los efectos del *shadow fading*.

Ya que de acuerdo a nuestros resultados, el número de colisiones es casi siempre nulo a velocidades promedio de 1 metro por segundo. De hecho, cuando esta velocidad crece, podemos ver un aumento muy pequeño en las colisiones. Aún así, las pérdidas de paquetes se atribuyen principalmente a los dos factores antes mencionados.

6.2.2. Conclusión del consumo energético

En cuanto al consumo energético, podemos concluir que la cantidad de energía de las baterías utilizada es suficiente para abastecer las necesidades tanto de cada una de las boyas como de cada módulo GPS. El valor consumido por el GPS con respecto al total de energía por boya en todos los casos, no supera el 10% del total de la energía de la batería. Eso quiere decir que las boyas tendrán suficiente energía para realizar todo el recorrido a lo largo de la ruta del río, incluso si utilizan velocidades muy bajas. Además, aquellas boyas que se estancuen podrán ser recuperadas fácilmente ya que el consumo energético no resultaría un problema para el funcionamiento del GPS que proporcionara su ubicación exacta.

6.2.3. Conclusiones en retardo y relación de entrega de paquetes

La relación de entrega de paquetes (*PDR*) tiende a mantener su mismo valor a lo largo de la simulación, incluso cuando se aumenta cada vez más el número de boyas en un determinado escenario. Esto ocurre porque la pérdida de paquetes está sujeta a los diferentes efectos implementados en el módulo, como lo son las pérdidas por interferencia, pérdidas por boya invertida, la desviación estándar y la media que comprende el *shadow fading*.

6. CONCLUSIONES

También se concluye que el promedio de los retardos depende completamente de la posición geográfica de los *river core*. Pues mientras más dispositivos de estos estén posicionados a lo largo del río, el retardo será menor porque la boya tomará menos tiempo en encontrarse con alguno de ellos y la transmisión y recepción de paquetes se realizará de una manera más temprana, lo que resultara en retardos menores.

Apéndice A

Anexos

A.1. Tablas data rate

A.1.1. Tabla ancho de banda de 125kHz

Tabla A.1: Data rate ancho de banda 125kHz.

SF	Rate code	Bw	$R_b = SF * (\text{Rate code} / (2^{SF} / Bw))$	$T_s = 2^{SF} / Bw$	$R_s = 1 / T_s$
7	1	125000	6835.9375	0.001024	976.5625
8	1	125000	3906.25	0.002048	488.28125
9	1	125000	2197.265625	0.004096	244.140625
10	1	125000	1220.703125	0.008192	122.070313
11	1	125000	671.3867188	0.016384	61.0351563
12	1	125000	366.2109375	0.032768	30.5175781
7	0.8	125000	5468.75	0.001024	976.5625
8	0.8	125000	3125	0.002048	488.28125
9	0.8	125000	1757.8125	0.004096	244.140625
10	0.8	125000	976.5625	0.008192	122.070313
11	0.8	125000	537.109375	0.016384	61.0351563
12	0.8	125000	292.96875	0.032768	30.5175781
7	0.6666667	125000	4557.291895	0.001024	976.5625
8	0.6666667	125000	2604.166797	0.002048	488.28125
9	0.6666667	125000	1464.843823	0.004096	244.140625
10	0.6666667	125000	813.802124	0.008192	122.070313
11	0.6666667	125000	447.5911682	0.016384	61.0351563
12	0.6666667	125000	244.1406372	0.032768	30.5175781
7	0.5714286	125000	3906.250195	0.001024	976.5625
8	0.5714286	125000	2232.142969	0.002048	488.28125
9	0.5714286	125000	1255.58042	0.004096	244.140625
10	0.5714286	125000	697.5446777	0.008192	122.070313
11	0.5714286	125000	383.6495728	0.016384	61.0351563
12	0.5714286	125000	209.2634033	0.032768	30.5175781
7	0.5	125000	3417.96875	0.001024	976.5625
8	0.5	125000	1953.125	0.002048	488.28125
9	0.5	125000	1098.632813	0.004096	244.140625
10	0.5	125000	610.3515625	0.008192	122.070313
11	0.5	125000	335.6933594	0.016384	61.0351563
12	0.5	125000	183.1054688	0.032768	30.5175781

A.1.2. Tabla ancho de banda de 250kHz

Tabla A.2: Data rate ancho de banda 250kHz.

SF	Rate code	Bw	$R_b = SF * (\text{Rate code} / (2^S F / Bw))$	$T_s = 2SF / Bw$	$R_s = 1 / T_s$
7	1	250000	13671.3867188	0.000512	1953.125
8	1	250000	7812.5	0.001024	976.5625
9	1	250000	4394.53125	0.002048	488.28125
10	1	250000	2441.40625	0.004096	244.140625
11	1	250000	1342.773438	0.008192	122.070313
12	1	250000	732.421875	0.016384	61.0351563
7	0.8	250000	10937.5	0.000512	1953.125
8	0.8	250000	6250	0.001024	976.5625
9	0.8	250000	3515.625	0.002048	488.28125
10	0.8	250000	1953.125	0.004096	244.140625
11	0.8	250000	1074.21875	0.008192	122.070313
12	0.8	250000	585.9375	0.016384	61.0351563
7	0.6666667	250000	9114.583789	0.000512	1953.125
8	0.6666667	250000	5208.333594	0.001024	976.5625
9	0.6666667	250000	2929.687646	0.002048	488.28125
10	0.6666667	250000	1627.604248	0.004096	244.140625
11	0.6666667	250000	895.1823364	0.008192	122.070313
12	0.6666667	250000	488.2812744	0.016384	61.0351563
7	0.5714286	250000	7812.500391	0.000512	1953.125
8	0.5714286	250000	4464.285938	0.001024	976.5625
9	0.5714286	250000	2511.16084	0.002048	488.28125
10	0.5714286	250000	1395.089355	0.004096	244.140625
11	0.5714286	250000	767.2991455	0.008192	122.070313
12	0.5714286	250000	418.5268066	0.016384	61.0351563
7	0.5	250000	6835.9375	0.000512	1953.125
8	0.5	250000	3906.25	0.001024	976.5625
9	0.5	250000	2197.265625	0.002048	488.28125
10	0.5	250000	1220.703125	0.004096	244.140625
11	0.5	250000	671.3867188	0.008192	122.070313
12	0.5	250000	366.2109375	0.016384	61.0351563

A.1.3. Tabla ancho de banda de 500kHz.**Tabla A.3:** Data rate ancho de banda 500kHz

SF	Rate code	Bw	$R_b = SF * (\text{Rate code} / (2^{SF} / Bw))$	$T_s = 2^{SF} / Bw$	$R_s = 1 / T_s$
7	1	500000	27343.75	0.000256	3906.25
8	1	500000	15625	0.000512	1953.125
9	1	500000	8789.0625	0.001024	976.5625
10	1	500000	4882.8125	0.002048	488.28125
11	1	500000	2685.546875	0.004096	244.140625
12	1	500000	1464.84375	0.008192	122.070313
7	0.8	500000	21875	0.000256	3906.25
8	0.8	500000	12500	0.000512	1953.125
9	0.8	500000	7031.25	0.001024	976.5625
10	0.8	500000	3906.25	0.002048	488.28125
11	0.8	500000	2148.4375	0.004096	244.140625
12	0.8	500000	1171.875	0.008192	122.070313
7	0.6666667	500000	18229.16758	0.000256	3906.25
8	0.6666667	500000	10416.66719	0.000512	1953.125
9	0.6666667	500000	5859.375293	0.001024	976.5625
10	0.6666667	500000	3255.208496	0.002048	488.28125
11	0.6666667	500000	1790.364673	0.004096	244.140625
12	0.6666667	500000	976.5625488	0.008192	122.070313
7	0.5714286	500000	15625.00078	0.000256	3906.25
8	0.5714286	500000	8928.571875	0.000512	1953.125
9	0.5714286	500000	5022.32168	0.001024	976.5625
10	0.5714286	500000	2790.178711	0.002048	488.28125
11	0.5714286	500000	1534.598291	0.004096	244.140625
12	0.5714286	500000	837.0536133	0.008192	122.070313
7	0.5	500000	13671.875	0.000256	3906.25
8	0.5	500000	7812.5	0.000512	1953.125
9	0.5	500000	4394.53125	0.001024	976.5625
10	0.5	500000	2441.40625	0.002048	488.28125
11	0.5	500000	1342.773438	0.004096	244.140625
12	0.5	500000	732.421875	0.008192	122.070313

Bibliografía

- [1] R. Gomathi, G. Hari, S. Krishna, and E. Brumancia, *A Survey on IoT Technologies, Evolution and Architecture*. International Conference on Computer, Communication, and Signal Processing (ICCCSP), 2018. [XIII, 9, 12, 13](#)
- [2] F. García Jiménez, A. Burbano Abril, A. Gutiérrez Gómez, A. Sinahue García, and V. Rangel Licea, “Cloud based ad-hoc networks for the ewin project,” National University of Mexico, Department of Engineering. Mexico Radio Team, Tech. Rep. RN-CS-99-03, 2019. [XIII, 14](#)
- [3] J. de Carvalho Silva, J. J. P. C. Rodrigues, A. M. Alberti, P. Solic, and A. L. L. Aquino, “Lorawan — a low power wan protocol for internet of things: A review and opportunities,” in *2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech)*, 2017, pp. 1–6. [XIII, 23, 27](#)
- [4] semtech, “Lora® and lorawan®: A technical overview,” <https://lora-developers.semtech.com/library/tech-papers-and-guides/lora-and-lorawan/>, 2019. [XIII, xv, 24, 27](#)
- [5] V. Talla, M. Hesar, B. Kellogg, A. Najafi, J. Smith, and S. Gollakota, “Lora backscatter: Enabling the vision of ubiquitous connectivity,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, 05 2017. [XIII, 25](#)
- [6] Bouras, Christos, V. Kokkinos, Papachristos, and Nikolaos, “Performance evaluation of lorawan physical layer integration on iot devices,” in *2018 Global Information Infrastructure and Networking Symposium (GIIS)*, 2018, pp. 1–4. [XIII, 25, 26](#)
- [7] L. Alliance, “Sx1272/3/6/7/8: Lora modem designer’s guide an1200.13 lora™ alliance,” *Tech. Rep.*, 2013. [XIII, 28, 34](#)

BIBLIOGRAFÍA

- [8] A. Gutiérrez Gómez, V. Rangel Licea, M. Edwards, G. D. John, R. Aquino, J. López, O. Mendoza Cano, M. Lopez Guerrero, and Y. Geng, “A propagation study of lora p2p links for iot applications: The case of near-surface measurements over semitropical rivers in mexico,” *Sensor 2021*. [XIII](#), [XIII](#), [xv](#), [37](#), [55](#), [57](#)
- [9] L. Alliance, “Rp002-1.0.3 lorawan regional parameters,” *Tech. Rep.*, 2020. [xv](#), [21](#), [22](#)
- [10] Semtech, “Lora sx1276/77/78/79 datasheet,” <https://www.semtech.com/products/wireless-rf/lora-core/sx1276>, 2015. [xv](#), [23](#), [35](#)
- [11] *Semtech SX1272 Long Range, Low Power RF Transceiver 860-1000 MHz with LoRa® Technology*, Semtech, 2019, rev.4. [xv](#), [59](#), [60](#)
- [12] O. Mendoza Cano, R. Aquino Santos, J. López de la Cruz, M. Edwards, A. Khouakhi, I. Pattison, V. Rangel Licea, E. Castellanos Berjan, A. Martinez Preciado, P. Rincón Avalos, P. Lepper, A. Gutiérrez Gómez, J. Uribe Ramos, J. Ibarreche, I. Perez, G. Sankar, and Lawrence, “Experiments of an iot-based wireless sensor network for flood monitoring in colima, mexico.” in *Journal of Hydroinformatics*, vol. 23, no. 3, 2021. [2](#)
- [13] EMDAT, “Emdat: The emergency events database.” <https://www.emdata.be>, 2020. [2](#)
- [14] Constantino, R. M, and H. R. Dávila, “Una aproximación a la vulnerabilidad y la resiliencia ante eventos hidrometeorológicos extremos en méxico,” *Política y cultura*, no. 36, pp. 15–44, 2011. [2](#)
- [15] M. Centenaro, L. Vangelista, A. Zanella, and M. Zorzi, “Long-range communications in unlicensed bands: the rising stars in the iot and smart city scenarios,” *IEEE Wireless Communications*, vol. 23, no. 5, pp. 60–67, 2016. [11](#)
- [16] L. Atzori, A. Iera, and G. Morabito, *The internet of things: A survey*. Comput. Netw., vol. 54, no. 15, pp. 2787-2805, 2010. [11](#)
- [17] C. C. Aggarwal, N. Ashish, and A. Sheth, *The Internet of Things: A Survey From the Data-Centric Perspective*. Boston, MA, USA:Springer, pp. 383-428, 2013. [12](#)
- [18] G. A. Akpakwu, B. J. Silva, G. P. Hancke, and A. M. Abu-Mahfouz, *A survey on 5G networks for the Internet of Things: Communication technologies and challenges*. IEEE Access, vol. 6, pp. 3619-3647, 2018. [13](#)

- [19] R. Edwards, R. A. Santos, P. Lepper, O. Mendoza-Cano, I. Pattison, V. Rangel, F. Jimenez, E. Adegoke, B. Clark, J. Davis, and et al., “Emergency water information network (ewin),” Aug 2019. [Online]. Available: <https://hdl.handle.net/2134/9216746.v1> 16
- [20] O. Mendoza-Cano, R. Aquino-Santos, J. López-De la Cruz, R. M. Edwards, A. Khouakhi, I. Pattison, V. Rangel-Licea, E. Castellanos-Berjan, M. Martinez-Preciado, P. Rincón-Avalos *et al.*, “Experiments of an iot-based wireless sensor network for flood monitoring in colima, mexico,” *Journal of Hydroinformatics*, vol. 23, no. 3, pp. 385–401, 2021. 16, 18
- [21] D. Magrin, M. Capuzzo, and A. Zanella, *A Thorough Study of LoRaWAN Performance Under Different Parameter Settings*. IEEE Internet of Things Journal., 2019. 17, 18
- [22] I. Pattison, A. Khouakhi, R. Edwards, P. Lepper, O. Mendoza-Cano, J. Lopez de la Cruz, R. Aquino, and V. Rangel, “Development of a flood forecasting system in developing country contexts using low cost hydrological sensors and mobile phone networks,” in *EGU General Assembly Conference Abstracts*, 2018, p. 10671. 18
- [23] J. Ibarreche, R. Aquino, R. Edwards, V. Rangel, I. Pérez, M. Martínez, E. Castellanos, E. Álvarez, S. Jimenez, R. Rentería *et al.*, “Flash flood early warning system in colima, mexico,” *Sensors*, vol. 20, no. 18, p. 5231, 2020. 18
- [24] J. Elshabrawy, T. and Robert, *Closed-form approximation of lora modulation ber performance*. IEEE Communications Letters, vol. 22, no. 9, pp. 1778–1781, 2018. 18, 42
- [25] A. Gutiérrez-Gómez, V. Rangel, R. Edwards, J. Davis, R. Aquino, J. López, O. Mendoza-Cano, M. Lopez-Guerrero, and Y. Geng, “A propagation study of lora p2p links for iot applications:the case of near-surface measurements over semitropical rivers in mexico,” *Sensors*, vol. 21, 2021. 20, 92
- [26] U. Raza, P. Kulkarni, and M. Sooriyabandara, “Low power wide area networks: An overview,” *IEEE Communications Surveys Tutorials*, vol. 19, no. 2, pp. 855–873, 2017. 21
- [27] M. Capuzzo, D. Magrin, and A. Zanella, *Confirmed traffic in LoRaWAN: Pitfalls and countermeasures*. 2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), Capri, 2018. 21

BIBLIOGRAFÍA

- [28] Devalal, Shilpa, and A. Karthikeyan, “Lora technology - an overview,” in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 2018, pp. 284–290. [22](#)
- [29] L. Vangelista, “Frequency shift chirp modulation: The lora modulation,” *IEEE Signal Processing Letters*, vol. 24, no. 12, pp. 1818–1821, 2017. [23](#)
- [30] L. P. Fraile, S. Tsampas, G. Mylonas, and D. Amaxilatis, “A comparative study of lora and ieee 802.15.4-based iot deployments inside school buildings,” *IEEE Access*, vol. 8, pp. 160 957–160 981, 2020. [24](#)
- [31] A. Berni and W. Gregg, “On the utility of chirp modulation for digital signaling,” *IEEE Transactions on Communications*, vol. 21, no. 6, pp. 748–751, 1973. [25](#)
- [32] A. Zourmand, A. L. Kun Hing, C. Wai Hung, and M. AbdulRehman, “Internet of things (iot) using lora technology,” in *2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, 2019, pp. 324–330. [25](#), [27](#)
- [33] L. Alliance, “Lorawan specification v1. 1”, 2017,” 2017. [28](#), [29](#)
- [34] Aras, Emekcan, Ramachandran, G. Sankar, Lawrence, Piers, Hughes, and Danny, “Exploring the security vulnerabilities of lora,” in *2017 3rd IEEE International Conference on Cybernetics (CYBCONF)*, 2017, pp. 1–6. [33](#)
- [35] L. Alliance, “Lorawan™ specification” lora™ alliance,” *Tech. Rep.*, 2015. [33](#), [35](#)
- [36] A. Semtech, “Lora™ modulation basics, application note,” 22. [36](#)
- [37] V. Erceg, L. Greenstein, S. Tjandra, S. Parkoff, A. Gupta, B. Kulic, A. Julius, and R. Bianchi, “An empirically based path loss model for wireless channels in suburban environments,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 7, pp. 1205–1211, 1999. [38](#)
- [38] M. R. Stiglitz and C. Blanchard, 1992. [39](#)
- [39] C. G. Cassandras and S. Lafortune. Springer, 2009. [39](#)
- [40] Ramadge and W. Wonham, “The control of discrete event systems,” *Proceedings of the IEEE*, vol. 77, no. 1, pp. 81–98, 1989. [40](#)
- [41] Ivey, S. Jared, Swenson, P. Brian, Riley, and F. George, “Simulating networks with ns-3 and enhancing realism with dce,” in *2017 Winter Simulation Conference (WSC)*, 2017, pp. 690–704. [40](#)

- [42] Henderson, R. Thomas, Lacage, Mathieu, Riley, F. George, Dowell, Craig, Kopena, and Joseph, “Network simulations with the ns-3 simulator,” *SIGCOMM demonstration*, vol. 14, no. 14, p. 527, 2008. 40
- [43] C. S. Magirl and J. W. Gartner, *Water velocity and the nature of critical flow in large rapids on the Colorado River, Utah*, 2009. 69
- [44] J. Ibarreche, R. Aquino, R. M. Edwards, V. Rangel, I. Pérez, M. Martínez, E. Castellanos, E. Álvarez, S. Jimenez, R. Rentería, and et al., “Flash flood early warning system in colima, mexico,” *Sensors*, vol. 20, no. 18, p. 5231, Sep 2020. [Online]. Available: <http://dx.doi.org/10.3390/s20185231> 82
- [45] Singh, Deepa, Pandey, Bishwajeet, G. Tomar, and B. Sarkar, “Performance evaluation of backoff method – effect of backoff factor on exponential backoff algorithm,” in *2013 5th International Conference and Computational Intelligence and Communication Networks*, 2013, pp. 82–86. 83

ACK Acknowledgement. [5](#), [6](#), [48](#), [54](#), [61](#)

BER Bit Error Ratio. [5](#), [16](#), [18](#)

BLE Bluetooth Low Energy. [13](#)

BW Bandwidth. [27–29](#)

CR Coding Rate. [27](#), [28](#), [30](#), [31](#), [33](#), [34](#)

CRC Cyclic Redundancy Check. [25](#), [34](#)

CSS Chirp Spread Spectrum. [24](#)

DE Data rate optimization Enable. [34](#)

DSSS Direct Sequence Spread Spectrum. [25](#)

EPSRC Engineering and Physical Sciences Research Council. [2](#)

EWIN Emergency Water Information Network. [2–4](#), [6](#), [7](#), [13](#), [20](#), [21](#), [36](#), [55](#), [56](#), [58](#)

FHSS Frequency Hopping Spread Spectrum. [25](#)

GCRF Global Challenges Research Fund. [2](#)

H Header. [34](#)

IEEE Institute of Electrical and Electronics Engineers. [13](#)

IoT Internet of Things. [1](#), [3](#), [4](#), [6](#), [9–13](#)

LAN Local Area Network. [14](#)

LoRa Long Range. [1–4](#), [6](#), [7](#), [10](#), [11](#), [14](#), [17](#), [18](#), [21–28](#), [33](#), [41](#), [42](#), [47](#), [55](#)

LPAN Low Power Area Networks. [11](#)

LPWAN Low Power Wide Area Networks. [11](#), [13](#), [14](#)

LTE Long Term Evolution. [14](#)

MAC Media Access Control. [4](#), [6](#), [7](#), [41](#), [48](#), [54](#), [55](#)

NFC Near Field Communication. [13](#)

NS-3 Network Simulator 3. [1](#), [4](#), [5](#), [7](#), [16–18](#), [36](#), [40](#), [58](#)

PER Packet Error Rate. [29](#)

PL Payload. [34](#)

RFID Radio Frequency Identification. [13](#)

SF Spreading Factor. [26–28](#), [30](#), [33](#), [34](#), [36](#), [91](#)

SNR Signal-to-Noise Ratio. [18](#), [19](#)

ToA Time on Air. [26](#), [33](#), [46](#), [61](#)

WPAN Wide Personal Area Network. [14](#)

WWAN Wireless Wide Area Network. [14](#)

WWW World Wide Web. [12](#)