

INSTITUTO POLITÉCTICO NACIONAL -
UNIDAD PROFESIONAL INTERDISCIPLINARIA DE
INGENIERÍA Y TECNOLOGÍAS AVANZADAS



PROYECTO TERMINAL:

**SIMULADOR DE MODELOS DE
TRÁFICO PARA NODOS IOT EN
UNA RED CELULAR DE 5G**

Autores:

Luis Fernando SALAZAR ORDOÑEZ
Rolando SOTELO ALARCON

Asesores:

Dr. Domingo LARA RODRIGUEZ
Dr. Noe TORRES CRUZ

*Una tesis presentada en cumplimiento de los requisitos
para el grado de Ingeniería en Telemática*

CDMX, Julio 2020

Resumen

En este documento se presenta el desarrollo de un simulador a nivel de sistema, programado bajo el paradigma de eventos discretos, que permite modelar el servicio que la red de comunicación celular de quinta generación (5G), ofrece a nodos de Internet de las cosas (IoT). El simulador se enfocó en el caso de uso mIoT, el cual comprende principalmente de nodos IoT estáticos de baja complejidad que además se encuentran en gran cantidad dentro de los escenarios de esta red. La arquitectura del simulador contempló cuatro módulos clave para su ejecución: un modelo de despliegue de UEs, un modelo de canal, un esquema de acceso múltiple al medio no ortogonal y modelos de tráfico adecuados para modelar distintos servicios. Asimismo, se consideró el fundamentar la fiabilidad de los resultados obtenidos por el simulador mediante la previa prueba e implementación de modelos de tráfico ya estudiados en la literatura concerniente al desempeño de sistemas celulares. Con los resultados del simulador se determinaron qué configuraciones y parámetros iniciales de la arquitectura de red propuesta satisfacen una óptima calidad de servicio (QoS) para el caso de uso mIoT.

Palabras clave: mMTC, NB-IoT, PD-NOMA, 5G, simulador de eventos discretos, QoS.

Índice general

Resumen	III
1. Introducción	1
1.1. ANTECEDENTES	1
1.2. PLANTEAMIENTO DEL PROBLEMA	2
1.3. OBJETIVOS	3
1.3.1. OBJETIVO GENERAL	3
1.3.2. OBJETIVOS ESPECÍFICOS	4
1.4. JUSTIFICACIÓN	4
1.5. PROPUESTA DE SOLUCIÓN	5
1.6. ALCANCES	6
2. Marco Teórico	9
2.1. MODELADO DEL DESPLIEGUE CELULAR	9
2.1.1. Despliegue aleatorio	10
2.1.2. Despliegue determinístico	12
2.1.3. Interferencia en los sistemas de comunicaciones	14
2.1.4. Capacidad en los sistemas de comunicaciones	16
2.2. MODELADO DEL CANAL CELULAR	17
2.2.1. Relaciones Generales de Propagación	18
2.2.2. Pérdida por trayectoria en el Espacio Libre	19
2.2.3. Caracterización del canal de radio	20
2.3. ESQUEMAS DE ACCESO MÚLTIPLE AL MEDIO	23
2.3.1. OMA	23
2.3.2. NOMA	24
2.4. GENERACIONES DE LOS SISTEMAS CELULARES	26
2.5. SISTEMAS CELULARES 5G	27
2.6. MODELADO DEL TRÁFICO EN TELECOMUNICACIONES	29
2.7. SIMULACIONES A NIVEL DE SISTEMA	32
2.7.1. Lenguajes de Programación para simulaciones orientadas a eventos discretos(DES)	33
2.7.2. Python	34
2.7.3. Multiprocesamiento	34
3. Estado del Arte	37

4. Análisis	41
4.1. REDES 5G/IoT	41
4.2. CLASIFICACIÓN Y ANÁLISIS DE LOS ÁMBITOS DE IoT	41
4.2.1. Ciudades inteligentes (<i>Smart Cities</i>):	42
4.2.2. Ambiente inteligente (<i>Smart Environment</i>)	42
4.2.3. Energía inteligente (<i>Smart Energy</i>):	42
4.2.4. Transporte y movilidad inteligentes (<i>Smart Transport and Mobility</i>):	45
4.3. CARACTERÍSTICAS DEL ESCENARIO A IMPLEMENTAR	45
4.3.1. Análisis de las aplicaciones de IoT y selección de casos considerados	46
4.3.2. Análisis de las tecnologías para IoT y selección de casos considerados	47
4.4. ANÁLISIS DEL ESTÁNDAR NB-IoT	50
4.4.1. Modos de operación	50
4.4.2. Bandas de frecuencia	51
4.4.3. Clases de Potencia	52
4.4.4. Modos de transmisión en enlace ascendente (UL)	52
4.4.5. Características del tráfico IoT	53
Informes de excepción	53
Informes periódicos	53
4.4.6. Indicadores clave de rendimiento (KPIs)	54
4.5. ANÁLISIS DE MODELOS PARA LA EVALUACIÓN DE REDES 5G/IoT	57
4.5.1. MODELO DE DESPLIEGUE DE BSs Y UEs	57
4.5.2. MODELO DE CANAL	58
4.5.3. ESQUEMA DE ACCESO MÚLTIPLE AL MEDIO	60
Acceso Múltiple No Ortogonal (NOMA)	60
4.5.4. MODELOS DE TRÁFICO	62
Modelos de tráfico agregado	63
Modelos de tráfico fuente	65
5. Diseño	69
5.1. MODELO DE SISTEMA	69
5.1.1. Uso de una geometría estocástica, es decir, despliegue de UEs siguiendo un PPP	70
5.1.2. Pérdidas de canal usando un modelo CI para ambientes exteriores	70
5.1.3. Incorporación de un esquema de acceso al medio no ortogonal (NOMA) usando una técnica de agrupamiento de usuarios	71
5.1.4. Diferentes modelos de tráfico que simulen distintos servicios para NB-IoT	72
5.2. Parámetros generales del simulador	76
6. Implementación	79
6.1. Generación de una geometría estocástica	79
6.1.1. Escenario de Prueba de PPP	80
6.2. Generación de ganancias de canal	80
6.2.1. Escenario de Prueba del Modelo de canal CI	81
6.3. Esquema de acceso múltiple al medio no ortogonal, basado en potencia (PD-NOMA)	83

6.3.1. Algoritmo de Agrupación de dispositivos uRLLC y mMTC	83
6.3.2. Algoritmo de Asignacion de Subportadoras	85
6.3.3. Escenario de Prueba de NOMA	88
6.4. Generación de Tráfico Fuente	91
6.4.1. Tráfico CMMPP	91
6.4.2. Tráfico Periódico	93
6.4.3. Algoritmo de generación de tráfico fuente	93
6.4.4. Escenario de Prueba para la Generación de Tráfico	95
6.5. Interconexión de los 4 módulos del Simulador	99
6.6. Generador de tráfico IoT	101
6.6.1. Interfaz de Usuario	101
6.6.2. Archivos de Salida	102
6.7. Simulador de Eventos Discretos	104
6.7.1. Definición de eventos	105
6.7.2. Descripción de los <i>logs</i> de salida	106
6.7.3. Resultados a la salida del simulador	107
6.8. Optimización de tiempos de simulación	108
7. Resultados	111
7.1. Escenario I	111
7.1.1. Descripción del escenario	111
7.1.2. Parámetros de entrada	111
7.1.3. Resultados obtenidos	111
7.2. Escenario II	120
7.2.1. Descripción del escenario	120
7.2.2. Parámetros de entrada	121
7.2.3. Resultados obtenidos	121
8. Conclusiones	127
A. Distribuciones estadísticas en Telecomunicaciones	129
A.1. Generación de números aleatorios	130
A.2. Generación de distribuciones estadísticas para el simulador	131
A.2.1. Generación de variable aleatoria tipo <i>Poisson</i>	131
A.2.2. Generación de variables aleatorias tipo <i>Exponencial</i> y <i>Rayleigh</i>	132
A.2.3. Generación de variable aleatoria tipo <i>Pareto</i>	133
Bibliografía	135

Índice de figuras

2.1. Diferentes estrategias de despliegue para BSs	10
2.2. Localización de celdas co-canal con distintos factores de reúso	14
2.3. Procedimiento de generación de coeficientes de canal	18
2.4. Propagación de señales celulares en ambientes urbanos	20
2.5. Ejemplo de niveles de señal con pérdidas por trayectoria, desvanecimiento lento y desvanecimiento rápido	21
2.6. Cancelación Sucesiva de Interferencia (SIC)	24
2.7. Ejemplo del esquema NOMA en un enlace de bajada con dos usuarios y una sub-portadora.	25
2.8. Ejemplo del espectro compartido para OFDMA y NOMA con dos usuarios.	26
2.9. Diferentes tipos de acceso múltiple al medio ocupados en generaciones anteriores (1G, 2G y 3G)	27
2.10. Comparación de diversos escenarios de uso de la tecnología 5G por la UIT-R, el 3GPP y la NGMN	28
2.11. El diseño usual de una red celular en simulaciones a nivel de sistema	33
2.12. Concurrencia Vs Paralelismo	35
 3.1. Modelo de sistema para el sistema de enlace descendente NOMA	38
3.2. Modelo de sistema para el sistema de enlace ascendente NOMA	38
3.3. Grupos NOMA que incluyen dispositivos mMTC y URLLC, donde los dispositivos MTC comparten los subcanales asignados a cada clúster NOMA.	40
 4.1. 5G NR con LTE-M y NB-IoT en banda	49
4.2. Tecnologías líderes para el caso de uso LPWA	51
4.3. 5G NR soportará múltiples servicios con distintos requerimiento de QoS	52
4.4. Estructura de ancho de banda y subportadoras en NB-IoT.	53
4.5. Modos de Operación en NB-IoT.	54
4.6. Estructura de los estados principales del tráfico M2M	64
4.7. Cadena de Markov del modelo SMM	66
4.8. Modelo CMMPP en dispositivos MTC	67
 5.1. Diagrama general de bloques del simulador, constando principalmente de 4 módulos.	70
5.2. Cadena de Markov propuesta	73
5.3. Generación de tráfico con modelo CMMPP	74
5.4. Ejemplo ilustrativo de un escenario mIoT unicelular aproximado, usando agrupaciones de 4 dispositivos	76

6.1.	Distribución espacial de puntos PPP en un círculo 2D	80
6.2.	Distribución espacial de puntos PPP en un círculo 2D	80
6.3.	Pérdidas por trayectoria según el Modelo CI	81
6.4.	Ganancias de Canal Modelo CI	82
6.5.	Ejemplo ilustrativo del ordenamiento de usuarios en los grupos NOMA	85
6.6.	Ejemplo ilustrativo de la distribución de potencia para PD-NOMA en <i>Multitone</i>	86
6.7.	Coeficientes de canal del modelo de sistema en [37]	89
6.8.	Escenario de pruebas de esquema NOMA	90
6.9.	Segundo escenario de pruebas de esquema NOMA	91
6.10.	Histograma de nacimientos de paquetes	95
6.11.	Otros dispositivos mMTC, Histograma de nacimientos y sus distribuciones	98
6.12.	Semáforos inteligentes, Histograma de nacimientos y sus distribuciones	99
6.13.	Propagación de alarma en dispositivos de control de iluminación	100
6.14.	Propagación de alarma en otros dispositivos mMTC	101
6.15.	Interconexión de los 4 módulos en un simulador	102
6.16.	Interfaz de Usuario, Generador de Tráfico IoT	103
6.17.	Archivo de dispositivos	103
6.18.	Archivo de eventos	104
6.19.	Simulación de eventos discretos	105
6.20.	Procesamiento del evento: nacimiento de paquete	106
6.21.	Procesamiento del evento: proceso noma	107
6.22.	<i>logs</i> resultantes de una rutina	108
6.23.	Gráfica de comparación de tiempos de simulación	109
7.1.	Modelo NOMA en un TTI con modelo de canal CI	112
7.2.	Resultados generales Escenario I	113
7.3.	Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo, kmax 4 . .	115
7.4.	Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo (%), kmax 4	116
7.5.	Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo, kmax 3 . .	117
7.6.	Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo (%), kmax 3	118
7.7.	Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo, kmax 2 . .	119
7.8.	Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo (%), kmax 2	120
A.1.	Relaciones entre algunas de las distribuciones de probabilidad univariadas	131
A.2.	Gráfica comparación PMF e histograma de distribución Poisson en Python	132
A.3.	Gráfica comparación PDF e histograma de distribución Exponencial en Python .	133
A.4.	Gráfica comparación PDF e histograma de distribución Rayleigh en Python . . .	134
A.5.	Gráfica comparación PDF e histograma de distribución Pareto en Python	134

Índice de tablas

4.1. Características de las aplicaciones de Ciudades Inteligentes	43
4.2. Características de las aplicaciones de Ambiente Inteligente	44
4.3. Características de las aplicaciones de Energía Inteligente	44
4.4. Características de las aplicaciones de Transporte y Movilidad inteligentes	45
4.5. Aplicaciones seleccionadas para la simulación	47
4.6. Características de las tecnologías de red para IoT en la red celular	50
4.7. Caracterización del tráfico de paquetes en aplicaciones seleccionadas para la simulación.	55
4.8. Modelos de tráfico agregado propuestos por la 3GPP para comunicaciones M2M	64
4.9. Comparativa entre los modelos de tráfico MTC abordados	67
5.1. Parámetros Modelo CI [Fuente: [48]]	71
5.2. Caracterización de las aplicaciones seleccionadas	75
5.3. Parámetros de la simulación a nivel de sistema	77
6.1. Parámetros necesarios para generar tráfico CMMPP	93
6.2. Parámetros necesarios para generar tráfico Periódico	93
6.3. Parámetros para el escenario de pruebas de tráfico	96
6.4. Parámetros para el escenario de pruebas de tráfico	97

Lista de Acrónimos

3GPP	3rd Generation Partnership Project
4G	Cuarta generación de Comunicaciones Móviles
5G NR	5G New Radio
ARIB	Association of Radio Industries and Businesses
ARPU	Average Revenue Per User
ATIS	Alliance for Telecommunications Industry Solutions
AWGN	Additive White Gaussian Noise
apd	Average Power Decay (a.k.a. PLE)
BS	Base Station
BSs	Base Stations
BW	Bandwidth
CCSA	China Communications Standards Associations
CD-NOMA	Code Division NOMA
CIoT	Cellular IoT
CMMPP	Coupled Markov Modulated Poisson Process
CT	Central Terminal
DES	Discrete Event Simulation
DL	DownLink
ED	Event-Driven
EIRP	Effective Isotropic Radiated Power
eMBB	Enhanced Mobile Broadband
eMTC	Enhanced Machine Type Communications
ETSI	European Telecommunications Standards Institute
EUTRA	Evolved UMTS Terrestrial Radio Access
fc	Frecuencia Portadora
GIL	Global Interpreter Lock
HPPP	Homogeneous Poisson Point Process
HTC	Human Type Communication
IEEE	Institute of Electrical and Electronics Engineers
IFT	Instituto Federal de Telecomunicaciones
InH	Indoor Hotspot
IoT	Internet of Things
ITU	International Telecommunication Union
ITU-R	ITU Radiocommunications
KPI	Key Performance Indicator
KPIs	Key Performance Indicators

LoS	Line of Sight
LPWAN	Low Power Wide Area Networks
LSP	Large-Scale Parameters
LTE	Long Term Evolution
LTE-M	LTE-MTC
M2M	Machine to Machine
MA	Multiple Access
MAC	Media Access Control
MIMO	Multiple Inputs Multiple Outputs
mIoT	Massive IoT
MMPP	Markov Modulated Poisson Process
mMTC	Massive Machine Type Communications
MS	Mobile Station
MT	Mobile Terminal
MTC	Machine Type Communication
MTs	Mobile Terminals
M2M	Machine to machine
NB-IoT	Narrow-Band Internet of Things
NGMN	Next Generation Mobile Networks
NLoS	Non-Line of Sight
NOMA	Non-Orthogonal Multiple Access
OFDMA	Orthogonal Frequency Division Multiple Access
OMA	Orthogonal Multiple Access
PD-NOMA	Power Division NOMA
PE	Payload Exchange
PL	Path Loss
PLE	Path Loss Exponent
PPP	Poisson Point Process
PRB	Physical Resource Block
PSM	Power Saving Mode
PU	Periodic Update
QoS	Quality of Service
RAN	Radio Access Networks
RF	Radio Frequency
RFID	Radio Frequency Identification
Rx	Receiver
SA	Stand-alone
SA 2	System Aspects
SC-FDMA	Single Carrier Frequency Division Multiple Access
SF	Shadow Fading
SIC	Successive Interference Cancelation
SINR	Signal to Interference plus Noise Ratio
SISO	Single Input Single Output
SM	Suburban Macro

SMM	Semi-Markov Model
TDD	Time Division Duplexing
TDMA	Time Division Multiple Access
TTI	Transmission Time Interval
TR	Technical Report
TSDO	Telecommunications Standards Development Organization
TSDOs	Telecommunications Standards Development Organizations
TSDSI	Telecommunications Standards Development Society, India
TSG	Technical Specifications Group
TTA	Telecommunications Technology Association of Korea
TTC	Telecommunication Technology Committee
Tx	Transmitter
UE	User Equipment
UL	UpLink
UMa	Urban Macro
UMi	Urban Micro
UMTS	Universal Mobile Telecommunications System
URLLC	Ultra-Reliable and Low-Latency Communications

Capítulo 1

Introducción

1.1. ANTECEDENTES

Imaginar nuestra vida sin los beneficios brindados por los sistemas de comunicación de hoy en día, o tan sólo sin la tecnología presente en este ámbito desde los últimos 20 años es ya muy difícil, y esto se debe a que en el presente una gran parte de las tareas y actividades, muchas de ellas cruciales para el funcionamiento de nuestras sociedades, operan eficientemente si y sólo si se está propiamente conectado y en facultades de compartir información [1].

Los sistemas de comunicación celular han tenido saltos generacionales desde la conocida como primera generación, la cual saldría al mercado a finales de la década de los 70's e inicio de los 80's, hasta el presente con el desarrollo de la próxima generación (5G), la cual comenzará su implementación en el año 2020. En [1] comentan que: "La primera y segunda generación de comunicaciones móviles estuvieron dominadas por señales analógicas de audio y posteriormente señales digitales de audio y texto. La tercera generación se trató más de escalar el número de usuarios en la red [...] pero fue abrumada por un tsunami de contenido de imágenes y videos."

Cada uno de estos saltos de generación ha estado motivado por distintos requerimientos de servicio, necesidades de los usuarios y la aparición de nuevas tecnologías que han buscado ser una vía para facilitar la comunicación entre individuos de todo el mundo y ahora, más recientemente, la comunicación entre máquinas.

El aumento de la tasa de transmisión de datos ha sido siempre un factor a tener en cuenta para el desarrollo de los estándares de las nuevas generaciones de redes móviles, por ejemplo, para la nueva generación se espera "un pico de transmisión de al menos 1 Gb/s al tiempo de su introducción en 2020, esperando que crezca hasta los 10Gb/s para 2025", [2]. Pero el sistema de comunicaciones móviles de quinta generación ha estado, además de eso, motivado por un mayor volumen de transmisión de datos, un incremento radical en la cantidad de dispositivos conectados a la red, una menor latencia y una mayor duración de batería para los dispositivos de bajo consumo.

Las limitaciones presentes hasta ahora para las comunicaciones celulares del internet de las cosas (*Internet of Things - IoT*), se deben principalmente a que la red de comunicación móvil fue creada para voz y aplicaciones de texto, evolucionando eventualmente a una transmisión

de archivos como imágenes y videos predominante en el enlace de bajada. Por otro lado, la comunicación de dispositivos IoT tiene un conjunto de requerimientos muy distintos. Pero es ahora con la quinta generación que se promete brindar las herramientas que esta tecnología necesita para alcanzar su máximo potencial. 5G se trata entonces de la propuesta de crear una red de comunicaciones que logre implementar tanto los servicios inherentes a las necesidades de comunicación entre los humanos y aquellas necesidades de comunicación entre las máquinas. El cumplir con estas últimas necesidades, aseguraría brindar una mejor calidad de servicio para la nueva ola de dispositivos de IoT que se espera estén ya conectados a la red para 2020.

Este proyecto presenta el diseño de un simulador de eventos discretos, el cual modeló el servicio prestado por la arquitectura de red celular que aquí se propone a nodos IoT. Su arquitectura contempló la próxima generación móvil a implementarse (5G) y los servicios seleccionados para atender a aplicaciones del caso de uso del internet de las cosas masivo (*massive IoT - mIoT*). Este simulador se enfocó en el tráfico generado por dispositivos IoT de banda estrecha (*Narrow-Band IoT - NB-IoT*) y con los resultados obtenidos se esclareció qué configuraciones de red podrían mejorar significativamente la calidad de servicio.

1.2. PLANTEAMIENTO DEL PROBLEMA

En los recientes años se ha estado presenciando la definición de la tecnología de comunicación móvil 5G en estándares, para su posterior introducción a partir de 2020, y como se resalta en [2] la nueva generación no sólo seguirá la línea de incrementar la velocidad de transmisión como se ha venido haciendo en cada salto generacional, sino que también traerá consigo la posibilidad de una conectividad adicional sin precedentes, todo esto motivado por la cantidad masiva de dispositivos de IoT que se esperan. Según *Ericsson Mobility Report* [3], 22.3 mil millones de dispositivos en el 2024, pertenecerán a una aplicación de IoT. De manera que 5G dará servicio a una enorme cantidad de dispositivos IoT, cada dispositivo enviará pequeños paquetes de datos a lo largo de mucho tiempo, además de que existirá una cierta sincronía en el tráfico generado. Lo anterior ha provocado el desarrollo de nuevas tecnologías que proponen distintas formas de agrupamiento de estos nodos, distintas formas de acceder a los recursos, distintas propuestas de comunicación entre ellos y nuevas formas de que ahorren energía. Siempre teniendo en cuenta los parámetros clave de desempeño (*Key Performance Parameters - KPI*) de la red, como dar servicio a una cantidad masiva de nodos, la duración de la batería de estos y la menor latencia en comunicaciones críticas¹, como se menciona en [4].

Esto ha resultado en nuevos retos para la implementación de la red, para la cual se desarrolla tecnología o se mejora la ya existente. En [5], se presenta la tecnología NB-IoT, originalmente creada como una solución que brindara servicio a nodos IoT en LTE. Esta tecnología formará

¹ Comunicaciones que requieren de una urgente respuesta debido a su naturaleza, por ejemplo los coches auto-dirigidos.

parte de los estándares de 5G, como 3GPP² lo ha indicado a la Unión Internacional de Telecomunicaciones (*International Telecommunication Union - ITU*). Se pretende que con esta tecnología y algunas mejoras, la red 5G sea capaz de brindar servicio a aplicaciones del caso de uso de comunicaciones tipo maquina masivo (*massive Machine Type Communications - mMTC*)³, para el cual se esperan tener decenas de miles de dispositivos conectados por celda.

En el futuro, los escenarios de IoT, tendrán una enorme cantidad de dispositivos conectados en comparación con los actuales escenarios de la red 4G [6]. Por lo tanto, las tecnologías de 5G deberán brindar servicio a muchos dispositivos usando recursos limitados. Pero dada la elevada complejidad con la que el modelo de un sistema de comunicación como la red 5G puede contar, si de éste se quieren obtener resultados útiles, resulta casi imposible el analizar su comportamiento sino a través de una simulación.

Los patrones de tráfico de nodos de IoT varían según su caso de uso, los cuales se dividen normalmente en tres: comunicaciones ultra confiables y de baja latencia (*ultra Reliable and Low Latency Communications - uRLLC*), comunicaciones tipo maquina masivo (mMTC) y comunicaciones con un ancho de banda mejorado (*enhanced Mobile Broadband - eMBB*). Para las comunicaciones tipo maquinas masivo (mMTC), en el cual se enfocará el proyecto, se tienen nodos en su amplia mayoría estáticos y como se ve en [7], para estos nodos podemos considerar por los menos dos patrones de tráfico: el periódico, y el aleatorio ,de manera que el modelo de tráfico deberá tener en consideración esto. Como se puede apreciar en [7], un problema crítico del Internet de las cosas masivo (mIoT) en las redes móviles es que los dispositivos de IoT causarán una gran congestión en ésta si es que no se incorporan mejoras en las arquitecturas de dichas redes. Este problema se ve acrecentado debido a que el tráfico de los nodos mIoT presenta cierta sincronía espacial y temporal dependiendo de la aplicación a la que pertenezcan.

Por lo anterior, para la red 5G se necesitan realizar simulaciones en sus distintos casos de uso, que generen resultados sobre qué arquitectura de red y tecnologías brindan un mejor desempeño.

1.3. OBJETIVOS

1.3.1. OBJETIVO GENERAL

Diseñar e implementar un simulador de teleráfico para el ambiente de Internet de las cosas masivo (mIoT) en una red celular de quinta generación (5G), por medio de la programación de eventos discretos, con la finalidad de evaluar el desempeño de esta red en términos de la cantidad de recursos requeridos para satisfacer niveles esperados de calidad de servicio (*Quality of Service - QoS*).

² The 3rd Generation Partnership Project (3GPP), desarrolladores del estándar NB-IoT y el 5G NR

³Cabe mencionar que los casos de uso o tecnologías mIoT y mMTC son análogas.

1.3.2. OBJETIVOS ESPECÍFICOS

- Determinar el escenario a implementar, mediante el análisis de requerimientos, para definir después los parámetros de entrada en el modelado del simulador.
- Seleccionar los modelos (despliegue, propagación, técnica de acceso múltiple) aplicables a una simulación a nivel de sistema para la comunicación entre nodos IoT y la red 5G.
- Seleccionar y determinar los modelos de tráfico a analizar para nodos IoT que hayan sido propuestos en la red 5G, mediante la lectura de distintas publicaciones en revistas científicas, para simular el modelo más adecuado según los alcances propuestos.
- Determinar los parámetros de desempeño de la red (KPIs), mediante el análisis de distintas publicaciones científicas, con la finalidad de establecer métricas de QoS.
- Definir el procedimiento de la simulación mediante la especificación de su arquitectura y elaboración de diagramas de su funcionamiento y procesos, con el fin de integrar una metodología para su implementación.
- Implementar los modelos y protocolos que definen a la comunicación entre los nodos IoT y la red 5G, mediante algoritmos computacionales (incluyendo la técnica de eventos discretos) y de acuerdo a la arquitectura previamente definida.
- Implementar una técnica de paralelismo, mediante el uso de multiprocesamiento, con la finalidad de reducir los tiempos de ejecución de las simulaciones.
- Evaluar y analizar cada modelo analítico con el uso de escenarios de pruebas y calibración para poder realizar comparaciones con los resultados teóricos esperados y verificar la fiabilidad del simulador
- Simular el modelo de sistema propuesto, mediante la variación de los parámetros de entrada, para caracterizar el desempeño del sistema en términos del tráfico que se puede ofrecer y la cantidad de recursos requeridos para satisfacer objetivos de QoS.

1.4. JUSTIFICACIÓN

En la gran mayoría de los trabajos de investigación revisados, se han realizado estudios de rendimiento de los sistemas de comunicación móvil de quinta generación. Estos se concentran en evaluar distintos modelos, frecuentemente se considera, un modelo de distribución de BSs y UEs, uno de canal, y un esquema de acceso multiple, sin embargo no hay mucha investigación acerca de incorporar todos estos componentes en conjunto con modelos de tráfico para el caso específico de mIoT. La aportación que se pretende hacer con este proyecto se encuentra precisamente en ese ámbito.

Los enlaces de comunicación inalámbrica experimentan fenómenos físicos perjudiciales al canal como lo son las multi-trayectorias y los desvanecimientos debido a grandes objetos que se interceptan en la trayectoria de la propagación, además el rendimiento de los sistemas celulares

inalámbricos tiende a limitarse debido a la interferencia de otros usuarios. Estas condiciones complejas del canal son difíciles de describir con un modelo analítico, es por esto que las aproximaciones de las simulaciones son necesarias. Estas permiten analizar el rendimiento de los enlaces de comunicaciones celulares [8], modelando un gran número de eventos aleatorios a través del tiempo, mediante el uso de simulaciones orientadas a eventos discretos.

Una simulación permite observar muchas de las interacciones de un sistema, que de otra forma tomaría mucho trabajo predecirlas o calcular, además de que proporciona un método importante de análisis, que resulta sencillo de comunicar y comprender. En todas las industrias y disciplinas, la creación de simulaciones brinda soluciones valiosas al proporcionar información clara sobre sistemas complejos [8]. Los resultados de una simulación que haga las suposiciones adecuadas y modele correctamente el sistema propuesto, brindarán confianza y claridad, ahorrarán tiempo y muy posiblemente también dinero.

Este proyecto servirá como referencia a los investigadores y estudiantes que busquen comparar los modelos y técnicas propuestos en este trabajo con otras selecciones posibles para la futura red 5G y el servicio brindado a los nodos IoT.

Por otra parte, se debe enfatizar que el desarrollo de este proyecto requiere de la aplicación de conocimientos relacionados a la informática (entre los que se incluyen desarrollo de software y algoritmos computacionales), así como del dominio de conceptos propios de las telecomunicaciones (por ejemplo, análisis de tráfico y caracterización de enlaces inalámbricos). De acuerdo a lo anterior, se considera que este proyecto pertenece al campo de aplicación de la ingeniería telemática. Además, se debe notar que, si bien los fenómenos simulados corresponden al proceso de transmisión de información, dichos fenómenos están siendo analizados en el contexto de un sistema con características telemáticas (nodos mIoT conectados a la red 5G).

1.5. PROPUESTA DE SOLUCIÓN

Considerando lo expuesto anteriormente, se propuso desarrollar un simulador a nivel sistema programado de acuerdo con el paradigma de eventos discreto. La elección de una simulación a nivel de sistema se deriva del enfoque que tiene nuestro proyecto hacia los distintos tipos de tráfico de nodos NB-IoT y la simulación de cada uno de ellos como fuente de tráfico. Las simulaciones a nivel de sistema permiten modelar el comportamiento de múltiples radio bases, múltiples nodos como fuentes de tráfico, la propagación de las señales y la interferencia que estas causan, a la vez que se realizan abstracciones más simples de lo que sucede más allá de estas interacciones. Esto facilita la implementación de una gran cantidad de actores. La generación de variables aleatorias vino de la mano de las distintas aplicaciones mIoT y sus patrones de transmisión estocásticos, además de la localización de nodos en un plano la cual no será uniforme.

El simulador es capaz de evaluar la calidad del servicio que la red celular propuesta ha de brindar a nodos de mIoT. Dicha arquitectura de red, propuesta en este mismo trabajo está basada

a su vez en los avances hechos, por grupos como 3GPP, hacia el despliegue de la red 5G. La base de la que se partió es la tecnología NB-IoT, la cual abordó el caso de uso mMTC en la red 4G donde ha estado prestando servicio a nodos mIoT a una escala menor que la esperada en 5G.

Se modeló el servicio brindado a nodos estáticos de IoT cuyas aplicaciones pertenecen al caso de uso mIoT. Con la ayuda del estándar NB-IoT, para el que se proponen mejoras en el acceso múltiple, esto en búsqueda de cumplir con los KPI's de la red 5G, ya que si bien NB-IoT pertenecerá al paradigma de 5G, no es viable tal y como existe ahora para cumplir con los requerimientos. La importancia de esto recae, como se menciona en [9], en que la tecnología de IoT ha creado una revolución en la última década con la creación de aplicaciones pensadas alrededor de todo tipo de sensores, lo que resulta en una proyección estimada de miles de millones de dispositivos IoT para el 2020 [3]. Esta misma referencia asegura que IoT está tomando un papel principal en el desarrollo de la quinta generación, debido a que se espera que los dispositivos de IoT formen la gran mayoría de dispositivos en esta nueva generación que se avecina.

Se definió un análisis fundamental principalmente del modelado de cuatro componentes que son esenciales para la caracterización de un sistema de comunicación móvil. Estos componentes corresponden al: modelado de despliegue de usuarios, canal, tráfico y un esquema de acceso múltiple al medio. Esto se encuentra en el capítulo 4, que comprende un análisis de forma detallada de éstos, pero a continuación se abordan de tal forma que se esclarezca la arquitectura del modelo de sistema, presentada también más adelante.

Se consideró un modelo de despliegue de nodos IoT que seguirá un proceso puntual de Poisson con el fin de crear una geometría estocástica. Se representaron las pérdidas por medio de un modelo de canal estadístico para ambientes celulares de quinta generación. Se consideró en la simulación un modelo de tráfico fuente en el que cada nodo mIoT genera tráfico ya sea periódico o aleatorio, cada caso con distintas tasas. Finalmente, referente al método de acceso múltiple, se implementó la mejora a la tecnología NB-IoT, que trató de la implementación del esquema NOMA en el dominio de la potencia, de forma que agrupamientos (de longitud fija) de nodos estarán compartiendo un mismo recurso (una sub-banda).

Este análisis conllevó distintos parámetros de la red 5G y los dispositivos NB-IoT en un ambiente masivo, los resultados brindaron una base fundamental para evaluar el desempeño de estas redes y por supuesto, su dimensionamiento en términos de objetivos de calidad de servicio (QoS).

1.6. ALCANCES

Se obtuvieron resultados que permitieron analizar las configuraciones de la arquitectura de red propuesta que conllevan a mejorar significativamente la calidad de servicio. Teniendo como métricas principales la densidad de usuarios soportada y la tasa de transmisión máxima alcanzada. Estos resultados reflejaron a su vez las ventajas que puede traer la selección de cierta arquitectura de red y su despliegue. Es aquí donde se encuentra una de las ventajas de realizar un

simulador, ya que con la ayuda de múltiples computadoras, se pueden simular miles de nodos mIoT en esta red. Las configuraciones y parámetros de la red se pueden modificar al inicio de cada simulación, y también inspeccionar mientras esta corre. La variación de estos parámetros a lo largo de múltiples simulaciones permitió generar tablas y gráficas de los resultados obtenidos.

Este proyecto no cubrió los aspectos de movilidad entre celdas para los nodos de IoT, ya que el caso de uso mMTC representa a los nodos estáticos en su mayoría o con velocidades menores a 3Km/h. No se desarrollaron nuevos modelos probabilísticos o matemáticos de ninguna clase, sino que se implementaron los existentes para el escenario propuesto. Por último, el análisis se concentró a un escenario unicelular, es decir, una sola celda.

Capítulo 2

Marco Teórico

El objetivo de este capítulo fue revisar los fundamentos teóricos de los sistemas de comunicaciones móviles. Se comenzó con los diferentes modelos de despliegue para estaciones base (BSs, *base stations*) y dispositivos ,estos últimos comunmente conocidos como equipos de usuario (UEs, *user equipments*). Se estudiaron las características de una geometría estocástica y se repasó la teoría del concepto celular, la cual aborda los efectos de la interferencia en estos sistemas.

Después se ahondó en las pérdidas producidas en un sistema celular por medio de los modelos de canal más comunes, caracterizados por los parámetros a larga y pequeña escala. Además, se revisaron los diferentes esquemas de acceso al medio en comunicaciones móviles.

Finalmente, se revisaron los aspectos de la teoría del tráfico en telecomunicaciones, los organismos más importantes de estandarización de redes móviles. También, algunos conceptos de las simulaciones a nivel de sistema y el paradigma de eventos discretos, junto con los lenguajes de programación más utilizados para programarlas.

2.1. MODELADO DEL DESPLIEGUE CELULAR

La teoría del diseño celular ofrece la forma simplificada de un diseño para redes móviles. Esta teoría se desarrolló en la literatura desde el comienzo del concepto celular [1975 aprox.], sin embargo, el despliegue uniforme que propone la teoría celular es idealizado, ya que en la realidad nunca se tenían despliegues de este tipo. Frecuentemente, en ubicaciones donde se producía mayor cantidad de tráfico, se instalaba otra BS y por lo tanto se rompía con la uniformidad.

Existen diferentes estrategías de despliegue de BSs (como se puede ver en la *Figura 2.1*). Como se mencionaba en el parrafo anterior, un despliegue uniforme no es realista es por esto que para modelar la ubicaciones de BSs y UEs se han utilizado modelos estadísticos como los procesos puntuales de Poisson (*Poisson Point Process - PPP*) o en su caso estrategias híbridas que consideren ambos modelos, uno uniforme y uno estadístico.

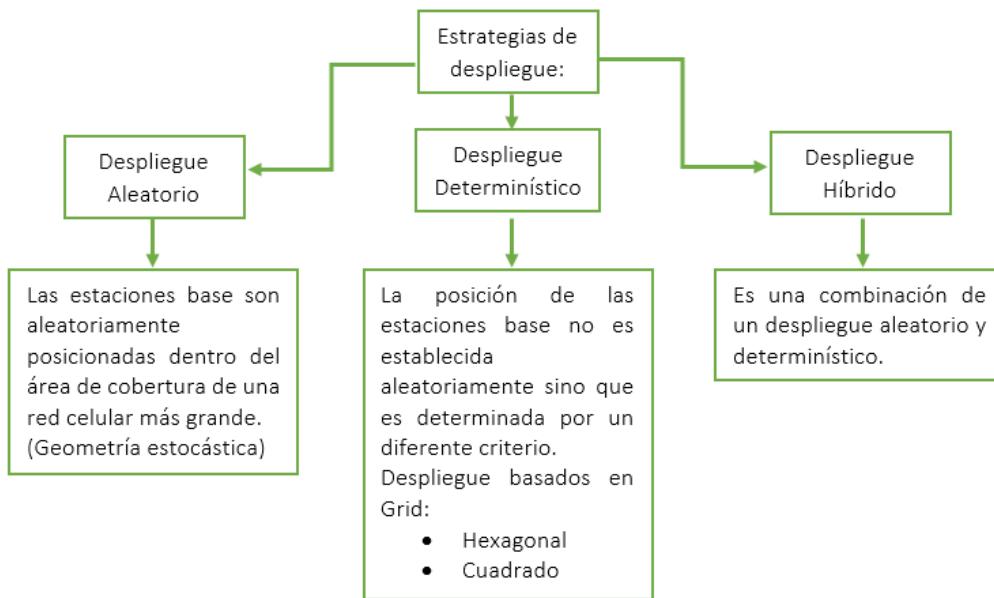


FIGURA 2.1: Diferentes estrategias de despliegue para BSs

2.1.1. Despliegue aleatorio

En primer lugar, para realizar un despliegue aleatorio, se recurre a los procesos puntuales (*Point Process - PP*) [10] ya que son los objetos elementales estudiados en la teoría de geometría estocástica.

Visualmente, un PP se puede representar como una colección aleatoria de puntos en el espacio. Más formalmente, un proceso puntual es un mapeo medible Φ desde algún espacio de probabilidad al espacio de medidas puntuales (una medida puntual es una medida que es localmente finita y que solo toma valores enteros) en algún espacio E .

Algunas dicotomías sobre procesos puntuales en el espacio Euclíadiano \mathbb{R}^d son las siguientes:

- Un PP puede ser simple o no. Es simple si la multiplicidad de un punto es como máximo uno (no hay dos puntos en la misma ubicación).
- Un PP puede ser estacionario o no. La estacionariedad se cumple si la ley del proceso puntual es invariante por traslación.
- Un PP puede ser de Poisson o no. En la siguiente subsección se proporciona una definición formal de los procesos puntuales de Poisson (PPP).

Antes de abordar la teoría de los PPP, es necesario definir un Proceso de Poisson.

Los procesos de Poisson son frecuentemente utilizados para representar o modelar fenómenos en las telecomunicaciones, p. ej. la generación de llamadas telefónicas.

Algunas propiedades de los procesos de Poisson son las siguientes [11]:

- Se compone de una secuencia de variables aleatorias $X_1, X_2, X_3, \dots, X_k$, de modo que cada variable representa el número de ocurrencias de algún evento, durante un intervalo de tiempo.
- Es un proceso estocástico. Cada vez que ejecuta el proceso de Poisson, producirá una secuencia de resultados aleatorios diferentes según alguna distribución de probabilidad.
- Es un proceso discreto. Los resultados del proceso de Poisson son el número de ocurrencias de algún evento en el período de tiempo especificado, que sin duda es un número entero, es decir, un número discreto.
- Tiene incrementos independientes. Lo que esto significa es que el número de eventos que el proceso predice que ocurrirán en cualquier intervalo dado, es independiente del número en cualquier otro intervalo disjunto.
- Las variables constitutivas del proceso de Poisson $X_1, X_2, X_3, \dots, X_k$ tienen una distribución idéntica.
- Las variables constitutivas del proceso de Poisson $X_1, X_2, X_3, \dots, X_k$ tienen una distribución de Poisson, que viene dada por la Función Masa de Probabilidad (PMF):

$$P_X(k) = \frac{e^{-\lambda} * \lambda^k}{k!} \quad (2.1)$$

La fórmula anterior nos da la probabilidad de ocurrencia de k eventos en unidad de tiempo, dado que la tasa de ocurrencia promedio es λ eventos por unidad de tiempo.

Los procesos de Poisson tienen una subestructura notable. Aunque el número de eventos ocurridos se modela usando una distribución de Poisson discreta, el intervalo de tiempo entre eventos consecutivos se puede modelar usando una distribución exponencial, que por el contrario, es una distribución continua [11].

Finalmente, se puede proceder a la definición formal de un proceso puntual de Poisson (*Poisson Point Process - PPP*).

Sea γ una medida localmente finita en algún espacio métrico E . Un proceso puntual Φ es de Poisson en E si:

- Para todos los subconjuntos disjuntos A_1, \dots, A_n de E , las variables aleatorias $\Phi(A_i)$ son independientes.

- Para todos los conjuntos A de E , las variables aleatorias $\Phi(A)$ son Poisson.

Una propiedad clave establece que condicionalmente $\Phi(A) = n$, estos n puntos están ubicados independientemente (y de manera uniforme para un PPP homogéneo) en A .

- Un PPP puede ser homogéneo o no. En el caso homogéneo, la densidad de los puntos es constante en el espacio (también conocidos como HPPP).
- El PPP homogéneo es estacionario y simple. Esto puede considerarse como el proceso puntual más simple (y más natural).

Los PPP no homogéneos se pueden usar para modelar distribuciones de usuarios que no son uniformes en el espacio.

2.1.2. Despliegue determinístico

Ahora bien, para un despliegue determinístico, se suele optar por un esquema hexagonal el cual recurre a la teoría clásica celular donde se busca que los sistemas celulares brinden determinada cobertura para un servicio. Para tal despliegue se divide el área geográfica en segmentos llamados celdas. El espectro de frecuencia también es dividido en canales y estos últimos son agrupados para repartirse entre las celdas. Estos sistemas logran una alta capacidad gracias al reúso del canal de comunicación permitiendo a las estaciones base compartir los recursos, sin embargo, este reúso resulta en una interferencia co-canal generada entre usuarios que comparten el mismo canal. Esta interferencia limita el rendimiento y la capacidad de un sistema celular [12].

Los sistemas celulares son caracterizados por [13]:

- Ser limitados por la interferencia (capacidad).
- Servir a una alta densidad de usuarios.
- Considerar la disponibilidad del espectro sólo como un factor limitante.
- Reúso de frecuencias.
- Uso de Estaciones Base de baja potencia.
- Tener celdas de distintas coberturas.
- Permitir *handover*.

Una buena planeación celular es de crucial importancia para lograr un buen rendimiento en el sistema y una buena Calidad de servicio (QoS).

La planeación celular consiste en:

- La colocación de BSs y establecimiento de coberturas

- Una óptima administración de recursos de radio.
- La minimización de interferencia.

La planeación celular se desempeña de acuerdo a:

- La morfología del área de servicio y los modelos de propagación.
- Los perfiles de usuario y modelos de tráfico.

Considerando:

$$R \rightarrow \text{Radio de cobertura de la celda}$$

$$D_r \rightarrow \text{Distancia de reuso de celda co_canal}$$

$$d_u \rightarrow \text{Distancia unitaria entre los centros de dos celdas adyacentes}$$

$$d_u = \sqrt{3}R$$

$$\text{Llegando a una distancia de reuso normalizada } D_n$$

$$D_n = i^2 + ij + j^2, \text{ donde } i! = 0, j! = 0 \quad (2.2)$$

$$\Rightarrow D_r = D_n d_u \quad (2.3)$$

$$\text{área de una celda } S_{cel}$$

$$S_{cel} = (3\sqrt{3}/2)R^2 \quad (2.4)$$

$$\text{área de un clúster } S_{clu}$$

$$S_{clu} = (3\sqrt{3}/2)D_n^2 d_u^2 \quad (2.5)$$

Número de celdas por clúster, N_{cc} ó también conocido como "factor de reuso"

$$N_{cc} = \frac{S_{clu}}{S_{cel}} \quad (2.6)$$

Siendo valores posibles para N_{cc} , 1, 3, 4, 7, 9, 12, 13, ... [Véase Figura 2.2].

Para implementar el reúso, se deben asignar un conjunto de canales disponibles para un grupo de celdas, el clúster y repetir ese conjunto a través de toda el área de servicio.

$$N_{ch/c} = \frac{N_{ch/s}}{N_{cc}} \quad (2.7)$$

$$N_{ch/c} \rightarrow \text{Número de canales por celda}$$

$$N_{ch/s} \rightarrow \text{Número de canales en el sistema}$$

$$N_{cc} \rightarrow \text{Número de celdas por clúster}$$

La relación de reutilización co-canal, r_{cc} es usada para caracterizar clústeres

$$r_{cc} = \frac{D_r}{R} = \sqrt{3N_{cc}} \quad (2.8)$$

Un valor grande de r_{cc} corresponde a:

1. Una baja interferencia co-canal
2. Baja capacidad del sistema

El clúster es escogido lo más pequeño posible, tomando los umbrales de interferencia en consideración [13].

$$GSM, N_{cc} = 4$$

$$UMTS, N_{cc} = 1$$

$$LTE, N_{cc} = 3$$

Diferentes tamaños de celda son usados para distribuciones no uniformes de tráfico [14]. El uso

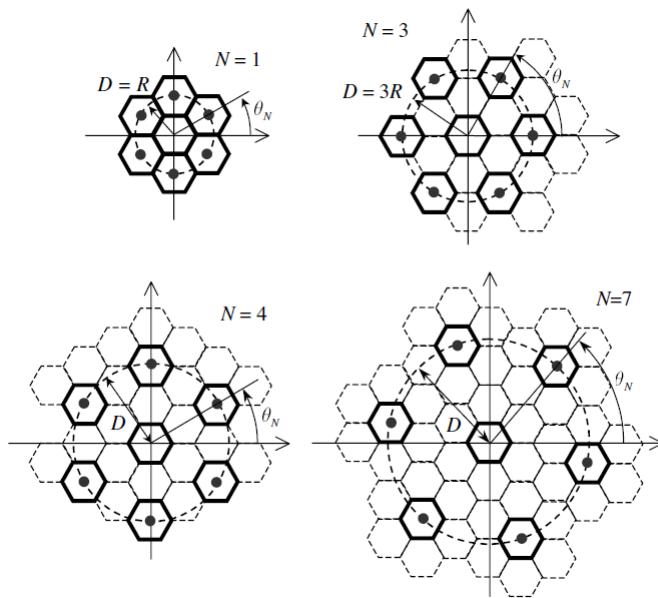


FIGURA 2.2: Localización de celdas co-canal con distintos factores de reúso, [Fuente: Tranter 2003]

de diferentes tamaños de celdas:

- Conduce a un incremento de la interferencia, debido a la poca uniformidad en la estructura celular.
- Requiere cuidado adicional en el despliegue de BSs

2.1.3. Interferencia en los sistemas de comunicaciones

SINR representa la relación entre el nivel de la señal recibida y la suma de la interferencia y el ruido. Es una cantidad utilizada para dar límites superiores teóricos sobre la capacidad del canal (o la tasa de transferencia de información) en sistemas de comunicación inalámbrica como

las redes celulares.

En general el receptor recibe:

$$\text{SNIR} = \frac{S}{I + N} \quad (2.9)$$

Donde:

$S \rightarrow$ Potencia de portadora de la señal deseada

$I \rightarrow$ Potencia de las portadoras de las señales interferentes

$N \rightarrow$ Potencia del ruido

Los sistemas de comunicaciones móviles se caracterizan por ser sistemas limitados por interferencia, donde I domina sobre N , por lo tanto la participación del ruido N puede ser ignorada [13].

Quedando entonces como la relación portadora a interferencia:

$$\frac{S}{I + N} \cong \frac{S}{I} \rightarrow \text{SIR} \quad (2.10)$$

La interferencia co-canal I_{cc} , es decir, la interferencia que se produce por los canales con reuso de frecuencias, es un problema trabajado en los sistemas de comunicaciones móviles. La estimación de la relación portadora a interferencia co-canal S/I_{cc} es calculada de acuerdo a las siguientes asunciones:

1. Todas las celdas son del mismo tamaño
2. Las potencias radiadas por todas las BSs son iguales
3. Todas las BSs tienen antenas omnidireccionales
4. El decaimiento de la potencia promedio (a_{pd}) con la distancia es de la forma:

$$P_r = P_0(d/d_0)^{-a_{pd}} \quad (2.11)$$

En un sistema celular en general, la interferencia co-canal es calculada tomando la interferencia de todas las celdas N_{Icc} .

$$\frac{S}{I_{cc}} = \frac{S}{\sum_{k=1}^{N_{Icc}} I_k} \quad (2.12)$$

Usualmente la interferencia puede ser estimada tomando únicamente el primer anillo de interferencia [13].

$$\frac{S}{I_{cc}} = \frac{S}{\sum_{k=1}^6 I_k} \quad (2.13)$$

En el caso de transmisión de bajada (*downlink*), para un usuario en los límites de la celda, el cálculo de la interferencia co-canal se puede aproximar a:

$$\frac{S}{I_{cc}} = \frac{R^{-a_{pd}}}{2(D_r - R)^{-a_{pd}} + 2(D_r)^{-a_{pd}} + 2(D_r + R)^{-a_{pd}}} \quad (2.14)$$

Donde:

R : radio de cobertura de la celda

D_r : distancia de reuso de celda co – canal

En términos generales, la interferencia puede disminuir [13]:

- implementando celdas sectorizadas.
- haciendo *downtilting* del lóbulo principal de la antena de la BS.
- reduciendo la altura de la BS
- optimizando la localización de la BS
- implementando control de potencia
- implementando *frequency hopping*

En resumen, la planeación de una red celular es ejecutada de la siguiente manera [13]:

1. El mínimo valor para la relación portadora a interferencia impone el tamaño del clúster.
2. Se estima el tráfico en una celda determinada.
3. Se calcula el número de canales para una calidad de servicio determinada.
4. En caso de que el número de canales disponibles no sea suficiente, el tráfico se reducirá, es decir, la cobertura se reducirá.
5. Se establece el plan de frecuencias y la estructura del despliegue de celdas.
6. Cuando se propone una estructura de celdas no uniforme, los canales deberán ser distribuidos de acuerdo a las necesidades de capacidad, y a los valores topes permitidos de interferencias co- canal y canal-adyacente.

2.1.4. Capacidad en los sistemas de comunicaciones

La teoría de la información de Shannon nos dice la cantidad de información que un canal puede transportar. En otras palabras, especifica la capacidad del canal. La capacidad de un sistema de comunicación es la velocidad de datos máxima en bits por segundo que se puede transferir de manera confiable del transmisor al receptor. En el sentido estricto de la teoría de la información, este es un límite superior infranqueable. En un único enlace Tx-Rx de ancho de banda de unidad sujeto a AWGN, la capacidad en bits por uso de canal (es decir, bps / Hz) viene dada por la fórmula de *Shannon-Hartley*:

$$C = B \log_2 \left(1 + \frac{S}{I} \right) \text{ [bps]} \quad (2.15)$$

Donde:

B : es el ancho de banda del canal en Hertzios.

C : es la capacidad del canal (tasa de bits de información bit/s)

S : es la potencia de la señal útil (W, mW, etc.)

I : es la potencia de las señales interferentes

También descrita como:

$$C = B \log_2 \left(1 + \frac{P_{tx}H}{N_0BI} \right) \quad (2.16)$$

Donde:

P_{tx} : es la potencia de transmisión promedio

H : es la ganancia de potencia del canal

N₀ : es la densidad de potencia del ruido (aprox. -173dBm/Hz)

I : es la potencia de las señales interferentes

2.2. MODELADO DEL CANAL CELULAR

Los modelos de propagación por radio se clasifican en modelos a gran escala y a pequeña escala. Los efectos a gran escala generalmente ocurren en el orden de cientos a miles de metros de distancia. Los efectos a pequeña escala se localizan y ocurren temporalmente (en el orden de unos pocos segundos) o espacialmente (en el orden de unos pocos metros). Los parámetros del canal generalmente se dividen en Pérdida por trayectoria (PL), parámetros de gran escala (LSP, como ensombrecimiento, dispersión de retardo, dispersión angular, etc.) y parámetros de pequeña escala (como demora, ángulo de llegada y salida, etc.), que reflejan conjuntamente las características de desvanecimiento del canal. El procedimiento de generación de los coeficientes del canal se puede apreciar en la *Figura 2.3*. La pérdida por trayectoria generalmente se expresa en una o dos fórmulas y un conjunto de valores numéricos de parámetros, que reflejan las relaciones con el entorno de transmisión, la distancia y la frecuencia, etc.

El rendimiento a nivel de enlace es un fenómeno de pequeña escala el cual lida con cambios instantáneos en el canal a través de áreas e instantes de tiempo pequeños. A lo largo de estos cambios se considera la potencia recibida como constante, por otra parte, en las simulaciones a nivel de sistema para determinar el rendimiento en general del sistema para un gran número de usuarios esparcidos en un área geográfica es necesario incorporar parámetros de larga escala como el comportamiento estadístico de la interferencia, así como los niveles de señal experimentados por cada usuario a través de largas distancias, ignorando las características transitorias del canal (las de pequeña escala) [12].

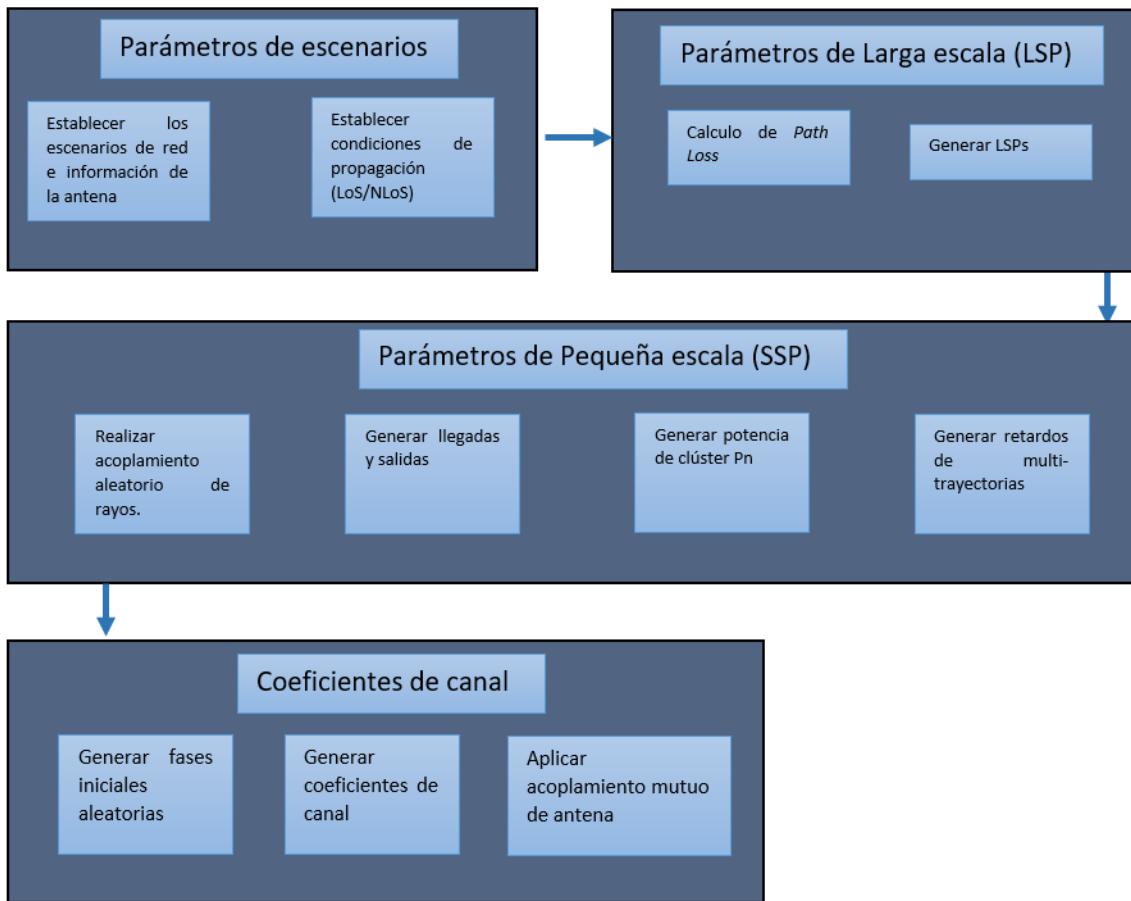


FIGURA 2.3: Procedimiento de generación de coeficientes de canal, [Fuente: 3GPP TR-36.873]

En una simulación a nivel de sistema, principalmente se busca la probabilidad de que un usuario en particular alcance un servicio aceptable en el sistema, para esto es necesario contemplar los efectos de los múltiples usuarios para cada enlace individual entre un móvil y la estación base. Por lo tanto en las simulaciones a nivel de sistema se suelen omitir los parámetros a pequeña escala.

2.2.1. Relaciones Generales de Propagación

La pérdida por trayectoria L_p se define como [13]:

$$L_p[dB] = P_{tx}[dBm] + G_{tx}[dBi] - P_{rx}[dBm] + G_{rx}[dBi] \quad (2.17)$$

Donde:

P_{tx} → Potencia transmitida

G_{tx} → Ganancia de la antena transmisora

$P_{rx} \rightarrow$ Potencia recibida

$G_{rx} \rightarrow$ Ganancia de la antena receptora

En muchas aplicaciones la ganancia de la antena es referida al dipolo de media longitud de onda:

$$G_{[dBi]} = G_{[dBd]} + 2,15 \quad (2.18)$$

La Potencia Isotrópica Radiada Efectiva (EIRP) se define como:

$$P_{EIRP[dBm]} = P_{tx[dBm]} + G_{tx[dBi]} \quad (2.19)$$

2.2.2. Pérdida por trayectoria en el Espacio Libre

El receptor puede recibir una señal atenuada directa (también llamada señal de línea de vista (LoS)) del transmisor. Las pérdidas en el espacio libre (FSPL, *Free Space Path Loss*) se utilizan para predecir la pérdida de trayectoria cuando hay una linea de vista (LoS, *Line of Sight*) clara y sin obstrucciones entre el transmisor y el receptor. Se basa en la ley de distancia al cuadrado inverso que establece que la potencia recibida (P_{rx}) decae por un factor de cuadrado de la distancia (d) desde el transmisor.

Se considera a la propagación en el espacio libre como la mínima atenuación que una señal puede sufrir en el medio.

La potencia de la señal receptora P_{rx} con una propagación en el espacio libre se define como (esta fórmula es conocida como fórmula de Friis):

$$P_{rx[W]} = \left(\frac{\lambda[m]}{4\pi d[m]} \right)^2 P_{tx[W]} G_{tx} G_{rx} \quad (2.20)$$

6

$$P_{rx[dBW]} = -32,44 + P_{tx[dBW]} + G_{tx[dBi]} + G_{rx[dBi]} - 20\log(d[km]) - 20\log(f[MHz]) \quad (2.21)$$

Donde:

$d \rightarrow$ Distancia entre Rx y Tx

$f \rightarrow$ Frecuencia de operación

$\lambda \rightarrow$ Longitud de onda, $\lambda = \frac{c}{f}$

$c \rightarrow$ Velocidad de la luz (299 792 458 m/s)

Por lo tanto, la pérdida por trayectoria en el espacio libre L_0 se define como:

$$L_0[dB] = 32,44 + 20\log(d[km]) + 20\log(f[MHz]) \quad (2.22)$$

Tomando el modelo del decaimiento de potencia promedio con la distancia a_{pd} :

$$L_{p[dB]} = L_{ref} + 10a_{pd}\log(d[km]) \quad (2.23)$$

$a_{pd} = 2$, para una propagación en el espacio libre

El a_{pd} o también conocido como exponente de pérdida (PLE, *Path Loss Exponent*) es un valor que va de 2 a 4 frecuentemente. El valor mínimo (i.e. 2) proviene de la perdida por el espacio libre y el máximo (i.e. 4) de la pérdida del modelo de tierra plana ()*Flat Earth*). En algunos modelos se llega a incluir valores de PLE más altos que los aquí descritos.

2.2.3. Caracterización del canal de radio

Usualmente en ambientes urbanos no hay LoS entre la estación base (BS) y la terminal móvil (MT¹) [véase Figura 2.4] por lo que la transmisión es realizada por reflexión, difracción y dispersión de las señales.

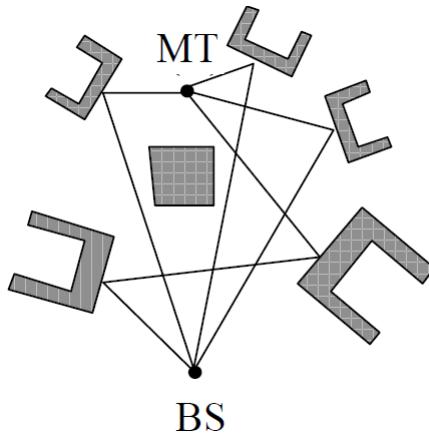


FIGURA 2.4: Propagación de señales celulares en ambientes urbanos, [Fuente: L. Correia 2018]

Sin embargo estas señales sufren de desvanecimiento con caídas de potencia. Este desvanecimiento depende de la posición y del ambiente en el cual se propague la señal.

Características de desvanecimiento:

- Desvanecimiento lento: Depende esencialmente de la distancia, sigue una distribución Log-normal
- Desvanecimiento rápido: Es asociado al movimiento del usuario, sigue una distribución Rice

¹MT y UE son términos análogos.

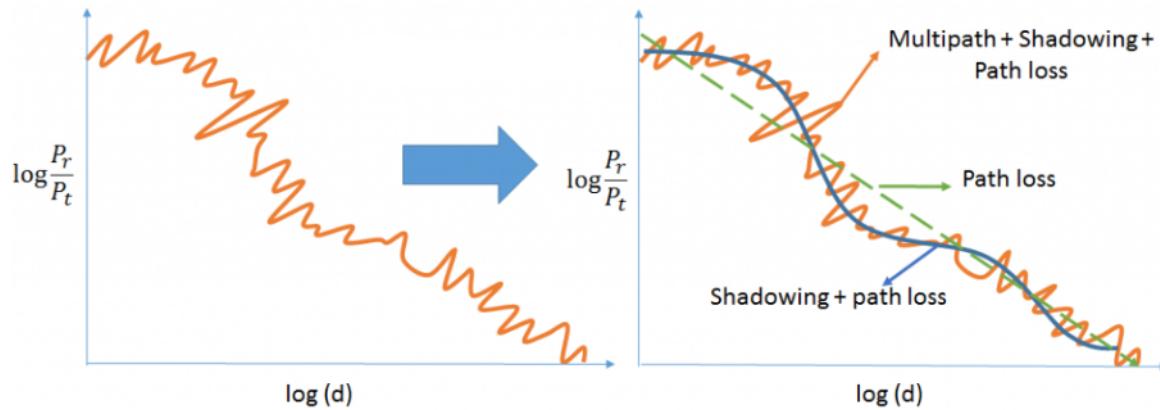


FIGURA 2.5: Ejemplo de niveles de señal con pérdidas por trayectoria, desvanecimiento lento y desvanecimiento rápido, [Fuente: V. Mathuranathan, 2016]

En la *Figura 2.5* se observa que al principio, la señal parece muy aleatoria. Mirando más de cerca podemos observar tres componentes principales que forman a la señal, como se muestra en la mitad derecha [15].

1. Pérdida por trayectoria (*Path loss*)
2. Ensombrecimiento (*Shadowing*) o desvanecimiento lento
3. Multitrayectoria (*Multipath*) o desvanecimiento rápido.

El desvanecimiento lento puede ser causado por eventos como el *shadowing*, donde una gran obstrucción, como una colina o un gran edificio, oscurece la trayectoria de la señal principal entre el transmisor y el receptor. Se considera un parámetro a gran escala.

El desvanecimiento rápido ocurre cuando la amplitud y el cambio de fase impuestos por el canal varían considerablemente durante el período de uso. Una señal que viaja en un entorno puede verse reflejada por varios objetos en el camino. Esto da lugar a varias señales reflejadas. Las señales reflejadas llegan al receptor en diferentes instantes de tiempo y con diferentes intensidades que conducen a la propagación multitrayectoria. Se considera un parámetro a pequeña escala.

Para una banda estrecha (*narrowband*), donde prevalece el desvanecimiento plano en lugar de un desvanecimiento selectivo de frecuencia, el desvanecimiento se caracteriza de la siguiente manera:

- Desvanecimiento rápido:
 - LoS: Distribución Rice (no intenso)
 - NLoS: Distribución Rayleigh (intenso)
- Desvanecimiento lento:

- Distribución Log-Normal
- Ambos desvanecimiento rápido y lento:
 - Distribución Susuki

Cuando se habla del desvanecimiento de Rayleigh en enlaces inalámbricos en la literatura [16], se encuentra que las componentes en cuadratura y en fase de la señal recibida son variables aleatorias Gaussianas con media cero que se distribuyen de forma independiente e idéntica (iid), siendo así, que la magnitud de la señal banda base compleja sigue una distribución de Rayleigh [Ecuación 2.24]. Por otra parte, la distribución de la potencia normalizada de una señal de banda base compleja recibida bajo el desvanecimiento unitario de Rayleigh es modelada por medio de una distribución exponencial unitaria [Ecuación 2.25].

$$P_X(x) = \frac{x}{\sigma_i^2} \exp\left(-\frac{x^2}{2\sigma_i^2}\right), \quad x \geq 0 \quad (2.24)$$

Donde:

σ → Desviación estándar

$$P_X(x) = \lambda e^{-\lambda x} \quad (2.25)$$

Donde:

$\lambda \rightarrow 1$

Los márgenes de desvanecimiento deben tomarse en cuenta para caracterizar la variación de las señales alrededor de un valor promedio, esto depende de:

- Características del ambiente (LoS o NLoS)
- Calidad de servicio (QoS, *Quality of Service*)

Además de estos modelos estadísticos de estimación de señal, existen otros que se han sido utilizados para modelar las pérdidas en entornos celulares. Estos son los teóricos y los empíricos.

Teóricos: Son una aproximación a la realidad, no toman en cuenta todos los factores de la propagación pero permiten cambios fáciles de los parámetros. Algunos ejemplos son los siguientes:

- *Ray Tracing*
- Modelo Ikegami [1984]
- Modelo Walfish-Bertoni [1988]

Empíricos: Están basados en la observación de mediciones, conduciendo al mejor ajuste de ecuaciones. Tienen la ventaja de tomar en cuenta todos los factores que influyen en la propagación. Para ambientes exteriores hay dos modelos fundamentales:

- COST 231 Okumura-Hata
 - Largas distancias (>5km)
 - Ambientes rurales, urbanos y suburbanos
 - Alta desviacion estandar
 - Rango de frecuencias aplicables [1.5,2.0] GHz
- COST 231 Walfish-Ikegami [1999]
 - Cortas distancias (<5km)
 - Ambientes urbanos y suburbanos
 - Rango de frecuencias aplicables [.8,2.0] GHz

2.3. ESQUEMAS DE ACCESO MÚLTIPLE AL MEDIO

Las técnicas de acceso múltiple (MA) generalmente se pueden dividir en enfoques ortogonales y no ortogonales [17]. En MA ortogonal (OMA), los recursos de radio se dividen ortogonalmente entre dispositivos, donde las señales de diferentes dispositivos no se superponen entre sí. Las distintas instancias de OMA son: acceso múltiple por división de tiempo (TDMA), acceso múltiple por división de frecuencia (FDMA), acceso múltiple por división de frecuencia ortogonal (OFDMA), y FDMA de portadora única (SC-FDMA).

2.3.1. OMA

Los enfoques OMA no tienen la capacidad de combatir la interferencia entre células [18]; por lo tanto, se requieren técnicas cuidadosas de planificación celular y gestión de interferencia para resolver este problema.

Existen 4 técnicas básicas de acceso múltiple (OMA):

1. Frecuencia: asignación de una portadora - FDMA (Acceso múltiple por división de frecuencia)
2. Tiempo: asignación de un intervalo de tiempo - TDMA (Acceso múltiple por división de tiempo)
3. Código: asignación de un código - CDMA (Acceso múltiple por división de código)
4. Frecuencia Ortogonal: asignación de un conjunto de sub-portadoras - OFDMA (Acceso múltiple por división de frecuencia ortogonal).

En muchos sistemas prácticos, se utiliza una mezcla o combinación de estas técnicas básicas [13]. Y también, de acuerdo a cada generación de comunicaciones móviles, el esquema de acceso al medio suele cambiar [véase Figura 2.9].

2.3.2. NOMA

Igualmente, el acceso múltiple no ortogonal (NOMA) se ha convertido en un principio importante para el diseño de técnicas de acceso por radio para las redes inalámbricas de quinta generación (5G) [19]. NOMA se puede clasificar en dos categorías, NOMA en el dominio de código (CD-NOMA) y NOMA en el dominio de la potencia (PD-NOMA). CD-NOMA utiliza diferentes códigos en el mismo recurso para lograr una ganancia de multiplexación, mientras que PD-NOMA asigna a los usuarios niveles de potencia distintos para maximizar el rendimiento. En comparación con el acceso múltiple ortogonal (OMA) que se ha aplicado ampliamente en los sistemas de comunicación inalámbrica existentes, NOMA posee el potencial de mejorar aún más la eficiencia espectral del sistema y la capacidad de conectividad.

PD-NOMA utiliza el dominio de la potencia para el acceso múltiple donde diferentes usuarios son servidos con diferentes niveles de potencia, como las señales de los diferentes usuarios se sobreponen, los receptores aprovechan la cancelación sucesiva de interferencia (SIC) para distinguir a cada señal. Como varios usuarios son admitidos en la misma ranura de tiempo, frecuencia o código, la interferencia co-canal será fuerte en los sistemas NOMA [20], por lo que no es realista el asegurar a todos los usuarios del sistema utilicen NOMA conjuntamente. Una alternativa es utilizar un esquema híbrido donde NOMA sea combinado con el esquema convencional ortogonal OMA. El rendimiento de este esquema híbrido es muy dependiente de cómo son agrupados los usuarios [20]. El agrupamiento de usuarios seleccionará quienes compartirán un mismo bloque de recursos ortogonales y quienes no.

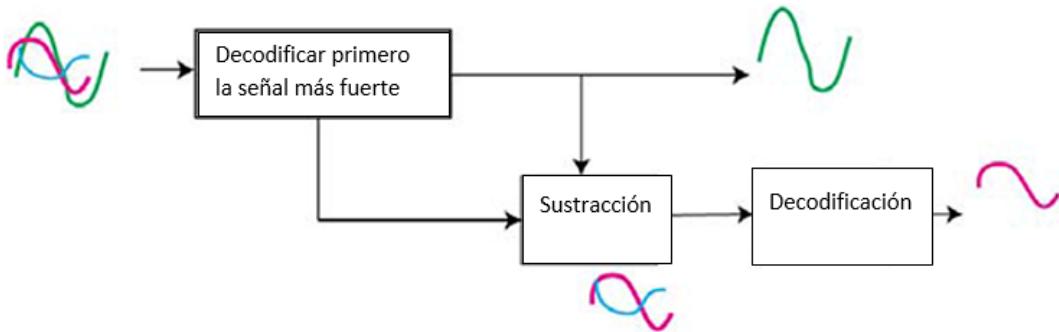


FIGURA 2.6: Cancelación Sucesiva de Interferencia (SIC), [Fuente: R. Kizilirmak 2017]

En la *Figura 2.6*, las dos señales de información indicadas con diferentes colores se superponen en el transmisor. La señal recibida en el receptor SIC incluye todas estas tres señales. La primer señal que SIC decodifica es la más fuerte, mientras que la restante es tratada como ruido. La primera señal decodificada se resta de la señal recibida y si la decodificación es perfecta, la forma de onda de la otra señal se obtiene con precisión. SIC itera el proceso hasta que encuentra todas las señales deseadas [21].

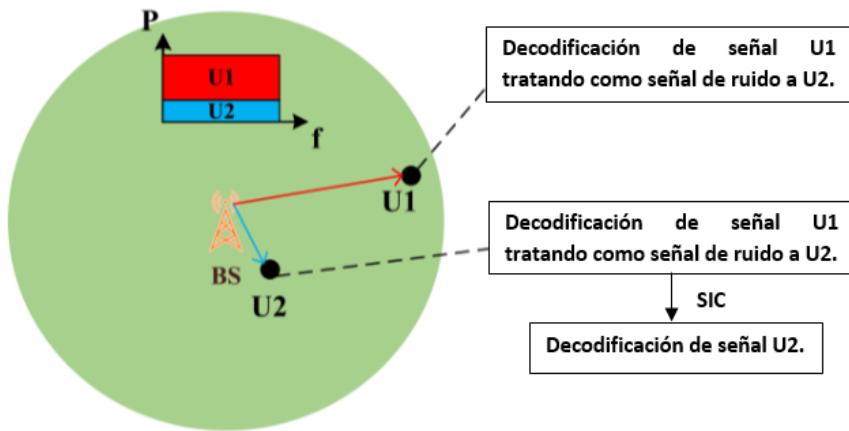


FIGURA 2.7: Ejemplo del esquema NOMA en un enlace de bajada con dos usuarios y una sub-portadora, [Fuente: Ding 2017]

Como se muestra en la *Figura 2.7*, la idea clave de NOMA en el dominio de potencia es asignar más potencia al usuario con condiciones de canal más pobres. El usuario 1 decodifica su propio mensaje directamente tratando el mensaje del usuario 2 como ruido y por otro lado, el usuario 2 realiza SIC, es decir, primero decodifica el mensaje del usuario 1 y luego elimina este mensaje de su observación antes de decodificar su propio mensaje.

Suponga que el usuario 1 es un dispositivo IoT que requiere solo una baja tasa de datos, y el usuario 2 es un usuario que exige una alta tasa de datos. Cuando se utiliza OFDMA, que es un ejemplo típico de OMA, cada usuario se asigna a una sub-portadora. En este ejemplo, la eficiencia espectral de OMA es pobre ya que el dispositivo IoT tiene más ancho de banda de lo que realmente necesita, mientras que al usuario de banda ancha no se le asigna suficiente ancho de banda. Por otro lado, el uso de NOMA fomenta el intercambio de espectro, es decir, el usuario de banda ancha también puede tener acceso a la sub-portadora ocupada por el dispositivo IoT, en la *Figura 2.4* se puede observar gráficamente la asignación del espectro en los dos esquemas.

Dada la madurez técnica de OFDMA, es muy probable que este tipo de OMA se incorpore a las redes 5G [19], sin embargo la forma en que múltiples sub-portadoras OFDMA se pueden combinar de manera eficiente con NOMA ha recibido mucha atención en la actualidad y podría ayudar a resolver algunos problemas vigentes.

Acerca del uso de arreglos múltiples de antenas (MIMO) para NOMA. Estos han sido tema de mucho interés. El ordenamiento de usuarios en escenarios de MIMO-NOMA es una tarea difícil [19]. En el caso de SISO, los canales de los usuarios son escalados, por lo que es sencillo ordenar a los usuarios de acuerdo con las condiciones de su canal. Sin embargo, cuando los nodos están equipados con antenas múltiples, los canales de los usuarios están en forma de vectores o matrices, lo que significa que ordenar a los usuarios de acuerdo con las condiciones de sus canales como en el caso SISO se vuelve complejo.

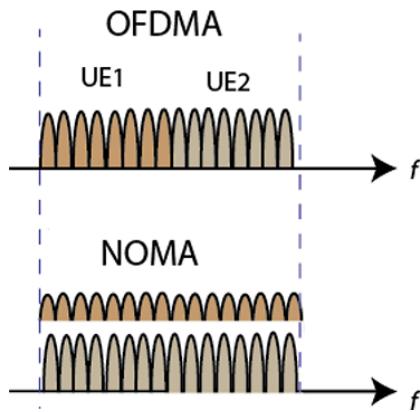


FIGURA 2.8: Ejemplo del espectro compartido para OFDMA y NOMA con dos usuarios, [Fuente: R. Kizilirmak 2017]

2.4. GENERACIONES DE LOS SISTEMAS CELULARES

La generación 1G fue la primera generación de tecnología celular inalámbrica. 1G se introdujo en la década de 1980. Las señales de radio utilizadas por la red 1G fueron analógicas y proporcionaba comunicación sólo por voz, alcanzaba una velocidad de 2.4 Kbps. La técnica de acceso múltiple utilizada en 1G es el acceso múltiple por división de frecuencia (FDMA) [véase Figura 2.9][22]. Esta generación utilizó el método de conmutación de circuitos para la transmisión de datos.

Las redes celulares 2G de segunda generación fueron lanzadas comercialmente en Finlandia en 1991, en el estándar GSM [22]. 2G utilizó el método de conmutación de paquetes para la transmisión de datos y habilitó el cifrado digital de la conversación por teléfono, además proporcionaron servicios multimedia como SMS (Servicios de mensajes cortos) MMS (Servicios de mensajes multimedia) y las velocidades de descarga y carga fueron de hasta 236 Kbps. Se utilizó el acceso múltiple por división de tiempo (TDMA), como el número de usuarios aumentó con el tiempo, TDMA se volvió obsoleto por causar una velocidad más baja para cada usuario.

3G (conocida también como UMTS) fue la tercera generación de tecnología inalámbrica de telecomunicaciones móviles y utilizó el acceso múltiple por división de código de secuencia directa (DS-CDMA) [22]. Las redes 3G ofrecieron velocidades de 3.1 megabits por segundo (Mbps) o más y se instalaron por primera vez en 1998. La aplicación del concepto 3G se encuentra en telefonía inalámbrica, acceso a Internet móvil, acceso a Internet inalámbrico, llamadas de conferencia y TV portátil.

4G (conocida también como LTE) es el término utilizado para describir la cuarta generación de servicio celular inalámbrico y es el estándar actual del servicio celular. Es hasta 10 veces más rápido que los servicios 3G. Sprint fue el primer operador en ofrecer velocidades 4G en los

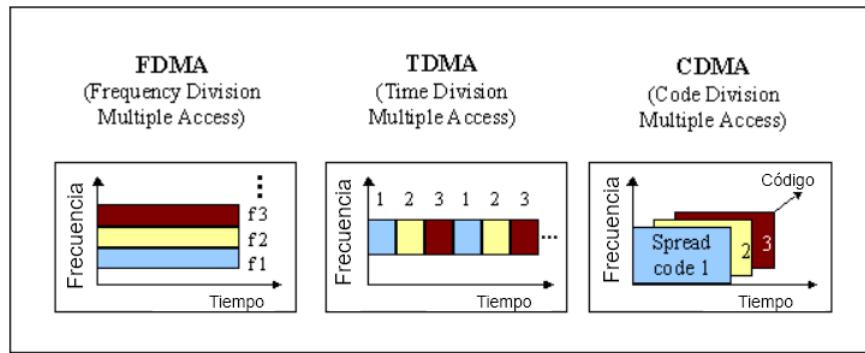


FIGURA 2.9: Diferentes tipos de acceso múltiple al medio ocupados en generaciones anteriores (1G, 2G y 3G), [Fuente: <https://www.itu.int/osg/spuold/ni/3g/technology/index.html>]

EE. UU a partir de 2009. Las redes 4G pueden ofrecer velocidades de descarga entre 5 y 12 Mbps y velocidades de carga de entre 2 y 5 Mbps, lo que finalmente da una velocidad máxima de 50Mbps. El sistema de comunicación celular 4G utiliza una versión avanzada del esquema FDMA, es decir, OFDMA (acceso múltiple por división de frecuencia ortogonal) [22].

2.5. SISTEMAS CELULARES 5G

La NGMN define su visión de una red 5G de la siguiente manera: "5G es un ecosistema de extremo a extremo para permitir una sociedad totalmente móvil y conectada."

Las anteriores generaciones de comunicaciones móviles (1G, ..., 4G) han sido transformadoras en el sentido de que fueron motivadas por mejorar los tradicionales KPIs de la red, sin embargo la nueva generación de comunicaciones móviles (5G) aparte de ser transformadora viene a ser disruptiva ante las generaciones anteriores ya que propone nuevas técnicas, modelos y KPIs que prometen habilitar una amplia gama de servicios con alta fiabilidad, ayudando a conformar toda una red heterogénea global móvil interconectada con altos índices de rendimiento.

Las investigaciones sobre los casos de uso de una red 5G y sus requisitos técnicos han sido realizadas por la ITU-R, el 3GPP y la NGMN:

5G admitirá una gran variedad de casos de uso que están surgiendo ahora o surgirán en el futuro. Los diversos casos de uso tienen características y requisitos variables. Es útil agrupar esta gran cantidad de casos de uso emergentes en varias familias. La Figura 2.10 muestra las familias de casos de uso consideradas por la NGMN con un ejemplo de caso de uso para cada familia, se ve además la correspondencia que existe con los casos de uso de la 3GPP y la ITU-R.

En términos generales, ITU-R ha concluido tres casos de uso para abordar la gran variedad de requisitos y características [23]:

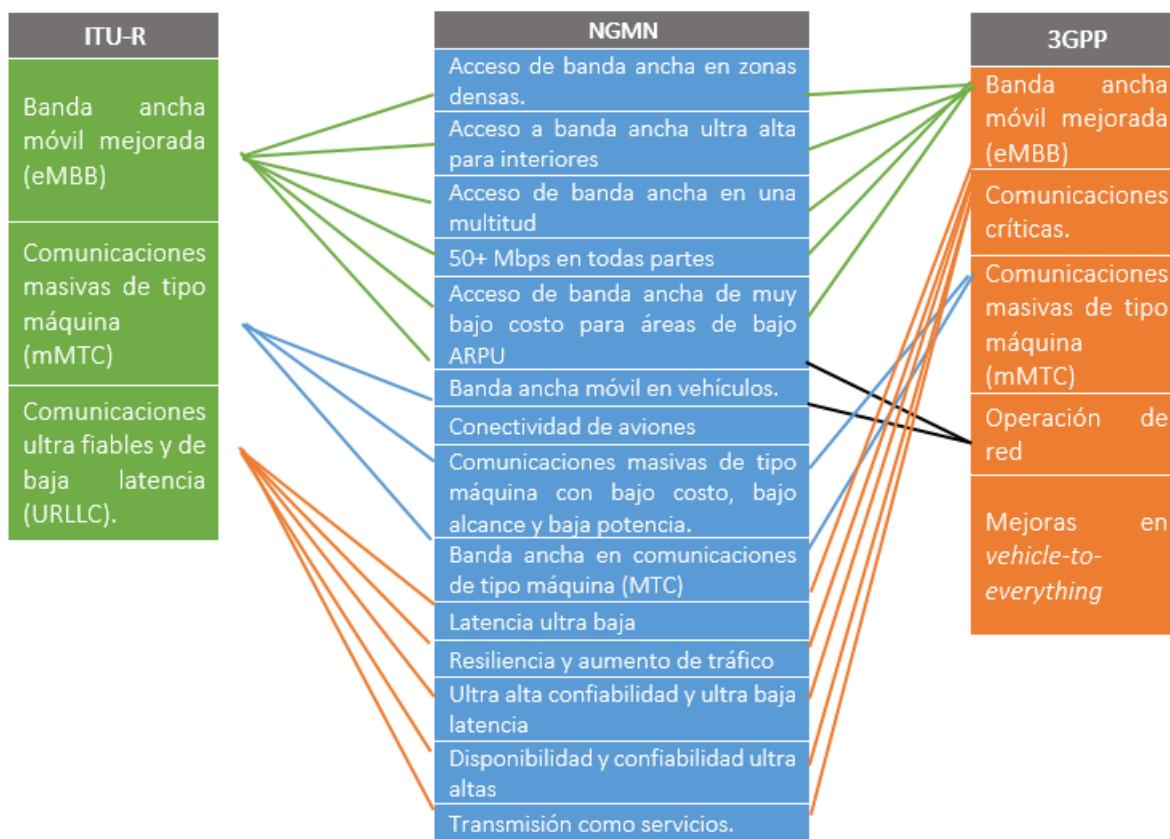


FIGURA 2.10: Comparación de diversos escenarios de uso de la tecnología 5G por la UIT-R, el 3GPP y la NGMN

1. **Banda ancha móvil mejorada (eMBB):** la banda ancha móvil aborda casos de uso centrados en los humanos, como el acceso a contenido multimedia, servicios y datos. La demanda de banda ancha móvil seguirá aumentando, lo que dará lugar a una banda ancha móvil mejorada. El escenario mejorado de uso de banda ancha móvil vendrá con nuevas áreas de aplicación y requisitos, además de las aplicaciones de banda ancha móvil existentes para un mejor rendimiento y una experiencia de usuario cada vez mejor.
2. **Comunicaciones ultra fiables y de baja latencia (URLLC):** este caso de uso tiene requisitos estrictos para capacidades tales como rendimiento, latencia y disponibilidad. Algunos ejemplos incluyen el control inalámbrico de la fabricación industrial o los procesos de producción, cirugía médica remota, automatización de la distribución en una red inteligente, seguridad en el transporte, manejo autónomo de automóviles, etc.
3. **Comunicaciones masivas de tipo máquina (mMTC):** este caso de uso se caracteriza por un gran número de dispositivos conectados que transmiten un volumen relativamente bajo de datos no sensibles al retardo. Los dispositivos deben ser de bajo costo y tener una batería de larga duración.

Se espera que en redes 5G, el internet de las cosas (IoT) atienda las demandas para el caso de uso mMTC. IoT es una emergente y prometedora tecnología que habilitará la interconexión global de objetos físicos (comúnmente dispositivos de bajo consumo) mediante el uso del internet [24]. Una de las características más importantes de esta tecnología es que hace uso de la comunicación máquina a máquina (M2M) con el fin de que los dispositivos se conecten y comuniquen entre sí sin alguna intervención humana.

Para habilitar esta tecnología se requiere del soporte para conexiones masivas, es decir, admitir una gran cantidad de sensores por cada celda. Debido a que la mayoría de estos sensores deben operar durante varios años, la eficiencia energética en las transmisiones inalámbricas es un requisito importante. Finalmente, se debe también reducir el costo de implementación de tales sensores [25].

2.6. MODELADO DEL TRÁFICO EN TELECOMUNICACIONES

Es importante calcular el tráfico en unidades bien definidas mediante modelos matemáticos y determinar la relación existente entre calidad de servicio y capacidad del sistema, de tal manera que la teoría se convierta en una herramienta útil para la planificación de recursos.

Algunos conceptos básicos sobre la teoría del teletráfico se enlistan a continuación:

1. Tráfico: Es el acumulado de peticiones de servicio de todos los usuarios atendidos por la red o por una parte de ella.
2. Recurso o servidor: medio físico, usualmente sólo es capaz de atender un solo servicio.
3. Sistema de colas: conjunto de servidores de uso compartido.

Para caracterizar el tráfico se deben definir previamente los siguientes conceptos:

1. Volumen de Tráfico Cursado. Suma de la duración de todos los servicios atendidos por el sistema.
2. Intensidad de Tráfico Cursado u Ocupación promedio de recursos (a'). Número promedio de servicios atendidos simultáneamente. Se puede obtener mediante dividir el volumen de tráfico cursado entre el tiempo que tomó cursar dicho volumen. También se puede interpretar como el número promedio de servidores ocupados.
3. Intensidad de Tráfico Ofrecido (a). Número promedio de servicios atendidos simultáneamente, si todas las peticiones fueran atendidas.

Suponiendo un sistema que es capaz de atender todas las peticiones de servicio y al que arriban λ peticiones/segundo. Esto implica que en un intervalo T segundos se recibirían $\lambda \cdot T$ peticiones de servicio. Si la duración promedio de estos servicios es μ segundos, entonces el volumen de

tráfico ofrecido (y en este caso también cursado) es $\lambda \cdot \mu \cdot T$ y la intensidad de tráfico ofrecido se reduce a:

$$a = \lambda \cdot \mu \quad (2.26)$$

Para un sistema que atiende todas las peticiones el tráfico cursado y el ofrecido son iguales, sin embargo, cuando se analizan sistemas que no cumplen esta característica, a se vuelve un valor hipotético (la intensidad de tráfico cursado, si todas las peticiones se atendieran), sin embargo, seguirá describiendo la intensidad del tráfico que se ofrece. Es importante mencionar que el desempeño de un sistema (en términos de algún parámetro de calidad de servicio) no va a depender del valor de λ o de μ por sí solos, sino del producto de ellos. Por ejemplo, un sistema se puede saturar tanto por una alta tasa de arribos como por una gran duración del tiempo de servicio.

Ahora bien, para el modelado de llegadas, sean $X_1, X_2, X_3, \dots, X_i$ variables aleatorias tales que:

- X_1 = el intervalo de tiempo entre el inicio del proceso y el primer evento, es decir, la primera llegada,
- X_2 = el tiempo entre llegadas entre la primera y la segunda llegada,
- X_3 = el tiempo entre llegadas entre la segunda y la tercera llegada, y así sucesivamente.

La distribución de la variable aleatoria X_k que representa el tiempo entre llegadas entre la llegada $(k-1)th$ y $(k)th$ es [11]:

$$X_k = \text{Exponential}(\lambda) \quad (2.27)$$

La Función Densidad de Probabilidad (PDF) de la variable aleatoria X_k es la siguiente:

$$P_X(t) = \lambda e^{-\lambda t} \quad (2.28)$$

Y describe la PDF de tiempos entre llegadas en un proceso de Poisson.

Hasta este punto el tráfico sólo se ha caracterizado en término de dos valores promedio: la tasa de arribos (λ) y la duración promedio de los servicios (μ). Estos parámetros representan información parcial de dos variables aleatorias, el tiempo entre arribos (TEA) y el tiempo duración de servicio (TDS). Estos tiempos por su carácter aleatorio en muchos análisis se modelan como variables aleatorias. El tipo de comportamiento aleatorio en las tasas se establece al de una distribución exponencial negativa [26]. La tasa de arribos corresponde una distribución como sigue:

$$f(y) = \lambda e^{-\lambda y} \cdot u(y) \quad (2.29)$$

La duración promedio de servicios corresponde a:

$$f(y) = \frac{1}{\mu} e^{-\frac{1}{\mu} y} \cdot u(y) \quad (2.30)$$

Es necesario establecer fórmulas que relacionen a la cantidad de recursos y el tráfico ofrecido con los parámetros de calidad de servicio, se requiere como paso intermedio determinar la probabilidad de que el sistema se encuentre en determinado estado.

Se considera que el sistema está en estado j , si la suma de los servicios que están siendo atendidos más los servicios en espera es j .

Una técnica que simplifica significativamente el cálculo de las probabilidades de estado es el uso de cadenas de Markov, sin embargo, la solución mediante este método implica el análisis del sistema exclusivamente en el dominio de las probabilidades, por lo que las probabilidades de estado no tienen dependencia del tiempo. Esta independencia del tiempo se consigue sólo si:

1. El sistema se analiza en estado estable, es decir, si el sistema ha estado operando por un intervalo de tiempo lo suficientemente grande, de modo que ya no depende de las condiciones iniciales.
2. Si la probabilidad de que el sistema cambie de estado no depende de cuánto tiempo haya permanecido en el estado actual.

La segunda condición sólo se puede cumplir si las distribuciones del TEA y del TDS son exponenciales negativas [26].

Si el número de servidores que posee un sistema es menor a la cantidad de fuentes que generan el tráfico, y se entiende como fuentes de tráfico a los posibles usuarios, resulta imposible atender a todas las peticiones de servicio de forma instantánea. Básicamente se puede proceder de 2 formas con aquellas llamadas que hallen al sistema saturado:

1. Negarles el servicio. Se tiene un sistema con bloqueo o con pérdidas.
2. Mantenerlas en espera y asignarles servidores cuando sean liberados. Se tiene un sistema con retardo.

En el modelado de los sistemas con bloqueo también se debe tomar en cuenta que sucede con las llamadas que no son atendidas y dependiendo de la naturaleza del servicio analizado se puede considerar que las llamadas bloqueadas regresan al sistema en forma de reintentos o bien son eliminadas en forma definitiva.

Otra consideración de suma importancia en el sistema de colas es tomar en cuenta que tan grande es la cantidad de posibles fuentes de tráfico en comparación con la cantidad de recursos; cuando la cantidad de fuentes es muy grande se puede aproximar con infinito y los análisis se simplifican considerablemente.

También es importante mencionar que hay sistemas en los que las peticiones pueden experimentar bloqueo o retardo. Además de las consideraciones previas: la distribución del TEA y del TDS, la cantidad de recursos y el tamaño de la cola de espera son parámetros que influyen en

el desempeño de un sistema.

Se refiere como λ a la tasa de arribos generada por todas las fuentes de tráfico y se denotó anteriormente μ a la duración promedio de un servicio. El inverso de dicho tiempo se considera la tasa a la que finaliza un servicio en curso. En caso de que existan j servicios en curso la tasa a la que finalizan las llamadas es j/μ . Esta fórmula concuerda con el hecho de que a medida que hay más servicios en curso, menos tiempo se tiene que esperar a que alguno de ellas finalice, es decir, la tasa de finalización se incrementa.

La probabilidad de bloqueo está dada por:

$$P_j = \frac{\frac{(\lambda\mu)^j}{j!}}{\sum_{k=0}^s \frac{(\lambda\mu)^k}{k!}} \quad (2.31)$$

De todas las probabilidades de estado, P_s tiene particular importancia, ya que representa la probabilidad de que el sistema esté saturado, en otras palabras, la probabilidad de bloqueo:

$$P_s = \frac{\frac{a^s}{s!}}{\sum_{k=0}^s \frac{a(k)}{k!}} \quad (2.32)$$

La Ecuación 2.32 fue desarrollada originalmente por el danés A. K. Erlang, por lo que es comúnmente conocida como fórmula de **Erlang-B**.

Notese que en la Ecuación 2.31 el producto $\lambda\mu$ se puede sustituir por a , el tráfico ofrecido, y con esto se corrobora que la calidad del servicio (en este caso la probabilidad de bloqueo) no depende de λ o de μ por si solos, sino de su producto.

2.7. SIMULACIONES A NIVEL DE SISTEMA

Debido a las complicadas estructuras de los sistemas de comunicación celular móvil, no se pueden describir completamente a través de un modelo matemático simple y abstracto. Por lo tanto, se recurre a la simulación para evaluar su rendimiento. Una simulación es casi siempre la imitación de algún proceso o sistema que toma o podría tomar lugar en el mundo real. Estas son implementadas utilizando un paradigma de programación y ejecutadas por computadoras.

La simulación a nivel de sistema se ha utilizado ampliamente para evaluar el rendimiento integral de diferentes sistemas celulares móviles [27]. El rendimiento de estos sistemas se puede reflejar en última instancia por los resultados obtenidos de los programas de simulación.

El escenario para la simulación a nivel de sistema generalmente consiste en una red con múltiples BSs y UEs [véase Figura 2.11]. A diferencia de la simulación a nivel de enlace, la simulación a nivel de sistema se centra en las métricas de rendimiento de la capa de aplicación expresadas por el rendimiento del sistema, la imparcialidad del usuario, la calidad de servicio percibida por

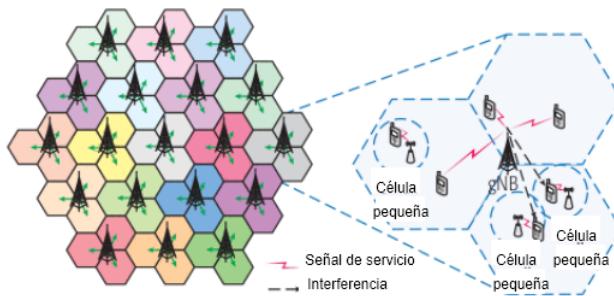


FIGURA 2.11: El diseño usual de una red celular en simulaciones a nivel de sistema

el usuario (QoS), el retraso de la transferencia o la tasa de éxito, etc.

Para realizar un simulador, debe tenerse en cuenta un previo estudio del sistema real o de algún modelo o modelos existentes, junto con un análisis detallado de las variables presentes. Además es necesario hacer distintas suposiciones que faciliten la realización sin comprometer los resultados. Todo esto para permitir una simulación que simplifique el sistema, pero aun siendo capaz de generar resultados que puedan ayudar a su caracterización y a su dimensionamiento.

En [28] se describe a un sistema discreto como: “[...] aquel en el cual las variables de estado cambian únicamente en un número discreto de instantes en el tiempo”. Por lo que la simulación de eventos discretos es la implementación en hardware de un sistema en el que sus variables de estado cambian de tal forma con el arribo de eventos, los cuales se tratan de ocurrencias que se presentan de forma instantánea y cuentan con un nacimiento y una muerte.

2.7.1. Lenguajes de Programación para simulaciones orientadas a eventos discretos(DES)

Considerando solamente los lenguajes que sean de código abierto, se tiene a *JAVA (DESMO-J y Ptolomeo I)*, *Python* y *C++ (SystemC y PowerDEVS)* como los lenguajes más conocidos y de entre estos sobresalen (*Python*) y *SystemC (C++)* como los que han tenido más soporte y actualización (2018), lo cual será un factor importante al elegir el lenguaje.

Simpy es una librería basada en procesos, con la cual se pueden definir diferentes entornos, y todos los procesos interactúan mediante eventos con el mismo entorno y entre ellos mismos. Durante su estancia, los procesos crean eventos. Cuando un proceso produce un evento, el proceso se suspende, *SimPy* reanuda el proceso, cuando ocurre el evento. Varios procesos pueden esperar el mismo evento, *SimPy* los reanuda en el mismo orden en que dieron lugar a ese evento [29]. Además proporciona varios tipos de recursos compartidos para modelar puntos de congestión de capacidad limitada (p. ej. servidores).

La documentación de SimPy contiene tutoriales, guías detalladas y una gran cantidad de ejemplos [29]. SimPy se lanza como software de código abierto bajo la licencia del MIT. La primera versión fue lanzada en diciembre de 2002 y hoy en día su última versión estable es la 3.0.11 / 16 de noviembre de 2018.

Python es un lenguaje de programación dinámico de alto nivel, interpretado y de propósito general que se enfoca en la legibilidad del código. La sintaxis en Python ayuda a los programadores a codificar en menos pasos en comparación con Java o C ++ [30]. Python es ampliamente utilizado en organizaciones más grandes debido a sus múltiples paradigmas de programación. Usualmente involucran programación funcional imperativa y orientada a objetos.

2.7.2. Python

Ventajas o beneficios de Python [30]:

1. Característica de integración
2. Productividad mejorada del programador
3. Amplias librerías de soporte:

Scipy es una librería de Python utilizada para la informática científica y la informática técnica.

NumPy es una librería de Python utilizada para la informática científica que, aparte de sus usos científicos, puede utilizarse como un contenedor multidimensional para datos genéricos.

Matplotlib es una librería para la generación de gráficos a partir de datos contenidos en listas o arrays en el lenguaje de programación Python y su extensión matemática NumPy.

Desventajas o Limitaciones de Python:

1. Lenguaje interpretado: Python se ejecuta con la ayuda de un intérprete en lugar del compilador, lo que hace que se ralentice.

El mecanismo que degrada el desempeño de CPython es la ejecución de bytecode por varios hilos a la vez, conocido como *Global Interpreter Lock* o GIL. Este es un mecanismo utilizado en intérpretes de lenguaje de computadora para sincronizar la ejecución de subprocesos y que sólo un subproceso nativo pueda ejecutarse a la vez. Un intérprete que usa GIL siempre permite que se ejecute exactamente un subproceso a la vez, incluso si se ejecuta en un procesador multinúcleo.

2.7.3. Multiprocesamiento

Afortunadamente, existen métodos para evitar este comportamiento del intérprete, haciendo que se usen todos los núcleos del PC con ayuda de la librería *multiprocessing* [31].

El módulo de multiprocessing permite al programador aprovechar al máximo múltiples procesadores en paralelo. Se ejecuta tanto en Unix como en Windows, es importante no confundir la técnica con la concurrencia (esta hace uso de múltiples hilos) [véase Figura 2.12].

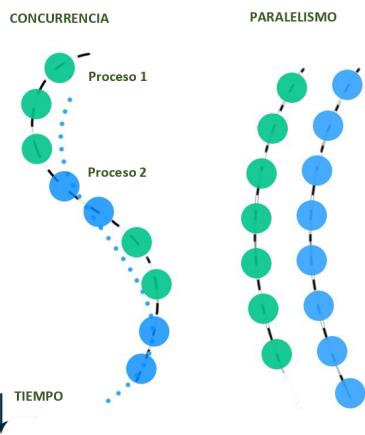


FIGURA 2.12: Concurrencia Vs Paralelismo

Capítulo 3

Estado del Arte

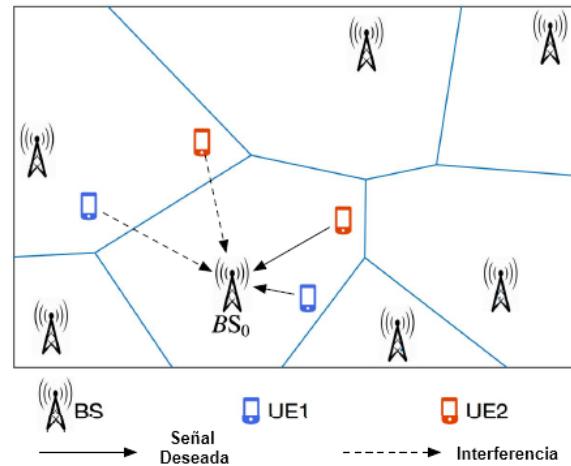
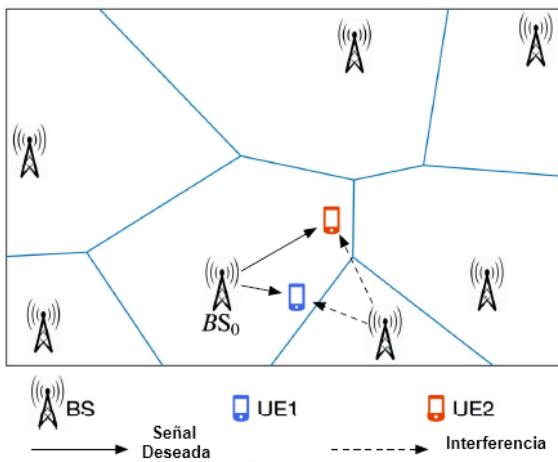
En este Capítulo se encuentran trabajos anteriormente realizados que están relacionados con este proyecto, esto con la finalidad de presentar antecedentes, y dejar ver las discrepancias y similitudes existentes.

El trabajo de dimensionar los sistemas de comunicación móviles es una necesidad recurrente en cada nueva generación celular. En [32], se encuentra un proyecto terminal realizado por alumnos de la UPIITA, en el cual se realizó un simulador, bajo el paradigma de eventos discretos. El sistema objetivo era el celular 4G, con un enfoque en distintos esquemas de reúso de frecuencias y calendarizadores para obtener resultados sobre qué combinación de estos y bajo qué condiciones el sistema tenía un mayor desempeño. En este proyecto la simulación se llevó a cabo utilizando Matlab.

Por otra parte, el reciente crecimiento de los casos de uso de IoT en una amplia gama de aplicaciones ha traído la necesidad de una mejor caracterización del tráfico tipo máquina. Reconociendo la importancia de esta cuestión, la 3GPP ha propuesto dos modelos para el tráfico tipo máquina. Se tratan de modelos de tráfico agregado . El primero modela el tráfico generado aleatorio, mientras que el segundo modela el tráfico síncrono con el tiempo. Dado que los modelos sólo se centran en el tráfico agregado, estos pueden no ser apropiados para el análisis práctico en algún sistema que esté más enfocado en el tráfico y requiera mayor precisión.

En [33], los autores consideraron modelar el tráfico de dispositivos IoT conectados a través de tecnologías LPWAN. Debido a las diversas aplicaciones de IoT, no es trivial tener un solo modelo de tráfico para representarlas a todas, el tráfico puede clasificarse ampliamente como periódico, activado por eventos o una combinación de ambos. Evaluaron el rendimiento de LoRaWAN, en presencia de un híbrido de ambos tipos de tráfico, donde los eventos se propagan espacialmente a lo largo del tiempo. Utilizaron el modelo CMMPP para representar dicho tráfico característico de dispositivos IoT, que suelen ser activados por eventos.

De igual manera, pero ahora con un enfoque en sistemas celulares LTE, en [34] los autores analizaron el tráfico M2M con velocidad de datos variable bajo el supuesto de que la red LTE tiene recursos limitados. Los resultados muestran las características del tráfico M2M de una manera más realista, identificando las diferencias del tráfico estándar en la red celular. Revelan que el tráfico cuenta con la propiedad de auto similitud sólo para una gran cantidad de MTC.



Estos trabajos muestran el uso de modelos de tráfico CMMPP para evaluar el impacto de la tecnología IoT. La integración del tráfico M2M será una parte inevitable de la evolución de las redes. En este proyecto se implementó el modelo CMMPP, en una red de 5G, teniendo en cuenta la arquitectura NB-IoT y distintas aplicaciones de IoT.

Ahora bien, del lado de los esquemas de acceso no ortogonales (NOMA) en redes de quinta generación, se sigue trabajando en las propuestas de su implementación. En [35], los autores proponen un sistema usando geometría estocástica (PPP) para modelar un ambiente inalámbrico denso que admite NOMA tanto en el enlace de subida como en el enlace de bajada.

En la implementación de NOMA propuesta, se tienen dos esquemas de emparejamiento de usuario: uno aleatorio y otro selectivo:

1. Cuando el agrupamiento es aleatorio, los UE son seleccionados aleatoriamente.
2. Cuando el agrupamiento es selectivo, el primer UE deberá tener una relación señal-interferencia más ruido (SINR) por encima del umbral T_1 y el segundo UE tiene un SINR por debajo del umbral T_2 , $T_2 \leq T_1$.

Consideraron un error de propagación SIC durante el proceso de decodificación por parte del UE. Además, optaron por una estrategia de asignación de potencia fija, donde la potencia de enlace de bajada asignada a un UE está predefinida y permanece sin cambios.

Las ganancias implementan el desvanecimiento de Rayleigh entre la BS_0 y UE_i . La ganancia en potencia de desvanecimiento Rayleigh entre BS y UE sigue una distribución exponencial con

media 1 y se distribuye de forma independiente e idéntica (i.i.d.)

En el enlace descendente (DL), agregaron perdidas por trayectoria con un exponente de pérdida y calcularon la interferencia entre celdas acumulativa de todas las bases adyacentes. En el enlace ascendente (UL), la interferencia inter-celdas proviene de todos los otros UEs que comparten la misma sub-banda. [Véanse Figuras 3.1, 3.2]

Este trabajo se enfocó en realizar NOMA para sistemas 5G, sin embargo, no consideraron la actuación de dispositivos IoT, los cuales al tener diferentes calidades de servicio, impactarían en la toma de decisiones del modelo de acceso múltiple, más en concreto en el emparejamiento de usuarios.

En [36] se trabajó en emplear NOMA para mejorar la densidad de conexión en los sistemas NB-IoT. En su propuesta cada subportadora puede dar servicio máximo a dos dispositivos con distintos requisitos de QoS. Formularon problemas de asignación de potencia de transmisión y subportadoras conjuntas para los modos singleton y multitono. Además, propusieron algoritmos heurísticos con baja complejidad y dieron como resultado un rendimiento cercano a las soluciones óptimas y subóptimas en ambos casos. Los resultados de la simulación mostraron que el uso de NOMA aumentó la densidad de conexión hasta en un 87 % en comparación con OMA en el modo singleton y en el modo multitone, la densidad de conexión también se incrementó hasta en un 24 %.

En [37] desarrollaron un esquema NOMA en el dominio de potencia con agrupación de usuarios en un sistema NB-IoT. Resolvieron un problema de optimización para maximizar el rendimiento total de la red al optimizar la asignación de recursos de los dispositivos MTC y la agrupación de NOMA al tiempo que se satisfacen los requisitos de potencia de transmisión y QoS. Además, diseñaron un algoritmo heurístico eficiente para resolver el problema de optimización propuesto mediante la agrupación NOMA y la asignación de recursos a dispositivos de tipo máquina.

En su modelo de sistema, consideraron el escenario de una única celda (eNB), que admite dispositivos de tipo máquina operando con la tecnología NB-IoT. Asumieron que no hay interferencia proveniente de otras células vecinas.

Propusieron un esquema NOMA en el dominio de la potencia agrupando (de entre 1 a 4) dispositivos mMTC y URLLC en una red NB-IoT como se muestra en la Figura 3.3 . Según el esquema NOMA, los dispositivos mMTC y URLLC comparten cada subportadora (subcanal) y transmiten datos de manera no ortogonal. Por lo tanto, los dispositivos se dividen en diferentes grupos, llamados “clusters”. Para decodificar con éxito los mensajes de la suma de mensajes recibida, el eNB emplea el esquema SIC. Por lo tanto, los usuarios deben ordenarse en cada grupo teniendo en cuenta el método SIC.

Se ha investigado bastante acerca del desempeño de los esquemas NOMA para dispositivos IoT, sin embargo nuestra propuesta resulta diferente ya que además de utilizar PD-NOMA,

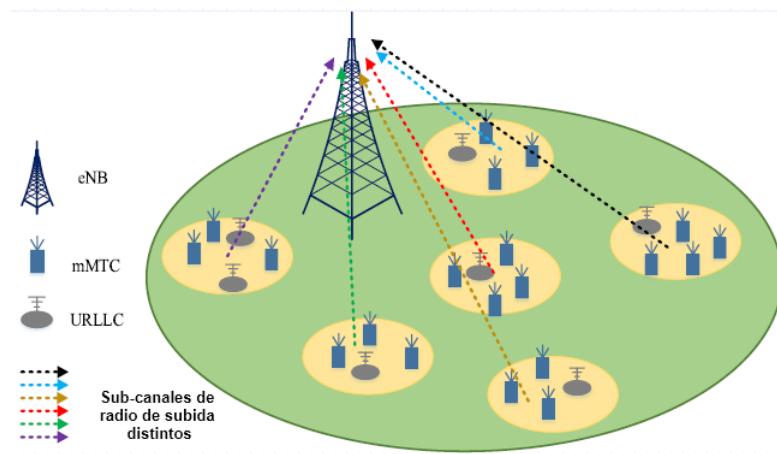


FIGURA 3.3: Grupos NOMA que incluyen dispositivos mMTC y URLLC, donde los dispositivos MTC comparten los subcanales asignados a cada clúster NOMA.

se implementó un despliegue de UEs usando un PPP, un modelo de canal más realista y por último, un modelo de tráfico de tipo fuente.

Capítulo 4

Análisis

Este capítulo contiene un análisis de las diversas aplicaciones que tienen los dispositivos IoT, además se abordan las técnicas y modelos necesarios para diseñar un sistema que brinde servicio a estos dispositivos. Se comienza con una breve descripción de la red 5G y el papel que los dispositivos IoT tienen en esta. Posteriormente se profundiza en el caso de uso mMTC, donde se mencionan los escenarios más comunes de implementación, su clasificación y sus características. Se revisa también el estándar actual (NB-IoT¹) que se ha seleccionado como una de las tecnologías que brindará servicio a los dispositivos IoT en la red 5G.

Por último, se presentan los modelos que fueron usados para caracterizar el despliegue, la condición del canal y la generación de tráfico de comunicaciones de tipo máquina. Se presenta también una propuesta de técnica de acceso múltiple al medio no ortogonal (NOMA), que permitiría a la tecnología NB-IoT brindar servicio a más dispositivos IoT usando agrupamientos (*clustering*).

4.1. REDES 5G/IoT

El Internet de las cosas (IoT) tendrá un importante papel en las redes 5G. La actual red celular LTE (4G) no está diseñada para satisfacer las demandas de conectividad de múltiples dispositivos, velocidad de datos, calidad de servicio (QoS) de baja latencia y de eficiencia energética. Para abordar estos desafíos, 5G contempló a IoT como uno de sus pilares [38].

4.2. CLASIFICACIÓN Y ANÁLISIS DE LOS ÁMBITOS DE IoT

Se tomó como referencia el trabajo realizado en [39] como una guía de los servicios que se espera los nodos IoT brinden en un futuro próximo. Los servicios se presentan en 8 dominios: edificios inteligentes y vivienda (*Smart buildings and living*), cuidado de la salud inteligente (*Smart healthcare*), medio ambiente inteligente (*Smart environment*), ciudades inteligentes (*Smart city*),

¹Muchos de los modelos aquí propuestos están basados en trabajos de la 3GPP, como se revisó en el Capítulo II, la 3GPP, es una organización que está respaldada por organismos alrededor de todo el mundo, además de que se trata del grupo que estandariza tecnologías como LTE-M y NB-IoT, de manera que es una indudable referencia en su ahora inmersión en la estandarización de 5G.

energía inteligente (*Smart energy*), transporte y movilidad inteligentes (*Smart transport and mobility*), fabricación y venta inteligentes (*Smart manufacturing and retail*), agricultura inteligente (*Smart agriculture*). Para cada uno de los dominios se especifican aplicaciones típicas que se podrían encontrar, sus características de tráfico y las tecnologías de red más adecuadas para darles servicio entre otras cosas.

La primera parte del análisis correspondió a la selección de los dominios que resultasen adecuados para el sistema que se diseñó. Estos fueron aquellos cuya red que les brindará servicio primordialmente es una red de área amplia de bajo consumo (LPWAN, *Low Power Wide Area Network*). Por el contrario, algunas de las aplicaciones en los dominios antes mencionados están pensadas para redes con tecnologías como RFID, *Bluetooth* o *ZigBee*. A continuación se presenta la caracterización de cada uno de los dominios que en [39] se consideran viables para redes LPWAN.

4.2.1. Ciudades inteligentes (*Smart Cities*):

Con la rápida concentración de la población en zonas urbanas, se ha convertido en una prioridad la reducción del uso de recursos públicos, así como la reducción de costos de operación del día a día de una ciudad. Las aplicaciones en este dominio tratan justamente de abordar estos problemas y los servicios que brindan son bastante variados. Los ejemplos van desde el control de luminarias hasta el manejo de desechos, estos y otros pueden encontrarse en la *Tabla 4.1*, acompañados de más información tal como la caracterización de su tráfico y su demanda de QoS.

4.2.2. Ambiente inteligente (*Smart Environment*)

Este dominio comprende las aplicaciones que se encargan de monitorear lo que ocurre a nuestro alrededor. Una de las ventajas de monitorear el ambiente es el poder reaccionar con antelación a eventos que de otra forma causarían muchos daños. En la *Tabla 4.2* podemos encontrar la caracterización de las aplicaciones consideradas en [39] para Ambiente Inteligente.

4.2.3. Energía inteligente (*Smart Energy*):

El dominio de Energía Inteligente (*Smart Energy*) contempla mejoras en la distribución y el consumo de fuentes de energía o recursos vitales, tales como la electricidad, el gas y el agua. Actualmente el foco de atención está en la electricidad ya que existe una tendencia más marcada hacia su ahorro y la utilización de fuentes renovables.

Los nodos de IoT que operen dentro de este dominio podrían monitorear las condiciones cambiantes de la red eléctrica, para posteriormente generar una reconfiguración apropiada del servicio. En la *Tabla 4.3* podemos encontrar la caracterización descrita en [39] para distintas aplicaciones de Energía Inteligente.

TABLA 4.1: Características de las aplicaciones de Ciudades Inteligentes

<i>Servicio</i>	<i>Tamaño de red</i>	<i>Tasa de tráfico</i>	<i>Demandas de QoS</i>	<i>Fuente de energía</i>
<i>Monitoreo del consumo de agua y electricidad en la ciudad</i>	Media a grande, cientos a miles de dispositivos	Periódico, 1 msj cada 10 min por dispositivo	Baja, tolerante al retardo 1 min	Alimentado por la red eléctrica/ auto-alimentado
<i>Control de iluminación</i>	Grande, miles de dispositivos	Aleatorio, poco frecuente	Media, tolerante al retardo 15 seg	Alimentado por la red eléctrica
<i>Vigilancia de establecimientos</i>	Grande, miles de dispositivos	Aleatorio, poco frecuente	Media, tolerante al retardo 10 seg	Alimentado por batería
<i>Control del tráfico</i>	Grande, miles de dispositivos	Periódico, 1 msj cada 10 min por dispositivo, aleatorio para alarmas	Media, tolerante al retardo 15 seg, alta para alarmas	Alimentado por batería
<i>Mantenimiento de desechos</i>	Grande, miles de dispositivos	Aleatorio, poco frecuente	Media, tolerante al retardo 30 seg	Alimentado por batería
<i>Monitoreo de condiciones urbanas</i>	Media a grande, cientos a miles de dispositivos	Periódico, 1 msj cada 15 min por dispositivo, aleatorio para alarmas	Media, tolerante al retardo 30 seg, alta para alarmas	Alimentado por batería
<i>Monitoreo de la salud estructural de edificios</i>	Media a grande, cientos a miles de dispositivos	Periódico, 1 msj cada 15 min por dispositivo, aleatorio para alarmas	Media, tolerante al retardo 30 seg, alta para alarmas	Alimentado por batería

TABLA 4.2: Características de las aplicaciones de Ambiente Inteligente

Servicio	Tamaño de red	Tasa de tráfico	Demandas de QoS	Fuente de energía
<i>Detección de incendios forestales</i>	Media a grande, cientos a miles de dispositivos	Aleatorio, poco frecuente	Media, tolerante al retardo 15 seg	Alimentado por batería
<i>Detección de terremotos</i>	Media a grande, cientos a miles de dispositivos	Aleatorio, poco frecuente	Alta, tolerante al retardo 5 seg	Alimentado por batería
<i>Detección de Tsunamis</i>	Media a grande, cientos a miles de dispositivos	Aleatorio, poco frecuente	Alta, tolerante al retardo 5 seg	Alimentado por batería
<i>Detección de derrumbes y avalanchas</i>	Media a grande, cientos a miles de dispositivos	Aleatorio, poco frecuente	Alta, tolerante al retardo 5 seg	Alimentado por batería
<i>Monitoreo de actividad volcánica</i>	Pequeña, 10s de dispositivos	Aleatorio, poco frecuente	Alta, tolerante al retardo 5 seg	Alimentado por batería
<i>Monitoreo de la contaminación del aire</i>	Media a grande, cientos a miles de dispositivos	Periódico, 1 msj cada 15 min por dispositivo	Media, tolerante al retardo 15 seg	Alimentado por batería
<i>Rastreo de vida salvaje.</i>	Media, cientos de dispositivos	Periódico, 1 msj cada 30 min por dispositivo	Baja, tolerante a unas horas	Alimentado por batería

TABLA 4.3: Características de las aplicaciones de Energía Inteligente

Servicio	Tamaño de red	Tasa de tráfico	Demandas de QoS	Fuente de energía
<i>Medición inteligente</i>	Media a grande, 1 dispositivo por hogar	Periódico, 1 msj cada 15 min por dispositivo	Media, tolerante al retardo 15 seg	Alimentado por la red eléctrica/ Baterías
<i>Gestión de activos</i>	Media a grande, cientos a miles de dispositivos	Periódico, 1 msj cada 15 min por dispositivo	Media, tolerante al retardo 15 seg	Alimentado por la red eléctrica/ Baterías
<i>Detección de interrupciones en el servicio</i>	Media a grande, 1 dispositivo por hogar	Aleatorio, poco frecuente	Alta, tiempo real	Alimentado por la red eléctrica/ Baterías

TABLA 4.4: Características de las aplicaciones de Transporte y Movilidad inteligentes

<i>Servicio</i>	Tamaño de red	Tasa de tráfico	Demandas de QoS	Fuente de energía
<i>Automatización de vehículos</i>	Grande, miles de dispositivos	Periódico, 1 msj cada 24 hrs por vehículo.	Baja, tolerante al retardo 1 min	Alimentado por batería del vehículo
<i>Localización y monitoreo de vehículos</i>	Grande, miles de dispositivos	Periódico, 1 msj cada 30 seg por vehículo	Media, tolerante al retardo 10 seg	Alimentado por batería del vehículo
<i>Monitoreo de la calidad del embarque</i>	Media, cientos de dispositivos	Periódico, 1 msj cada 15 min por dispositivo	Media, tolerante al retardo 15 seg	Alimentado por batería
<i>Control dinámico de semáforos</i>	Grande, miles de dispositivos	Periódico, 1 msj cada min por dispositivo	Alta, tolerante al retardo 5 seg	Alimentado por la red eléctrica
<i>Monitoreo de las condiciones del camino</i>	Grande, miles de dispositivos	Aleatorio, poco frecuente	Media, tolerante al retardo 30 seg	Alimentado por batería

4.2.4. Transporte y movilidad inteligentes (*Smart Transport and Mobility*):

Tanto el crecimiento urbano como el crecimiento de las fuentes de transporte de pasajeros y de mercancías, han creado la necesidad de administrar más eficientemente la movilidad en ciudades y carreteras.

El objetivo de aplicaciones IoT en el dominio de Transporte y movilidad inteligentes (*(Smart Transport and Mobility)*) es ayudar a resolver el problema de movilidad tanto de pasajeros como de mercancías, haciéndolo más rápido, más barato y más seguro. En la *Tabla 4.4* se encuentra la caracterización presentada en [39] para distintas aplicaciones de este dominio.

4.3. CARACTERÍSTICAS DEL ESCENARIO A IMPLEMENTAR

Las distintas aplicaciones presentadas en la sección anterior muy seguramente se verán desplegadas en un futuro próximo en ciudades alrededor de todo el mundo. Una asercción que resulta importante es la gran cantidad de aplicaciones que se esperan para IoT [3], pero en este análisis se tomaron en consideración únicamente aquellas que operan en una red LPWAN. Para estas aplicaciones, una red celular de 5G podría ser la más idónea para brindarles servicio.

En las aplicaciones seleccionadas, lo primordial es tener un bajo consumo y complejidad de los dispositivos, además de una amplia cobertura y permitir una gran densidad de dispositivos [40]. Esto contrasta con las aplicaciones más inclinadas a los casos de uso eMBB y URLLC, para las cuales lo primordial es el amplio ancho de banda disponible para las aplicaciones y una menor latencia, respectivamente.

4.3.1. Análisis de las aplicaciones de IoT y selección de casos considerados

De todos los servicios descritos en las tablas anteriores [*Tabla 4.1...4.4*], se decidió implementar solamente un grupo diverso de aplicaciones provenientes de distintos dominios. Este grupo se seleccionó de tal manera que fuera representativo de distintos comportamientos y requerimientos de QoS. El escenario propuesto consta de los servicios de: control de iluminación, monitoreo del consumo de agua y electricidad, detección de terremotos, monitoreo de contaminación del aire, detección de interrupciones en el servicio (agua, luz, gas), control dinámico de semáforos y un último servicio que se llamó “otros dispositivos mMTC”. Este último representó el conglomerado de todas las demás aplicaciones IoT que estarán presentes en la red y no corresponden a una de las primeras aplicaciones seleccionadas.

Se decidió entonces situar la simulación en un escenario urbano micro celular con nodos en exteriores, en el que se encontrarán todas las aplicaciones mencionadas en el párrafo anterior. El escenario urbano micro representa a zonas urbanas densamente pobladas. La elección de un escenario urbano micro celular se debió a que en este se puede experimentar una congestión producida por las MTC y HTC más facilmente. Otra razón fue que en un escenario tal se tiene presente una mayor diversidad de aplicaciones IoT, con distintas características de tráfico y requerimientos de QoS.

En la *Tabla 4.5* se presentan las aplicaciones IoT del escenario urbano que se implementaron.

A continuación se hace una descripción de la *Tabla 4.5*. Se tienen tres aplicaciones con una alta demanda de QoS, después dos con una demanda media y finalmente una con demanda baja. Las distintas demandas de QoS se traducen en diferentes tolerancias a la latencia, las cuales van desde los minutos hasta aquellas que requieren de una respuesta en el orden de segundos. Otra característica que se consideró para decidir entre las aplicaciones fue su tráfico, se trató entonces de tener nodos con distintos períodos de transmisión, por ejemplo el servicio que brindan nodos monitoreando la contaminación del aire tiene un periodo de 15 minutos, mientras que el de nodos controlando los semáforos es de 1 minuto. Existen también nodos con tasas de transmisión aleatorias como el caso de los nodos de detección de terremotos.

La decisión de agregar un servicio más (Otros dispositivos mMTC) que fuera genérico surge a raíz de la necesidad de representar en la red los dispositivos restantes con comportamientos de lo más diversos.

TABLA 4.5: Aplicaciones seleccionadas para la simulación

Servicio	Tamaño de red	Tasa de tráfico	Demanda de QoS	Fuente de energía
Control de iluminación (Ciudad Inteligente)	Grande, miles de dispositivos	Aleatorio, poco frecuente	Media, tolerante al retardo 15seg	Alimentado por la red eléctrica
Monitoreo del consumo de agua y electricidad en la ciudad (Ciudad Inteligente)	Media a grande, cientos a miles de dispositivos	Periódico, 1 msj cada 10 min por dispositivo	Baja, tolerante al retardo 1min	Alimentado por la red eléctrica/ auto-alimentado
Detección de terremotos (Ambiente Inteligente)	Media a grande, cientos a miles de dispositivos	Aleatorio, poco frecuente	Alta, tolerante al retardo 3seg	Alimentado por batería
Monitoreo de contaminación del aire (Ambiente Inteligente)	Media a grande, cientos a miles de dispositivos	Periódico, 1 msj cada 15 min por dispositivo	Media, tolerante al retardo 15seg	Alimentado por batería
Control dinámico de semáforos (Transporte y Movilidad Inteligentes)	Grande, miles de dispositivos	Periódico, 1 msj cada min por dispositivo	Alta, tolerante al retardo 5seg	Alimentado por la red
Otros dispositivos mMTC	Grande, miles de dispositivos	Aleatorio, poco frecuente	Alta, tolerante al retardo 5seg	Alimentado por batería

Finalmente se consideraron también dispositivos URLLC en el simulador. Esto permitió realizar un agrupamiento de dispositivos mMTC y URLLC para que pudieran compartir recursos. Los detalles de la implementación son discutidos en los siguientes capítulos.

4.3.2. Análisis de las tecnologías para IoT y selección de casos considerados

NB-IoT y LTE-M (*LTE for MTC*) son dos tecnologías LPWA desarrolladas para aplicaciones IoT. Ambas tecnologías contemplan comunicaciones celulares con un ancho de banda bajo, y conectan dispositivos que necesitan transmitir pequeñas cantidades de datos, a bajo coste. Debido a que se espera que NB-IoT y LTE-M cumplan con los requerimientos LPWAN del estándar 5G, 3GPP indicó a ITU-R que ambas tecnologías formarán parte estándar. Estas tecnologías

coexistirán con los demás componentes de 5G NR que en conjunto brindarán servicio a los distintos casos de uso. Esto convierte a NB-IoT y LTE-M parte de 5G [41].

Las redes de largo alcance LPWAN utilizan tecnología capaz de transferir mensajes a decenas de kilómetros de distancia y cubrir una amplia área. Son redes especializadas en interconectar dispositivos en ambientes restringidos, de difícil acceso o que simplemente buscan reducir el consumo de energía en estos, manteniendo un bajo costo y complejidad. Por lo tanto se concentran en un eficiente consumo de la energía y una cobertura amplia [39]. Esto se acopla bastante bien a lo que requieren los nodos de IoT pertenecientes a las aplicaciones seleccionadas en la sección anterior.

Las tecnologías que darán servicio a dispositivos IoT son entonces:

1. *eMTC (enhanced Machine Type Communication:)* Forma parte de la familia LTE-M y es una evolución de LTE optimizada para IoT. Se desarrolló con el objetivo de una eficiencia energética.
2. *NB-IoT (Narrow-Band Internet of Things:)* fue estandarizado en el *release 13* de 3GPP y se espera que consiga dar servicio a más dispositivos con energía limitada que eMTC. NB-IoT no requiere ningún desarrollo adicional de redes ya que se implementa en funcionalidades ya existentes de LTE [39] y en un futuro será implementado directamente en la banda de frecuencias de 5G NR [véase Figura 4.1] [41].

El soporte para IoT masivo (mIoT) ya se brinda en las redes LTE de hoy con NB-IoT y eMTC. Estas tecnologías se complementan entre sí y existe una tendencia emergente de los proveedores de servicios de implementar una red común que admita ambas tecnologías. eMTC es adecuada para casos de uso que requieren un rendimiento relativamente más alto, una latencia más baja y soporte de voz, mientras que la tecnología NB-IoT es conveniente para casos de uso que toleran demoras pero requieren una cobertura extendida [41].

Según [3] a finales de 2024, se espera que NB-IoT y eMTC representen cerca del 45 por ciento de todas las conexiones CIoT (*Celular IoT*). Además, en el futuro NB-IoT y eMTC podrán coexistir completamente en las bandas de frecuencia de 5G NR, *Figura 4.1*.

En la *Tabla 4.6* se pueden encontrar características de estas tecnologías antes descritas, tales como la banda de frecuencia a la que operan y su tasa de transmisión. Si bien pareciera que la diferencia entre ambas tecnologías es sutil, en realidad, esta marca una clara pauta en el servicio que pueden brindar.

En la *Figura 4.2* se puede observar otra comparación entre ambas tecnologías pero en esta ocasión se comparan las aplicaciones a las que NB-IoT y LTE-M estarían dando servicio preferentemente. A la izquierda de la *Figura 4.2* tenemos las aplicaciones LPWAN a las que NB-IoT daría servicio. Estas aplicaciones coinciden con una menor velocidad de transferencia y mayor tolerancia a la latencia mientras que a la derecha se agrupan las aplicaciones que requieren

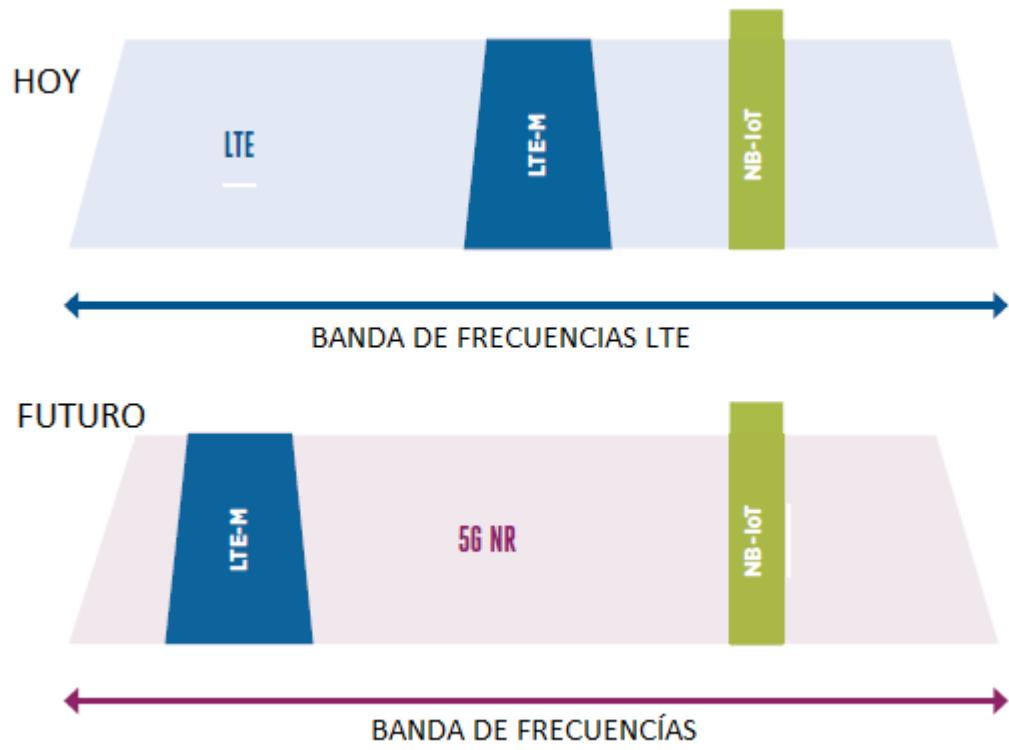


FIGURA 4.1: 5G NR con LTE-M y NB-IoT en banda

una comunicación de baja latencia y una mayor tasa de transmisión. A estas últimas aplicaciones les estaría dando servicio preferentemente la tecnología eMTC.

Se decidió entonces concentrarse en la tecnología NB-IoT puesto que la totalidad de los servicios que se considerarán en nuestro sistema pueden situarse a la izquierda de la *Figura 4.2*, donde se presenta una mínima movilidad de los dispositivos. Por ejemplo el control de la iluminación y el control dinámico de los semáforos podríamos colocarlos en *Iluminación pública y Ciudades inteligentes* respectivamente, mientras que el monitoreo de consumo energético y el de la condición del aire podrían corresponder a *Medidores inteligentes*, de manera que el único servicio que se sitúa en los límites de la tecnología NB-IoT o del otro lado incluso es el de dispositivos URLLC. Debido a esto se seleccionó a NB-IoT como la tecnología fundamental para diseñar el simulador a la que después se agregaron mejoras propuestas en otros trabajos.

La *Figura 4.3* muestra las distintas tecnologías con las que 5G NR estaría trabajando para poder brindar servicio al amplio espectro de casos de uso de MTC [40]. La tecnología NB-IoT podemos situarla en las frecuencias de operación baja y con una tolerancia al retardo mayor que la mayoría de las demás tecnologías.

TABLA 4.6: Características de las tecnologías de red para IoT en la red celular

Tecnología	Banda de Frecuencia	Rango	Tasa de transmisión	Vida de la batería	Topología	Estandarización	Grupo
NB-IoT	450 MHZ – 3.5 GHz (Espec tro de 2G/3G/4G)	10-15 km	250 kbps	10+ años	Estrella	Abierta	3GPP
eMTC	450 MHZ – 3.5 GHz (El mismo que LTE)	10-15 km	1 Mbps	10+ años	Estrella	Abierta	3GPP

4.4. ANÁLISIS DEL ESTÁNDAR NB-IoT

El estándar NB-IoT fue especificado en el reporte TR 45.820 (*release 13*) de la 3GPP [42]. Sus parámetros fundamentales son:

Para el enlace de subida (*uplink*), como su nombre lo indica, tiene un ancho de banda estrecho de 180 kHz y un espacio de sub-portadora de 3.75 kHz (ancho de banda de transmisión mínimo para un dispositivo). Por lo tanto puede asignar 48 sub-portadoras [véase Figura 4.4].

En el enlace de bajada (*downlink*), se conserva la estructura de transmisión del enlace descendente de *Long Term Evolution* (LTE) con un espaciado de sub-portadora de 15 kHz. Por lo tanto, NB-IoT puede proporcionar velocidades de datos de casi 250 kb / s en el enlace descendente y 20 kb / s en el enlace ascendente.

Es preciso puntualizar que para lograr una mayor tasa de datos, de acuerdo con el Teorema de Shannon-Hartley (Ecuación 2.15), el ancho de banda debe ser elevado o se debe tener una relación S/N alta. Para el caso de NB-IoT se cuenta con un ancho de banda muy pequeño (3.75KHz), por lo cual alcanzar una buena relación S/N (S/I para sistemas celulares) es de suma importancia.

4.4.1. Modos de operación

Independiente (*stand-alone*).

NB-IoT puede implementarse como una portadora autónoma utilizando cualquier espectro disponible con un ancho de banda superior a 180 kHz. Esto se conoce como la implementación stand-alone. Un caso de uso de este despliegue autónomo es que un operador GSM despliegue

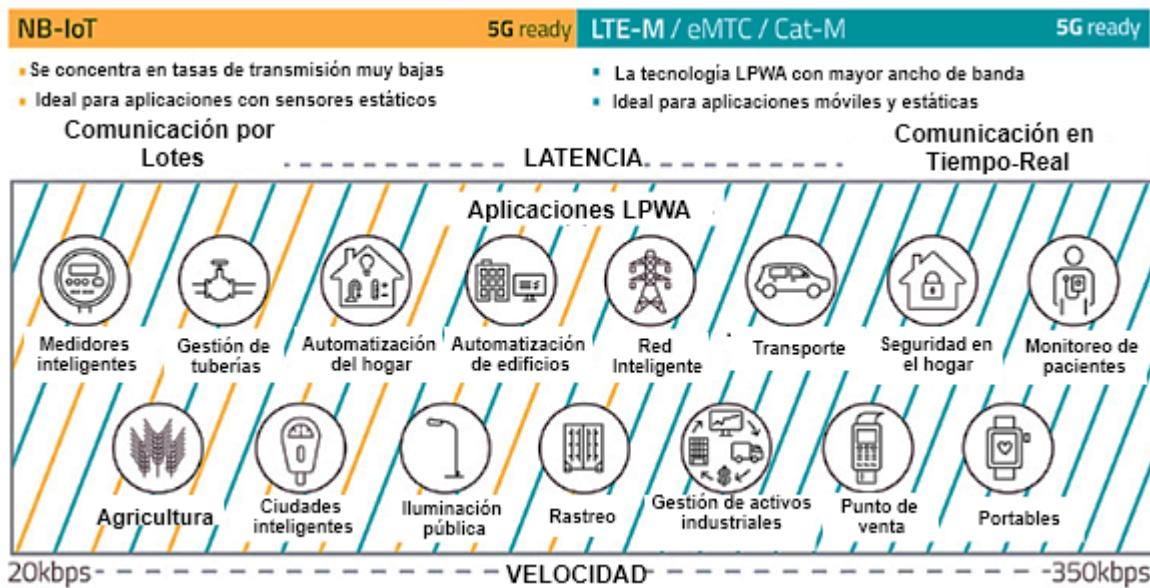


FIGURA 4.2: Tecnologías líder para el caso de uso LPWA, [Fuente: <https://www.iotforall.com/cellular-iot-explained-nb-iot-vs-lte-m/>]

NB-IoT en su banda GSM reajustando parte de su espectro GSM [43].

Despliegue en banda (*in-band*) y en banda de guarda (*guard-band*) LTE.

NB-IoT también está diseñado para su despliegue en las redes LTE existentes, ya sea utilizando uno de los bloques de recursos físicos (PRB) de LTE o utilizando la banda de guarda LTE.

Para el modo de operación independiente y de banda de guarda, el PRB de enlace descendente y ascendente debe establecerse simétricamente y para el modo en banda, el despliegue del PRB estará restringido a algunos prefijos de PRB's de acuerdo al ancho de banda LTE, (ya sea 3, 5, 10, 15 o 20 MHz.) esto debido a la sincronización entre el UE y la eNB [44].

4.4.2. Bandas de frecuencia

Como se muestra en la Figura 4.1, en el futuro se espera que NB-IoT sea implementada en las bandas 5G NR. Por el momento, la 3GPP ha definido un conjunto de bandas de frecuencia para las cuales se pueda usar NB-IoT en bandas LTE (EUTRA).

En la especificación técnica TS-36.101 del *release* 13, la 3GPP proporciona la lista de las bandas compatibles: 1, 2, 3, 5, 8, 12, 13, 17, 18, 19, 20, 26, 28, 66, el *release* 14 agregó las bandas : 11, 25, 31 y 70. Y el *release* 15 agregó más bandas: 4, 14 y 71. La información recibida por los miembros de *Mobile IoT*, hasta ahora, indica que se han utilizado una variedad de estas bandas

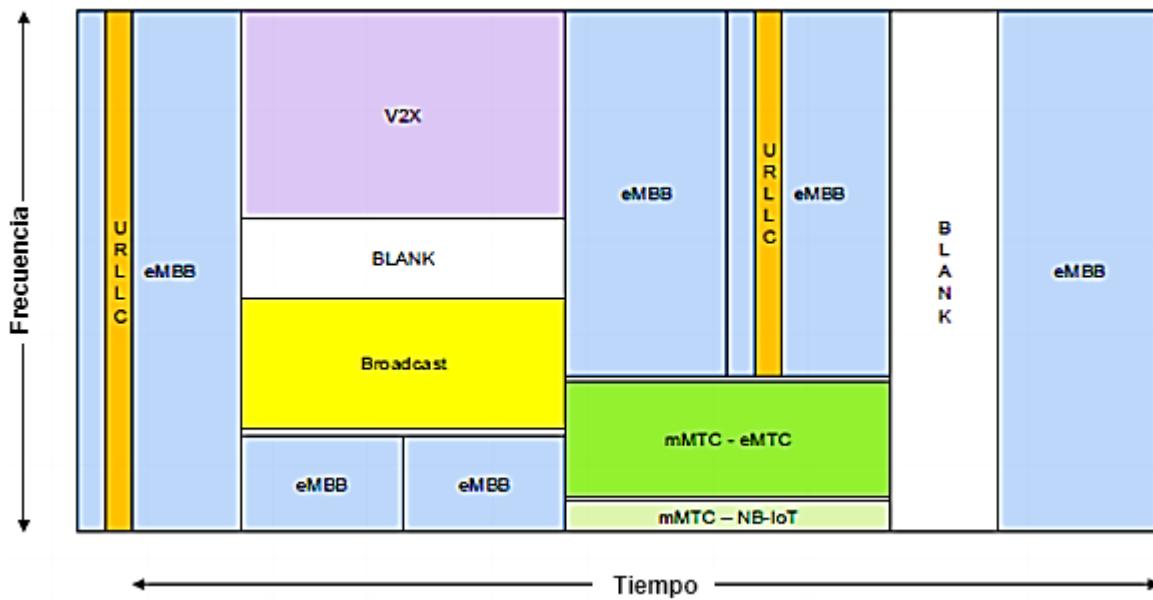


FIGURA 4.3: 5G NR soportará múltiples servicios con distintos requerimiento de QoS

en diferentes regiones[44]. Por ejemplo, en el caso de Latino América las bandas disponibles son las: **2** (1850-1990 MHz), **3** (1710-1880 MHz), **5** (824-894MHz) y **28** (703-803 MHz) [44].

4.4.3. Clases de Potencia

Algunas aplicaciones de IoT son particularmente sensibles al consumo de energía. Para minimizar el impacto de la conectividad en la duración de la batería del dispositivo, en el *release 13*, se determinó que los UE podrán usar dos opciones de clase de potencia. Uno es el nivel de potencia del dispositivo móvil LTE tradicional de **23dBm** (*Power Class 3*) y uno nuevo, con menos potencia de salida, de **20dBm** (*Power Class 5*). El *release 14* de 3GPP agrega una nueva clase de potencia aún menor, de **14dBm** (*PowerClass 6*) [44].

4.4.4. Modos de transmisión en enlace ascendente (UL)

Existen dos modos de transmisión para el enlace UL, en el modo *singletone* sólo se puede asignar una subportadora a cada dispositivo NB-IoT, por el contrario en el modo *multitone*, la agregación de subportadoras es posible, esto con el fin de alcanzar una mayor tasa de transmisión [45].

Solo se admite el modo de *singletone*, cuando el PRB se divide en 48 subportadoras donde el ancho de banda de la subportadora es de 3.75 kHz. En [37] se evaluó el uso *multitone* en anchos de banda de subportadoras de 3.75 kHz.

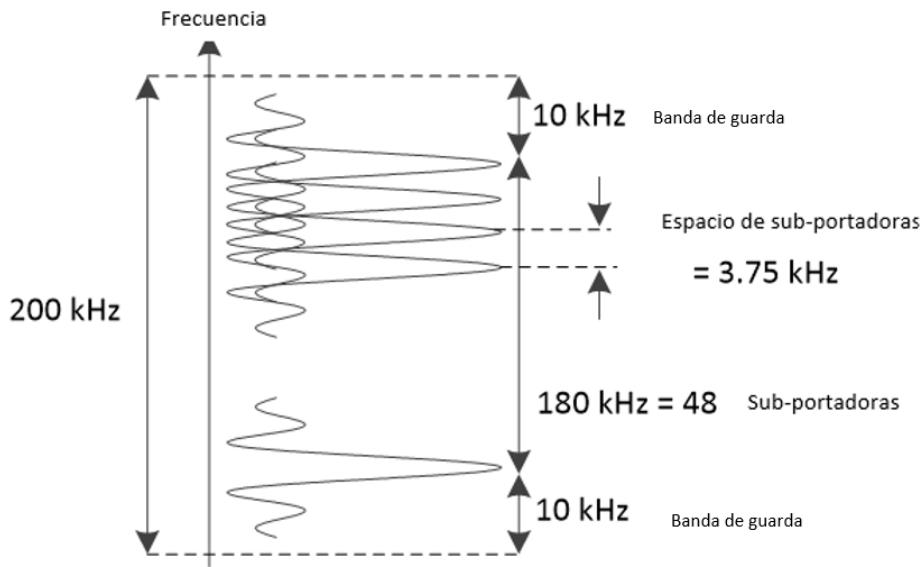


FIGURA 4.4: Estructura de ancho de banda y subportadoras en NB-IoT.

Finalmente, se admiten los modos de *singletone* y *multitone*, cuando el PRB se divide en 12 subportadoras, cada una de ellas con un ancho de banda de 15 kHz. En [36] se estudió más a fondo el uso de *singletone* y *multitone* en transmisiones UL. Una de las reglas que incorporaron fue que en *multitone*, para cada dispositivo IoT únicamente se le pueden asignar 0, 1, 3, 6 o 12 subportadoras.

4.4.5. Características del tráfico IoT

En los Reportes autónomos móviles (MAR, *Mobile Autonomous Reporting*) se definen cuatro tipos de paquetes diferentes:

Informes de excepción

Se espera que muchas aplicaciones de tipo sensor monitorean una condición física y activen un informe de excepción cuando se detecte un evento en específico. Estos eventos serán, en general, raros y ocurrirán cada par de horas, pocos días o incluso meses. Ejemplos de tales paquetes incluyen los producidos por detectores de humo, notificaciones de fallas de energía de medidores inteligentes, notificaciones de manipulación, etc.

Informes periódicos

Se espera que los informes periódicos de enlace ascendente sean comunes para aplicaciones de IoT celular como informes de medición de servicios inteligentes (gas / agua / electricidad),

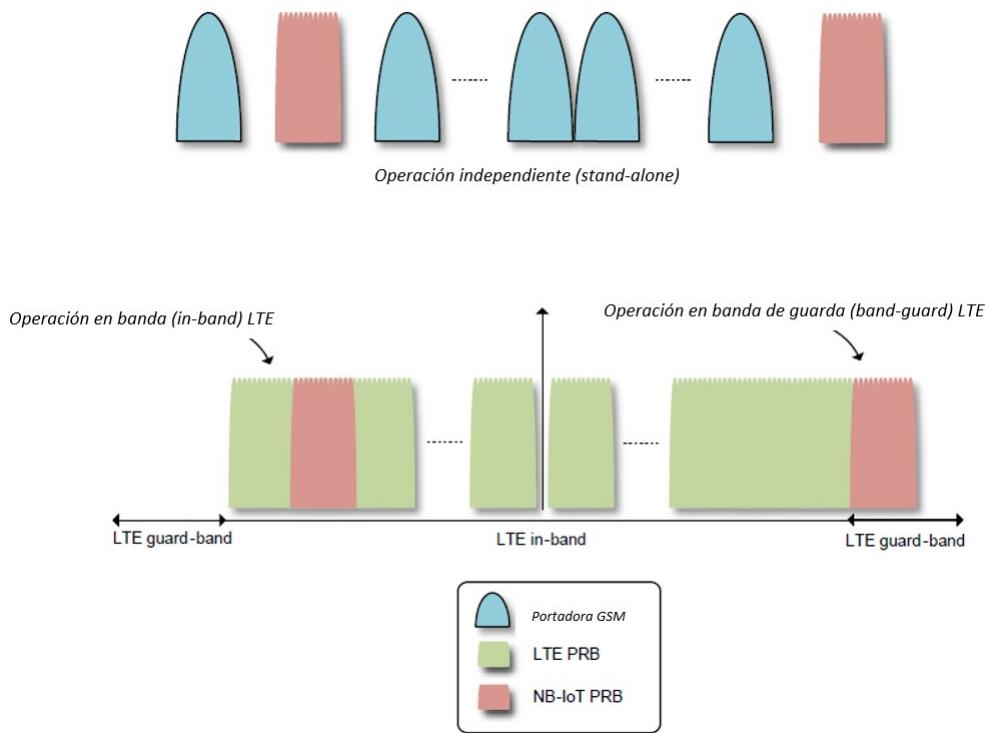


FIGURA 4.5: Modos de Operación en NB-IoT. [43]

agricultura inteligente, entorno inteligente, etc. El modelo de tráfico de informes de enlace ascendente periódico MAR se utiliza en simulaciones a nivel de sistema para análisis de capacidad.

El tamaño de la carga útil sigue una distribución de Pareto con parámetro $\alpha = 2.5$ y tamaño mínimo de carga útil = 20 bytes con un corte en 200 bytes, es decir, las cargas superiores a 200 bytes serán limitadas a 200 bytes.

Una vez revisadas estas clasificaciones de tráfico compatibles para NB-IoT, se asignaron los modelos de tamaño de paquete a los servicios seleccionados en la *Tabla 4.5* de la sección anterior.

La adición de la columna "Tamaños de paquete" para la *Tabla 4.5* se da en la *Tabla 4.7*.

4.4.6. Indicadores clave de rendimiento (KPIs)

Debido a que 5G representará un cambio radical a las generaciones anteriores, se puede prever que habrá nuevos indicadores de evaluación. El diseño de estos indicadores directamente medibles, por un lado, necesita combinar las características de los nuevos servicios, y por otro lado, debe aprender completamente de la experiencia de los KPI clásicos de generaciones anteriores como lo son: el *throughput*, dada una probabilidad de salida y la latencia. Además la densidad de conexión, la densidad de volumen de tráfico y el consumo de energía son KPI's que preocupan

TABLA 4.7: Caracterización del tráfico de paquetes en aplicaciones seleccionadas para la simulación.

<i>Servicio</i>	<i>Tamaño de paquetes</i>
<i>Control de iluminación (Smart City)</i>	Activación aleatoria UL: 20 bytes <i>payload</i> DL: ACK de 0 bytes
<i>Monitoreo del consumo de agua y electricidad en la ciudad (Smart City)</i>	Activación periódica UL: distribución de Pareto con parámetro alfa = 2.5 y tamaño mínimo de carga útil de la aplicación = 20 bytes con un corte a 200 bytes DL: ACK de 0 bytes 50 % de las veces.
<i>Detección de terremotos (Smart Environment)</i>	Activación aleatoria UL: 20 bytes <i>payload</i> DL: ACK de 0 bytes
<i>Monitoreo de contaminación del aire (Smart Environment)</i>	Activación periódica UL: distribución de Pareto con parámetro alfa = 2.5 y tamaño mínimo de carga útil de la aplicación = 20 bytes con un corte a 200 bytes DL: ACK de 0 bytes 50 % de las veces.
<i>Control dinámico de semáforos (Smart Transport and Mobility)</i>	Activación aleatoria UL: distribución de Pareto con parámetro alfa = 2.5 y tamaño mínimo de carga útil de la aplicación = 20 bytes con un corte a 200 bytes DL: ACK de 0 bytes 50 % de las veces.
<i>Otros dispositivos mMTC</i>	Activación aleatoria UL: 20 bytes <i>payload</i> DL: ACK de 0 bytes

también a las redes 5G/IoT [8].

Para cumplir con el conjunto de requisitos de mMTC, NB-IoT debe admitir principalmente cuatro indicadores clave de rendimiento (KPI):

1. Vida útil de la batería del dispositivo más allá de 10 años, suponiendo una capacidad de energía almacenada de 5 Wh.
2. Densidad de conexión masiva de hasta 1M dispositivos por km cuadrado en un entorno urbano.
3. Latencia de como máximo 10 s.
4. Una tasa máxima alcanzable de hasta 200kbps (subida).

El simulador obtuvo como resultados la tasa de datos alcanzada por el sistema y el porcentaje de dispositivos que transmitieron sus paquetes a la tasa requerida.

4.5. ANÁLISIS DE MODELOS PARA LA EVALUACIÓN DE REDES 5G/IoT

Los modelos necesarios para caracterizar el comportamiento de redes móviles, desde el punto de vista de simulaciones a nivel de sistema, se presentan a continuación [8]:

1. Modelo de despliegue de BSs y UEs.
2. Modelo de antenas (MIMO, MISO, entre otras) y formación de haz.
3. Modulación y codificación.
4. Modelo de canal.
5. Patrones de Movilidad.
6. Calendarizadores (planificadores de recursos).
7. Esquema de acceso múltiple al medio.
8. Modelos de tráfico.

Los modelos que se incluyeron en el simulador son los siguientes:

4.5.1. MODELO DE DESPLIEGUE DE BSs Y UEs

Los modelos del posicionamiento de las estaciones base y los nodos IoT utilizan diferentes estrategias de despliegue (como se puede ver en la *Figura 2.1*). Este aspecto como los demás es de mayor o menor importancia dependiendo de los objetivos de la simulación, es decir, para un alcance comercial resulta importante simular el despliegue determinístico de los actores de la red, por otro lado para un alcance con fines de análisis en el diseño y dimensionamiento de estas redes resulta más adecuado un despliegue aleatorio.

Diversos autores coinciden en que varias distribuciones de redes móviles siguen un proceso estocástico [46][35]. La geometría estocástica es una rama de la probabilidad con muchas aplicaciones que permiten el estudio de fenómenos aleatorios en el plano o en dimensiones superiores [10]. Recientemente se ha utilizado con éxito para modelar la distribución espacial de células pequeñas como las femtoceldas [14].

En este proyecto con fines de diseño y análisis, se模拟aron ambos despliegues, uno uniforme y otro siguiendo una geometría estocástica, es decir, un PPP.

4.5.2. MODELO DE CANAL

Un buen diseño de sistema necesitó de conocer las características del canal de propagación a través de las frecuencias de microondas y ondas milimétricas.

Los modelos de canal son necesarios para simular la propagación de una manera reproducible y rentable, y se utilizan para diseñar y comparar con precisión las interfaces de radio y el despliegue del sistema. Los parámetros comunes del modelo de canal inalámbrico incluyen frecuencia de portadora, ancho de banda, distancia entre el transmisor (Tx) y el receptor (Rx), los efectos ambientales y otros requisitos necesarios. El desafío definitivo para un modelo de canal 5G es proporcionar una base física fundamental, a la vez flexible y precisa, especialmente en un amplio rango de frecuencias como 0.5–100 GHz [47]. Los modelos de canal investigados se dividen principalmente de acuerdo al escenario en el que se están diseñando, ya sea *Urban Macro (UMa)* o *Urban Micro (UMi)*, además de la condición del ambiente si es que hay línea de vista (*LoS*) entre el UE y la BS.

Existe una gama amplia de modelos de canal propuestos para redes 5G, (p.ej. 3GPP, WINNER I/II, QuaDRiga/ mmMagic, 5GCM, METIS, MiWEBA, IEEE [8]). Aunque existen diversos modelos, los modelos de canal 3GPP y WINNER II son los más conocidos y empleados en la industria de comunicaciones móviles [48], conteniendo una gran diversidad de escenarios de despliegue como lo son *UMi*, *UMa*, *indoor office (InH)*, etc. Además proveen parámetros clave del canal incluyendo probabilidades de línea de vista (*LoS*), modelos de pérdida por trayectoria, retardos y niveles de potencia por trayectoria [48].

El modelo implementado se trató de uno teórico y estocástico en vez de uno empírico, ya que este proyecto está enfocado en el teletráfico. Los modelos empíricos suelen ser más sofisticados y piden una gran cantidad de parámetros de entrada. Por lo tanto se eligieron modelos estocásticos que se adaptaran al rango de frecuencia de transmisión y a los ambientes urbanos que se propusieron.

En [48], los autores evaluaron tres diferentes modelos de propagación estocásticos de perdida por trayectoria a larga escala para ser implementados a través de la banda de frecuencias de microondas y ondas milimétricas. ABG, CI y CIF son modelos estadísticos de propagación para multi-frecuencias (estocásticos) que describen los parámetros de larga escala con pérdida de trayectoria de acuerdo a la distancia.

Los modelos evaluados fueron:

1. ABG: Modelo Alpha-Beta-Gamma.
2. CI: Modelo de pérdida por trayectoria de distancia de referencia de espacio libre cercano.
3. CIF: Modelo CI con un exponente de pérdida de trayectoria ponderado en frecuencia.

Para el primero, la ecuación del modelo ABG está dada por:

$$L_p^{ABG}(f, d)_{[dB]} = 10\alpha \log_{10} \left(\frac{d}{1m} \right) + \beta + 10\gamma \log_{10} \left(\frac{f}{1GHz} \right) + x_{\sigma}^{ABG}, \text{ donde } d \geq 1m \quad (4.1)$$

$\alpha \rightarrow$ coeficiente que representa la dependencia de la perdida por trayectoria con la distancia

$\gamma \rightarrow$ coeficiente que representa la dependencia de la perdida por trayectoria con la frecuencia

$\beta \rightarrow$ es un valor de compensación para la pérdida por trayectoria (en dB's)

$x_{\sigma}^{ABG} \rightarrow$ es una variable aleatoria gaussiana de media cero con una desviación estándar sigma [dB], que describe las fluctuaciones de señal a gran escala (es decir, multirayectoria)
desvanecimiento tipo Rayleigh

Para el segundo, la ecuación del modelo CI está dada por:

$$L_p^{CI}(f, d)_{[dB]} = 32,4 + 10 n \log_{10} \left(\frac{d}{d_0} \right) + 20 \log_{10} (d_0) + 20 \log_{10} (f) + x_{\sigma}^{CI}, \text{ donde } d \geq d_0 \quad (4.2)$$

$x_{\sigma}^{CI} \rightarrow$ es una variable aleatoria gaussiana de media cero con una desviación estándar sigma [dB], que describe las fluctuaciones de señal a gran escala (es decir, multirayectoria)
desvanecimiento tipo Rayleigh

Para el tercero, la ecuación del modelo CIF está dada por:

$$L_p^{CIF}(f, d)_{[dB]} = 32,4 + 10 n \left(1 + b \left(\frac{f - f_0}{f_0} \right) \right) \log_{10} (d) + 20 \log_{10} (f) + x_{\sigma}^{CIF}, \text{ donde } d \geq 1m \quad (4.3)$$

$x_{\sigma}^{CIF} \rightarrow$ es una variable aleatoria gaussiana de media cero con una desviación estándar sigma [dB], que describe las fluctuaciones de señal a gran escala (es decir, multirayectoria)
desvanecimiento tipo Rayleigh

Cada uno de estos modelos han sido recientemente estudiados por organizaciones de estandarización como 3GPP y son propuestos para su uso en el diseño de sistemas inalámbricos de comunicación de 5G enfocados en escenarios UMa, UMi, InH, y SM.

De acuerdo al análisis de sensibilidad en [48], se demostró que el modelo CI es el más adecuado para entornos al aire libre debido a su precisión, simplicidad y rendimiento de sensibilidad, dado que la pérdida de trayectoria medida depende poco de la frecuencia en ambientes exteriores más allá del primer metro de propagación de espacio libre.

Por otro lado, el modelo CIF es muy adecuado para entornos interiores, ya que proporciona una desviación estándar más pequeña que el modelo ABG en muchos casos, incluso con menos parámetros del modelo y tiene una precisión superior cuando se analiza con el análisis de sensibilidad.

Los modelos CI y CIF son más robustos y precisos en comparación con el modelo ABG, por lo que es confiable la aplicación del modelo CI para simular entornos en exteriores y el CIF para interiores [48].

De acuerdo a lo propuesto en la [sección 4.3](#), el ambiente urbano que consideraremos está dirigido a un entorno en exteriores. Por lo que se seleccionó al modelo CI (Modelo de pérdida por trayectoria de distancia de referencia de espacio libre cercano).

Los parámetros que requiere este modelo (*Ecuación 4.2*) son: la distancia entre la BS y el UE, la frecuencia fundamental de operación y una variable aleatoria de media cero con una desviación estándar σ [dB], que describirá las fluctuaciones de señal a gran escala (es decir, la multirayectoria [*multipath*]).

Por último, los autores proponen a $d_0 = 1m$ en los modelos de pérdida por trayectoria para sistemas 5G ya que se espera que las distancias de cobertura serán más cortas a frecuencias más altas. Además, lo que se espera son futuras celdas pequeñas, es probable que las BS se monten más cerca de las obstrucciones. La estandarización a una distancia de referencia de 1 m simplifica las comparaciones de mediciones y modelos y proporciona una definición estándar para el PLE, al tiempo que permite la intuición y el cálculo rápido de la pérdida de trayectoria.

4.5.3. ESQUEMA DE ACCESO MÚLTIPLE AL MEDIO

Acceso Múltiple No Ortogonal (NOMA)

Se estudió en la Sección [2.3.2](#), el uso de NOMA y su soporte eficientemente de conectividad masiva. El diseño de NOMA en transmisiones de subida (*uplink*) ha sido propuesto en [49] y el diseño óptimo de NOMA en transmisiones de bajada (*downlink*) ha sido propuesto en [50].

En [35] se implementó NOMA emparejando selectivamente dos usuarios, es decir, se escogía a un usuario con una condición de canal muy buena (cerca de la BS) y otro con una condición de canal muy pobre (en el borde de la celda). Por otro lado en [37], se implementó NOMA usando la técnica de agrupamiento de usuarios, considerando un entorno donde conviven dispositivos mMTC y URLLC (estos tienen mayores requisitos de tasa en comparación con los dispositivos mMTC), de igual manera se agrupan con otros usuarios (p.ej., 2, 3 o 4 usuarios) de diferente tipo (mMTC y URLLC) y se ordenan convenientemente para implementar SIC.

NB-IoT no es capaz de proveer conectividad a una cantidad masiva de dispositivos IoT como se espera en el futuro, así para el diseño se consideró la metodología en [37], donde los dispositivos activos de URLLC y mMTC comparten un PRB para la transmisión de datos en el enlace ascendente. El ancho de banda disponible se divide en un conjunto de frecuencias (subcanales)

S. De hecho, el ancho de banda del sistema se puede dividir por igual en 48 o 12 subportadoras en los sistemas NB-IoT. La relación de la distribución de dispositivos mMTC con los URLLC se fijó a 3 a 1. Todo esto con base en las consideraciones del modelo de sistema en [37].

En [37], se tiene en cuenta lo siguiente:

- U: Lista de dispositivos uRLLC
- M: Lista de dispositivos mMTC
- S: Lista de Subportadoras s
- C: Lista de Grupos NOMA
- k_{max} : Rango máximo de usuarios en un grupo NOMA
- R_m^{th} : Tasa objetivo del enésimo dispositivo m mMTC
- R_u^{th} : Tasa objetivo del enésimo dispositivo u uRLLC
- P_m^{max} : Potencia máxima del enésimo dispositivo m mMTC
- P_u^{max} : Potencia máxima del enésimo dispositivo u uRLLC (i.e. 23dBm)
- P_m^s : Potencia del enésimo dispositivo m mMTC
- P_u^s : Potencia del enésimo dispositivo u uRLLC
- h_m^s : Ganancia de canal del enésimo dispositivo m mMTC sobre la portadora s
- h_u^s : Ganancia de canal del enésimo dispositivo u uRLLC sobre la portadora s
- \hat{S} : Lista de subportadoras asignadas
- S_a^c : Lista de subportadoras asignadas al enésimo cluster
- C_{ns} : Lista de cluster aún no asignados
- W : Ancho de banda en un tono NB-IoT (3.75KHz)
- α, β son variables binarias que indican asignaciones.

La tasa de datos alcanzable por un dispositivo m (mMTC) en términos de la tasa agregada sobre las subportadoras asignadas se puede expresar como [37]:

$$R_m = \sum_{c \in \mathcal{C}} \sum_{k \in \mathcal{K}} \alpha_m^{c,k} \sum_{s \in \mathcal{S}} \gamma^{s,c} W \times \log_2 \left(1 + \frac{|h_m^s|^2 p_m^s}{N_0 W + \sum_{d \in \mathcal{M} \setminus m} \sum_{h=k+1}^{k_{\max}} \alpha_d^{c,h} |h_d^s|^2 p_d^s} \right) \quad (4.4)$$

Del mismo modo, la Tasa de datos alcanzable por un dispositivo u (URLLC) puede determinarse mediante el teorema de Shannon-Hartley. Hay que tomar en cuenta que los rangos de URLLC siempre son mayores que los de mMTC en cada clúster NOMA. Por lo tanto, reciben interferencia de todos los miembros del clúster mMTC, así como de los miembros del clúster URLLC con rangos más altos. Por lo tanto, la tasa de datos alcanzable de un dispositivo u URLLC sobre las subportadoras asignadas es [37]:

$$R_u = \sum_{c \in \mathcal{C}} \sum_{k \in \mathcal{K}} \beta_u^{c,k} \sum_{s \in \mathcal{S}} \gamma^{s,c} W \times \log_2 \left(1 + \frac{|h_u^s|^2 p_u^s}{N_0 W + \sum_{d \in \mathcal{U} \setminus u} \sum_{h=k+1}^{k_{\max}} \beta_d^{c,h} |h_d^s|^2 p_d^s + \sum_{m \in \mathcal{M}} \sum_{h=k+1}^{k_{\max}} \alpha_d^{c,h} |h_m^s|^2 p_m^s} \right) \quad (4.5)$$

4.5.4. MODELOS DE TRÁFICO

Los modelos de tráfico en comunicaciones móviles buscan acercarse, lo más posible a cómo transmiten datos o realizan peticiones de acceso los dispositivos que intentan modelar. Estos modelos de tráfico pueden clasificarse en modelos de tráfico **agregado** y modelos de tráfico **fuente** [51]. El tráfico agregado simula un flujo de tráfico que se agrupa para recibir un tratamiento común, mientras que en los modelos de tráfico fuente es justamente cada una de las fuentes generadoras del tráfico la que se simula y frecuentemente se hace acompañada de una cadena de Markov que intenta representar los distintos estados del dispositivo fuente y la probabilidad de transición entre ellos.

Sin importar el modelo de tráfico a utilizar, en [51] se señala que los modelos de tráfico que pretendan simular el comportamiento de dispositivos de MTC (*Machine Type Communications*) deben:

- Capturar con precisión el comportamiento de un solo dispositivo de MTC
- Permitir la simulación concurrente de una cantidad masiva de dispositivos con su potencial reacción síncrona a un evento.

Antes de elegir un modelo es importante conocer las propiedades del tráfico máquina a máquina (M2M, *Machine to Machine*), el cual se considera una forma de transmisión de datos que no requiere necesariamente de la interacción humana (ETSI, 2010) y corresponde justamente al tráfico de los nodos IoT, de [51] tenemos:

- Cantidad masiva de dispositivos
- Pocos paquetes de un tamaño pequeño a ser transmitidos por dispositivo
- Periodos largos entre dos transmisiones consecutivas
- Tráfico de subida (*uplink*) dominante

- Transmisiones en tiempo real y transmisiones tolerantes al retraso
- Paquetes no sincronizados y paquetes sincronizados
- Activación de tráfico que depende del espacio y tiempo

Además, se hace la distinción de 3 patrones de tráfico que pueden presentarse en estos dispositivos:

1. Actualización periódica (PU, Periodic Update): Este tipo de tráfico ocurre cuando el dispositivo transmite reportes de estado y/o actualizaciones de estado de manera periódica. Puede verse como una activación por evento que ocurre por el mismo dispositivo en un intervalo periódico. Típicamente, el tráfico PU no necesita transmitirse en tiempo real y cuenta además de un patrón periódico de tiempo con un tamaño constante en sus paquetes. Un ejemplo típico de estos dispositivos son medidores inteligentes (por ejemplo gas, electricidad, agua).
2. Activación por evento (ED, Event-Driven): En caso de que un evento desencadene la transmisión de datos de un dispositivo, el patrón de tráfico corresponde a esta segunda clase. Un evento puede ser causado ya sea por la medición de un parámetro que sobrepasó un límite y activó alguna alarma o bien por el nodo que actúa como servidor y envía comandos al dispositivo. El tráfico *Event-Driven* puede requerir ser transmitido tanto en tiempo real o no, un ejemplo de mensajes de subida que debieran ser transmitidos en tiempo real son alarmas y notificaciones médicas de emergencia, en cuanto a los mensajes de bajada, estos podrían ser la distribución de mensajes de emergencia locales, por ejemplo en caso de sismo o tsunamis. En algunos casos, como ya se mencionó, este tráfico no necesita ser transmitido en tiempo real. Por ejemplo, cuando un dispositivo IoT envía una actualización de su ubicación al servidor o se reciba una actualización de *firmware* desde este.
3. Intercambio de carga útil (PE, Payload Exchange): este último tipo de tráfico ocurre después de una transmisión previa (PU o ED). Comprende todos los casos en los que es necesario un mayor intercambio de datos entre el dispositivo que envía y su servidor, este tráfico se espera sea predominantemente de subida y puede ser de tamaño constante o variable según la aplicación.

Las aplicaciones en el mundo real que brindan servicio a dispositivos de IoT generarán casi siempre tráfico en una combinación de estos tipos, más un estado de reposo o de ahorro de batería.

Ahora se presentan los modelos de tráfico más recurrentes para la simulación de comunicaciones M2M.

Modelos de tráfico agregado

Han sido propuestos por la 3GPP al reconocer la importancia de caracterizar el tráfico M2M. Se trata en realidad de 2 modelos de tráfico agregado generado por una gran cantidad de usuarios,

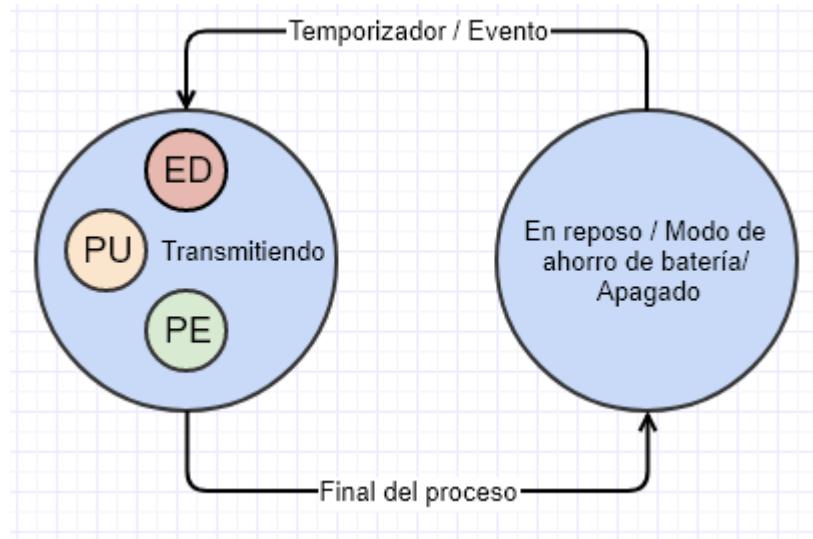


FIGURA 4.6: Estructura de los estados principales del tráfico M2M

TABLA 4.8: Modelos de tráfico agregado propuestos por la 3GPP para comunicaciones M2M

Sincronizado/Coordinado/Correlacionado (En un intervalo limitado en el tiempo)	No sincronizado/No coordinado/ No correlacionado (En un intervalo limitado en el tiempo)
Distribución de probabilidad de arribo de paquetes/peticiones $f(t)$ en $[0,1]$: Beta (3,4) Número de dispositivos: 1 000, 3 000, 5 000, 10 000, 30 000. Periodo T : 10 s	Distribución de probabilidad de arribo de paquetes/peticiones $f(t)$ en $[0,1]$: Uniforme Número de dispositivos: 1 000, 3 000, 5 000, 10 000, 30 000. Periodo T : 60 s

el primero modela el tráfico generado de forma aleatoria y el segundo modela tráfico síncrono en el tiempo, esto se puede observar en la *Tabla 4.8*.

- *Modelo 1* - Modelo de tráfico agregado sin correlación 3GPP: Genera tráfico sin correlación en un intervalo específico de tiempo. Lo que significa que no se tomarían en cuenta la correlación entre los dispositivos IoT. Utiliza una distribución uniforme para modelar el tráfico agregado en un intervalo de tiempo específico.
- *Modelo 2* - Modelo de tráfico agregado con correlación 3GPP: Este modelo genera tráfico correlacionado en un intervalo de tiempo, asumiendo que todas las máquinas se encuentran sincronizadas. Utiliza una distribución beta para modelar el tráfico agregado en un intervalo de tiempo específico.

La principal ventaja de los modelos de tráfico agregado es su fácil implementación (en términos de una baja complejidad computacional) cuando se simulan una gran cantidad de dispositivos.

Por otro lado, como se menciona en [7], la precisión de estos modelos al reflejar el comportamiento real del sistema es su principal desventaja.

Modelos de tráfico fuente

Los modelos de tráfico fuente, modelan justamente el tráfico que genera cada uno de los dispositivos. Este tipo de modelado es más preciso que el de tráfico agregado ya que modela el comportamiento de cada nodo, sin embargo, puede fácilmente volverse muy complejo cuando se agrega una gran cantidad de dispositivos (fuentes) y comportamientos. A continuación se presentan y analizan dos modelos de tráfico fuente.

- *Modelo 3: Modelo de fuente de Semi-Markov (Semi-Markov Models, SMM)*

En este modelo de fuente cada dispositivo se modela utilizando una cadena de Markov en la que se define la probabilidad de transición entre estados. Los estados que se encontrarán casi siempre modelados son los mencionados anteriormente: el de actualización periódica (PU), el de activación por evento (ED) y el de intercambio de carga útil. La *Figura 4.7* muestra cómo se verían modelados los estados de un dispositivo en una cadena de Markov.

La probabilidad de transición entre el mismo estado es 0, además los tiempos de espera y la longitud de los mensajes son generados de acuerdo a una distribución de probabilidad que es independiente de cada estado y potencialmente distinta para cada uno de ellos [7].

La principal ventaja del modelo de tráfico fuente SMM es que permitiría una descripción más detallada del comportamiento de los dispositivos IoT de manera individual, sin embargo no es capaz de capturar la relación que existe entre dos dispositivos cercanos que pudieran tener una cierta sincronía. Otra desventaja es que la complejidad del sistema aumenta considerablemente entre más dispositivos se simulan a diferencia de los modelos de tráfico agregado.

- *Modelo 4: Modelo de fuente de Procesos de Poisson emparejados Markov-modulados (CMMPP, Coupled Markov Modulated Poisson Process)*

En el modelo de tráfico CMMPP cada dispositivo MTC es representado como una entidad por separado y a diferencia del modelo SMM sí puede representarse una sincronización espacial y temporal entre dispositivos similares. La clave en el diseño del modelo CMMPP se presenta en encontrar un balance entre el emparejamiento entre distintos dispositivos y una complejidad tolerable del sistema cuando se tiene una gran cantidad de dispositivos [33].

Los procesos de Poisson Modulados con Markov (*Markov modulated Poisson processes, MMPP*) consisten en procesos de Poisson que son modulados por la tasa $\lambda_{i[t]}$, que viene determinada por el estado de una cadena de Markov $s_n[t]$, este principio se ve presentado en la *Figura 4.8* donde $p_{i,j}$ son las probabilidades de transición entre los estados de la cadena. En este modelo cada dispositivo n del total N se encuentra representado por una cadena de Markov y un correspondiente proceso de Poisson. Debido a que existe

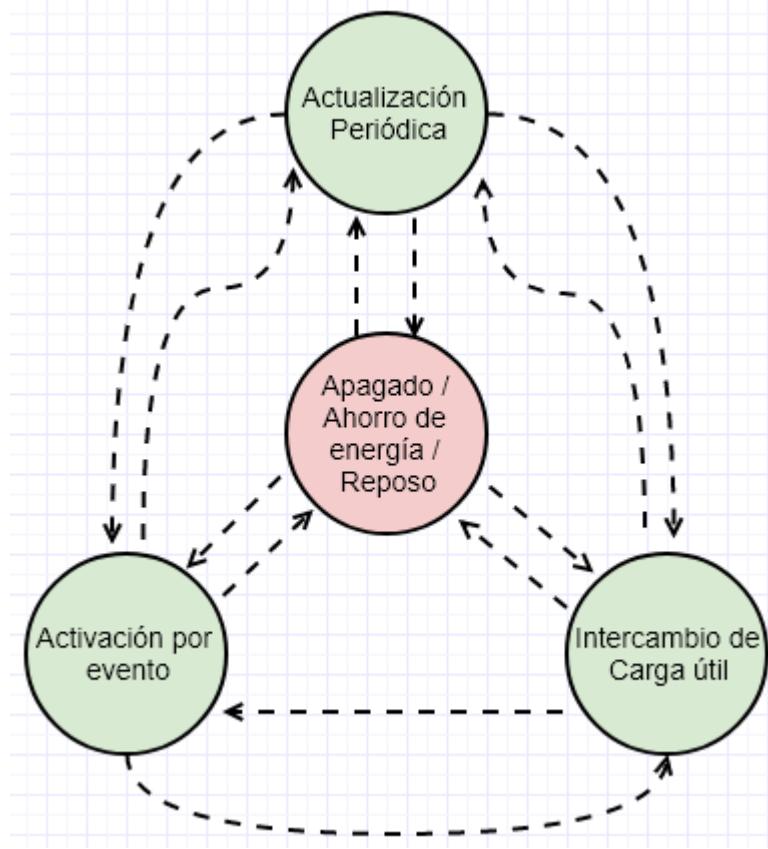


FIGURA 4.7: Cadena de Markov del modelo SMM

una alta correlación en el cambio de estados de distintos dispositivos, tanto en el espacio como en el tiempo, es necesario realizar un emparejamiento. En los modelos genéricos, el emparejamiento se realiza introduciendo enlaces bidireccionales entre los dispositivos, pero esto sería sin lugar a dudas muy complejo de simular, de manera que en [33] se propone un proceso de fondo actuando como *maestro* el cual modula todos los dispositivos del mismo tipo.

La principal ventaja como ya se mencionó del tráfico fuente frente al tráfico agregado es su precisión, por otra parte, la del tráfico agregado es su fácil implementación para un gran número de dispositivos. El modelo CMMPP es un intermedio entre estos dos casos, es decir mantiene la precisión del modelado de tráfico fuente mientras se mantiene la viabilidad para un gran número de máquinas. A continuación se presenta la *Tabla 4.9* con una comparativa entre los distintos modelos mencionados.

Como puede observarse en la *Tabla 4.9*, las ventajas que trae consigo la utilización de un modelo de tráfico como el CMMPP son bastante convenientes generar tráfico de dispositivos mIoT, pues este es capaz de simular la relación espacial y temporal que existiría entre los nodos. Si se regresa a la *Tabla 4.5* en la que se presentan los servicios que se simularán, se encuentra que

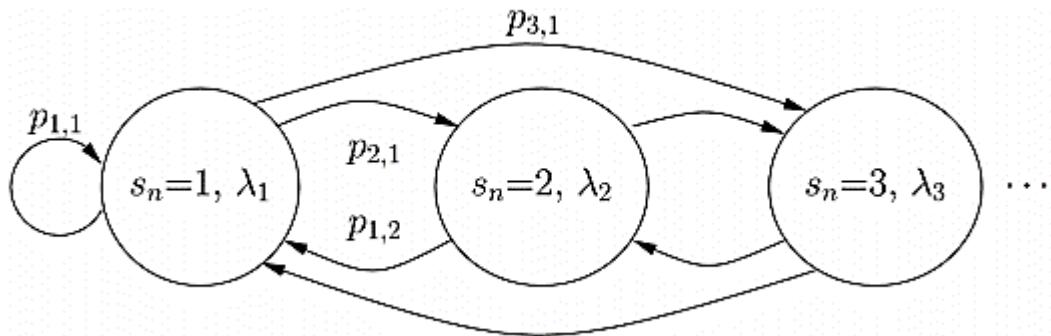


FIGURA 4.8: Modelo CMMPP: cada dispositivo MTC n está representado por una cadena de Markov con estados s_n , que establecen el parámetro λ . Este parámetro es el promedio de la tasa de arribos, el cual modela el respectivo proceso de Poisson

TABLA 4.9: Comparativa entre los modelos de tráfico MTC abordados

Métrica	Agregado	SMM	CMMPP
<i>Modelado de los dispositivos</i>		✓	✓
<i>Modelado de dispositivos coordinados</i>	✓		✓
<i>Coordinación espacial y temporal</i>			✓
<i>Modelado de los paquetes</i>	✓		
<i>Modelado de la tasa de arribo</i>	✓	✓	✓
<i>Tiempo de ejecución aleatorio factible</i>		✓	✓
<i>Ubicación del dispositivo</i>		✓	✓
<i>Emparejamiento de estados</i>			✓
<i>Complejidad (N número de dispositivos)</i>	O(1)	O(N)	O(N)

servicios como el monitoreo de la condición del aire en la ciudad, la detección de terremotos, la manipulación de la iluminación y demás tendrá un comportamiento similar en un espacio confinado y para poder hacer hincapié en la importancia del diseño de la red para a la hora de cumplir con la QoS esperada, es necesario que se implemente la posibilidad de recibir una gran cantidad de dispositivos en el caso de uso mMTC, los cuales estarían tratando de acceder a los recursos del sistema simultáneamente. De manera que el modelo CMMPP se seleccionó para la simulación de este sistema, complementándolo con un modelo determinístico para las aplicaciones que sólo producen tráfico periódico.

Una explicación resumida del modelado de tráfico fuente CMMPP se realiza a continuación [33]:

1. Un conjunto de k estados se definen, cada uno asociado con una tasa de generación de paquetes λ_k . Un dispositivo IoT se encuentra en todo momento en algún de estos estados representados en una cadena de Markov formada por los estados antes mencionados.

2. La transición entre los k estados para cualquier n -ésimo dispositivo está definida por una matriz $k \times k$ de P_n la cual es a su vez una función de dos matrices de transición P_u y P_c (Para comportamiento no coordinado/no sincronizado y comportamiento coordinado/sincronizado respectivamente) definidas por la red de N nodos.
3. Un factor de correlación espacial δ_n se asigna a cada dispositivo n . Esto modela qué tanto se involucra un dispositivo durante la generación de tráfico coordinado en la red y dicta efectivamente la contribución de P_c en la matriz resultante de probabilidad de transición de ese dispositivo.
4. Se define un proceso $\Theta(t)$ el cual controla la matriz de transición instantánea de estado del n -ésimo dispositivo en el instante t .

Capítulo 5

Diseño

En este capítulo se muestra el diseño del simulador programado. El objetivo de esta sección fue crear un modelo de sistema, para el cual se especificó un escenario y el modelado de cuatro principales componentes que acabaron siendo implementados en el simulador: el despliegue de UEs, el modelo de canal, el esquema de acceso al múltiple al medio y los modelos de tráfico.

5.1. MODELO DE SISTEMA

En principio, se comenzó con la etapa de análisis (*Capítulo 4*), donde se analizaron los requisitos de la red, las características que debía tener el escenario que se propuso y los distintos modelos que formaron parte de la simulación.

En términos generales, se consideró una red celular uni-celda para transmisiones de subida (*uplink*), el escenario se trata de uno urbano macro celular (UMa) en donde frecuentemente los UEs se sitúan estáticos y en el exterior. De acuerdo a los modelos analizados y ya elegidos en la *Sección 4.5*, el modelo de sistema del simulador implementó los siguientes sub-sistemas [*Figura 5.1*]:

1. Uso de una geometría estocástica, es decir, despliegue de UEs siguiendo un PPP.
2. Pérdidas de canal usando un modelo CI para ambientes exteriores.
3. Incorporación de un esquema de acceso al medio no ortogonal (NOMA) usando una técnica de agrupamiento (*clustering*) de usuarios.
4. Diferentes modelos de tráfico que simulen distintos servicios para NB-IoT.

El simulador se enfocó en el caso de uso de mMTC (o mIoT) el cual se caracteriza por brindar servicio a un gran número de dispositivos, esto es, teniendo una alta densidad de volumen de tráfico en escenarios con aglomeración de dispositivos. Por lo tanto, como se revisó en el capítulo anterior un excelente candidato para cumplir con los requerimientos del caso de uso mIoT y que forma ahora parte del estándar 5G, fue la tecnología NB-IoT. De este estándar, se tomaron sus especificaciones técnicas (*Sección 4.4*), tales como los parámetros fundamentales para la comunicación entre la BS y los UEs en la simulación.

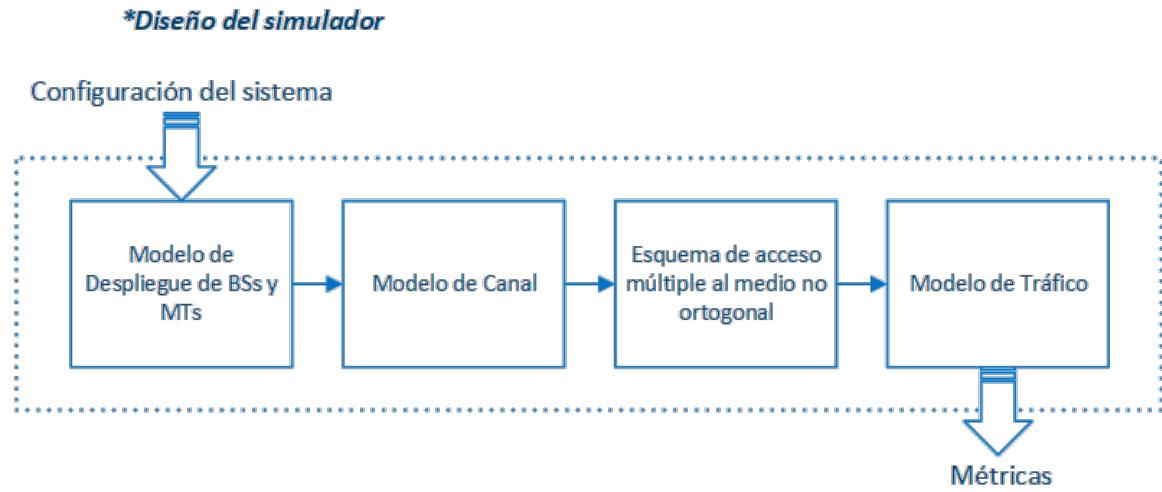


FIGURA 5.1: Diagrama general de bloques del simulador, constando principalmente de 4 módulos.

A continuación se definen las características generales de cada sub-sistema:

5.1.1. Uso de una geometría estocástica, es decir, despliegue de UEs siguiendo un PPP

De igual manera que se realizó en [46] y [35] con el fin de obtener un análisis fundamental y más realista, se propuso la generación de una geometría estocástica en 2D para la distribución de los UEs. Se tomó como modelo de despliegue de UE un *Homogeneous Poisson Point Process* (proceso puntual de Poisson homogéneo, HPPP) con distintas densidades para los diferentes tipos de dispositivos que se implementaron.

5.1.2. Pérdidas de canal usando un modelo CI para ambientes exteriores

En la *Sección 4.3* se establecieron las aplicaciones de los diferentes dominios de IoT que se utilizaron en el simulador. Todas estas aplicaciones se encuentran presentes, no exclusivamente pero sí particularmente, en escenarios urbanos y en exteriores, por lo tanto se implementó el Modelo CI (ecuación 4.2) el cual de acuerdo a lo estudiado en [48] es el modelo que mejor estima la señal en este tipo de ambientes. Además, se agregarán pérdidas por el desvanecimiento rápido, usando el modelo de desvanecimiento de Rayleigh (siguiendo una distribución Rayleigh con desviación estándar unitaria y que se distribuyó de forma independiente e idéntica [i.i.d.]), este es ideal para entornos urbanos en situaciones donde hay un gran número de multi-trayectorias de la señal y reflexiones causadas por los edificios y objetos que obstruyen la línea de vista.

TABLA 5.1: Parámetros Modelo CI [Fuente: [48]]

Escenario	Ambiente	Rango de Frecuen- cias	# Puntos de Datos analizados	Rango de distan- cias entre BS y UE	Modelo Canal	PLE
UMa	<i>LoS</i>	2 - 38 GHz	1032	60 - 930 m	<i>CI</i>	2.0
	<i>NLoS</i>	2 - 38 GHz	1869	61 - 1238 m	<i>CI</i>	2.9

Con base en [48], para el Modelo CI en ambientes urban macro se tienen los parámetros mostrados en la Tabla 5.1 que dependiendo el tipo de ambiente (ya sea con línea de vista [LoS] o sin línea de vista [NLoS]) los parámetros rango de distancia y el exponente de pérdida (PLE) varían. El modelo de sistema propuesto se trata de un ambiente en exteriores, se considera que sí habrá línea de vista entre los UE y la BS, por lo tanto, se utilizó LoS con distancias entre [60 - 900 m] y un PLE igual a 2.

Algunos parámetros siguieron las características de las mediciones que se hicieron para validar el modelo CI, en [48]. Las mediciones se realizaron en Vestby, Aalborg, Dinamarca, en las bandas de frecuencia de 2, 10, 18 y 28 GHz en marzo. Vestby representa una típica ciudad europea de tamaño mediano con construcciones y anchos de calle regulares, que son de aproximadamente 17 m (cinco pisos) y 20 m, respectivamente. Para el escenario UMa, la antena BS está por encima de la altura de la azotea, típicamente 25 m más o menos por encima del suelo [48].

Por último, para la definición del rango de frecuencia de las transmisiones, como se pudo leer en la Sección 4.4, en la actualidad el despliegue de redes NB-IoT se ha realizado en bandas EUTRA (Acceso de radio en LTE) y bandas GSM (SA), por lo que este sistema se implementó en la banda de microondas, se ocupó la banda mas baja en la cual el modelo CI es válido, es decir, la banda de 2GHz.

5.1.3. Incorporación de un esquema de acceso al medio no ortogonal (NOMA) usando una técnica de agrupamiento de usuarios

Se predefinió que exista un bloque de recursos (PRB) para la BS dedicado para el estándar NB-IoT en la banda de 2GHz, dando servicio a comunicaciones tipo maquina (MTC). El recurso de radio tiene un ancho de banda de 180kHz y este recurso se dividió en 48 sub-portadoras de 3.75kHz con operación *singletone* y *multitone*.

Por lo tanto, para la compartición de recursos, en vías de dar servicio a un gran número de dispositivos, propuso implementar un esquema de acceso múltiple al medio no ortogonal (NOMA) descrito en la Sección 4.5.3 y la implementación de una técnica de agrupamiento de usuarios

(*clustering*) y se aprovechará la no ortogonalidad, para agrupar diferentes clases de nodos IoT en una misma sub-banda.

En este caso como consideraremos dos tipos de sensores los mMTC y los URLLC, (con mayores requisitos de transmisión de datos los segundos). La relación de la distribución de dispositivos mMTC con los URLLC fue de 3 a 1.

El cálculo de las tasas de los dispositivos URLLC y mMTC, se describen con las Ecuaciones 4.5 y 4.4 respectivamente.

Con la incorporación del Modelo de Canal CI, el cálculo de las ganancias h se dió de la siguiente manera:

$$h = 10^{\left(\frac{-L_{p[dB]}^{CI}}{10}\right)} \cdot \gamma [W] \quad (5.1)$$

Donde:

$\gamma \rightarrow$ ganancia desvanecimiento Rayleigh

$L_{p[dB]}^{CI} \rightarrow$ Path Loss Modelo CI

5.1.4. Diferentes modelos de tráfico que simulen distintos servicios para NB-IoT

En la implementación de modelos de tráfico se tuvo en cuenta el modelo CMMPP y un modelo determinístico. El modelo CMMPP pudo tener una instancia distinta para cada una de las aplicaciones que se propusieron en *Tabla 4.5* a excepción de aquellas aplicaciones con transmisiones periódicas, para las cuales se utilizó un modelo determinístico.

Cada tipo de dispositivo tiene una instancia de *procesos maestros*. Estos procesos, están modificando las probabilidades de cambio de estado de cada nodo según su proximidad a otros nodos que estén cambiando de estado. Por ejemplo, digamos que un nodo, que llamaremos N1, que detecta terremotos se encuentra en estado de reposo y su probabilidad de transición al estado de transmisión por alarma es del 1%, ahora de pronto otro nodo (de la misma aplicación) en su proximidad cambia de estado a transmisión por evento, y unos instantes después la probabilidad de transición en nuestro nodo que antes era del 1% aumenta hasta 90%, lo que desencadena que este nodo también comience a transmitir al cambiar de estado unos momentos después. El ejemplo anterior, en la vida real se traduciría como un terremoto ocurriendo en un lugar y una gran cantidad de nodos que se encargan de detectarlo comenzando de pronto a transmitir con una indudable coordinación espacial y temporal.

Habiendo explicado esto, la implementación del modelo CMMPP en las aplicaciones, control de iluminación, detección de terremotos y genérico contará con un diseño de cadena de Markov como el que se presenta en la *Figura 7.12*. En esta figura se puede ver que se modelarán dos distintos estados para cada nodo IoT de estas aplicaciones. El primer estado, llamado normal

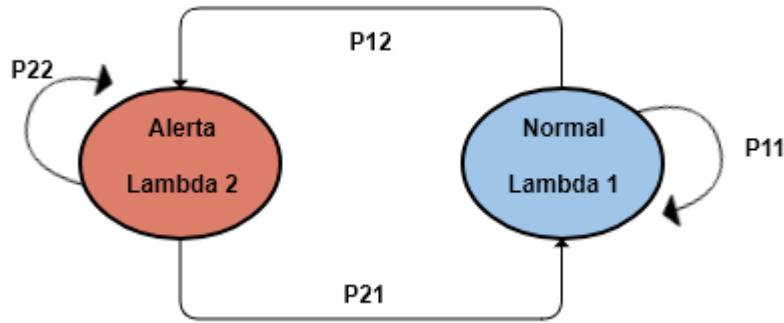


FIGURA 5.2: Cadena de Markov propuesta

corresponde al funcionamiento **normal** o principal de la aplicación, con una tasa de transmisión Lambda 1 (λ_1). A la vez el segundo estado, llamado **Alerta** o Alarma con tasa de transmisión Lambda 2 (λ_2), corresponde al estado que acudirán los dispositivos IoT de acuerdo con eventos de interés producidos aleatoriamente en el área de la célula. Es justamente con la ayuda de este estado que el modelo es capaz de simular la coordinación espacial y temporal de los dispositivos.

$$P_n [k] = n [k] P_c + (1 - n [k]) P_u \quad (5.2)$$

donde : n corresponde al n – ésmo nodo
y k a un determinado instante de tiempo

En la ecuación 5.2 vemos la forma en la que se modulan las matrices de probabilidad de transición entre los estados. Entonces para calcular la matriz $P_n [k]$, es decir la perteneciente al nodo n en el instante k , necesitamos de las matrices P_c y P_u que corresponden al comportamiento coordinado y no coordinado respectivamente y del valor $n [k]$. Las matrices P_c y P_u marcan el comportamiento que tendría el nodo en los casos extremos de coordinación o en la ausencia de esta, esto debido a que $n [k]$ varía entre $[0, 1]$, entonces si existe una perfecta coordinación entre nodos y este valor es 1, en algún momento, el segundo sumando de la ecuación 5.2 sería 0 y $P_n [k] = P_c$. La propuesta para P_c y P_u , tal y como se ha utilizado en [33] y [34] fue:

$$P_u = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \quad (5.3)$$

$$P_c = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (5.4)$$

Ahora se estudia el término $n [k]$ que es el que se encarga de simular la correlación existente entre los nodos, el cual se compone de $n [k] = \delta_n [k]$, siendo δ_n el término encargado de la coordinación espacial y $[k]$ el de la coordinación temporal. Para δ_n se utilizaron dos funciones: una exponencial decreciente *Decaying exponential* y una ventana de coseno alzado *Rised-cosine window* según convino para cada aplicación, tal y como se propone en [33]. La ventana de coseno

alzado permitió simular un término abrupto en la transmisión de las alarmas. Finalmente para $[k]$ se calculó su valor utilizando una función delta, que sólo toma el valor 1 cuando la alarma se ha propagado hasta el sitio del nodo.

Mientras un nodo se encuentre en el estado normal, este genera paquetes a la tasa λ_{normal} . Un cambio de estado se da por una alarma que se propagó hasta el sitio del nodo. En la *Figura 5.3* se presenta el diagrama de flujo con el algoritmo que se utilizó como base para generar el tráfico que sigue este modelo, hizo falta la incorporación de tráfico periódico para las aplicaciones que así lo requirieron.

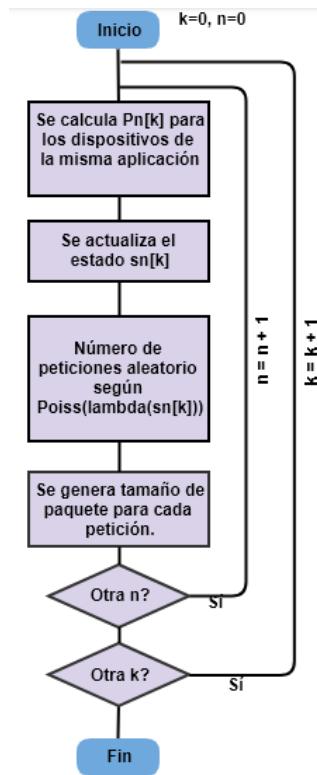


FIGURA 5.3: Generación de tráfico con modelo CMMPP

El segundo modelo se trata de uno determinístico, y fué el que se implementó en las aplicaciones cuya tasa de tráfico es periódica. De manera que lo único que se debe conocer es la tasa de tráfico de los nodos con la cual se calendarizan los nacimientos de paquetes. Las aplicaciones contempladas para este modelo son: el monitoreo del consumo de agua y electricidad en la ciudad, el monitoreo de la contaminación del aire y el control dinámico de los semáforos.

Ahora, en la *Tabla 5.2* se presentan las aplicaciones mencionadas en la sección 4.3 junto a su caracterización. Adicionalmente se anexó la distinción de las aplicaciones en dos clases distintas, mMTC y URLLC que permitió realizar el algoritmo de agrupaciones NOMA. Se buscó que por

TABLA 5.2: Caracterización de las aplicaciones seleccionadas

Número de Servicio y Nombre	Tamaño de red	Tasa de tráfico	Demanda de QoS	Clase	Tamaño de paquete
1 - Control de iluminación (Ciudad Inteligente)	Grande, miles de dispositivos	Aleatorio, poco frecuente	Media, tolerante al retardo 15seg	mMTC	Activación aleatoria UL: 20 bytes payload DL: ACK de 0 bytes
2 - Monitoreo del consumo de agua y electricidad en la ciudad (Ciudad Inteligente)	Media a grande, cientos a miles de dispositivos	Periódico, 1 msj cada 10 min por dispositivo	Baja, tolerante al retardo 1min	mMTC	Activación periódica UL: distribución de Pareto con parámetro alfa = 2.5 y tamaño mínimo de carga útil de la aplicación = 20 bytes con un corte a 200 bytes DL: ACK de 0 bytes 50 % de las veces.
3 - Detección de terremotos (Ambiente Inteligente)	Media a grande, cientos a miles de dispositivos	Aleatorio, poco frecuente	Alta, tolerante al retardo 3seg	mMTC	Activación aleatoria UL: 20 bytes payload DL: ACK de 0 bytes
4 - Monitoreo de contaminación del aire (Ambiente Inteligente)	Media a grande, cientos a miles de dispositivos	Periódico, 1 msj cada 15 min por dispositivo	Media, tolerante al retardo 15seg	mMTC	Activación periódica UL: distribución de Pareto con parámetro alfa = 2.5 y tamaño mínimo de carga útil de la aplicación = 20 bytes con un corte a 200 bytes DL: ACK de 0 bytes 50 % de las veces.
5 - Control dinámico de semáforos (Transporte y Movilidad Inteligentes)	Grande, miles de dispositivos	Periódico, 1 msj cada min por dispositivo	Alta, tolerante al retardo 5seg	mMTC	Activación periódica UL: distribución de Pareto con parámetro alfa = 2.5 y tamaño mínimo de carga útil de la aplicación = 20 bytes con un corte a 200 bytes DL: ACK de 0 bytes 50 % de las veces.
6 - Otros Dispositivos mMTC	Grande, miles de dispositivos	Aleatorio, poco frecuente	Alta, tolerante al retardo 5seg	mMTC	Activación aleatoria UL: 20 bytes payload DL: ACK de 0 bytes
7 - Dispositivos URLLC	Grande, miles de dispositivos	Aleatorio, poco frecuente	Alta, tolerante al retardo 3seg	URLLC	Activación aleatoria UL: 20 bytes payload DL: ACK de 0 bytes

cada nodo URLLC hubieran por lo menos 3 nodos mMTC. por lo tanto se utilizaron distintas intensidades de dispositivos que conservaran la relación.

En la *Figura 5.4* se muestra una aproximación del escenario implementado, usando un agrupamiento con cuatro nodos.

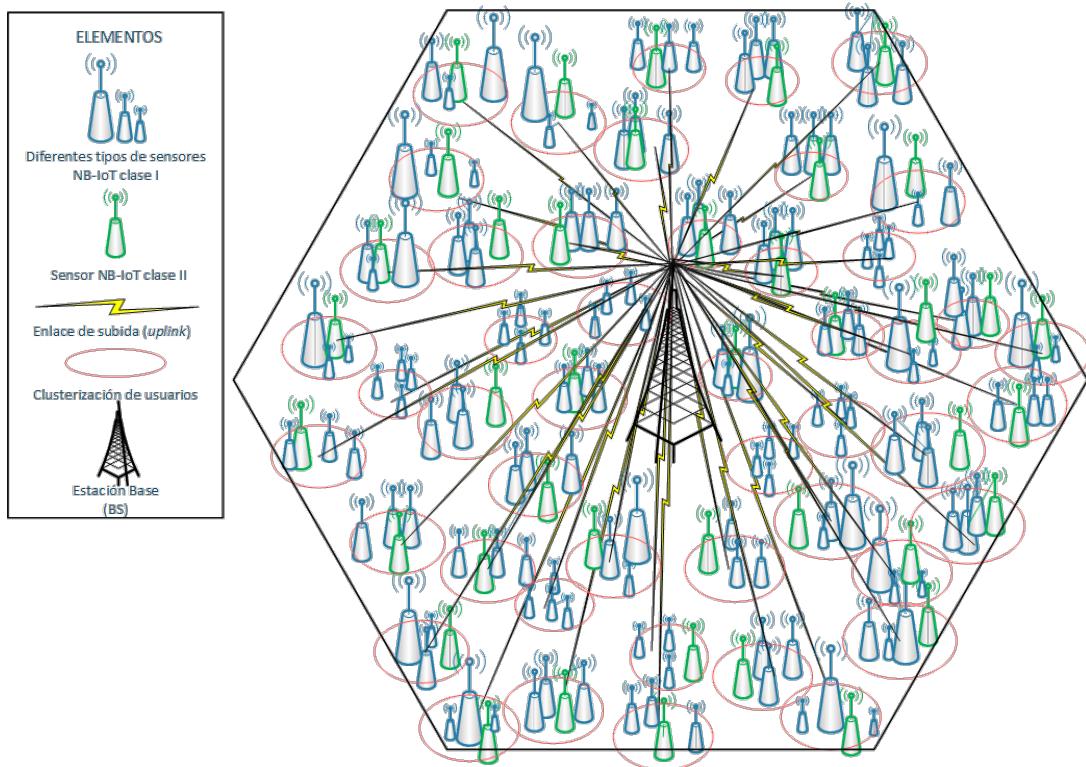


FIGURA 5.4: Ejemplo ilustrativo de un escenario mIoT unicelular aproximado, usando agrupaciones de 4 dispositivos

5.2. Parámetros generales del simulador

Para planear un modelo de sistema válido y coherente se incorporaron algunos parámetros generales de diseño revisados en los artículos [37], [36] y [33]. La *Tabla 5.3* describe el conjunto de parámetros de la simulación.

Para la generación de variables aleatorias se usaron las librerías *scipy*, *numpy* ó *random* en Python.

TABLA 5.3: Parámetros de la simulación a nivel de sistema

Parámetro	Valor
<i>Escenario</i>	UMa
<i>Ambiente</i>	LoS
<i>Diseño celular</i>	uni-cellular
<i>Transmisión</i>	UL
<i>Radio de celula</i>	200 m
<i>Movilidad de UEs</i>	Nula - 0km/h
<i>Distribución de UEs</i>	Procesos puntuales de Poisson Homogeneos (HPPP)
<i>Modelo de canal de propagación</i> <i>'Path Loss'</i>	Modelo CI $L_p^{CI}(f, d)_{[dB]} = 32,4 + 10 \log_{10} \left(\frac{d}{d_0} \right) + 20 \log_{10} (d_0) + 20 \log_{10} (f) + x_{\sigma}^{CI}$
<i>PLE</i>	2
d_0	1m
<i>Modelo de Tráfico</i>	CMMPP y Periódico
<i>Esquema de acceso múltiple al medio</i>	NOMA usando agrupamientos
<i>Relación entre dispositivos mMTC y uRLLC</i>	3 a 1
K_{max}	1, 2, 3 y 4
<i>Número de antenas BS</i>	1
<i>Número de antenas UE</i>	1
<i>Potencia máxima de transmisión de nodos uRLLC</i>	23 dBm
<i>Potencia máxima de transmisión de nodos mMTC</i>	23, 20 o 14 dBm
<i>Densidad de Ruido Térmico</i>	-174 dbm/Hz
<i>Frecuencia de operación</i>	En banda y guarda de banda LTE estándar, Banda de 2GHz
<i>Ancho de banda del sistema para un PRB (BW)</i>	180 KHz
<i>Espacio entre sub-portadoras Uplink</i>	3.75kHz singleton y multitone [37]
<i>Tasa de datos máxima</i>	UL: 20kbps

Capítulo 6

Implementación

En este capítulo se explica la implementación del modelo de sistema propuesto en los capítulos anteriores, se generaron los escenarios de prueba pertinentes para los cuatro modelos que comprenden el simulador y se muestran resultados específicos por cada modelo. Por último, se definió la metodología de cómo se interconectaron estos cuatro modelos con el fin de dar por terminada la fase de implementación y con esto poder generar distintos resultados globales del simulador en el capítulo siguiente.

6.1. Generación de una geometría estocástica

Generación de Puntos PPP en un círculo:

Para la generación de puntos en un círculo correcta, que cumpla con la definición de un HPPP [Sección 2.1.1], el número total de puntos que se formaron en el círculo siguió una variable aleatoria de Poisson, además, estos puntos se distribuyeron de acuerdo a una distribución uniforme dentro del círculo, por último, al considerar un proceso homogéneo, la densidad de los puntos fue constante.

1. Número total de puntos:

En primer lugar, para cumplir con la definición de HPPP, el número total de puntos tendrá que seguir una distribución de Poisson. Para esto en cada realización se generó una variable aleatoria de Poisson con tasa constante λ y media λA donde $A = \pi r^2$.

2. Ubicación de los puntos:

Después, para la distribución uniforme de puntos en un círculo, se generaron dos variables aleatorias uniformes, una que representa el radio y otra el ángulo (en coordenadas polares).

Entonces, debido a que el área aumenta de manera cuadrática con relación al radio, un punto aleatorio ubicado uniformemente en un círculo de radio r tiene las coordenadas polares $[r(\sqrt{U}), 2\pi V]$, donde U y V son las dos variables aleatorias antes mencionadas, $U, V \in [0, 1]$.

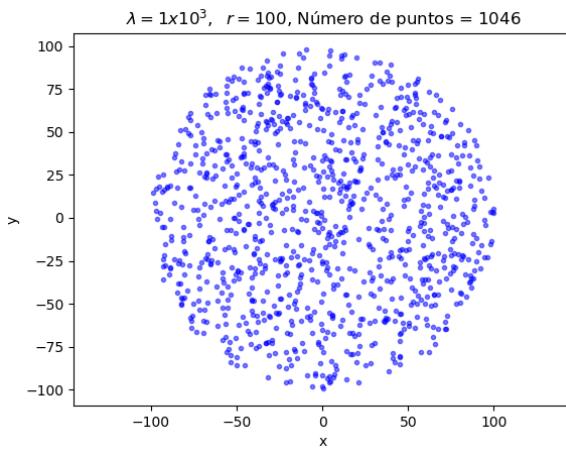


FIGURA 6.1: Distribución espacial de puntos PPP con media $\lambda = 1 \times 10^3$ y $r = 100$

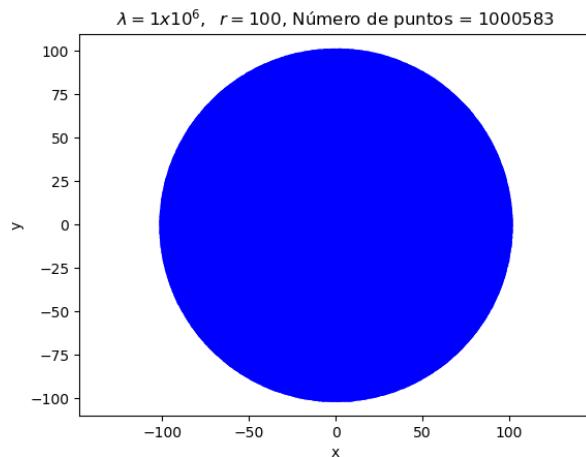


FIGURA 6.2: Distribución espacial de puntos PPP con media $\lambda = 1 \times 10^6$ y $r = 100$

6.1.1. Escenario de Prueba de PPP

En este escenario se validó la generación de una geometría estocástica siguiendo un HPPP.

En primer lugar, el número total de puntos debe seguir una distribución discreta de Poisson, esta se implementó con la librería *scipy* [Véase Apéndice A].

Después, se comprobó la generación de puntos aleatorios ubicados uniformemente en un círculo de radio r en coordenadas polares $[r(\sqrt{U}), 2\pi V]$.

Se graficaron los puntos en un círculo, con tasa constante $\lambda = 1 \times 10^3$ y $r = 100$ con media λA donde $A = \pi r^2$ [véase Figura 6.1].

Por último, se validó que todos los puntos se generen dentro del área del círculo, para esto se graficó de igual manera la distribución espacial de los puntos en un círculo en plano 2D pero ahora con una densidad mayor ($\lambda = 1 \times 10^6$), de modo que se observe que los puntos llenan toda el área del círculo de manera uniforme [véase Figura 6.2].

6.2. Generación de ganancias de canal

El canal propuesto, con modelo CI (Ecuación 4.2 de [48]), implementa pérdidas de acuerdo con la distancia y frecuencia, dado un exponente de pérdida por trayectoria (PLE) y agrega pérdidas por el desvanecimiento rápido de Rayleigh.

En primera instancia, para el cálculo de la pérdida por trayectoria, se utilizó la ecuación 4.2.

Retomando lo estudiado en la Sección 2.2.3, se tiene que cuando el desvanecimiento es tipo Rayleigh, la magnitud (amplitud) de la señal es distribuida por una variable aleatoria tipo Rayleigh pero en potencia es distribuida exponencialmente con media unitaria.

La ganancia de desvanecimiento Rayleigh se modeló por medio de la generación de una variable aleatoria que siguió una distribución Rayleigh con desviación estándar unitaria, la generación de esta variable se puede ver en el Apéndice A.

6.2.1. Escenario de Prueba del Modelo de canal CI

Se propuso un escenario UMa en exteriores en un ambiente con línea de vista (LoS), entonces de acuerdo con la Tabla 5.1, los parámetros válidos del modelo de canal CI en ambientes LoS, son: PLE = 2; frecuencia de operación = 2- 38GHz; rango de distancias 60- 930 m. La Figura 6.3 muestra las pérdidas por trayectoria que dependen de las distancias (entre BS y UE) y frecuencias en todo el rango en el cual es válido este modelo (sin tomar en cuenta el desvanecimiento). Se observa el incremento de las pérdidas conforme aumentan la distancia y la frecuencia.

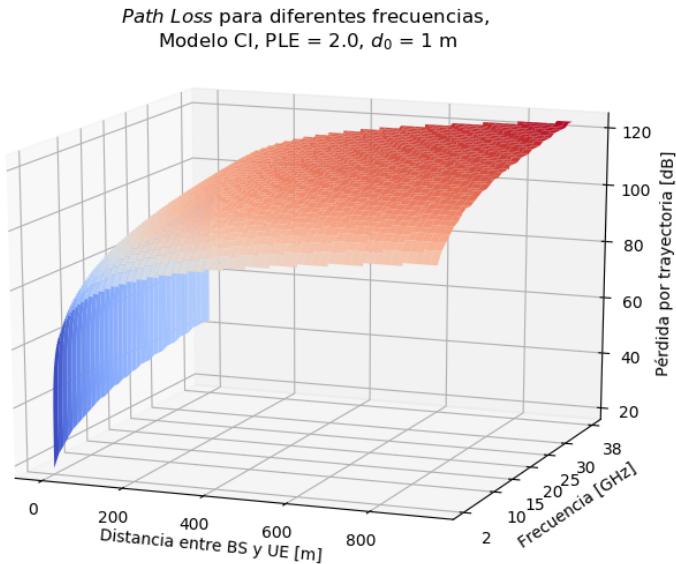


FIGURA 6.3: Pérdidas por trayectoria según el Modelo CI

Ahora bien, al incorporar el desvanecimiento tipo Rayleigh y las pérdidas por trayectoria, siguiendo la Tabla 5.3 donde se definen los parámetros de la simulación (PLE= 2 con radio celular de 200m, en una banda de 2 GHz), se generó el histograma normalizado (pdf) de los coeficientes

de canal (ganancias de canal) esperados para el simulador, véase Figura 6.4, se puede observar que el valor promedio de las ganancias (h) es de 80.35 dB.

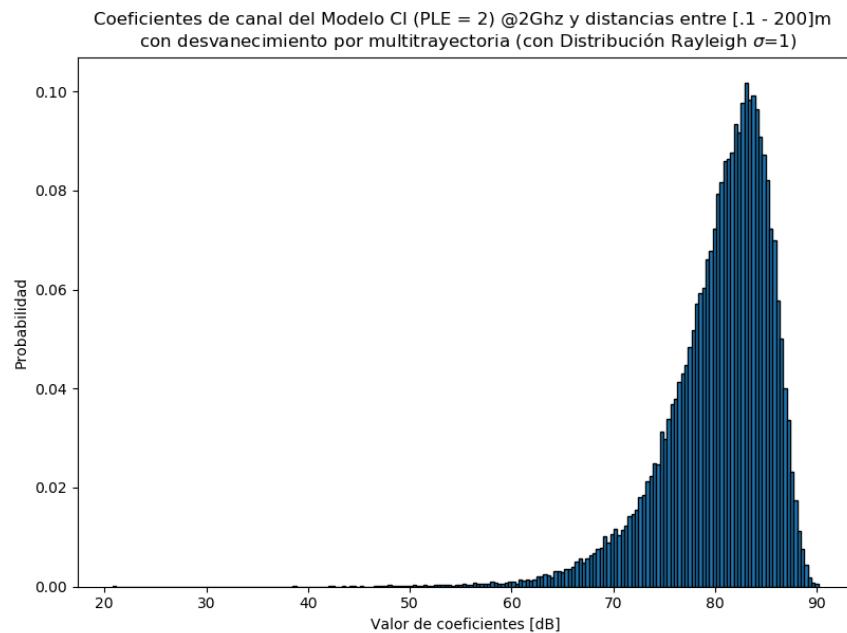


FIGURA 6.4: Ganancias de Canal Modelo CI

6.3. Esquema de acceso múltiple al medio no ortogonal, basado en potencia (PD-NOMA)

En el diseño del sistema se decidió implementar un esquema de acceso múltiple no ortogonal (NOMA), con base en [37], los autores desarrollaron un esquema NOMA basado en NB-IoT, esto se realizó por medio de dos algoritmos, un algoritmo para el agrupamiento de los usuarios y otro para la asignación de recursos.

6.3.1. Algoritmo de Agrupación de dispositivos uRLLC y mMTC

Primero se implementó el algoritmo de agrupamiento NOMA para los dispositivos mMTC y uRLLC descrito en [37][véase *Algorithm 1*], este algoritmo realiza un ordenamiento conveniente con respecto al uso de la Cancelación Sucesiva de Interferencia (SIC), es decir, se ordenan los dispositivos URLLC y mMTC de acuerdo con su ganancia de canal promedio dentro de los diferentes grupos NOMA para que puedan compartir el mismo recurso espectral (subportadora) asignado a cada grupo[véase *Figura 6.5*]. El Algoritmo da prioridad a los dispositivos URLLC para que obtengan los rangos más bajos.

Por lo tanto, un mensaje combinado de los dispositivos mMTC y URLLC con ruido aditivo es recibido en la BS, la BS emplea la recepción SIC de acuerdo a cómo son ordenados los dispositivos.

La recepción SIC decodifica primero el mensaje del dispositivo con el rango más bajo, por consiguiente los usuarios con los rangos siguientes (o más altos) le introducen interferencia y por el contrario el usuario con el rango más alto no experimenta interferencia de ninguna señal.

Es importante notar que los dispositivos URLLC tienen requerimientos de tasas de datos más altos, por lo tanto, su potencia de transmisión será mayor que la de los dispositivos mMTC. Es por esto que en cada grupo los dispositivos uRLLC tendrán los rangos más bajos y los dispositivos mMTC los más altos. De hecho, el decodificador SIC en la BS comienza a decodificar el mensaje URLLC y, en consecuencia, los dispositivos mMTC no se ven afectados por la alta interferencia [véase *figura 6.5*].

Siguiendo la definición de las variables de los algoritmos en [37] [descritas en la Sección 4.5.3], se tiene:

Algorithm 1 Algoritmo de agrupamiento de dispositivos uRLLC y mMTC para NOMA

Entrada U, M, S, C, h_m^s , and $h_u^s, \forall m \in \mathcal{M}, \forall u \in \mathcal{U}, \forall s \in \mathcal{S}$

Salida Lista de Clusters (C) con agrupamiento de dispositivos

- 1: **procedure** AGRUPACIÓN DE DISPOSITIVOS uRLLC
 - 2: Cálculo de la ganancia de canal promedio del enésimo dispositivo u
 - 3: $\tilde{h}_u = \sum_{s \in \mathcal{S}} h_u^s / |\mathcal{S}|$
 - 4: Se ordenan descendentemente las ganancias de canal promedio de cada dispositivo u , i.e.
 $\forall u \in \mathcal{U} : \tilde{h}_1 \geq \tilde{h}_2 \geq \dots \geq \tilde{h}_U$
 - 5: **for each** u in U **do**
 - 6: **if** $|U| < |C|$ **then**
 - 7: Asignar uRLLC al rango mas bajo ($k=1$)
 - 8: **else**
 - 9: Asignar uRLLC al siguiente rango ($k=2$) [Solo se podrán asignar hasta un segundo rango]
 - 10: Encontrar \tilde{k} , rango y grupo en el que se quedó la última asignación uRLLC
 - 11: **procedure** AGRUPACIÓN DE DISPOSITIVOS mMTC(\tilde{k})
 - 12: Cálculo de la ganancia de canal promedio del enésimo dispositivo u
 - 13: $\tilde{h}_m = \sum_{s \in \mathcal{S}} h_m^s / |\mathcal{S}|$
 - 14: Se ordenan descendentemente las ganancias de canal promedio de cada dispositivo u , i.e.
 $\forall m \in \mathcal{M} : \tilde{h}_1 \geq \tilde{h}_2 \geq \dots \geq \tilde{h}_M$
 - 15: **for each** m in M **do**
 - 16: **if** $|M| < |C|$ **then**
 - 17: Asignar mMTC al rango \tilde{k}
 - 18: **else**
 - 19: Asignar mMTC a los siguientes rangos ...
-

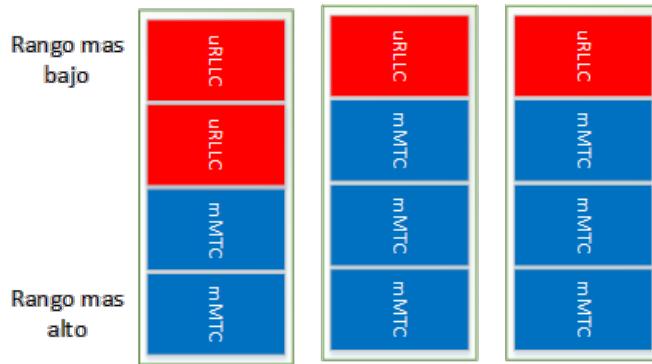


FIGURA 6.5: Ejemplo ilustrativo del ordenamiento de 4 usuarios uRLLC y 8 usuarios mMTC en 3 grupos NOMA

6.3.2. Algoritmo de Asignacion de Subportadoras

Este algoritmo garantiza una óptima asignación de portadoras (S), de acuerdo con la maximización de la tasa total de transmisión, para los grupos NOMA. En [37] se plantea la metodología para la asignación de subportadoras a los grupos NOMA.

Para cada subportadora, el mejor grupo (c^*) es el que maximiza el rendimiento total. Luego, por consecuencia, las velocidades de transmisión de los dispositivos mMTC y uRLLC y sus potencias de transmisión se actualizan. Durante el proceso de asignación de subportadoras, los clústeres asignados se excluyen del conjunto de grupos no asignados (variable C_{ns}).

En el algoritmo [37], mencionan que dado que cada dispositivo MTC divide por igual su potencia de transmisión máxima entre todas las subportadoras asignadas, las potencias de transmisión de los dispositivos a través de las diferentes subportadoras no son óptimas.

Durante la implementación del Algoritmo en [37], se notó que los autores [37] siempre buscan alcanzar la tasa objetivo de cada dispositivo uRLLC y mMTC en cada cluster haciendo la agregación de multiples subportadoras a cada clúster, el problema que surge con esto, es que muchas de las veces que se corría la simulación, la mayoría de las subportadoras se asignaban de entre 1 a 3 grupos, esto porque parecía difícil cumplir con las tasas de todos los dispositivos en estos grupos, dejando a los otros grupos sin subportadora. La modificación que se implementó fue que siempre se logre asignar una subportadora para cada grupo aunque sus dispositivos no alcancen su tasa objetivo, es decir se adoptó un número fijo de grupos NOMA igual a 48.

En el algortimo implementado en este proyecto la fase de asignación de recursos termina hasta que las 48 subportadoras se asignan a los 48 grupos NOMA, esto para una transmisión *singleton*.

Los autores en [37] proponen la oportunidad de asignar más de una subportadora (*multitone*) por grupo NOMA en transmisiones con anchos de banda UL de 3.75KHz para NB-IoT.

A pesar de que aún no ha sido especificada la asignación de múltiples subportadoras por cluster en anchos de banda de 3.75KHz para NB-IoT, se tomó la propuesta de [37] en este simulador.

Se intentó aplicar una transmisión *multitone* de otro modo, analizemos el siguiente ejemplo:

Si en un TTI, 50 dispositivos desean transmitir, en un modo *singletone* se crearían siempre 48 grupos, entonces, 48 usuarios se agruparían en el primer rango de cada grupo y los otros dos restantes se agruparían en el segundo rango de los dos primeros grupos. En este caso no puede hacerse *multitone* ya que las 48 subportadoras son ocupadas por cada grupo NOMA.

Ahora bien, un modo *multitone* se podría aplicar de la siguiente manera, si se consideran de igual manera 50 dispositivos queriendo transmitir, estos se pueden dividir en dos grupos de 25 dispositivos, es decir crear 25 grupos con su primer y segundo rangos completos, esto con la finalidad de que las 23 subportadoras restantes se agreguen en los grupos donde sus dispositivos aún no alcanzan su tasa objetivo, esto se implementó siguiendo la metodología del *algorithm 3*.

En la Figura 6.6 se puede observar la estrategia multitone propuesta.

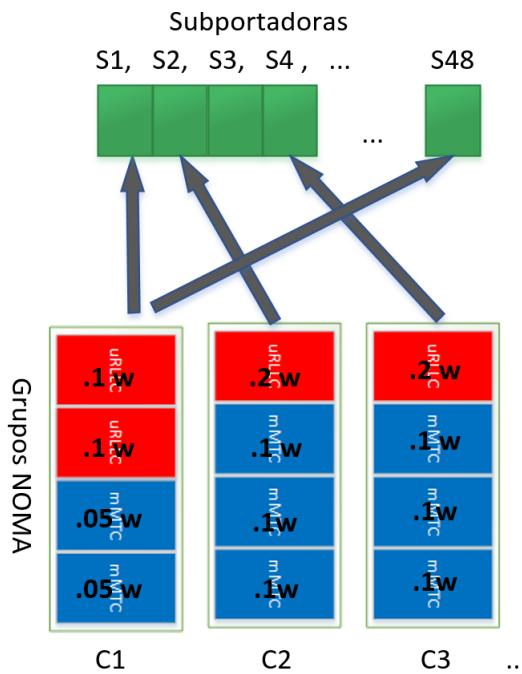


FIGURA 6.6: Ejemplo ilustrativo de la distribución de potencia para PD-NOMA en *Multitone*

Algorithm 2 Algoritmo de asignación de recursos para NOMA

Entrada $U, M, S, C, R_m^{th}, R_u^{th}, P_m^{max}, P_u^{max}, h_m^s, \text{ and } h_u^s, \forall m \in \mathcal{M}, \forall u \in \mathcal{U}, \forall s \in \mathcal{S}$

Salida Asignaciones de grupos con subportadoras (Asignación de todas las subportadoras [48, NB-IoT singleton])

```

1: procedure ASIGNACIÓN DE SUBPORTADORAS
2: Inicialización :
3:  $R_u = 0, R_m = 0, p_m^s = P_m^{max} \text{ y } p_u^s = P_u^{max}, \forall m \in \mathcal{M}, \forall u \in \mathcal{U}, \forall s \in \mathcal{S}$ .  $\hat{S} \leftarrow \emptyset, S_a^c \leftarrow \emptyset, C_{ns} \leftarrow \mathcal{C}$ 
4:   while  $\mathcal{S} \neq \emptyset$  do
5:     Asignación de una subportadora a un grupo NOMA:
6:       for each  $s$  in  $S$  do
7:         Seleccionar al mejor cluster  $c^*$  (el que maximiza el throughput):
8:          $c^* = \arg \max_{c \in C_{ns}} (\sum_{u \in \mathcal{U}} R_u + \sum_{m \in \mathcal{M}} R_m)$ ;
9:         donde:  $R_u$  (Ecuación 4.5) y  $R_m$  (Ecuación 4.4), de acuerdo con [37]
10:        Asignar la subportadora  $s$  con  $c^*$ :
11:        Actualizar  $S_a^{c^*} \leftarrow S_a^{c^*} \cup \{s\}, \hat{S} \leftarrow \hat{S} \cup \{s\}$ 
12:        Actualizar las tasas :  $R_u = R_u + R_{u,s}, R_m = R_m + R_{m,s}$ 
13:        Actualizar las potencias de los URLLC y mMTC de  $c^*$  individualmente :
14:         $p_m^s = \frac{p_m^s}{|S_a^{c^*}|+1}, p_u^s = \frac{p_u^s}{|S_a^{c^*}|+1}, \forall s \in \mathcal{S}$ 
15:        if  $S_a^{c^*} == 1$  then
16:           $C_{ns} \leftarrow C_{ns} \setminus \{c^*\}$ 
17:        Asignación de subportadoras restantes:
18:        if  $C_{ns} \neq \emptyset$  then
19:          for each  $s$  in  $S$  do
20:             $\mathcal{S} \leftarrow \mathcal{S} \setminus \hat{S}$ 
21:             $c^* = \arg \max_{c \in C} (\sum_{u \in \mathcal{U}} R_u + \sum_{m \in \mathcal{M}} R_m)$ 
22:            Actualizar  $S_a^{c^*} \leftarrow S_a^{c^*} \cup \{s\}$ 
23:            Actualizar  $p_m^s = \frac{p_m^s}{|S_a^{c^*}|+1}, p_u^s = \frac{p_u^s}{|S_a^{c^*}|+1}$ 

```

Algorithm 3 Modo de Operacion *Multitone*

Entrada NumeroDispositivos

Output NumeroGrupos

```

1: procedure EJEMPLO DE PROCEDIMIENTO MULTITONE CON KMAX=4
2:   if  $kmax == 4$  then
3:     if  $NumeroDispositivos <= 48$  then return NumeroDispositivos
4:     if  $NumeroDispositivos > 48$  and  $NumeroDispositivos <= 96$  then return Nu-
       meroDispositivos/2
5:     if  $NumeroDispositivos > 96$  and  $NumeroDispositivos <= 144$  then return Nu-
       meroDispositivos/3
6:     if  $NumeroDispositivos > 144$  and  $NumeroDispositivos < 192$  then return Nu-
       meroDispositivos/4
7:   if  $NumeroDispositivos >= 192$  then return 48

```

6.3.3. Escenario de Prueba de NOMA

En este escenario de pruebas, se evaluó el rendimiento del sistema del esquema NB-IoT basado en NOMA propuesto en [37], el cual implementó dos algoritmos: uno para la formación de grupos y otro para la asignación de recursos.

Para la validación de los algoritmos, se consideró el modelo de sistema que proponen en [37], con el fin de poder realizar comparaciones con los resultados obtenidos en [37].

Entonces, en su modelo de sistema, los autores en [37] consideraron los siguiente:

Una celda con un radio de 0.5 km donde las ubicaciones de los dispositivos mMTC y URLLC se generaron aleatoriamente (de manera uniforme) dentro de la celda. Además, consideraron un solo PRB con 48 subportadoras de 3.75 kHz para las transmisiones de enlace ascendente (UL) en un intervalo de tiempo (TTI).

Modelaron las ganancias de canal de los dispositivos mMTC como $h_s^m = \gamma d_{m,s}^{-\beta}$ (de manera similar h_s^u para URLLCs), donde γ es un valor aleatorio generado por la PDF de Rayleigh, $d_{-\beta}^{m,s}$ representa la distancia entre el UE y la BS, y β es el PLE el cual lo establecieron en $\beta = 3$ y d varía entre 0.1 y 500 m.

Los coeficientes de canal del modelo de sistema en [37], se representan en la Figura ??.

Consideraron el ruido gaussiano blanco aditivo (*AWGN*) con una densidad espectral de potencia de $-173dBm/Hz$. Los presupuestos de potencia de transmisión máxima de todos los dispositivos URLLC y mMTC, P_{max}^u y P_{max}^m ($\forall m \in \mathcal{M}, \forall u \in \mathcal{U}$), se establecen en 23 dBm. Los umbrales de velocidad de datos de los dispositivos mMTC siguen una distribución uniforme, es decir, $R_{th}^m = Uniforme(0,1,2)kb/s$, $R_{th}^u = Uniforme(0,1,20)kb/s$.

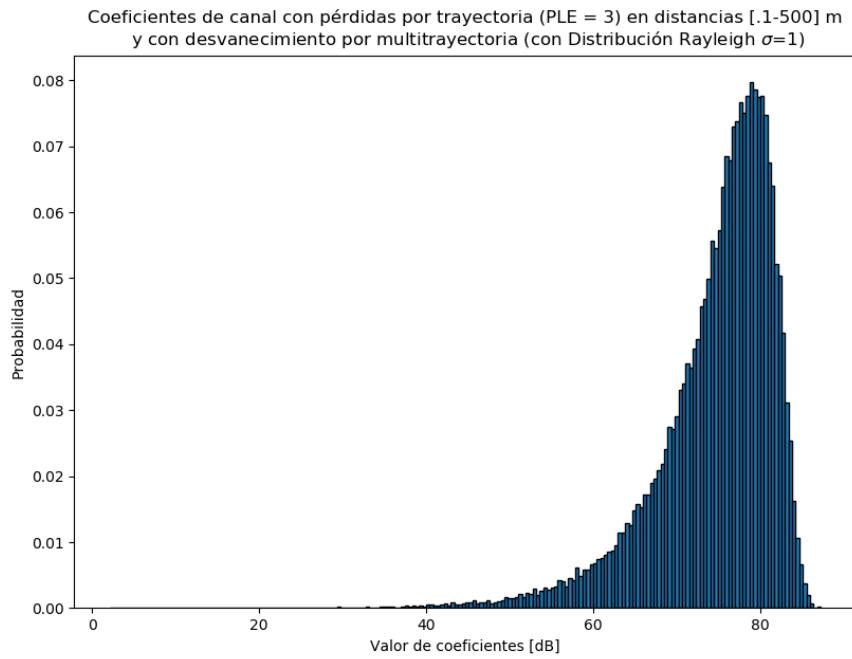


FIGURA 6.7: Coeficientes de canal del modelo de sistema en [37]

Cabe destacar que no necesariamente deben coincidir nuestros resultados con los del artículo ya que se tomaron algunas consideraciones diferentes a la de ellos, explicadas en la sección anterior. Esto puede comprobarse en la figura 6.8, al comparar los resultados de nuestro algoritmo con el de los autores en [37], se puede observar que nuestro algoritmo tiene un menor desempeño especialmente en los casos de 75 a 175 usuarios, empatando cuando son 192 usuarios.

Además, al comparar los resultados entre *multitone* y *singletone* de nuestro algoritmo (en la figura 6.8) se puede ver que el desempeño de ambos modos de transmisión es similar en la mayoría de los casos.

El impacto de la agregación de las subportadoras solo beneficia a aquellos dispositivos que tienen una buena condición de canal o que están en los últimos rangos (ya que experimentan menos interferencia).

Se estudió de igual manera el impacto de *multitone* con diferentes clases de potencia para los dispositivos MTC con $k_{max} = 4$ y su desempeño no resultó ser favorable. Igualmente, se evaluó el desempeño con diferentes tamaños de grupos. Por ese motivo esos resultados no se reportaron.

Como se observa, no se logran satisfacer todos los requisitos de tasa de datos de los dispositivos mMTC y URLLC pero sí se incrementa significativamente la conectividad de usuarios en comparación con OFDMA donde sólo se puede dar servicio a un usuario por subportadora, lo que

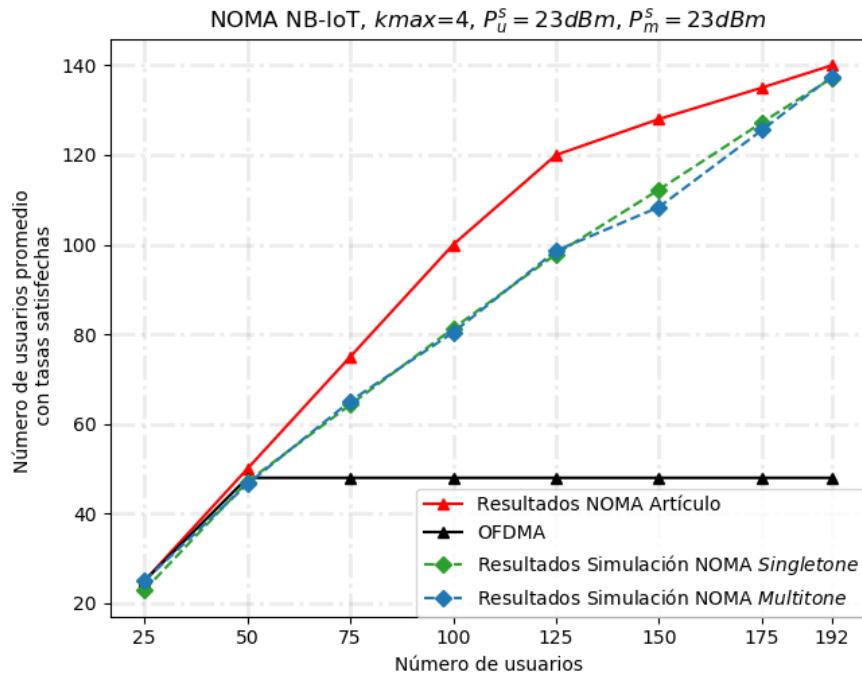


FIGURA 6.8: Escenario de pruebas de esquema NOMA

son 48 usuarios.

En la figura 6.9, se muestran la comparación de resultados de la *sum-rate* alcanzada de los resultados de nuestro algoritmo con el de los autores en [37], se puede ver que en un modo multitone se alcanza mas tasa, esto es porque se agregan las subportadoras sobrantes, esto cuando el número de dispositivos es menor al número de grupos.

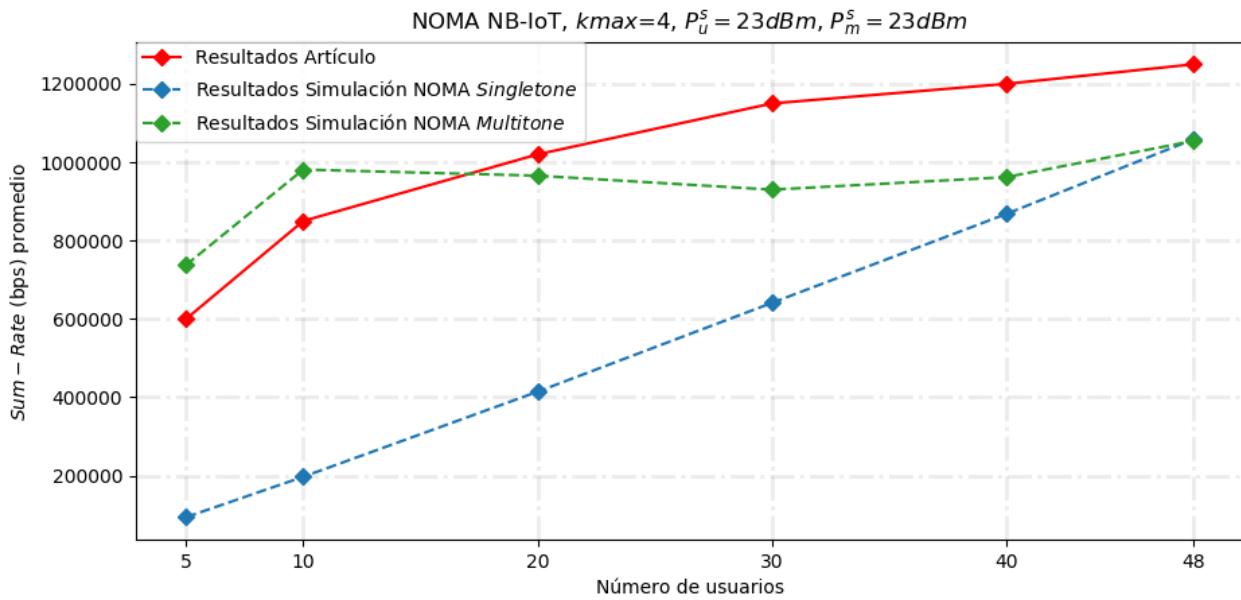


FIGURA 6.9: Segundo escenario de pruebas de esquema NOMA

6.4. Generación de Tráfico Fuente

La generación de tráfico fuente requirió de un paradigma distinto al de eventos discretos, pues como se deja ver en el diagrama de la [figura 5.3](#), un avance periódico en el tiempo es necesario para calcular las matrices de cambio de estado en los dispositivos y generar así el tráfico CMMPP, de esta manera el algoritmo de generación de tráfico fuente sigue un paradigma de simulación normalmente llamado *time driven*. Adicional a este algoritmo se anexó la lógica para generar tráfico periódico en los dispositivos que así lo requerieran, teniendo así dos posibles modelos. Tal algoritmo corresponde al [algoritmo 4](#).

El programa Generador de Tráfico IoT se trata de la interconexión entre el algoritmo de distribución de usuarios, tratado en [6.1](#) y el de generación de tráfico fuente tratado en esta sección. A este programa se agrega una interfaz gráfica para facilitar la introducción de los parámetros que se consideran para generar el tráfico IoT y se describe más a detalle en la sección [6.6](#).

En las siguientes dos secciones se muestran los parámetros que fueron implementados para caracterizar el tráfico generado por los modelos CMMPP y Periódico.

6.4.1. Tráfico CMMPP

La generación de tráfico CMMPP, sigue el diseño mostrado en la [figura 5.2](#), para el cual cada dispositivo puede estar en estado de alarma o estado normal en un instante dado. El estado

normal representa la operación habitual del dispositivo y a esta la caracteriza una tasa de arribos λ_{normal} que representa la cantidad promedio de paquetes por segundo que un dispositivo genera al estar en este estado. El estado de alarma por otra parte representa la detección de un evento o estado de interés especial para el dispositivo. Esta detección genera un único paquete de tamaño fijo detrás del cual el dispositivo volvería a su estado de operación habitual.

El cambio de estado de normal al estado de alarma requiere primeramente de un proceso en la simulación que se encargue de generar alarmas para cada tipo de dispositivo, este proceso se caracteriza por una tasa de arribo de alarmas λ_{alarm} . Esta tasa representa la cantidad promedio de alarmas por segundo que ocurrirán en el área de la célula, además de este parámetro se necesita especificar la velocidad a la que las alarmas se transmiten desde su lugar de origen.

El siguiente proceso que propicia un cambio de estado, es el cálculo de la matriz de cambio de estado $P_n [k]$ que se lleva a cabo en cada dispositivo. La ecuación 5.2 utiliza la información de cuándo y donde se produjeron eventos de alarma, además de su velocidad de propagación para determinar el momento en el que la alarma llega al dispositivo. Adicionalmente es necesario un modelo que caracterice la propagación espacial. Los modelos implementados fueron los propuestos en [33] :

$$\delta_n = e^{-\alpha \cdot d_n} \quad (6.1)$$

$$\delta_n = \begin{cases} 1 & \text{si } d_n < d_{th} \\ \frac{1}{2} \left(1 - \sin \frac{\pi(d_n - W)}{2(W - d_{th})}\right) & \text{si } d_{th} < d_n < 2W - d_{th} \\ 0 & \text{si } d_n > 2W - d_{th} \end{cases} \quad (6.2)$$

El modelo que describe la ecuación 6.1 se trata del *decaying exponential* y está pensado para simular alarmas cuyo efecto en los dispositivos se espera siga un comportamiento exponencial. Los tipos de dispositivos que utilicen este modelo deberán especificar la constante α que caracteriza al modelo. Por otra parte la ecuación 6.2 se trata de una *raised-cosine window*, esta función es útil para modelar alarmas que se transmiten fuertemente por cierta distancia pero que después se detienen abruptamente. Los tipos de dispositivos que utilicen este segundo modelo deberán especificar las constantes d_{th} y W , donde W es el ancho de la ventana y d_{th} es la distancia de corte. En ambas ecuaciones d_n es la distancia del dispositivo al origen de la alarma.

Expuesto esto, los parámetros necesarios para la generación de tráfico con el modelo CMMPP, los cuales variarán según se decida para cada tipo de dispositivo, se encuentran en la tabla 6.1.

Con lo que respecta al tamaño de los paquetes, aquellos generados a partir del estado de alarma son de 20 *bytes* y los generados en el estado normal siguen una distribución de Pareto con $\alpha = 2.5$, tamaño mínimo de 20 *bytes* y máximo de 200 *bytes*.

TABLA 6.1: Parámetros necesarios para generar tráfico CMMPP

Parámetro	Descripción
<i>Tasa de nacimiento de paquetes</i> λ_{normal}	Media de nacimiento de paquetes por segundo en cada dispositivo
<i>Tasa de nacimiento de alarmas</i> λ_{alarm}	Media de nacimiento de alarmas
<i>Velocidad de propagación de alarmas</i>	Velocidad a la que se propaga una alarma desde su posición de origen hacia todas direcciones.
<i>Modelo de propagación espacial de las alarmas</i>	Ya sea <i>decaying exponential</i> o <i>raised-cosine window</i> .
<i>Constante 1 propagación espacial</i>	Si el modelo de propagación corresponde a <i>decaying exponential</i> : se trata de la constante α , factor de decaimiento. Si el modelo de propagación corresponde a <i>raised-cosine window</i> : se trata de la constante W , ancho de la ventana.
<i>Constante 1 propagación espacial</i>	Si el modelo de propagación corresponde a <i>decaying exponential</i> : esta constante no se utiliza. Si el modelo de propagación corresponde a <i>raised-cosine window</i> : se trata de la constante d_{th} , distancia de corte.

TABLA 6.2: Parámetros necesarios para generar tráfico Periódico

Parámetro	Descripción
<i>Frecuencia de nacimiento de paquetes</i> $F_{nacimiento}$	Cantidad de paquetes que genera un dispositivo por segundo, inversa al periodo entre dos nacimientos $T_{nacimiento}$.

6.4.2. Tráfico Periódico

La generación del tráfico periódico no requiere mas que de un parámetro, la frecuencia de nacimiento de paquetes $F_{nacimiento}$, que es inversa al periodo de nacimiento de estos mismos $T_{nacimiento}$. En la implementación además, el tiempo del primer nacimiento de cada dispositivo que siga este modelo se calcula siguiendo una distribución uniforme que genera un valor entre 0 y el periodo de nacimiento y a partir de ahí se genera un nuevo nacimiento una vez transcurrido un periodo $T_{nacimiento}$. De esta manera, la transmisión de paquetes desde distintos dispositivos del mismo tipo se encuentra distribuida uniformemente a lo largo de un periodo.

La tabla 6.2 muestra el parámetro necesario para generar tráfico con el modelo periódico.

6.4.3. Algoritmo de generación de tráfico fuente

El programa de generación de tráfico fuente, implementa la generación de tráfico CMMPP y periódico según lo requiera cada tipo de dispositivo. El tráfico en cada tipo de dispositivo está

caracterizado por los parámetros de entrada descritos en las *tablas 6.1 y 6.2*. Esta caracterización es compartida entre todos los dispositivos del mismo tipo, pero será diferente con dispositivos de otros tipos, según se seleccionen los parámetros.

El *algoritmo 4* describe el proceso seguido por el programa de generación de tráfico.

Algorithm 4 Generación de tráfico fuente

Entrada $T_{límite}$, Δ_{tiempo} , $R_{célula}$, $S_1, S_2, S_3, S_4, S_5, S_6, S_7$,
Salida $ListaEventos$, $ListaDispositivos$

```

1: procedure GENERACIÓN DE TRÁFICO IoT
   Inicialización :
2:  $D = []$ ,  $G = []$  // listas de dispositivos y generadores de alarmas
3:   for  $i$  in range(1,8) do
4:     // se crean los dispositivos especificados para cada servicio
5:      $D.append(\text{crearDispositivos}(S_i, R_{célula}))$ 
6:     // se crea un generador de alarma para cada servicio
7:      $G.append(\text{crearGeneradorAlarmas}(S_i, R_{célula}))$ 
8:  $tiempo = 0$ 
9: // rutina principal
10:  while  $tiempo < T_{límite}$  do
11:    for  $dispPorServicio, generadorAlarma$  in  $D, G$  do
12:      if  $tiempo == 0$  then
13:        // se crea la primera alarma para cada servicio
14:         $nuevaAlarma = generadorAlarma.gerarAlarma(tiempo)$ 
15:      for dispositivo in  $dispPorServicio$  do
16:        // ciclo que recorre cada dispositivo de un servicio
17:        if  $dispositivo.modeloTrafico == \text{CMMPP}$  then
18:           $dispositivo.registrarAlarma(nuevaAlarma)$ 
19:           $dispositivo.calcularPnk(tiempo, generadorAlarma)$ 
20:           $dispositivo.actualizarEstado()$ 
21:           $dispositivo.gerarPaquete(tiempo)$ 
22:        else
23:           $dispositivo.gerarPaquetePeriodico(tiempo)$ 
24:         $nuevaAlarma = generadorAlarma.gerarAlarma(tiempo)$ 
25:         $tiempo += \Delta_{tiempo}$ 
26:   $ListaEventos = D.ListaNacimientos$ 
27:   $ListaDispositivos = D.ListaDispositivos$ 
28:
```

El *algoritmo 4* se programó en python 3.6 para realizar el escenario de pruebas y hacer inmediata su reutilización en el programa Generador de tráfico IoT. Los parámetros utilizados en el escenario de pruebas y sus resultados se encuentran en la siguiente sección.

6.4.4. Escenario de Prueba para la Generación de Tráfico

El escenario de pruebas para el algoritmo de generación de tráfico considera los parámetros presentados en las *tablas 6.3 y 6.3*. En la primera tabla se encuentran los parámetros que caracterizan el tráfico de los 7 distintos servicios seleccionados en la sección de diseño. Estos parámetros fueron determinados a partir de la *tabla 5.2*. Los demás parámetros necesarios para la simulación se encuentran en la segunda tabla, y corresponden al Tiempo de corte de la simulación y al diferencial ΔT . Este diferencial corresponde al avance de tiempo que tiene la simulación en cada ciclo. Los valores se seleccionaron como 3600 y 1 segundos respectivamente, debido a que las tasas de nacimiento de paquete y de alarma se encuentran a lo mas en el orden unas cuantas por minuto, de manera que no hay necesidad en tener una definición mayor a 1 segundo. Algunas tasas de nacimiento de paquetes y alarmas se encuentran en el orden de unas cuantas por hora como es el caso de los dispositivos de detección de terremotos, razón por la cual se decidió realizar la simulación a lo largo de 3600 segundos.

Los resultados de la simulación se presentan en las siguientes gráficas las cuales muestran los histogramas de arribos para algunos servicios junto con de las distribuciones que estos siguen, además de algunos mapas de calor que muestran el comportamiento de la propagación espacial de las alarmas.

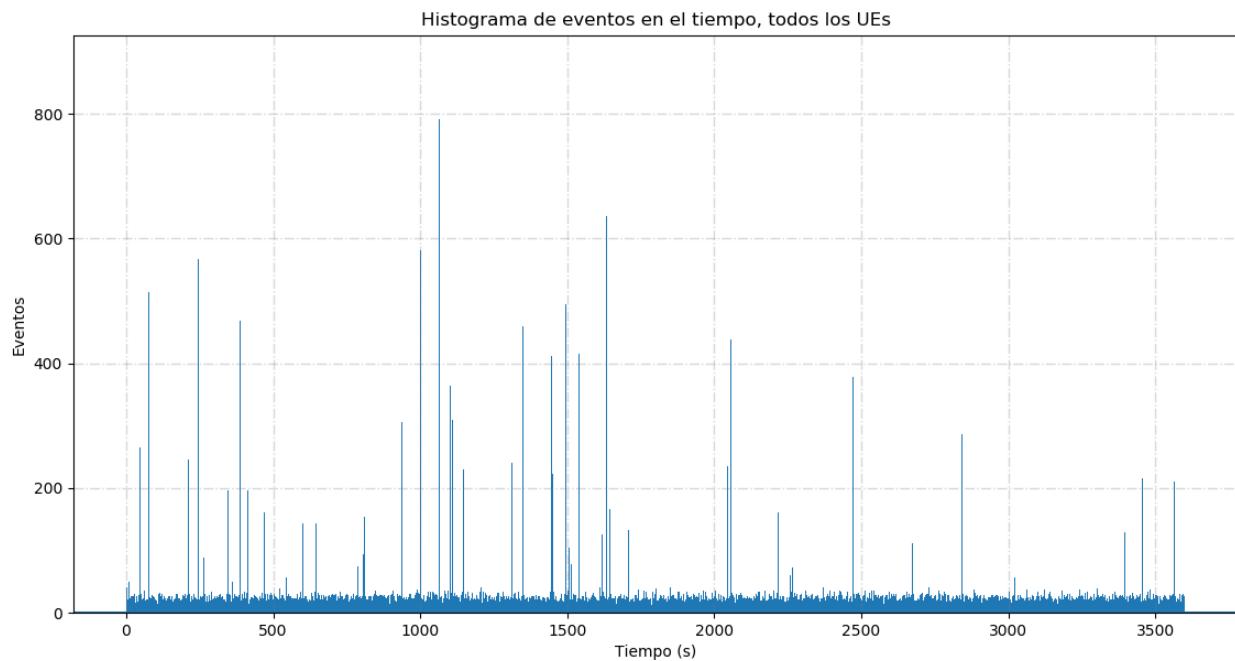


FIGURA 6.10: Histograma de nacimiento de paquetes

La *figura 6.10* muestra el histograma de nacimientos a lo largo de toda la simulación, el ancho de las barras es el mismo que ΔT . En este histograma se contempla el tráfico generado por los

TABLA 6.3: Parámetros para el escenario de pruebas de tráfico

<p>Tipo: Control de Iluminación Cantidad de dispositivos: 1000 Modelo de tráfico: CMMPP Tasa/Frecuencia de nacimiento de paquetes: 0.000556 <i>paquetes/seg.</i> Tasa de nacimiento de alarmas: 0.002778 <i>alarmas/seg.</i> Velocidad de alarma: 500 <i>metros/seg.</i> Modelo de propagación espacial: <i>Raised-cosine window</i> W: 200 d_{th}: 100</p>
<p>Tipo: Monitoreo del consumo de agua y electricidad Cantidad de dispositivos: 500 Modelo de tráfico: Periódico Tasa/Frecuencia de nacimiento de paquetes: 0.001667 <i>paquetes/seg.</i></p>
<p>Tipo: Detección de terremotos Cantidad de dispositivos: 500 Modelo de tráfico: CMMPP Tasa/Frecuencia de nacimiento de paquetes: 0.000278 <i>paquetes/seg.</i> Tasa de nacimiento de alarmas: 0.002778 <i>alarmas/seg.</i> Velocidad de alarma: 3000 <i>metros/seg.</i> Modelo de propagación espacial: <i>Decaying exponential</i> α: 0.4</p>
<p>Tipo: Monitoreo de contaminación del aire Cantidad de dispositivos: 500 Modelo de tráfico: Periódico Tasa/Frecuencia de nacimiento de paquetes: 0.001111 <i>paquetes/seg.</i></p>
<p>Tipo: Control dinámico de semáforos Cantidad de dispositivos: 1000 Modelo de tráfico: Periódico Tasa/Frecuencia de nacimiento de paquetes: 0.016667 <i>paquetes/seg.</i></p>
<p>Tipo: Otros dispositivos mMTC Cantidad de dispositivos: 3000 Modelo de tráfico: CMMPP Tasa/Frecuencia de nacimiento de paquetes: 0.016667 <i>paquetes/seg.</i> Tasa de nacimiento de alarmas: 0.033333 <i>alarmas/seg.</i> Velocidad de alarma: 500 <i>metros/seg.</i> Modelo de propagación espacial: <i>Decaying exponential</i> α: 0.007</p>

TABLA 6.4: Parámetros para el escenario de pruebas de tráfico

Tipo: Dispositivos URLLC
Cantidad de dispositivos: 1000
Modelo de tráfico: CMMPP
Tasa/Frecuencia de nacimiento de paquetes: 0.016667 paquetes / seg.
Tasa de nacimiento de alarmas: 0.033333 alarmas / seg.
Velocidad de alarma: 500 metros / seg.
Modelo de propagación espacial: Decaying exponential
α : 0.007
Tiempo límite de simulación: 3600 segundos
ΔT : 1 segundo
Radio de la célula: 200 metros

7 distintos servicios, de manera que es difícil extraer mucha información. A eso ayudan más las *figuras 6.11 y 6.12*, las cuales se concentran en el tráfico de un sólo servicio, es decir de todos los dispositivos de un mismo tipo.

La *figura 6.11* contiene a la izquierda el histograma de nacimiento de paquetes junto con el nacimiento de las alarmas para los dispositivos del tipo Otros dispositivos mMTC. Las gráficas a la derecha por otra parte son más fáciles de interpretar y se trata de la distribuciones que sigue el tiempo entre nacimiento de paquetes y de alarmas, las cuales podemos ver siguen una distribución exponencial a la tasas establecidas como parámetros en la *tabla 6.3*, justo como se esperaba.

La *figura 6.12* es muy parecida a la *figura 6.11* puesto que presenta las mismas gráficas, los histogramas a la izquierda y las distribuciones al derecho. Pero en esta ocasión el servicio que observamos es el control de semáforos. Este servicio genera tráfico con el modelo periódico, razón por la cual el tiempo entre nacimiento de paquetes es constante y no sigue una distribución exponencial.

Finalmente en las *figuras 6.13 y 6.14* se ven los mapas de calor producidos por una alarma que se transmite siguiendo el modelo *Raised-cosine window* y *Decaying exponential* respectivamente. Se puede notar cómo la propagación de la alarma siguiendo el modelo *Raised-cosine window* tiene lugar de manera uniforme por dentro del círculo de radio dth y con origen en su lugar de su nacimiento, para después desvanecerse y terminar abruptamente para radios mayores al ancho de la ventana . Por otro lado la alarma que sigue el modelo de propagación exponencial se desvanece desde el origen y su efecto decae según crezca la distancia (de manera exponencial), pero no se detiene abruptamente.

Los resultados que presentan estas imágenes muestran que el algoritmo de generación de tráfico fuente está efectivamente generando arribos siguiendo los parámetros y modelos seleccionados, de manera que de aquí se procedió a su implementación en el programa **Generador de tráfico**

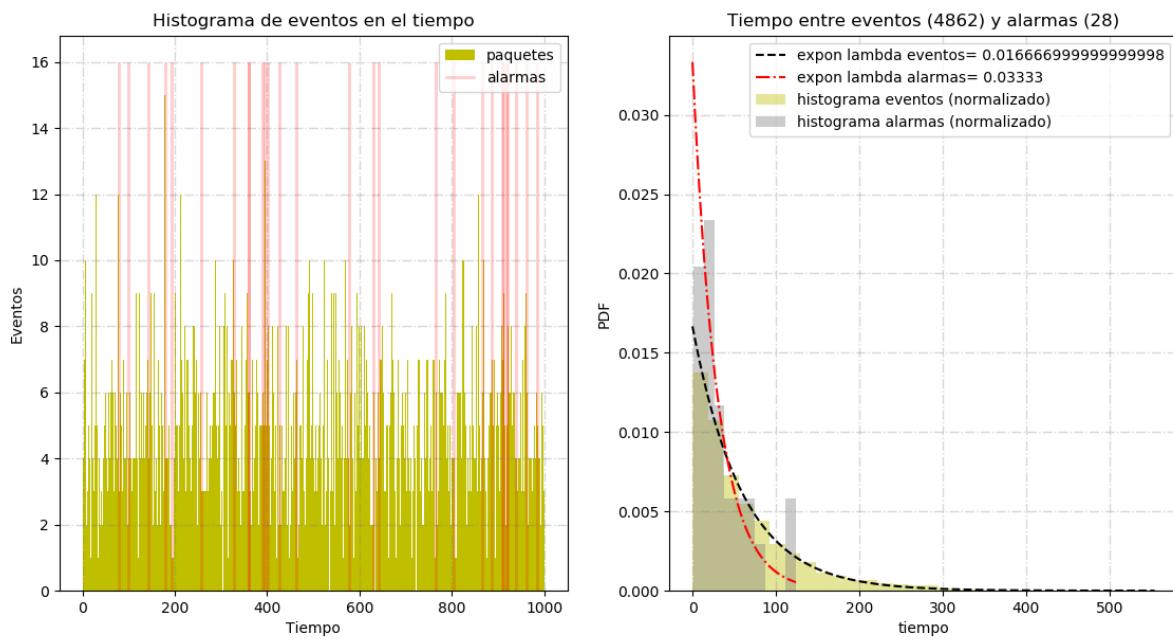


FIGURA 6.11: Otros dispositivos mMTC, Histograma de nacimientos y sus distribuciones

IoT que se trata en la siguiente sección.

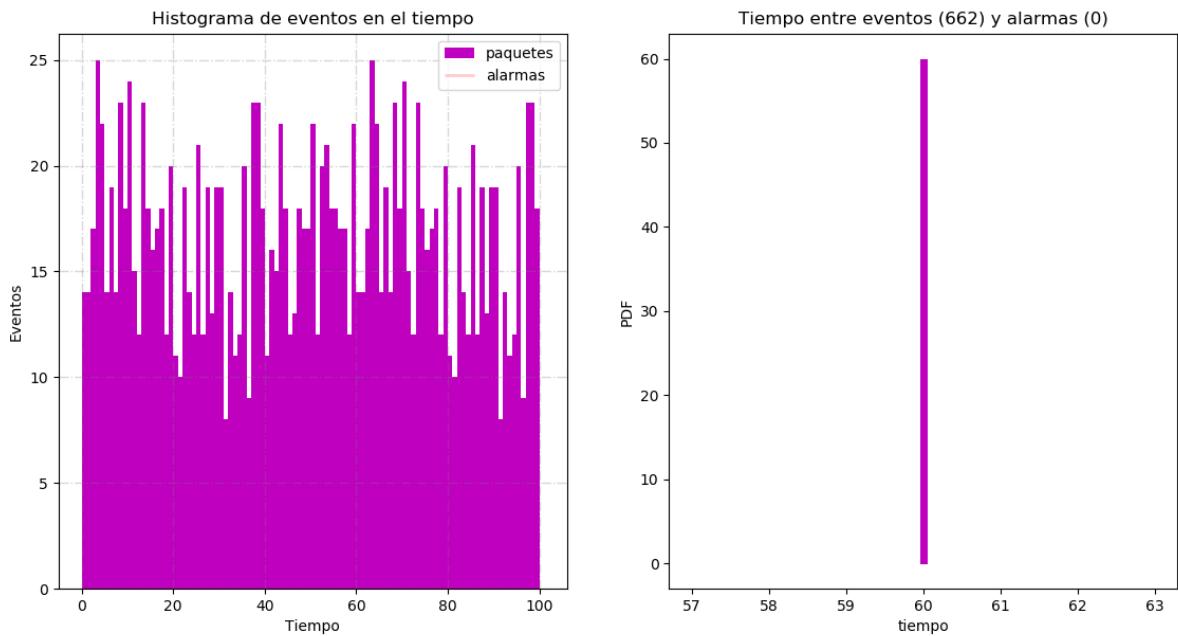


FIGURA 6.12: Semáforos inteligentes, Histograma de nacimientos y sus distribuciones

6.5. Interconexión de los 4 módulos del Simulador

Los 4 módulos del sistema, cuya implementación se ha discutido en la primera parte de este capítulo son: el despliegue de los dispositivos, el modelo del canal, el esquema de acceso múltiple al medio y los modelos de tráfico IoT. La forma en la que se llevó a cabo la interconexión de estos módulos se encuentra descrita en el diagrama de la figura 6.15.

El simulador se encuentra entonces dividido en 2 programas, el **Generador de Tráfico IoT** y el **Simulador de eventos discretos**. La razón por la que se implementó esta arquitectura tiene que ver con los paradigmas que ambos programas siguen, ya que mientras el Generador de tráfico IoT tiene una rutina que realiza avances constantes en el tiempo, el Simulador de eventos discretos avanza en el tiempo apartir de los eventos evaluados.

Ambos sistemas fueron programados en Python 3.6 y fueron implementados de manera modular. Es decir el programa Generador de Tráfico IoT es una herramienta funcional y modificable por sí sola para generar tráfico de tipo máquina con los modelos CMMPP y Periódico y con usuarios distribuidos a partir de un PPP. Los resultados de este programa podrían entonces ser utilizados en algún otro, realizando los ajustes de interfaz necesarios. De la misma manera el Simulador de Eventos discretos que contiene el modelo del canal y el esquema de acceso múltiple al medio, podría procesar tráfico y usuarios generados y distribuidos en un programa distinto al

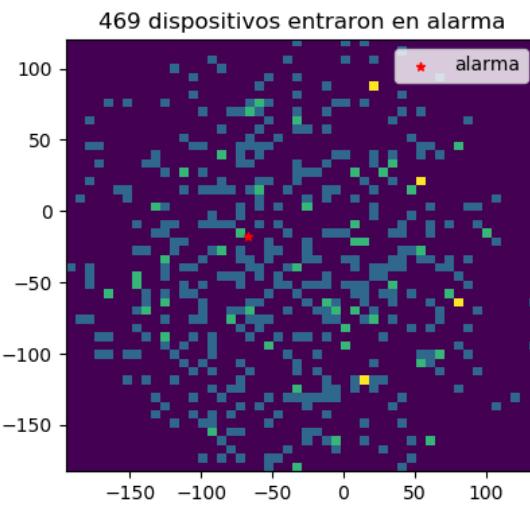


FIGURA 6.13: Propagación de alarma en dispositivos de control de iluminación

presentado en este proyecto. Esta modularidad es posible en gran parte gracias a que la interfaz existente entre ambos programas es el uso de archivos .csv, los cuales son fáciles de generar y modificar.

En las siguientes secciones se aborda con más detalle la implementación de los dos programas que integran el simulador.

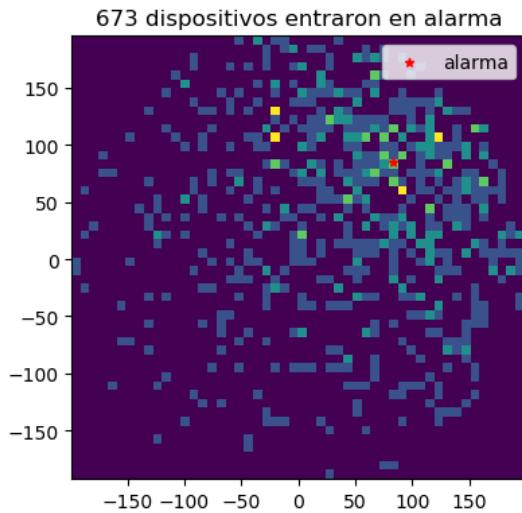


FIGURA 6.14: Propagación de alarma en otros dispositivos mMTC

6.6. Generador de tráfico IoT

El Generador de tráfico IoT distribuye dispositivos IoT en una célula circular siguiendo un PPP, para después generar tráfico de estos dispositivos con los modelos CMMPP y Periódico. La rutina principal del programa sigue el paradigma *time-driven*, puesto que realiza avances constantes en el tiempo. La salida del programa se da en forma de archivos .csv, de los cuales los dos más importantes son el archivo de dispositivos y el archivo de eventos. El archivo de dispositivos contiene información referente a cada uno de los dispositivos generados mientras que el archivo de eventos contiene la información del nacimientos de cada paquete.

La forma en que python se generaron y manipularon archivos .csv en python 3.6 fue con la librería *pandas*, la cual permite mapear listas a archivos .csv y viceversa.

Aunado a la funcionalidad principal del programa, se programó una interfaz de usuario para interactuar con el sistema e ingresar los parámetros de la simulación. La interfaz de usuario se realizó con la librería *tkinter*. Esta interfaz se presenta con mayor detalle en la siguiente sección.

6.6.1. Interfaz de Usuario

La figura 6.16 es una captura de pantalla de la interfaz de usuario del Generador de Tráfico IoT. En el primer bloque arriba a la izquierda se encuentran parámetros referentes a la distribución de usuarios, como son el radio del área circular en la que se distribuirán y su modelo de distribución. Los siguientes 7 bloques sirven para introducir los parámetros de generación de tráfico para los 7 distintos servicios contemplados. Finalmente se debe indicar hasta abajo: el

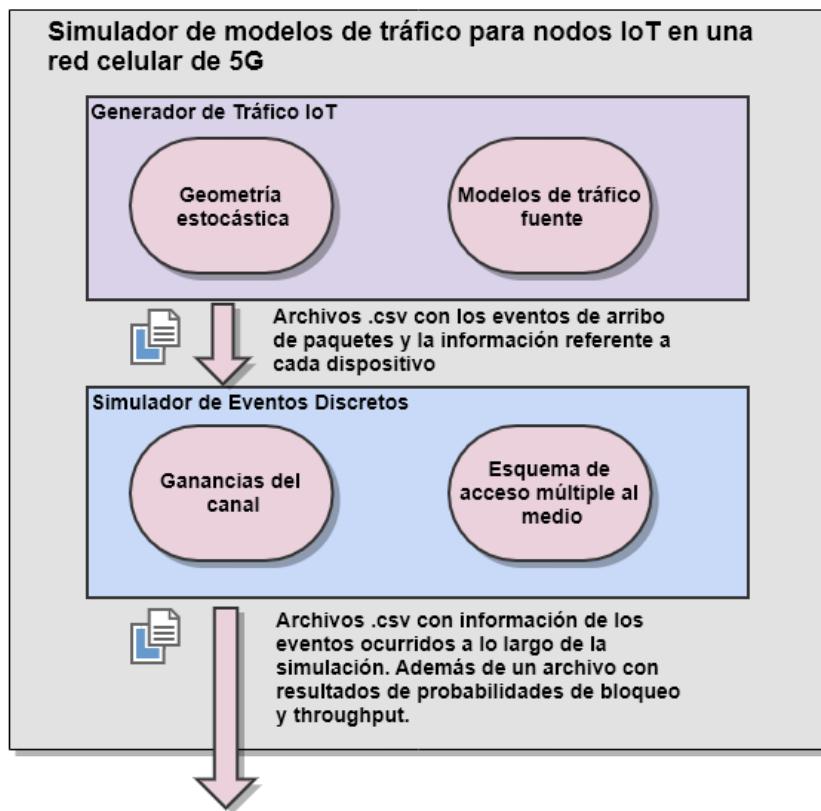


FIGURA 6.15: Interconexión de los 4 módulos en un simulador

tiempo de corte de la simulación y el diferencial de tiempo entre ciclos antes de poder iniciar la rutina. El botón de cargar datos abajo a la derecha permite leer los parámetros de un archivo y desplegarlos en las casillas correspondientes, lo que permite guardar una configuración inicial y acelerar la introducción de valores.

Al terminar la rutina se generarán en la misma carpeta del programa 4 archivos .csv de los cuales 2 son utilizados en el **Simulador de eventos discretos**. Estos archivos son descritos con más detalle en la siguiente sección.

6.6.2. Archivos de Salida

Los principales archivos que resultan de ejecutar cada rutina del Generador de Tráfico son el archivo de dispositivos y el archivo de eventos. En la figura 6.17 se puede ver un ejemplo del primer archivo, en él se guarda: a qué tipo de dispositivo corresponde cada servicio y su posición en el plano. Estos datos son después utilizados en el Simulador de eventos discretos para instanciar nodos del tipo especificado en esa posición y poder adjudicarles sus nacimientos de paquetes. La figura 6.18 muestra un ejemplo de archivo de eventos, en el cual encontramos los

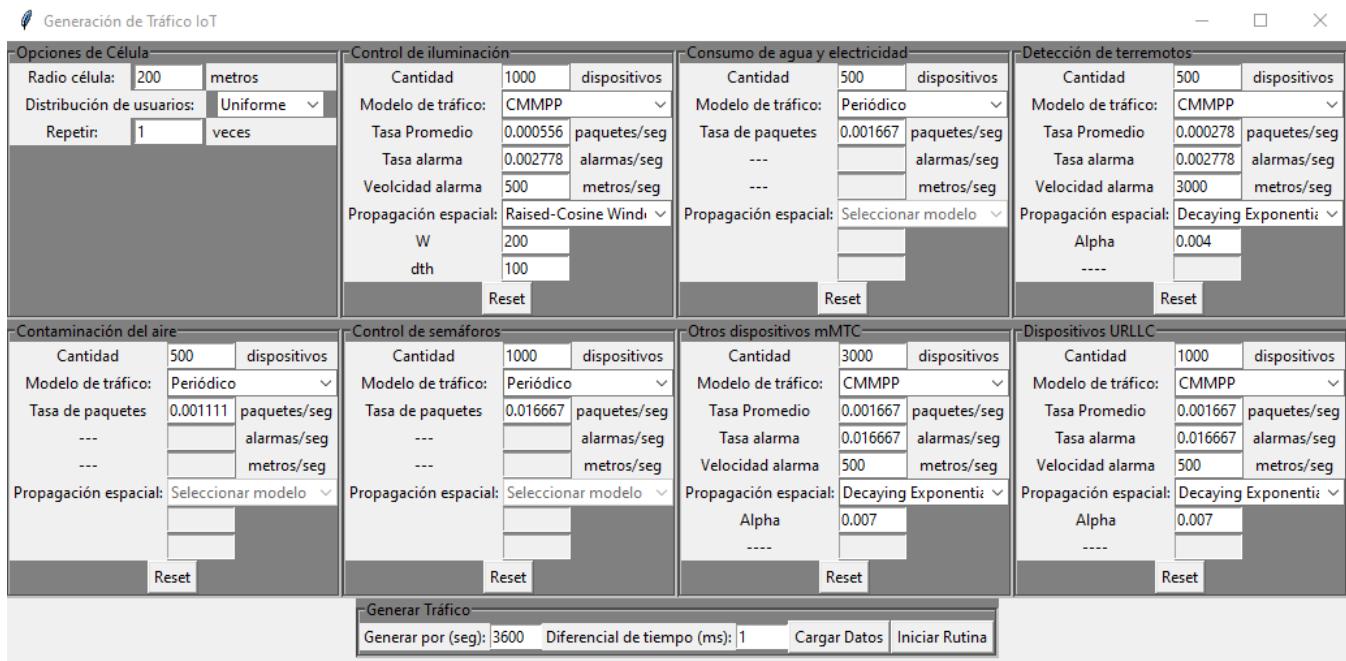


FIGURA 6.16: Interfaz de Usuario, Generador de Tráfico IoT

nacimientos de paquetes a lo largo del tiempo de simulación. Desde tal archivo se podrá conocer el dispositivo que originó el paquete, su tamaño y si este surgió apartir de una alarma, todos estos datos importantes para que el simulador pueda procesar los eventos.

id dispositivo	tipo de dispositivo	posición x	posición y
1	Control de iluminacion	1.159265717	119.880951
2	Monitoreo de agua y electricidad	-54.08404078	-27.6700029
3	Monitoreo de agua y electricidad	155.9103528	15.8279243
4	Deteccion de terremotos	33.95728836	-123.066625
5	Contaminacion del aire	117.6325011	-140.520496
6	Contaminacion del aire	103.7747744	0.96012387
7	Semaforos inteligentes	22.23262869	148.757204
8	Otros dispositivos mMTC	14.55895415	4.39320858
9	Otros dispositivos mMTC	16.81418758	-169.945927
10	Dispositivos URLLC	91.47651632	-99.7301979

FIGURA 6.17: Archivo de dispositivos

Los últimos dos archivos resultados de cada rutina de este programa son el de configuración y el de alarmas. Pero estos no son utilizados por el Simulador de eventos, sino para generar gráficas de la distribución de usuarios y los modelos de tráfico y de distribución de alarmas.

id alarma	tiempo	id dispositivo	tipo de dispositivo	paquete alarma	paquete (bytes)	mod. tráfico
0	4.593	8	Otros dispositivos mMTC	0	24.68	0
1	4.7873	9	Otros dispositivos mMTC	1	20	0
3	5.0534	4	Detección de terremotos	1	20	0
0	5.1027	1	Control de iluminacion	0	20.57	0
0	5.6052	1	Control de iluminacion	0	21.06	0
0	6.4651	1	Control de iluminacion	0	20.13	0
0	8.1003	6	Contaminacion del aire	0	108.38	1
0	9.7237	2	Monitoreo de agua y electricidad	0	20.53	1
0	9.7799	1	Control de iluminacion	0	21.95	0
2	9.8411	8	Otros dispositivos mMTC	1	20	0
0	9.8947	10	Dispositivos URLLC	0	20.06	0

FIGURA 6.18: Archivo de eventos

6.7. Simulador de Eventos Discretos

El Simulador de eventos discretos se trata del otro programa además del Generador de tráfico IoT que conforma al **Simulador de modelos de tráfico para nodos IoT en una red de 5G**. Fue también programado en python 3.6 y sigue el paradigma de eventos discretos. La Inicialización del simulador se hace a través de un archivo de configuración .json que permite modificar varios parámetros de la simulación sin necesidad de alterar el código.

La funcionalidad básica del programa se puede observar en el diagrama de la *figura 6.19*. En el diagrama se puede ver que cuando se toma el siguiente evento en la cola, este es procesado y es aquí donde la mayor parte de la lógica del simulador recide. En la siguiente sección se describen los distintos eventos y cómo son procesados.

Para realizar la funcionalidad de eventos discretos en python se utilizó la librería *heapq*, la cual permite la utilización de *heaps*, que son arboles binarios, para los cuales cada nodo padre tiene un valor menor o igual a cualquiera de sus hijos. De tal manera es posible mantener una lista que siga esta estructura con el calendario de eventos (el arbol se ordena teniendo en cuenta el tiempo en el que ocurre el evento) y utilizar las funciones *heappush* y *heappop* para agregar nuevos eventos y removerlos del arbol respectivamente. Esta estructura asegura que el primer evento de la lista sea siempre el siguiente a ser evaluado y la librería representa una manera fácil de interactuar con la lista.

A la salida del simulador, se tiene un archivo .csv con un historial de *logs* correspondiente a los eventos que fueron procesados además de otro archivo con el resultado de las probabilidades de bloqueo por falta de cluster, el porcentaje de paquetes que fueron transmitidos a la tasa deseada y el *throughput* del sistema. Los eventos que tienen lugar y son procesados por el sistema se presentan en la siguiente sección.

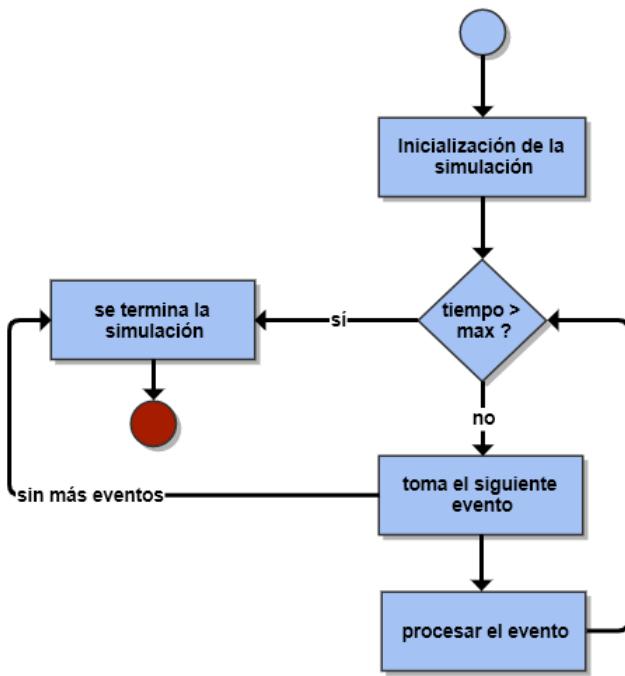


FIGURA 6.19: Simulación de eventos discretos

6.7.1. Definición de eventos

El primero de los posibles eventos que el simulador procesa es el nacimiento de un nuevo paquete. Este evento tiene una correspondencia entera con el archivo de eventos (*figura 6.18*) puesto que cada linea en ese archivo resulta en un evento de nacimiento en este Simulador. La *figura 6.20* muestra la manera en que se procesa el evento, de ella se puede ver que si el dispositivo está disponible para transmitir se calendará un proceso noma en el que posiblemente se le asignará un conjunto de recursos y comenzará a transmisiones su paquete.

El siguiente posible evento es el de proceso noma. La *figura 6.21* muestra cómo se procesa en el simulador y se puede observar que se realiza el algoritmo noma (*algoritmo 2*) con los dispositivos que están haciendo uso del canal y aquellos que quieren transmitir. Los dispositivos que tengan la suerte de contar con un cluster (un conjunto de recursos) comenzarán a transmitir y crearán o actualizarán su evento de fin de transmisión, teniendo en cuenta su nueva tasa de transmisión. Aquellos dispositivos que no consigan recursos cancelarán sus eventos de final de transmisión, y sus paquetes serán considerados en la estadística de bloqueos por falta de recursos.

El último evento se trata del fin de transmisión, en este evento un dispositivo termina de transmitir todo su paquete y un nuevo proceso noma es calendarizado para redistribuir los recursos. El dispositivo que recién termina de transmitir vuelve al estado *idle* si no tiene paquetes en la cola, de lo contrario transmite el siguiente. Es aquí también cuando se calcula la tasa efectiva a la que se transmitió el paquete haciendo la resta entre el momento en que se comenzó a

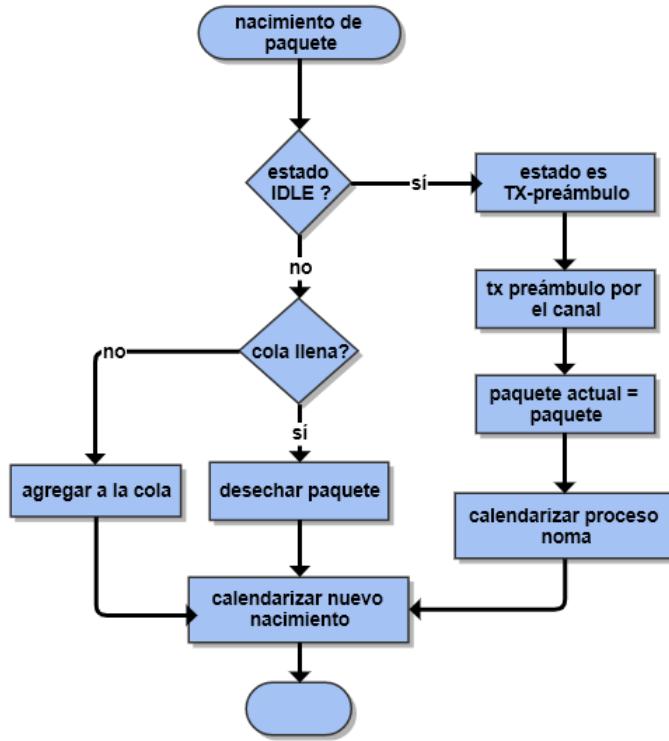


FIGURA 6.20: Procesamiento del evento: nacimiento de paquete

transmitir y el tiempo actual de la simulación. Esta tasa efectiva es considerada al calcular el porcentaje de paquetes que fueron transmitidos a las tasas requeridas.

La condición de paro del simulador es que se llegue al tiempo límite de la simulación o que se terminen los eventos en la cola. Cuando pase cualquiera de esas dos cosas la simulación termina y un archivo .csv con *logs* de los eventos es generado utilizando la librería *pandas*. Este archivo es descrito en la siguiente sección.

6.7.2. Descripción de los *logs* de salida

La figura 6.22 muestra una porción del archivo de *logs* resultante del Simulador de eventos discretos. En él se puede ver cuándo y donde ocurrieron los eventos. Por ejemplo en el instante de tiempo 0 hubo el nacimiento de 5 paquetes provenientes de distintos dispositivos, cada uno de estos nacimientos propició que los dispositivos contendieran por recursos del canal, razón por la cual todos enviaron un preámbulo. Un instante de tiempo después se realiza el proceso NOMA con estos 5 dispositivos (no había más dispositivos utilizando el canal) y aquellos dispositivos que consiguieron recursos comienzan a transmitir. En el evento nuevo cluster cada dispositivo que consiguió recursos imprime a qué cluster pertenece y a qué tasa transmite. Finalmente el último evento que se ve en la imagen muestra cuantos dispositivos están transmitiendo a una

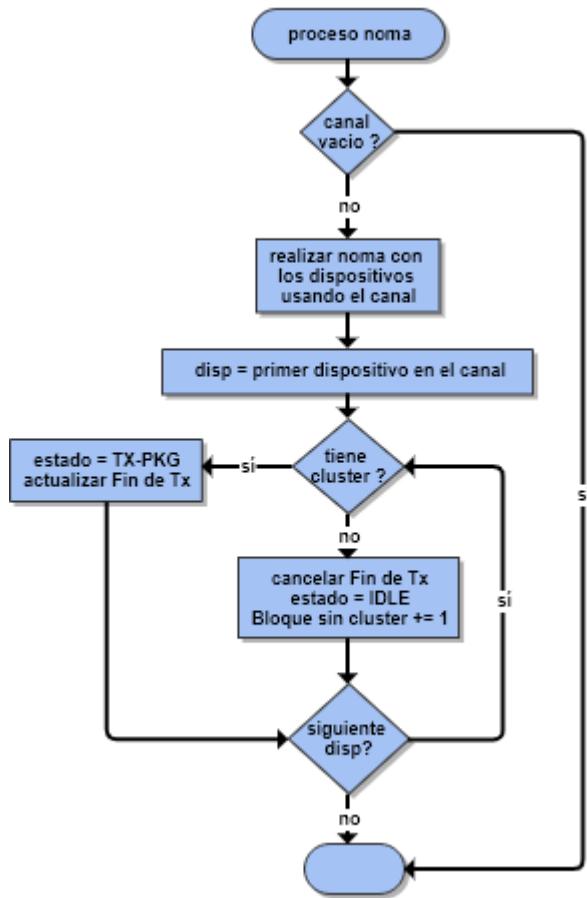


FIGURA 6.21: Procesamiento del evento: proceso noma

tasa menor de la que requieren.

Con este archivo es posible ver cómo reaccionó el sistema en momentos específicos de la simulación. Con ayuda del archivo de eventos se puede analizar el comportamiento del sistema en distintas situaciones, como cuando ocurre una alarma y de pronto hay muchos nacimientos de paquetes o para ver cuánto tiempo después de estas el sistema se estabiliza por ejemplo.

6.7.3. Resultados a la salida del simulador

A la salida del simulador se obtiene además del archivo con los *logs* de los eventos que se evalúan: El tráfico ofrecido, el *throughput* del sistema, la probabilidad de bloqueo por negación de servicio (por falta de recursos, que ocurre cuando un dispositivo no alcanza lugar en algún cluster), y el porcentaje de tasas no cubiertas tanto para dispositivos mMTC como para dispositivos URLLC.

tiempo	fuente	tipo	destino	tipo	evento	descripcion	tam/estado	detalles
0	553	Monitoreo d		553	Monitoreo d	3 paquete generado	20 bytes	
0	553	Monitoreo d		0 eNB		6 nuevo estatus	6	PREAMBULO
0	3211	Otros dispositos		3211	Otros dispositos	3 paquete generado	20 bytes	
0	3211	Otros dispositos		0 eNB		6 nuevo estatus	6	PREAMBULO
0	4538	Dispositivos		4538	Dispositivos	3 paquete generado	20 bytes	
0	4538	Dispositivos		0 eNB		6 nuevo estatus	6	PREAMBULO
0	4600	Dispositivos		4600	Dispositivos	3 paquete generado	23 bytes	
0	4600	Dispositivos		0 eNB		6 nuevo estatus	6	PREAMBULO
0	5153	Dispositivos		5153	Dispositivos	3 paquete generado	20 bytes	
0	5153	Dispositivos		0 eNB		6 nuevo estatus	6	PREAMBULO
0.0001		0 eNB		0 eNB		9 NOMA inicio	3 URLLC	2 mMTC
0.0001		0 eNB		553	Monitoreo d	11 Nuevo cluster	5	314 bytes/s
0.0001	553	Monitoreo d		553	Monitoreo d	6 nuevo estatus	1	TX-PKG
0.0001	0 eNB		3211	Otros dispositos		11 Nuevo cluster	4	2130 bytes/s
0.0001	3211	Otros dispositos		3211	Otros dispositos	6 nuevo estatus	1	TX-PKG
0.0001	0 eNB		4538	Dispositivos		11 Nuevo cluster	2	919 bytes/s
0.0001	4538	Dispositivos		4538	Dispositivos	6 nuevo estatus	1	TX-PKG
0.0001	0 eNB		4600	Dispositivos		11 Nuevo cluster	1	3297 bytes/s
0.0001	4600	Dispositivos		4600	Dispositivos	6 nuevo estatus	1	TX-PKG
0.0001	0 eNB		5153	Dispositivos		11 Nuevo cluster	3	369 bytes/s
0.0001	5153	Dispositivos		5153	Dispositivos	6 nuevo estatus	1	TX-PKG
0.0001	0 eNB		0 eNB			10 Tasas insatisfechas	1 URLLC	0 mMTC

FIGURA 6.22: *logs* resultantes de una rutina

6.8. Optimización de tiempos de simulación

La generación de distintas variables aleatorias y el uso de algoritmos de asignación de recursos nos obligaron a pensar que los tiempos de simulación probablemente llegarían a ser excesivamente largos, por tal se propuso utilizar una técnica de paralelismo en python.

La técnica de paralelización se implementó pero sólamente en la generación de resultados de NOMA para un TTI ya que en esta parte fue que se necesitaron obtener resultados promedio donde cada punto o muestra de los datos necesitaba simularse 1000 repeticiones para así obtener un promedio. En la Figura 6.23 se observa la comparación de los tiempos de simulación en un caso normal (solamente 1 núcleo en uso) e implementando la paralelización (uso de todos los núcleos de la PC).

Cabe mencionar que esta simulación se montó en una PC con procesador Intel(R) Core(TM) i7-7700HQ CPU @2.80 GHz y 16 GB de Memoria RAM.

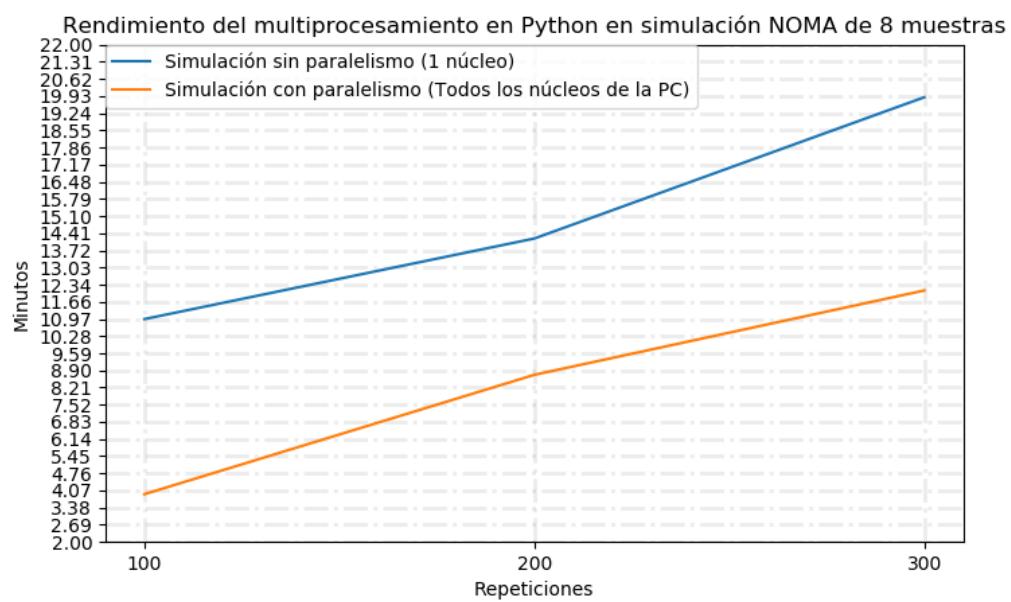


FIGURA 6.23: Gráfica de comparación de tiempos de simulación

Capítulo 7

Resultados

El objetivo de este capítulo fue el de brindar resultados para distintos escenarios de interés, con el fin de dar comparaciones analíticas con base en la variación de los parámetros de entrada del simulador.

7.1. Escenario I

7.1.1. Descripción del escenario

Este escenario se concentró en obtener resultados para un solo intervalo de tiempo de transmisión (TTI), con el fin de analizar el rendimiento del modelo de despliegue de UE y el modelo de canal en conjunto con NOMA y evaluar su rendimiento. Se simuló el rendimiento de *multitone* con diferentes clases de potencia para los dispositivos MTC y con diferentes tamaños de grupos (k_{max}) pero su desempeño no resultó ser importante ya que resultaba ser similar al de *singletone*. Por este motivo los resultados con la propuesta *multitone* no se reportaron. Sin embargo, en estos resultados se implementó un modo de operación híbrido donde se adoptó un modo de operación *multitone* solamente cuando el número de dispositivos es menor al número de grupos (48), esto con la finalidad de no desperdiciar recursos. Y *singletone* en los demás casos. También es importante señalar que la relación entre dispositivos mMTC y uRLLC es de 3 a 1.

7.1.2. Parámetros de entrada

De acuerdo con los parámetros generales del modelo de sistema [véase Tabla 5.3], los criterios considerados para este escenario fueron los siguientes:

- $k_{max} \rightarrow 1, 2, 3$ y 4 grupos
- $p_m^s \rightarrow 23, 20, 14$ dBm

7.1.3. Resultados obtenidos

En primera instancia, en el capítulo anterior se analizó el histograma de las pérdidas de canal, del Modelo CI y también del Modelo de canal que proponen en el artículo [37] , de acuerdo con los histogramas se tiene que el valor promedio de las perdidas [dB] en el Modelo CI son de 80.35 dBs. En contrario con las pérdidas del modelo de canal del artículo en [37], tienen un

valor promedio de 74.67 dBs. Es decir nuestro canal tiene 3.7 (5.68 dBs) veces más perdidas en comparación del canal que se implementa en [37], por lo que se espera que el rendimiento sea menor, esto se puede observar en la Figura 7.1 donde se observa que en el caso de 192 usuarios el desempeño de la simulación del artículo es aproximadamente 140 usuarios, es decir 73 % de los usuarios alcanzan su tasa objetivo. Por el contrario en la evaluación de la simulación con el modelo de canal CI en NOMA se obtiene que aproximadamente 115 usuarios alcanzan su tasa objetivo, es decir, un 60 %. Hay un impacto del canal CI de aproximadamente 13 % del desempeño en comparación con el canal propuesto en [37].

Cabe destacar que en [37], el modelo de sistema no es implementado para una banda de frecuencias en específico, las ganancias de las subportadoras son estadísticas. En el caso de este sistema como se ocupa un modelo de canal que depende de la frecuencia, se tuvo que escoger un PRB de 180 KHz, con un conjunto de subportadoras fijadas en una banda LTE, se escogió la banda de 2GHz, esto por el hecho de que LTE se implementa en bandas de microondas.

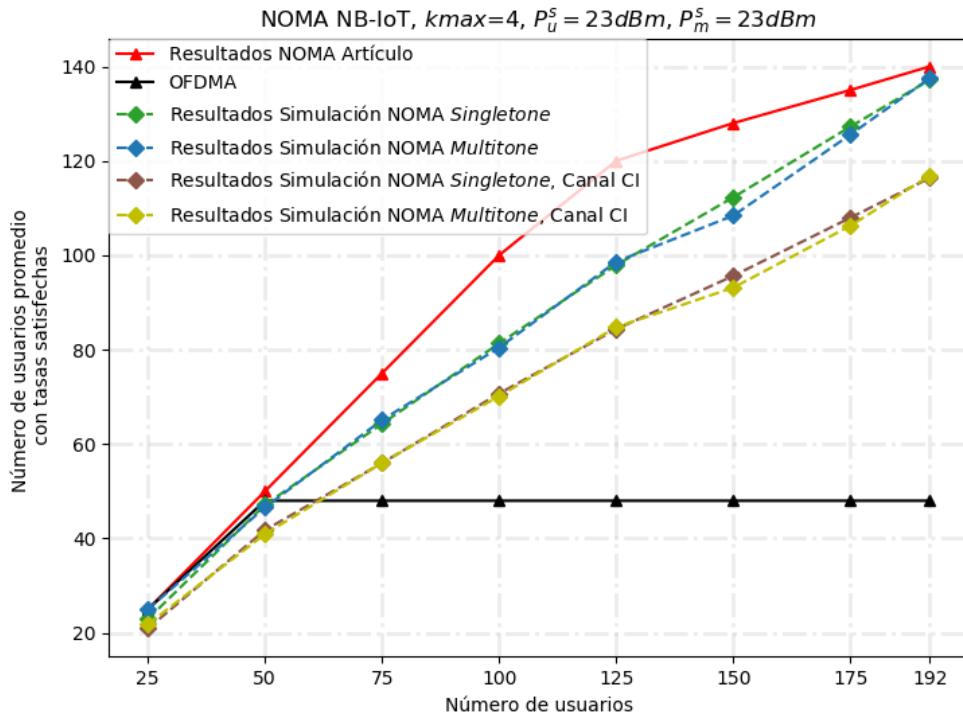


FIGURA 7.1: Modelo NOMA en un TTI con modelo de canal CI

En la Figura 7.2 se evaluó el número de usuarios que alcanzaron su tasa objetivo, se realizaron comparaciones con respecto a tres tipos de clase de potencia para los mMTC y un variable número de dispositivos por grupo (k_{max}).

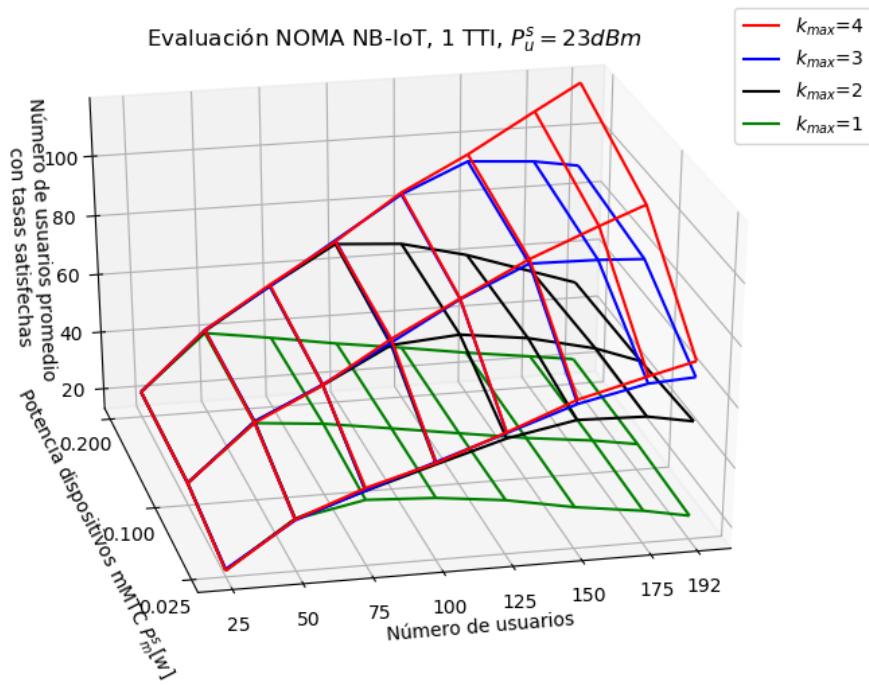


FIGURA 7.2: Resultados generales Escenario I

Empezando con una agrupación de 4 dispositivos, se puede observar que entre menor es la potencia de los dispositivos mMTC, el rendimiento de usuarios que alcanzan su tasa objetivo va decayendo y esto es por que al bajar su potencia los dispositivos mMTC varios de ellos comienzan a tener dificultades para alcanzar su tasa objetivo. Si se analiza el caso de 192 usuarios el desempeño con una potencia de usuarios MTC de 23dBm es de aproximadamente 115 usuarios que alcanzan su tasa objetivo, es decir, un 60 %. Y cuando la potencia de usuarios mMTC de 14dBm, 73 dispositivos alcanzan su tasa objetivo, un 38 %. Es decir el rendimiento decrece un 22 % de una potencia de 23 a 14 dBm.

Con una agrupación de 3 dispositivos, vemos que el rendimiento en general decae cuando son 150 usuarios y es porque en este caso, el máximo de usuarios que pueden ser atendidos es de 144, por lo que en los casos de 175 y 192 usuarios el rendimiento va bajando esto es por la relación 3 a 1 que se propuso en los parámetros de entrada. También conforme se baja la potencia de los dispositivos mMTC, se puede ver que el rendimiento decae. Si se analiza el caso de 150 usuarios el desempeño con una potencia de usuarios MTC de 23dBm es de aproximadamente 89 usuarios que alcanzan su tasa objetivo, es decir, un 59 %. Y cuando la potencia de usuarios mMTC de 14dBm, 67 dispositivos alcanzan su tasa objetivo, un 44 %. Es decir el rendimiento decrece un 15 % de una potencia de 23 a 14 dBm.

Con una agrupación de 2 dispositivos, se observa que el rendimiento en general decae cuando son 100 usuarios y es porque en este caso, el máximo de usuarios que pueden ser atendidos es de 96, por lo que en los casos mayores a 100 usuarios el rendimiento va bajando esto igualmente es por la relación 3 a 1. De igual manera también se puede ver que el rendimiento decae cuando se baja la potencia de los dispositivos mMTC. Si se analiza el caso de 100 usuarios el desempeño con una potencia de usuarios MTC de 23dBm es de aproximadamente 53 usuarios que alcanzan su tasa objetivo, es decir, un 53 %. Y cuando la potencia de usuarios mMTC de 14dBm, 49 dispositivos alcanzan su tasa objetivo, un 49 %. Es decir el rendimiento decrece solamente 4 % de una potencia de 23 a 14 dBm.

Con una agrupación de 1 dispositivo, de la misma forma se observa que el rendimiento en general decae cuando son 50 usuarios y es porque en este caso, el máximo de usuarios que pueden ser atendidos es de 48, por lo que en los casos mayores a 100 usuarios el rendimiento va bajando esto igualmente es por la relación 3 a 1. También conforme se baja la potencia de los dispositivos mMTC, se puede ver que el rendimiento no decae de manera significativa como lo fue en los otros casos.

En las siguientes figuras se evaluó la misma métrica del número de dispositivos que alcanzan su tasa objetivo pero este caso considerando cuántos de estos dispositivos son uRLLC y cuántos mMTC.

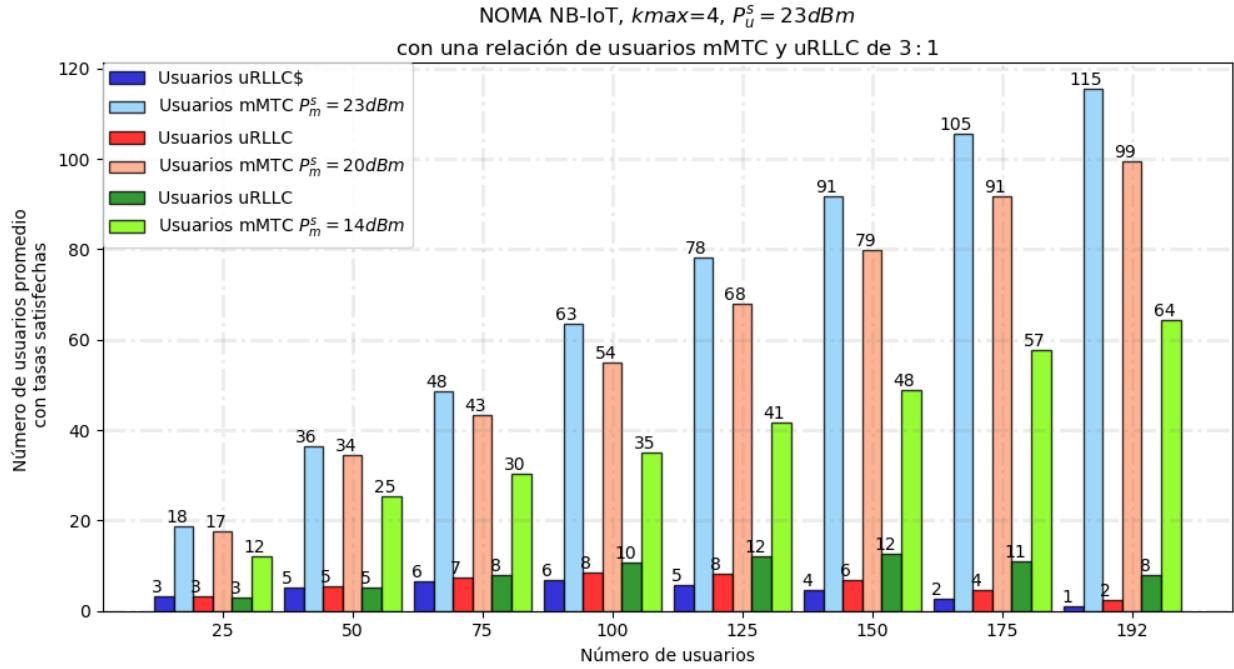


FIGURA 7.3: Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo, kmax 4

La Figura 7.3 muestra la comparación del número de dispositivos mMTC y uRLLC que alcanzaron su tasa objetivo en un TTI, esto con una agrupación de 4 dispositivos (i.e. 192 dispositivos como máximo), se realizaron comparaciones con diferentes potencias de los dispositivos mMTC. Igualmente, en la Figura 7.4 se representan evaluaciones acerca del número de dispositivos mMTC y uRLLC que alcanzaron su tasa objetivo, pero esta vez mostrando el porcentaje de dispositivos mMTC y uRLLC que alcanzan su tasa objetivo, de acuerdo con la relación 3 a 1 que se planteó en los parámetros de entrada.

Primeramente, con una potencia de 23dBm para los dispositivos mMTC (color azul), se observa que entre mayor sea el número de dispositivos, el porcentaje de dispositivos uRLLC que alcanzan su tasa va disminuyendo, esto se puede ver más claramente en la Figura ???. Por ejemplo, cuando son 25 dispositivos el porcentaje de uRLLC y mMTC es de 51 %- 98 % respectivamente y cuando el número de dispositivos aumenta a 192, el porcentaje de uRLLC y mMTC es de 2 %- 80 % respectivamente (esto es con base en la relación 3 a 1). Como se observa la relación entre uRLLC y mMTC que alcanzan su tasa es muy desproporcional, esto debido a que al transmitir con la misma potencia todos los dispositivos, la contribución de interferencia de los dispositivos

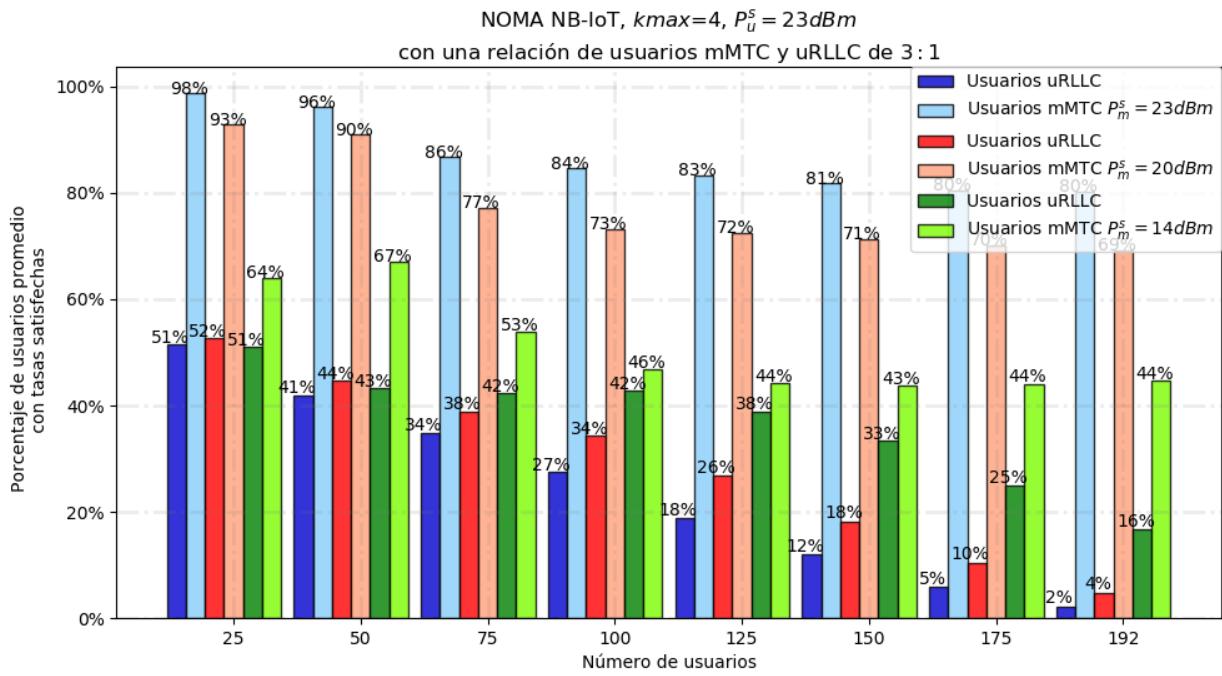


FIGURA 7.4: Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo(%), $k_{max} 4$

mMTC (en rangos altos) para los uRLLC, es bastante alta, impidiendo que los uRLLC no alcancen sus tasas objetivo.

En los casos en donde la potencia de los dispositivos mMTC es menor a la de los uRLLC, se observa una mejor proporción entre los dispositivos. Por ejemplo, con una potencia de 14dBm para los dispositivos mMTC (color verde) el porcentaje de dispositivos uRLLC y mMTC es de 51 %- 64 % respectivamente y cuando el número de dispositivos aumenta a 192, el porcentaje de dispositivos uRLLC y mMTC es de 16 %- 44 % respectivamente (esto es con base en la relación 3 a 1). Como se observa la relación de uRLLC y mMTC que alcanzan su tasa es mejor comparado con el análisis de potencia de los dispositivos mMTC con 23 dBm.

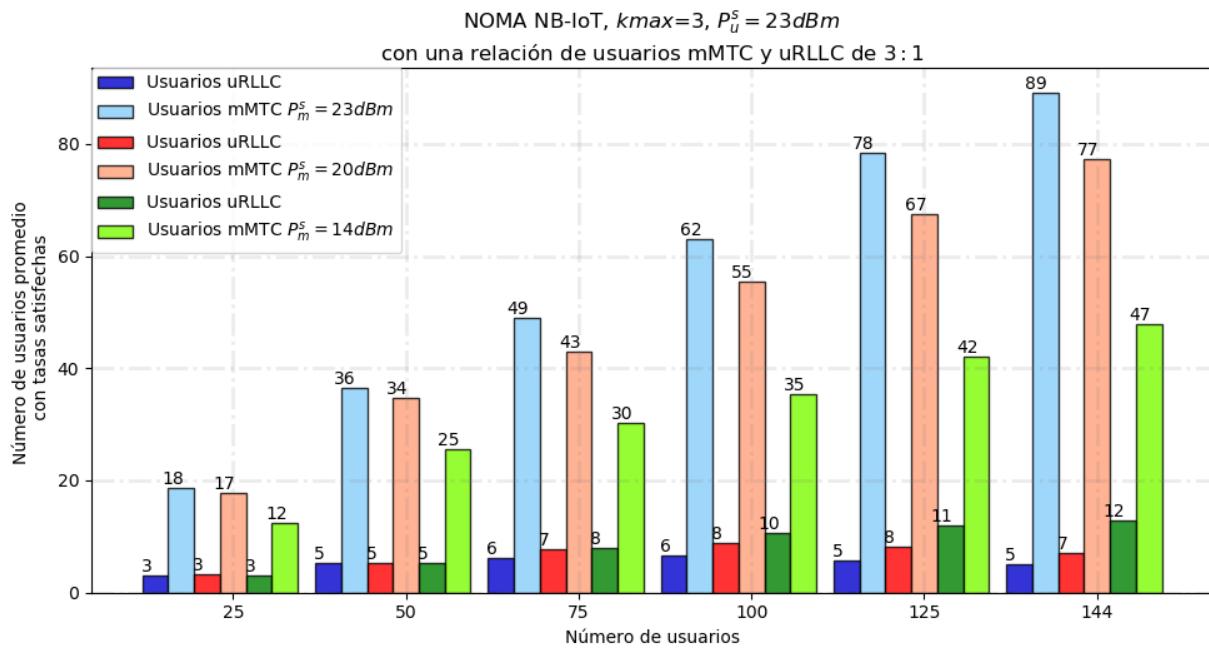


FIGURA 7.5: Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo, $k_{max} 3$

Las Figuras 7.5 y 7.6 representan la relación entre el número de dispositivos mMTC y uRLLC que alcanzaron su tasa objetivo en un TTI, esto con una agrupación de 3 dispositivos (i.e. 144 dispositivos como máximo), se realizaron comparaciones con diferentes potencias de los dispositivos mMTC.

Primeramente, con una potencia de 23dBm para los dispositivos mMTC (color azul), se observa que entre mayor sea el número de dispositivos, la relación de dispositivos uRLLC y mMTC que alcanzan su tasa se hace más desproporcional, esto se puede ver más claramente en la Figura 7.6. Por ejemplo, cuando son 25 dispositivos el porcentaje de dispositivos uRLLC y mMTC es de 52 %- 98 % respectivamente y cuando el número de dispositivos aumenta a 144, el porcentaje de dispositivos uRLLC y mMTC es de 14 %- 82 % respectivamente (esto es con base en la relación 3 a 1). Como se observa la relación de uRLLC y mMTC que alcanzan su tasa sigue siendo desproporcional pero no tanto comparado con agrupaciones de 4 dispositivos, esto es porque son menos los dispositivos agrupados.

En los casos en donde la potencia de los dispositivos mMTC es menor a la de los uRLLC, se observó de la misma manera se obtiene una mejor proporción entre los dispositivos. Por ejemplo, con una potencia de 14dBm para los dispositivos mMTC (color verde) el porcentaje de dispositivos uRLLC y mMTC es de 51 %- 64 % respectivamente y cuando el número de dispositivos aumenta a 144, el porcentaje de dispositivos uRLLC y mMTC es de 35 %- 44 % respectivamente (esto es con base en la relación 3 a 1).

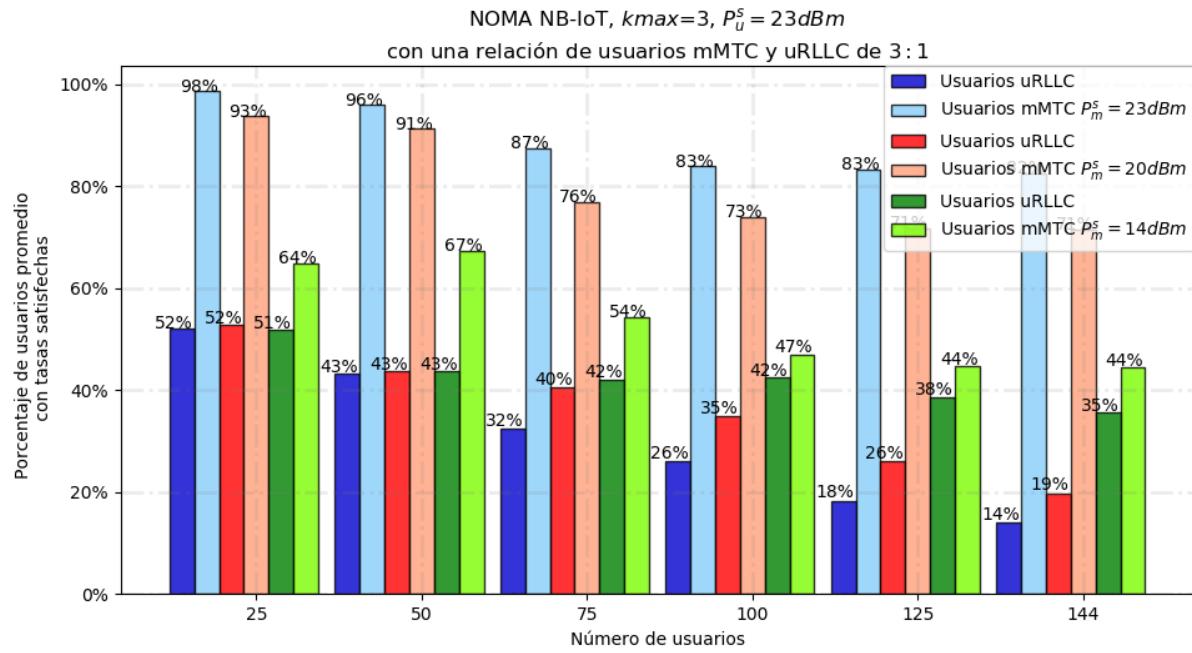


FIGURA 7.6: Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo(%), $k_{max} 3$

Las Figuras 7.7 y 7.8 representan la relación entre el número de dispositivos mMTC y uRLLC que alcanzaron su tasa objetivo en un TTI, esto con una agrupación de 2 dispositivos (i.e. 96 dispositivos como máximo), se realizaron comparaciones con diferentes potencias de los dispositivos mMTC.

Primeramente, con una potencia de 23dBm para los dispositivos mMTC (color azul), se observa que entre mayor sea el número de dispositivos, la relación de dispositivos uRLLC y mMTC que alcanzan su tasa se hace más desproporcional, esto se puede ver más claramente en la Figura 7.8. Por ejemplo, cuando son 25 dispositivos, los porcentajes de dispositivos uRLLC y mMTC es de 52 %- 98 % respectivamente y cuando el número de dispositivos aumenta a 96 dispositivos, el porcentaje de dispositivos uRLLC y mMTC es de 28 %- 85 % respectivamente (esto es con base en la relación 3 a 1).

En los casos en donde la potencia de los dispositivos mMTC es menor a la de los uRLLC, se observó la relación entre los dispositivos alcanzan una mejor proporción. Por ejemplo, Con una potencia de 14dBm para los dispositivos mMTC (color verde) el porcentaje de dispositivos uRLLC y mMTC es de 52 %- 64 % respectivamente y cuando el número de dispositivos aumenta a 144 dispositivos, el porcentaje de dispositivos uRLLC y mMTC es de 41 %- 47 % respectivamente (esto es con base en la relación 3 a 1).

Para finalizar esta sección, se pudo observar que de la Figura 7.2, que entre menor sea la

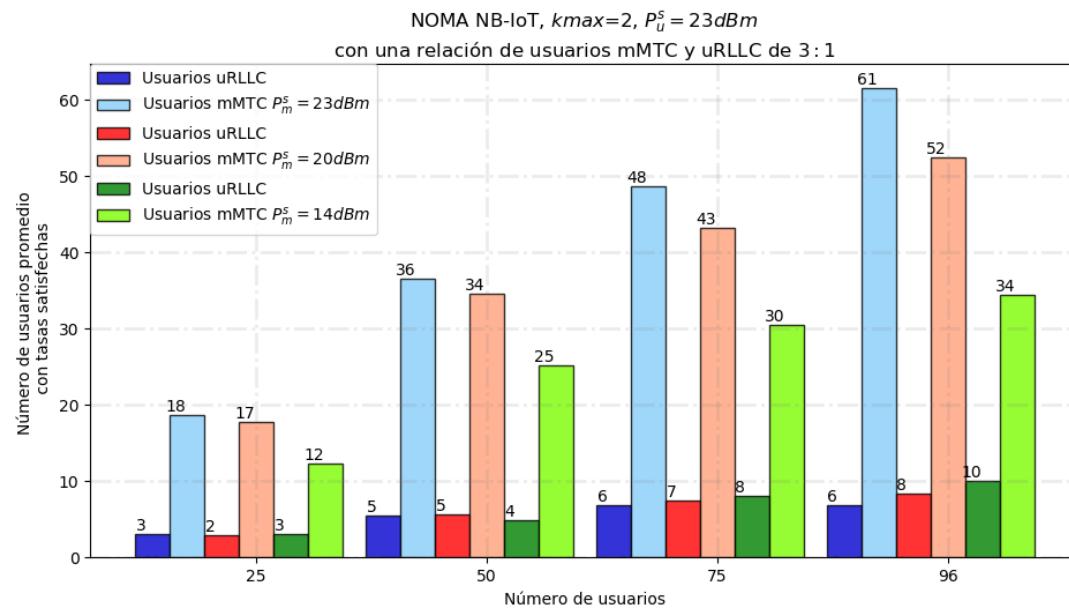


FIGURA 7.7: Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo, $k_{max} 2$

potencia de transmisión de los dispositivos mMTC, el número de dispositivos que alcanzan su tasa objetivo disminuirá, pero al hacer el análisis de las gráficas de la relación de dispositivos uRLLC y mMTC, se concluye, que aunque disminuyen los dispositivos que alcanzan su tasa, mejora equitativamente la proporción entre los dispositivos uRLLC y mMTC que logran su tasa objetivo.

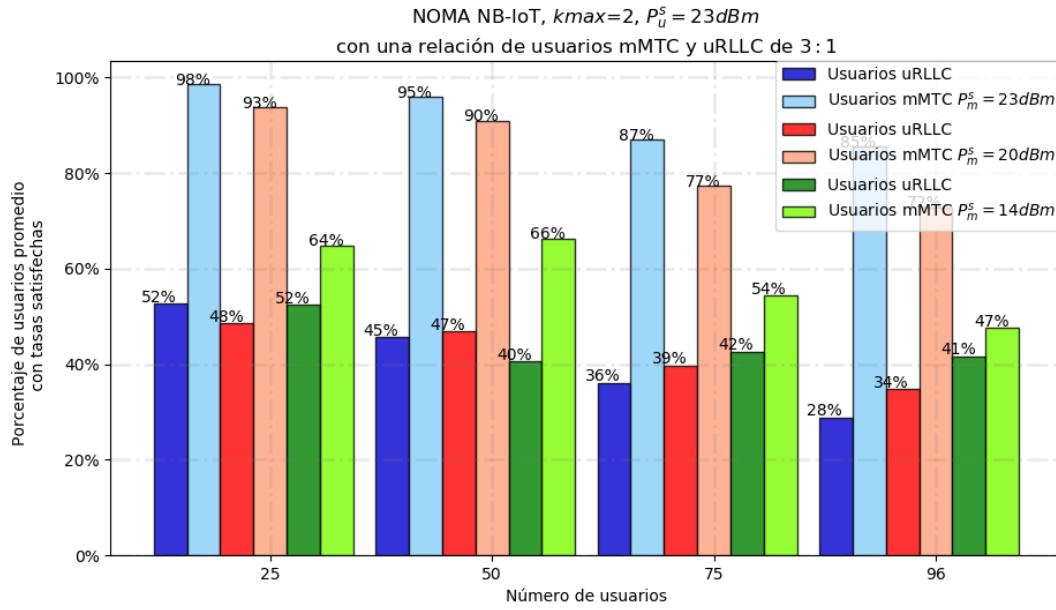


FIGURA 7.8: Relación de usuarios uRLLC y mMTC que alcanzan su tasa objetivo (%), $k_{max} 2$

7.2. Escenario II

Este segundo escenario pone a prueba la capacidad que tiene el programa principal **Simulador de modelos de tráfico para nodos IoT en una red celular de 5G**, de generar resultados que permitan comparar el efecto que tiene establecer distintos parámetros de la red en el *throughput* del sistema y en el porcentaje de dispositivos que cumplen su tasa deseada.

7.2.1. Descripción del escenario

En este escenario se consideraron únicamente los dispositivos de tipo URLLC y los llamados Otros dispositivos mMTC, esto para facilitar el cumplir con la relación de 3 a 1 entre ambos tipos de dispositivos. Entonces en el programa **Generador de tráfico IoT** se generó tráfico dentro de una célula de radio igual a 200 metros. La distribución de usuarios se hizo con la opción PPP y se establecieron las intensidades de la siguiente forma: La intensidad de los dispositivos mMTC se fijo en $0.3 \text{ dispositivos}/m^2$ y la de los dispositivos URLLC se fijo en $0.1 \text{ dispositivos}/m^2$. La relación de dispositivos se eligió 3 a 1 para después fijar las tasas de nacimientos de paquetes y de alarmas idénticas en ambos tipos de servicios y garantizar que el tráfico ofrecido al sistema conserve esa relación. La tasa de nacimiento de paquetes es $\lambda_{normal} = 0,0167 \text{paquetes/seg.}$ y la de nacimiento de alarmas es $\lambda_{alarm} = 0,1 \text{alarmas/seg.}$ para ambos tipos de dispositivos. Finalmente, las características en las que se transmiten las

alarmas son compartidas también entre ambos dispositivos: $velocidad_{alarm} = 500m/s$ y modelo de propagación espacial = *Rised-cosine window* con $d_{th} = 200$ y $d_{th} = 100$.

De manera que virtualmente, ambos tipos los dispositivos generan tráfico del mismo tipo, pero al haber 3 veces más dispositivos mMTC, en los algoritmos NOMA se conservará en promedio esta relación.

El tráfico generado por el **Generador de tráfico IoT** correspondió a 10 segundos y la iteración seleccionada para ser evaluada por el simulador de eventos discretos contenía 2 alarmas, una para cada tipo de dispositivo, lo que es el promedio esperado, dado que $\lambda_{alarm} = 0,1alarmas/seg..$ Se seleccionó esta iteración por ser una buena representación de los parámetros ingresados. Finalmente se inició la simulación con 48 dispositivos ya utilizando el canal, 26 dispositivos mMTC y 12 URLLC, esto para que el sistema se encontrara ya en un equilibrio de operación.

7.2.2. Parámetros de entrada

La instancia de tráfico utilizada como entrada del simulador de eventos discretos, comprendía 37541 dispositivos mMTC y 12611 dispositivos URLLC. El tiempo de la simulación se fijo en 10 segundos y las potencias máximas de transmisión en los dispositivos se fijaron como $23dBm$ para los URLLC y $20dBm$ para los mMTC. Finalmente se usó $d0 = 1m$, $PLE = 2,0$ y el bloque de frecuencias que inicia en $2Ghz$.

Se corrieron 4 rutinas en las que se variaron los valores de k desde 1 hasta 4.

7.2.3. Resultados obtenidos

Para **k=1** se obtuvo:

Tráfico ofrecido = $48182,749345bytes/s$

Throughput = $31578,300434bytes/s$

Probabilidad de bloqueo sin clúster = $36,674849porciento$

Tasas no cubiertas URLLC = $25,670367porciento$

Tasas no cubiertas mMTC = $17,796712porciento$

Para **k=2** se obtuvo:

Tráfico ofrecido = $48182,749345bytes/s$

Throughput = $33391,109178bytes/s$

Probabilidad de bloqueo sin clúster = $32,644499porciento$

Tasas no cubiertas URLLC = $22,322801porciento$

Tasas no cubiertas mMTC = $11,487657porciento$

Para **k=3** se obtuvo:

Tráfico ofrecido = $48182,749345bytes/s$

Throughput = $34457,869843bytes/s$

Probabilidad de bloqueo sin clúster = $30,363758porciento$

Tasas no cubiertas URLLC = $24,560205porciento$

Tasas no cubiertas URLLC = 11,829281*porciento*

Para **k=4** se obtuvo:

Tráfico ofrecido = 48182,749345*bytes/s*

Throughput = 35150,834596*bytes/s*

Probabilidad de bloqueo sin clúster = 28,829673*porciento*

Tasas no cubiertas URLLC = 24,184458*porciento*

Tasas no cubiertas URLLC = 11,427156*porciento*

De los resultados se puede ver que aumentar k también aumenta el *throughput* y reduce la probabilidad de bloqueo por negación de servicio sin comprometer el porcentaje de tasas no cubiertas, pero variar la potencia de transmisión y evaluar más iteraciones de tráfico son necesarias para dar una respuesta definitiva, sin embargo se puede ver que el simulador genera los datos requeridos, y apartir de estos se puede identificar el comportamiento del sistema para distintos parámetros de entrada.

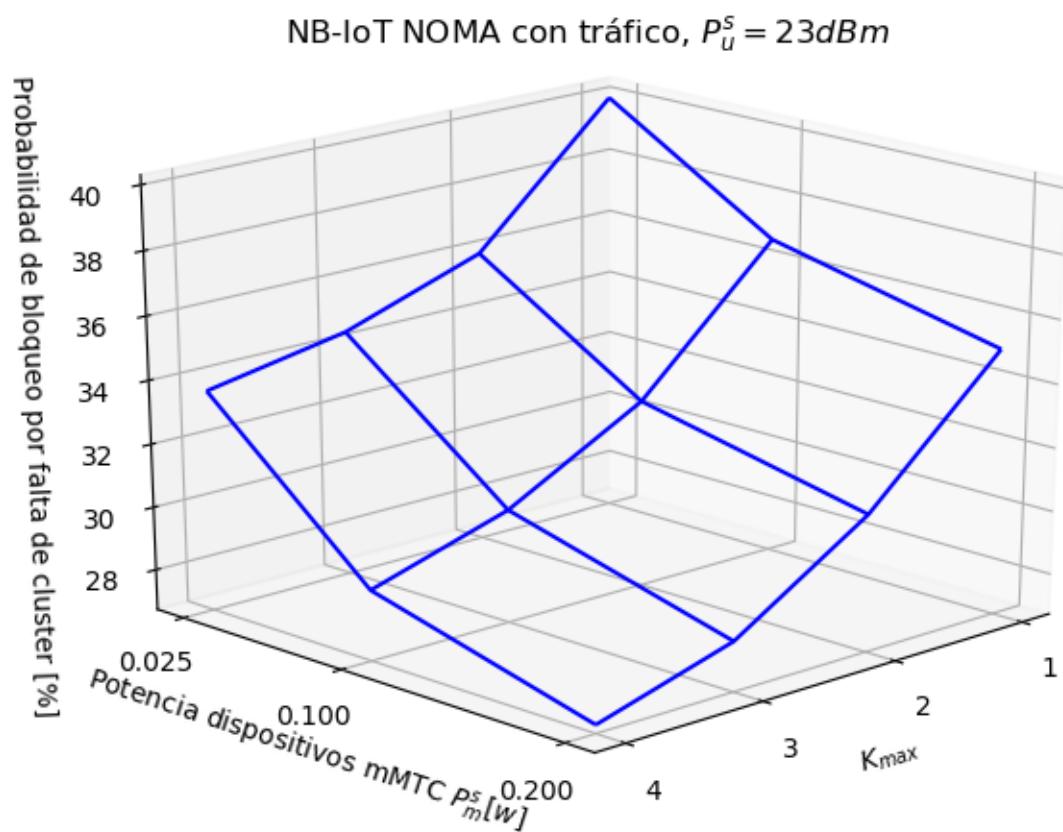


FIGURA 7.9

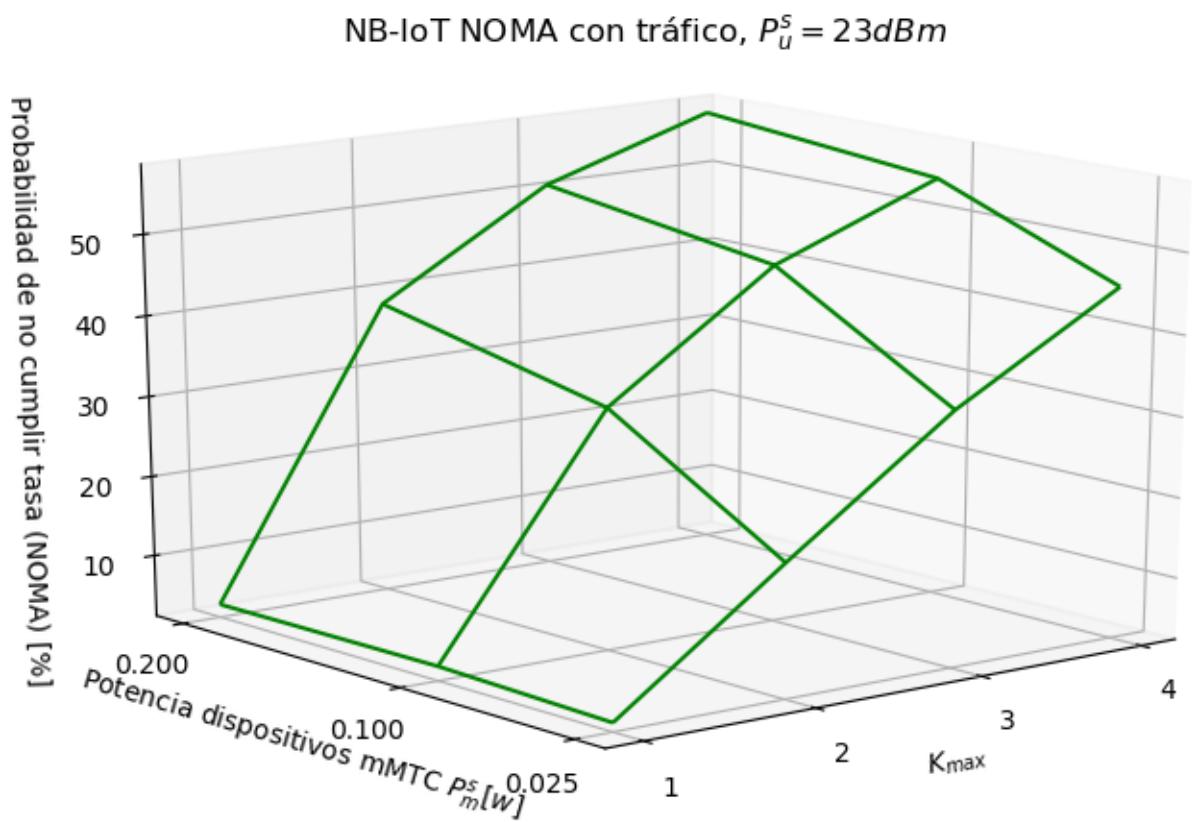


FIGURA 7.10

NB-IoT NOMA con tráfico, $P_u^S = 23dBm$

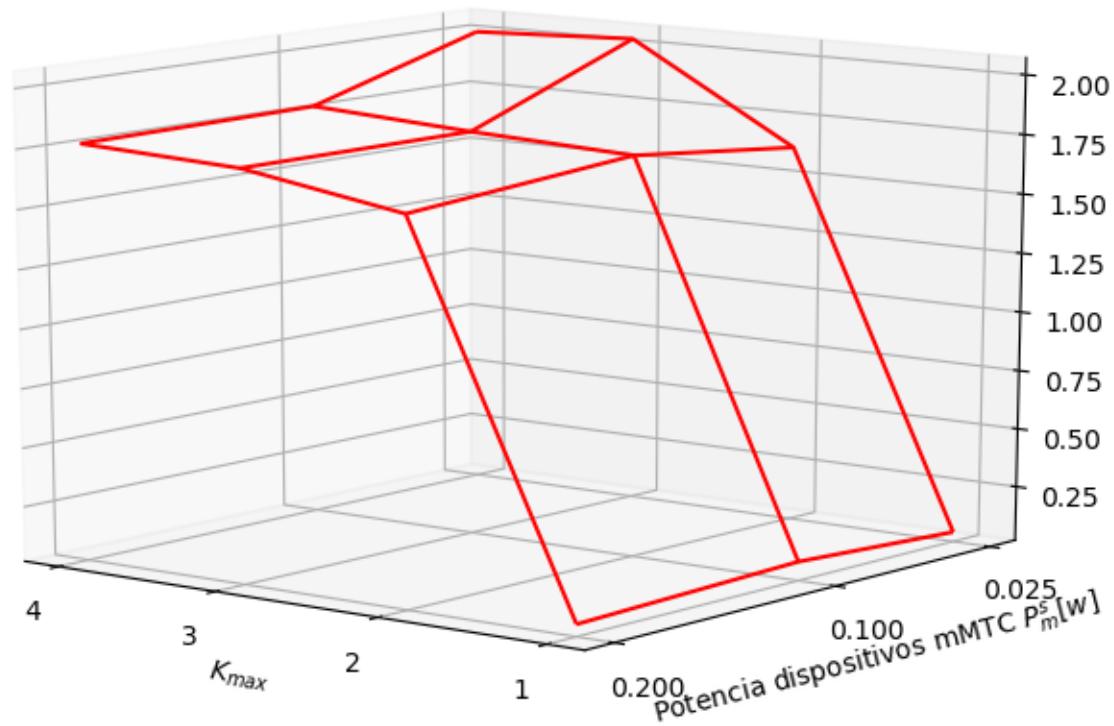


FIGURA 7.11

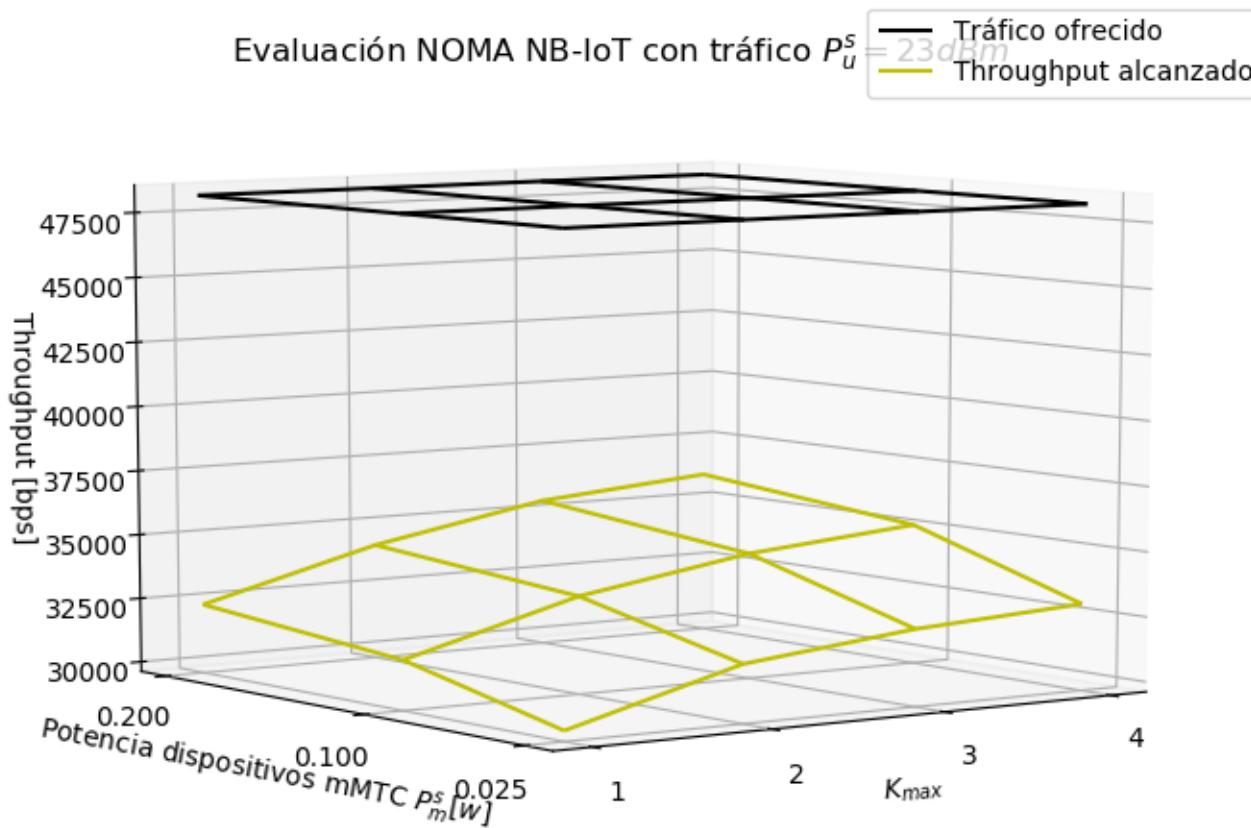


FIGURA 7.12

Capítulo 8

Conclusiones

En este capítulo se concluye el presente proyecto, compilando una discusión y un análisis crítico de los resultados en general, presentando las posibilidades de evolución futura de las comunicaciones móviles, así como las propuestas de trabajo académico futuro a mediano plazo.

Al analizar todos los elementos que hay en una simulación a nivel de sistema en redes celulares, se llegó a la conclusión que se puede realizar una buena representación modelando principalmente estos cuatro aspectos:

- Modelo de despliegue
- Modelo de canal
- Esquema de acceso múltiple al medio
- Modelos de tráfico

El análisis individual de los modelos sentó las bases para los resultados que se obtuvieron de manera conjunta, en el análisis por separado se obtuvieron las siguientes conclusiones:

- La opción mas realista para el modelado de la localización de usuarios en un plano es por medio de un PPP.
- Tras revisar los modelos de canal de distintos escenarios urbanos se concluyó que un modelo de canal idoneo es considerar pérdidas por trayectoria junto con el desvanecimiento rápido. El modelo CI es un modelo de canal mas realista que predice las pérdidas en un ambiente urbano (UMa). El desvanecimiento tipo Rayleigh, es de suma importancia en anchos de banda pequeños como el de NB-IoT, debido a las variaciones rápidas que pueden ocurrir en el canal.
- Los esquemas no ortogonales han tenido un gran auge en la literatura científica, por el hecho de que consiguen aumentar la capacidad en los sistemas.
- Se llegó a la conclusión de que para el uso de esquemas no ortogonales con dos clases de dispositivos, existe una compensación entre la conectividad de usuarios y la potencia de los usuarios que tienen menos requerimientos de tasas.

- Se optó por escoger un modelo de tráfico fuente CMMPP ya que da una buena representación del tráfico espacial y temporal, que lo caracteriza por ser mas realista.

Es importante mencionar que este proyecto se hicieron algunas suposiciones que difieren de la realidad:

- No se consideraron los efectos de la interferencia intercelular, la que proviene de otras células.
- Se tomó la consideración de que en el escenario por cada tres usuarios tipo MTC hay uno con requerimientos de tasas mas altos URLLC.
- El modelo de la generación de tráfico tipo máquina omitió los estados de ahorro de energía y *sleep* en los que los dispositivos pueden estar.
- no se modeló el movimiento de los dispositivos.
- se obvio la arquitectura del canal de subida NB-IoT.

Dadas las limitaciones enlistadas previamente, se considera como trabajo a futuro a las siguientes líneas de investigación:

- NOMA para NB IoT con mejores estrategias para un modo de operación multitone
- MIMO para esquemas NOMA
- Efectos del control de potencia en esquemas NOMA
- Análisis de tráfico tipo máquina en conjunto con un tráfico tipo humano.
- Una cuestión importante es el consumo de energía de los dispositivos, por lo tanto un acceso eficiente debería ser diseñado para minimizar los altos consumos de energía

Apéndice A

Distribuciones estadísticas en Telecomunicaciones

El objetivo de este apéndice fue revisar las distribuciones de probabilidad mas utilizadas en los sistemas de comunicaciones móviles para caracterizar los fenómenos más importantes en este ámbito, después se describió la implementación y puesta a prueba de la generación de las variables aleatorias utilizadas en el simulador, esto con el fin de brindar fiabilidad en los resultados obtenidos.

El uso de modelos estadísticos es importante para describir diferentes fenomenos en el campo de las telecomunicaciones[13]:

- Llamadas telefónicas y conexiones de datos
- Influencia del usuario en el rendimiento de la red
- Propagación no guiada en ambientes aleatorios
- Movilidad del usuario

Comúnmente se utilizan las siguientes distribuciones de probabilidad en telecomunicaciones [13]:

1. Distribución Uniforme: Es usada para describir la fase de una señal. También, se ha utilizado para simular el despliegue de BSs [14].
2. Distribución Normal (Gaussiana): Es usada para describir fluctuaciones alrededor de un valor medio, p.ej. *shadowing*. Esta distribución no puede ser usada para describir entidades que no pueden ser negativas.
3. Distribución Log-Normal: Es usada para describir entidades como la potencia de una señal, amplitudes, principalmente el desvanecimiento lento.
4. Distribución Rayleigh: Es usada para describir el desvanecimiento rápido-intenso.
5. Distribución Susuki: Describe conjuntamente el desvanecimiento lento y rápido.
6. Distribución Rice: Es usada para describir el desvanecimiento rápido - no-intenso.

7. Distribución Exponencial: Es ampliamente usada para describir la duración de diferentes fenómenos, principalmente asociados con el desvanecimiento de señales y las llamadas telefónicas.
8. Distribución de Bernoulli: Es usada para describir la ocupación de canales de telecomunicaciones.
9. Distribución binomial: Es usada para describir llamadas telefónicas.

A.1. Generación de números aleatorios

La distribución uniforme (también llamada distribución rectangular) es una familia de curvas de dos parámetros que es notable porque tiene una función de distribución de probabilidad constante (PDF) entre sus dos parámetros delimitadores. La distribución uniforme se utiliza en técnicas de generación de números aleatorios, como el método de inversión [52].

Se puede usar la distribución uniforme estándar para generar números aleatorios para cualquier otra distribución continua mediante el método de inversión. El método de inversión se basa en el principio de que las funciones de distribución acumulativa continua (CDFs) varían uniformemente durante el intervalo abierto $(0, 1)$. Si u es un número aleatorio uniforme en $(0, 1)$, entonces $x = F^{-1}(u)$ genera un número aleatorio x a partir de la distribución continua con la CDF especificada F [52].

En teoría de la probabilidad y estadística, hay varias relaciones entre las distribuciones de probabilidad. Estas relaciones se pueden clasificar en los siguientes grupos [53]:

- Una distribución es un caso especial de otra con un espacio de parámetros más amplio.
- Transformaciones (función de una variable aleatoria).
- Combinaciones (función de varias variables).
- Relaciones de aproximación (límite).
- Relaciones compuestas (útiles para la inferencia bayesiana [*Bayesian inference*]).

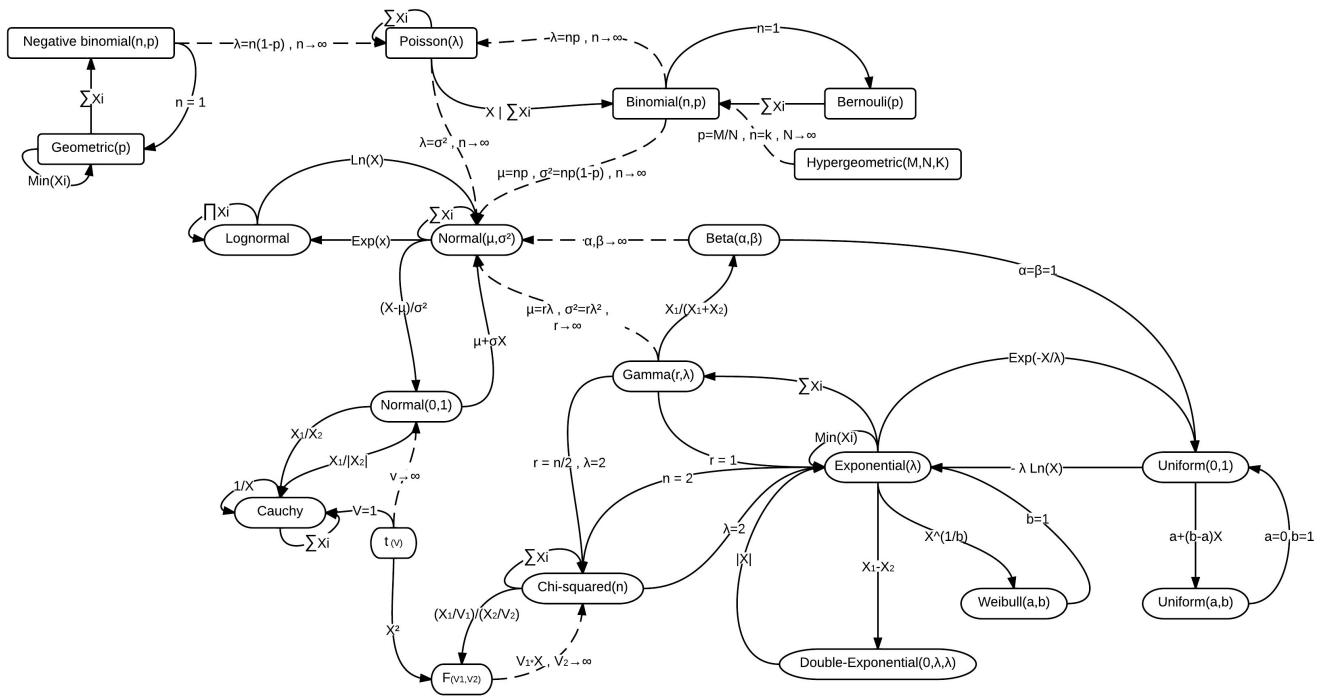


FIGURA A.1: Las relaciones entre algunas de las distribuciones de probabilidad univariadas se ilustran con líneas conectadas, las líneas discontinuas significan relación aproximada. [Fuente: [53]]

A.2. Generación de distribuciones estadísticas para el simulador

A.2.1. Generación de variable aleatoria tipo *Poisson*

Como se revisó en la sección 6.1, uno de los requisitos para generar una geometría estocástica es que el número de puntos en el plano sea Poisson, es por esto que en esta sección se comprobó la generación de la variable aleatoria Poisson en Python usando la librería *scipy*.

Para esto, se realizaron 10000 generaciones de números siguiendo una distribución de Poisson con tasa $\lambda = 20$, se obtuvo el histograma de todos los números generados y se comparó con su función de masa de probabilidad (PMF).

Se observa que la distribución del histograma sigue a la función masa de probabilidad de Poisson [véase Figura A.2], por lo que se valida la generación de números Poisson en Python.

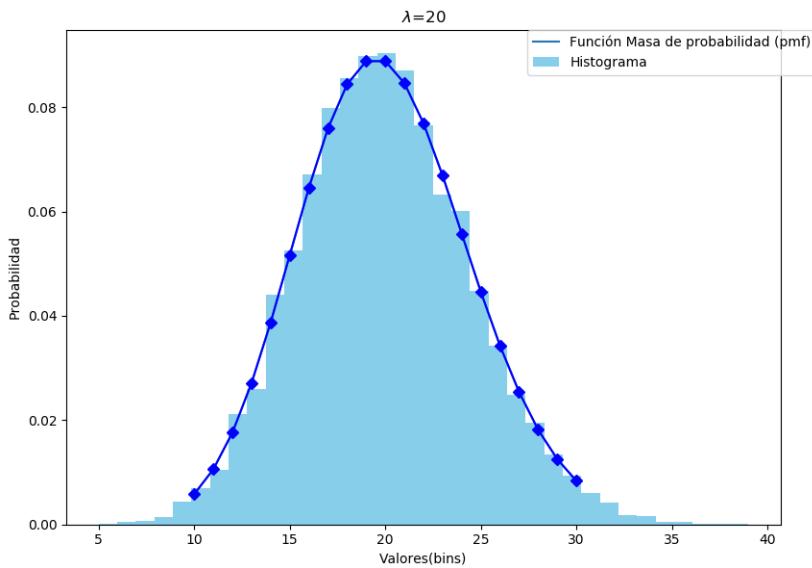


FIGURA A.2: Gráfica comparación PMF e histograma de distribución Poisson en Python

A.2.2. Generación de variables aleatorias tipo *Exponencial* y *Rayleigh*

Como se revisó en la sección 2.2.3, cuando el desvanecimiento es de tipo Rayleigh, la magnitud o amplitud de la señal es Rayleigh y la potencia es exponencial. En esta sección, se comprobó la generación de estas variables aleatorias en Python usando la librería *random*.

Se realizaron 10000 generaciones de numeros siguiendo una distribución Exponencial negativa con media $\mu = 1$, se obtuvo el histograma de todos los números generados y se comparó con su función de densidad de probabilidad (PDF).

De la misma manera con la otra distribución, se realizaron 10000 generaciones de numeros siguiendo una distribución Rayleigh con desviación estándar $\sigma = 1$, se obtuvo el histograma de todos los números generados y se comparó con su función de densidad de probabilidad (PDF).

Se observa que la distribuciones de los histogramas siguen a su respectiva función densidad de probabilidad (PDF) [véanse Figuras A.3 y A.4], por lo que se valida la generación de números Exponenciales y Rayleigh en Python.

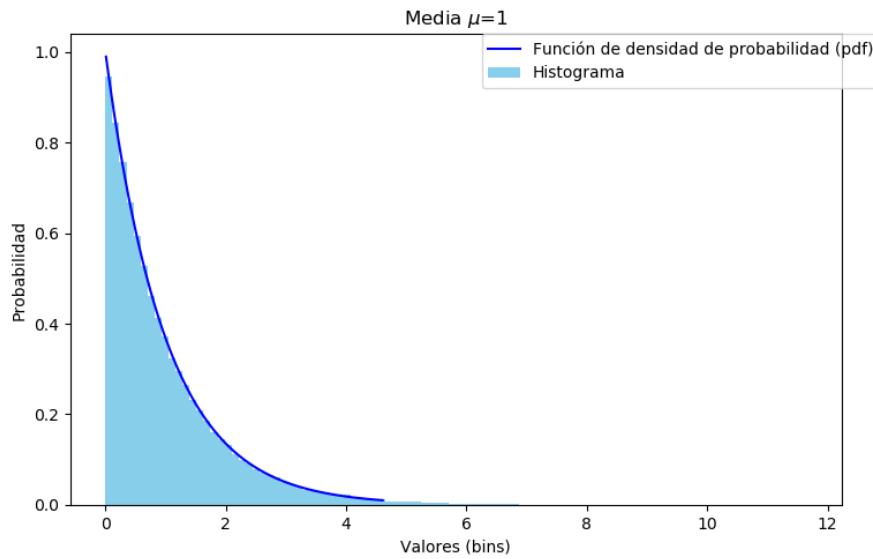


FIGURA A.3: Gráfica comparación PDF e histograma de distribución Exponencial en Python

A.2.3. Generación de variable aleatoria tipo *Pareto*

Como se revisó en la sección 4.4.5, la longitud de paquetes en la transmisión periódica siguió una distribución de Pareto con parámetro alfa = 2.5 y tamaño mínimo de carga útil de la aplicación = 20 bytes con un corte a 200 bytes.

La distribución de Pareto a veces se conoce como el Principio de Pareto o la regla '80 –20 ', en este caso la regla establece que el 80 % de los tamaños de paquete que se formarán, los producirán solo el 20 % de los dispositivos del sistema y viceversa.

La distribución de Pareto se puede replicar en Python usando el módulo Scipy.stats o NumPy. El módulo Scipy.stats abarca varias distribuciones de probabilidad y una biblioteca cada vez mayor de funciones estadísticas. Se comprobó la generación de esta variable aleatoria en Python usando la librería *scipy*.

Se realizaron 100000 generaciones de números siguiendo una distribución Pareto con parámetro alfa $\alpha = 2.5$ acotada entre 20 y 200, se obtuvo el histograma de todos los números generados y se comparó con su función de densidad de probabilidad (PDF).

Se observa que la distribución del histograma sigue a la función densidad de probabilidad Pareto [véase Figura A.5], por lo que se valida la generación de números Pareto en Python.

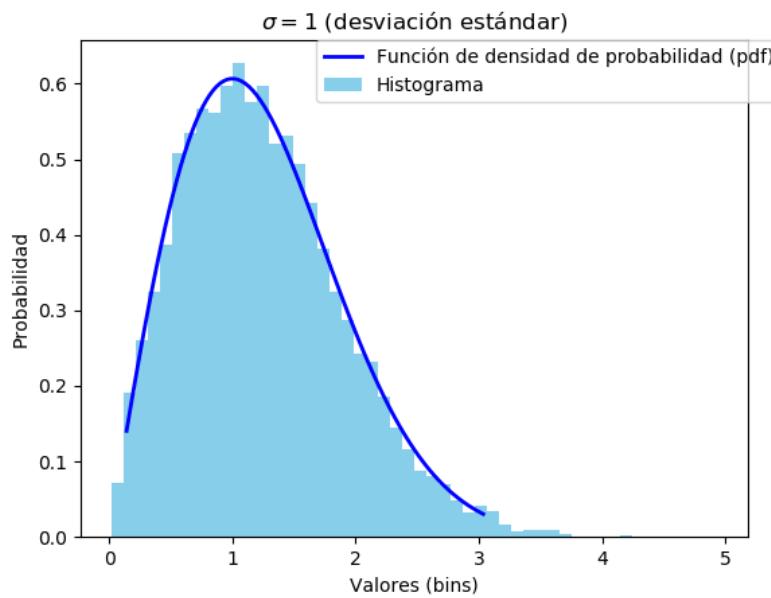


FIGURA A.4: Gráfica comparación PDF e histograma de distribución Rayleigh en Python

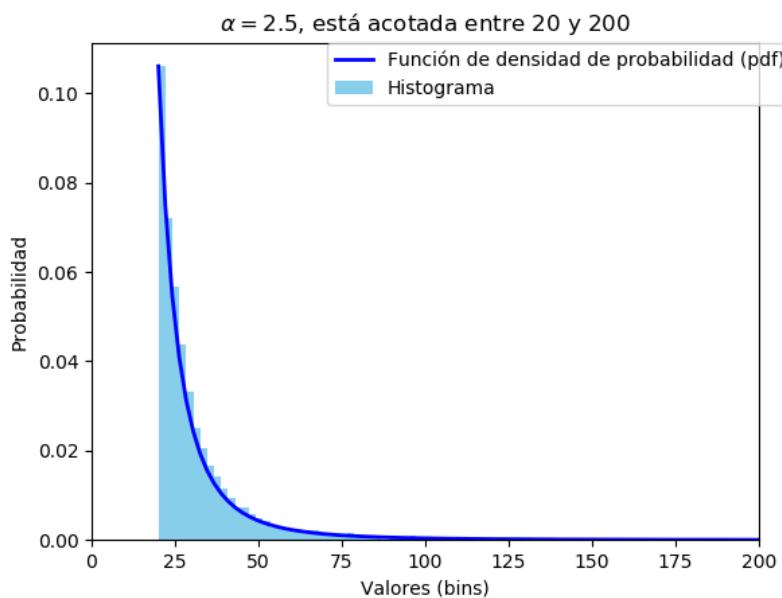


FIGURA A.5: Gráfica comparación PDF e histograma de distribución Pareto en Python

Bibliografía

- [1] G. Fettweis y S. Alamouti, «5G: Personal mobile internet beyond what cellular did to telephony,» *IEEE Communications Magazine*, 2014, ISSN: 01636804. DOI: [10.1109/MCOM.2014.6736754](https://doi.org/10.1109/MCOM.2014.6736754).
- [2] G. P. Fettweis, «5G and the Future of IoT And on its Hardware / Software Impact,» n.^o June 2016, págs. 21-24, 2020.
- [3] Ericsson, «Ericsson Mobility Report (June 2019),» n.^o June, pág. 28, 2019.
- [4] Next Generation Mobile Networks Alliance 5G Initiative, «5G White Paper,» *A Deliverable by the NGMN Alliance*, pág. 124, 2015, ISSN: 0027-9684. DOI: [10.1021/la100371w](https://www.ngmn.org/uploads/media/NGMN%7B%5C_%7D5G%7B%5C_%7DWhite%7B%5C_%7DPaper%7B%5C_%7DV1%7B%5C_%7D0%7B%5C_%7D01.pdf). dirección: https://www.ngmn.org/uploads/media/NGMN%7B%5C_%7D5G%7B%5C_%7DWhite%7B%5C_%7DPaper%7B%5C_%7DV1%7B%5C_%7D0%7B%5C_%7D01.pdf.
- [5] GSM Association, *GSMA Mobile IoT LPWA - LTE-M & NB-IoT Commercial Launches / GSMA*, 2019.
- [6] H. Yu, «What is 5G ? Emerging 5G Mobile Services and Network Requirements,» págs. 1-22, 2017. DOI: [10.3390/su9101848](https://doi.org/10.3390/su9101848).
- [7] T. Hossfeld y F. Metzger, «Traffic Modeling for Aggregated Periodic IoT Data,» n.^o March, 2018. DOI: [10.13140/RG.2.2.30475.39203](https://doi.org/10.13140/RG.2.2.30475.39203).
- [8] I. Dd, J. Xu, G. Shi y C.-X. Wang, *5G Wireless Systems: Simulation and Evaluation Techniques*. 2018, ISBN: 978-3-319-61868-5. DOI: [10.1007/978-3-319-61869-2](https://doi.org/10.1007/978-3-319-61869-2).
- [9] W. Ejaz, M. A. Imran y M. Jo, «Internet of Things (IoT) in 5G Wireless Communications,» n.^o January, 2016. DOI: [10.1109/ACCESS.2016.2646120](https://doi.org/10.1109/ACCESS.2016.2646120).
- [10] M. Haenggi, J. G. Andrews, F. Baccelli, O. Dousse y M. Franceschetti, «Stochastic geometry and random graphs for the analysis and design of wireless networks,» *IEEE Journal on Selected Areas in Communications*, 2009, ISSN: 07338716. DOI: [10.1109/JSAC.2009.090902](https://doi.org/10.1109/JSAC.2009.090902).
- [11] *The Poisson Process: Everything you need to know - Towards Data Science*. dirección: <https://towardsdatascience.com/the-poisson-process-everything-you-need-to-know-322aa0ab9e9a> (visitado 05-07-2020).
- [12] W. H. Tranter, K. S. Shanmugan, T. S. Rappaport y K. L. Kosbar, *Principles of Communication Systems Simulation with Wireless Applications*. 2003, ISBN: 0134947908.
- [13] L. M. Correia, «Mobile Communications Systems - Course Notes,» n.^o February, 2018, ISSN: 0009-4978. DOI: [10.1016/b978-075064355-9/50006-4](https://doi.org/10.1016/b978-075064355-9/50006-4).

- [14] F. Al-turjman, E. Ever y H. Zahmatkesh, «Small Cells in the Forthcoming 5G / IoT : Traffic Modelling and Deployment Overview,» *IEEE Communications Surveys & Tutorials*, vol. PP, n.º c, pág. 1, 2018. DOI: [10.1109/COMST.2018.2864779](https://doi.org/10.1109/COMST.2018.2864779).
- [15] V. Mathuranathan, «Simulation of Digital Communication Systems Using Matlab,» *Kindle*, 2016, ISSN: 1098-6596. DOI: [10.1017/CBO9781107415324.004](https://doi.org/10.1017/CBO9781107415324.004). arXiv: [arXiv:1011.1669v3](https://arxiv.org/abs/1011.1669v3).
- [16] *Rayleigh Fading - an overview / ScienceDirect Topics*. dirección: <https://www.sciencedirect.com/topics/computer-science/rayleigh-fading> (visitado 21-02-2020).
- [17] D. Tse y P. Viswanath, «Fundamentals of Wireless Communication 1,» *Cambridge University Press*, 2004, ISSN: 0018-9448. DOI: [10.1109/TIT.2008.2009814](https://doi.org/10.1109/TIT.2008.2009814). arXiv: [arXiv:1011.1669v3](https://arxiv.org/abs/1011.1669v3).
- [18] M. Shirvanimoghaddam, M. Condoluci, M. Dohler y S. J. Johnson, «On the Fundamental Limits of Random Non-Orthogonal Multiple Access in Cellular Massive IoT,» *IEEE Journal on Selected Areas in Communications*, vol. 35, n.º 10, págs. 2238-2252, 2017, ISSN: 07338716. DOI: [10.1109/JSAC.2017.2724442](https://doi.org/10.1109/JSAC.2017.2724442). arXiv: [1705.10471](https://arxiv.org/abs/1705.10471).
- [19] Z. Ding, X. Lei, G. K. Karagiannidis, R. Schober, J. Yuan y V. K. Bhargava, «A Survey on Non-Orthogonal Multiple Access for 5G Networks: Research Challenges and Future Trends,» *IEEE Journal on Selected Areas in Communications*, vol. 35, n.º 10, págs. 2181-2195, 2017, ISSN: 07338716. DOI: [10.1109/JSAC.2017.2725519](https://doi.org/10.1109/JSAC.2017.2725519).
- [20] Z. Ding, P. Fan y H. V. Poor, «Impact of User Pairing on 5G Nonorthogonal Multiple-Access Downlink Transmissions,» *IEEE Transactions on Vehicular Technology*, vol. 65, n.º 8, págs. 6010-6023, 2016, ISSN: 00189545. DOI: [10.1109/TVT.2015.2480766](https://doi.org/10.1109/TVT.2015.2480766).
- [21] R. C. Kizilirmak, «Non-Orthogonal Multiple Access (NOMA) for 5G Networks,» en *Towards 5G Wireless Networks - A Physical Layer Perspective*, 2016. DOI: [10.5772/66048](https://doi.org/10.5772/66048).
- [22] A. Nair, G. Gupta y K. Srinivas, «REVIEW ON MULTIPLE ACCESS TECHNIQUES USED IN MOBILE TELECOMMUNICATION GENERATIONS,» *International Research Journal of Engineering and Technology (IRJET)*, 2018.
- [23] W. Xiang y K. Z. Xuemin, *5G Mobile Communications*, ISBN: 9783319342061.
- [24] G. A. Akpakwu, G. S. Member, B. J. Silva y S. Member, «A Survey on 5G Networks for the Internet of Things : Communication Technologies and Challenges,» n.º December, 2017. DOI: [10.1109/ACCESS.2017.2779844](https://doi.org/10.1109/ACCESS.2017.2779844).
- [25] G. Editorial, «Internet of Things (IoT) in 5G Wireless Communications,» vol. 4, págs. 10310-10314, 2020.
- [26] J. M. Carter, B. V. Gnedenko e I. N. Kovalenko, «Introduction to Queueing Theory,» *The Mathematical Gazette*, 1990, ISSN: 00255572. DOI: [10.2307/3618175](https://doi.org/10.2307/3618175).
- [27] L. Chen, W. Chen, B. Wang, X. Zhang, H. Chen y D. Yang, «System-level simulation methodology and platform for mobile cellular systems,» *IEEE Communications Magazine*, 2011, ISSN: 01636804. DOI: [10.1109/MCOM.2011.5936168](https://doi.org/10.1109/MCOM.2011.5936168).
- [28] J. Banks y J. H. Carson, *Discrete Event Simulation System*, 2005.

- [29] Descripción general: documentación de SimPy 3.0.11. dirección: <https://simpy.readthedocs.io/en/latest/> (visitado 14-11-2019).
- [30] Ventajas y desventajas del lenguaje de programación Python. dirección: <https://medium.com/@mindfiresolutions.usa/advantages-and-disadvantages-of-python-programming-language-fd0b394f2121> (visitado 14-11-2019).
- [31] multiprocessing — Process-based parallelism — Python 3.8.0 documentation. dirección: <https://docs.python.org/3/library/multiprocessing.html> (visitado 20-11-2019).
- [32] E. Celis y L. Martínez, «Análisis de rendimiento en sistemas celulares 4G mediante algoritmos de asignación de recursos (canal, sub-portadoras),» *Tesis IPN*, 2016.
- [33] V. Gupta, S. K. Devar, N. H. Kumar y K. P. Bagadi, «Modelling of IoT Traffic and Its Impact on LoRaWAN,» *2017 IEEE Global Communications Conference, GLOBECOM 2017 - Proceedings*, vol. 2018-Janua, págs. 1-6, 2018. DOI: [10.1109/GLOCOM.2017.8254512](https://doi.org/10.1109/GLOCOM.2017.8254512).
- [34] K. Smiljkovic, V. Atanasovski y L. Gavrilovska, «Machine-to-Machine traffic characterization: Models and case study on integration in LTE,» *2014 4th International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace and Electronic Systems, VITAE 2014 - Co-located with Global Wireless Summit*, págs. 1-5, 2014. DOI: [10.1109/VITAE.2014.6934482](https://doi.org/10.1109/VITAE.2014.6934482).
- [35] Z. Zhang, H. Sun y R. Q. Hu, «Downlink and uplink non-orthogonal multiple access in a dense wireless network,» *IEEE Journal on Selected Areas in Communications*, vol. 35, n.º 12, págs. 2771-2784, 2017, ISSN: 07338716. DOI: [10.1109/JSAC.2017.2724646](https://doi.org/10.1109/JSAC.2017.2724646).
- [36] A. E. Mostafa, Y. Zhou y V. W. Wong, «Connection Density Maximization of Narrowband IoT Systems with NOMA,» en *IEEE Transactions on Wireless Communications*, 2019. DOI: [10.1109/TWC.2019.2927666](https://doi.org/10.1109/TWC.2019.2927666).
- [37] A. Shahini y N. Ansari, «NOMA Aided Narrowband IoT for Machine Type Communications With User Clustering,» *IEEE Internet of Things Journal*, vol. 6, n.º 4, págs. 7183-7191, 2019, ISSN: 2327-4662. DOI: [10.1109/jiot.2019.2914947](https://doi.org/10.1109/jiot.2019.2914947). arXiv: [1812.08866](https://arxiv.org/abs/1812.08866).
- [38] L. Chettri y R. Bera, *A Comprehensive Survey on Internet of Things (IoT) Toward 5G Wireless Systems*, 2020. DOI: [10.1109/JIOT.2019.2948888](https://doi.org/10.1109/JIOT.2019.2948888).
- [39] J. Mocnej, A. Pekar y W. K. G. Seah, «Network Traffic Characteristics of the IoT Application Use Cases,»
- [40] 5G Americas, *5G the future of IoT*. dirección: https://www.5gamericas.org/wp-content/uploads/2019/07/5G%7B%5C_%7DAmericas%7B%5C_%7DWhite%7B%5C_%7DPaper%7B%5C_%7Don%7B%5C_%7D5G%7B%5C_%7DIoT%7B%5C_%7DFINAL%7B%5C_%7D7.16.pdf (visitado 20-11-2019).
- [41] Ericsson AB, «Cellular networks for Massive IoT – enabling low power wide area applications,» *Ericsson*, 2016.
- [42] 3GPP, *The 3rd Generation Partnership Project (3GPP)*, 2019. dirección: <https://www.3gpp.org/about-3gpp/about-3gpp>.

- [43] O. Liberg, M. Sundberg, Y.-P. E. Wang, J. Bergman y J. Sachs, «The Cellular Internet of Things,» en *Cellular Internet of Things*, 2018. DOI: [10.1016/b978-0-12-812458-1.00001-0](https://doi.org/10.1016/b978-0-12-812458-1.00001-0).
- [44] GSMA, *NB-IoT Deployment Guide*. dirección: <https://www.gsma.com/iot/wp-content/uploads/2019/07/201906-GSMA-NB-IoT-Deployment-Guide-v3.pdf> (visitado 04-07-2020).
- [45] J. Schlienz y D. Raddino, «Narrowband Internet of Things Whitepaper NarrowBand_IoT_1MA266_0e,» inf. téc. dirección: www.rohde-schwarz.com/appnote/.
- [46] N. Kouzayha, Z. Dawy, J. G. Andrews y H. ElSawy, «Joint Downlink/Uplink RF Wake-Up Solution for IoT Over Cellular Networks,» *IEEE Transactions on Wireless Communications*, vol. 17, n.º 3, págs. 1574-1588, 2018, ISSN: 15361276. DOI: [10.1109/TWC.2017.2781696](https://doi.org/10.1109/TWC.2017.2781696).
- [47] T. S. Rappaport, Y. Xing, G. R. MacCartney, A. F. Molisch, E. Mellios y J. Zhang, *Overview of Millimeter Wave Communications for Fifth-Generation (5G) Wireless Networks-With a Focus on Propagation Models*, 2017. DOI: [10.1109/TAP.2017.2734243](https://doi.org/10.1109/TAP.2017.2734243).
- [48] S. Sun, T. S. Rappaport, T. A. Thomas, A. Ghosh, H. C. Nguyen, I. Z. Kovacs, I. Rodriguez, O. Koymen y A. Partyka, «Investigation of Prediction Accuracy, Sensitivity, and Parameter Stability of Large-Scale Propagation Path Loss Models for 5G Wireless Communications,» *IEEE Transactions on Vehicular Technology*, vol. 65, n.º 5, págs. 2843-2860, 2016, ISSN: 00189545. DOI: [10.1109/TVT.2016.2543139](https://doi.org/10.1109/TVT.2016.2543139).
- [49] M. Al-Imari, P. Xiao, M. A. Imran y R. Tafazolli, «Uplink non-orthogonal multiple access for 5G wireless networks,» *2014 11th International Symposium on Wireless Communications Systems, ISWCS 2014 - Proceedings*, págs. 781-785, 2014. DOI: [10.1109/ISWCS.2014.6933459](https://doi.org/10.1109/ISWCS.2014.6933459). arXiv: [1705.04108](https://arxiv.org/abs/1705.04108).
- [50] L. Zhu, J. Zhang, Z. Xiao, X. Cao y D. O. Wu, «Optimal User Pairing for Downlink Non-Orthogonal Multiple Access (NOMA),» *IEEE Wireless Communications Letters*, vol. 8, n.º 2, págs. 328-331, 2019, ISSN: 21622345. DOI: [10.1109/LWC.2018.2853741](https://doi.org/10.1109/LWC.2018.2853741).
- [51] M. Laner, P. Svoboda, N. Nikaein y M. Rupp, «Traffic models for machine type communications,» en *Proceedings of the International Symposium on Wireless Communication Systems*, 2013, ISBN: 9783800735297.
- [52] *Distribución uniforme (continua) - MATLAB y Simulink - MathWorks América Latina*. dirección: <https://la.mathworks.com/help/stats/uniform-distribution-continuous.html> (visitado 05-07-2020).
- [53] L. M. Leemis y J. T. Mcqueston, «Univariate Distribution Relationships,» DOI: [10.1198/000313008X270448](https://doi.org/10.1198/000313008X270448). dirección: www.math.wm.edu/.