

ANÁLISIS DEL DESEMPEÑO DE LA ANTIGÜEDAD DE LA INFORMACIÓN EN UNA RED LTE-A PARA APLICACIONES DE INTERNET DE LAS COSAS

PERFORMANCE ANALYSIS OF THE AGE OF INFORMATION IN AN LTE-A NETWORK FOR INTERNET OF THINGS APPLICATIONS

Ronald Patricio Vaca Guadalupe¹, Luis Tello-Oquendo²

ronald.vaca@unach.edu.ec¹, luis.tello@unach.edu.ec²

Fecha de recepción: 09/09/2025 / Fecha de aceptación: 14/09/2025 / Fecha de publicación: 15/09/2025

RESUMEN: Un sistema ciberfísico (CPS) integra sensores, actuadores y redes de comunicación que permiten la interacción continua entre el entorno físico y el digital. Las actualizaciones de información oportunas en este tipo de sistemas son críticas. El objetivo de esta investigación es analizar el desempeño de la métrica Antigüedad de la Información (Aol) en un CPS. Este sistema utiliza conectividad celular basado en LTE-Advanced (LTE-A) para aplicaciones de Internet de las Cosas (IoT) en tiempo real. Se desarrolló una investigación cuantitativa de tipo experimental-simulada, utilizando un simulador de eventos discretos implementado en Python y funciones complementarias en MATLAB. El estudio se centró en el canal de acceso aleatorio (RACH) y evaluó indicadores clave como la Aol promedio, la probabilidad de acceso exitoso y el número de transmisiones de preámbulo. Los resultados evidencian que un mayor número de dispositivos y una mayor frecuencia de actualizaciones incrementan la Aol, reducen la probabilidad de acceso exitoso y aumentan la carga sobre el RACH. Además, se demostró que los patrones de generación de actualizaciones asincrónicos de generación de mensajes (Φ_2) ofrecen mejor rendimiento que los sincronizados (Φ_1) en condiciones de alta demanda de tráfico. En conclusión, la métrica Aol se confirma como un indicador eficaz para dimensionar redes celulares IoT y diseñar estrategias dinámicas de gestión de recursos en escenarios de tráfico masivo.

Palabras clave: Antigüedad de la Información, sistemas celulares, comunicaciones máquina-máquina, análisis de desempeño, fábrica inteligente

ABSTRACT: A cyber-physical system (CPS) integrates sensors, actuators, and communication networks that enable continuous interaction between the physical and digital environments.

¹Posgrado, Universidad Nacional de Chimborazo, Riobamba 060108, Ecuador, <https://orcid.org/0009-0007-2780-2010>.

²College of Engineering, Universidad Nacional de Chimborazo, Riobamba 060108, Ecuador, <https://orcid.org/0000-0002-5274-666X>



Timely updates of information in this kind of system are critical. The objective of this research is to analyze the performance of the Age of Information (AoI) metric in a CPS. This system uses cellular connectivity based on LTE-Advanced (LTE-A) for real-time Internet of Things (IoT) applications. A quantitative research study of an experimental-simulated type was developed, using a discrete event simulator implemented in Python and complementary functions in MATLAB. The study focused on the random access channel (RACH) and evaluated key indicators such as the average AoI, the probability of successful access, and the number of preamble transmissions. The results show that a higher number of devices and a greater frequency of updates increase the AoI, reduce the probability of successful access, and increase the load on the RACH. Moreover, it was demonstrated that asynchronous message generation update patterns (Phi2) offer better performance than synchronized ones (Phi1) under high traffic demand conditions. In conclusion, the AoI metric is confirmed as an effective indicator for dimensioning IoT cellular networks and designing dynamic resource management strategies in massive traffic scenarios.

Keywords: *Age of Information, cellular systems, machine-type communications, performance analysis, smart factory*

INTRODUCCIÓN

Un sistema ciberfísico (CPS) es un sistema a gran escala, interconectado e integrado mediante el mundo cibernético a través de la computación, una red de comunicación y componentes que interactúan con el entorno físico (sensores y actuadores) (1). En este sistema, los procesos físicos afectan a la computación y viceversa mediante la transmisión y recepción de datos utilizando ciclos de retroalimentación por medio de una red de comunicación. Los CPS tienen aplicaciones en múltiples áreas y disciplinas, incluyendo medicina y salud, suministro y consumo de energía, transporte y automóviles, agricultura, automatización industrial y manufactura, entre otras. La aplicación de un CPS en un sistema de producción industrial se denomina sistema de producción ciberfísico (CPPS), dando lugar a lo que actualmente se conoce como fábrica inteligente (2).

Las redes inalámbricas basadas en tecnología 5G son prometedoras para proporcionar conectividad a los CPS. Dentro del ámbito de las redes inalámbricas 5G en aplicaciones en tiempo real, la «Antigüedad de la Información» (AoI) ha surgido como una métrica interesante que cuantifica con qué frecuencia se actualiza la información del estado de un sistema dentro de las aplicaciones subyacentes (3), (5). Desde una perspectiva de sistema, las aplicaciones en tiempo real comparten una descripción común: una fuente (sensor o sistema remoto) genera mensajes de actualización del estado con sello temporal, los cuales se transmiten por la red de comunicación hacia uno o más monitores. El conocimiento del estado del sensor o sistema remoto debe ser lo más oportuno posible (2). La dificultad para que los monitores en una red reciban información actualizada oportunamente motivó esta investigación.



En este estudio se considera un conjunto de sensores con vida útil limitada de batería que monitorean un conjunto de procesos estocásticos en una fábrica inteligente (1), (2). Estos sensores comparten un canal de acceso inalámbrico basado en tecnología 5G para transmitir sus muestras hacia un controlador, quien realiza su posterior procesamiento (6). A continuación, el controlador envía el comando de control necesario al actuador correspondiente para controlar el proceso subyacente. Una vez terminada esta acción de control, se detecta nuevamente el cambio de estado del proceso y se repite el ciclo anterior hasta alcanzar el número máximo de iteraciones (4). Con el fin de mantener estable un CPS basado en tecnología inalámbrica 5G que controla procesos tecnológicos, cuyo fallo podría provocar una interrupción del CPS, una parada temporal de los procesos operados en la planta u otras consecuencias potencialmente graves (como la toma de decisiones antes o después del momento adecuado), se requiere una red con dinámicas en constante cambio que actualice continuamente los estados del sistema y las tareas en curso (5), (7). Estas actualizaciones se logran a través del intercambio de información de estado que es crítica en el tiempo, impulsada por eventos o por el tiempo (8), (9).

La tecnología IoT celular (CIoT, por sus siglas en inglés) se ha convertido en una tecnología que permite proporcionar conectividad a dispositivos IoT y dar soporte a las Comunicaciones de Tipo Máquina (MTC) (6), (7). CIoT se basa en tecnologías de redes celulares como LTE-Advanced (LTE-A) y quinta generación (5G) (17), (18). Estas redes han experimentado un rápido crecimiento, y la ubicuidad de los dispositivos y las diferentes aplicaciones han aumentado significativamente la demanda de mensajes de actualización del estado de los sistemas en tiempo real (10), (11). Por lo tanto, estas redes se enfrentan al aumento en la demanda y al incremento del tráfico generado por las actualizaciones de información de estado, lo que puede provocar situaciones de sobrecarga. Estas situaciones deben gestionarse eficientemente para evitar la pérdida de datos o que se entregue información obsoleta al centro de control, donde se llevan a cabo el análisis y la toma de decisiones (12), (13).

La frecuencia con la que se generan los mensajes de actualización de información del estado de un sistema (siguiendo ciertos patrones en la fuente) y el protocolo de acceso aleatorio (RA) utilizado en la red de acceso por radio (RAN) desempeñan un papel importante en la provisión de actualizaciones oportunas. Existe un compromiso entre la capacidad del canal de acceso aleatorio (conocido como RACH) y la generación de mensajes de actualización de información del estado de un sistema (17), (18). Si el canal RACH tuviera capacidad infinita, generar mensajes de actualización del estado con mayor frecuencia implicaría mejores indicadores clave de rendimiento (KPIs); pero dado que este no es el caso en implementaciones reales, una mayor frecuencia de generación de mensajes incrementa la carga de tráfico, lo que resulta en mayores retardos en el RACH e incluso pérdidas de información (19). Sin embargo, disminuir la frecuencia podría deteriorar seriamente la frescura de la información de estado, especialmente en aplicaciones IoT/CPS en tiempo real (20).



Este artículo tiene como objetivo estudiar el impacto que tienen los diferentes patrones de generación de actualizaciones del estado en la red de acceso por radio del CloT desde la perspectiva de la Aol (3), (8), (19), (20). Para evaluar el rendimiento de la RAN se desarrolló un simulador basado en eventos discretos de la RAN 5G utilizando Python. Además, se realizaron simulaciones independientes en MATLAB para corroborar los resultados (12), (14). Se emplearon dos tipos de tráfico en cada simulación, MTC y tráfico Humano a Humano (H2H), con diferentes intensidades de solicitudes de acceso (17), (18). Esto permitió analizar el impacto de los patrones de actualización de información del estado con diferentes frecuencias en el rendimiento de la RAN en un CPS con conectividad CloT (10).

El documento se estructura de la siguiente manera: en la Sección II se describe la metodología, incluyendo la definición de Aol, los esquemas de generación de actualizaciones y detalla los parámetros de configuración de la red y los indicadores de desempeño (KPIs) empleados en la evaluación de la red (15), (18). Posteriormente, en la Sección III se presentan y analizan los resultados. La Sección IV introduce la discusión. Finalmente, la Sección V proporciona las conclusiones y las líneas de trabajo futuro.

MATERIALES Y MÉTODOS

1. Modelo del sistema

El sistema considerado está compuesto por una estación base (gNB) que atiende de manera simultánea a equipos de usuario (UEs) de tipo MTC y H2H (6), (7). En el instante inicial $t = 0$, todos los UEs se encuentran inactivos y desconectados. A partir de ese momento, los UEs MTC generan mensajes de actualización del estado con una frecuencia determinada por los patrones de generación descritos en la Sección II.5 (3), (19), (20). Cada vez que se produce un mensaje, el UE cambia de inactivo a conectado, estableciendo comunicación con la BS mediante el procedimiento de acceso aleatorio (RA) presentado en la Sección II.2 (15), (16), (17).

Para incorporar tráfico de fondo H2H y aproximar el entorno a condiciones reales de red, se utilizaron registros de llamadas (*Call Detail Records*, CDR) proporcionados por el operador Telecom Italia en el marco del *Big Data Challenge* 2014 (12), (14).

El acceso aleatorio puede operar en dos modalidades: sin contienda (*contention-free*) y con contienda (*contention-based*). La primera se aplica en situaciones críticas como traspasos de conexión o llegada de datos en el enlace descendente, mientras que la segunda constituye el modo estándar utilizado en la mayoría de los accesos de red (15). En este último, los intentos de acceso se realizan en intervalos de tiempo y frecuencia predefinidos, conocidos como oportunidades de acceso aleatorio (RAOs). La estación base anuncia periódicamente la disponibilidad de RAOs a través del parámetro *PRACH Configuration Index*, cuya periodicidad varía entre 1 RAO cada trama (10 ms) y 10 RAOs por trama (1 RAO por ms) (16), (17).



El canal físico de acceso aleatorio (PRACH) transporta los preámbulos necesarios para iniciar la conexión en la RAN. Cada celda dispone de hasta 64 preámbulos ortogonales (16). En el modo sin contienda, estos se asignan de forma coordinada para evitar colisiones y solo pueden emplearse en intervalos específicos por UEs previamente autorizados (15). En el modo con contienda, los preámbulos se seleccionan de manera aleatoria, lo que introduce la posibilidad de colisiones entre dispositivos. En tal caso, es indispensable un procedimiento de resolución de contienda (17), (18). En lo que sigue, el presente trabajo se centra en el análisis del acceso aleatorio con contienda, dado que representa el escenario más crítico para el rendimiento del sistema.

Tabla 1. Resumen de notaciones

Notación	Descripción
R	Preámbulos disponibles para acceso aleatorio basado en contención
W_{RAR}	Tamaño de ventana RAR
UID	Identificación del dispositivo
ϕ	Desfase para los patrones de generación de mensajes de actualización del estado del sistema
$\Phi i1$	Intervalo constante entre actualizaciones en el patrón de dispositivos sincronizados
$\Phi i2$	Intervalo constante entre actualizaciones en el patrón de dispositivos no sincronizados
T_u	Tiempo entre actualizaciones
U	Tasa de llegada de actualizaciones
K	Número de transmisiones del preámbulo por intento de acceso
P_s	Probabilidad de éxito del acceso
ηA	Número total de intentos de acceso exitoso
ηT	Número total de transmisiones del preámbulo necesarias para completar el proceso de acceso aleatorio (RA)

Fuente: (12), (14)

Esta formulación metodológica constituye el marco de simulación adoptado en el estudio. El enfoque es de carácter cuantitativo y computacional, basado en el modelado matemático de los procedimientos de acceso y en la implementación de simulaciones para evaluar el comportamiento del sistema bajo diferentes condiciones de tráfico (12), (14), (17), (18).



2. Proceso de acceso aleatorio basado en contención

La Figura 1 ilustra el procedimiento de acceso aleatorio implementado, que cumple con los estándares del 3GPP. Este procedimiento implica el intercambio de cuatro mensajes entre el equipo de usuario (UE) y la estación base (gNB). Para más detalles, consulte las referencias (10), (12), (14), (17), (18).

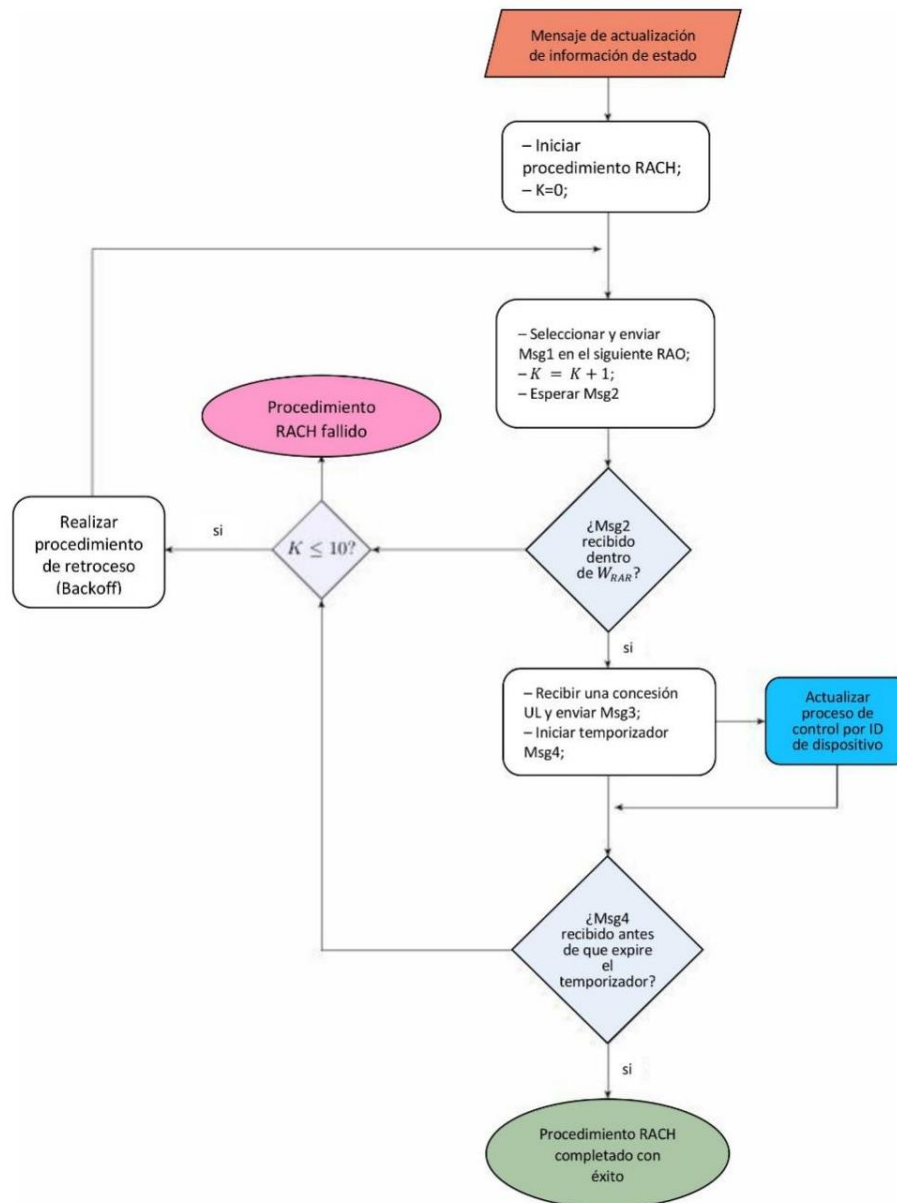


Figura. 1. Proceso de acceso aleatorio basado en contención.



Es necesario tomar en cuenta que se creó un bloque de proceso de control (Actualizar proceso por ID de dispositivo en la Figura 1) para los mensajes de actualización del estado del sistema, con el fin de asegurar que la gNB reciba información precisa y actualizada del estado del sistema generada por los dispositivos, los cuales son identificados mediante una variable U_{ID} . La Sección II.4 proporciona una explicación detallada de este proceso.

3. Antigüedad de la información (Aol)

Desde la perspectiva del sistema, las aplicaciones en tiempo real comparten una descripción común: una fuente (sensor o sistema remoto) genera mensajes de actualización de estado con sello temporal que se transmiten a través de la red de comunicación hacia uno o más monitores. El conocimiento del estado del sensor o sistema remoto debe ser lo más oportuno posible en la red. Los monitores deben recibir información del estado del sistema de forma oportuna, lo cual se define como la edad de la actualización de estado más reciente que un monitor ha recibido.

3.1. Definición:

Sea un nodo fuente encargado de muestrear un proceso dinámico que varía en el tiempo y transmitir dicho estado a través de un canal de comunicación inalámbrico sujeto a retardos. La métrica Aol en el destino se define como el intervalo transcurrido desde la generación de la última actualización recibida (19), (20). Siendo t'_k el instante en que una actualización de estado es recibida en el destino, y ξ el tiempo de observación. El índice de la actualización más reciente que ha llegado se expresa como:

$$N(\xi) = \max\{k | t'_k \leq \xi\} \quad (1)$$

y la marca temporal correspondiente a esa actualización es:

$$u(\xi) = t_{N(\xi)} \quad (2)$$

La Aol en el destino se describe como un proceso aleatorio:

$$AoI(t) = t - u(t) \quad (3)$$

Consideremos que en $t = 0$ se inicia la observación del sistema, la Aol en el destino es $AoI(0) = AoI_0$. La actualización de estado i se genera en el tiempo t_i y el destino la recibe en el tiempo t'_i . Entre t'_{i-1} y t'_i , no se reciben actualizaciones en el destino, por lo tanto, la Aol aumenta linealmente con el tiempo. Al recibir una actualización de estado, la Aol se reinicia al valor del retardo experimentado por el paquete que transporta dicha actualización al atravesar el canal.

El tiempo transcurrido entre la generación de la actualización i y la inmediatamente anterior se define como el intervalo de tiempo transcurrido desde la generación de la actualización i hasta la



generación de la actualización anterior. A este tiempo se le denomina Y_i y es una variable aleatoria:

$$Y_i = t_i - t_{i-1} \quad (4)$$

Además,

$$T_i = t'_i - t_i \quad (5)$$

Al tomar T_i como el tiempo de actualización del sistema i , que representa el retardo introducido por la transferencia a través del canal. Considerando que el intervalo de observación es desde $t = 0$ hasta $t = T = t'_n$, expresamos el número de llegadas durante un tiempo determinado T como $N(T)$:

$$N(T) = \max\{n | t'_n \leq T\} \quad (6)$$

Entonces, en t'_i al denotar $i = \{1, 2, \dots, N(T)\}$, la edad $AoI(t'_i)$ se reinicia a $T_i = t'_i - t_i$. La dinámica de la AoI en el destino refleja la frescura de la información que proviene de la fuente. Cada vez que llega un paquete actualizado, la métrica se reinicia, y en el intervalo hasta la siguiente actualización vuelve a crecer de manera lineal. Este comportamiento se caracteriza por un patrón en forma de dientes de sierra, lo que evidencia claramente el ciclo de aumento y reinicio de la métrica, como se muestra en la Figura 2 (19), (20).

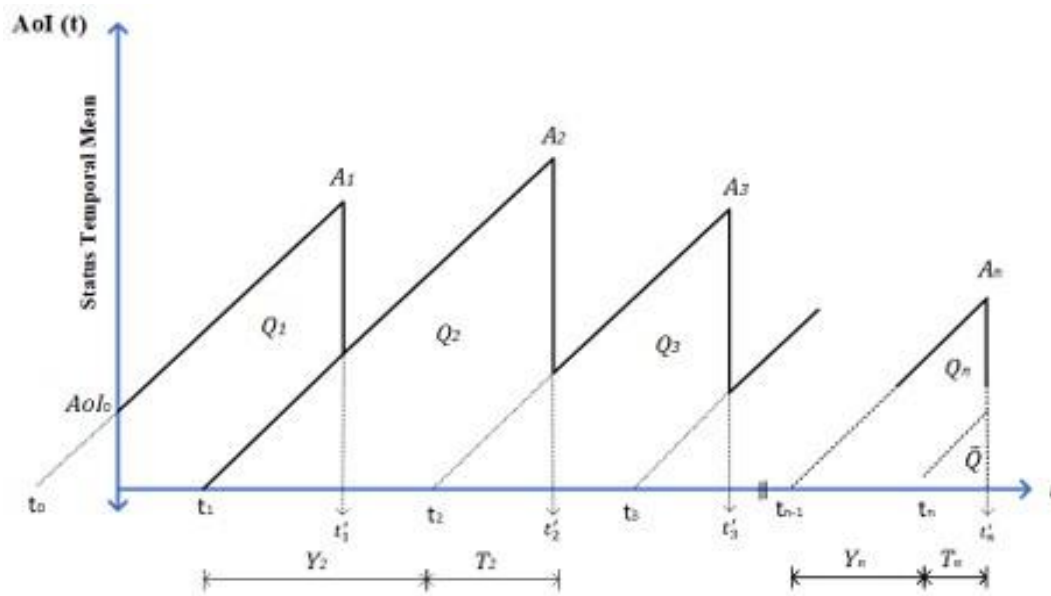


Figura 2. Aol a lo largo del tiempo (20).



3.2. Edad promedio de la información ($\overline{AoI}$):

La manera más utilizada para caracterizar la AoI de una fuente en el destino consiste en calcular su valor promedio a lo largo del tiempo, denotado también como $\overline{AoI}$ (19), (20). Un valor reducido de esta métrica implica que, en promedio, la información del estado almacenada por el monitor es más reciente y, por lo tanto, representa con mayor fidelidad la condición real del sistema observado. La $\overline{AoI}$ sobre un intervalo de duración T se obtiene considerando el área bajo la curva de la Figura 2 durante dicho intervalo, normalizada respecto al tiempo total T (8, 9). Considerando un intervalo de tiempo $(0, T)$ para la observación, la $\overline{AoI}$ de un sistema de actualización de estado se expresa como:

$$\overline{AoI} = \frac{1}{T} \int_0^T AoT(t) dt \quad (7)$$

Si se considera el instante inicial $t = 0$, el área bajo la curva puede descomponerse en diferentes elementos geométricos: el área del polígono Q_i , los trapecios Q_i para $i = \{1, 2, 3, \dots, N(T)\}$, y el área triangular con base T_n , al cual se designa como $\tilde{Q}$. A partir de esta descomposición, la expresión de la $\overline{AoI}$ se reescribe de manera que cada componente corresponde a la contribución de una de estas áreas en el cálculo del promedio dando lugar a:

$$\overline{AoI} = \frac{1}{T} \left(Q_1 + \tilde{Q} + \sum_{i=2}^{N(T)} Q_i \right) = \frac{Q_1 + \tilde{Q}}{T} + \frac{N(T)-1}{T} \frac{1}{N(T)-1} \sum_{i=2}^{N(T)} Q_i \quad (8)$$

3.3. Evaluación computacional de AoI:

Como complemento a la evaluación de desempeño realizada con el simulador de eventos discretos en Python, se implementaron también funciones específicas en MATLAB para analizar de manera detallada el comportamiento temporal de la métrica AoI, tal como se muestra en el Algoritmo 1. Estas funciones permitieron calcular la AoI promedio en diferentes escenarios y contrastar los resultados.

Cabe señalar que la validación del simulador se efectuó a través de dos enfoques: en primer lugar, se compararon los resultados generados en Python y MATLAB, observándose una coincidencia consistente en las tendencias de las métricas; en segundo lugar, se verificó que dichos resultados fueran coherentes con los comportamientos descritos en investigaciones previas sobre LTE-A y comunicaciones masivas de tipo máquina (3), (8), (17). Aunque no se dispuso de un entorno físico de pruebas, la doble verificación mediante distintas plataformas de simulación y la concordancia con la literatura garantizan la fiabilidad y solidez de los resultados. Estas funciones permiten calcular de forma precisa la AoI promedio en distintos escenarios:

- **'sourceTemporalMean_AoI'**: calcula la AoI media temporal directamente desde la fuente, bajo el supuesto de que no existen retardos de transmisión. Este análisis se fundamenta en la integración de áreas triangulares entre actualizaciones, siguiendo la fórmula:



$$\overline{AoI}(t) = \frac{1}{T} \cdot \frac{1}{2} \sum_{i=1}^{n-1} (t_i - t_{i-1})^2 \quad (9)$$

donde $\overline{AoI}(t)$ es la edad promedio de la información, $T = t_n - t_i$ es el tiempo total de información y $(t_i - t_{i-1})^2$ es el área triangular bajo la curva de AoI.

- **'temporalMean_AoI'**: evalúa la AoI considerando retardos de red o transmisiones no instantáneas. Utiliza la interpolación trapezoidal entre actualizaciones, aplicando la fórmula:

$$\overline{AoI}(t) = \frac{1}{T} \sum_{i=1}^{n-1} \Delta t_i \cdot (AoI(t_i^-) - \frac{1}{2} \Delta t_i) \quad (10)$$

donde $AoI(t_i^-)$ es el valor de AoI justo antes de la actualización, $\Delta t_i = t_i - t_{i-1}$ es la duración del intervalo entre actualizaciones.

- **'aggregate_AoI'**: construye una curva de AoI agregada a partir de múltiples usuarios bajo modos de agregación promedio:

$$\overline{AoI}_{agg}(t) = \frac{1}{N} \sum_j AoI_j(t), \quad \text{modo 'avg'} \quad (11)$$

donde $\overline{AoI}_{agg}(t)$ es la Edad de la Información agregada, $AoI_j(t)$ es AoI del usuario j en el instante t y N es el número total de usuarios.

Estas funciones que se utilizan en el Algoritmo de la Tabla 4 se integraron al flujo metodológico para validar cuantitativamente las curvas de AoI generadas en las simulaciones, así como para explorar comportamientos extremos y casos límite del sistema. La comparación entre las curvas derivadas de simulación y las obtenidas por integración directa en MATLAB refuerza la confiabilidad de los resultados presentados en la Sección III.

4. Proceso de control de los mensajes de actualización del estado del sistema

Durante el intervalo entre RAO_{i-1} y RAO_i , la gNB controla el proceso de los mensajes de actualización de información del estado. En este período, cada UE genera mensajes de actualización indistintamente, lo que da lugar a dos escenarios distintos.

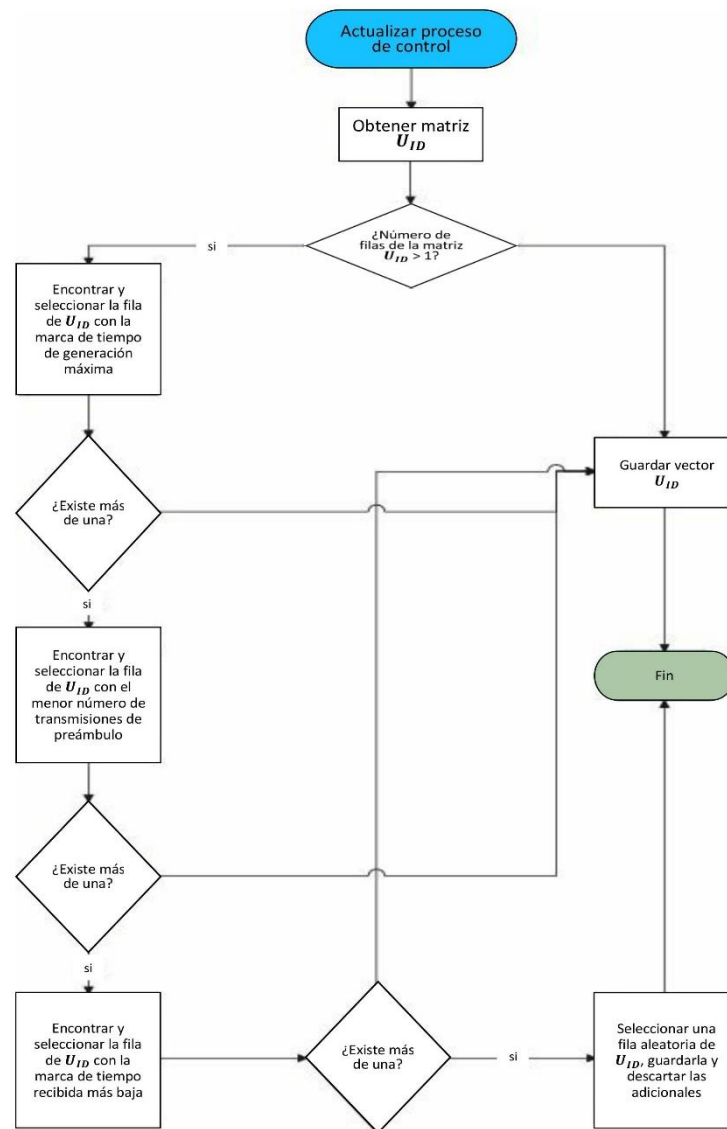


Figura 3. Diagrama de flujo del proceso de control de los mensajes de actualización del estado por dispositivo IoT.

Un UE llega y es transmitido en el siguiente RAO. En el segundo escenario, múltiples paquetes de un determinado UE se reciben dentro del mismo RAO, es necesario aplicar un procedimiento de depuración para descartar la información desactualizada y conservar únicamente la actualización más reciente en el lado del UE.

El mecanismo de depuración y eliminación de mensajes de actualización del estado abarca distintas fases secuenciales, representadas en la Figura 3, donde la matriz U_{ID} guarda información de varios UEs, incluyendo las marcas temporales de las actualizaciones y el número de



transmisiones de preámbulo realizadas. El proceso de control de actualizaciones por ID de dispositivo consiste en lo siguiente:

- 4.1. **Descartar de acuerdo con la marca de tiempo de tiempo de generación:** Conservando así el mensaje de actualización de estado más reciente.
- 4.2. **Descartar según el número de transmisiones de preámbulo:** Cada dispositivo tiene un máximo de 10 intentos para enviar el preámbulo, según lo indicado en (21). Una vez superada la primera depuración, se selecciona el valor correspondiente a la variable indicada y aquel con el menor número de intentos, ya que entre cada intento de envío del preámbulo se añade un cierto retardo, lo cual afecta la frescura del mensaje de actualización del estado que será enviado. En caso de que más de dos mensajes de actualización de estado hayan superado las dos etapas de depuración anteriores, se propone una tercera.
- 4.3. **Descartar con base en la marca de tiempo de cada mensaje de actualización:** De este modo, se selecciona el que tenga la marca de tiempo más reciente, representando así el estado más actualizado del sistema. Finalmente, si aún hay más de un mensaje de actualización de estado, se elige aleatoriamente uno solo entre los disponibles para ser procesado.

5. Patrones de generación de mensajes de actualización del estado del sistema

Se definieron dos patrones diferentes de generación de mensajes de actualización del estado desde la fuente para el tráfico MTC, los cuales utilizan distintas frecuencias de actualización. Estos son:

- 5.1. **El tiempo no varía entre actualizaciones con dispositivos sincronizados (Phi1):** Todos los dispositivos IoT generan mensajes de actualización de estado en $\phi + kT_u$, con $k = \{0, 1, 2, \dots\}$
- 5.2. **El tiempo no varía entre actualizaciones con dispositivos no sincronizados (Phi2):** El dispositivo IoT genera mensajes de actualización de estado en $\phi_i + kT_u$, con $k = \{0, 1, 2, \dots\}$

Se debe tener en cuenta que el tiempo está dividido en intervalos (slots), y el tiempo entre actualizaciones T_u es equivalente al inverso de la tasa de llegada de actualizaciones, es decir: $T_u = (\frac{U}{1000})^{-1}$; $0 \leq \phi, \phi_i < T_u$. La diferencia fundamental entre ambos patrones radica en que, en el esquema sincronizado, todos los dispositivos generan actualizaciones al mismo tiempo, mientras que, en el asincrónico, cada dispositivo lo hace en instantes distintos, evitando la coincidencia simultánea de mensajes.



6. Parámetros de configuración de la red y métricas de desempeño

Para la evaluación de las propuestas se implementó un simulador de eventos discretos que representa la red de acceso por radio (RAN) 5G para evaluar las propuestas (12), (14), (17), (18). El sistema acomoda tráfico MTC y H2H en cada simulación, con diferentes intensidades de solicitudes de acceso (15), (16). El procedimiento de acceso aleatorio, descrito en la Sección II, fue replicado utilizando los parámetros resumidos en la Tabla 2. Las simulaciones se ejecutaron j veces hasta que el promedio de los resultados de la simulación j difiriera en menos del 1% con respecto a la simulación $j - 1$ (17). El simulador proporciona la flexibilidad de seleccionar los parámetros de interés, tales como: tipo de tráfico, número de dispositivos, temporización, procesamiento y características del canal. Además, es posible ajustar el número de preámbulos disponibles, la prioridad de acceso, el tamaño de la ventana de retroceso y los temporizadores de contención, entre otros. Las simulaciones se realizaron considerando los parámetros listados en la Tabla 3.

**Tabla 2. Configuración RACH.**

Parámetro	Configuración
Índice de configuración PRACH	$prach - ConfigIndex = 6$
Periodicidad de los RAOs	5 ms
Longitud de subtrama	1 ms
Preámbulos disponibles para acceso aleatorio por contienda	$R = 54$
Número máximo de transmisiones de preámbulo	$preambleTransMax = 10$
Tamaño de la ventana RAR	$W_{RAR} = 2 \text{ subtramas}$
Número máximo de concesiones UL por subtrama	$N_{RAR} = 3$
Número máximo de concesiones UL por ventana RAR	$N_{UL} = W_{RAR} \times N_{RAR} = 15$
Temporizador de retroceso (<i>Backoff</i>)	$BI = 20 \text{ ms}$
Temporizador de resolución de contención	48 subtramas
Probabilidad de retransmisión para Msg3 y Msg4	0.1
Número máximo de transmisiones de Msg3 y Msg4	5
Retardo de procesamiento del preámbulo	2 subtramas
Retardo de procesamiento de la concesión subida	5 subtramas
Retardo de procesamiento de solicitud de conexión	4 subtramas
Tiempo de ida y vuelta (RTT) de Msg3	5 subtramas
RTT de Msg4	5 subtramas

Fuente: (15), (16)

**Tabla 3. Configuración de la simulación.**

Parámetro	Configuración
Número de mensajes de actualización de estado por segundo (U)	[5, 50]
Número de dispositivos MTC (N)	[4, 100]
Tiempo de simulación (T)	1 minuto (fijo)
Criterio de parada (error)	< 1%

Con respecto a las métricas de desempeño, se consideran tres indicadores clave de rendimiento (KPIs) (17), (21), para la evaluación de la capacidad del RACH bajo cada política de actualización. Estos son los siguientes:

- Edad promedio de la información ($\overline{AoI}$), tal como se definió en la Sección II.3.
- Probabilidad de éxito en el acceso (P_s), la misma que se define como la probabilidad de que el procedimiento de acceso aleatorio se complete satisfactoriamente dentro del número máximo de transmisiones de preámbulo permitidas. Matemáticamente, puede expresarse como:

$$P_s = \frac{\eta_A}{\eta_T} \quad (12)$$

donde η_A representa el número total de intentos en los que el acceso se realizó de manera exitosa, mientras que η_T corresponde al número total de transmisiones de preámbulo efectuadas.

- Número promedio de transmisiones de preámbulo por intento de acceso $E[K]$.

Para asegurar la robustez estadística en la representación de los resultados de simulación, cada punto en las gráficas fue representa múltiples ejecuciones independientes del algoritmo. Las repeticiones continuaron iterativamente hasta que la variación relativa entre los promedios acumulados correspondientes a las repeticiones $j - 1$ y j fue inferior al 1%. La Tabla 4 detalla paso a paso el procedimiento lógico del algoritmo desarrollado para calcular la antigüedad agregada de la Información ($\overline{AoI}$) promedio, basado en la función '*aggregate_AoI*', utilizado dentro del entorno de simulación.



Tabla 4. Algoritmo simulador para el cálculo de la Aol promedio.

Entradas: *updates_users*, *aggregation_mode*, *numUEs* (Celda de usuarios con matrices de actualizaciones; *modo* = 'avg'; número total de UEs)

Salidas: *updates_aggregated* (matriz [tiempo, Aol_promedio])

1 Inicializar el vector *t_ud* vacío para almacenar todos los tiempos de llegada de actualizaciones.

2 Inicializar *t_last_first_ud* = 0 para registrar el tiempo de la última primera actualización entre todos los usuarios.

3 **for** *i* = 1 hasta *numUEs* **do**

4 Obtener el vector *t_ud_i* de tiempos de actualización del usuario *i*.

5 Añadir todos los elementos de *t_ud_i* al vector *t_ud*.

6 **if** *t_ud_i*(1) > *t_last_first_ud* **then**

7 Actualizar *t_last_first_ud* = *t_ud_i*(1).

8 **end if**

9 **end for**

10 Ordenar el vector *t_ud* de forma ascendente.

11 Eliminar todos los tiempos *t* < *t_last_first_ud* del vector *t_ud*.

12 Definir *n_ud* como la longitud del vector *t_ud* (número total de instantes considerados).

13 Inicializar una matriz Aol de tamaño [*n_ud* x *numUEs*] con ceros.

14 **for** *i* = 1 hasta *numUEs* **do**

15 Calcular el Aol del usuario *i* evaluado en los tiempos *t_ud* mediante la función *Aol_t*.

16 Almacenar el resultado en la columna *i* de la matriz Aol.

17 **end for**

18 **if** *aggregation_mode* == 'avg' **then**

19 Calcular Aol agregado *a Aol* como el promedio fila a fila: *a Aol* = *mean*(Aol, 2).

20 Generar un error indicando 'modo de agregación no válido'.

21 **end if**

22 Construir la matriz final *updates_aggregated* = [*t_ud*, *aAol*].

23 Retornar la matriz *updates_aggregated*.



El algoritmo tiene una complejidad computacional de orden:

$$\mathcal{O} = (n \times m)$$

donde n es el número total de instantes evaluados (longitud del vector t_{ud}), y m es el número total de usuarios (UEs). Esta complejidad está dominada por la construcción de la matriz $AoI[n \times m]$ y el cálculo del promedio fila a fila (modo 'avg'), ya que se evalúa la AoI de cada usuario en todos los instantes contenidos en t_{ud} .

RESULTADOS

A continuación, se presentan los resultados del desempeño de la red para tráfico MTC, de acuerdo con la configuración de red presentada en la Tabla 2. Para ello se desarrolló un simulador por eventos discretos de la RAN 5G en Python y corroborados independientemente con simulaciones en MATLAB. Las simulaciones se ejecutaron en un PC con Windows 10 (64 bits), procesador Intel Core i7-6600U a 2.81 GHz, y memoria RAM de 16 GB, con una precisión de reloj de 10^{-7} segundos.

Se considera que el número de mensajes de actualización del estado por segundo (U) por dispositivo MTC varía entre 5 y 50. Además, el número de dispositivos MTC en la red N varía entre 40 y 100. El tráfico H2H se considera como tráfico de fondo en cada escenario. Durante cada prueba, el tiempo total de simulación se fijó en un minuto, es decir, $T = 1$ minuto.

1. Patrón de generación de mensajes de actualización del estado: Phi1

La Figura 4 ilustra el comportamiento de la Edad de la Información $\overline{AoI}$ en función del número de mensajes de actualización del estado por dispositivo U , para diversas cargas del sistema. Se observa que, a partir de cierto valor de U , el deterioro en el rendimiento del sistema se hace evidente, manifestándose en un aumento significativo de la $\overline{AoI}$. Existe un punto de inflexión a partir del cual incrementar U afecta negativamente al sistema. Por ejemplo:

- Cuando el sistema opera bajo condiciones de carga baja MTC, resulta más tolerante al aumento de U (obsérvese la curva cuando $N = 50$; desde $U = 40$, la $\overline{AoI}$ comienza a aumentar).
- En condiciones de sobrecarga del sistema ($N = 100$), se evidencia que con valores de $U > 15$, la entrega de los mensajes de actualización del estado a una aplicación se vea negativamente afectada.

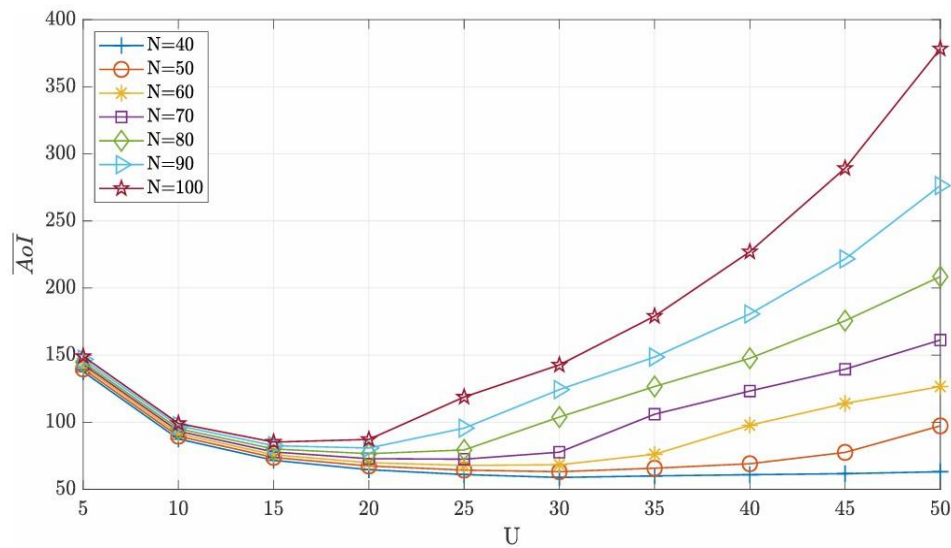


Figura 4. Patrón de generación de mensajes de actualización del estado Φ_1 . $\overline{AoI}$.

El comportamiento de la probabilidad de éxito en el acceso (P_s) en relación con el número de mensajes de actualización generados por dispositivo se muestra en la Figura 5, para diversas cargas del sistema. Esta métrica permite observar y determinar que, a partir de un cierto valor de U , el deterioro en el rendimiento del sistema se vuelve notorio (cuando $P_s < 0,9$). En concordancia con la evolución de la $\overline{AoI}$, la reducción de P_s señala el punto de inflexión a partir del cual un incremento adicional en U ocasiona un impacto negativo en el desempeño global del sistema.

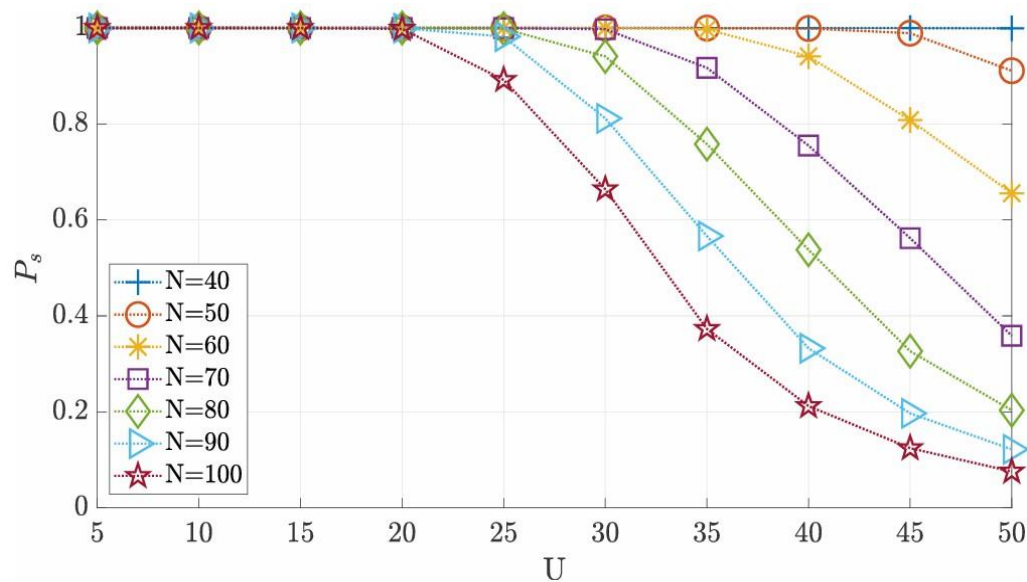


Figura 5. Patrón de actualización Φ_1 . Probabilidad de acceso exitoso P_s .



Por ejemplo:

- Cuando el sistema opera bajo condiciones de carga baja, muestra una mayor tolerancia frente al incremento de U . En el caso $N = 50$, se observa que a partir de $U = 45$, la probabilidad de éxito P_s comienza a decrecer.
- Cuando el sistema se encuentra en situación de sobrecarga ($N = 100$), se evidencia que para valores de $U > 15$, la entrega de mensajes de actualización del estado deteriore el rendimiento del sistema.

En la Figura 6 se ilustra el comportamiento del número promedio de intentos de transmisión (K) como función del número de mensajes de actualización del estado por dispositivo U , para diversas cargas del sistema. Como se observa:

- Bajo valores bajos de actualizaciones por segundo ($U < 10$), el número de intentos realizados por los dispositivos IoT que acceden exitosamente a la red es mayor a dos en todos los escenarios, excepto cuando $N = 40$.
- En una situación de sobrecarga ($N = 100$) y con frecuencia alta de actualización ($U = 50$), los dispositivos IoT deben realizar al menos cinco intentos para completar el procedimiento de acceso aleatorio (RA).

Por lo tanto, a mayor intensidad de mensajes de actualización del estado U , también aumenta el número promedio de transmisiones de preámbulo K , lo cual constituye un comportamiento esperado en escenarios de comunicación masiva tipo CIoT [16].

2. Patrón de generación de mensajes de actualización del estado: Phi2

Con respecto a la Aol promedio en el tiempo ($\overline{Aol}$), el patrón de generación de mensajes de actualización Phi2 presenta un comportamiento similar al del primer patrón analizado; es decir:

- Cuanto menor es la carga del sistema, mejor es la $\overline{Aol}$ en todos los escenarios para un mayor número de mensajes de actualización del estado por segundo.
- El punto de inflexión, en el cual la frecuencia de mensajes de actualización comienza a impactar de forma negativa en el rendimiento del sistema, se presenta bajo condiciones de sobrecarga, específicamente cuando $U > 15$ (ver Figura 7).

Se pueden observar algunas diferencias menores, pero no despreciables, con respecto al patrón Phi1 en las métricas de P_s (Figura 8) y K (Figura 9).

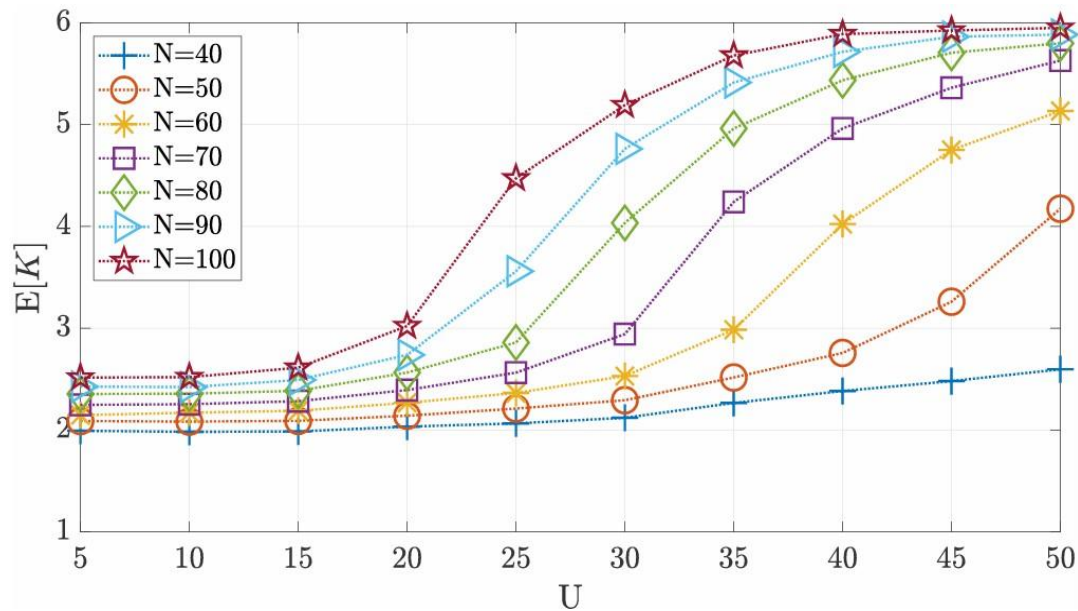


Figura 6. Patrón de generación de mensajes de actualización del estado Φ_1 . Número promedio de transmisiones de preámbulo requeridas para acceso exitoso $E[K]$.

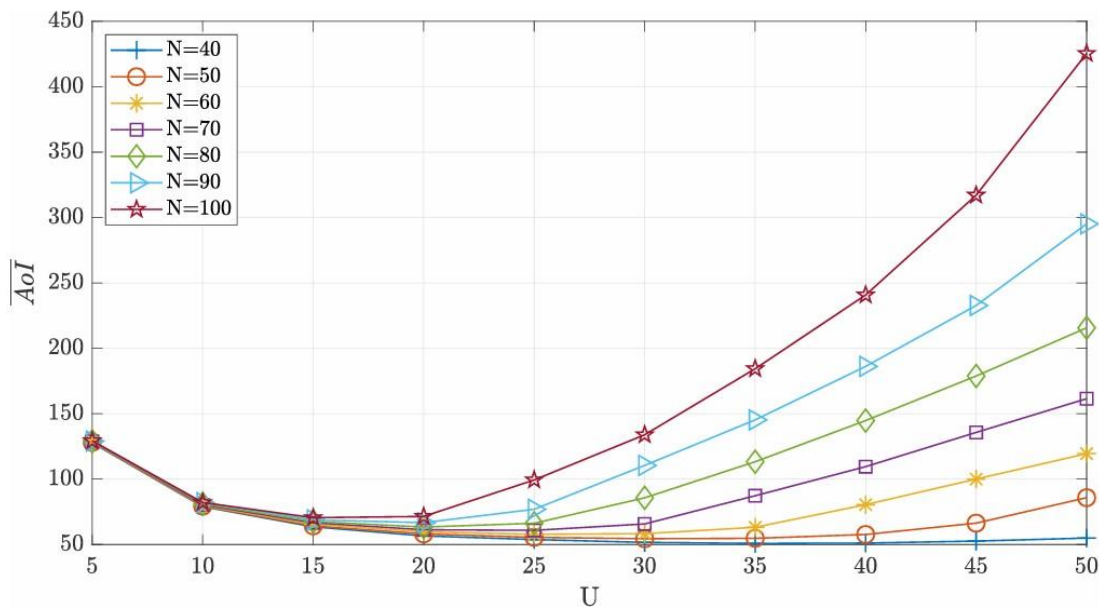


Figura 7. Patrón de generación de mensajes de actualización del estado Φ_2 . Aol .

El patrón de generación Φ_2 presenta mejor rendimiento para $U \leq 25$ y $N \leq 100$ en comparación con el patrón Φ_1 ($U \leq 20$), en relación con la métrica K . Cuando el sistema está sobrecargado, el patrón de generación Φ_2 permite hasta 20 mensajes de actualización del



estado por segundo con un retardo de acceso menor a 75 ms, lo cual es mejor que el patrón de generación Phi1, que solo tolera hasta $U \leq 15$.

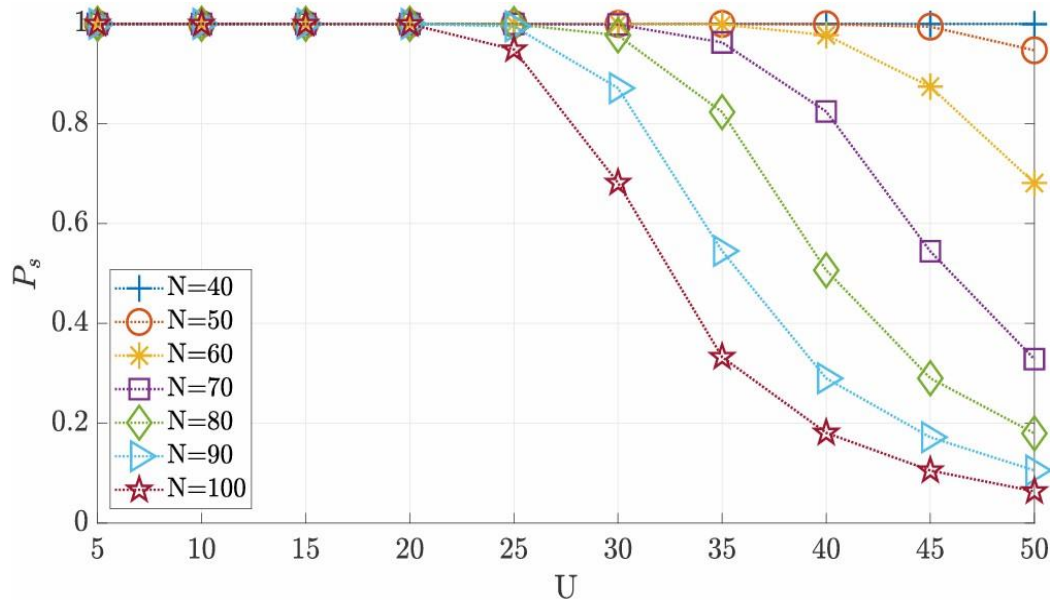


Figura 8. Patrón de generación de mensajes de actualización del estado Phi2. Probabilidad de acceso exitoso P_s .

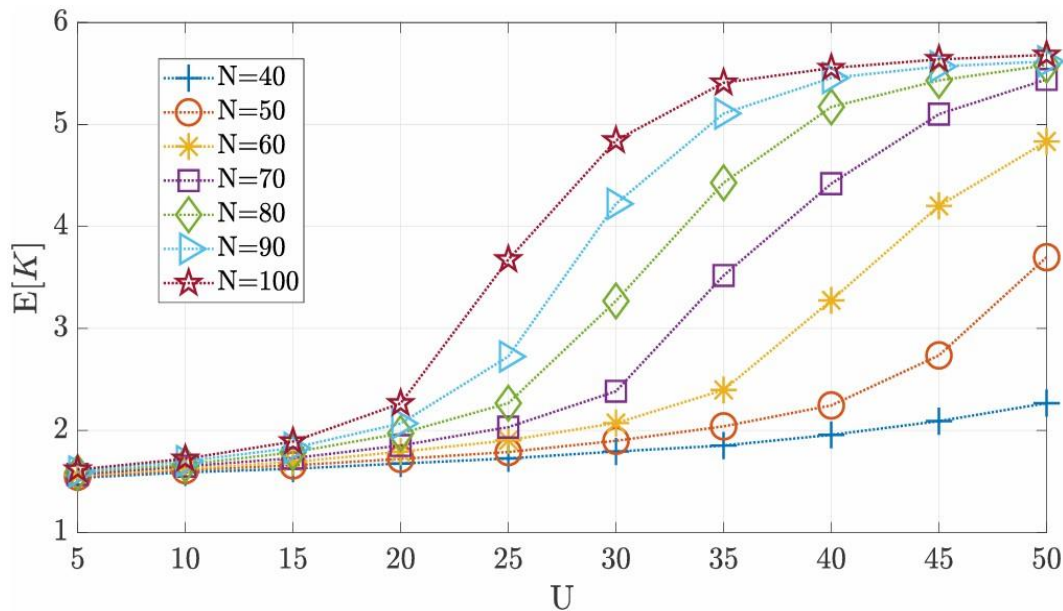


Figura 9. Patrón de generación de mensajes de actualización del estado Phi2. Número promedio de transmisiones de preámbulo requeridas para acceso exitoso $E[K]$.



DISCUSIÓN

Los resultados de este estudio proporcionan una visión profunda sobre el comportamiento de la Antigüedad de la Información (Aol) en entornos de redes LTE-A para aplicaciones IoT masivas, considerando tanto la variabilidad de los dispositivos conectados como las condiciones dinámicas de acceso al canal. Los resultados obtenidos muestran que, al incrementar la frecuencia de generación de actualizaciones por dispositivo (U) y el número de UEs, la probabilidad de acceso exitoso (P_s) disminuye, mientras que el número promedio de transmisiones de preámbulo $E[K]$ aumenta, lo que eleva la Aol del sistema. Este comportamiento es coherente con la teoría de colas aplicada a Aol y con el efecto de contienda en el PRACH reportado en la literatura (3), (8), (19). En nuestras simulaciones, el deterioro de P_s y el aumento de $E[K]$ se hacen notorios a medida que crece U , especialmente en escenarios de alta carga, lo que confirma el vínculo entre congestión en el acceso y pérdida de frescura de la información.

Al comparar los patrones Phi1 (dispositivos sincronizados) y Phi2 (no sincronizados), se evidencia que Phi2 ofrece una zona operativa más amplia antes del colapso por contienda: bajo sobrecarga, admite hasta 20 mensajes/s con un retardo de acceso menor a 75 ms, superando el umbral tolerado por Phi1. Este resultado sugiere que la desincronización de fuentes reduce las colisiones y preserva mejor la frescura del estado en la RAN (10), (12).

Estos hallazgos coinciden con trabajos previos sobre control de acceso ACB para mMTC en LTE-A (17), (18) y con análisis recientes sobre el impacto del tráfico de actualizaciones en la RAN (11). Además, extienden dicha evidencia al incorporar explícitamente Aol como KPI de diseño para delimitar regiones de operación seguras. La verificación cuantitativa mediante funciones MATLAB (por ejemplo, *aggregate_Aol*, *temporalMean_Aol*) reforzó la consistencia entre el marco analítico y las simulaciones de eventos discretos, incrementando la confiabilidad de los resultados.

CONCLUSIONES

Los resultados muestran que el incremento en la frecuencia de actualización (U) y en el número de dispositivos MTC (N) ocasiona un deterioro progresivo en el desempeño de la red LTE-A. En particular, la Antigüedad de la Información (Aol) tiende a aumentar, la probabilidad de acceso exitoso (P_s) disminuye y el número promedio de transmisiones de preámbulo (K) se incrementa. Estos hallazgos confirman que tanto U como N son factores limitantes que condicionan la eficiencia del canal de acceso aleatorio en escenarios de tráfico masivo IoT.

El análisis comparativo entre los patrones de generación de actualizaciones revela diferencias significativas. El esquema sincrónico (Phi1) concentra la transmisión de mensajes y conduce a un rápido deterioro del sistema en escenarios de alta carga. En contraste, el esquema asíncrono (Phi2) distribuye los intentos de acceso y logra mantener la estabilidad del sistema por más



tiempo, soportando hasta 20 mensajes/s con retardos menores a 75 ms antes de alcanzar el punto de colapso. Este resultado evidencia la relevancia de mecanismos de desincronización temporal en aplicaciones IoT masivas.

Finalmente, la investigación valida a la Aol como un KPI esencial para caracterizar y dimensionar redes celulares que proporcionan conectividad a aplicaciones IoT. La integración de simulación por eventos discretos y funciones específicas en MATLAB permitió obtener resultados consistentes con los modelos analíticos y con la literatura reciente. Este aporte constituye una contribución significativa al campo, ya que propone la Aol como métrica de referencia para el diseño de estrategias de gestión dinámica de recursos y priorización de información en sistemas celulares de próxima generación, orientados a aplicaciones críticas de monitoreo en tiempo real.

AGRADECIMIENTOS

Los autores agradecen el respaldo institucional y académico brindado por la Universidad Nacional de Chimborazo (UNACH) a lo largo del desarrollo de esta investigación. Asimismo, se agradece al Dr. Vicent Pla de la Universitat Politècnica de València, España, por su aporte científico en el modelizado matemático y análisis formal del estudio. A los docentes del programa de Maestría en Matemática Aplicada, mención Matemática Computacional, quienes con sus conocimientos y sugerencias aportaron al fortalecimiento de las competencias investigativas, lo cual contribuyó significativamente a enriquecer la calidad del estudio.

CONTRIBUCIONES DE AUTOR

La elaboración del presente artículo se realizó bajo un esquema colaborativo, en el cual cada autor asumió responsabilidades específicas:

- Ronald Patricio Vaca Guadalupe: formulación del problema de investigación, implementación de los simuladores en Python y MATLAB, procesamiento de datos, análisis detallado de resultados y redacción inicial del manuscrito.
- Dr. Luis Tello-Oquendo: supervisión general del trabajo, orientación metodológica, validación de resultados, revisión crítica del marco teórico y de la discusión, además de aportar sugerencias clave para la mejora y coherencia del documento.

En conjunto, las contribuciones de los autores permitieron que la investigación se desarrolle con un alto nivel de rigurosidad académica, integrando tanto la perspectiva teórica como la práctica, y logrando un documento sólido que aporta al análisis del desempeño de la Antigüedad de la Información (Aol) en redes LTE-A aplicadas al Internet de las Cosas.



REFERENCIAS BIBLIOGRÁFICAS

1. Condoluci M, Dohler M, Araniti G. Comunicaciones tipo máquina sobre sistemas 5G: desafíos y tendencias de investigación para aplicaciones industriales CPS. *Cyber-Phys Syst.* 2017;3(1):75-89.
2. Sinha D, Roy R. Programación de actualizaciones de estado para optimizar la antigüedad de la información en sistemas ciberfísicos industriales. *IEEE Access.* 2019;7:95677-95.
3. Kaul S, Yates R, Gruteser M. Real-time status: How often should one update? *Proc IEEE INFOCOM.* 2012;2731-5.
4. Guo C, Wang X, Liang L, Li GY. Age of information, latency, and reliability in intelligent vehicular networks. *IEEE Netw.* 2022;37(6):109-16.
5. Li J, Zhou Y, Chen H. Age of information for multicast transmission with fixed and random deadlines in IoT systems. *IEEE Internet Things J.* 2020;7(9):8178-91.
6. Tello-Oquendo L, Lin SC, Akyildiz IF, Pla V. Software-defined architecture for QoS-aware IoT deployments in 5G systems. *Ad Hoc Netw.* 2019;93:101911.
7. Santander D, Cicienia-Cárdenas K, Astudillo-Salinas F, Aranda J. Impacto del tiempo de backoff sobre el rendimiento de IoT celular en entornos de comunicación masiva. *Rev Digit Novasinergia.* 2022;5(1):17-30.
8. Bedewy AM, Sun Y, Shroff NB. Minimizing the age of information through queues. *IEEE Trans Inf Theory.* 2019;65(8):5215-32.
9. Li S, Li M, Chen R, Sun Y. Optimal transmission access scheduling for cellular IoT device status update with multi-antenna. *China Commun.* 2022;19(4):302-12.
10. Flores-Vázquez ME, Tello-Oquendo L, Pla V, Aranda J, Capella JV, Guamo Morocho AK. Impact of information update generation on the radio access network in cellular IoT. *Telecommun Syst.* 2023;84:409-20.
11. Zhang L, Yan L, Pang Y, Fang Y. FRESH: Freshness-aware and energy-efficient scheduler for cellular IoT systems. *Proc IEEE ICC.* 2019;1-6.
12. Tello-Oquendo L, Pacheco-Paramo D, Pla V, Martínez-Bauset J. Reinforcement learning mechanism for ACB in LTE-A networks handling massive M2M/H2H traffic. *Proc IEEE ICC.* 2018;1-7.
13. Pacheco-Paramo D, Tello-Oquendo L, Pla V, Martínez-Bauset J. Deep reinforcement learning for dynamic access control in wireless networks with mMTC. *Ad Hoc Netw.* 2019;94:101939.
14. Pacheco-Paramo D, Tello-Oquendo L. Delay-aware dynamic access control in wireless networks using deep reinforcement learning. *Comput Netw.* 2020;182:107493.
15. 3GPP. Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification. TS 36.321 V10.0.0. 2012.
16. 3GPP. Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation. TS 36.211 V12.3.0. 2014.



17. Tello-Oquendo L, Leyva-Mayorga I, Pla V, Martínez-Bauset J, Vidal JR, Casares-Giner V, et al. Performance analysis and optimal configuration of ACB parameters in LTE-A networks with massive M2M traffic. IEEE Trans Veh Technol. 2017;67(4):3505-20.
18. Vidal JR, Tello-Oquendo L, Pla V, Guijarro L. Performance study and enhancement of access barring for massive machine-type communications. IEEE Access. 2019;7:63745-59.
19. Ceran ET, Gunduz D, György A. Average age of information with hybrid ARQ under a resource constraint. IEEE Trans Wirel Commun. 2019;18(3):1900-13.
20. Kosta A, Pappas N, Angelakis V. Age of information: A new concept, metric, and tool. Found Trends Netw. 2017;12(3):162-259.
21. 3GPP. Study on RAN improvements for Machine-type Communications (MTC). TR 37.868 V11.0.0. 2011.

