

## Impacto del Jitter en un control de formación multiagente

John Anzola,<sup>a,\*</sup> Fredys A. Simanca<sup>b</sup>, Vicente García-Díaz<sup>c</sup>

<sup>a</sup>Departamento de Ingeniería Electrónica y Mecatrónica, Fundación Universitaria Los Libertadores, Carrera 16 # 63A-68, Bogotá, D.C., Colombia.

<sup>b</sup>Facultad de Ingeniería, Universidad Cooperativa de Colombia, Avenida Caracas # 37-63, Bogotá, D.C., Colombia.

<sup>c</sup>Departamento de Informática y Ciencias de la Computación, Universidad de Oviedo, C/Calvo Sotelo s/n. 33007 Oviedo, España.

**To cite this article:** Anzola, J., Simanca, F., García-Díaz, V. 2024. Jitter impact in a multi-agent formation control. Revista Iberoamericana de Automática e Informática Industrial 21, 17-28. <https://doi.org/10.4995/riai.2023.19440>

### Resumen

En este artículo se analizan las características de una comunicación inalámbrica WiFi que emplea el protocolo TCP, incluyendo sus mecanismos de retransmisión de paquetes y DCF. El enfoque del estudio es el análisis de la influencia de un esquema de control de acceso al medio no prioritario en la formación robótica multiagente centralizada en el líder. Se examina específicamente el impacto del Jitter en el control de la formación, ya que cada paquete transmitido por el agente líder contiene información crítica sobre su posición objetivo. Las variaciones temporales en la entrega de estos paquetes pueden causar variaciones en la posición de los agentes seguidores, lo que a su vez puede afectar el mantenimiento de la formación con cierto grado de error. Cada agente robótico en la formación está compuesto por un robot Pioneer 3-DX y un control PID que le permite desplazarse hacia un punto objetivo bajo las restricciones no holonómicas. Para analizar el impacto del Jitter, se presenta un entorno de simulación que incorpora fuentes de tráfico externo como señal de perturbación que varía la entrega de paquetes, lo que afecta el control de formación robótica multiagente.

*Palabras clave:* Control automático de formaciones, sistema multiagente, Jitter, retardo, entorno de simulación.

### Jitter impact in a multi-agent formation control

#### Abstract

This article analyzes the characteristics of a WiFi wireless communication employing the TCP protocol, including its packet retransmission mechanisms and DCF. The focus of the study is the analysis of the influence of a non-priority medium access control scheme on centralized multi-agent robotic formation with a leader. Specifically, the impact of Jitter on formation control is examined, as each packet transmitted by the leader agent contains critical information about its target position. Temporal variations in the delivery of these packets can cause variations in the positions of the follower agents, which in turn can affect the maintenance of the formation with some degree of error. Each robotic agent in the formation consists of a Pioneer 3-DX robot and a PID controller that enables it to move towards a target point under non-holonomic constraints. To analyze the impact of Jitter, a simulation environment is presented that incorporates external traffic sources as disturbance signals that vary the packet delivery, thereby affecting the multi-agent robotic formation control.

*Keywords:* Automatic control of formations, multi-agent systems, Jitter, delay, simulation entorno.

### 1. Introducción

La tecnología de red inalámbrica Wi-Fi es una de las más extendidas a nivel mundial, que ha permitido la conectividad de millones de dispositivos móviles, como computadoras, teléfo-

nos inteligentes, impresoras, cámaras y, en general, con el Internet de las Cosas (IoT-Internet de las Cosas), cualquier dispositivo o agente que interactúe con Internet o con una Red de Área Local Inalámbrica (WLAN-Red de Área Local Inalámbrica).

\*Autor para correspondencia: [jpanzola@libertadores.edu.co](mailto:jpanzola@libertadores.edu.co)

Sin lugar a dudas, la robótica móvil no es ajena a esta tecnología, ya que el intercambio de información e interacción con otros robots o agentes requieren obligatoriamente de una red de comunicación inalámbrica.

Las redes de comunicaciones inalámbricas Wi-Fi emplean el estándar IEEE 802.11, cuya función principal es controlar el acceso al medio inalámbrico y coordinar la transmisión de datos de la forma más eficiente posible. Para cumplir con esta función, el estándar IEEE 802.11 utiliza el Control de Acceso al Medio (MAC - Control de Acceso al Medio), en conjunto con la Función de Coordinación Distribuida (DCF - Distributed Coordination Function) y el algoritmo Acceso Múltiple con Detección de Portadora y Evitación de Colisiones (CSMA/CA - Carrier Sense Multiple Access with Collision Avoidance). Estos mecanismos permiten que un agente que desee transmitir en un medio inalámbrico y compartido comience por escuchar el canal para verificar su actividad. Si el canal está libre, el agente puede comenzar a transmitir de inmediato (Cheng et al., 2019).

Sin embargo, en la dinámica y naturaleza del medio inalámbrico, se presentan procesos que no se pueden controlar en el acceso. Entre ellos, destacan la fluctuabilidad en los enlaces de radio debido al límite del rango de cobertura y la potencia de transmisión, los cuales dependen del nivel de energía de cada dispositivo. Además, existe la posibilidad de que se produzcan retransmisiones debido a colisiones entre paquetes. Estas colisiones ocurren cuando dos o más agentes intentan enviar datos simultáneamente, conteniendo así por el acceso al medio.

El estándar cuenta con el mecanismo DCF, el cual se encarga de administrar el canal y regular el acceso al mismo, con el fin de proporcionar una solución equitativa entre todos los agentes que desean acceder. Durante este proceso de acceso, pueden surgir problemas de retardos que forman parte de la latencia del sistema en general, tanto en sistemas de control como de comunicaciones.

La latencia se refiere a la suma de todos los retardos temporales presentes dentro de una red, ya sea debido a los efectos de los tiempos de propagación o a la transmisión de paquetes. El retardo en la transmisión de paquetes está influenciado por el tamaño del paquete transmitido, el tamaño del búfer en cada agente y el tiempo que tarda un paquete en procesar dicho agente. Esto puede generar retardos debido a la congestión de paquetes en cola (Peng, 2018).

Desde el punto de vista de los agentes que desean acceder al canal y se consideran sistemas de control, experimentan retardos temporales debido al procesamiento, muestreo y acciones de control. Hasta este punto, tanto los sistemas de comunicaciones como los sistemas de control comparten la problemática común del retardo. Esta variable es tan sensible que puede provocar que la acción de control en el sistema no se realice a tiempo, lo que resulta en desfases que pueden llevar al sistema a la inestabilidad y dar lugar a acciones erráticas en sus actuadores.

En la actualidad, los sistemas de control y comunicación no están separados, lo que implica que un problema que afecte a uno de los sistemas también afectará al otro. Del mismo modo, la presencia de un retardo en un sistema de comunicación influye en cierta medida en la acción de control. Un efecto de causalidad del Jitter es precisamente el retardo. El Jitter se define como la variabilidad temporal entre los intervalos de tiempo

entre paquetes, tanto en los procesos de transmisión como de recepción.

Para contextualizar la problemática, consideremos un paquete de datos que contiene el valor de una acción de control. Si el envío de estos paquetes se realiza con una frecuencia uniformemente distribuida (es decir, con una velocidad de transmisión constante), el sistema de control se mantiene estable. Sin embargo, si la frecuencia de envío de los paquetes disminuye, se produce un aumento en el retardo, lo cual afecta la frecuencia de las acciones de control y puede llevar a la inestabilidad del sistema.

El lapso temporal entre un paquete y el siguiente, o entre una señal de control y la siguiente, se traduce en un retardo en el sistema de comunicación que afecta directamente el tiempo de envío de la señal de control. La medición de esta variación temporal se conoce como Jitter (Tardioli et al., 2019).

Como contribución en este artículo, se destaca el impacto causal del efecto del Jitter en un sistema de comunicación Wi-Fi y un sistema de control. Se ha seleccionado un sistema de control de formaciones, ya que el seguimiento y control de un patrón de formación entre varios agentes (robots) implica que la posición sea una variable crítica en el sistema. En un sistema de formación multiagente, cualquier variación en las señales de control puede alterar la posición y, por lo tanto, la formación.

Se eligió un sistema de control de formación debido a su alta sensibilidad a los retardos (Hu et al., 2022b; Chen et al., 2022). En este sentido, el Jitter, como medida de la variabilidad temporal entre señales de control, aumenta el retardo. Esto se debe a que, en el estándar IEEE 802.11, se producen eventos de transmisión simultánea de paquetes que generan colisiones y requieren la retransmisión de paquetes. Otro evento que implica la retransmisión de paquetes es la transmisión o recepción en el límite del área de cobertura. La energía está asociada de manera proporcional al área de cobertura para la transmisión o recepción. Estos eventos y variables mencionados son ejemplos de variables aleatorias que no se pueden controlar y que generan retransmisiones de paquetes, lo que resulta en el aumento de retardos aleatorios.

El sistema de formación evaluado en el artículo está compuesto por cuatro agentes robóticos homogéneos, con uno de ellos desempeñando el rol de agente líder y los otros tres actuando como agentes seguidores. Se seleccionó el robot no holonómico Pioneer 3-DX como agente robótico. El esquema de comunicación es centralizado en el líder, formando así una topología estrella alrededor de él. El líder es responsable de transmitir las posiciones objetivo a los agentes seguidores. Todos los agentes están equipados con un controlador PID. El agente líder tiene previamente definidas las coordenadas objetivo que forman una trayectoria. A medida que el agente líder avanza, calcula las posiciones objetivo de la formación y las transmite a los agentes seguidores, quienes se desplazarán hacia el punto indicado por el líder.

En este sentido, cada paquete transmitido por el líder se convierte en una señal de control para cada agente seguidor, cuyo controlador debe reaccionar en tiempos cortos. Cualquier alteración en estas señales de control afectará la velocidad de los agentes, llevándolos desde un punto de estabilidad hacia la zona de inestabilidad del sistema. Por lo tanto, la variabilidad temporal impactará el esquema de formación.

Para evaluar el impacto del Jitter en un entorno multiagente robótico, se utilizó el software de simulación CoppeliaSim. La Red de Área Local Inalámbrica (WLAN - Wireless Local Area Network) bajo el estándar IEEE 802.11 fue simulada en el simulador de eventos discretos NS-2 (Network Simulator), configurando la red inalámbrica con el método DCF. Para establecer la comunicación entre las dos plataformas de simulación, se empleó Python a través de una Interfaz de Programación de Aplicaciones (API - Application Programming Interface) de CoppeliaSim.

Dentro del experimento, se plantearon fuentes de tráfico externas que generaran una carga de peticiones al agente líder, ya que el estándar IEEE 802.11 con el esquema DCF no discrimina ni prioriza ningún paquete. El agente líder acepta todo el tráfico de datos externo, lo que incrementa el tiempo que los paquetes destinados a los agentes seguidores pasan en el búfer, aumentando así su tiempo de entrega. De esta manera, se afecta el Jitter y se observa el impacto que genera en el control de la formación multiagente.

El artículo está estructurado de la siguiente manera: la sección 2 aborda los preliminares matemáticos relacionados con la cinemática diferencial del robot y las consideraciones de un algoritmo de control Proporcional Integral Derivativo (PID - Proportional Integral Derivative). En la sección 3, se exponen las consideraciones del mecanismo DCF. La sección 4 presenta el diseño y las consideraciones del control de formaciones, describiendo el planteamiento del experimento computacional y sus métricas de evaluación. La sección 5 contiene el análisis de los resultados obtenidos. Por último, la sección 6 se dedica a las conclusiones.

## 2. Seguimiento de trayectoria para robots no holonómicos

Los robots holonómicos u omnidireccionales tienen la capacidad de cambiar instantáneamente su dirección, lo que les permite girar en cualquier dirección y sentido sin necesidad de realizar giros, rotaciones o alineaciones previas. Por el contrario, los robots no holonómicos requieren realizar una serie de maniobras previas para poder rotar o girar y así alinearse en la dirección objetivo. Esta distinción se basa en la capacidad de los robots para moverse de manera flexible y eficiente en diferentes direcciones y entornos Sánchez-Sánchez and Arteaga-Pérez (2020); Jiménez et al. (2020).

En esta sección, se proporciona una descripción de la cinemática de un robot diferencial que consta de dos ruedas. Este tipo de robot tiene un movimiento no holonómico, lo que implica que no puede moverse lateralmente hacia la izquierda o hacia la derecha. En cambio, su movimiento está limitado a avanzar en la dirección determinada por la velocidad relativa de sus ruedas. La dirección y la velocidad del robot se controlan mediante la velocidad angular de cada rueda, lo que permite ajustar el giro y la trayectoria del robot. A través de este modelo de robot diferencial, se puede explorar y comprender cómo los diferentes valores de velocidad de las ruedas afectan el movimiento global y la trayectoria del robot en un entorno dado.

### 2.1. Cinemática diferencial

Un robot de tracción diferencial en un plano, con dos ruedas controlables y una rueda libre, puede planificar una trayectoria

mediante el control de la velocidad total ( $v$ ) y orientación ( $\theta$ ), como se muestra en la Figura 1(a).

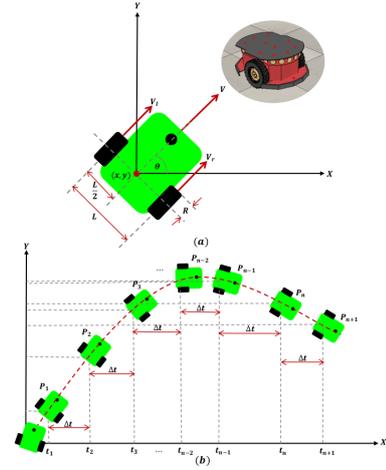


Figura 1: Robot diferencial. (a) Parámetros cinemáticos. (b) Consideraciones de movimiento.

En la Figura 1(a),  $v_r$  representa la velocidad de la rueda derecha,  $v_l$  la velocidad de la rueda izquierda,  $L$  la distancia entre el centro de las ruedas,  $\theta$  el ángulo de orientación del robot dentro de un sistema referencial de coordenadas y  $R$  el radio de las ruedas.

Los parámetros presentados en la Figura 1(b), se realizan bajo las siguientes suposiciones:

1. El robot se desplaza de un punto  $P_n$  a  $P_{n+1}$  con velocidad constante ( $V$ ).
2. La duración del desplazamiento entre puntos es dinámica en un período de tiempo  $\Delta t$ , que está en función del tiempo existente entre paquetes.
3. Las ruedas de los robots no se resbalan y la superficie de movimiento del robot es plana.

Bajo las suposiciones anteriores, se puede plantear que la velocidad del robot es el promedio de las velocidades de las ruedas individuales y viene dada por la ecuación (1).

$$V = \frac{V_r + V_l}{2} \quad (1)$$

Estas suposiciones se hacen porque las velocidades individuales son componentes variables y cambian con el paso del tiempo. En este sentido, las ecuaciones cinemáticas que se describen en esta sección se utilizan para transformar el planteamiento del robot como un sistema de coordenadas polares ( $r, \theta$ ) a un sistema de coordenadas rectangulares ( $x, y$ ).

Las entradas del sistema requeridas para el movimiento de un robot diferencial son la velocidad lineal total ( $V$ ) y la orientación ( $\theta$ ). La tasa de cambio de posición del robot en la dirección  $x$  es  $\dot{x}$  y en la dirección  $y$  es  $\dot{y}$ , dadas por la ecuación (2).

$$\dot{x} = V \cos \theta \quad \dot{y} = V \sin \theta \quad (2)$$

La velocidad angular  $\omega$  del robot viene dada por (3).

$$\dot{\theta} = \omega = \frac{V_r - V_l}{L} \quad (3)$$

Sustituyendo la velocidad lineal total ( $V$ ) de (1) en (2), se obtiene (4).

$$\dot{x} = \frac{V_r + V_l}{2} \cos \theta \quad \dot{y} = \frac{V_r + V_l}{2} \sin \theta \quad (4)$$

La velocidad ( $V$ ) del robot en un punto fijo dentro del sistema de coordenadas está dada por (5).

$$V = \sqrt{\dot{x}^2 + \dot{y}^2} \quad (5)$$

Remplazando (4) en (5) se obtiene (6).

$$V = \sqrt{\left(\frac{V_r + V_l}{2} \cos \theta\right)^2 + \left(\frac{V_r + V_l}{2} \sin \theta\right)^2} = \frac{V_r + V_l}{2} \quad (6)$$

Las velocidades individuales  $V_r$  y  $V_l$  expresadas en (7), se tratarán como señales de control y se obtienen de (6) y (3).

$$V_r = V + \frac{L}{2} \omega \quad V_l = V - \frac{L}{2} \omega \quad (7)$$

## 2.2. Control PID

En esta sección se describe el algoritmo de control digital PID empleado, con los parámetros ajustados a la estructura genérica de un robot móvil diferencial no holonómico (Lin et al., 2021; Zhou et al., 2021). La expresión discreta del controlador PID se muestra en la ecuación (8).

$$u(k) = K_p e(k) + K_i \sum_{j=1}^k e(j) + K_d (e(k) - e(k-1)) \quad (8)$$

El coeficiente integral ( $K_i$ ) está dado por  $K_i = K_p \frac{T}{T_i}$ . El coeficiente diferencial ( $K_d$ ) está dado por  $K_d = K_p \frac{T_d}{T}$ . Donde  $T$ ,  $T_d$ , y  $T_i$  corresponden al período de muestreo, la constante de tiempo diferencial y la constante de tiempo integral, respectivamente.

Para el término integral, debido a limitaciones de espacio de almacenamiento, se utiliza una aproximación que retiene todos los datos de error en una sola variable. El cálculo del término integral se realiza mediante una expresión incremental, como se muestra en la ecuación (9), de la siguiente forma:

$$\begin{aligned} \Delta u(k) &= u(k) - u(k-1) = c_p + c_i + c_d \\ c_p &= K_p (e(k) - e(k-1)) \\ c_i &= K_i e(k) \\ c_d &= K_d (e(k) - 2e(k-1) + e(k-2)) \end{aligned} \quad (9)$$

Reduciendo la expresión (9) del algoritmo PID incremental, se tiene:

$$u(k) = u(k-1) + c_p + c_i + c_d \quad (10)$$

## 3. Operación IEEE 802.11

El estándar IEEE 802.11 se basa en CSMA/CA y se ha definido principalmente como una tecnología de conectividad de control de acceso de corto alcance, convirtiéndose en la tecnología de conectividad más extendida en el mundo. Para comprender su funcionamiento, es necesario abstraerlo a través de un modelo de capas, como se muestra en la Figura 2. El estándar

especifica parámetros relacionados con la frecuencia, esquemas de modulación, ancho de banda, modo de acceso y velocidades máximas en la capa física (PHY - Physical) para un medio o canal inalámbrico.

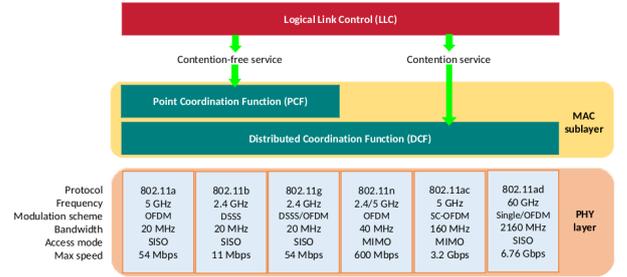


Figura 2: Control de acceso IEEE 802.11.

La capa MAC, responsable de coordinar el acceso al canal o medio inalámbrico, se subdivide en dos subcapas que contienen dos métodos de acceso al canal: DCF y la Función de Coordinación Punto a Punto (PCF - Point Coordination Function). Además, existe un paso obligatorio en la subcapa de Control de Enlace Lógico (LLC - Logical Link Control), que es responsable del control de enlace lógico y realiza funciones como el control de errores, control de flujo, diálogo, entramado y direccionamiento MAC. Esta subcapa proporciona servicios entre las capas de red y PHY (kumar Gupta and Venkatesh, 2022).

### 3.1. Módulo DCF

DCF es el mecanismo central utilizado por un nodo para determinar si puede transmitir datos. Una estación que utiliza DCF emplea un temporizador de espera aleatorio (random back-off timer) para detectar si el canal de comunicación está ocupado. Si el canal está ocupado, la estación debe esperar un período de tiempo aleatorio antes de intentar acceder nuevamente al medio o canal. Este mecanismo garantiza que múltiples estaciones que desean enviar datos no transmitan al mismo tiempo. El retardo aleatorio hace que las estaciones esperen diferentes períodos, evitando que todas las estaciones detecten el canal inactivo exactamente al mismo tiempo y colisionen entre sí. El temporizador de espera aleatorio reduce significativamente el número de colisiones y las correspondientes retransmisiones, especialmente cuando aumenta el número de nodos activos (Manzoor et al., 2020).

En este mecanismo, cuando un nodo está listo para transmitir datos, activa un temporizador de espera aleatorio (random back-off timer) que se elige de manera uniforme en el rango de  $[0, W - 1]$ , para una Ventana de Contención ( $CW$  - Containment Window). Inicialmente, el valor de  $CW$  se establece en un valor mínimo  $CW_{\min}$ , lo que permite detectar si el canal está inactivo durante un período de espera distribuido entre tramas ( $T_{DIFS}$  - Distributed Inter Frame Spacing). Después de esto, el temporizador de espera aleatorio comienza a disminuir su valor. Durante este proceso, el temporizador puede detenerse si otros nodos están transmitiendo o comienzan a transmitir datos en el canal. Si la ocupación del canal persiste, el temporizador de espera se reiniciará cuando el canal esté nuevamente inactivo. Una vez que el temporizador expira, el nodo comienza a enviar paquetes de datos.

Cuando el receptor recibe la trama de datos, espera un período de tiempo de espacio corto entre tramas ( $T_{SIFS}$  - Short Inter Frame Spacing) y envía una trama de acuse de recibo (ACK - Acknowledgment). Si el remitente recibe la trama ACK, se considera que el paquete se transmitió exitosamente. En caso de producirse una colisión o si los paquetes de datos o la trama de ACK se interrumpen, el remitente duplica su ventana de contención, asegurándose de no exceder un valor máximo de tamaño de ventana ( $CW_{max}$ ). Luego, se establece un nuevo temporizador de espera aleatorio y se intenta transmitir de nuevo hasta que el paquete se transmita correctamente o hasta que se supere un umbral de tiempo de retransmisión (RETRY), momento en el cual el paquete se descarta.

Complementariamente, si las demás estaciones de la red WLAN detectan que una trama proviene de la estación transmisora y receptora, configuran su campo de duración del vector de asignación de red (NAV - Network Allocation Vector) y esperan hasta que este expire. Una descripción gráfica de lo descrito anteriormente se puede observar en la Figura 3(a).

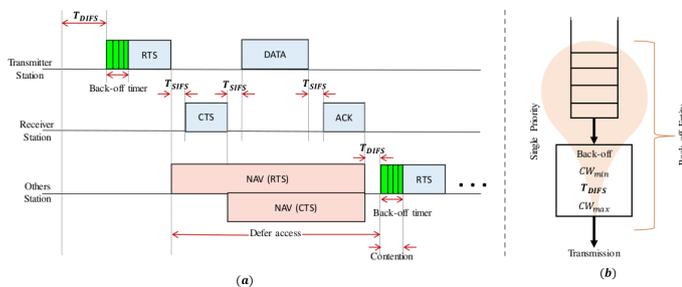


Figura 3: Secuencia de eventos DCF.

### 3.2. Módulo PCF

PCF es un esquema de acceso basado en sondeo (polling), sin contención. Esto quiere decir que PCF admite peticiones de servicios con límites de tiempo y acceso prioritario al medio inalámbrico, coordinado por una estación denominada coordinador de puntos (PC - Point Coordinator). PCF tiene mayor prioridad que DCF, ya que puede iniciar transmisiones en un tiempo menor que  $T_{DIFS}$  en DCF, denominado para este esquema  $T_{PIFS}$ , conformando estos dos tiempos un periodo denominado superframes. Para acceder al medio, PCF emplea un periodo sin contención (CFP - Contention Free Period), mientras que DCF admite un periodo con contención (CP - Contention Period), alternándose ambos a lo largo del tiempo, como se muestra en la Figura 4.

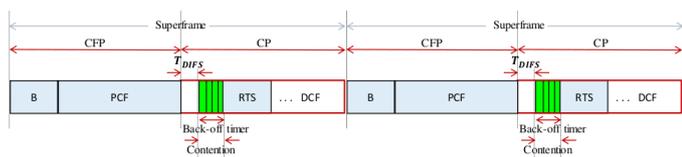


Figura 4: Supertrama IEEE 802.11.

Un supertrama comienza con una trama denominada "Beacon" (B) seguida de las tramas PCF y DCF. La trama Beacon es una trama de gestión que mantiene la sincronización de los temporizadores en las estaciones y proporciona parámetros relacionados con el protocolo. El PC, que está asociado con la función

de Punto de Acceso (AP - Access Point), genera tramas Beacon a intervalos regulares. De esta manera, cada estación sabe cuándo comenzará la siguiente supertrama (Zheng et al., 2018).

## 4. Descripción del experimento

Para analizar el impacto que tiene el Jitter en un control de formación multiagente, en esta sección se plantean las restricciones en un escenario de formación multiagente y su red de transmisión de datos. Se describen las métricas de evaluación y el entorno integrador que contiene el modelo de experimentación.

### 4.1. Consideraciones multiagente

El control de formación multiagente está conformado por un grupo de cuatro agentes robóticos que realizan un patrón geométrico triangular. Cada agente robótico está compuesto por un robot Pioneer 3-DX. El control de formación estará guiado y controlado por un agente que está posicionado en el centro de la formación, con tres agentes a su alrededor. Cada agente contiene un controlador PID descrito en la sección 2.2. El agente controlador o líder calcula y transmite las posiciones objetivo a cada uno de sus agentes.

La plataforma de simulación multiagente utilizada es CoppeliaSim. Esta plataforma cuenta con una API remota que permite la comunicación con aplicaciones externas, lo que permite ejecutar diferentes procesos en una máquina con conectividad local (aplicaciones instaladas en la misma máquina) o en una red de área local (LAN) o en Internet a través de un socket TCP. CoppeliaSim es una plataforma multiplataforma que ofrece diferentes modelos de simulación robótica, incluyendo el robot móvil Pioneer 3-DX. Dentro del modelo del Pioneer 3-DX en CoppeliaSim, se incluyen sensores de posición y orientación absoluta para el proceso de retroalimentación, así como los modelos matemáticos de los objetos que conforman el escenario de simulación, teniendo en cuenta características como la gravedad, el peso, el volumen y las superficies.

### 4.2. Consideraciones de red

CoppeliaSim no integra modelos ni protocolos de comunicación de redes. Por lo tanto, para el análisis de redes se emplea NS2, que es un software de simulación de eventos discretos que cuenta con una variedad de componentes de red y protocolos implementados. En particular, se utiliza el esquema WLAN IEEE 802.11 con su componente MAC y DCF, y se emplea el mecanismo de acceso básico predeterminado de solicitud y autorización (RTS/CTS - Request-To-Send/Clear-To-Send) sobre el modelo del protocolo de control de transmisión/Protocolo de Internet (TCP/IP - Transmission Control Protocol/Internet Protocol).

En el proceso de simulación, se asume que cada agente está equipado con un hardware de comunicación de bajo costo con recursos limitados, en el cual está implementado el protocolo IEEE 802.11b. Se considera que cada agente interactúa dentro de un modelo de propagación omnidireccional con sombreado (shadowing) y con un área de cobertura de diez metros. Otros parámetros de simulación se encuentran detallados en la Tabla 1.

Tabla 1: Consideraciones de red y multiagente.

| Parámetros de Simulación      |                                |
|-------------------------------|--------------------------------|
| Agentes                       | 3 seguidores y un agente líder |
| Fuentes de tráfico background | 9                              |
| Modelo de propagación         | Two Ray Ground                 |
| Capa física                   | Wireless Phy                   |
| Protocolo de enrutamiento     | IP                             |
| Protocolo de Transporte       | TCP                            |
| Interface de encolamiento     | DropTail                       |
| Longitud de la cola           | 10                             |
| Velocidad de los agentes      | 1.2 - 1.8 m/s                  |
| Tiempo de simulación          | 20s                            |
| Tamaño del paquete            | 50                             |

La comunicación es centralizada en el nodo líder, formando una topología en estrella. Con el fin de evaluar el impacto del Jitter en el control de formación, se emplean diez fuentes de tráfico externas, las cuales serán activadas una por una, evaluando métricas que permiten visualizar el impacto generado por el tráfico en el control de formación. Por otro lado, los nodos en NS2 responden a las mismas condiciones de posición y velocidad que los agentes en CoppeliaSim.

#### 4.3. Métricas de evaluación

El entorno de simulación cuenta con un proceso de supervisión de las variables de retardo, Jitter y posición de los agentes. Estas medidas permiten analizar el estado, la capacidad y el rendimiento del control PID y su respuesta al tratamiento que le da DCF en IEEE 802.11 en reacción al tratamiento dado a las fuentes de tráfico y cómo estas impactan la medida del retardo y, por ende, el Jitter. A continuación, se dará una breve descripción de estas métricas.

##### 4.3.1. Latencia promedio

Se considera como latencia en la red la suma de todos los retardos temporales que pueden ocurrir debido a factores y variables como la demora en la propagación y transmisión de paquetes, el tamaño de los paquetes transmitidos, el tamaño del búfer de cada nodo, el tiempo promedio que transcurre desde el envío de un paquete hasta su recepción, y el tiempo de procesamiento y ejecución de cada paquete en cada agente (Bhatia et al., 2022; William et al., 2022). Para estimar la latencia, se utiliza un modelo de caja negra que mide el tiempo que tarda un paquete desde el momento en que es enviado por el agente líder hasta que es recibido por un agente de la formación, teniendo en cuenta todos los retardos presentes en la trayectoria del paquete. En este sentido, la latencia promedio  $E2E_{AVG}$  está dada por 11.

$$E2E_{AVG} = \frac{\sum_{i=0}^n TR_i - TS_i}{n + 1} \quad (11)$$

Donde  $TR_i$  es el instante de tiempo en el que el agente seguidor recibe el  $i$ -ésimo paquete de datos proveniente del agente líder, y  $TS_i$  es el instante de tiempo en el que el agente líder transmite el  $i$ -ésimo paquete.  $n$  representa el número total de paquetes transmitidos.

##### 4.3.2. Jitter

El Jitter se utiliza para estimar la variabilidad temporal en un enlace de comunicación y se define como la diferencia de tiempo entre la llegada de un paquete y el paquete anteriormente transmitido (Hu et al., 2022a; Choi and Lee, 2019) y se calcula en 12.

$$Jitter = \frac{1}{n-1} \sum_{i=2}^n |TR_i - TR_{i-1}| \quad (12)$$

Donde  $TR_i$  representa el instante de tiempo en el que se recibe el  $i$ -ésimo paquete. El Jitter se promedia a lo largo de los paquetes recibidos, excepto para el primer paquete, ya que no hay paquete anterior con el cual calcular la diferencia de tiempo.

##### 4.3.3. Error de la posición

En el seguimiento de una trayectoria dada por un patrón geométrico de control de formación, se tiene que la trayectoria patrón está compuesta por un vector de posiciones objetivo  $[x_o, y_o]$  y por un vector de posición sensada  $[x_m, y_m]$ . Con el valor de estos vectores se obtiene un error geométrico de distancia, estimado mediante el error cuadrático (SE) como métrica evaluadora de la variabilidad de la distancia y está dado por 13.

$$SE = (|y_o - y_m| - |x_o - x_m|)^2 \quad (13)$$

#### 4.4. Entorno de simulación

Dentro de la etapa de experimentación, se diseñó un entorno de experimentación como se muestra en la Figura 5. Este entorno proporciona una estructura base para la construcción de escenarios de simulación y la evaluación de métricas. Sirve como punto de partida para la organización y desarrollo de pruebas que permiten evaluar el impacto del Jitter en el control de formación multiagente, tal como se propone en el objetivo del artículo.

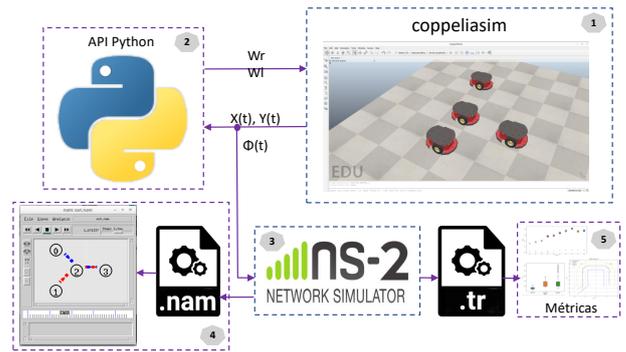


Figura 5: Entorno de simulación.

El entorno representado en la Figura 5 consta de tres módulos principales: el entorno de simulación robótica (1) que utiliza CoppeliaSim, el entorno de control (2) implementado en Python y el entorno de simulación de red (3) basado en NS2. Cada uno de estos entornos incorpora las consideraciones descritas en las secciones 4.1 y 4.2.

La comunicación entre los módulos del entorno se establece en modo standalone.

El entorno de simulación robótica (1) establece una conexión con el localhost, es decir, con el propio equipo, utilizando el puerto 19999. A través de esta conexión, se transmiten los parámetros de posición relativa  $x(t)$ ,  $y(t)$  y ángulo  $\theta(t)$  al entorno de control (2) mediante el API de Python. Una vez que Python recibe los parámetros de entrada, los utiliza como entrada para el algoritmo PID descrito en la sección 2.2, y luego devuelve las variables de control  $w_r$  y  $w_l$  a CoppeliaSim para cerrar el lazo de control.

En el proceso de transmisión de parámetros, las posiciones de los agentes robóticos en CoppeliaSim se traducen a posiciones de nodos en NS2 utilizando comandos como “\$node\_(\$i) set X\_ [expr x(t)], \$node\_(\$i) set Y\_ [expr y(t)]”. Estas posiciones interpretadas por NS2 en el entorno de simulación de red (3) se consideran como paquetes de datos. NS2 actúa como un módulo de monitoreo entre los entornos de simulación robótica (1) y control (2), por lo que no tiene ninguna influencia sobre las señales de control.

Durante la simulación en NS2, se generan dos tipos de trazas. El módulo (4) utiliza un archivo con extensión “.nam” para visualizar la simulación de manera gráfica. Por otro lado, el módulo (5) utiliza un archivo con extensión “.tr” para extraer datos y métricas de la simulación. El módulo (4) se ejecuta directamente para visualizar la simulación, mientras que el módulo (5) filtra los datos utilizando scripts en formato “.awk” para visualizar las métricas de red específicas.

#### 4.5. Escenario de evaluación

Como se mencionó en la sección 4.1 (consideraciones) para el control de formación multiagente, el sistema está compuesto por cuatro robots, con la formación representada en la Figura 6. En esta formación, el agente  $a_0$  actúa como líder, mientras que los agentes  $a_1$ ,  $a_2$  y  $a_3$  actúan como seguidores.

Cada agente tiene una distancia de seguridad  $d_{i,j}$  entre ellos, donde  $i$  representa el subíndice de los agentes seguidores y  $j$  es el subíndice del agente líder. En este caso, se establece que  $d_{1,0} = d_{2,0} = d_{3,0} = d$ , lo que implica que todos los agentes seguidores mantienen una distancia de seguridad de  $d$  con respecto al agente líder. Para esta formación específica, se ha establecido una distancia de seguridad de  $1m$  ( $d = 1m$ ).

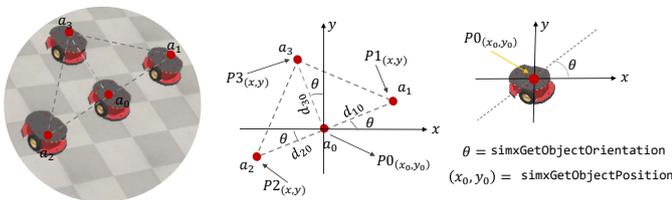


Figura 6: Formación multiagente.

Dentro de las consideraciones de red y las condiciones geométricas de la formación, el agente líder utiliza su posición absoluta  $(x_0, y_0, \theta)$  para calcular las posiciones objetivo de los agentes seguidores ( $P1_{x,y}$ ,  $P2_{x,y}$  y  $P3_{x,y}$ ). Estas posiciones objetivo se calculan utilizando las siguientes expresiones:

$$\begin{aligned} P1_x &= x_0 + d \cos(\theta) \\ P1_y &= y_0 + d \sin(\theta) \\ P2_x &= x_0 + \frac{d}{2} \cos(\theta + \frac{2\pi}{3}) \\ P2_y &= y_0 + \frac{d}{2} \sin(\theta + \frac{2\pi}{3}) \\ P3_x &= x_0 + \frac{d}{2} \cos(\theta - \frac{2\pi}{3}) \\ P3_y &= y_0 + \frac{d}{2} \sin(\theta - \frac{2\pi}{3}) \end{aligned} \quad (14)$$

Donde  $d$  representa la distancia de seguridad establecida para la formación,  $x_0$  y  $y_0$  son las coordenadas  $x$  y  $y$  del agente líder, respectivamente, y  $\theta$  es el ángulo de orientación del agente líder. Estas expresiones permiten al agente líder calcular las posiciones objetivo de los agentes seguidores con base en su propia posición y orientación, asegurando así la formación deseada.

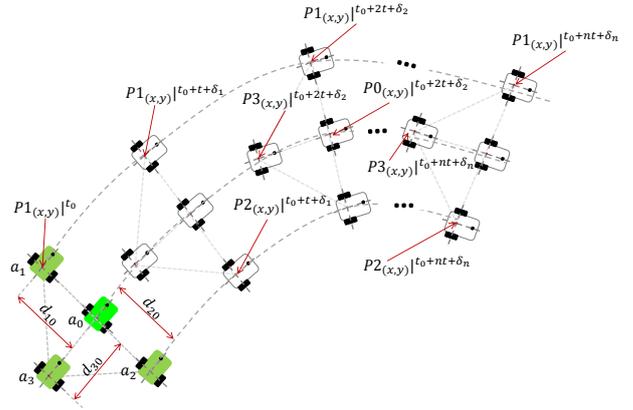


Figura 7: Evaluación de puntos de control.

Una vez que el agente líder ha evaluado las posiciones objetivo de los agentes seguidores según la ecuación (14), como se muestra en la Figura 7, procede a transmitir las a cada uno de los agentes. Esta transmisión se realiza en un periodo temporal de  $t = 10ms$ . La simulación se inicia en el tiempo relativo  $t_0$ . El retardo instantáneo en la transmisión de las posiciones objetivo se representa como  $nt + \delta_n$ , donde  $n$  es el número de agente (1, 2 o 3) y  $\delta_n$  es el retardo específico para cada agente. De esta manera, se acumula un tiempo de recorrido  $t_R$  para cada agente  $a_k$  de acuerdo a la siguiente expresión:

$$t_R^{(k)} = t_0 + k \cdot t + \delta_k \quad (15)$$

Donde  $k$  es el número de agente (0, 1, 2 o 3),  $t$  es el periodo de transmisión de las posiciones objetivo,  $\delta_k$  es el retardo específico para el agente  $a_k$  y  $t_0$  es el tiempo de inicio relativo de la simulación. Este retardo acumulado representa el tiempo que tarda la información en llegar a cada agente seguidor a partir de la transmisión del agente líder.

##### 4.5.1. Descripción del experimento

El escenario planteado para evaluar el impacto del Jitter en un control de formaciones consta de un agente líder ( $a_0$ ) y tres agentes seguidores ( $a_1 - 3$ ), como se muestra en la Figura 8. El agente ( $a_0$ ) es el único que conoce de antemano la trayectoria que va a seguir, en este caso, en forma de U. A medida que el agente ( $a_0$ ) avanza cada  $10ms$ , calcula las posiciones objetivo ( $P1_{x,y}$ ,  $P2_{x,y}$ ,  $P3_{x,y}$ ) de los agentes seguidores según la ecuación (14) y las transmite.

Los agentes ( $a_{1-3}$ ) toman estas posiciones recibidas en cada paquete y las utilizan como señales de control en el algoritmo PID descrito en la sección 2.2. Estas posiciones objetivo actúan como referencia para el control de cada agente, guiándolos hacia la formación deseada en cada instante de tiempo.

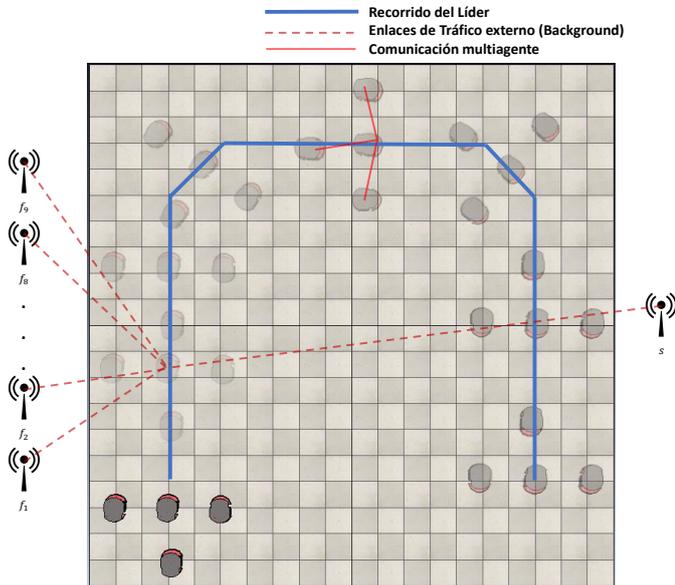


Figura 8: Escenario de evaluación.

Los parámetros de las fuentes de tráfico de datos externos se encuentran en la Tabla 1. En la primera evaluación, se realizó el experimento en condiciones ideales sin fuentes de tráfico externo ( $f_0$ ), lo que significa que la única comunicación existente fue entre el agente líder  $a_0$  y los agentes seguidores  $a_{1-3}$ . En este experimento se evaluaron las siguientes métricas: la latencia promedio, el Jitter, la variación de la distancia de seguridad  $d_{i,0}$  y el tiempo de recorrido  $t_R(a_k)$ .

En el siguiente experimento, se incorporó una fuente de tráfico externa llamada  $f_1$ , que utilizó al agente  $a_0$  como punto de acceso (AP) para enlazar el tráfico con el nodo  $n_s$  de destino. En este caso, el nodo  $n_s$  de destino actúa como punto central de recopilación y agregación de datos.

Se evaluaron las mismas métricas que en el experimento  $f_0$ . En los experimentos siguientes, se incrementó el número de fuentes de tráfico externo hasta llegar a  $f_9$ , manteniendo las mismas métricas de evaluación.

Dado que todo el tráfico externo (tráfico background) debe pasar a través del agente  $a_0$ , se pudo observar el impacto del Jitter en el control de formación a medida que se incrementaban las fuentes de tráfico externo.

## 5. Análisis de resultados

Las constantes del controlador PID  $k_p = 2,0$ ,  $k_i = 0,01$  y  $k_d = 0,001$  se tomaron del trabajo realizado por (Apriaskar et al., 2020), aunque se modificó el valor integral  $k_d$  para ajustar el algoritmo propuesto en la sección 2.2 e implementado en cada agente.

En esta sección, se analiza el efecto del Jitter en un sistema de control de formación multiagente y cómo este se ve afectado

por fuentes de tráfico externas que se incrementan gradualmente durante la experimentación.

En el primer caso, se muestra en la Figura 9 la latencia y el Jitter obtenidos en el escenario de simulación de la Figura 8, sin ninguna fuente de tráfico externa ( $f_0$ ). Este caso se considera ideal, ya que el único tráfico de red generado es la comunicación entre el agente líder  $a_0$  y los agentes seguidores  $a_{1-3}$ .

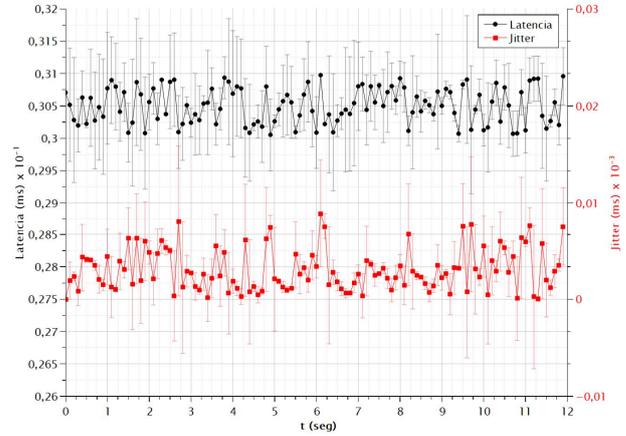


Figura 9: Latencia y Jitter sin fuentes externas de tráfico ( $f_0$ ).

La Figura 9 muestra que la latencia promedio es de 0,305 ms y el Jitter es de 0,00301 ms, con una variación de  $7,4892 \times 10^{-6}$  y  $4,6457 \times 10^{-6}$ , respectivamente. En la Figura 10, se puede observar el recorrido realizado por los agentes sin la influencia de tráfico de datos externo y con baja variabilidad (los agentes no muestran signos de perturbación en sus movimientos).

En la Figura 11, se muestra la distancia de seguridad entre los agentes seguidores y el agente líder, con valores de desviación estándar inferiores a  $\pm 0,12$  m durante el recorrido. La desviación estándar de la distancia  $d_{10}$  y  $d_{30}$  es de  $\pm 0,0158$  m y  $\pm 0,01174$  m, respectivamente, siendo mayor y menor debido a sus respectivos recorridos.

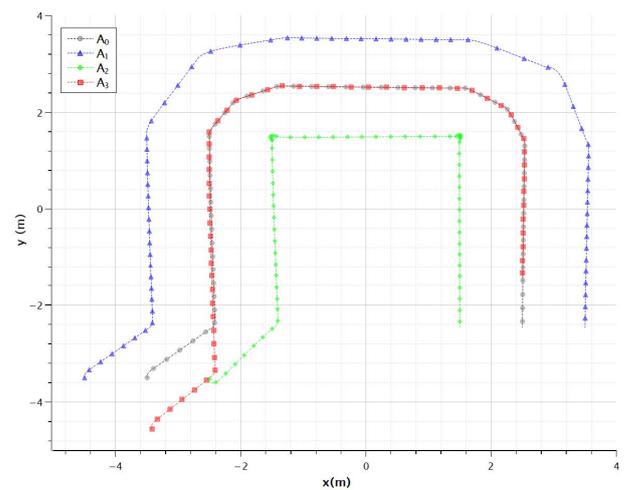


Figura 10: Recorrido de los Agentes sin fuentes externas de tráfico ( $f_0$ ).

En este caso, sin fuentes de tráfico externo ( $f_0$ ), los agentes completaron por completo el recorrido en un tiempo de  $t_R = 12,34$  segundos.

La Figura 12 muestra el comportamiento de la latencia para las fuentes de tráfico  $f_0$ ,  $f_1$ ,  $f_5$  y  $f_9$ . La fuente de tráfico  $f_0$  representa el tráfico multiagente sin fuentes de tráfico externas. La fuente de tráfico  $f_1$  corresponde a un único agente generando tráfico externo, mientras que  $f_5$  y  $f_9$  representan cinco y nueve fuentes de tráfico externas o tráfico de fondo, respectivamente.

En comparación con  $f_0$ , la fuente de tráfico  $f_1$  muestra un incremento de la latencia del 1,8%. La visualización del tráfico ( $f_5$ ) indica que, a partir de esta fuente de tráfico, solo se completaron la formación en 7 de cada 10 simulaciones realizadas. Con  $f_9$ , se observaron dos completaciones de la formación en las diez simulaciones realizadas. Se encontró que las fuentes de tráfico  $f_1$  y  $f_4$  permitieron completar el recorrido de la formación en todas las simulaciones realizadas.

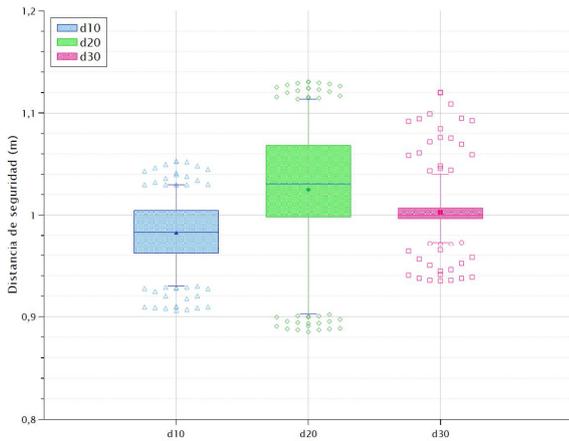


Figura 11: Distancia de seguridad sin fuentes externas de tráfico ( $f_0$ ).

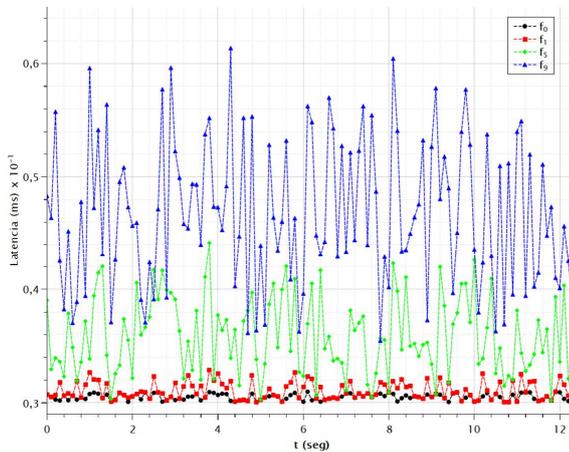


Figura 12: Latencia para fuentes de tráfico  $f_0$ ,  $f_1$ ,  $f_5$  y  $f_9$ .

De manera similar, la Figura 13 muestra el comportamiento de la latencia en una de las diez simulaciones realizadas. Se identificó un umbral de 0,014 ms, en el cual cada pico que superaba este umbral generaba una perturbación en el recorrido de la formación. Esto hacía que el controlador PID de cada agente realizara más iteraciones para alcanzar el punto objetivo, lo que a su vez resultaba en un aumento en la desviación estándar de la distancia de seguridad entre los agentes.

En el caso de un agente que se encontraba en un ciclo iterativo del controlador PID para alcanzar el punto objetivo, si

el Jitter era inferior a 0,00511 ms, no se observaba una desviación estándar del punto de consigna ( $w$ ) superior al  $\pm 0,349\%$  rad. Por lo tanto, la posición no era un indicador significativo de perturbación en el recorrido del agente. Este tiempo se tomó como referencia ( $Ref_2$ ) en la Figura 15.

La generalización de los resultados de simulación se puede observar en las Figuras 14 a 16, que representan la latencia, el Jitter y la distancia de seguridad de la formación multiagente, respectivamente. En la Figura 14, se muestra el comportamiento de la latencia en una red multiagente ( $f_0$ ) y la influencia directa y proporcional de una red externa de tráfico de fondo en función del número de fuentes de tráfico ( $f_{1-9}$ ).

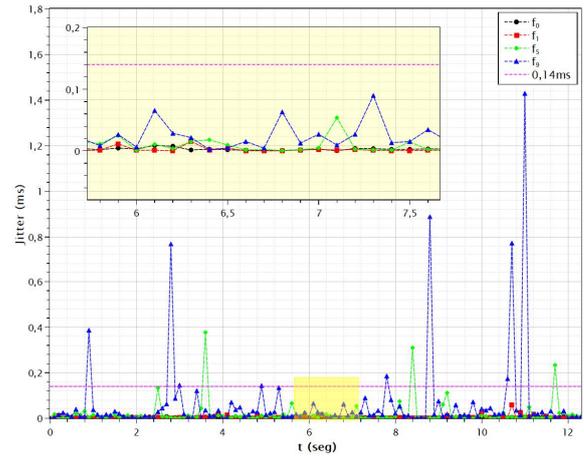


Figura 13: Jitter para fuentes de tráfico  $f_0$ ,  $f_1$ ,  $f_5$  y  $f_9$ .

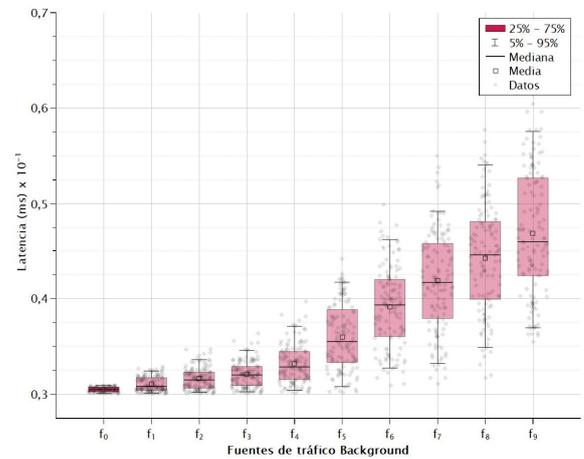


Figura 14: Comportamiento de la Latencia para 10 casos de experimentación.

En la Figura 15, se identificaron dos umbrales,  $Ref_1$  y  $Ref_2$ . Si el Jitter instantáneo es superior a  $Ref_2$ , se genera un movimiento oscilatorio en la posición de los agentes, cuya desviación estándar está relacionada con el error del PID. Por otro lado, si el Jitter instantáneo supera el umbral de referencia  $Ref_1$ , el ciclo iterativo del PID es más largo. Si llegan uno o varios paquetes de datos que contienen la señal de consigna del nuevo punto objetivo, para los agentes seguidores, estos paquetes no serán atendidos y quedarán en un bucle de posiciones generado por el protocolo TCP, ya que este protocolo no admite la pérdida de paquetes.

Esto hace que el agente seguidor siga posiciones objetivo desfasadas con respecto a su posición actual, debido a la retransmisión de paquetes.

Este proceso de retransmisión de paquetes aumenta el tiempo de recorrido acumulado  $t_R$  entre los agentes. En las simulaciones realizadas, se encontró que los agentes finalizaron el recorrido en un intervalo de  $12,34 < t_R < 12,52$  segundos para fuentes de tráfico  $f_0$  hasta  $f_4$ . A partir de la fuente de tráfico  $f_5$  en adelante, se observaron recorridos incompletos de los agentes, lo que aumentó el promedio de  $t_R$  de finalización.

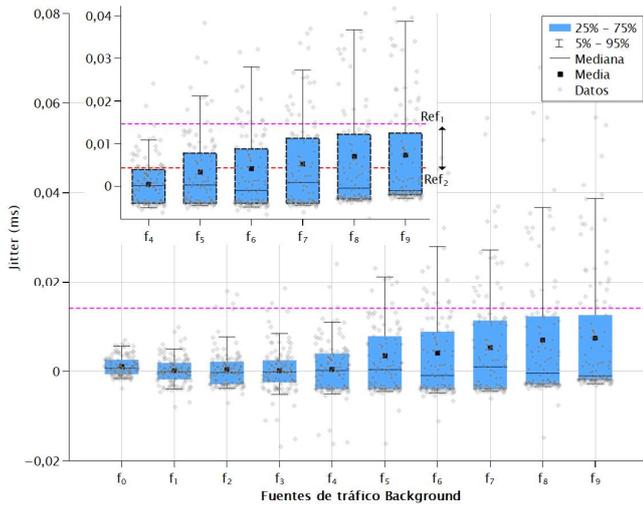


Figura 15: Comportamiento del Jitter para 10 casos de experimentación.

En la Figura 16, se muestra el comportamiento de la distancia de seguridad para las fuentes de tráfico  $f_1$  hasta  $f_9$ . Esta figura abarca tres posibles escenarios: finalización exitosa de un recorrido con control de formación, evento de choque (como se muestra en la Figura 17) y evento de bucle iterativo del PID (mostrado en la Figura 18), donde los agentes no logran recuperar su posición dentro de la formación.

Los valores cercanos a cero en el intervalo de  $f_5$  hasta  $f_9$ , mostrados en la Figura 16, representan eventos de choque en las simulaciones realizadas.

En este intervalo, se observan desviaciones estándar de la distancia de seguridad superiores a 1.5 metros, lo cual indica eventos de bucle iterativo del PID con presencia de retransmisión de paquetes que resultan en desfases en el control de formación.

Es cierto que los resultados obtenidos muestran que la latencia está influenciada por el número de fuentes de tráfico. A medida que aumenta el número de fuentes de tráfico, la latencia tiende a incrementarse. Sin embargo, es importante tener en cuenta que los resultados obtenidos son producto de simulaciones y que existen diversos factores que pueden influir en la latencia.

Uno de los factores identificados que puede afectar la latencia es el bucle iterativo del controlador proporcional-integral-derivativo (PID). Durante el ciclo iterativo del PID, el agente o controlador se enfoca en el cálculo y ajuste de las acciones de control para mantener la estabilidad y el rendimiento óptimo del sistema. Sin embargo, esta atención enfocada puede llevar a que el agente no preste la debida atención al canal de comunicación utilizado para la transmisión de datos.

Como resultado, pueden ocurrir una serie de problemas. Por un lado, los retrasos en el procesamiento de los paquetes de datos pueden ocurrir debido a que el agente necesita tiempo para completar sus cálculos antes de atender los datos entrantes. Esto puede resultar en un aumento de la latencia, ya que los paquetes deben esperar en cola antes de ser procesados y transmitidos.

Adicionalmente, durante el ciclo iterativo del PID, existe el riesgo de que el agente no esté disponible para recibir nuevos datos o responder a ellos de manera oportuna. Esto puede llevar a la pérdida de paquetes, ya que los datos entrantes pueden llegar mientras el agente está ocupado realizando cálculos o ajustes.

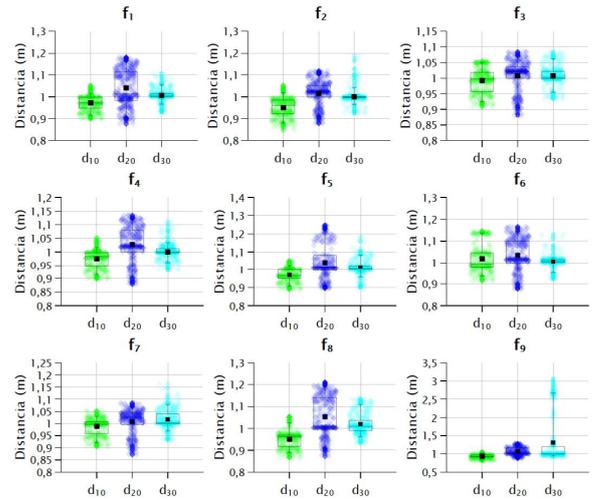


Figura 16: Comportamiento de la distancia de seguridad para 10 casos de experimentación.

En tales casos, puede ser necesario volver a transmitir los paquetes perdidos, lo que agrega una sobrecarga adicional y potencialmente aumenta aún más la latencia. La retransmisión de paquetes también puede contribuir a incrementar la latencia. Si se producen colisiones o interrupciones en la transmisión de datos, es posible que los paquetes deban ser retransmitidos, lo que genera un mayor tiempo de respuesta y, por lo tanto, una mayor latencia.

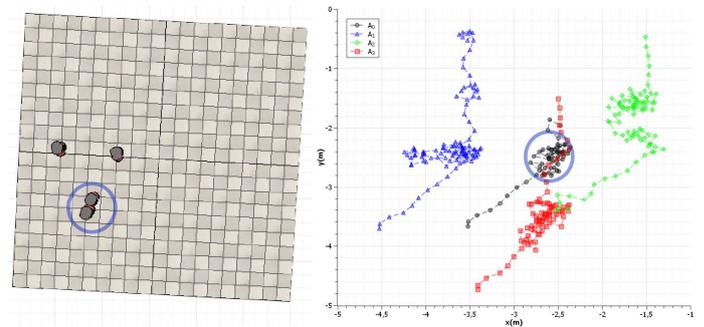


Figura 17: Rompimiento de la formación por choque multiagente.

Estos factores deben ser considerados en el diseño e implementación de sistemas de control de formación multiagente, ya que pueden tener un impacto significativo en el desempeño y la estabilidad del sistema.

Cuando el buffer del agente líder  $a_0$  se satura debido a las fuentes de tráfico externas, se generan retrasos en el procesamiento y enrutamiento de los paquetes. Esto puede resultar en la retransmisión de paquetes, lo que implica que los paquetes se “pierden” en el sentido de que no se entregan en el instante de tiempo esperado. Sin embargo, el protocolo TCP está diseñado para garantizar la integridad de la información y proporciona mecanismos para la retransmisión de paquetes perdidos.

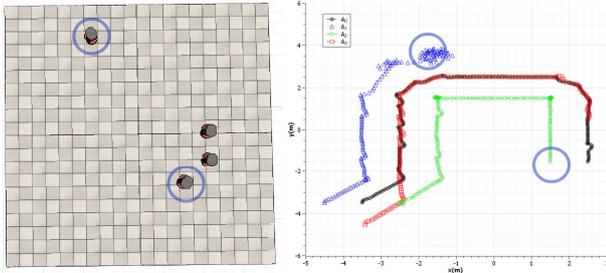


Figura 18: Rompimiento de la formación por bucle PID.

Todo el tráfico generado, tanto el tráfico multiagente para el control de formación como el tráfico de fondo, es centralizado, procesado y enrutado por el agente líder  $a_0$ . En este agente se presentan dos procesos dentro del protocolo TCP: el manejo de paquetes fuera de orden y la detección de paquetes perdidos.

En el primer proceso, TCP cuenta con un mecanismo que le permite detectar paquetes que, debido a las retransmisiones generadas, quedan fuera de orden de llegada. Esto se logra mediante el uso de números de secuencia y confirmación.

Cuando los nodos de destino, como  $n_s$  y  $a_{1-3}$ , observan un número de secuencia más alto del que ha sido confirmado por  $a_0$ , pueden detectar que falta al menos un paquete. En respuesta a esta detección, los nodos de destino envían un paquete con un número de confirmación igual al número de secuencia que esperaban. De esta manera, TCP, a través de DCF, toma el paquete faltante, lo coloca al final de la cola y lo transmite después de cierto tiempo.

El entorno de simulación (ver Figura 5) registra el proceso descrito anteriormente en el archivo de trazas “.tr”. Esto permite visualizar y detectar el proceso de manejo de paquetes fuera de orden y retransmisión en NS2, que es un simulador de eventos discretos. Sin embargo, dado que este proceso es aleatorio, no se pueden obtener simulaciones idénticas de forma instantánea. No obstante, este proceso es transparente para el usuario y se puede analizar a través de las trazas generadas en la simulación.

La falta de un paquete en el tráfico multiagente ( $f_0$ ) tiene implicaciones en el control de la formación de los agentes seguidores. Cada paquete contiene la posición objetivo o coordenada a la que los agentes deben dirigirse. Cuando se produce un retraso en la retransmisión de paquetes, esto se traduce en efectos negativos, como:

1. Desfase de la posición: Al recibir una posición objetivo atrasada, un agente seguidor puede encontrarse desalineado con respecto a la formación. Esto puede requerir que el agente gire y retroceda, incluso hasta  $180^\circ$ , lo cual implica complicaciones cinemáticas y dinámicas no holónicas en una plataforma robótica diferencial.

2. Incremento de ciclos en el bucle PID: El retraso en la recepción de la posición objetivo puede generar oscilaciones en el recorrido o trayectoria de los agentes. Esto puede deberse a que el controlador PID tiene que realizar más iteraciones para ajustar el comportamiento de los agentes a la nueva posición objetivo.

Estas implicaciones pueden afectar significativamente el control de la formación, ya que la falta de sincronización entre los agentes y la presencia de movimientos oscilantes pueden comprometer la estabilidad y precisión del sistema.

Como se mencionó en la sección 4.3.2, el Jitter se refiere a la diferencia temporal entre paquetes consecutivos. Cuando se produce una retransmisión de paquete y la diferencia temporal con respecto al paquete anterior supera el umbral  $Ref_1$ , los paquetes que llegan con cierto retraso corresponden a posiciones ligeramente atrasadas. Esto puede desestabilizar la formación de los agentes.

La Figura 13 muestra el comportamiento instantáneo del Jitter para las fuentes de tráfico  $f_5$  y  $f_9$ , las cuales superan el umbral  $Ref_1$ . En estos casos, alguno de los agentes no logra completar la simulación y se produce una ruptura en la formación, como se ilustra en las Figuras 17 y 18.

Esto demuestra que el Jitter, cuando supera ciertos umbrales críticos, puede tener un impacto negativo en el control de la formación de los agentes, generando inestabilidad y eventos disruptivos en el recorrido planificado. Es fundamental minimizar y controlar el Jitter para garantizar un desempeño óptimo del sistema de formación multiagente.

Los umbrales de referencia  $Ref_1$  y  $Ref_2$  fueron establecidos de manera heurística a través de la observación del comportamiento de movimiento de los agentes y los datos obtenidos en el archivo de trazas durante las simulaciones.

Se observó que cuando el Jitter instantáneo de un agente se acerca al umbral de referencia  $Ref_2$ , el agente comienza a oscilar ligeramente. Si el Jitter instantáneo supera  $Ref_2$  pero se mantiene por debajo de  $Ref_1$ , las oscilaciones se vuelven más notorias, pero el agente no retrocede en relación a su posición.

Estos comportamientos en el movimiento de los agentes son indicativos de la influencia del Jitter en el control de la formación. El Jitter dentro de ciertos rangos puede generar oscilaciones y perturbaciones en el recorrido planificado de los agentes, lo cual puede afectar su desempeño y estabilidad en la formación. Es importante tener en cuenta estos umbrales al analizar y optimizar el sistema de formación multiagente.

Abordando el segundo proceso que emplea TCP en DCF para la detección de paquetes perdidos a través del mecanismo de tiempo de espera, este consiste en que el agente líder  $a_0$ , después de enviar un paquete, inicia un temporizador y coloca el paquete en el buffer o cola de retransmisión. En caso de que el temporizador termine y el agente líder  $a_0$ , no reciba un paquete ACK de los nodos seguidores  $a_{1-3}$  y  $n_s$ , entonces reenvía el paquete que tiene en la cola.

La retransmisión de paquetes en TCP puede resultar en que los nodos de destino  $a_{1-3}$  y  $n_s$  reciban paquetes duplicados si el paquete original no se consideraba perdido y los nodos de destino no lo habían confirmado todavía. Sin embargo, los nodos de destino descartarán los paquetes duplicados para evitar procesarlos nuevamente.

Este mecanismo de tiempo de espera y retransmisión de paquetes en TCP ayuda a garantizar la entrega confiable de los paquetes en entornos de red con pérdida de paquetes. Aunque puede haber paquetes duplicados debido a retransmisiones, TCP se encarga de manejarlos correctamente para mantener la integridad y confiabilidad de la comunicación.

## 6. Conclusiones

En el análisis realizado se han contemplado los dos mecanismos de retransmisión de paquetes que presenta TCP. Además, DCF trata todos los paquetes de la red con la misma prioridad y, sumado a esto, el incremento en el retardo por las fuentes de tráfico externas, se observa un aumento en la latencia a nivel general. Esto afecta al Jitter y, a su vez, impacta el control de formación multiagente, ya que todo gira en torno a la información que llevan los paquetes que obedecen a las posiciones objetivo de los agentes seguidores. Por lo tanto, cualquier variación temporal que sufra un paquete condiciona la posición de los agentes en la formación multiagente.

Las simulaciones realizadas permitieron identificar heurísticamente dos umbrales de referencia del Jitter:  $Ref_1 = 0,014ms$  y  $Ref_2 = 0,00511ms$ . Se encontró un comportamiento estable para los valores del Jitter instantáneo que estén por debajo del umbral de referencia  $Ref_2$ , un comportamiento oscilatorio en la posición de los agentes para valores de Jitter en el rango de  $Ref_2 < Jitter_{instantaneo} < Ref_1$  y un comportamiento inestable para valores de  $Jitter_{instantaneo} > Ref_1$ . Cabe destacar que estos valores se encontraron con los parámetros de simulación registrados en la Tabla 1, las consideraciones de red y multiagente de la sección 4.2, y los mecanismos de tratamiento de retransmisión de paquetes que presenta TCP en DCF.

Para futuras investigaciones, es necesario explorar otros estándares, como IEEE 802.11g/n/ac. Para ello, se requiere modificar el entorno propuesto y utilizar otro software de simulación de eventos discretos. Se necesitan análisis adicionales del comportamiento iterativo del bucle del controlador PID, la relación entre la señal de control y la desviación estándar de la señal de consigna, así como las consideraciones de retransmisión de paquetes en TCP.

## Agradecimientos

Este trabajo ha sido realizado gracias al apoyo de la Facultad de Ingeniería de la Universidad Cooperativa de Colombia, del departamento de Informática y Ciencias de la Computación de la Universidad de Oviedo y del departamento de Ingeniería Electrónica y Mecatrónica de la Fundación Universitaria Los Libertadores.

## Referencias

Apriaskar, E., Fahmizal, F., Cahyani, I., Mayub, A., May 2020. Autonomous mobile robot based on BehaviourBased robotic using v-REP simulator-pioneer 3-DX robot. *Jurnal Rekayasa Elektrika* 16 (1).  
URL: <https://doi.org/10.17529/jre.v16i1.15081>  
DOI: 10.17529/jre.v16i1.15081

Bhatia, A., Kumar, A., Jain, A., Kumar, A., Verma, C., Illes, Z., Aschilean, I., Raboaca, M. S., Nov. 2022. Networked control system with MANET communication and AODV routing. *Heliyon* 8 (11), e11678.  
URL: <https://doi.org/10.1016/j.heliyon.2022.e11678>  
DOI: 10.1016/j.heliyon.2022.e11678

Chen, L., Li, C., Guo, Y., Ma, G., Li, Y., Xiao, B., Sep. 2022. Formation-containment control of multi-agent systems with communication delays. *ISA Transactions* 128, 32–43.  
URL: <https://doi.org/10.1016/j.isatra.2021.09.012>  
DOI: 10.1016/j.isatra.2021.09.012

Cheng, Y., Yang, D., Zhou, H., Wang, H., Jul. 2019. Adopting IEEE 802.11 MAC for industrial delay-sensitive wireless control and monitoring applications: A survey. *Computer Networks* 157, 41–67.  
URL: <https://doi.org/10.1016/j.comnet.2019.04.002>  
DOI: 10.1016/j.comnet.2019.04.002

Choi, H.-H., Lee, J.-R., 2019. Local flooding-based on-demand routing protocol for mobile ad hoc networks. *IEEE Access* 7, 85937–85948.  
URL: <https://doi.org/10.1109/access.2019.2923837>  
DOI: 10.1109/access.2019.2923837

Hu, D., Yang, S., Gong, M., Feng, Z., Zhu, X., Dec. 2022a. A cyber-physical routing protocol exploiting trajectory dynamics for mission-oriented flying ad hoc networks. *Engineering* 19, 217–227.  
URL: <https://doi.org/10.1016/j.eng.2021.10.022>  
DOI: 10.1016/j.eng.2021.10.022

Hu, J., Lennox, B., Arvin, F., Jun. 2022b. Robust formation control for networked robotic systems using negative imaginary dynamics. *Automatica* 140, 110235.  
URL: <https://doi.org/10.1016/j.automatica.2022.110235>  
DOI: 10.1016/j.automatica.2022.110235

Jiménez, A. C., Anzola, J. P., García-Díaz, V., González Crespo, R., Zhao, L., 07 2020. Pysdlrep: A domain-specific language for robotic simulation in v-rep. *PLOS ONE* 15 (7), 1–24.  
URL: <https://doi.org/10.1371/journal.pone.0235271>  
DOI: 10.1371/journal.pone.0235271

kumar Gupta, A., Venkatesh, T., Jun. 2022. Design and analysis of IEEE 802.11 based full duplex WLAN MAC protocol. *Computer Networks* 210, 108933.  
URL: <https://doi.org/10.1016/j.comnet.2022.108933>  
DOI: 10.1016/j.comnet.2022.108933

Lin, J., Miao, Z., Zhong, H., Peng, W., Wang, Y., Fierro, R., Jul. 2021. Adaptive image-based leader-follower formation control of mobile robots with visibility constraints. *IEEE Transactions on Industrial Electronics* 68 (7), 6010–6019.  
URL: <https://doi.org/10.1109/tie.2020.2994861>  
DOI: 10.1109/tie.2020.2994861

Manzoor, S., Yin, Y., Gao, Y., Hei, X., Cheng, W., 2020. A systematic study of IEEE 802.11 DCF network optimization from theory to testbed. *IEEE Access* 8, 154114–154132.  
URL: <https://doi.org/10.1109/access.2020.3018088>  
DOI: 10.1109/access.2020.3018088

Peng, J., 2018. Network state conservation in dynamic spectrum access: the IEEE 802.11 DCF case. *Procedia Computer Science* 134, 275–282.  
URL: <https://doi.org/10.1016/j.procs.2018.07.171>  
DOI: 10.1016/j.procs.2018.07.171

Sánchez-Sánchez, P., Arteaga-Pérez, M. A., Sep. 2020. Control de posición y fuerza con estimación de masa para sistemas cooperativos. *Revista Iberoamericana de Automática e Informática industrial* 17 (4), 368.  
URL: <https://doi.org/10.4995/riai.2020.12432>  
DOI: 10.4995/riai.2020.12432

Tardioli, D., Parasuraman, R., Ögren, P., Jan. 2019. Pound: A multi-master ROS node for reducing delay and jitter in wireless multi-robot networks. *Robotics and Autonomous Systems* 111, 73–87.  
URL: <https://doi.org/10.1016/j.robot.2018.10.009>  
DOI: 10.1016/j.robot.2018.10.009

William, P., Pawar, A., Jawale, M., Badholia, A., Verma, V., Dec. 2022. Energy efficient framework to implement next generation network protocol using ATM technology. *Measurement: Sensors* 24, 100477.  
URL: <https://doi.org/10.1016/j.measen.2022.100477>  
DOI: 10.1016/j.measen.2022.100477

Zheng, G., Zhi-Jun, Y., Min, H., Wen-Hua, Q., Jan. 2018. Energy-efficient analysis of an IEEE 802.11 PCF MAC protocol based on WLAN. *Journal of Ambient Intelligence and Humanized Computing* 10 (5), 1727–1737.  
URL: <https://doi.org/10.1007/s12652-018-0684-8>  
DOI: 10.1007/s12652-018-0684-8

Zhou, X., Li, D., Zhang, L., Duan, Q., Aug. 2021. Application of an adaptive PID controller enhanced by a differential evolution algorithm for precise control of dissolved oxygen in recirculating aquaculture systems. *Biosystems Engineering* 208, 186–198.  
URL: <https://doi.org/10.1016/j.biosystemseng.2021.05.019>  
DOI: 10.1016/j.biosystemseng.2021.05.019