DESDE 2013
https://repository.uaeh.edu.mx/revistas/index.php/icbi/issue/archive
Pädi Boletín Científico de Ciencias Básicas e Ingenierías del ICBI

ISSN: 2007-6363

Publicación Semestral Pädi Vol. 14 No. Especial 3 (2026) 264-274

Sintonización de un controlador PID robusto para un robot paralelo tipo delta usando el algoritmo de optimización del zorro Rüppell.
Tuning a robust PID controller for a parallel delta robot using Rüppell's fox optimizer.

T. M. Morales Mendoza [a], N. J. Martínez López [a,*], O. Aguilar-Mejía [a], H. Minor-Popocatl [b], C. Enríquez-Ramírez [c], M. A. Andrade-Soto [d]

[a] *Área Académica de Ingeniería y Arquitectura, Universidad Autónoma del Estado de Hidalgo, 42184, Pachuca, Hidalgo, México.*
[b] *Faculty of Industrial Systems; UPAEP University - CONCYTEP, 72410 Puebla, PUE, México*
[c] *Universidad Politécnica de Tulancingo, Tulancingo, Hidalgo, 43628, México.*
[d] *Facultad de Ingeniería Mecánica y Eléctrica, Universidad Autónoma de Nuevo León, San Nicolás de los Garza, Nuevo León 66451, México*

Resumen

En este trabajo se utiliza un enfoque integrado para regular la trayectoria y posición de la plataforma móvil de un robot paralelo tipo delta (RPTD) de tres grados de libertad. El sistema de ecuaciones que describe la dinámica del sistema se obtiene a partir de la formulación lagrangiana, que considera los efectos de la inercia, el efecto Coriolis y la gravedad para generar el modelo matemático que describe la trayectoria de la plataforma móvil del RPTD. Se diseña un esquema de control basada en un controlador PID robusto, para cada articulación del RPTD, con parámetros óptimos calculados mediante algoritmos de optimización metaheurísticos. En el proceso para calcular las ganancias del PID robusto, la función objetivo está compuesta con restricciones no lineales para mejorar la robustes dinámica y la respuesta transitoria del esquema de control. El algoritmo de control PID robusto y el modelo matemático del RPTD se implementó en Matlab/Simulink para analizar su desempeño dinámico ante diferentes condiciones operativas. Los resultados muestran que el controlador PID robusto optimizado con la función objetivo tiene desempeño robusto en tareas del seguimiento a trayectorias complejas y cambios bruscos de dirección de la plataforma móvil cuando se aplica a manipuladores robóticos paralelos.

Palabras Clave: robots paralelos, algoritmo de optimización, control PID, seguimiento de trayectorias.

Abstract

This work uses an integrated approach to regulate the trajectory and position of the mobile platform of a three-degree-of-freedom delta-type parallel robot (DPT). The system of equations describing the system's dynamics is derived from the Lagrangian formulation, which considers the effects of inertia, the Coriolis effect, and gravity to generate the mathematical model describing the trajectory of the DPT's mobile platform. A control scheme based on a robust PID controller is designed for each joint of the DPT, with optimal parameters calculated using metaheuristic optimization algorithms. In the process of calculating the robust PID gains, the objective function is composed with nonlinear constraints to improve the dynamic robustness and transient response of the control scheme. The robust PID control algorithm and the mathematical model of the DPT were implemented in Matlab/Simulink to analyze their dynamic performance under different operating conditions. The results show that the optimized robust PID controller with the objective function has robust performance in tasks of tracking complex trajectories and abrupt changes in direction of the mobile platform when applied to parallel robotic manipulators.

Keywords: parallel robots, optimization algorithm, PID control, trajectory tracking.

1. Introducción

Los manipuladores basados en robots son estructuras mecánicas empleadas en la industria para realizar tareas

*Autor para la correspondencia: ma524031@uaeh.edu.mx
Correo electrónico: mo458949@uaeh.edu.mx (Tayli Merlin Morales Mendoza), ma524031@uaeh.edu.mx (Nancy Josselyn Martínez López), omar_aguilar@uaeh.edu.mx (Omar Aguilar Mejía), hertwin.minor@upaep.mx (Hertwin Minor Popocatl), carlos.enriquez@upt.edu.mx (Carlos Enríquez Ramírez), manuel.andradest@uanl.edu.mx (Manuel Antonio Andrade-Soto).

Historial del manuscrito: recibido el 08/06/2026, última versión-revisada recibida el 03/08/2026, aceptado el 29/09/2026, publicado el 06/11/2026. **DOI:** https://doi.org/10.29057/icbi.v14iEspecial3.18098

repetitivas en los procesos de fabricación. Estos sistemas mecatrónicos garantizan un alto volumen de producción con altos estándares de calidad (Asif et al., 2024). En función de su estructura mecánica, los robots industriales se clasifican en manipuladores seriales y paralelos. Un robot serial tiene sus eslabones unidos secuencialmente. En un robot paralelo, sus eslabones se unen entre dos plataformas, una móvil y otra fija (Elassal et al., 2024). En los últimos 30 años se ha observado un aumento exponencial en el uso cada vez mayor de los robots paralelos de alta velocidad para tareas de pick-and-place. Las principales industrias que usan el robot paralelo tipo Delta (RPTD) son en aplicaciones en ensamble de dispositivos electrónicos, industria alimenticia, farmacéutica, y empaquetado y embalado de productos ligeros (Liu et al., 2019), (Rahimi et al., 2022). Pero diseñar esquemas de control eficientes para el RPTD presentan diversos desafíos técnicos debido a su naturaleza multivariable, la presencia de elementos no lineales y la alta interacción entre las restricciones cinemáticas de los eslabones del robot (Mahdi et al., 2025).

En la literatura se han reportado diversos esquemas de control para regular de la posición, aceleración, velocidad y el seguimiento de la trayectoria de la plataforma móvil del (RPTD). En (Mahdi et al., 2025), (Kong et al., 2026) utilizan estrategias tradicionales de regulación, tal como el controlador Proporcional Integral Derivativo (PID) independiente en cada eje, un control PID con retroalimentación de la aceleración, el controlador PID con compensación de gravedad y control PID no lineal; han sido utilizados para regular el seguimiento de la trayectoria del efector final del RPTD. Se debe considerar que los controladores PID no utilizan, o utilizan parcialmente, el modelo dinámico del RPTD, omitiendo la interacción dinámica entre los eslabones del robot. Obteniendo como resultado, un bajo desempeño dinámico de estos controladores cuando la condición operativa requiere altas velocidades en las tareas del RPTD. Para disminuir esta problemática en (Taghirad, 2025) utiliza la estrategia de Control con Torque Calculado (CTC) que emplea todo el modelo matemático del RPTD. Sin embargo, la estrategia CTC proporciona un buen desempeño únicamente cuando el modelo matemático es preciso y detallado, pero su desempeño baja considerablemente cuando se presentan incertidumbres y perturbaciones externas no modeladas (Tajdari & Toulkani, 2021), (Ohno & Takeda, 2020).

En (Serrano & Cardona, 2024), (Zhao et al., 2020) utilizan la estrategia de control robusto H∞, donde muestran que se obtienen una excelente respuesta transitoria y buen desempeño dinámico del sistema ante la presencia de incertidumbres paramétricas y perturbaciones externas. Pero realizar el diseño e implementación del controlador H∞ es demasiado complejo. Por lo tanto, se están realizando esquemas de control utilizando reguladores tradicionales PID que presentan una estructura simple y fácil de implementar. Estas técnicas se enfocan principalmente en cómo ajustar los parámetros del controlador PID para que el sistema en lazo cerrado garantice robustez y atenuación de perturbaciones. En (Kong et al., 2026b), se realiza el ajuste de los parámetros de un controlador PID para un RPTD resolviendo el problema de optimización como un sistema no lineal, mediante el algoritmo de evolución diferencial (DE). Sin embargo, la técnica de optimización solo considera restricciones de estabilidad y del rendimiento de seguimiento de la trayectoria, para ajustar los parámetros del controlador.

En (Yigit & Celik, 2023) se presenta un esquema de control PID de orden fraccionario, donde muestra que una de sus características es mejorar la estabilidad y desempeño dinámico de todo el sistema, incluso cuando aparecen incertidumbres paramétricas y perturbaciones externas. Un método típico para ajustar los parámetros del controlador PID de orden fraccionario es utilizando algoritmos evolutivos basados en enjambre de partículas (Haji & Monje, 2017). Este método de sintonización tiene la ventaja de ajustar el las ganancias del regulador de tal forma que el sistema de control en lazo cerrado cumpla varias restricciones relacionadas con incertidumbres paramétricas de la planta, perturbaciones externas y ruido de alta frecuencia. Las restricciones que proponen hacen que el ajuste de los parámetros del controlador se reduzca a un problema de optimización no lineal con restricciones no lineales.

Sin embargo, en algunos casos para resolver el problema de optimización no lineal, presentan algunas desventajas, como la convergencia hacia mínimos locales, alta sensibilidad a seleccionar las condiciones iniciales y no convergen cuando existen discontinuidades. Por lo tanto, la solución que entrega el algoritmo de búsqueda puede no ser una solución global óptima, lo que se deriva en la necesidad de proponer alternativas que superen dichos problemas, motivando el estudio presentado en este trabajo de investigación. El proceso de optimización que propone Haji and Monje (2017), se utiliza principalmente para buscar los parámetros óptimos de diferentes esquemas de control en procesos industriales (Choubey & Ohri, 2021), (Tepljakov et al., 2015), (Haji & Monje, 2017a).

Mahdi et al. (2026) realiza la sintonización de un esquema de control basado en controladores PID para regular la trayectoria de un RPTD utilizando tres algoritmos de optimización inspirados en la naturaleza el algoritmo de optimización de las ballenas (AOB), el algoritmo de optimización de la polinización de las flores (AOPF) y el algoritmo de optimización por enjambre (PSO). Saleem et al. (2025) utiliza el algoritmo de optimización armonía para sintonizar un algoritmo Tilt-ID y un controlador PID para regular la trayectoria de un RPTD y comparar su desempeño. Pak et al. (2022) propone un enfoque para buscar los parámetros óptimos de un controlador PID robusto mediante una estrategia de control PID para calcular el par que regula el seguimiento de la trayectoria de un RPTD, utilizando un algoritmo de optimización híbrido de optimización por enjambre de partículas (PSO) y evolución diferencial (DE). Pero en los tres trabajos anteriores no utilizan restricciones o elementos técnicos durante la búsqueda de los parámetros para mejorar la robustez del esquema de control, simplemente realizan una búsqueda en función a métricas de desempeño.

Por todo lo anterior, en este trabajo se utiliza el algoritmo de optimización del zorro Rüppell (AOZR) para buscar las mejores ganancias de los controladores PID del RPTD, debido a que en la literatura no se encontró el uso del AOZR para esta tarea. Además, el AOZR tiene la capacidad de resolver el problema de optimización no lineal para regular la trayectoria del seguimiento de la posición, aceleración y velocidad de la plataforma móvil del RPTD ante diferentes condiciones operativas.

La principal contribución de este artículo es diseñar un controlador PID robusto que cumpla requisitos de robustez, atenuación de ruido de alta frecuencia y rechazo de

perturbaciones internas y externas, utilizando el AOZR para aplicarlo al control de seguimiento de trayectoria de un RPTD en diferentes tareas. En este trabajo, el problema de diseño del controlador PID robusto para el RPTD se realiza en las siguientes etapas: a) establecer las especificaciones de diseño, b) el problema de diseño del esquema de control se plantea como un problema de optimización con varias restricciones; c) obtener la solución global del problema de optimización utilizando AOZR; y d) para comparar el desempeño del controlador propuesto se emplea un controlador PID-Adaptativo utilizando el método que se describe en la referencia (Escorcia-Hernández et al., 2020). Para comparar el desempeño del AOZR se emplea el Algoritmo de Optimización del Puma (AOP) (Abdollahzadeh et al., 2024) y el Algoritmo de Optimización del Hipopótamo (AOH) (Amiri et al., 2024). Los tres algoritmos fueron seleccionados debido a su equilibrio dinámico en relación a la fase de exploración y de explotación mediante mecanismos de inteligencia colectiva, con la incorporación de mecanismos estocásticos para evitar la convergencia prematura y evitar óptimos locales. De la misma forma se consideró que los tres algoritmos de búsqueda se publicaron en un intervalo de tiempo muy cercano (2024-2025). Además, los algoritmos no requieren un ajuste de parámetros muy complejos.

Los resultados de simulación e implementación del esquema de control PID Robusto propuesto muestran una respuesta dinámica rápida, un desempeño dinámico robusto y una atenuación activa de perturbaciones durante el seguimiento de la trayectoria de la posición del RPTD. El resto de este artículo está organizado de la siguiente manera: La Sección 2 muestra la estructura, así como los modelos cinemáticos y dinámicos del RPTD. La Sección 3 presenta la estructura de control del controlador PID robusto con las especificaciones de diseño y el algoritmo de ajuste basado en AOZR. Los resultados de simulación y la implementación en código se discuten en la Sección 4. Finalmente, las conclusiones se presentan en la Sección 5.

2. Modelo dinámico del robot paralelo tipo delta

El RPTD es un robot de tres grados de libertad, que consiste de dos plataformas, una fija y una móvil. Las plataformas están unidas una de la otra por medio de tres eslabones, cada eslabón consta de dos partes: un brazo y un antebrazo. El brazo de cada eslabón está unido a la base fija por medio de una articulación rotacional donde se encuentra el actuador (Rahimi et al., 2022).

El antebrazo del robot se une al brazo por medio de articulaciones esféricas pasivas, en el otro extremo se une a la plataforma móvil de igual manera por articulaciones esféricas pasivas, como se puede observar en la Figura 1. Tradicionalmente, los actuadores se colocan en la plataforma fija del RPTD por lo cual, puede alcanzar altas velocidades para realizar tareas de pick and place, en la industria manufacturera (Serrano & Cardona, 2024), (Ohno & Takeda, 2020), (Elassal et al., 2024).

En la Figura 2 se muestran los parámetros físicos del RPTD en un sistema de coordenadas (O-*xyz*) y los sistemas de coordenadas de cada eslabón $(O_i - x_i y_i z_i, i = 1,2,3)$ definidos para el análisis matemático del RPTD. Los parámetros físicos del RPTD para la implantación se describen en la Tabla 1.

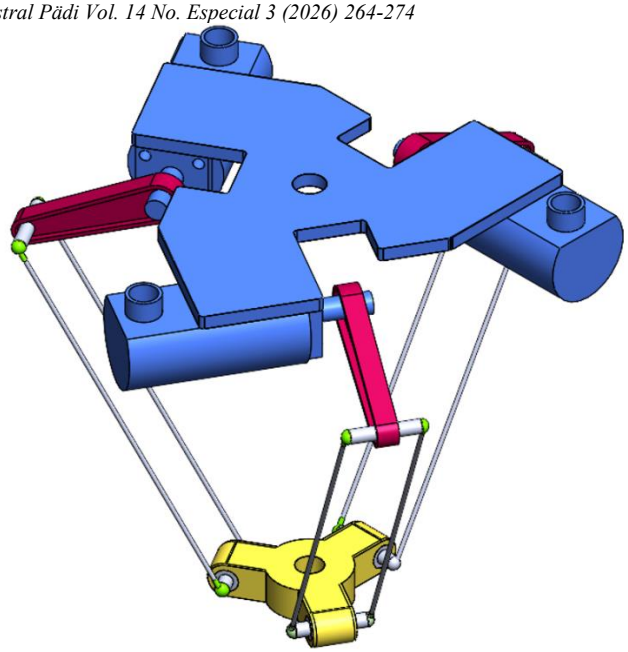


Figura 1: Diagrama esquemático del robot paralelo tipo delta

En la Figura 2 se puede observar que el RPTD tiene un sistema de coordenadas con referencia a la posición O-*xyz* ubicado en el centro de la plataforma fija; donde el eje z está en dirección opuesta a la fuerza de gravedad g, y el eje x está dirigido hacia el enlace superior del primer eslabón.

Para el análisis matemático se asume que cada cadena cinemática, tiene un sistema de coordenadas $O_i - x_i y_i z_i$, con $i = 1,2,3$ el cual se ubica a una distancia R del sistema $O - xyz$y el cual puede rotar a ángulos $\phi_i(0°, 120°, 240°)$, respectivamente. Para simplificar los eslabones superiores e inferiores de las tres cadenas cinemáticas se trasladan al origen del sistema de coordenadas base mediante el radio r de la plataforma móvil, como se muestra en la Figura 2.

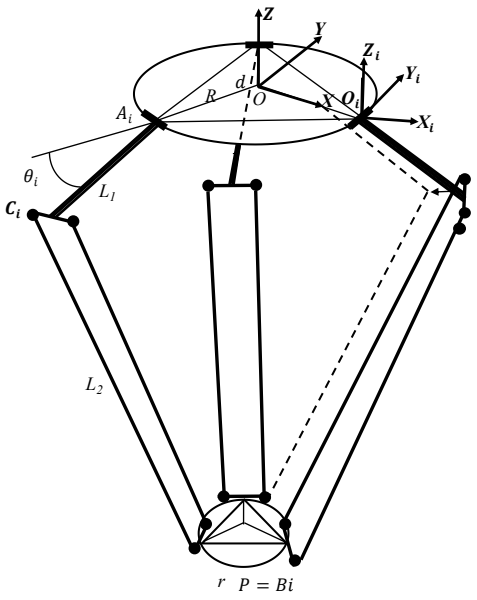


Figura 2. Sistemas de coordenadas base y parámetros físicos del RPTD para realizar el modelo matemático.

Tabla 1. Parámetros físicos del robot paralelo Delta.

Símbolo	Parámetro	Valor
R	Radio de la plataforma fija (m)	0.155
r	Radio de la plataforma móvil (m)	0.045

d	$R-r$	0.11
L_1	Longitud de la articulación superior (m)	0.3
L_2	Longitud de la articulación inferior (m)	0.65
θ	Angulo de posición de la articulación de las articulaciones superiores	$\theta = [\theta_1, \theta_2, \theta_3]$
I_m	Momento de inercia del eje del motor ($\mathrm{kg.m^2}$)	1.8×10^{-4}
m_1	Masa de la articulación superior (kg)	0.457
m_{1j}	Masa de la articulación superior (kg)	0.05
m_2	Masa de la articulación inferior (kg)	0.313
m_3	Masa de la plataforma móvil (kg)	0.406
m_4	Masa de la carga (kg)	0.1

2.1. Modelo cinemático

En general, el modelo cinemático de un robot consiste en el modelo cinemático inverso y el modelo cinemático directo. El modelo cinemático inverso del RPTD ha sido ampliamente investigado en (Asif et al., 2024), (Liu et al., 2019), (Rahimi et al., 2022). Definiendo las coordenadas $P = [x_p, y_p, z_p]^T$ de la plataforma móvil en el sistema de coordenadas cartesianas base, el modelo cinemático inverso calcula los ángulos $\left(\theta = [\theta_p, \theta_p, \theta_p]^T\right)$ de las articulaciones de revolución actuadas de los eslabones superiores. El modelo cinemático inverso puede resumirse de la siguiente manera (Taghirad, 2025). Dadas las coordenadas $P = [x_p, y_p, z_p]^T$ de la plataforma móvil en el sistema *O–xyz*, su posición en cada sistema de coordenadas de las cadenas O_i–$x_i\ y_i z_i (i = 1,2,3)$ se convierte utilizando una matriz de transformación de rotación, como se muestra a continuación.

$$\begin{bmatrix} x_{p_i} \\ y_{p_i} \\ z_{p_i} \end{bmatrix} = \begin{bmatrix} cos\emptyset_i & -sin\emptyset_i & 0 \\ sin\emptyset_i & cos\emptyset_i & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} \quad \left(\emptyset_i = 0, \frac{2\pi}{3}, \frac{4\pi}{3}; i = 1,2,3\right) \tag{1}$$

dado que el eslabón de la articulaciones superior L_1solo puede rotar en el plano $x_i O_i z_i$, la articulación $J_i$se encuentra sobre una trayectoria circular con centro en A_iy con radio L_1.

$$(x_i - d)^2 + z_i^2 = L_1^2,\ (i = 1,2,3) \tag{2}$$

Además, la articulación J_i también se encuentra ubicada sobre una superficie esférica con centro en B_iy radio L_2.

$$\left(x_i - x_{p_i}\right)^2 + \left(y_i - y_{p_i}\right)^2 + \left(z_i - z_{p_i}\right)^2 = L_2^2, (i = 1,2,3) \tag{3}$$

Entonces, $J_i$se encuentra en la intersección entre (2) y (3) en el siguiente plano

$$y_i = 0,\ (i = 1,2,3) \tag{4}$$

de (2) a (4), la posición de la articulación J_i en cada sistema de coordenadas de la cadena puede expresarse como:

$$x_{J_i} = \frac{(d - A_i B_i) \pm \sqrt{(1 + A_i^2) L_1^2 - (A_i d + B_i)^2}}{1 + A_i^2}, y_{J_i} = 0, z_{J_i} = A_i x_{J_i} + B_i, \tag{5}$$

donde:

$$A_i = -\frac{d - x_{p_i}}{z_{p_i}}, B_i = -\frac{x_{p_i}^2 + y_{p_i}^2 + z_{p_i}^2 + L_1^2 - L_2^2 - d^2}{2 z_{p_i}}$$

Considerando el ángulo de revolución de la articulación del eslabón superior como se puede definir como:

$$\theta_i = arctan\,2\ \left(z_{J_i}, x_{J_i} - d\right), i = 1,2,3. \tag{6}$$

En el RPTD, el proceso para calcular la cinemática directa se calcula mediante la posición cartesiana $P = [x_p, y_p, z_p]^T$ de la plataforma móvil en el sistema de coordenadas base, dados los ángulos de las articulaciones de revolución $\theta = [\theta_1, \theta_2, \theta_3]^T$de las articulaciones superiores.

La posición de la articulación J_i en cada eslabón en el sistema de coordenadas se obtiene como sigue:

$$x'_{j_i} = d + L_1 cos\theta_i, y'_{j_i} = 0,\ z'_{j_i} = -L_1 sin\theta_i. \tag{7}$$

Entonces, la posición de J_i en el sistema de coordenadas base se obtiene utilizando la matriz de transformación de rotación de la siguiente manera:

$$\begin{bmatrix} x_{J_i} \\ y_{J_i} \\ z_{J_i} \end{bmatrix} = \begin{bmatrix} cos\emptyset_i & -sin\emptyset_i & 0 \\ sin\emptyset_i & cos\emptyset_i & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x'_{j_i} \\ y'_{j_i} \\ z'_{j_i} \end{bmatrix}, \quad \left(\emptyset_i = 0, \frac{2\pi}{3}, \frac{4\pi}{3}; i = 1,2,3\right) \tag{8}$$

Además, la articulación J_i también se encuentra sobre una superficie esférica centrada en Pcon radio L_2

$$\left(x_{J_i} - x_p\right)^2 + \left(y_{J_i} - y_p\right)^2 + \left(z_{J_i} - z_p\right)^2 = L_2^2\ (i = 1,2,3) \tag{9}$$

Utilizando (7) a (9),

$$\begin{cases} \left(x_{J_1} - x_p\right)^2 + y_p^2 + \left(z_{J_1} - z_p\right)^2 = L_2^2 \\ \left(x_{J_2} - x_{J_1}\right)x_p + y_{J_2} y_p + \left(z_{J_2} - z_{J_1}\right)z_p = \frac{1}{2}(\rho_2 - \rho_1) \\ \left(x_{J_3} - x_{J_1}\right)x_p + y_{J_3} y_p + \left(z_{J_3} - z_{J_1}\right)z_p = \frac{1}{2}(\rho_3 - \rho_1) \end{cases} \tag{10}$$

donde $\rho_i = x_{J_i}^2 + y_{J_i}^2 + z_{J_i}^2, (i = 1,2,3)$. De las dos últimas ecuaciones de (10),

$$x_p = K_1 z_p + H_1,\ y_p = K_2 z_p + H_2, \tag{11}$$

donde

$$K_1 = \frac{y_{J_3}\left(z_{J_2} - z_{J_1}\right) - y_{J_2}\left(z_{J_3} - z_{J_1}\right)}{\left(x_{J_3} - x_{J_1}\right)y_{J_2} - \left(x_{J_2} - x_{J_1}\right)y_{J_3}},$$

$$H_1 = \frac{1}{2} \cdot \frac{y_{J_2}(\rho_3 - \rho_1) - y_{J_3}(\rho_2 - \rho_1)}{\left(x_{J_3} - x_{J_1}\right)y_{J_2} - \left(x_{J_2} - x_{J_1}\right)y_{J_3}}$$

$$K_2 = \frac{\left(x_{J_2} - x_{J_1}\right)\left(z_{J_3} - z_{J_1}\right) - \left(x_{J_3} - x_{J_1}\right)\left(z_{J_2} - z_{J_1}\right)}{\left(x_{J_3} - x_{J_1}\right)y_{J_2} - \left(x_{J_2} - x_{J_1}\right)y_{J_3}},$$

$$H_2 = \frac{1}{2} \cdot \frac{\left(x_{J_3} - x_{J_1}\right)(\rho_2 - \rho_1) - \left(x_{J_2} - x_{J_1}\right)(\rho_3 - \rho_1)}{\left(x_{J_3} - x_{J_1}\right)y_{J_2} - \left(x_{J_2} - x_{J_1}\right)y_{J_3}}$$

A partir de la primera ecuación de (10) y (11):

$$\begin{aligned} (1 + K_1^2 + K_2^2) z_p^2 + 2\left(K_1 H_1 + K_2 H_2 - K_1 x_{J_1} - z_{J_1}\right) z_p \\ + \left(H_1 - x_{J_1}\right)^2 + H_2^2 + z_{J_i}^2 - L_2^2 = 0. \end{aligned} \tag{12}$$

Entonces, la posición de P se obtiene de la siguiente manera:

$$x_p = K_1 z_p + H_1,\ y_p = K_2 z_p + H_2, z_p = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, \tag{13}$$

donde:

$$a = (1 + K_1^2 + K_2^2), b = 2\left(K_1 H_1 + K_2 H_2 - K_1 x_{J_1} - z_{J_1}\right),$$

$$c = \left(H_1 - x_{J_1}\right)^2 + H_2^2 + z_{J_i}^2 - L_2^2.$$

2.2. Modelo dinámico

La ecuación dinámica para el RPTD puede expresarse de la siguiente manera:

$$M(\theta)\ddot{\theta} + C\left(\theta, \dot{\theta}\right)\dot{\theta} + N(\theta) = \tau, \tag{14}$$

donde $M(\theta)$es la matriz de inercia, $C\left(\theta, \dot{\theta}\right)$es la matriz de Coriolis, $N(\theta)$es el vector de gravedad, y $\tau = [\tau_1, \tau_2, \tau_3]^T$es el vector de torque aplicado en las articulaciones superiores. La matriz de inercia $M(\theta)$ en (14) se define como:

$$M(\theta) = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} = I_{bt} + \left(m_3 + m_4 + \frac{1}{2}m_2\right) J^T J, \tag{15}$$

donde I_{bt}es la matriz de inercia del eje de accionamiento y se define de la siguiente manera:

$$I_{bt}=\begin{bmatrix} I_{bti} & 0 & 0 \\ 0 & I_{bti} & 0 \\ 0 & 0 & I_{bti} \end{bmatrix}, I_{bti}=I_m+L_1^2\left(\frac{m_1}{3}+m_{1j}\frac{2}{3}m_2\right),$$

Y J se muestra en (15), la cual se define como la matriz Jacobiana definida como:

$$J=-\begin{bmatrix} s_1^T \\ s_2^T \\ s_3^T \end{bmatrix}^{-1}\begin{bmatrix} s_1^T b_1 & 0 & 0 \\ 0 & s_2^T b_2 & 0 \\ 0 & 0 & s_3^T b_3 \end{bmatrix},$$

$$s_i=\begin{bmatrix} x \\ y \\ z \end{bmatrix}-T_i\left(\begin{bmatrix} d \\ 0 \\ 0 \end{bmatrix}+\begin{bmatrix} L_i cos\theta_i \\ 0 \\ -L_i cos\theta_i \end{bmatrix}\right)$$

$$b_i=T_i\begin{bmatrix} L_i cos\theta_i \\ 0 \\ -L_i cos\theta_i \end{bmatrix},$$

$$T_i=\begin{bmatrix} cos\phi_i & -sin\phi_i & 0 \\ sin\phi_i & cos\phi_i & 0 \\ 0 & 0 & 1 \end{bmatrix}, \left(\phi_i=0,\frac{2\pi}{3},\frac{4\pi}{3}; i=1,2,3\right)$$

donde $X=[x,y,z]^{\mathrm{T}}=f(\theta)$ es la posición del centro de la plataforma móvil. La matriz $C\left(\theta,\dot{\theta}\right)$ en (14) se puede calcular como se muestra a continuación

$$C\left(\theta,\dot{\theta}\right)=-J^T\left(m_3+m_4+\tfrac{1}{3}m_2\right)\begin{bmatrix} s_1^T \\ s_2^T \\ s_3^T \end{bmatrix}^{-1}\left(\begin{bmatrix} \dot{s}_1^T \\ \dot{s}_2^T \\ \dot{s}_3^T \end{bmatrix}^{-1} J+M_{sb}\right) \quad (16)$$

$$M_{sb}=\begin{bmatrix} \dot{s}_1^T b_1+s_1^T\dot{b}_1 & 0 & 0 \\ 0 & \dot{s}_2^T b_2+s_2^T\dot{b}_2 & 0 \\ 0 & 0 & \dot{s}_3^T b_3+s_3^T\dot{b}_3 \end{bmatrix},$$

$$\dot{s}_i=\dot{X}+b_i\dot{\theta}_i=J\dot{\theta}+b_i\dot{\theta}_i,$$

$$\dot{b}_i=T_i\begin{bmatrix} L_i cos\theta_i \\ 0 \\ -L_i cos\theta_i \end{bmatrix}\dot{\theta}_i, (i=1,2,3).$$

Además, $N(\theta)$ en (14) se calcula de la siguiente manera:

$$N(\theta)=J^T\left(m_3+m_4+\frac{3}{2}m_2\right)\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} -L_1\left(\frac{1}{2}m_1+m_{1j}+\frac{1}{2}m_2\right)g\begin{bmatrix} cos\theta_1 \\ cos\theta_2 \\ cos\theta_3 \end{bmatrix} \quad (17)$$

3. Control PID basado en un algoritmo híbrido de optimización AOZR

Para regular el movimiento y posición de la plataforma móvil del RPTD se utiliza la estrategia de control CTC, que emplea un enfoque de linealización por retroalimentación. Al aplicar esta estrategia de control, el seguimiento de la trayectoria de la posición de la plataforma móvil del RPTD se puede regular mediante técnicas de control lineal. Considerando la siguiente ley de linealización:

$$\tau=\widehat{M}(\theta)u+\hat{C}\left(\theta,\dot{\theta}\right)\dot{\theta}+\widehat{N}(\theta), \quad (18)$$

donde u es una entrada del nuevo sistema linealizado, $\widehat{M}(\theta)$, $\hat{C}\left(\theta,\dot{\theta}\right)$, $\widehat{N}(\theta)$ son estimaciones de $M(\theta)$, $C\left(\theta,\dot{\theta}\right)$ y $N(\theta)$, respectivamente. Suponiendo que la Ecuación (18) es lo suficientemente precisa para describir el modelo dinámico del RPTD, se puede asumir que:

$$M(\theta)=\widehat{M}(\theta), \quad C\left(\theta,\dot{\theta}\right)=\hat{C}\left(\theta,\dot{\theta}\right), \quad N(\theta)=\widehat{N}(\theta).$$

Entonces, obtenemos (19) a partir de (14) y (18), como:

$$\widehat{M}(\theta)u=M(\theta)\ddot{\theta} \quad (19)$$

Considerando que la matriz de inercia $M(\theta)$es invertible, el modelo dinámico (14) se puede linealizar y se desacoplar, por lo cual

$$\ddot{\theta}(t)=u(t). \quad (20)$$

Entonces el nuevo modelo del RPTD es un sistema linealizado y desacoplado de las variables $\theta(t)$. Sin embargo, en la práctica, las incertidumbres paramétricas del RPTD y las perturbaciones externas no deben ignorarse, ya que afectan el desempeño del esquema de control. En consecuencia, las perturbaciones internas, externas y los errores pueden atenuarse mediante el esquema de control por retroalimentación para el sistema lineal (20). Para el sistema linealizado (20), se propone la siguiente estrategia de control basada en controladores tradicionales PID:

$$u(t)=\ddot{\theta}_d(t)+k_p e(t)+k_d\dot{e}(t) \quad (21)$$

donde $e(t)=\theta_d(t)-\theta(t)$ es el error de la articulación, $\dot{e}(t)$es el error de velocidad de la articulación, $\ddot{\theta}_d(t)$es la aceleración deseada de la articulación, k_i, k_p, T_f y k_d son los parámetros del esquema de control que buscará el AOZR para que $e(t)$ tienda a cero en un tiempo finito. Si reemplazamos el controlador derivativo ideal en (21) por un controlador derivativo real, entonces el controlador PID resultante puede expresarse de la siguiente manera:

$$C_{(s)}=K_P\left(1+T_i+T_d s\right)\frac{1}{T_f s+1}, \quad (22)$$

donde K_p es el coeficiente proporcional, T_i es la constante del integrador, T_d es la constante de tiempo derivativa. Además, el filtro de primer orden del lado derecho con constante de tiempo T_f filtra el ruido de alta frecuencia.

$$T_f=\min\{K_P,\ T_i/10,\ T_f/10,\ T_d/10\} \quad (23)$$

La Figura 3 muestra la estructura del controlador PID el RPTD. La mayoría de los controladores para RPTD utilizan sensores en las articulaciones, para medir fácilmente los ángulos de las articulaciones. La trayectoria se planifica en el espacio cartesiano, y la trayectoria en el espacio articular se obtiene utilizando el modelo cinemático inverso. El controlador utiliza la trayectoria en el espacio articular y la posición medida de las articulaciones. Posteriormente, la posición en el espacio cartesiano se obtiene utilizando el modelo cinemático directo.

En la Figura 3, el ángulo theta (θ) es la posición medida de la articulación y θ_d es la posición deseada de la articulación. El error en la posición de la articulación se reduce mediante un controlador PID individual con filtro de primer orden. Y el vector de torque τ se calcula utilizando la ecuación (18).

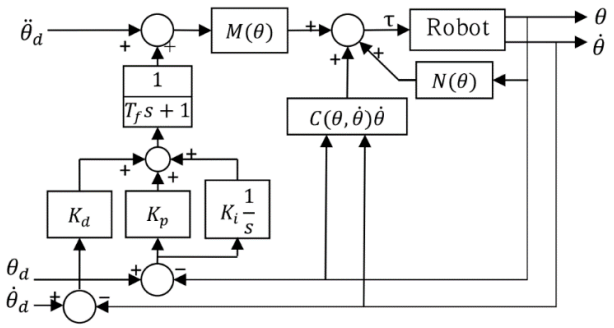


Figura 3. Estructura para el controlador PID robusto para el RPTD.

3.1. Especificaciones de diseño para el controlador PID

En esta sección, se diseña el esquema de control PID para obtener un controlador robusto ante variaciones paramétricas de la planta. Para mejorar la robustez del PID la respuesta en fase del sistema en lazo abierto debe ser plana en la frecuencia de cruce de la ganancia ω_{cg} y que sea casi constante en un

intervalo alrededor de ω_{cg}. Por lo tanto, el diseño del controlador PID puede plantearse como la solución de un problema de optimización, el cual minimiza la siguiente función costo

$$J = \int_{\omega_i}^{\omega_f} e_{FD}^2(\omega)d\omega = \int_{\omega_i}^{\omega_f}\left(\varphi(\omega) - (\varphi_m - \pi)\right)^2 d\omega \quad (24)$$

donde el error $e_{FD}(\omega)$ se calcula en un intervalo predefinido entre ω_iy $\omega_f$alrededor de ω_{cg}. $\phi(\omega)$es la fase del sistema en lazo abierto a la frecuencia ωy ϕ_m es el margen de fase. La ventaja de este enfoque es que la banda de frecuencia $[\omega_i, \omega_f]$, en la cual el sistema en lazo cerrado es robusto frente a variaciones de las ganancias, puede seleccionarse en función de los requerimientos del usuario. En este sentido, se pueden incluir más especificaciones en la función de costo (24) con las siguientes restricciones(14, 15): a) la frecuencia de cruce ω_{cg} debe estar dentro del intervalo de frecuencia $[\omega_{cgl}, \omega_{cgh}]$, y el margen de fase debe ser al menos ϕ_m. Entonces, estas restricciones pueden describirse de la siguiente manera:

$$\omega_{cgl} \leq \omega_{cg} \leq \omega_{cgh}, \quad (25)$$

$$\arg\left(C(j\omega_{cg})G(j\omega_{cg})\right) \geq -\pi + \varphi_m, \quad (26)$$

donde $C(j\omega_{cg})$ y $G(j\omega_{cg})$son las funciones de transferencia del controlador (22) y de la planta (20), respectivamente; b) para atenuar el ruido de alta frecuencia, la función de sensibilidad complementaria debe cumplir la siguiente condición:

$$\left|T(j\omega) = \frac{C(j\omega)G(j\omega)}{1+C(j\omega)G(j\omega)}\right|_{dB} \leq M_t dB, \quad \forall\omega \geq \omega_t \frac{rad}{s}, |T(j\omega_t)|_{dB} = M_t dB \quad (27)$$

donde M_t representa la atenuación de ruido deseada para frecuencias $\omega \geq \omega_t$rad/s; y c) para asegurar el rechazo de perturbaciones en la salida, la función de sensibilidad $S(j\omega)$ está restringida por la siguiente condición

$$\left|S((j\omega) = \frac{1}{1+C(j\omega)G(j\omega)}\right|_{dB} =\leq M_s dB, \quad \forall\omega \geq \omega_s \frac{rad}{s}, |S(j\omega_s)|_{dB} = M_s dB \quad (28)$$

donde $M_s$es el valor deseado de la función de sensibilidad para frecuencias $\omega \leq \omega_s$. Por lo tanto, el problema de diseño del controlador PID Robusto se plantea como un problema de optimización no lineal con restricciones no lineales, como se muestra a continuación:

$$min\ \ J(K_p, K_i, T_f T_d) = \int_{\omega_i}^{\omega_f}\left(\varphi(\omega) - (\varphi_m - \pi)\right)^2 d\omega \quad (29)$$

Para el problema de optimización (29), es difícil encontrar la solución óptima global con métodos tradicionales de optimización por aproximación, como el método del gradiente, ya que la función de costo y las restricciones son funciones no lineales y demasiado complejas con respecto a los parámetros de diseño (K_p, K_d, K_i, T_f). Por lo tanto, en la siguiente sección, el problema de optimización no lineal con restricciones no lineales se resolverá de manera eficiente mediante el AOZR.

4. Algoritmo de Optimización del Zorro de Rüppell (AOZR)

El AOZR es un algoritmo de optimización metaheurístico desarrollado recientemente que se inspira en las estrategias de búsqueda de alimento de los zorros de Rüppell [7]. Estos animales poseen sentidos excepcionales de la vista, el oído y el olfato, los cuales utilizan para localizar eficazmente a sus presas. El algoritmo modela matemáticamente y abstrae estos comportamientos guiados por los sentidos, con el objetivo de lograr un equilibrio sólido entre la exploración del espacio de búsqueda y la explotación de soluciones prometedoras. En las siguientes secciones se describen el proceso de inicialización, las formulaciones matemáticas principales y los mecanismos clave que conforman el algoritmo AOZR [7-8].

4.1 Inicialización del AOZR

Como algoritmo basado en población, el AOZR comienza generando aleatoriamente un conjunto inicial de soluciones candidatas, que representan las posiciones de los zorros de Rüppell dentro del espacio de búsqueda. En un escenario con una población de *n* zorros en un espacio de *d* dimensiones, estas posiciones se representan como:

$$y = \begin{bmatrix} y_1^1 & y_2^1 & \cdots & y_d^1 \\ y_1^2 & y_2^2 & \cdots & y_d^2 \\ \vdots & \vdots & \ddots & \vdots \\ y_1^n & y_2^n & \cdots & y_d^n \end{bmatrix} \quad (30)$$

donde y_j^i representa la posición del *i-ésimo* zorro en la *j-ésima* dimensión. Las posiciones iniciales se generan de manera uniforme y aleatoria dentro de los límites del espacio de búsqueda [7]:

$$y_j^i = l_j + r \times (u_j - l_j) \quad (31)$$

donde l_j y $u_j$son los límites inferior y superior de la *j-ésima* dimensión, respectivamente, y *r* es un número aleatorio en el intervalo [0,1].

4.2 Modelo matemático del AOZR

El AOZR simula el comportamiento de búsqueda de alimento de los zorros de Rüppell mediante distintas fases de comportamiento, cada una de las cuales contribuye al equilibrio entre exploración y explotación del algoritmo. La búsqueda durante el día representa una exploración global basada en la vista y el oído, mientras que la búsqueda nocturna se centra en una exploración local más refinada utilizando señales auditivas y olfativas. El algoritmo se enfoca en las soluciones más prometedoras mediante un seguimiento basado en el olfato, mientras que el movimiento colectivo dirige a la población hacia la mejor solución encontrada. En conjunto, estas estrategias permiten que el AOZR explore eficazmente espacios de búsqueda complejos y evite quedar atrapado en óptimos locales [6].

4.3 Búsqueda de presas durante el día

Durante el día, los zorros de Rüppell utilizan principalmente su aguda vista y su oído para localizar a sus presas. La actualización de la posición durante la búsqueda diurna está dada por:

$$y_{k+1}^i = x_k^i + r_index \times (x_randx_k^i) + r_index \times (fbest_k - x_k^i) \quad (32)$$

donde x_k^i representa la posición del *i-ésimo* zorro en la iteración *k*, mientras que $fbest_k$ representa la mejor solución global encontrada hasta ese momento. El término *x_index* es un índice generado de forma estocástica, calculado como $[4 \times rand_1 + 1] \times rand_2$. Además, *x_rand* corresponde a una posición seleccionada aleatoriamente de la población existente.

4.4 Búsqueda de presas durante la noche

Durante la noche, los zorros de Rüppell dependen en mayor medida de su oído y del movimiento de rotación de sus orejas para detectar a sus presas. La actualización de la posición es:

$$y_{k+1}^{i} = y_{rotate}^{i} + \beta \times randn\,(1,d) \times flag \quad (33)$$

donde y_{rotate}^{i} representa la posición ajustada con base en el comportamiento de rotación de las orejas característico de los zorros de Rüppell. El parámetro β representa un tamaño de paso pequeño, generalmente establecido en (1×10^{-10}), lo que permite realizar ajustes precisos. La variable *flag* alterna entre -1 y 1, permitiendo cambios de dirección durante el proceso de búsqueda.

4.5 Localización de la presa mediante el sentido del olfato

Cuando la presa se encuentra cerca, los zorros utilizan su sentido del olfato para refinar la búsqueda. La actualización de la posición está dada por:

$$y_{k+1}^{i} = \begin{cases} x_k^i + \rho \times \left(x_{rand1} - x_k^i\right) \times r_2 \\ +\rho \times \left(fbest_k - x_k^i\right) \times r_3 \; if \; r_1 \geq smell \\ \\ x_{rand1} + \beta \times randn\,(1,d) \text{ otra cosa} \end{cases} \quad (34)$$

donde ρ es un parámetro aleatorio que controla la influencia del olfato, y *smell* es una función iterativa que representa la capacidad olfativa:

$$smell = \frac{0.1}{\left|acos\left(\frac{2}{1+e^{(k/2-k)/100}}\right)\right|} \quad (35)$$

4.6 Movimiento hacia el mejor zorro de Rüppell

Los zorros también se desplazan colectivamente hacia las mejores posiciones conocidas. La actualización de la posición está dada por [7]:

$$y_{k+1}^{i} = \begin{cases} x_k^i + c_0 \times \left(x_{rand} - x_k^i\right) \times r_1 + \\ c_1\left(fbest_k - x_k^i\right) \times r_2 \quad \text{si } rand \geq 0.1 \\ \\ \frac{x_k^i + z_k^i}{2} \qquad \text{otra cosa} \end{cases} \quad (36)$$

donde z_k^i es una posición intermedia que se actualiza utilizando las mejores posiciones conocidas.

4.7 Implementación del algoritmo AOZR

El algoritmo AOZR comienza inicializando una población de zorros y posteriormente actualiza sus posiciones mediante cuatro comportamientos fundamentales: (1) búsqueda durante el día utilizando la vista y el oído, acompañada de movimientos de rotación; (2) explotación durante la noche basada en el oído y el olfato; (3) refinamiento local basado en el olfato cuando la presa se encuentra cerca; y (4) movimiento colaborativo hacia las soluciones más prometedoras. Los parámetros sensoriales vista, oído y olfato se adaptan dinámicamente a lo largo del proceso. Durante cada iteración, las nuevas posiciones son evaluadas, se actualiza la mejor solución global y se preserva la diversidad mediante mecanismos de aleatorización. El algoritmo finaliza al alcanzar un número máximo predeterminado de iteraciones y devuelve la solución óptima encontrada a través de este enfoque bioinspirado que equilibra la exploración y la explotación. Los pasos principales presentados en el Algoritmo 1 pueden utilizarse para resumir el pseudocódigo del AOZR

Pseudocódigo 1: Algoritmo AOZR

1: Inicializar los parámetros del RPTD y del AOZR.
2: Evaluar la posición de la población inicial.
3: **while** $(k < K)$ **do**
4: **if** $(p - 0.5) \geq 0$ **then**
5: **for** $i = 1$ to n **do**
6: **if** ($s \geq$ h) **then**
7: **if** (*rand* ≥ 0.25) **then**
8: Actualizar la posición de los zorros usando (32).
9: **else**
10: Actualizar la posición de los zorros usando (33).
11: **end if**
12: **else**
13: **if** (*rand* ≥ 0.75) **then**
14: Actualizar la posición de los zorros usando (34).
15: **end if**
16: **else**
17: **for** *i*=1 to *n* **do**
18: **if** (*rand* ≥ 0.1) **then**
19: Actualizar la posición de los zorros usando (36).
20: **end if**
21: **end for**
22: $k = k+1$
23: **end while**
24: Devolver la mejor solución encontrada hasta el momento

5. Resultados del esquema de control PID robusto para el robot paralelo Delta

Para diseñar el esquema de control robusto del RPTD se definen las especificaciones que debe cumplir el controlador en el dominio de la frecuencia definidas en (25)-(28); I) margen de fase mínimo $\phi_m = 45°$; II) la frecuencia de cruce de ganancia ω_{cg} deberá estar en el intervalo $[\omega_{cgl}, \omega_{cgh}] = [50rad/s, 150rad/s]$; III) el intervalo $(\omega_f - \omega_i)$, donde se pretende que el margen de fase sea plana alrededor de la frecuencia de cruce de ganancia igual a 0.25 dec.; y IV) la atenuación de ruido de alta frecuencia: $\omega t = 350\,rad/s$, $Mt = -15\,dB$, es decir, $\forall\omega \geq \omega t, |T(j\omega)|dB \leq -15dB$, para rechazar perturbaciones cuando: $\omega_s = 10\,rad/s$, $Ms = -15\,dB$, es decir, $\forall\omega \leq \omega_s, |S(j\omega)|dB \leq -15dB$.

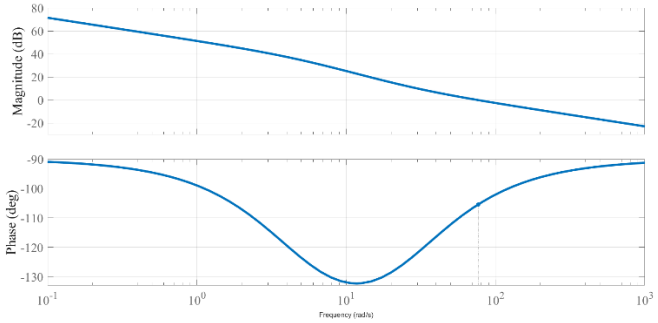


Figura 4. Diagrama de bode de la respuesta en la frecuencia en lazo abierto (Pm=74.6° a los 76.6 rad/seg)

La tarea de encontrar los parámetros (k_p, k_i, T_d, T_f) del controlador PID robusto para cada uno de los actuadores del RPTD, este problema se convierte en un problema de optimización no lineal (29). La población seleccionada para los tres algoritmos fue de 50 individuos de búsqueda, con un número máximo de 30 iteraciones.

Los valores estadísticos de los algoritmos de los resultados después de realizar la tarea de optimización (AOZR, AOP y AOH) tales como la media, la desviación estándar (SD) y los valores mínimos (min) y máximo (max), se muestran en la Tabla 2, mientras que los resultados del análisis comparativo

mediante diagramas de caja se muestran en la Figura 5. La tabla y la figura anteriores muestran que el AOZR tiene un mejor rendimiento estadístico comparado con los otros dos algoritmos de optimización.

Tabla 2: Valores estadísticos de la respuesta de la FC

Índice estadístico	AOZR	AOP	AOH
Media	0.3139	0.8484	1.7644
Desviación estándar	0.2423	0.7249	0.7847
Mejor valor	0.0480	0.1101	0.3180
Peor valor	1.0390	2.1806	2.7579

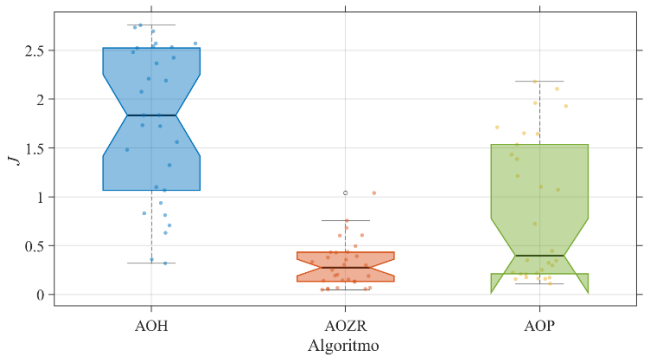


Figura 5: Resultados de los diagramas de caja de los algoritmos AOH, AOP y AOZR.

Los parámetros óptimos del controlador PID robusto para cada uno de los actuadores del RPTD calculados por el AOZR se muestran en la Tabla 3.

Tabla 3: Parámetros óptimos del controlador PID robusto para los actuadores del RPTD calculados por el AOZR.

# Actuador	k_p	k_i	T_d	T_f
Actuador 1	691.583	964.645	0.0294	0.0014
Actuador 2	408.114	192.069	0.0242	0.0021
Actuador 3	1296.169	3.500	0.0317	0.0015

El desempeño del esquema de control PID robusto se compara con un esquema de control PD adaptativo basado en redes neuronales artificiales denominadas B-spline (PD-BSNN) desarrollada en (Escorcia-Hernández et al., 2020) y un PD+G (Mahdi S. M., et al., 2025). Con las ganancias optimas del controlador PID robusto encontradas, se construye el modelo en MATLAB/Simulink utilizando los parámetros físicos del RPTD que se muestran en la Tabla 1, con el modelo dinámico no lineal de la sección dos y la estructura de control de la Figura 4.

5.1. Caso 1: Condición operativa espiral de Arquímedes

Para comparar el desempeño de los controladores en el seguimiento de la trayectoria de la posición de la plataforma móvil se utiliza como referencia la espiral de Arquímedes, donde $P_{xd}=cos(t)r_s$, $P_{yd}=sen(t)r_s$ y $P_{zd}=-0.6+0.006t$ con $r_s=0.005t$ para $t=[0\ 9\pi]$. Las condiciones iniciales de las posiciones angulares y velocidades angulares del robot son $q_i=[0.3122, 0.3122, 0.3122, 0, 0, 0]$. Las figuras 6-8 muestran el desempeño dinámico de las variables mecánicas de interés del robot en lazo cerrado. La figura 6 muestra el seguimiento de la trayectoria de posición en 3D del RPTD con los tres esquemas de control utilizados. La figura 7 muestra la respuesta dinámica de la posición de la plataforma móvil del robot en coordenadas x e y.

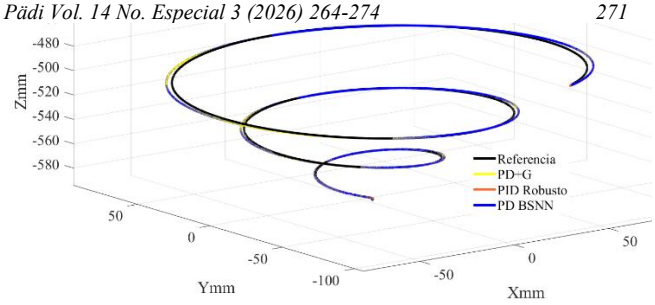


Figura 6: Respuesta al seguimiento de la trayectoria de espiral de Arquímedes de la plataforma móvil, en perspectiva 3D.

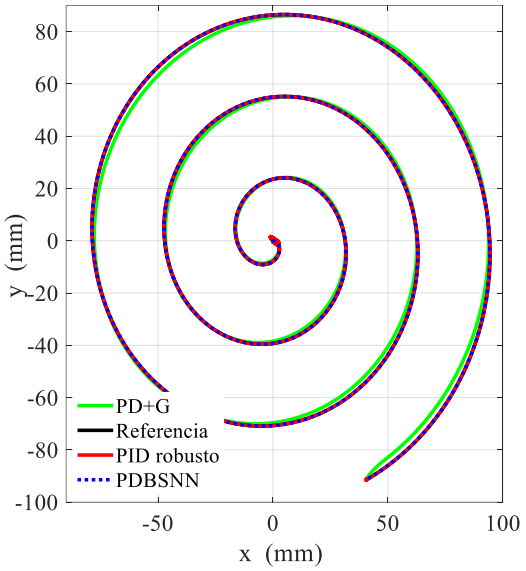


Figura 7: Respuesta al seguimiento de la trayectoria de espiral de Arquímedes de la plataforma móvil en coordenadas cartesianas (xy).

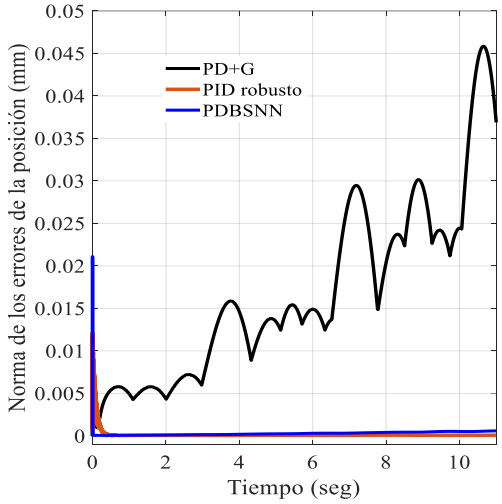


Figura 8: Norma de los errores del seguimiento a la trayectoria de espiral de Arquímedes de la plataforma móvil

En las figuras 6-7 se puede ver que los tres controladores están muy cerca de la señal de referencia, pero el esquema de control PID robusto presenta un menor error que el controlador PDBSNN en estado transitorio. La norma de los errores de los tres esquemas de control en lazo cerrado se muestra en la Figura 8. En la figura se aprecia que la señal de error para el controlador PID Robusto es prácticamente cero para el desplazamiento cartesiano, a diferencia del controlador PDBSNN, pero en el regulador PD+G se observa claramente que el error aumenta en función del tiempo. Cabe destacar que en el esquema de control PID Robusto tiene doce ganancias optimas que fueron ajustadas de manera adecuada por el

AOZR, por lo cual, si el algoritmo está bien sintonizado, por consecuencia es posible lograr un excelente desempeño para de tareas de seguimiento de trayectorias de referencia para la plataforma móvil del RPTD.

5.2. *Caso 2: Condición operativa "pick-and-place" (recoger y colocar)*

Para analizar el desempeño del controlador PID robusto propuesto, se evalúa utilizando una tarea del tipo "pick-and-place" utilizada comúnmente en la industria manufacturera. La trayectoria "pick-and-place" de la plataforma móvil se implementa utilizando la curva de aceleración trapezoidal modificada, en el cual la aceleración y el cambio de aceleración son pequeños. En la Tabla 4 se muestran los puntos de trayectoria del recorrido de "pick-and-place" para la plataforma móvil del RPTD.

Tabla 4: Puntos de trayectoria de la condición operativa "pick-and-place"

Punto	Posición	Punto	Posición
1	(-0.1, -0.1,-0.6)	3	(0.1, 0.1,-0.6)
2	(-0.1, 0.1,-0.6)	4	(-0.1, 0.1,-0.6)

La trayectoria se genera usando un polinomio de interpolación de quinto orden. La función del polinomio se puede definir como:

$$r(t) = \frac{10t^3}{t_f^3} - \frac{15t^4}{t_f^4} + \frac{6t^5}{t_f^5} \quad (37)$$

donde $r(t)$ es función de la trayectoria en dos puntos y t_f es la duración del movimiento.

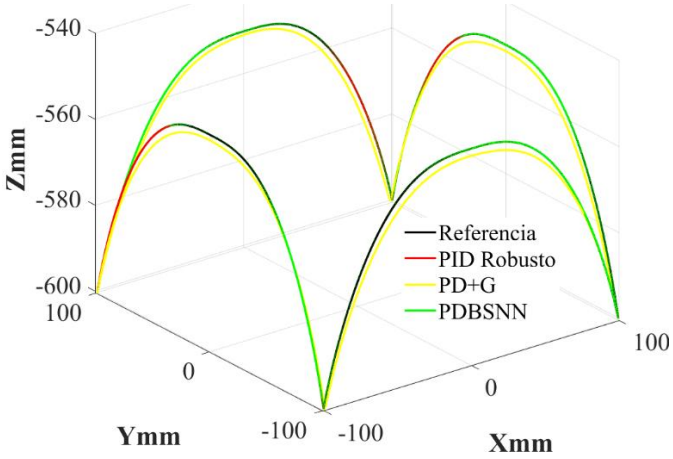


Figura 9: Seguimiento de la trayectoria "pick-and-place" planeada para el RPTD de los tres esquemas de control

La Figura 9 muestra la respuesta del seguimiento de la trayectoria "pick-and-place" planeada para el RPTD de los tres esquemas de control. Para observar a detalle la respuesta en el tiempo de las variables de interés, es necesario realizar las transformaciones cinemáticas correspondientes.

La Figura 10 muestra la respuesta transitoria de las posiciones del RPTD en coordenadas cartesianas. Las Figuras 11 y 12 muestran los errores del seguimiento a la trayectoria de la posición de la plataforma móvil, del controlador PID robusto, el PD+G y el esquema de control PDBSNN. En ambas figuras se aprecia que los controladores PID Robusto y el PDBSNN siguen adecuadamente la señal de la trayectoria de la referencia, sin embargo, el PID robusto muestra un error menor en el estado transitorio a diferencia del controlador PDBSNN; lo cual se verifica en la Tabla 5.

Tabla 5: Comparación del ECM generada por los tres controladores.

	Error RMS de posición			Error RMS eslabones		
Algoritmo	x	y	z	θ_1	θ_2	θ_3
PID Robusto	0.0132	0.0154	0.0167	3.1104	2.8417	1.1877
PD+G	7.468	7.527	4.958	13.102	22.714	11.097
PDBSNN	0.0415	0.0392	0.0283	3.109	2.8442	1.1880

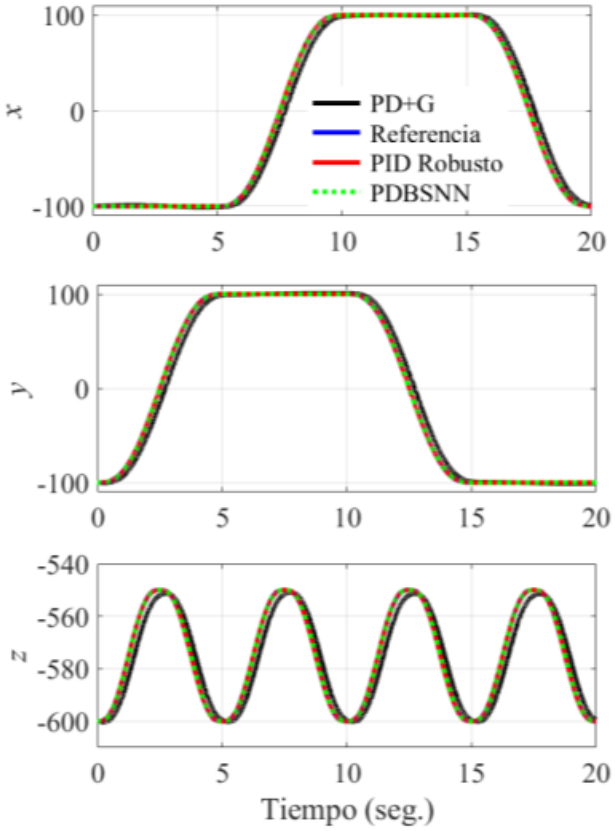


Figura 10: Trayectoria de los ejes x, y y z del RPTD en coordenadas cartesianas.

Pero el controlador tradicional PD+G el error se acumula a medida que el robot sigue la trayectoria deseada. En la Tabla 5 se muestran los resultados del error cuadrático medio del seguimiento a la trayectoria deseada que tiene como resultado cada controlador. Como se observa en estas figuras y en la Tabla 5, el controlador PID robusto presenta un menor error de seguimiento de la trayectoria deseada.

Para realizar la comparación de métricas de desempeño, en la Tabla 6 se muestran las magnitudes de cuatro indicadores de desempeño ampliamente conocidos en el ámbito del control automático y la optimización, denominadas: el error cuadrático integral (ISE), el error absoluto integral (IAE), la integral del error cuadrático del tiempo (ITSE), la integral del error absoluto ponderado en el tiempo (ITAE).

Tabla 6: Comparación de métricas de desempeño.

	ITAE			ISE		
Algoritmo	x	y	z	x	y	z
PID Robusto	2.5850	2.6681	3.3238	0.0035	0.0048	0.0056
PD+G	1174.5	772.474	844.664	1115.6	1133.5	491.749
PDBSNN	6.1852	3.7667	4.6888	0.0344	0.0307	0.0160
	ITSE			IAE		
PID Robusto	0.0367	0.0439	0.0560	0.2478	0.2785	0.3323
PD+G	14235	8737.5	4825.5	95.3918	96.7376	84.8381
PDBSNN	0.4302	0.2304	0.1570	0.4970	0.4882	0.4741

Analizando los resultados de la Tabla 6, se puede observar que el esquema de control PID Robusto registra los menores valores en todas las métricas de desempeño. En el análisis de esta sección se muestran que el PID Robusto tiene un mejor desempeño en el seguimiento de la trayectoria "Pick and Place" porque muestra el mejor desempeño en cuatro de cuatro indicadores en términos de error absoluto acumulado.

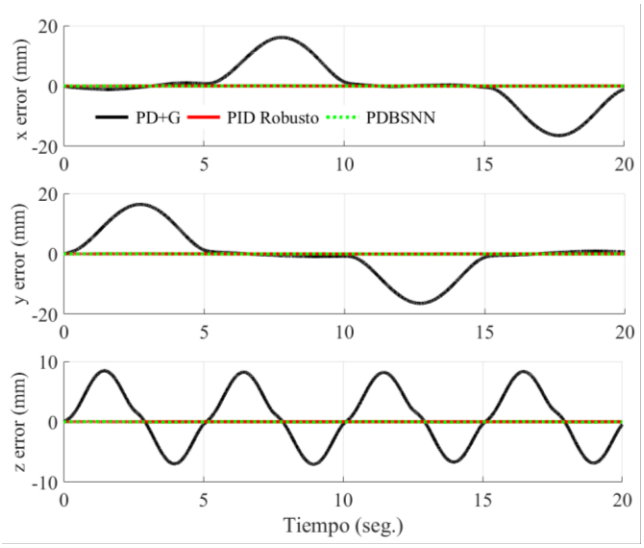


Figura 11: Error del seguimiento de la trayectoria en coordenadas cartesianas.

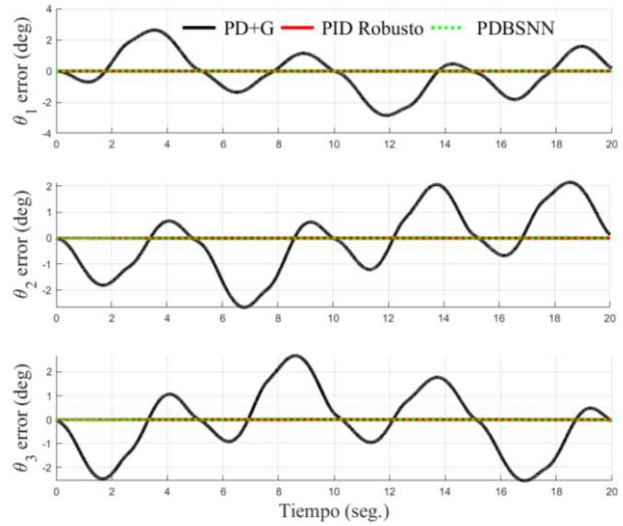


Figura 12: Error del seguimiento de la trayectoria en desplazamiento angular.

6. Conclusiones

En este trabajo, se propone una metodología basada en restricciones de respuesta a la frecuencia para ajustar un controlador PID robusto óptimo para resolver el problema del seguimiento de la trayectoria de la plataforma móvil de un RPTD mediante el AOZR. El problema de diseñar un controlador PID robusto para el RPTD se debe a resolver el modelo dinámico no lineal del RPTD en conjunto con la formulación de restricciones no lineales, para garantizar la robustez y atenuar perturbaciones ante diferentes condiciones operativas. Para obtener un óptimo global más preciso de este problema, se ha utilizado el AOZR.

Los resultados de simulación y la implementación en Matlab muestran que el controlador PID robusto presenta un desempeño robusto ante la presencia de incertidumbres en el modelo y perturbaciones externas. Aunque el esquema de control propuesto requiere una gran cantidad de cálculos para encontrar los valores óptimos de sintonización, esto se resuelve mediante software con buena capacidad de cómputo. Vale la pena mencionar que el controlador PID robusto tiene cuatro parámetros, por lo que presenta un mejor desempeño en estado transitorio como en estado estacionario en comparación con el PD+G y el PDBSNN.

Declaración sobre el uso de Inteligencia Artificial generativa y tecnologías asistidas por IA

Durante la preparación del presente manuscrito, los autores utilizaron Grammarly para la corrección gramatical y mejorar la redacción del texto. Todo el contenido generado o sugerido por la herramienta fue revisado y editado por los autores, quienes asumen plena responsabilidad por la exactitud, originalidad, integridad y rigor científico del contenido final del manuscrito.

Referencias

Abdollahzadeh, B., Khodadadi, N., Barshandeh, S., Trojovský, P., Gharehchopogh, F. S., El-Kenawy, E. M., Abualigah, L., & Mirjalili, S. (2024). Puma optimizer (PO): a novel metaheuristic optimization algorithm and its application in machine learning. *Cluster Computing, 27*(4), 5235–5283. https://doi.org/10.1007/s10586-023-04221-5

Amiri, M. H., Hashjin, N. M., Montazeri, M., Mirjalili, S., & Khodadadi, N. (2024). Hippopotamus optimization algorithm: a novel nature-inspired optimization algorithm. *Scientific Reports, 14*(1), 5032. https://doi.org/10.1038/s41598-024-54910-3

Asif, S., Bueno, M., Ferreira, P., Anandan, P., Zhang, Z., Yao, Y., Ragunathan, G., Tinkler, L., Sotoodeh-Bahraini, M., Lohse, N., Webb, P., Hutabarat, W., & Tiwari, A. (2024). Rapid and automated configuration of robot manufacturing cells. *Robotics and Computer-Integrated Manufacturing, 92*, 102862. https://doi.org/10.1016/j.rcim.2024.102862

Braik, M., & Al-Hiary, H. (2025). Rüppell's fox optimizer: a novel meta-heuristic approach for solving global optimization problems. *Cluster Computing, 28*(5). https://doi.org/10.1007/s10586-024-04950-1

Choubey, C., & Ohri, J. (2021). Tuning of LQR-PID controller to control parallel manipulator. Neural Computing And Applications, 34(4), 3283-3297. https://doi.org/10.1007/s00521-021-06608-0

Saleem, D., Eleashy, H., & Shamseldin, M. A. (2025). Optimal TID Tracking Control for Industrial Delta Robot Based on Harmony Search. Journal Of Robotics And Control (JRC), 6(2), 862-870. https://doi.org/10.18196/jrc.v6i2.23678

Elassal, A., Abdelaal, M., Osama, M., & Elhnidy, H. (2024). Low-cost parallel delta robot for a pick-and-place application with the support of the vision system. *e-Prime - Advances in Electrical Engineering Electronics and Energy, 8*, 100518. https://doi.org/10.1016/j.prime.2024.100518

Escorcia-Hernández, J. M., Aguilar-Sierra, H., Aguilar-Mejia, O., Chemori, A., & Arroyo-Núñez, J. H. (2020). A new adaptive RISE feedforward approach based on associative memory neural networks for the control of PKMs. *Journal of Intelligent & Robotic Systems, 100*(3–4), 827–847. https://doi.org/10.1007/s10846-020-01242-9

Dehghani, M., Hubalovsky, S., & Trojovsky, P. (2022). Tasmanian Devil Optimization: a new Bio-Inspired optimization algorithm for solving optimization algorithm. IEEE Access, 10, 19599–19620. https://doi.org/10.1109/access.2022.3151641

Haji, V. H., & Monje, C. A. (2017a). Fractional-order PID control of a chopper-fed DC motor drive using a novel firefly algorithm with dynamic

control mechanism. *Soft Computing, 22*(18), 6135–6146. https://doi.org/10.1007/s00500-017-2677-5

Haji, V. H., & Monje, C. A. (2017b). Fractional order fuzzy-PID control of a combined cycle power plant using particle swarm optimization algorithm with an improved dynamic parameters selection. *Applied Soft Computing, 58*, 256–264. https://doi.org/10.1016/j.asoc.2017.04.033

Kong, Y., Ri, J., & Pak, Y. (2026). Robust fractional order PID control of a Delta parallel robot for trajectory tracking. *Journal of the Brazilian Society of Mechanical Sciences and Engineering, 48*(3). https://doi.org/10.1007/s40430-025-06121-8

Liu, C., Cao, G., Qu, Y., & Cheng, Y. (2019). An improved PSO algorithm for time-optimal trajectory planning of Delta robot in intelligent packaging. *The International Journal of Advanced Manufacturing Technology, 107*(3–4), 1091–1099. https://doi.org/10.1007/s00170-019-04421-7

Mahdi, S. M., Abdulkareem, A. I., & Humaidi, A. J. (2026). Improving the Tracking Performance of Par-4 Robot Using Nature-Inspired Optimization Algorithms to Tune the PID Controller. Iraqi Journal Of Computer Communication Control And System Engineering, 26(1), 101-116. https://doi.org/10.33103/2617-3352.1520

Mahdi, S. M., Abdulkareem, A. I., Humaidi, A. J., Mhdawi, A. K. A., & Al-Raweshidy, H. (2025). Comprehensive review of control techniques for various mechanisms of parallel robots. *IEEE Access, 13*, 63381–63416. https://doi.org/10.1109/access.2025.3557937

Ohno, M., & Takeda, Y. (2020). Design of target trajectories for the detection of joint clearances in parallel robot based on the actuation torque measurement. *Mechanism and Machine Theory, 155*, 104081. https://doi.org/10.1016/j.mechmachtheory.2020.104081

Pak, Y., Kong, Y., & Ri, J. (2022). Robust PID optimal tuning of a Delta parallel robot based on a hybrid optimization algorithm of particle swarm optimization and differential evolution. Robotica, 41(4), 1159-1178. https://doi.org/10.1017/s0263574722001606

Rahimi, S., Jalali, H., Yazdi, M. R. H., Kalhor, A., & Masouleh, M. T. (2022). Design and practical implementation of a neural network self-tuned inverse dynamic controller for a 3-DoF Delta parallel robot based on arc length function for smooth trajectory tracking. *Mechatronics, 84*, 102772. https://doi.org/10.1016/j.mechatronics.2022.102772

Serrano, F. E., & Cardona, M. (2024). Robust H-infinity control of Delta parallel robot with disturbances. In *Lecture notes in networks and systems* (pp. 56–67). https://doi.org/10.1007/978-3-031-54763-8_6

Taghirad, H. D. (2025). *Parallel robots: Mechanics and control*. CRC Press.

Tajdari, F., & Toulkani, N. E. (2021). Implementation and intelligent gain tuning feedback–based optimal torque control of a rotary parallel robot. *Journal of Vibration and Control, 28*(19–20), 2678–2695. https://doi.org/10.1177/10775463211019177

Tepljakov, A., Gonzalez, E. A., Petlenkov, E., Belikov, J., Monje, C. A., & Petráš, I. (2015). Incorporation of fractional-order dynamics into an existing PI/PID DC motor control loop. *ISA Transactions, 60*, 262–273. https://doi.org/10.1016/j.isatra.2015.11.012

Yigit, T., & Celik, H. (2023). Dimension synthesis and optimized FOPID control of the Delta robot with moth swarm algorithm. *Arabian Journal for Science and Engineering, 48*(5), 6889–6902. https://doi.org/10.1007/s13369-023-07601-6

Zhao, R., Wu, L., & Chen, Y. (2020). Robust control for nonlinear Delta parallel robot with uncertainty: an online estimation approach. *IEEE Access, 8*, 97604–97617. https://doi.org/10.1109/access.2020.2997093

.