

Optimización mediante algoritmos metaheurísticos de un control de velocidad PI para un motor de imanes permanentes

Optimization by metaheuristic algorithms of a PI speed control for a permanent magnet motor

Omar Aguilar-Mejía ^{a,*}, Nadia S. Zuñiga-Peña ^a, Norberto Hernández-Romero ^a
Juan C. Seck-Tuoh-Mora ^a, Joselito Medina-Marín ^a

^aÁrea Académica de Ingeniería y Arquitectura, Universidad Autónoma del Estado de Hidalgo, 42184, Pachuca, Hidalgo, México.

Resumen

El trabajo reporta la evaluación de desempeño de cuatro algoritmos metaheurísticos para optimizar el control de velocidad de un motor de imanes permanentes (MIP); el controlador tiene integrados tres controles PI, en consecuencia, hay que sintonizar seis ganancias. Los algoritmos considerados; son enjambre de partículas conocido como (PSO), evolución diferencial mejorada (LSHADE), optimización de búsqueda dorada (GSO), y axolote mexicano (MAO). El rendimiento de los controles del motor se evalúa sometiénolos a cambios en la referencia de velocidad y a perturbaciones en el par de carga. LSHADE obtiene el mejor rendimiento con un valor mínimo de la función de costo de 1.1474, le sigue el PSO convencional con un valor de 2.9567, el MAO con un valor de 13.09, y el GSO con 266.63. Aunque los algoritmos GSO y MAO son técnicas de optimización recientes, no han conseguido mejorar a los algoritmos PSO y LSHADE para este caso concreto de regulación de velocidad MIP.

Palabras Clave: Motor de imanes permanentes, optimización, algoritmos metaheurísticos, control PI, modelo no lineal.

Abstract

The paper reports the performance evaluation of four metaheuristic algorithms to optimize the speed control of a permanent magnet motor (MIP); the controller has three integrated PI controls, and consequently, six gains have to be tuned. The algorithms considered are particle swarm known as (PSO), enhanced differential evolution (LSHADE), golden search optimization (GSO), and Mexican axolotl (MAO). The performance of the motor controls is evaluated by subjecting them to changes in speed reference and load torque perturbations. LSAHADE obtains the best performance with a minimum cost function value of 1.1474; conventional PSO follows it with a value of 2.9567, MAO with a value of 13.09, and GSO with 266.63. Although the GSO and MAO algorithms are recent optimization techniques, they still need to improve upon the PSO and LSHADE algorithms for this particular case of MIP speed regulation.

Keywords: Permanent magnet motor, optimization, metaheuristics algorithms, PI controller, nonlinear model.

1. Introducción

Los motores de imanes permanentes (MIP) tienen una amplia gama de aplicaciones industriales como son los vehículos eléctricos e híbridos, bicicletas y patinetas eléctricas, drones, aviones eléctricos, aerogeneradores, robótica y hasta equipos médicos. Esta amplia variedad de aplicaciones del motor MIP es debido a sus ventajas respecto a los otros motores como el

de corriente directa, síncrono o asíncrono. Los motores de imanes permanentes son altamente eficientes en la conversión de energía eléctrica en mecánica alcanzando valores de hasta el 97%. Esto se debe a la reducción de pérdidas de energía en comparación con otros tipos de motores, lo que resulta en un menor consumo de energía. Estos motores pueden generar una gran cantidad de potencia en un tamaño relativamente pequeño

* Autor para correspondencia: omar.aguilar@uaeh.edu.mx

Correo electrónico: omar_aguilar@uaeh.edu.mx (Omar Aguilar-Mejía), nadia_zuniga@uaeh.edu.mx (Nadia Samantha Zuñiga-Peña), nromero@uaeh.edu.mx (Norberto Hernández-Romero), jseck@uaeh.edu.mx (Juan Carlos Seck-Touh-Mora), jmedina@uaeh.edu.mx (Joselito Medina-Marín).

Historial del manuscrito: recibido el 11/07/2024, última versión-revisada recibida el 15/09/2024, aceptado el 19/09/2024, publicado el 14/11/2024. **DOI:** <https://doi.org/10.29057/icbi.v12iEspecial3.13422>



de un controlador de velocidad en un MIP, el cual este formado por tres controles proporcional-integral (PI) y la finalidad es determinar cual proporciona un mejor rendimiento. Por lo tanto, las principales contribuciones de este trabajo son:

1. Un esquema de optimización de algoritmos metaheurísticos con un modelo no lineal del control de velocidad de MIP.
2. Se determinó quién tiene el mejor desempeño en el control de velocidad del MIP con los algoritmos metaheurísticos evaluados (LSHADE, PSO, GSO, MAO).
3. Se determinaron 6 ganancias de los controladores PI que proporciona una buena respuesta dinámica a perturbaciones en el par del carga y cambios en la velocidad de referencia del MIP.

2. Modelación matemática del motor de imanes permanentes (MIP) y control PI

Para describir el modelo matemático de la MSIP se hacen las siguientes consideraciones: a) el sistema magnético es lineal, b) las sobrecorrientes en el estator pueden ser toleradas sin provocar una desmagnetización significativa en el imán permanente, y c) no se consideran devanados de amortiguamiento. Considerando las suposiciones anteriores y utilizando conceptos básicos de electricidad y magnetismo, se pueden obtener las ecuaciones diferenciales eléctricas y mecánicas que describen el comportamiento dinámico del MSIP. Obtener y conocer el sistema de ecuaciones que describen al MSIP es fundamental para diseñar e implementar en simulación los esquemas de control que se presentan en las siguientes secciones. Las ecuaciones diferenciales que describen la dinámica de los enlaces del flujo en el estator del MSIP se pueden expresar como:

$$\begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} \frac{d\lambda_{sa}}{dt} \\ \frac{d\lambda_{sb}}{dt} \\ \frac{d\lambda_{sc}}{dt} \end{bmatrix} \quad (1)$$

donde i_a , i_b y i_c son las corrientes trifásicas del estator; v_a , v_b y v_c son las señales de voltaje trifásico aplicadas al estator del MSIP, R_s es la resistencia del estator, λ_{sa} , λ_{sb} y λ_{sc} son los enlaces de flujo en el estator, los cuales se pueden determinar como

$$[\lambda_{sabc}] = [L_{so} + L_{sv}] [i_{sabc}] + [\lambda_{rabc}] \quad (2)$$

donde $[\lambda_{rabc}]$ son los enlaces de flujo del rotor generados por el imán permanente, que se pueden definir como:

$$\begin{bmatrix} \lambda_{ra} \\ \lambda_{rb} \\ \lambda_{rc} \end{bmatrix} = \lambda_m \begin{bmatrix} \cos(n_p\theta) \\ \cos(n_p\theta - 2\pi/3) \\ \cos(n_p\theta + 2\pi/3) \end{bmatrix} \quad (3)$$

donde n_p es el número de pares de polos, λ_m es la magnitud del flujo magnético generado por el imán permanente montado en el rotor y θ es la posición del rotor. Para el MSIP las matrices de las inductancias L_{so} y L_{sv} se calculan con sigue:

$$[L_{so}] = \begin{bmatrix} L_{so} & M_{so} & M_{so} \\ M_{so} & L_{so} & M_{so} \\ M_{so} & M_{so} & L_{so} \end{bmatrix} \quad (4)$$

$$[L_{sv}] = L_{sv} \begin{bmatrix} \cos(k\theta) & \cos(k\theta - \frac{2\pi}{3}) & \cos(k\theta + \frac{2\pi}{3}) \\ \cos(k\theta - \frac{2\pi}{3}) & \sin(k\theta + \frac{2\pi}{3}) & \cos(k\theta) \\ \cos(k\theta + \frac{2\pi}{3}) & \cos(k\theta) & \cos(k\theta - \frac{2\pi}{3}) \end{bmatrix} \quad (5)$$

donde $k = 2n_p$, L_{so} , L_{sv} y M_{so} son parámetros positivos que dependen de la estructura de la máquina. Sustituyendo (Ec. 2) en (Ec. 1) se obtiene:

$$[V_{sabc}] = R_s [i_{sabc}] + \frac{d}{dt} \{ [L_{ss}] [i_{sabc}] + [\lambda_{rabc}] \} \quad (6)$$

La expresión para calcular el par electromagnético se puede escribir en variables de la máquina o coordenadas abc como:

$$T_e = \frac{n_p}{2} \frac{\partial W_c}{\partial \theta} \quad (7)$$

donde

$$W_c = \frac{1}{2} i_{sabc}^T L_{ss} i_{sabc} + i_{sabc} \lambda_{rabc} + W_p \quad (8)$$

donde W_p es la energía del acoplamiento magnético debido a la presencia del imán permanente. Notase que el modelo matemático anterior que representa la dinámica del MSIP en coordenadas trifásicas abc es muy complejo porque las inductancias varían con el tiempo y dependen de la posición del rotor. Para evitar trabajar con un modelo dinámico variante en el tiempo y disminuir la complejidad del diseño del controlador se utiliza la transformación de coordenadas $abc - dq$. El bloque $dq - abc$ que se muestra en la figura 1 es la matriz de la transformada de Park Ec. 9 que convierte un sistema de coordenadas trifásicas abc a un marco de referencia arbitrario de dos coordenadas ortogonales dq en sincronía con la posición del rotor. Además, si se trabaja en coordenadas dq el sistema de ecuaciones diferenciales se vuelven invariantes en el tiempo (Ec. 1-6) y el modelo dinámico en un marco de referencia dq estará sincronizado con el ángulo del rotor. Por lo tanto, el ángulo del rotor puede medirse utilizando un sensor acoplado al eje del rotor o puede calcularse utilizando alguna técnica de estimación (Yazdani e Iravani, 2010).

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin(\theta) & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} \quad (9)$$

El bloque $abc - dq$ es la transformada inversa de Park (Ec. 10) que convierte un sistema de voltajes ortogonales en un sistema de voltajes trifásicos:

$$\begin{bmatrix} m_a \\ m_b \\ m_c \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ \cos(\theta - \frac{2\pi}{3}) & \sin(\theta - \frac{2\pi}{3}) \\ \cos(\theta + \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \end{bmatrix} \begin{bmatrix} m_d \\ m_q \end{bmatrix} \quad (10)$$

Aplicando la transformada de Park a la (Ec. 6) se obtiene el modelo matemático del MSIP en un marco de referencia estacionario dq , las ecuaciones diferenciales que definen la dinámica del motor en un marco de referencia dq son:

$$V_d = -R_s i_d + \omega_e L_q i_q - L_d \frac{di_d}{dt} \quad (11)$$

$$V_q = -R_s i_q + \omega_e L_d i_d - \omega_e \lambda_m - L_q \frac{di_q}{dt} \quad (12)$$

donde V_d y V_q son las componentes de voltaje en el estator en coordenadas dq , i_d e i_q son las componentes de corriente en el estator en los ejes dq , L_d y L_q son las inductancias en el estator en los ejes dq , ω_e es la velocidad angular eléctrica y r . La ecuación mecánica que relacionan el par producido por el motor con la dinámica del rotor

$$T_e = 1,5 \frac{n_p}{2} [\lambda_m i_q + (L_d - L_q) i_d i_q] \quad (13)$$

donde T_e es el par electromagnético. La ecuación del movimiento angular del rotor queda definida por:

$$J \frac{d\omega_r}{dt} = T_e - T_L - B\omega_r \quad (14)$$

donde J es el momento de inercia del rotor, ω_r es la velocidad angular del rotor, T_L es el par de carga y B es el coeficiente de fricción viscosa de rodamientos.

El modelo de un MIP es un sistema no lineal, con un fuerte acoplamiento entre sus variables mecánicas y eléctricas. En la figura 1 se muestra un control vectorial de campo orientado que tiene como objetivo mantener el flujo eléctrico en el eje d en cero y para regular la velocidad, se ajusta el par mediante la componente de corriente en el eje q . El ángulo entre el flujo magnético del estator y el del rotor se denomina ángulo de carga. Para lograr el par máximo, el ángulo de carga debe ser de 90 grados. Los componentes de este sistema son: tres controladores PI, el inversor, la transformada Park junto con sus inversa, el sensor de posición y velocidad, y el inversor trifásico. El controlador está definido de la siguiente forma: w_r^* es la velocidad deseada en el motor, w_r es la velocidad del motor, e_w es desviación de la velocidad del motor, i_q^* es la corriente deseada en el eje q la cual se determina por la (Ec. 15),

$$i_q^* = k_{p1} e_w + k_{i1} \int e_w dt \quad (15)$$

donde e_q es la desviación de la corriente en el eje q , u_q es la acción de control en el eje q que se suministra al bloque $dq-abc$ y se determina por la (Ec. 16),

$$u_q = k_{p2} e_q + k_{i2} \int e_q dt \quad (16)$$

i_d^* es la corriente de referencia en el eje d para el caso es cero, e_d es la desviación de la corriente en el eje d y u_d es la acción de control en el eje q que se suministra al bloque $dq-abc$ y se determina por la (Ec. 17).

$$u_d = k_{p3} e_d + k_{i3} \int e_d dt \quad (17)$$

3. Información general de los algoritmos PSO, L-SHADE, GSO, MAO

Los algoritmos metaheurísticos son técnicas de optimización de alto nivel diseñadas para encontrar soluciones satisfactorias a problemas complejos de optimización en un tiempo razonable. Estos algoritmos son especialmente útiles para problemas

donde los métodos exactos resultan ineficaces debido a la complejidad del espacio de búsqueda o a la naturaleza no lineal y multimodal del problema (Brahim *et al.*, 2024). En esta sección se presentan los fundamentos de los algoritmos metaheurísticos a evaluar para la sintonización de parámetros de controlador PI. Se han seleccionado los algoritmos PSO, LSHADE, GSO y MAO, los cuales se describen a continuación.

3.1. Optimización por Enjambre de Partículas

El PSO es un método de optimización inspirado en el comportamiento que tienen las bandadas de pájaros o bancos de peces en los que, el movimiento de cada individuo (dirección, velocidad, aceleración), es el resultado de combinar las decisiones individuales de cada uno con el comportamiento del resto (Shabir y Singla, 2020), (Kennedy y Eberhart, 1995). En el PSO, cada miembro se denomina partícula, y estas partículas forman parte de un enjambre o nube. La figura 2 describe el proceso que lleva a cabo el PSO, donde inicialmente se genera la población de partículas y se inicializan los valores de posición y velocidad; se evalúa la función costo o *fitness*, para dar una calificación a cada partícula, posteriormente se actualizan los vectores de velocidad y posición, mediante las (Ecs. 18-19),

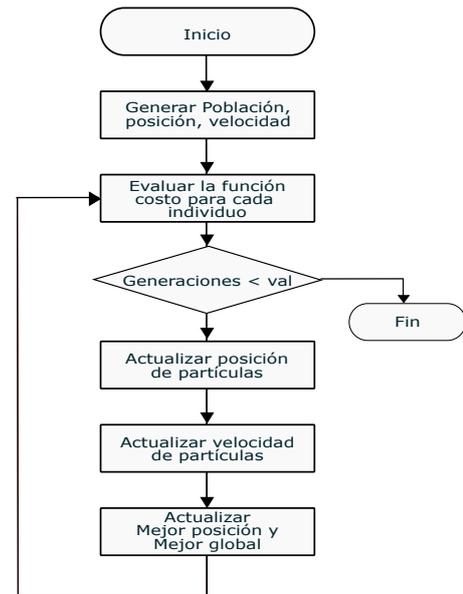


Figura 2: Diagrama de flujo del PSO.

$$v_{ij}(t + 1) = wv_{ij}(t) + c_1 R_1(P_{ij}(t) - x_{ij}(t)) + c_2 R_2(P_{gj}(t) - x_{ij}(t)), \quad (18)$$

$$x_{ij}(t + 1) = x_{ij}(t) + v_{ij}(t + 1), \quad (19)$$

Donde: $v_{ij}(t + 1)$, es la velocidad de la partícula i en el momento $t + 1$, es decir, la nueva velocidad. $v_{ij}(t)$, es la velocidad de la partícula i en el momento (t) , es decir, la velocidad actual. w , es el coeficiente de inercia, reduce o aumenta la velocidad de la partícula. c_1 , es el coeficiente cognitivo. R_1 , es un vector de valores aleatorios. $P_{ij}(t)$, es la mejor posición en la que ha estado la partícula i hasta el momento. $x_{ij}(t)$, es la posición de la partícula en el momento t . c_2 , es el coeficiente social. R_2 , es un vector de valores aleatorios. $P_{gj}(t)$, es la mejor posición de todo

el enjambre hasta el momento t , el mejor valor global. $x_{ij}(t+1)$, es la posición de la partícula i en el momento $t+1$, es decir, la nueva posición, después de la actualización de posición y velocidad, se vuelve a evaluar el *fitness* y se repite el proceso hasta alcanzar el número de generaciones establecido.

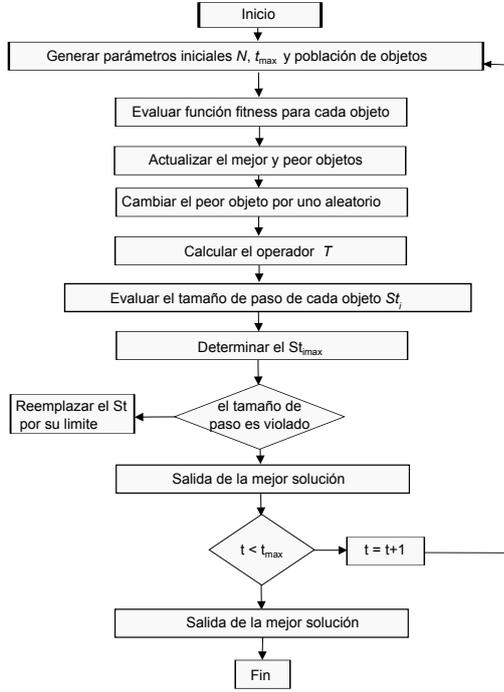


Figura 3: Diagrama de flujo del GSO.

3.2. Algoritmo de Optimización de Búsqueda Dorada

El Algoritmo de Optimización de Búsqueda Dorada (GSO) no imita un comportamiento biológico particular ni un fenómeno físico, sino que utiliza funciones matemáticas simples para diseñar su proceso de optimización. El GSO se basa en algoritmos como la optimización por enjambre de partículas (PSO) y los algoritmos de seno y coseno (SCA), logrando una mezcla óptima de exploración global y explotación local mientras evita la convergencia prematura. En el algoritmo GSO, las posiciones de los objetos se ajustan a través de un parámetro de tamaño de paso, similar al concepto de velocidad en PSO. Sin embargo, el GSO mejora este mecanismo mediante el empleo de funciones de seno y coseno, cuya naturaleza oscilatoria facilita el reposicionamiento estratégico de objetos para una explotación más efectiva dentro de límites espaciales definidos Noroozi *et al.* (2022). El proceso del GSO se muestra en la figura 3.

$$O_i = lbi + rand \times (ubi - lbi) \quad \text{for } i = 1, 2, \dots, N, \quad (20)$$

donde O_i representa la posición del i -ésimo objeto en el espacio de búsqueda, lbi y ubi son los límites inferior y superior del objeto, respectivamente. En la (Ec. 21) se calcula la evaluación del tamaño del paso, $C1$ y $C2$ son coeficientes, $r1$ y $r2$ son valores aleatorios entre 0 y 1, $Obest_i$ es la mejor posición previa del i -ésimo objeto, $Ogbest$ es la mejor posición global encontrada hasta ahora, y T es un operador de transferencia que controla

la transición de la exploración a la explotación.

$$St_i(t+1) = T \cdot St_i(t) + C1 \cdot \cos(r1) \cdot (Obest_i - x_i(t)) + C2 \cdot \sin(r2) \cdot (Ogbest - x_i(t)), \quad (21)$$

El Operador de Transferencia en (Ec. 22) ayuda al GSO a evitar trampas de óptimos locales al principio, mientras refina las soluciones cerca de la conclusión del algoritmo.

$$T = 100 \times \exp\left(-20 \times \frac{t}{t_{max}}\right). \quad (22)$$

La (Ec. 23) mejora la estabilidad de la convergencia del algoritmo con la limitación del tamaño del paso.

$$-St_{i,max} \leq St_i \leq St_{i,max} \quad (23)$$

donde $St_{i,max}$ es el tamaño máximo del paso, que es una fracción del rango de límites:

$$St_{i,max} = 0,1 \times (ubi - lbi) \quad (24)$$

La (Ec. 25) actualiza la posición del i -ésimo objeto basado en el tamaño del paso recién calculado.

$$O_i(t+1) = O_i(t) + St_i(t+1) \quad (25)$$

3.3. Algoritmo de Evolución Diferencial Mejorado

El Algoritmo L-SHADE es una mejora del algoritmo SHADE, el cual a su vez, es una variante del Algoritmo evolutivo Diferencial (DE), que agrega un parámetro de adaptación basado en el historial de éxito. El Algoritmo L-SHADE, extiende al SHADE (Tanabe y Fukunaga, 2013), mediante una reducción lineal del tamaño de la población (LPSR), con lo que la población decrece de manera continua, siguiendo el comportamiento de una función lineal (Tanabe y Fukunaga, 2014). El Algoritmo L-SHADE sigue la secuencia mostrada en la figura 4 y ha sido implementado para resolver funciones de prueba y se ha comparado contra algoritmos como el SHADE y otras variantes del DE, reportando una mejora significativa en el desempeño (Mohamed *et al.*, 2019).

$$N_{G+1} = \text{round} \left[\left(\frac{N^{\min} - N^{\text{init}}}{\text{MAX_NFE}} \right) \cdot \text{NFE} + N^{\text{init}} \right] \quad (26)$$

La (Ec. 26) describe el proceso de reducción lineal del tamaño de la población en el algoritmo L-SHADE, donde: N^{\min} es el tamaño mínimo de la población. N^{init} es el tamaño inicial de la población. MAX_NFE es el número máximo de evaluaciones de aptitud. NFE es el número actual de evaluaciones de aptitud. La Evolución Diferencial estándar tiene tres parámetros de control principales, que son el tamaño de la población N , el factor de escalado F y la tasa de cruce CR . La adaptación basada en el historial de éxito utiliza una memoria histórica M_{CR} , M_F que almacena un conjunto de valores de CR , F que han tenido un buen desempeño en el pasado, y genera nuevos pares de CR , F muestreando directamente el espacio de parámetros cerca de uno de estos pares almacenados.

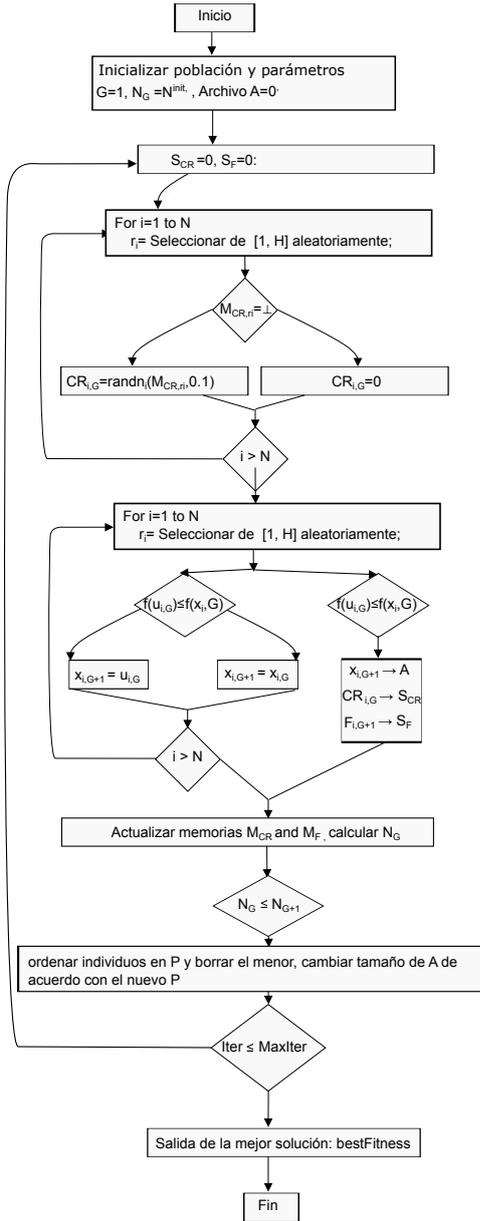


Figura 4: Diagrama de flujo del L-Shade.

3.4. Algoritmo de Ajolote Mexicano

El Algoritmo de Ajolote Mexicano (MAO) está inspirado en el ciclo de vida y las habilidades regenerativas del ajolote. Divide la población en machos y hembras, simulando su transición de estado de larva a adulto, procesos de lesión y restauración, y comportamientos reproductivos. El algoritmo *mao* consta de cuatro etapas iterativas, definidas por el acrónimo TIRA: Transición de estado de larva a adulto, Lesión y restauración, Reproducción y Clasificación (Villuendas-Rey et al., 2021). El proceso iterativo realizado en el algoritmo se muestra en la figura 5.

La (Ec. 27) representa la actualización de la posición de un ajolote macho m_{ji} en una dimensión i durante la transición de estado de larva a adulto.

El término $m_{best,i}$ es la posición del macho mejor adaptado en la misma dimensión. El parámetro λ es una constante de

transición que controla la magnitud de la adaptación.

$$m_{ji} \leftarrow m_{ji} + (m_{best,i} - m_{ji}) \cdot \lambda \quad (27)$$

La (Ec. 28) describe la actualización de la posición de un ajolote hembra f_{ji} en una dimensión i durante la transición de estado de larva a adulto. El término $f_{best,i}$ representa la posición de la hembra mejor adaptada. El parámetro λ es la constante de transición.

$$f_{ji} \leftarrow f_{ji} + (f_{best,i} - f_{ji}) \cdot \lambda \quad (28)$$

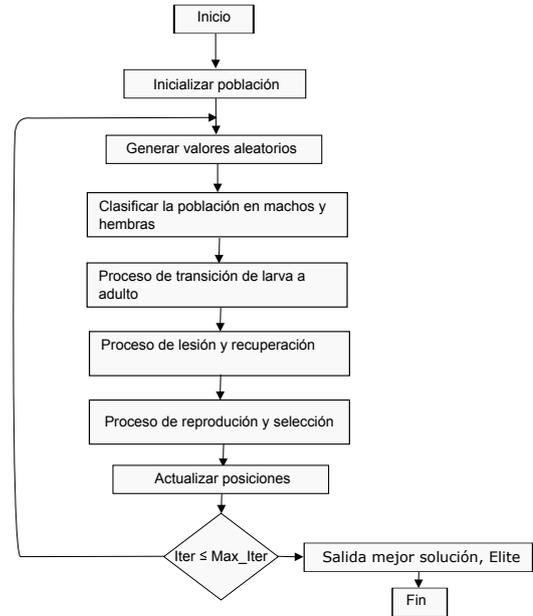


Figura 5: Diagrama de flujo del MAO.

La probabilidad inversa de transición para un ajolote macho p_{mj} basada en su valor de optimización o_{mj} es representada en la (Ec. 29). La probabilidad inversa se determina dividiendo el valor de optimización del individuo por la suma de los valores de optimización de todos los machos.

$$p_{mj} = \frac{o_{mj}}{\sum o_{mj}} \quad (29)$$

La (Ec. 30) calcula la probabilidad inversa de transición para un ajolote hembra p_{fj} basada en su valor de optimización o_{fj} . La probabilidad inversa se obtiene dividiendo el valor de optimización del individuo entre la suma de los valores de optimización de todas las hembras.

$$p_{fj} = \frac{o_{fj}}{\sum o_{fj}} \quad (30)$$

La (Ec. 31) describe la transición aleatoria de la posición de un axolote macho m_{ji} en una dimensión i . Aquí, \min_i y \max_i son los límites inferior y superior de la dimensión, respectivamente, y r_i es un número aleatorio en el rango $[0, 1]$.

$$m_{ji} \leftarrow \min_i + (\max_i - \min_i) \cdot r_i \quad (31)$$

La transición aleatoria de la posición de un ajolote hembra f_{ji} en una dimensión i se modela con la (Ec. 32), donde \min_i y \max_i son los límites inferior y superior de la dimensión, y r_i es

un número aleatorio en el rango [0, 1].

$$f_{ji} \leftarrow \min_i + (\max_i - \min_i) \cdot r_i \quad (32)$$

El proceso de restauración de una parte del cuerpo de un ajolote después de una lesión se representa mediante la (Ec. 33). El valor p'_{ji} se actualiza aleatoriamente dentro del rango de la dimensión, utilizando \min_i y \max_i como límites y r_i como un número aleatorio en el rango [0, 1].

$$p'_{ji} \leftarrow \min_i + (\max_i - \min_i) \cdot r_i \quad (33)$$

4. Proceso de optimización de controlador PI

En la figura 6 se muestra la forma de conectar un algoritmo metaheurístico al MIP, básicamente la función costo FF (Ec. 34) se construye con el valor cuadrático medio de la desviación de la velocidad angular del MIP.

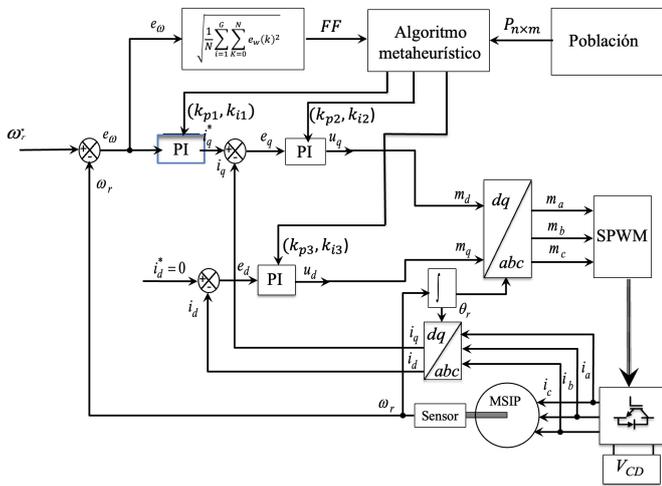


Figura 6: Conexión entre algoritmo metaheurístico y MIP.

$$FF^i(t) = \min RMSE = \min \left\{ \sqrt{\frac{1}{N} \sum_{t=0}^N (e_w^i(t))^2} \right\} \quad (34)$$

donde i es el número de individuo en la población, t es el número de muestra en el cierto instante de tiempo, dado que el tiempo de simulación del MIP es de $0 < t < 2$ s, el paso de integración para resolver el sistema de ecuaciones diferenciales es de $T = 0,00001$ s, entonces el número de muestras en todo el intervalo es $N = 20000$. Para cada individuo de la población definido por (Ec. 35) se integra por seis parámetros de los controles PI mostrados en las (Ecs. 15-17).

$$P^i = \{k_{p1} \ k_{i1} \ k_{p2} \ k_{i2} \ k_{p3} \ k_{i3}\}, \quad (35)$$

El orden de ejecución del algoritmo metaheurístico con el MIP de acuerdo con la figura 6 es el siguiente:

1. Se inicializa la población de forma aleatoria en el intervalo de búsqueda.

2. El algoritmo evalúa cada uno de los individuos de la población en el modelo dinámico del MIP, $P^i(t) \rightarrow FF^i(t)$.
3. Se evalúa si algún individuo cumple con el criterio de paro (CP), $FF^i(t) < CP$. Si cumple el programa termina, sino el programa continua.
4. Se realiza el algoritmo metaheurístico correspondiente y se obtiene una nueva población $P^i(t + 1)$.
5. Regresar al punto 2.

5. Experimentos y resultados

Las simulaciones se realizaron en una MAC OS Sonoma 14.5 (Intel Core i5, RAM 8GB y 1TB de disco duro), usando Matlab 2015. Para cada uno de los algoritmos PSO, LSHADE, GSO y MAO, el número de individuos 20, el número de iteraciones 20 y el número de repeticiones 30. El espacio de búsqueda para las ganancias de los controles está definido en el intervalo $S = [0, 300]$. Los parámetros usados para las simulaciones del MIP se reportan en la Tabla 1.

Tabla 1: Parámetros nominales del MIP

Parámetros del motor	Símbolo	Valor
Momento de inercia	J	$3,5e^{-5}$ Nm
Voltaje nominal	v	120 V
Resistencia del estator	r_s	$2,6 \Omega$
Inductancia del estator d	L_d	6,73 mH
Inductancia del estator q	L_q	6,73 mH
Inductancia de dispersión del estator	L_{ls}	$0,1L_q$
Flujo magnético	λ_m	0,319 Wb
Par de polos	n_p	2

Tabla 2: Estadística de valor RMSE

Estadística	PSO	LSHADE	GSO	MAO
Mejor	2.9567	1.1474	266.63	13.09
Promedio	5.6176	5.7586	321.73	758.36
Des. Estándar	0.57905	2.595	37.008	326.45
Peor	6.8291	13.288	345.34	1128.1

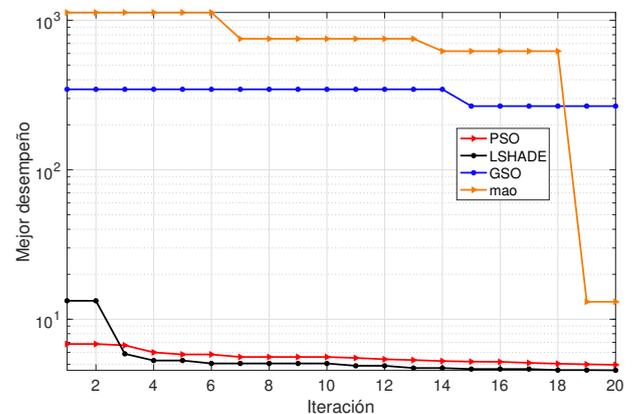


Figura 7: Respuesta función costo de los algoritmos durante el proceso de optimización.

La Tabla 2, muestra los valores: mejor, peor, promedio y la desviación estándar, obtenidos por cada uno de los algoritmos durante la optimización de los controladores PI. Estos valores estadísticos permiten observar el desempeño de cada algoritmo metaheurístico.

La figura 7 muestra el mejor valor obtenido por cada uno de los algoritmos durante la búsqueda de los parámetros del controlador PI. Se puede observar que el mejor desempeño lo obtiene el algoritmo LSHADE. Los algoritmos GSO y *mao* son los que se mantienen alejados del valor alcanzado por el algoritmo LSHADE y también presentan la mayor desviación estándar. La menor desviación estándar la presenta el PSO, sin embargo, terminadas las 20 iteraciones, no logra minimizar el valor de la función costo a un valor menor que el alcanzado por el LSHADE.

La figura 9 muestra la distribución de los datos obtenidos por los algoritmos, así como los valores atípicos durante la optimización del controlador PI.

Los parámetros encontrados por el el algoritmo LSHADE son los mostrados en la Tabla 3.

Tabla 3: Parámetros del controlador PI obtenidos mediante el LSHADE

Parámetro	K_{p1}	K_{i1}	K_{p2}	K_{i2}	K_{p3}	K_{i3}
Valor	0.4289	299.99	141.62	299.49	2.0500	299.77

La figura 8 muestra el desempeño de la velocidad del MIP cuando el controlador PI optimizado se implementa. Puede observarse que el controlador lleva al motor a la velocidad deseada, así mismo se muestra el efecto del Par que actúa como una perturbación al sistema. El par aplicado, T_L , se muestra en la figura 10, así como la respuesta del motor T_e .

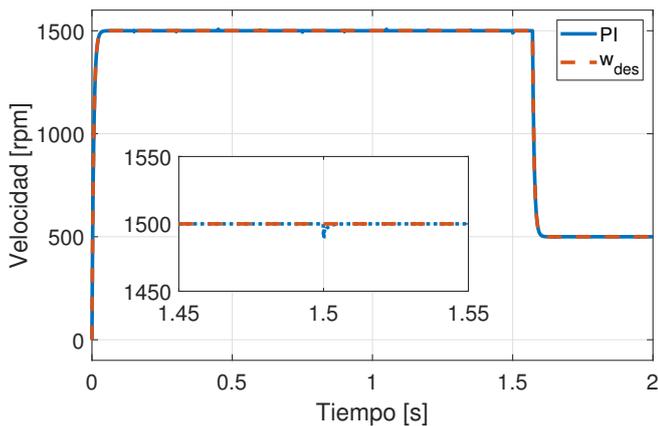


Figura 8: Velocidad del MIP con perturbaciones en el cambio de referencia y el par de carga.

El error de seguimiento entre la velocidad deseada y el desempeño del MIP se muestra en la figura 11. De la misma forma que la gráfica de la velocidad, presenta el efecto del par inducido y la reducción de la velocidad deseada.

Las señales de control V_d y V_q se muestran en la figura 12. La señal V_q presenta sobreimpulsos cuando la señal de control demanda un cambio en la velocidad.

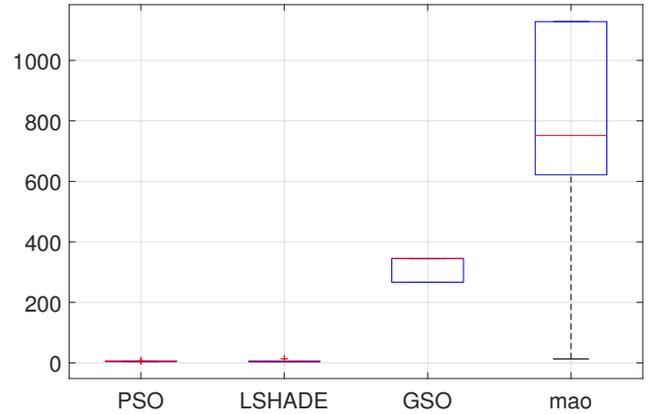


Figura 9: Representación gráfica de la estadística de los algoritmos durante el proceso de optimización.

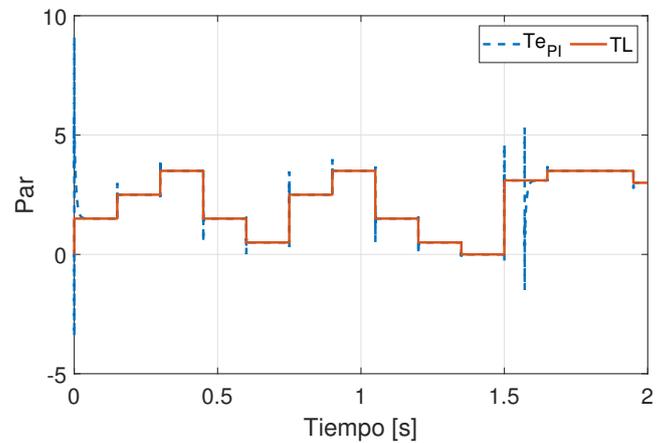


Figura 10: Respuesta del par electromagnético a los cambios en el par de carga.

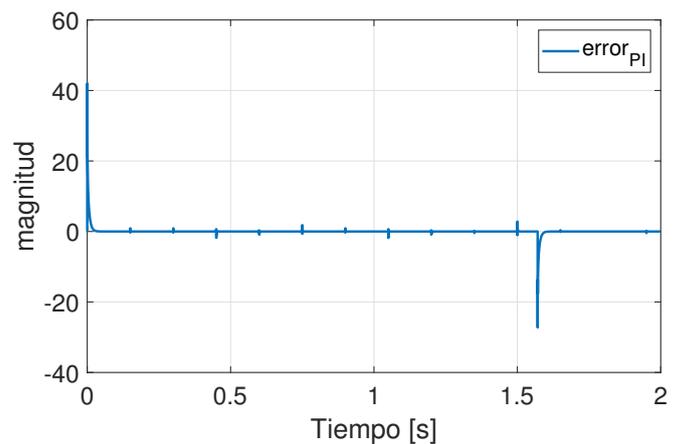


Figura 11: Error entre la velocidad deseada y a real.

6. Conclusiones

En este trabajo se ha abordado el problema de sintonización de los parámetros PI para el control de velocidad de un motor de imanes permanentes. Se describe el modelo dinámico no lineal

del MIP, posteriormente se compararon los algoritmos meta-heurísticos PSO, GSO, LSHADE y MAO, encontrando parámetros óptimos para seleccionar el algoritmo más adecuado.

Los resultados de la simulación demuestran que el algoritmo LSHADE puede optimizar las ganancias de un controlador de velocidad para un MIP. Las ganancias obtenidas mediante el algoritmo LSHADE permiten que el motor siga una velocidad deseada incluso ante cambios bruscos en el par de carga, con un valor *RMSE* de 1.1474.

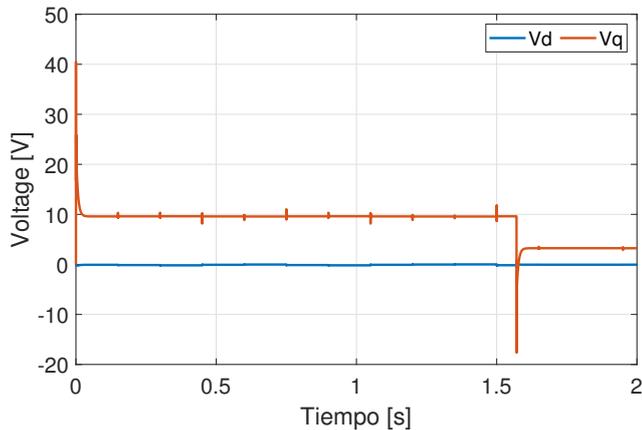


Figura 12: Voltajes en los ejes de cuadratura d-q.

Referencias

- Brahim, B., Kobayashi, M., Al Ali, M., Khatir, T., y Elmeliani, M. E. A. E. (2024). Metaheuristic optimization algorithms: an overview. *HCMCOU Journal of Science—Advances in Computational Structures*.
- Cai, W., Wu, X., Zhou, M., Liang, Y., y Wang, Y. (2021). Review and development of electric motor systems and electric powertrains for new energy vehicles. *Automotive Innovation*, 4(3)–22.
- Choi, H., Kim, E., Yu, D., Jung, J., y Kim, T. (2017). Precise pi speed control of permanent magnet synchronous motor with a simple learning feedforward compensation. *Electrical Engineering*, 99:133–139.
- Choi, H. H., Yun, H. M., y Kim, Y. (2013). Implementation of evolutionary fuzzy pid speed controller for pm synchronous motor. *IEEE Transactions on Industrial Informatics*, 11(2):540–547.
- Deng, L. y Liu, S. (2024). Metaheuristics exposed: Unmasking the design pitfalls of arithmetic optimization algorithm in benchmarking. *Applied Soft Computing*, 160:111696.
- Dorigo, M. y Stützle, T. (2019). *Ant colony optimization: overview and recent advances*. Springer.
- Gao, P., Zhang, G., Ouyang, H., y Mei, L. (2020). An adaptive super twisting nonlinear fractional order pid sliding mode control of permanent magnet synchronous motor speed regulation system based on extended state observer. *IEEE access*, 8:53498–53510.
- Kennedy, J. y Eberhart, R. (1995). Particle swarm optimization. En *Proceedings of ICNN'95-international conference on neural networks*, volumen 4, pp. 1942–1948. ieece.
- Mohamed, A. W., Hadi, A. A., y Jambi, K. M. (2019). Novel mutation strategy for enhancing SHADE and LSHADE algorithms for global numerical optimization. *Swarm and Evolutionary Computation*, 50.
- Noroozi, M., Mohammadi, H., Efatinasab, E., Lashgari, A., Eslami, M., y Khan, B. (2022). Golden search optimization algorithm. *IEEE Access*, 10:37515–37532.
- Nouaoui, T., Dendouga, A., y Bendaikha, A. (2024). Speed control of pmsm using a fuzzy logic controller with deformed mfs tuned by a novel hybrid meta-heuristic algorithm. *Electrical Engineering*, pp. 1–13.
- Özçiflikçi, O. E., Koç, M., Bahçeci, S., y Emiroğlu, S. (2024). Overview of pmsm control strategies in electric vehicles: a review. *International Journal of Dynamics and Control*, 12(6):2093–2107.
- Shabir, S. y Singla, R. (2020). A Comparative Study of Particle Swarm Optimization and Genetic Algorithm. *International Journal of Electrical Engineering*, 9(2):215–223.
- Soundirarajan, N. y Srinivasan, K. (2021). Performance evaluation of ant lion optimizer-based pid controller for speed control of pmsm. *Journal of Testing and Evaluation*, 49(2):1104–1118.
- Tanabe, R. y Fukunaga, A. (2013). Success-history based parameter adaptation for Differential Evolution. *2013 IEEE Congress on Evolutionary Computation, CEC 2013*, (3):71–78.
- Tanabe, R. y Fukunaga, A. S. (2014). Improving the search performance of SHADE using linear population size reduction. *Proceedings of the 2014 IEEE Congress on Evolutionary Computation, CEC 2014*, pp. 1658–1665.
- Templos-Santos, J. L., Aguilar-Mejía, O., Peralta-Sanchez, E., y Sosa-Cortez, R. (2019). Parameter tuning of pi control for speed regulation of a pmsm using bio-inspired algorithms. *Algorithms*, 12(3):54.
- Villuendas-Rey, Y., Velázquez-Rodríguez, J. L., Alanis-Tamez, M. D., Moreno-Ibarra, M.-A., y Yáñez-Márquez, C. (2021). Mexican axolotl optimization: A novel bioinspired heuristic. *Mathematics*, 9(7):781.
- Wang, H., Xu, S., y Hu, H. (2023). Pid controller for pmsm speed control based on improved quantum genetic algorithm optimization. *IEEE Access*, 11:61091–61102.
- Wolpert, D. H. y Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82.
- Wu, C., Zhang, K., y Zhang, X. (2024). Fpga-based speed control strategy of pmsm using improved beetle antennae search algorithm. *Energies*, 17(8):1870.
- Xia, C., Li, S., Shi, Y., Zhang, X., Sun, Z., y Yin, W. (2019). A non-smooth composite control approach for direct torque control of permanent magnet synchronous machines. *IEEE Access*, 7:45313–45321.
- Yazdani, A. e Iravani, R. (2010). *Voltage-sourced converters in power systems: modeling, control, and applications*. John Wiley and Sons.
- Zhao, J., Liu, X., Wang, S., y Zheng, L. (2023). Review of design and control optimization of axial flux pmsm in renewable-energy applications. *Chinese Journal of Mechanical Engineering*, 36(1):45.