

Comparación de algoritmos heurísticos para la síntesis óptima de un mecanismo de cuatro barras

Portilla-Flores Edgar Alfredo¹, Calva Yáñez Maria Bárbara¹, Villarreal-Cervantes Miguel Gabriel¹, Niño Suárez Paola Andrea², Sepúlveda-Cervantes Gabriel¹

¹INSTITUTO POLITÉCNICO NACIONAL

Centro de Innovación y Desarrollo Tecnológico en Cómputo (CIDETEC-IPN), México D.F., México
aportilla@ipn.mx, b_calva@hotmail.com, mvillarreal@ipn.mx, gsepulvedac@ipn.mx

²INSTITUTO POLITÉCNICO NACIONAL

Escuela Superior de Ingeniería Mecánica y Eléctrica Azcapotzalco (ESIME-AZC), México D.F., México,
pninos@ipn.mx

Resumen

En el presente artículo se lleva a cabo la comparación del funcionamiento de dos algoritmos heurísticos, Evolución Diferencial (ED) y Colonia Artificial de Abejas Modificado (CAAM), en la obtención de soluciones para la síntesis óptima de un mecanismo de cuatro barras. Dicha síntesis se realiza al proponer un problema de optimización numérica que integra la cinemática y dinámica del sistema así como un conjunto de restricciones cinemáticas y dinámicas. Un conjunto de Simulaciones son llevadas a cabo para cada uno de los algoritmos antes mencionados y los resultados son comparados desde el punto de vista de factibilidad de implementación así como del punto de vista de implementación mecánica. A partir de los resultados se puede concluir que este tipo de algoritmos son una forma alterna para el diseño en ingeniería.

Palabras clave: Algoritmos heurísticos, Evolución Diferencial, Colonia Artificial de Abejas Modificado, Mecanismo de cuatro barras, Optimización dinámica.

1. Introducción

Actualmente, uno de los problemas que se presenta al diseñar prototipos, es la complejidad de los mismos debido a la interacción y la sinergia entre las diferentes etapas que lo integran. Algunas de las etapas más representativas en un prototipo son: mecánica, eléctrica, electrónica, computacional y de control; por mencionar algunas.

La línea que delimita las áreas antes mencionadas no es muy clara y por lo tanto utilizar

técnicas de diseño tradicionales, donde los requerimientos mecánicos se cumplen sin tomar en cuenta aspectos de los actuadores, del sistema de energía, del control o de la lógica computacional, produce que la funcionalidad del prototipo sea limitada. Los sistemas actuales deben ser diseñados conceptualizándolos como sistemas mecatrónicos completos, es decir, se deben tomar en cuenta todos los subsistemas que los integran. Este concepto se conoce como Diseño Integrado [1], el cual, como su nombre lo dice, integra los componentes de estructura física y los de procesamiento de la información en la etapa de análisis y obtención de soluciones. Existen varios enfoques que abordan el diseño integrado de sistemas. Sin embargo, uno de los enfoques que más se utilizan es aquel en donde el problema de diseño de sistemas se resuelve mediante la propuesta de un problema de optimización numérica, el cual se resuelve mediante un método de optimización [2],[3],[4].

Debido a la complejidad de los problemas de optimización cuando se utiliza el enfoque de diseño integrado, un método de solución alternativo es utilizar algoritmos heurísticos bio-inspirados. Este tipo de algoritmos se pueden dividir en dos tipos [5]: Algoritmos evolutivos (AEs) y Algoritmos de Inteligencia de Enjambre o Cúmulo (AIEs). Los AEs emulan la evolución de las especies basados en la supervivencia del mejor. Los AIEs emulan el comportamiento social de búsqueda de comida algunas especies de la naturaleza. En el presente trabajo se utiliza un AE denominado Evolución Diferencial (DE) [6] y un AIE denominados Colonia de Abejas Modificado [7], para obtener el diseño óptimo de un mecanismo de cuatro barras. Se lleva a cabo una comparación de los resultados que se

obtienen de cada uno de ellos para analizar la factibilidad de su aplicación en el contexto de diseño integrado.

El presente artículo se organiza como sigue: una explicación breve de cada uno de los algoritmos utilizados en la obtención de la síntesis óptima del mecanismo de cuatro barras, se presenta en la sección 2. El caso de estudio para el mecanismo de cuatro barras se presenta en la sección 3. Los resultados obtenidos y la discusión de los mismos se exponen en la sección 4. Finalmente, las conclusiones y trabajos futuros se incluyen en la sección 5.

2. Algoritmos heurísticos

Una características de los algoritmos heurísticos bio-inspirados que los han hecho exitosos para resolver problemas del mundo real, es que estos basan su funcionamiento en un conjunto de puntos (vectores de variables de diseño) denominado “población”, el cual es generado de forma aleatoria en toda la región factible que se establece por medio de las restricciones de diseño. Lo anterior, evita la sensibilidad al punto inicial de búsqueda de la solución, es decir; aumenta la exploración en la región factible del problema, para localizar la mejor solución.

2.1 Evolución Diferencial

El algoritmo de ED propuesto por K.V. Price en [6], tiene un funcionamiento relativamente sencillo: utiliza un operador especial de mutación que realiza una combinación lineal con las diferencias entre un par de vectores (r_1 y r_2) y un vector llamado vector base (r_3), los cuales son seleccionados de forma aleatoria. Del resultado de aplicar el operador de mutación, se genera un vector denominado vector de mutación, el cual se recombinará con otro vector de la población denominado padre (el cual se encuentra sujeto a ser reemplazado) para producir un vector hijo. Con los vectores antes obtenidos (padre e hijo) se lleva a cabo un torneo determinista para ver cuál es el mejor y así permanecer en la población de la siguiente generación. Este proceso se lleva a cabo para cada uno de los individuos de la población hasta generar una nueva población con los mejores individuos. Lo anterior es ejecutado un determinado número de iteraciones denominadas generaciones. En la Fig. 1, se muestra el algoritmo estándar de ED denominada como “DE/rand/1/bin” [7].

```

1  BEGIN
2   $G = 0$ 
3  Crear aleatoriamente la población  $\vec{p}_G, \forall i = 1, \dots, NP$ 
6  Evaluar  $F(\vec{p}_G), g(\vec{p}_G), h(\vec{p}_G), \forall i = 1, \dots, NP$ 
7  Do
8  For  $i = 1$  to  $NP$  Do
9  Seleccionar aleatoriamente  $\{r_1 \neq r_2 \neq r_3 \neq i\} \in \vec{x}_G$ .
10  $j_{rand} = \text{randint}(1, D)$ 
11 For  $j = 1$  to  $D$  Do
12 If  $(\text{rand}_j[0, 1] < CR \text{ or } j = j_{rand})$  Then
13  $u_{j,G+1}^i = p_{j,G}^{r_1} + F(p_{j,G}^{r_2} - p_{j,G}^{r_3})$ 
14 Else
15  $u_{j,G+1}^i = p_{j,G}^i$ 
16 End If
17 End For
18 Evaluar  $F(\vec{u}_{G+1}^i), g(\vec{u}_{G+1}^i), h(\vec{u}_{G+1}^i)$ 
19 If  $F(\vec{u}_{G+1}^i)$  mejor que  $F(\vec{p}_G)$  con las RMR Then
20  $\vec{p}_{G+1} = \vec{u}_{G+1}$ 
21 Else
22  $\vec{p}_{G+1} = \vec{p}_G$ 
23 End If
24 End For
25  $G = G + 1$ 
26 While  $(G \leq G_{Max})$ 
27 END
    
```

Fig. 1. Algoritmo estándar de Evolución Diferencial DE/rand/1/bin con manejo de restricciones [7].

Donde la cantidad máxima de generaciones G_{MAX} , el tamaño de la población NP y los parámetros de mutación y cruza del algoritmo F y CR son definidos por el usuario. El parámetro $CR \in [0,1]$ controla la influencia del vector de mutación en la generación del vector hijo, valores cercanos a la unidad implican que el vector hijo será muy parecido al vector de mutación, por el contrario; valores cercanos a cero implican que el vector hijo será muy parecido al vector padre. El parámetro $F \in (0,1]$ escala las diferencias entre vectores para calcular el vector de mutación. La función $\text{randint}(\text{min}, \text{max})$ regresa un número entero entre min y max , la función $\text{rand}[0,1)$ es una función que regresa un número real entre 0 y 1. Ambas funciones están basadas en una distribución uniforme de números aleatorios. Finalmente, debido a que el algoritmo original de ED es para optimización sin restricciones, se integró al mismo las reglas de factibilidad propuestas por Deb, incorporadas a ED originalmente en [7]. Las reglas de manejo de restricciones (RMR) antes mencionadas, permiten dirigir la búsqueda dentro del espacio factible de las soluciones.

2.2 Colonia Artificial de Abejas Modificado

El proceso de búsqueda de néctar en las flores por parte de abejas melíferas ha sido visto como un proceso de optimización. La forma en que este tipo de insectos sociales logran esfuerzos en zonas con altas cantidades de alimento se ha modelado como una heurística para optimización. En el presente trabajo se

utiliza el algoritmo propuesto por Mezura et. al [8], el cual se aplica para la solución de problemas de optimización numérica con restricciones.

```

1 BEGIN
2 Initialize the set of food sources  $x_i^0, i = 1, \dots, SN$ 
3 Evaluate each  $x_i^0, i = 1, \dots, SN$ 
4  $g = 1$ 
5 IF There are equality constraints
6   Initialize  $\epsilon(g)$ 
7 END IF
8 REPEAT
9   IF There are equality constraints
10    Evaluate each  $x_i^0, i = 1, \dots, SN$  with  $\epsilon(g)$ 
11  END IF
12  FOR  $i = 1$  TO  $SN$ 
13    Generate  $v_i^g$  with  $x_i^{g-1}$  by using Eq. [1]
14    Evaluate  $v_i^g$ 
15    IF  $v_i^g$  is better than  $x_i^{g-1}$  (based on feasibility criteria)
16       $x_i^g = v_i^g$ 
17    ELSE
18       $x_i^g = x_i^{g-1}$ 
19    END FOR
20  FOR  $i = 1$  TO  $SN$ 
21    Select food source  $x_i^g$  based on binary tournament selection
22    Generate  $v_i^g$  with  $x_i^g$  by using Eq. [1]
23    Evaluate  $v_i^g$ 
24    IF  $v_i^g$  is better than  $x_i^g$  (based on feasibility criteria)
25       $x_i^g = v_i^g$ 
26    END IF
27  END FOR
28  Apply the smart flight by the scout bees (Eq. [4]) for those
29  solutions whose limit to be improved has been reached
30  Keep the best solution so far
31   $g = g + 1$ 
32  IF There are equality constraints
33    Update  $\epsilon(g)$  by using Eq. FF
34  END IF
35 UNTIL  $g = MCN$ 
36 END
    
```

Fig. 2. Algoritmo de Colonia Artificial de Abejas Modificado [8].

Los comportamientos en los que se basa el algoritmo antes mencionado son: El reclutamiento de abejas en una fuente de alimento y el abandono de una fuente. En el algoritmo de abejas, la colonia artificial de abejas se compone de 3 grupos de abejas: empleadas, observadoras y exploradoras. El número de abejas empleadas es usualmente igual al número de fuentes de alimento y se asignará una abeja empleada a cada una de las mismas. Al llegar a dicha fuente la abeja calculará una nueva solución (volará hacia otra fuente de alimento cercana) a partir de esta y conservará la mejor solución. El número de abejas observadoras es también usualmente igual al de abejas empleadas y se asignará a una fuente de alimento con base en la aptitud de éstas, al igual que las abejas empleadas, calculará una nueva solución a partir de una fuente de alimento. Cuando una fuente no mejora después de un cierto número de iteraciones se abandona, siendo remplazada por aquella encontrada por una abeja exploradora, lo cual implica calcular una nueva solución aleatoriamente. La Fig. 2 muestra el algoritmo de Colonia de abejas modificado [8].

Una de las ventajas de este algoritmo son la cantidad de parámetros que se necesitan para ajustar su funcionamiento. El número de soluciones o

fuentes de alimento SN , el número total de ciclos (iteraciones) que se ejecutará el algoritmo MCN y el número de ciclos que se conservará una solución sin mejora antes de ser reemplazada por una nueva solución generada por una abeja exploradora.

En el algoritmo de Colonia de abejas modificado, las abejas son conceptualizadas como operadores de variación, debido a que cuando una de ellas llega a una fuente de alimento, se calcula una nueva solución candidata v_i^g utilizando la siguiente ecuación:

$$v_i^g = \begin{cases} x_{i,j}^g + \phi_j(x_{i,j}^g - x_{k,j}^g) & \text{si } \text{rand}(0,1) < MR \\ x_{i,j}^g & \text{en otro caso} \end{cases} \quad (1)$$

Donde x_i^g representa la solución en la que la abeja se encuentra en ese momento, x_k^g es una fuente de alimento elegida aleatoriamente y distinta de la primera, g es el número actual del ciclo del programa y ϕ es un número real aleatorio en el intervalo $[-1,1]$. Finalmente, el mecanismo de recombinación que utiliza este algoritmo es definido por el usuario como $0 \leq MR \leq 1$. Las reglas de selección que se utilizan en el torneo de selección son las utilizadas en el algoritmo de Evolución Diferencial.

Por otro lado, en este algoritmo se utiliza una tolerancia dinámica para restricciones de igualdad, la cual se establece como:

$$\epsilon(g+1) = \frac{\epsilon(g)}{dec} \quad (2)$$

$$dec = e^{\left(\frac{\ln(\epsilon_0) - \ln(\epsilon_f)}{MCN}\right)} \quad (3)$$

Donde dec es el valor de la razón de disminución de cada iteración ($dec > 1$), ϵ_0 es el valor inicial de la tolerancia y ϵ_f el valor final de la misma.

Finalmente, un operador que se incluye en el algoritmo de colonia artificial de abejas modificado para elegir una nueva fuente debido a que una determinada fuente no mejora después de un cierto número de iteraciones, es el operador inteligente de vuelo $v_{i,j}^g$, donde x_B^g es la mejor solución de las existentes. Dicho operador se define como:

$$v_{i,j}^g = x_{i,j}^g + \phi \cdot (x_{k,j}^g - x_{i,j}^g) + (1 + \phi) \cdot (x_{B,j}^g - x_{i,j}^g) \quad (4)$$

3. Caso de estudio: problema de diseño óptimo

El caso de estudio del presente trabajo es relacionado con el diseño óptimo de un mecanismo de cuatro barras el cual tiene acoplado como sistema de impulsión un motor de C.D. en el eje de la manivela. Las Figs. 3 y 4 muestran los diagramas esquemáticos de los sistemas antes mencionados.

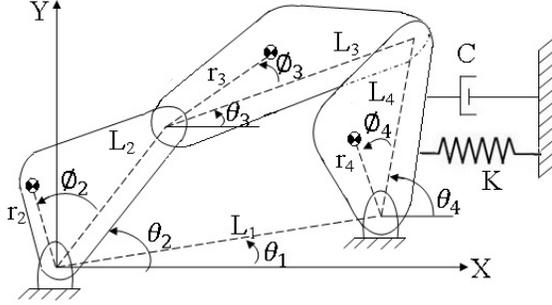


Fig. 3. Diagrama esquemático de un mecanismo de cuatro barras [9].

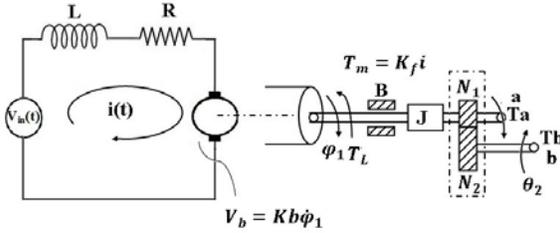


Fig. 4. Diagrama esquemático de un motor de C.D. [9].

Sin pérdida de generalidad, la cinemática del mecanismo de cuatro barras se puede expresar mediante las siguientes ecuaciones:

$$\dot{\theta}_i = \gamma_i \dot{\theta}_2 \quad (5)$$

$$v_{ix} = \alpha_i \dot{\theta}_2 \quad (6)$$

$$v_{iy} = \beta_i \dot{\theta}_2 \quad (7)$$

donde:

$$\alpha_2 = -r_2 \sin(\theta_2 + \phi_2) \quad (8)$$

$$\alpha_3 = -L_2 \sin \theta_2 - r_3 \gamma_3 \sin(\theta_3 + \phi_3) \quad (9)$$

$$\alpha_4 = -r_4 \gamma_4 \sin(\theta_4 + \phi_4) \quad (10)$$

$$\beta_2 = r_2 \cos(\theta_2 + \phi_2) \quad (11)$$

$$\beta_3 = L_2 \cos \theta_2 + r_3 \gamma_3 \cos(\theta_3 + \phi_3) \quad (12)$$

$$\beta_4 = r_4 \gamma_4 \cos(\theta_4 + \phi_4) \quad (13)$$

$$\gamma_2 = 1 \quad (14)$$

$$\gamma_3 = \frac{L_2 \sin(\theta_4 - \theta_2)}{L_3 \sin(\theta_3 - \theta_4)} \quad (15)$$

$$\gamma_4 = \frac{L_2 \sin(\theta_3 - \theta_2)}{L_3 \sin(\theta_3 - \theta_4)} \quad (16)$$

$$\theta_3 = 2 \arctan \left[\frac{-B_1 \pm \sqrt{B_1^2 + A_1^2 - C_1^2}}{C_1 - A_1} \right] \quad (17)$$

$$\theta_4 = 2 \arctan \left[\frac{-E_1 \pm \sqrt{D_1^2 + E_1^2 - F_1^2}}{F_1 - D_1} \right] \quad (18)$$

$$A_1 = 2L_3(L_2 \cos \theta_2 - L_1 \cos \theta_1) \quad (19)$$

$$B_1 = 2L_3(L_2 \sin \theta_2 - L_1 \sin \theta_1) \quad (20)$$

$$C_1 = L_1^2 + L_2^2 + L_3^2 - L_4^2 - 2L_1L_2 \cos(\theta_1 - \theta_2) \quad (21)$$

$$D_1 = 2L_4(L_1 \cos \theta_1 - L_2 \cos \theta_2) \quad (22)$$

$$E_1 = 2L_4(L_1 \sin \theta_1 - L_2 \sin \theta_2) \quad (23)$$

$$F_1 = L_1^2 + L_2^2 + L_4^2 - L_3^2 - 2L_1L_2 \cos(\theta_1 - \theta_2) \quad (24)$$

La ecuación de movimiento del sistema utilizando θ_2 como coordenada generalizada, se establece mediante:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_2} \right) - \frac{\partial L}{\partial \theta_2} + \frac{\partial D}{\partial \dot{\theta}_2} = T \quad (25)$$

donde:

$$L = K - P \quad (26)$$

$$K = \sum_{i=2}^4 \left(\frac{1}{2} m_i (v_{ix}^2 + v_{iy}^2) + \frac{1}{2} J_i \dot{\theta}_i^2 \right) = \frac{1}{2} A(\theta_2) \dot{\theta}_2^2 \quad (27)$$

$$P = \frac{1}{2} k(\theta_4 - \theta_{4,0})^2 \quad (28)$$

$$D = \frac{1}{2} C \dot{\theta}_4^2 \quad (29)$$

$$A(\theta_2) = \sum_{i=2}^4 (m_i (\alpha_i^2 + \beta_i^2) + \gamma_i^2 J_i) \quad (30)$$

Desarrollando las derivadas indicadas en (25), la ecuación de movimiento se expresa en (31).

$$T = A(\theta_2) \ddot{\theta}_2 + \frac{1}{2} \frac{dA(\theta_2)}{d\theta_2} \dot{\theta}_2^2 + k\gamma_4(\theta_4 - \theta_{4,0}) + C\gamma_4^2 \dot{\theta}_2 \quad (31)$$

donde:

$$A(\theta_2) = C_0 + C_1 \gamma_3^2 + C_2 \gamma_4^2 + C_3 \gamma_3 \cos(\theta_2 - \theta_3 - \phi_3) \quad (32)$$

$$\frac{dA(\theta_2)}{d\theta_2} = 2C_1\gamma_3 \frac{d\gamma_3}{d\theta_2} + 2C_2\gamma_4 \frac{d\gamma_4}{d\theta_2} + C_3 \frac{d\gamma_3}{d\theta_2} \cos(\theta_2 - \theta_3 - \phi_3) \quad (33)$$

$$-C_3\gamma_3(1-\gamma_3)\sin(\theta_2 - \theta_3 - \phi_3) \quad (34)$$

$$C_0 = J_2 + m_2r_2^2 + m_3L_2^2 \quad (35)$$

$$C_1 = J_3 + m_3r_3^2 \quad (36)$$

$$C_2 = J_4 + m_4r_4^2 \quad (37)$$

$$C_3 = 2m_3L_2r_3 \quad (38)$$

$$\frac{d\gamma_3}{d\theta_2} = \frac{L_2(D_1 + D_2)}{L_3 \sin^2(\theta_3 - \theta_4)} \quad (39)$$

$$\frac{d\gamma_4}{d\theta_2} = \frac{L_2(D_3 + D_4)}{L_4 \sin^2(\theta_3 - \theta_4)} \quad (40)$$

$$D_1 = (\gamma_4 - 1)\sin(\theta_3 - \theta_4)\cos(\theta_4 - \theta_2) \quad (41)$$

$$D_2 = (\gamma_4 - \gamma_3)\sin(\theta_4 - \theta_2)\cos(\theta_3 - \theta_4) \quad (42)$$

$$D_3 = (\gamma_3 - 1)\sin(\theta_3 - \theta_4)\cos(\theta_3 - \theta_2) \quad (43)$$

$$D_4 = (\gamma_4 - \gamma_3)\sin(\theta_3 - \theta_2)\cos(\theta_3 - \theta_4) \quad (44)$$

Para el motor de C.D. el modelo dinámico se expresa mediante [10]:

$$L \frac{di(t)}{dt} + Ri(t) = V_{in}(t) - K_b \dot{\phi}_1 \quad (45)$$

$$T_m - B\dot{\phi}_1 - T_a - T_L = J\ddot{\phi}_1 \quad (46)$$

Para el sistema de engranajes de la salida del motor de C.D., se cumple que:

$$\frac{T_b}{T_a} = \frac{\dot{\phi}_1}{\dot{\theta}_2} = \frac{\ell_2}{\ell_1} = \frac{N_2}{N_1} = n \quad (47)$$

Donde ℓ_1 , ℓ_2 , N_1 y N_2 representan los radios y el número de dientes de los engranes del sistema.

Utilizando (46), el modelo dinámico del motor de C.D. se expresa mediante:

$$T_b = nK_f i(t) - nT_L - n^2 B \dot{\theta}_2 - n^2 J \ddot{\theta}_2 \quad (48)$$

$$L \frac{di(t)}{dt} + Ri(t) = V_{in}(t) - nK_b \dot{\theta}_2 \quad (49)$$

Finalmente, la dinámica acoplada de los dos subsistemas se obtiene a partir de (31), (47) y (48). Proponiendo el vector de variables de estado $\bar{x} = [x_1, x_2, x_3]^T = [\theta_2, \dot{\theta}_2, i(t)]^T$, la representación del sistema acoplado en variables de estado se establece como:

$$\dot{x} = f(\bar{x}, \bar{p}, t) = \begin{bmatrix} x_2 \\ A_0[A_1 x_2^2 + A_2 x_2 + nK_f x_3 + A_3] \\ \frac{1}{L}(u(t) - nK_b x_2 - R x_3) \end{bmatrix} \quad (50)$$

donde:

$$A_0 = \frac{1}{A(x_1) + n^2 J_1} \quad (51)$$

$$A_1 = -\frac{1}{2} \frac{dA(x_1)}{dx_1} \quad (52)$$

$$A_2 = -(C\gamma_4^2 + n^2 B) \quad (53)$$

$$A_3 = -k\gamma_4(\theta_4 - \theta_{4,0}) - nT_L \quad (54)$$

Con el propósito de llevar a cabo la síntesis óptima del mecanismo de la Fig. 1, se eligió un conjunto de valores que determina de forma única al mecanismo de cuatro barras así como una función para evaluar el comportamiento del sistema. Una explicación detallada de la descripción de la función de desempeño del sistema, las restricciones de diseño y las cotas o restricciones geométricas se puede consultar en [11]. Tomando en cuenta el trabajo antes mencionado, el conjunto de valores para diseño óptimo, está integrado por las longitudes de los eslabones del mecanismo, así como del ángulo con respecto del eje horizontal del eslabón de referencia, es decir:

$$\bar{p} = [L_1, L_2, L_3, L_4, \theta_1]^T = [p_1, p_2, p_3, p_4, p_5]^T \quad (55)$$

Por lo tanto, la síntesis óptima del mecanismo de cuatro barras se lleva a cabo al proponer dicha síntesis como un problema de optimización dinámica mono-objetivo (PODM). La solución óptima del PODM es el vector de variables de diseño (54) el cual minimiza la función de desempeño (55) sujeto al modelo dinámico del sistema en variables de estado (56), así como a las restricciones de diseño de desigualdad e igualdad (57) a (61). Es importante mencionar que la restricción de diseño de desigualdad (60) es una restricción dinámica la cual se evalúa utilizando el perfil de las variables de estado del modelo dinámico con las condiciones iniciales (62) y las cotas (63).

$$\text{Min } \Phi(\vec{p}) = (\mu_{\max} - 90)^2 + (\mu_{\min} - 90)^2 \quad (55)$$

Sujeto a:

$$\dot{x} = f(\vec{x}, \vec{p}, t) \quad (56)$$

$$g_1(\vec{p}) = p_2 + p_3 - p_1 - p_4 \leq 0 \quad (57)$$

$$g_2(\vec{p}) = p_1 - p_3 \leq 0 \quad (58)$$

$$g_3(\vec{p}) = p_4 - p_3 \leq 0 \quad (59)$$

$$g_4(\vec{p}, t) = \frac{\pi}{4} - \mu(\vec{p}, t) \leq 0 \quad (60)$$

$$h_1(\vec{p}) = \pi - \theta_{4\max} - \theta_{4\min} = 0 \quad (61)$$

$$\vec{x}(0) = x_0 \quad (62)$$

$$0.05 \leq p_i \leq 0.5 \quad i = 1, \dots, 4 \quad \text{and} \quad -\frac{\pi}{4} \leq p_5 \leq \frac{\pi}{4} \quad (63)$$

Donde las unidades de los eslabones del mecanismo de cuatro barras (p_1 a p_4) son en [m] y el ángulo del eslabón de referencia (p_5) son [rad].

4. Resultados

En el presente trabajo se llevaron a cabo un conjunto de 10 simulaciones independientes para cada algoritmo, los valores de los parámetros utilizados en cada uno de ellos se resumen en la Tabla 1. Dichos parámetros son los establecidos como estándar para el funcionamiento de cada algoritmo. Es preciso mencionar que no se llevaron a cabo refinamientos en la selección de los parámetros antes mencionados, debido a que no es el objetivo principal de este trabajo.

Para la solución del modelo dinámico del sistema, el vector de variables de diseño es definido por $\vec{x}(t) = (\theta_2, \dot{\theta}_2, i(t))^T = (x_1(t), x_2(t), x_3(t))^T$ y las condiciones iniciales del mismo se establecieron con los valores $\vec{x}(0) = (0, 0, 0)^T$. Finalmente, la señal de control del motor de C.D. se fijó a un valor constante $u(t) = 30 \text{ V}$. Todos los algoritmos se codificaron en Matlab® 2008a y se ejecutaron en una computadora Laptop con 6 GB RAM Intel® procesador Core i5 @ 2.5 GHz, y Microsoft Windows® 7 OS.

Factores	ED	CAAM
Población/Fuentes	100	20
Factor de Cruza/MR	0.5-1.0	0.8
Factor de Mutación	0.3-0.9	-
Generaciones/Iteraciones	500	1000

Tabla 1. Parámetros internos de los algoritmos.

Las mejores y peores soluciones del conjunto de simulaciones numéricas llevadas a cabo para cada algoritmo se presentan en la Tabla 2. Para propósitos de presentación se muestran siete dígitos de precisión de las variables de diseño y de la función objetivo (F.O.). Como se observa en dichos resultados, las soluciones del problema de síntesis óptima son soluciones factibles de implementarse experimentalmente, ya que todas las soluciones cumplen con las restricciones impuestas en el problema de optimización.

	ED		CAAM	
	Mejor	Peor	Mejor	Peor
p_1	0.5000000	0.4999999	0.5000000	0.4999999
p_2	0.0500000	0.0500000	0.0500000	0.0500000
p_3	0.5000000	0.4999999	0.5000000	0.4999999
p_4	0.2233568	0.2233561	0.2233568	0.2233568
p_5	-0.2162201	-0.2240053	-0.2156728	-0.2172513
F.O.	0.2036933	0.2036933	0.2036933	0.2036933

Tabla 2. Comparación de resultados para el caso de estudio.

Como se puede observar, los algoritmos presentan un comportamiento estable en cuanto a la convergencia constante de los mismos hacia una solución factible por cada experimento realizado. El tiempo computacional promedio para la obtención de soluciones de cada algoritmo es de 1.06 Hrs. y 0.35 Hrs. para ED y CAMM, respectivamente. Por otro lado, la Fig. 5 muestra el comportamiento cinemático del elemento mecánico de salida y del ángulo de transmisión de un mecanismo de cuatro barras utilizando la mejor solución obtenida por cada algoritmo. En dicha gráfica se muestra el comportamiento de los elementos mencionados para varios ciclos de la manivela. Como se puede observar, el comportamiento de ambas soluciones es similar en dos aspectos fundamentales planteados en el problema de diseño: 1) el movimiento del elemento de salida (θ_4) cumple la restricción de igualdad (61), dicha restricción establece la simetría de movimiento alrededor del eje vertical, es decir alrededor del valor 1.5707 rad y 2) se observa que el valor del ángulo de transmisión a lo largo de la trayectoria completa de la manivela permanece arriba de la cota que establece la restricción (60), es decir; mayor a 0.7853 rad y lo más cercano al valor de 1.5707 rad, establecido en la función objetivo del problema de optimización.

A partir de los resultados obtenidos se puede establecer que el rendimiento computacional de ambos algoritmos es semejante en cuanto al valor de

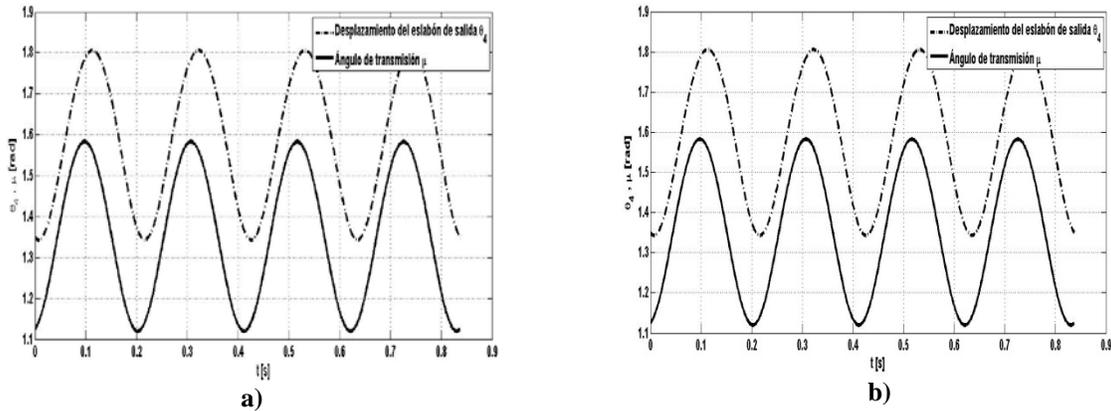


Fig. 5. Desplazamiento angular del eslabón de salida y ángulo de transmisión del mecanismo de cuatro barras para la mejor solución obtenida con las heurísticas a) CAAM, b) ED.

las soluciones obtenidas. Lo que demuestra que aun cuando la filosofía de funcionamiento de cada uno de los algoritmos es diferente y que su rendimiento depende en gran parte de la plataforma de ejecución de los mismos, la obtención de soluciones se lleva a cabo en tiempo razonable.

Sin embargo, es preciso mencionar que si el objetivo del ingeniero de diseño es llevar a cabo una exploración exhaustiva de posibles soluciones del problema de optimización, es preciso llevar a cabo una “sintonización” de los parámetros de cada algoritmo (los cuales se muestran en la Tabla 1), lo cual puede llegar a ser una tarea demorada ya que está relacionada con la naturaleza matemática de las funciones matemáticas que integran el problema de optimización planteado.

5. Conclusiones

En este trabajo se ha presentado una comparación de dos algoritmos heurísticos para la obtención de la síntesis óptima de un mecanismo de cuatro barras, mediante la proposición de un problema de optimización. Dicho problema de optimización integra el modelo dinámico y cinemático del mecanismo por diseñar, así como un conjunto de restricciones de diseño, tanto geométricas como dinámicas. La solución óptima del problema planteado es el conjunto de valores buscados para la síntesis óptima.

La convergencia del conjunto de simulaciones para ambos algoritmos, demuestran que la síntesis óptima utilizando este tipo de técnicas es una alternativa viable para esta clase de mecanismos. Lo

anterior producto del poder de cómputo y facilidad de implementación de los algoritmos utilizados lo que representan ventajas para el ingeniero de diseño. Es preciso mencionar que no se llevó a cabo una “puesta a punto” de los parámetros internos de los algoritmos, debido a que se deseaba observar el rendimiento de los mismos con los parámetros estándares que se encuentran en la literatura especializada y así comprobar su aplicabilidad y funcionamiento en este tipo de problemas.

Por otro lado, si el ingeniero de diseño no tiene un conocimiento a priori de lo que pretende obtener como producto final en su proceso de diseño, este tipo de técnicas ofrece una posibilidad aceptable a corto plazo para lograr una primera aproximación al mismo, ya que solo se necesita una descripción matemática del sistema así como conocimientos previos de las cotas máxima y mínima de sus parámetros, aspectos que también se deben desarrollar aun cuando se considerara utilizar otros enfoques de diseño.

Trabajos futuros están relacionados con la implementación de este tipo de algoritmos en la búsqueda de soluciones de otros problemas de ingeniería, así como un refinamiento en la “puesta a punto” de sus parámetros internos para mejorar el rendimiento del algoritmo.

Agradecimientos

Los autores agradecen el apoyo recibido por el Instituto Politécnico Nacional a través de la Secretaría de Investigación y Posgrado vía los proyectos SIP-20131350 y SIP-20131053, así como a

través de los programas de EDI y COFAA. La segunda autora así como el tercer autor agradecen al Consejo Nacional de Ciencia y Tecnología de México (CONACyT-México) la beca otorgada para estudios de Posgrado en el CIDETEC-IPN y el apoyo vía el proyecto 182298, respectivamente.

Referencias

- [1] R. Iserman, “*Mechatronic systems - innovative products with embedded control*”, Control Engineering Practice, No. 16, pp. 14-29, Jun 2007.
- [2] Mezura-Montes E., Portilla-Flores E.A., Coello Coello C.A., Alvarez Gallegos J., and Cruz-Villar C.A., “*An Evolutionary Approach to Solve a Novel Mechatronic Multiobjective Optimization Problem*”, Advances in Metaheuristic Methods for Hard Optimization, eds. P. Siarry and Z. Michalewicz, ISBN 978-3-540-72959-4, Berlin: Springer, pp. 329–351, 2008.
- [3] Mundo D., and Yan H., “*Kinematic Optimization of Ball-Screw Transmission Mechanism*”, Mechanism and Machine Theory, No.1, Vol. 42, pp. 34–47, 2007.
- [4] Portilla-Flores E.A., Mezura-Montes E., Alvarez Gallegos J., Coello Coello C.A., Cruz-Villar C.A., and Villareal-Cervantes M.G., “*Parametric Reconfiguration Improvement in Non-Iterative Concurrent Mechatronic Design Using an Evolutionary-Based Approach*”, Engineering Optimization of Artificial Intelligence, No. 24, Vol 5, pp. 757–771, 2011.
- [5] Engelbrecht, A.P., “*Computational Intelligence. An Introduction*”, Chichester: John Wiley & Sons, 2nd. Edition, 2007.
- [6] Price, K., Storn, R., and Lampinen, J., “*Differential Evolution: A Practical Approach to Global Optimization*”, Natural Computing Series, Berlin, Heidelberg: Springer-Verlag, 2005.
- [7] Mezura-Montes E., Coello Coello C.A. and Tun-Morales I., “*Simple feasibility rules and Differential Evolution for constrained optimization*”, In R. Monroy, G. Arroyo-Figueroa, L.E. Sucar and H. Sosa, Editors, Proceedings of the Third Mexican International Conference on Artificial Intelligence (MICAI’2004), pp. 707-716, Heidelberg, Germany, April 2004, México City, México Springer Verlag, Lecture Notes in Artificial Intelligence No. 2972.
- [8] Mezura-Montes E., Cetina-Domínguez O., “*Empirical analysis of a modified Artificial Bee Colony for constrained numerical optimization*”, Applied Mathematics and Computation, No. 218, pp. 10943-10973, 2012.
- [9] Calva-Yáñez M. B., Niño-Suárez P. A., Villareal-Cervantes M. G., Sepúlveda-Cervantes G., Portilla-Flores E. A., “*Differential Evolution for the control gain’s optimal tuning of a four-bar mechanism*”, Polibits, No. 47, pp. 65-72, January-June 2013
- [10] J. Chiasson, “*Modeling and high performance control of electrical machine*”, Wiley-IEEE Press, 2005.
- [11] Mezura-Montes E., Portilla-Flores E.A., Hernández-Ocaña B., “*Optimum synthesis of a four-bar mechanism using the modified bacterial foraging algorithm*”, International Journal of Systems Science, DOI:10.1080/00207721.2012.745023, 2013