

Revista Electrónica Nova Scientia

**Síntesis Óptima de Mecanismos utilizando un
EDA basado en la Distribución Normal
Optimal Synthesis of Mechanisms by using an
EDA based on the Normal distribution**

**Ivvan Valdez¹, Arturo Hernández Aguirre¹, Salvador
Botello¹ y Eusebio E. Hernández-Martínez²**

¹Centro de Investigación en Matemáticas, Guanajuato, Gto.

²SEPI-ESIME Ticomán, Instituto Politécnico Nacional, México D.F.

México

Sergio Ivvan Valdez Peña. E-mail: ivvan@cimat.mx

Resumen

El problema de síntesis de mecanismos consiste en determinar las dimensiones de los eslabones a fin de que éstos puedan completar una tarea, la cual es definida por el posicionamiento en un conjunto de puntos denominados puntos de precisión. En este trabajo se aborda este problema como un ejercicio de optimización, que minimiza la distancia entre la posición de un mecanismo propuesto y los puntos de precisión, tomando como variables de decisión las dimensiones del mecanismo, su sistema coordenado relativo y parámetros que determinan la velocidad del mismo y la posición inicial. La propuesta final es un algoritmo para el diseño automatizado de mecanismos que cumplan en lo mayor posible con la tarea deseada. Se utilizó un Algoritmo de Estimación de Distribución (EDA, por sus siglas en inglés), el cual aproxima el óptimo mediante muestreos subsecuentes de una distribución Normal multivariada, donde cada muestra representa una propuesta de un mecanismo. Después, se simula y se mide la diferencia entre los puntos de precisión y las posiciones del mecanismo propuesto, los mejores mecanismos se utilizan para actualizar la distribución de búsqueda en cada generación (iteración) del algoritmo hasta que converge a un punto de bajo costo de la función objetivo. Se utilizó el método propuesto para la síntesis automatizada de un mecanismo plano de 4 barras de cadena cinemática cerrada. Los resultados obtenidos son diseños óptimos aproximados, que superan a otros reportados en la literatura especializada. El método propuesto representa una alternativa eficaz para la síntesis automatizada de mecanismos. Los resultados son altamente competitivos con otros reportados en la literatura. La propuesta presenta además ventajas contra otras similares, en el sentido de que puede utilizarse para aproximar un número arbitrario de puntos de precisión, sin aumentar la dimensionalidad del problema, adicionalmente puede ser utilizado para la síntesis de otros mecanismos. Los resultados sugieren que los EDAs pueden utilizarse en este tipo de problemas de manera exitosa, por lo que a futuro se podría atacar problemas muchos más complejos, que involucren otras características deseadas como parámetros de control, o disminución de singularidades en sistemas de cadena cinemática cerrada.

Palabras clave: Síntesis de mecanismos, Técnicas evolutivas, Algoritmos de estimación de distribución, Optimización

Abstract

The problem of synthesis of mechanisms is to find the adequate dimensions in the elements in order to perform a given task. The task is defined by a set of points named: precision points. The problem is defined as minimizing the distance among the precision points and the actual position of the mechanism, the decision variables are the elements dimensions, the position and rotation of the relative coordinate system and two parameters related with the initial position and velocity. The proposal of this work is an algorithm for automatized synthesis of mechanisms. We use an Estimation of Distribution Algorithm (EDA), to approximate the optimum, by sampling and estimating a Normal multivariate probability distribution. Each sample is a candidate solution (a candidate mechanism), then we measure the difference between the precision points and the actual positions of the mechanism, the best candidate solutions are used to update the probability distribution, and the process is repeated until convergence is reached. We apply the proposed method for the synthesis of a Four-bar planar mechanism with closed chain. The obtained results are near-optimal designs that are better than others reported in the literature. The proposed method is a competitive alternative for the automatized synthesis of mechanism. We obtain high quality results when compare our approach with other reported in the literature. Additionally, the proposal presents other interesting features: the number of precision points can be increased without the need of increasing the dimension of the search space, in addition, the algorithm can be used for optimizing any mechanism. The results suggest that EDAs can successfully approach this kind of problems, hence, future work contemplate to use the same strategy for tackling more complex problems, possibly involving control parameters, or intending to design singularity-free closed kinematic chains.

Keywords: Synthesis of mechanisms, Evolutionary techniques, Estimation for distribution algorithms, optimization

Introducción:

La síntesis de mecanismos es el proceso inverso del análisis de mecanismos, consiste en conocer las dimensiones y disposición de los elementos del mecanismo con los cuales se logrará el movimiento predeterminado de un punto o de un conjunto de puntos. Esta tarea puede ser vista como un problema de optimización, desde un punto de vista físico y matemático la modelación no resulta sencilla, ya que pueden intervenir factores relativos al costo, cobertura del espacio de trabajo, singularidades, aceleraciones, velocidades y posiciones deseadas, entre otras. La modelación se puede hacer, entonces, como un problema multi-objetivo con restricciones o como mono-objetivo integrando todos los factores de forma ponderada en una sola función, intentando que el óptimo de la función ponderada cumpla con los requerimientos de diseño. Se han desarrollado diversos métodos para la síntesis de mecanismos, desde los métodos gráficos y analíticos propuestos por Hrones y Nelson (1951), Freudenstein, (1954), Sandor (1959), Hartenberg y Denavit (1964), la matriz de aproximaciones de Suh y Radcliffe (1967), los mínimos cuadrados de Levitskii y Shakvazian, (1960), hasta un modelo matemático y de simulación propuestos por Mallik et al., (1994) y Tzong-Mou y Chao-Kuang, (2005), respectivamente. En estas propuestas, se cubre en general con los requerimientos de diseño de los mecanismos, sin embargo tienen el inconveniente de que se restringe el número de puntos de posición para permitir la solución por los modelos propuestos.

En contraparte, y para abordar estos inconvenientes, se han diseñado métodos para considerar la síntesis de múltiples puntos de precisión y posición, con técnicas como la optimización no lineal o la optimización del proceso de sintetizado como el propuesto por Sancibrian et al., (2004), las redes neuronales de Walczak (2006), Vasiliu y Yannou (2001), Starosta (2008), los métodos de Monte Carlo por Kalnas y Kota (2001), o el método de desviación de control propuesto por Bulatovic y Djordjevic (2004), la mayoría de estos trabajos están relacionados con la generación de trayectorias y de posición para mecanismos de cuatro barras. Sin embargo, estos métodos requieren de una elevada capacidad de cálculo y pueden fallar al no entregar una solución global del problema de optimización, realizando sólo una búsqueda local. Para el caso específico de los métodos basados en redes neuronales, las principales desventajas son el entrenamiento, prueba y validación, que de acuerdo con el conocimiento de expertos en esta área, puede demandar una gran cantidad de información para el proceso de aprendizaje, y no ser interpretable físicamente.

Durante las últimas dos décadas, nuevas técnicas de optimización han comenzado a ser usadas para abordar los campos abiertos a la investigación dentro del área de los mecanismos y la robótica en general, los llamados algoritmos evolutivos. Estas técnicas definen una población inicial que es mejorada por aproximaciones a la función objetivo haciendo uso de la selección natural y las leyes de la genética. La principal ventaja de estos métodos es la simplicidad de su implantación, la facilidad con la que se pueden apegar a diversos modelos por las pocas exigencias del mismo, por ejemplo: no se requiere de un conocimiento a priori del espacio de búsqueda, no afecta si éste o la región factible es continuo o no, los mínimos locales u otras características que deben tomarse en cuenta con los algoritmos de optimización tradicionales no restringen la aplicabilidad de los algoritmos evolutivos.

El trabajo presentado aquí aborda la síntesis de mecanismos con técnicas de optimización evolutivas. Para tratar con este problema, se propone el uso de un algoritmo de estimación de distribución, (Larrañaga y Lozano, 2001). Los EDAs son algoritmos evolutivos que han mostrado una gran capacidad para la solución de problemas no convexos, que evitan los mínimos locales, no requieren de información de derivadas y bajo ciertas condiciones convergen al óptimo global, (Zhang y Muhlenbein, 2004). Los EDAs se derivaron del modelado probabilístico de los algoritmos genéticos (GAs, por sus siglas en inglés), y se ha mostrado que son más eficientes que los GAs cuando la correlación de las variables es un factor importante para el proceso de optimización, (Pelikan et al., 1999).

En la síntesis de mecanismos es obvio que los elementos del mismo están altamente correlacionados, es decir, los elementos del mecanismo tienen que ser variados o perturbados de manera conjunta para lograr el diseño óptimo. Debido a esto, se espera que el EDA obtenga resultados de bajo costo y sea, al menos, competitivo con otras estrategias de optimización en el estado del arte.

Con respecto a la síntesis de mecanismos, ya se han presentado algunas propuestas con algoritmos genéticos, entre ellos se encuentran las de Fang (1994), Roston y Sturges (1996), Kunjur y Krishnamurty (1997), Cabrera et al. (2002), Laribi et al. (2004), Quintero et al. (2004), Cabrera et al., (2007), y Marcelin (2010), con diferentes grados de éxito.

En este trabajo se propone un método para la síntesis de trayectoria de un mecanismo plano de cuatro eslabones utilizando un algoritmo de estimación de distribución, que utiliza como distribución de búsqueda una distribución Normal multivariada. La aplicación de este método se puede extender a cualquier otro tipo de síntesis de mecanismos, llamémosle la síntesis de generación de función o la síntesis de generación de movimiento. En esta propuesta puede aplicarse a 2 tipos de síntesis: 1) síntesis de un mecanismo que sea capaz de posicionarse en ciertos puntos del espacio de trabajo. Con este tipo de síntesis comparamos nuestra propuesta con otras similares y clásicas, a forma de validar que nuestro algoritmo es capaz de encontrar iguales o mejores resultados que otros trabajos de la literatura. 2) síntesis de un mecanismo que no solo pasa por ciertos puntos de una trayectoria dada, además, sigue la trayectoria a una velocidad prescrita, pudiendo atacar problemas con una discretización de la trayectoria en un número arbitrario de puntos.

El artículo tiene la siguiente organización: en la Sección 2 se presenta el planteamiento del problema, las preliminares sobre el mismo, y se detalla la función objetivo, en la Sección 3 presentamos el algoritmo de estimación de distribución utilizado para la síntesis óptima del mecanismo. La Sección 4 muestra los resultados obtenidos con nuestro método. La sección 5 discute los resultados, y finalmente en la Sección 6 se presentan las conclusiones generales y las perspectivas de trabajo futuro.

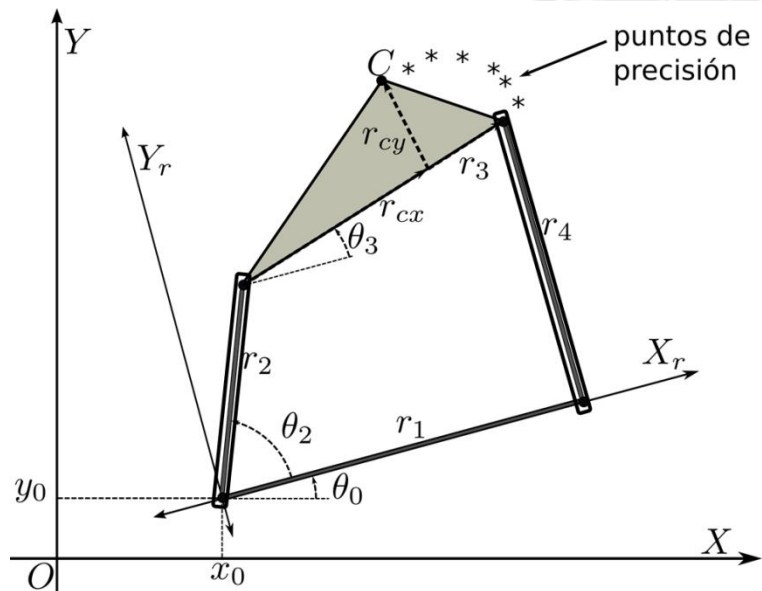


Figura 1. Esquema cinemático de un mecanismo plano de cuatro barras.

Preliminares y planteamiento del problema

El mecanismo plano tradicional de cuatro eslabones se muestra en la Figura 1, adaptada de Cabrera et al. (2002). El problema básico de la síntesis del mecanismo plano de cuatro eslabones consiste en aproximar el órgano terminal a un conjunto de puntos (denominados puntos de precisión). El órgano terminal está marcado con la letra C en la Figura 1.

El objetivo es que el órgano terminal C pase o se aproxime lo más posible cada uno de los puntos de precisión. El par motriz es el relativo a θ_2 , dado este valor, la longitud de los eslabones $r_2, r_3, r_4, r_{cx}, r_{cy}$, las coordenadas x_0, y_0 y rotación θ_0 del origen del sistema de coordenadas relativas, es posible determinar las coordenadas del punto C , denominaremos estas coordenadas como $C=[Cx_i, Cy_i]$, para $i=1,2,\dots,n$, siendo n el número de puntos de precisión que se desea aproximar.

Las coordenadas del punto C en el sistema de coordenadas relativas pueden ser determinada a partir de la suma de r_2, r_{cx} y r_{cy} :

$$\vec{C}^0 = \vec{r}_2 + \vec{r}_{cx} + \vec{r}_{cy} \quad (1)$$

Podemos escribir cada una de las componentes vectoriales como sigue:

$$C_x^0 = r_2 \cos(\theta_2) + r_{cx} \cos(\theta_3) + r_{cy} (-\sin(\theta_3)), \quad (2)$$

$$C_y^0 = r_2 \sin(\theta_2) + r_{cx} \sin(\theta_3) + r_{cy} \cos(\theta_3) \quad (3)$$

Dadas las Ecuaciones (2) y (3) es necesario conocer el ángulo θ_3 para calcular las coordenadas del punto terminal, para esto utilizamos la ecuación del lazo del mecanismo:

$$\vec{r}_2 + \vec{r}_3 - \vec{r}_4 - \vec{r}_1 = \vec{0} \quad (4)$$

También puede ser expresada en sus componentes de acuerdo al sistema de coordenadas relativas:

$$r_2 \cos(\theta_2) + r_3 \cos(\theta_3) - r_4 \cos(\theta_4) - r_1 = 0 \quad (5)$$

$$r_2 \sin(\theta_2) + r_3 \sin(\theta_3) - r_4 \sin(\theta_4) = 0 \quad (6)$$

Con las ecuaciones anteriores podemos eliminar θ_4 , resultando en:

$$k_1 \cos(\theta_3) + k_2 \cos(\theta_2) + k_3 = \cos(\theta_2 - \theta_3) \quad (7)$$

Donde $k_1 = r_1/r_2$, $k_2 = r_1/r_2$ y $k_3 = (r_4^2 - r_1^2 - r_2^2 - r_3^2)/(2r_2r_3)$. La Ecuación (7) tiene las soluciones:

$$\theta_3 = 2 \tan\left(\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}\right) \quad (8)$$

Para $a = \cos(\theta_2) - k_1 + k_2 \cos(\theta_2)$, $b = -2 \sin(\theta_2)$ y $c = k_1 + (k_2 - 1)\cos(\theta_2) + k_3$.

Las soluciones para θ_3 pueden ser: reales iguales, reales diferentes o complejas conjugadas. En el último caso simplemente significa que los eslabones no pueden conectarse. Excepto que en ese caso siempre existen dos valores diferentes de θ_3 , el que se obtiene con el signo positivo del radical se le denomina configuración cruzada, y al que se obtiene con el signo negativo se le denomina configuración abierta, o lazo de dos circuitos. Dadas las ecuaciones anteriores, se pueden calcular las coordenadas del punto C en el sistema universal coordenado de la siguiente manera:

$$C_x^0 = x_0 + C_x^0 \cos(\theta_3) - C_y^0 \sin(\theta_3), \quad (9)$$

$$C_y^0 = y_0 + C_x^0 \sin(\theta_3) + C_y^0 \cos(\theta_3). \quad (10)$$

Error de posición

El error de posición es la función objetivo más natural. Y puede ser simplemente medido como la distancia del punto calculado al punto deseado en la trayectoria. El error de posición se presenta en la Ecuación (11).

$$f(\mathbf{X}) = \sum_{i=1}^n [(C_{x_i} - C_{x_{d_i}})^2 + (C_{y_i} - C_{y_{d_i}})^2] \quad (11)$$

donde n representa el número de puntos de precisión y \mathbf{X} el vector de variables, en este caso:

$x_1 = \theta_0$, ángulo del sistema coordenado relativo.

$x_2 = \theta_2^0$, ángulo θ_2 inicial.

$x_3 = r_1$, longitud del eslabón uno.

$x_4 = r_2$, longitud del eslabón dos.

$x_5 = r_3$, longitud del eslabón tres.

$x_6 = r_4$, longitud del eslabón cuatro.

$x_7 = r_{cx}$, longitud de elemento.

$x_8 = r_{cy}$, longitud de elemento.

$x_9 = d\theta$, constante de avance en el ángulo θ .

$x_{10} = x_0$, traslación del sistema de coordenadas relativas en x.

$x_{11} = y_0$, traslación del sistema de coordenadas relativas en y.

Algunos autores, como por ejemplo (Cabrera et al., 2002), consideran un conjunto de ángulos θ_2^i como entrada, este esquema es más complejo ya que aumenta la dimensionalidad del espacio de búsqueda, puesto que cada ángulo de entrada θ_2^i es un parámetro a optimizar. En contraste en este trabajo se propone utilizar un ángulo inicial θ_2^0 , un tamaño de paso $d\theta$ y el número de puntos de precisión n . Esta información es suficiente para calcular θ_2 trayectorias con un número de puntos arbitrariamente largo (con una n arbitraria), sin incrementar la dimensionalidad del espacio de búsqueda o número de parámetros a optimizar, ya que para cualquier n , solo se requiere encontrar θ_2^0 y $d\theta$ óptimos. Adicionalmente, note que $d\theta$ puede representar el avance por unidad de tiempo de θ_2 lo que hace que sea un parámetro de velocidad. Es decir, nuestra propuesta puede ser utilizada tanto para aproximar puntos de precisión arbitrarios que describan una trayectoria o espacio de trabajo, como para aproximar una trayectoria y velocidad deseada en el órgano terminal, bajo la estrategia de proponer una velocidad constante óptima $d\theta^*$ en el par motor. Dado lo anterior, existen dos formas de calcular la función objetivo:

- 1) En donde los índices i de la Ecuación 11 pertenecen a un orden de los números naturales entre 1 y n : $i \in \{1..n\}$, siendo n el número de puntos de precisión, y el ángulo $\theta_{i2} = \theta_2^0 + (i-1) d\theta$. Por ejemplo, si $n=6$, un posible orden es [4, 3, 2, 5, 6, 1]. Lo que quiere decir, que se le requiere al mecanismo que aproxime los puntos de precisión sin ningún orden en particular. Note que el orden que da la mejor aproximación puede ser encontrado por búsqueda exhaustiva con un número de operaciones de orden n^2 .
- 2) Donde i está definido siempre por el orden particular: [1, 2, 3,..., n]. Lo que quiere decir que se aproxima una trayectoria y velocidad definida por la secuencia de ángulos $\theta_{i2} = \theta_2^0 + (i-1) d\theta$, incrementando i cada que transcurre una unidad de tiempo.

Restricciones

En la siguiente sección se describe a detalle el manejo de restricciones por parte del algoritmo de estimación de distribución, en esta subsección solo definiremos las restricciones consideradas:

- 1) Límites de las variables. $-0.5\pi \leq \theta_0 \leq 0.5\pi$, $0 \leq \theta_2^0 \leq 2\pi$, $0 \leq r_1, r_2, r_3, r_4, r_{cx}, r_{cy} \leq 2C^*$, donde C^* es un parámetro relacionado con la escala de las coordenadas de la trayectoria deseada, $0 \leq d\theta \leq 0.5$, $-C^* \leq x_0, y_0 \leq C^*$. Obviamente estos límites pueden variar de acuerdo al problema, sin embargo, para los experimentos mostrados en este trabajo sólo variamos el valor de C^* , y calculando todos los demás límites como se describe arriba.
- 2) Condición de Grashof. Para asegurar la movilidad adecuada del mecanismo se debe de cumplir la condición de Grashof. Para este trabajo la escribimos de la misma forma que Cabrera et al. (2002): $(r_1 + r_2 < r_3 + r_4)$ donde $(r_2 < r_3 < r_4 < r_1)$.
- 3) Condición de conectividad de los elementos. El mecanismo es factible si la Ecuación 8 tiene soluciones reales y es no factible si tiene soluciones complejas conjugadas.

Método: Algoritmo de Estimación de Distribución

Los algoritmos de estimación de distribución (EDAs) fueron derivados del modelado probabilístico de los algoritmos genéticos, en los cuales se sustituyeron los operadores de cruce y mutación por la estimación y muestreo de una distribución de probabilidad, (Larrañaga y Lozano, 2001). Los pasos básicos del algoritmo son: generar una población inicial, generalmente aleatoria uniforme, evaluar la población en la función objetivo, seleccionar un subconjunto de las mejores soluciones, y estimar los parámetros de una distribución utilizando el conjunto seleccionado, tomar muestras de la nueva distribución y repetir. Una de las grandes ventajas de los algoritmos de estimación de distribución es la posibilidad de codificar explícitamente dependencias entre las variables, dentro de la distribución de búsqueda.

El algoritmo que se presenta en este trabajo es similar al Estimation of Multivariate Normal Distribution (EMNA) de Larrañaga y Lozano (2001). Debido a las bondades de este tipo de algoritmo tales como la simplicidad, bajo costo computacional del muestreo y estimación de los parámetros (usualmente por máxima verosimilitud), así como resultados prometedores en un

conjunto problemas, existe una variedad de propuestas de algoritmos similares que utilizan una distribución Normal multivariada como distribución de búsqueda (Dong y Yao, 2008), (Grahl et al., 2006). Sin embargo, trabajos recientes reportan que los EDAs sufren de convergencia prematura (Handa, 2007), (Shapiro, 2006), (Dong y Yao, 2002), varios de estos trabajos están totalmente avocados a los EDAs basados en la Normal multivariada, tales como Grahl et al. (2007), Bosman et al. (2007), Cai et al. (2007), Yuan y Gallagher (2005); debido a una tendencia natural a reducir la varianza más rápido de lo necesario. En consideración de lo anterior, los investigadores del área se han dedicado a proponer algoritmos que mantienen una varianza más adecuada para evitar la convergencia prematura. Para el trabajo actual utilizamos una versión modificada del EDA basado en la Normal multivariada que utiliza la selección de potencial (potential selection) de Valdez et al. (2012), el cual se muestra en la Tabla 1. Los pasos que se detallan a continuación son para la *maximización* de una función, debido a que simplifica los cálculos, esto no representa mayor problema si se considera que minimizar $f(x)$ equivale a maximizar $-f(x)$.

Tabla 1. Algoritmo de estimación de distribución (para maximización)

Entrada: n_{pop} =tamaño de la población, MinCov= mínima norma de la covarianza para criterio de paro. n_{gen} = número máximo de generaciones.	
1	Generar y evaluar una población inicial X_0
2	Inicializar el umbral T_0 y la mejor solución encontrada x_{mejor}, f_{mejor} .
3	Seleccionar las soluciones con función de objetivo mayor o igual al umbral.
4	Calcular el potencial sobre las soluciones seleccionadas.
5	Estimar los parámetros de la matriz de covarianza y fijar la media de la distribución de búsqueda Q_t .
6	Generar y evaluar una nueva población X_t utilizando Q_t .
7	Actualizar el umbral T_t y la mejor solución encontrada x_{mejor}, f_{mejor} .
8	Si el criterio de paro no se cumple, ir al paso 3.

Aunque el algoritmo de la Tabla 1 presenta como valores de entrada el tamaño de la población, una varianza mínima y un máximo número de generaciones, el lector debe de considerar que el único parámetro real es el tamaño de la población, la mínima varianza sirve para compararla contra la norma de la matriz de covarianza, el cual es un parámetro suficientemente pequeño ($1e-7$ para todos nuestros experimentos), que determina si el algoritmo ha convergido y la mejor solución difícilmente mejorará, ya que la exploración se encuentra dentro de una vecindad muy pequeña, adicionalmente se utiliza el número de generaciones para detener el algoritmo.

En este trabajo proponemos una forma de muestreo que asegura que todos los individuos en la población son factibles, es decir que cumplen con las restricciones mencionadas en la sección anterior. La forma de realizar la evaluación y los pasos del EDA se detallan a continuación.

Pasos 1 y 6. Generar y evaluar la población inicial X_0

La evaluación de la función objetivo dentro del algoritmo se realiza de la siguiente forma:

$$f_{obj}(X) = \begin{cases} 1.0e100 & \text{si la condición de conectividad no se cumple, Ecuación 8} \\ 1.0e100 & \text{si la condición de Grashof no se cumple} \\ f(X) & \text{de la Ecuación 11 de otra forma} \end{cases}$$

Por lo tanto un valor de $1e100$ indica que la solución candidata no es factible. La población se genera de acuerdo con el algoritmo en la Tabla 2.

Tabla 2. Algoritmo para la generación de nuevos individuos.

1	While ($k < n_{pop}$) {
2	Generar y evalúa 1 solución candidata X_c .
3	Si la solución X_c es factible agregar a la población y hacer $k=k+1$
4	}

La población inicial se genera utilizando una distribución uniforme dentro de los límites definidos para cada variable. Sin embargo, a diferencia de los EDAs convencionales y la base del que aquí se presenta, (Valdez et al., 2012), en esta propuesta aseguramos que todas las soluciones

utilizadas en el aprendizaje son factibles. El EDA intenta *aprender* cuales son los valores adecuados de cada variable y codifica este aprendizaje en una distribución de probabilidad. Note que si se utilizaran soluciones no factibles en el aprendizaje, la calidad final de la aproximación sería menor, y el algoritmo convergería más lento debido a que se estaría aprendiendo de valores de soluciones que no tienen significado físico. El paso seis se realiza exactamente del mismo modo, solo considerando que la muestra es tomada de una Normal multivariada con un vector de medias μ y una matriz de covarianza Σ . La manera de calcular estos parámetros se especifica más adelante.

Pasos 2 y 3. Inicializar el umbral T_0 y la mejor solución encontrada x_{mejor}, f_{mejor}

El umbral T_t se utiliza para seleccionar un subconjunto de las mejores soluciones. Las soluciones cuya función objetivo son mayores que T_t son seleccionadas y utilizadas, junto con su valor de potencial, para el cálculo de los parámetros de la distribución de probabilidad. El umbral inicial T_0 toma el peor valor de función objetivo obtenido en la primera generación, considerando solamente soluciones factibles generadas de acuerdo con la subsección anterior. El paso 3 como ya se mencionó solo consiste en seleccionar las soluciones con valor de función objetivo mayor o igual al umbral (para maximización).

Paso 4. Calcular el potencial sobre las soluciones seleccionadas

La función de potencial definida en Valdez et al. (2012), se utiliza para mantener una varianza adecuada en el proceso de búsqueda, como se mencionó el EDA estándar basado en la Normal multivariada (como el EMNA), sufre de convergencia prematura. Debido a esto existen un número importante de algoritmos modificados que intentan mantener una varianza adecuada. La estrategia de utilizar la función de potencial se basa en un método de estimadores pesados, utilizados previamente por Valdez et al. (2010) y Teytaud y Teytaud (2009). En Valdez et al. (2012) se presenta una segunda propuesta, en la cual los pesos calculados ayudan a preservar una varianza adecuada durante la búsqueda, esto mejora evidentemente las aproximaciones del óptimo para una variedad de problemas con respecto de otros métodos similares. Dada una solución x_i en el conjunto seleccionado el potencial para esta solución está dada por la siguiente ecuación:

$$p(x_i) = \frac{\hat{p}(x_i)}{\sum_{i=1}^{n_{sel}} \hat{p}(x_i)} \quad (12)$$

para i en el conjunto seleccionado. Donde:

$$\hat{p}(x_i) = \max(p_d(x_i), p_f(x_i)), \quad (13)$$

$$p_d = \frac{\hat{p}_d(x_i) - \min(\hat{p}_d(x_i))}{\max(\hat{p}_d(x_i)) - \min(\hat{p}_d(x_i))}, \quad (14)$$

$$p_f = \frac{f(x_i) - \min(f(x_i))}{\max(f(x_i)) - \min(f(x_i))}, \quad (15)$$

y

$$\hat{p}_d = \sqrt{\sum_{j=1}^n (x_{i,j} - x_{mejor,j})^2}. \quad (16)$$

el potencial de la Ecuación (12), es un valor normalizado que suma 1, al igual que una distribución discreta de probabilidad y se puede interpretar como tal. Antes de normalizar, en la Ecuación (13), el potencial es el máximo entre los valores p_d y p_f , que se calculan para cada individuo en el conjunto seleccionado de acuerdo con las Ecuaciones (14) y (15). La Ecuación (14) devuelve un valor entre 0 y 1 para cada individuo de acuerdo a la distancia Euclidiana entre éste y la mejor solución encontrada x_{mejor} , la distancia se calcula con la Ecuación (16). Finalmente, p_f es la función objetivo normalizada entre 0 y 1. Es importante considerar lo siguiente:

- Los individuos utilizados en el cálculo del potencial y posteriormente en el cálculo de los parámetros de la distribución son solo los que son mejores que el umbral T_t por lo que se puede considerar que representan la mejor región del espacio de búsqueda.
- El potencial antes de normalizar (Ecuación 13) devuelve 1 para el mejor individuo x_{best} y para el individuo más alejado del mejor (el más alejado con función objetivo arriba del umbral).
- Por lo tanto, la selección de potencial se puede ver como una distribución apriori que confiere mayor probabilidad al mejor individuo, pero también a individuos muy alejados del mejor que tienen un valor de función objetivo suficientemente alto, lo que mantiene la varianza lo suficientemente grande como para cubrir la mejor región.

Estimar los parámetros de la matriz de covarianza y fijar la media de la distribución de búsqueda Q_t

La distribución de búsqueda $Q_t(\mu, \Sigma)$ utilizada para generar nuevas soluciones candidatas es una Normal multivariada con vector de medias μ y matriz de covarianza Σ , al vector de medias

simplemente se le asigna la mejor solución encontrada $\mu=x_{mejor}$, esto indica que se le confiere la mayor probabilidad a la vecindad de x_{mejor} . Para la matriz de covarianza se utiliza la siguiente ecuación:

$$\Sigma_{i,j} = \frac{\sum_{k \in I^s} (x_{k,i} - \mu_i)(x_{k,j} - \mu_j) p_k}{\sum_{k \in I^s} p_k} \quad (17)$$

Donde I^s representa el conjunto de los índices del conjunto seleccionado (individuos con funciones arriba del umbral). El índice k corre sobre los individuos seleccionados, los índices i y j indican que se está calculando la covarianza de la variable i con la variable j .

El criterio de paro, como ya se ha mencionado, es cuando la norma de la matriz de covarianza es menor que un parámetro denominado MinCov o el número máximo de generaciones.

Resultados

En esta sección presentamos los resultados obtenidos con la metodología propuesta. Para este fin utilizamos los experimentos reportados en Cabrera et al. (2002) definidos por 3 conjuntos de puntos de precisión listados a continuación junto con los parámetros de entrada

1) Cd=[(20,20), (20,25),(20,30),(20,35),(20,40),(20,45)]. Parámetros de entrada, npop=3000, ngen=600 y MinCov=1.0e-7. C^* para definir el espacio de búsqueda que es: 20.

2) Cd=[(3.0,3.0), (2.759,3.363),(2.372,3.663),(1.890,3.862),(1.355,3.943)]. Parámetros de entrada, npop=3000, ngen=600 y MinCov=1.0e-7. C^* para definir el espacio de búsqueda que es: 2.

3) Cd=[(0.5,1.1),(0.4, 1.1),(0.3, 1.1),(0.2, 1.0),(0.1, 0.9),(0.005, 0.75),(0.02, 0.6),(0.0, 0.5),(0.0,0.4),(0.03, 0.3),(0.1, 0.25),(0.15,0.2),(0.2, 0.3),(0.3, 0.4),(0.4, 0.5),(0.5, 0.7),(0.6, 0.9),(0.6, 1.0)]. Parámetros de entrada, npop=3000, ngen=600 y MinCov=1.0e-7. C^* para definir el espacio de búsqueda que es: 1.

Caso 1. La mejor solución encontrada para este caso se muestra en la Figura 2. El error mínimo encontrado es de 0.000087027. Los parámetros encontrados son: $x_0=24.201123$, $y_0=-20.707550$, $r_1=49.999990$, $r_2=38.526761$, $r_3=38.526762$, $r_4=49.999990$, $r_{cx}=17.760130$, $r_{cy}=50.000000$, $\theta_0=1.570796$, $\theta_2^0=5.506283$ y $d\theta=0.096570$.

Caso 2. La mejor solución encontrada para este caso se muestra en la Figura 3. El error mínimo encontrado es de 0.00000000000010790. Los parámetros encontrados son: $x_0=0.702302$,

$y_0=2.128946$, $r_1=2.049439$, $r_2=1.100611$, $r_3=1.843147$, $r_4=2.049431$, $r_{cx}=0.364729$,
 $r_{cy}=1.307486$, $\theta_0=-1.443613$, $\theta_2^0= 3.823973$ y $d\theta=0.398471$.

Caso 3. La mejor solución encontrada para este caso se muestra en la Figura 4. El error mínimo encontrado es de 0.0000091773883981. Los parámetros encontrados son: $x_0=0.258735$,
 $y_0=0.218143$, $r_1=1.109130$, $r_2=0.436477$, $r_3=0.832682$, $r_4=0.832699$, $r_{cx}=0.371051$,
 $r_{cy}=0.327572$, $\theta_0=-1.443613$, $\theta_2^0= 0.088915$ y $d\theta=0.339945$.

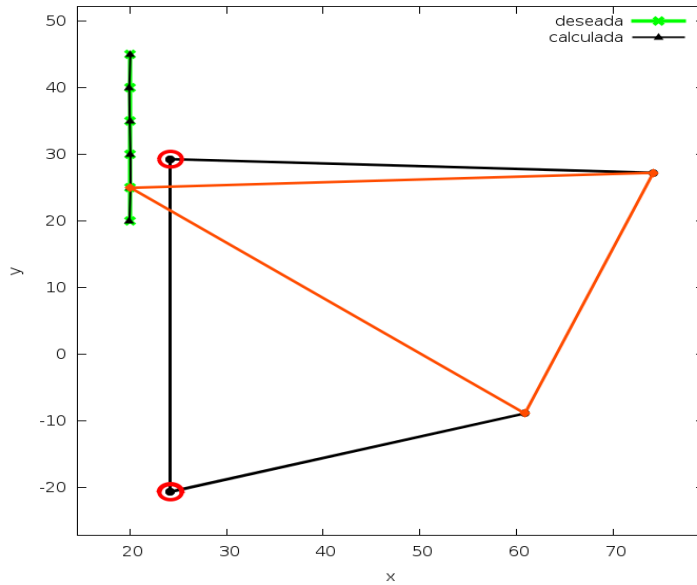


Figura 2. Mejor solución encontrada para el caso 1. Error= 0.000087027, de acuerdo a la Ecuación 11.

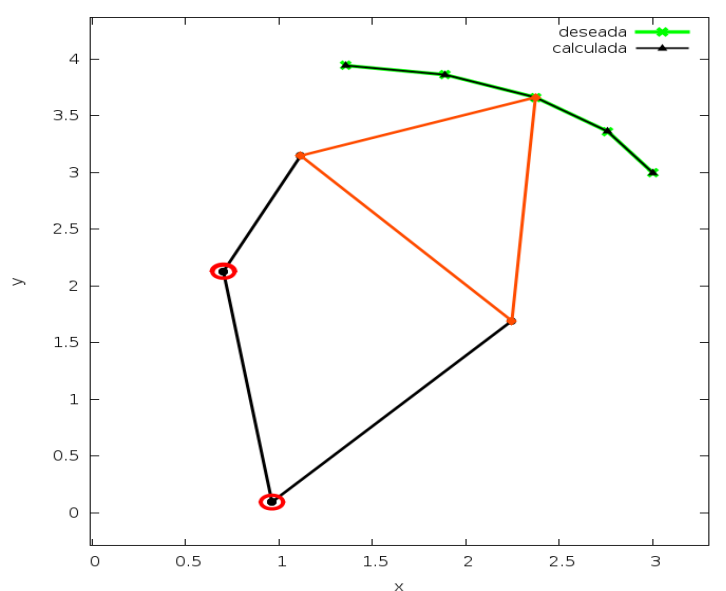


Figura 3. Mejor solución encontrada para el caso 2. Error 0.0000000000010790.

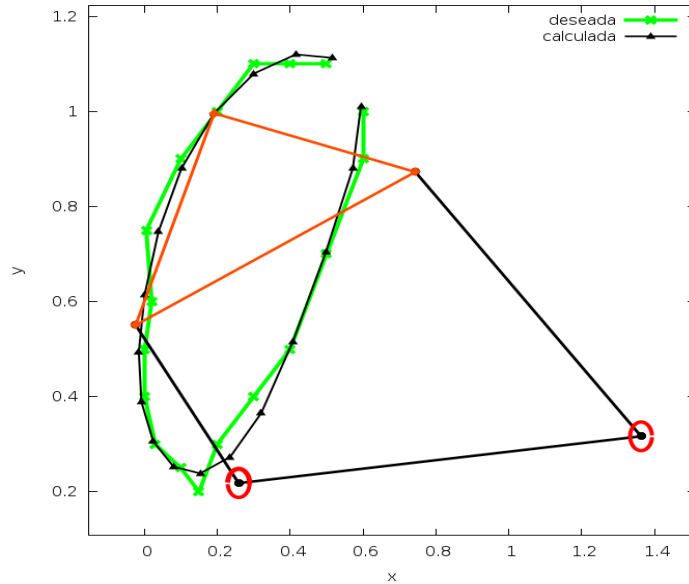


Figura 4. Mejor solución encontrada para el caso 3. Error 0.0000091773883981.

Discusión de los resultados

En la sección anterior se mostró la capacidad del algoritmo para encontrar soluciones muy aproximadas a las óptimas. Podemos comparar los resultados obtenidos con otros de la literatura, Cabrera et al. (2002). Como se muestra en la Tabla 3.

Tabla 3. Comparativo de métodos en la literatura para el problema de síntesis de mecanismos para las trayectorias propuestas por Cabrera et al. (2002)

Algoritmo:	Diferencia Central	Gradiente Exacto	Algoritmo Genético KK	Algoritmo Genético, Cabrera et al. (2002)	EDA Normal
Error Caso 1	No reportado	No reportado	No reportado	0.02617	0.000087027
Error Caso 2	6.276e-3	6.8e-5	9.54e-4	1.827e-6	1.079e-12
Error Caso 3	2.66e-2	1.68e-2	4.3e-2	2.45e-2	9.177388e-06

Como se puede ver en la Tabla 3, los resultados del algoritmo propuesto en este trabajo son muy superiores a los otros. Cabe mencionar, sin embargo que los algoritmos de estimación de distribución comúnmente utilizan una población mayor (y por lo tanto un mayor número de

evaluaciones) que otros algoritmos poblacionales como los algoritmos genéticos, pero es un precio a pagar por la calidad de los resultados, y la capacidad de utilizar las correlaciones entre las variables. Como en este caso, vale la pena pagar el tiempo de cómputo de aproximadamente 3 segundos por ejecución en una computadora pentium i7 quadcore, con 8 Gb de RAM. Es un tiempo realmente despreciable si se considera la ganancia a largo plazo en un mecanismo real.

El lector debe de notar que la forma en que se propuso el modelo de optimización utilizando una velocidad constante en el par motriz $d\theta$ como parámetro extra a optimizar, permite aproximar trayectorias definidas por un número arbitrario de puntos, sin modificar la dimensionalidad de los parámetros a optimizar. Por otro lado, otra posibilidad sería realizar un barrido de la trayectoria, discretizando los 360 grados del par motriz, lo cual podría eliminar el parámetro $d\theta$ del modelo de optimización y entregar un resultado con un error menor al que se reporta aquí. Sin embargo, esta otra estrategia no permitiría aproximar tanto la trayectoria como la velocidad.

Por otro lado los métodos de optimización convencionales como los basados en gradiente requieren una función suave, este no es el caso para los EDAs, por lo cual se puede anexar al modelo otras características deseadas sin preocuparse por mantener propiedades de convexidad y suavidad de la función objetivo.

Conclusiones y trabajo futuro

En el presente trabajo abordamos el problema de la síntesis de mecanismos, en particular, estudiamos el mecanismo plano de cuatro eslabones. El método utilizado es un algoritmo de estimación de distribución basado en la distribución Normal, las aportaciones más relevantes de este enfoque son: a) utilizar la información de correlaciones entre variables, b) obtener una muestra factible para el aprendizaje de la distribución, c) modelar el problema utilizando un parámetro $d\theta$ que permite la inserción de un número arbitrario de puntos de la trayectoria sin aumentar la dimensionalidad del problema de optimización, y d) propusimos una forma paramétrica de fijar los límites de búsqueda para este mecanismo. Por otro lado, existían bondades en el algoritmo utilizado de base, que ya de por sí eran importantes para la solución de este problema, tales como la selección de potencial, que permite realizar una búsqueda adecuada. Lo anterior resultó en un algoritmo que entrega resultados altamente competitivos. En el trabajo

presente los mecanismos sólo tienen restricciones básicas, el trabajo futuro puede contemplar el problema de minimizar las longitudes de los elementos, restringir el espacio de trabajo, insertando restricciones de singularidades, y posiblemente de control. La metodología propuesta entrega resultados alentadores, como para aplicarla a problemas más complejos, es posible aplicar la misma metodología a otros mecanismos de cadena cinemática cerrada. Por otro lado, se puede también resolver el mismo problema con otros métodos de optimización y analizar estadísticamente el desempeño de la propuesta con respecto de otras técnicas de optimización global.

Referencias

- Bosman P. A., Grahl J. y Rothlauf F. (2007). SDR: A Better Trigger for Adaptive Variance Scaling in Normal EDAs. In: GECCO '07: Proceedings of the 8th annual conference on Genetic and evolutionary computation, 516–522.
- Bulatovic R. y Djordjevic S. R. (2004). Optimal Synthesis of a Four bar Linkage by Method of Controlled Deviation. The first Conference on Computational Mechanics (CM04), 31(4), 265-280.
- Cabrera J. A., Simon A. y Prado M. (2002). Optimal Synthesis of Mechanisms with Genetic Algorithms. *Mechanism and Machine Theory*, 37, 1165-1177.
- Cabrera J. A., Nadal F., Muñoz J. P. y Simon A. (2007). Multiobjective Constrained Optimal Synthesis of Planar Mechanisms using a New Evolutionary Algorithm. *Mechanism and Machine Theory*, 42, 791-806.
- Cai Y., Sun X., Xu H. y Jia P. (2007). Cross Entropy and Adaptive Variance Scaling in Continuous EDA. In: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, GECCO '07. ACM, New York, NY, USA, 609–616.
- Dong W. y Yao X. (2002). NichingEDA: Utilizing the Diversity inside a Population of EDAs for Continuous Optimization. In: IEEE Congress on Evolutionary Computation, 1260–1267.
- Dong W. y Yao X. (2008). Unified Eigen Analysis on Multivariate Gaussian based Estimation of Distribution Algorithms. *Information Sciences*, 178(15), 3000 – 3023.
- Fang, W. E. (1994). Simultaneous Type and Dimensional Synthesis of Mechanisms by

- Genetic Algorithms-DE. Mechanism Synthesis Analysis, 70.
- Freudenstein F. (1954). An Analytical Approach to the Design of Four-link Mechanisms. Transactions of the ASME, 76, 483–492.
- Grahl J., Bosman P. A. N. y Minner S. (2007). Convergence Phases, Variance Trajectories, and Runtime Analysis of Continuous EDAs. In: GECCO '07: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, ACM, 516–522.
- Grahl J., Bosman P. A. N. y Rothlauf F. (2006). The Correlation-triggered Adaptive Variance Scaling IDEA. In: GECCO '06: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation, ACM Press, 397–404.
- Handa H. (2007). The Effectiveness of Mutation Operation in the Case of Estimation of Distribution Algorithms. Biosystems, 87(23), 243 – 251.
- Hartenberg R. y Denavit J. (1964). Kinematic Synthesis of Linkages, McGraw-Hill, New York.
- Hrones, J. A. y Nelson G. L. (1951). Analysis of the Four Bar Linkage. MIT Press and Wiley, New York.
- Kalnas R. y Kota S. (2001). Incorporating Uncertainty into Mechanism Synthesis. Mechanism and Machine Theory, 36(3), 843-851.
- Kunjur, J. A. y Krishnamurthy S. (1997). Genetic Algorithms in Mechanical Synthesis. Journal of Applied Mechanisms and Robotics, 4(2), 18-24.
- Larrañaga P. y Lozano J. A. (2001). Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation. Kluwer Academic Publishers, Norwell, MA, USA .
- Laribi M. A., Milak A., Romdhane L. y Zegloul S. (2004). A Combined Genetic Algorithm Fuzzy Logic Method (GA-FL) in Mechanisms Synthesis. Mechanism and Machine Theory, 39, 717-735.
- Levitskii N. L. y Shakvazian K. K. (1960). Synthesis of Four Element Spatial Mechanisms with Lower Pairs. International Journal of Mechanical Sciences, 2, 76-92.
- Mallik A. K., Ghosh A. (1994). Kinematic Analysis and Synthesis of Mechanisms. CRC-Press 1 edition, 688.
- Marcelin J. (2010). Integrated Optimization of Mechanisms with Genetic Algorithms. Engineering, 2, 438-444.
- Pelikan M., Goldberg D. E. y Cantú-Paz E. (1999). The Bayesian Optimization Algorithm. In:

- Proceedings of the 1999 Conference on Genetic and Evolutionary Computation, GECCO '99, ACM, Orlando, FL, USA, 525-532.
- Quintero R. H., Calle-Trujillo G. y Díaz-Arias A. (2004). Síntesis de generación de trayectorias y de movimiento para múltiples posiciones en mecanismos, utilizando algoritmos genéticos. *Scientia et Technica*, 10(25).
- Roston G. P. y Sturges R. H. (1996). Genetic Algorithm Synthesis of Four Bar Mechanisms. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 10, 371-390.
- Sancibrian R., Viadero F., Garcia P. y Fernandez A. (2004). Gradient-based Optimization of Path Synthesis Problems in Planar Mechanisms. *Mechanism and Machine Theory*, 39, 839-856.
- Sandor, G. N. (1959). A General Complex Number Method for Plane Kinematic Synthesis. PhD Thesis, Columbia University, New York.
- Shapiro J. L. (2006). Diversity Loss in General Estimation of Distribution Algorithms. In: Proceedings of the 9th International Conference on Parallel Problem Solving from Nature, PPSN'06, Springer-Verlag, Berlin, Heidelberg, 92-101.
- Starosta R., (2008). Application of Genetic Algorithm and Fourier Coefficients (GA-FC) in Mechanisms Synthesis. *Journal of Theoretical and Applied Mechanics*, 46(2), 395-411.
- Suh C. H. y Radcliffe C. W. (1967). Synthesis of Plane Linkages with use of Displacement Matrix. *Journal of Engineering for Industry*, 89(2), 206-214.
- Teytaud F. y Teytaud O. (2009). Why one Must Use Reweighting in Estimation of Distribution Algorithms. In: Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation, GECCO '09, ACM, New York, NY, USA, 453-460.
- Tzong-Mou W. y Chao-Kuang C. (2005). Mathematical Model and its Simulation of Exactly Mechanism Synthesis with Adjustable Link. *Applied Mathematics and Computation*, 160, 309-316.
- Valdez S. I., Hernández A. y Botello S. (2010). Efficient Estimation of Distribution Algorithms by Using the Empirical Selection Distribution. In: P. Korosec (ed.) *New Achievements in Evolutionary Computation*. InTech. 229-250.

Valdez I., Hernández-Aguirre y Botello S. (2012). Adequate Variance Maintenance in a Normal EDA via the Potential-Selection Method. *Advances in Intelligent Systems and Computing*, Springer Berlin Heidelberg, 221-235.

Vasiliu A. y Yannou B. (2001). Dimensional Synthesis of Planar Mechanism using Neural Network: Application to Path Generator Linkages. *Mechanism and Machine Theory*, 36(2), 229-310.

Walczak T. (2006). Mechanism Synthesis with the Use of Neural Network. Annual Meeting of GAMM, Book of abstracts. Berlin.

Yuan B. y Gallagher M. (2005). On the Importance of Diversity Maintenance in Estimation of Distribution Algorithms. In: *Proceedings of the 2005 Conference on Genetic and Evolutionary Computation, GECCO '05*, ACM, New York, NY, USA, 719–726.

Zhang Q. y Muhlenbein H. (2004). On the Convergence of a Class of Estimation of Distribution Algorithms. *IEEE Transactions on Evolutionary Computation* 8(2), 127–136.

