




Sistema de control para computación evolutiva basado en redes complejas

Control system for evolutionary computing based on complex networks

Joshua Triana Madrid , Víctor Bucheli Guerrero , Ángel García Banos 
Universidad del Valle, Colombia

Open Access

Recibido:

3 de marzo de 2020

Aceptado:

14 de julio de 2020

Publicado:

20 de agosto de 2020

Correspondencia:

victor.bucheli@correounivalle.edu.co

DOI:

<https://doi.org/10.17081/invinno.8.2.3752>



© Copyright: Investigación e Innovación en Ingenierías

Resumen

Objetivo: Implementar un mecanismo de control para las dinámicas de población de un algoritmo evolutivo basado en redes complejas. Se plantea la hipótesis de que la estrategia guiada por una red compleja tiene mejores resultados que un algoritmo evolutivo tradicional. **Metodología:** Se estudia la convergencia del modelo propuesto frente a la solución evolutiva tradicional. Se realizaron análisis estadísticos frente a los resultados experimentales obtenidos para diferentes problemas de optimización. **Resultados:** Las estrategias en las que se integran redes de pequeño mundo como mecanismo de control de las dinámicas de población, tienen un mejor desempeño en general que otras topologías de red. **Conclusiones:** La integración de una estructura de red compleja, como una red subyacente en la dinámica de un algoritmo evolutivo, muestra una ventaja competitiva en relación con estrategias tradicionales, específicamente las redes de pequeño mundo muestran un mejor desempeño.

Palabras claves: Redes Complejas, computación evolutiva, evolución diferencial.

Abstract

Objective: To develop a control mechanism for the population dynamics of an evolutionary algorithm, based on the hypothesis that one strategy guided by a complex network, generates better results than traditional evolutionary algorithms. **Methodology:** The convergence of the proposed model is evaluated experimentally against the traditional evolutionary solution. Statistical analysis were carried out with the experimental results obtained for different optimization problems. **Results:** The strategies in which small world networks are integrated as a mechanism to control population dynamics, have a better overall performance than other network topologies. **Conclusions:** The integration of a complex network structure, such as an underlying network in the dynamics of an evolutionary algorithm, shows a competitive advantage in relation to traditional strategies, specifically small world networks show better performance.

Keywords: Complex Networks, evolutionary computation, differential evolution.

Como citar (IEEE): J. Triana - Madrid., V. Bucheli - Guerrero., y A. García - Banos, "Sistema de control para computación evolutiva basado en redes complejas", Investigación e Innovación en Ingenierías, vol. 8, n°. 2, 2020. DOI: <https://doi.org/10.17081/invinno.8.2.3752>

Introducción

La computación evolutiva se ha implementado e investigado por muchos años desde su presentación en el año 1966 [1] y se ha convertido en un referente de la inteligencia artificial. La estrategia de computación evolutiva ha demostrado ser robusta y con buenos resultados [2], siendo una excelente alternativa para problemas de optimización en general. Para mejorar el desempeño de esta estrategia, se han utilizado variantes como computación evolutiva con restricciones [3], secuencias caóticas [4] y clustering [5], entre otras; las cuales exploran mecanismos de control sobre algunos parámetros como la tasa de mutación, tamaño de población, métodos de cruce, selección de la piscina de emparejamiento (*Mating pool*), entre otros [6]. La introducción de estas variantes mejora la calidad de las soluciones y convergencia, entre otros atributos.

En algunos algoritmos evolutivos se han encontrado redes subyacentes que presentan propiedades de redes complejas [7]. Las redes complejas modelan grafos en el que vértices y aristas tienen un conjunto de propiedades definidas que generan una complejidad y estructura de interés [8], por lo que han servido para entender el comportamiento de muchos sistemas a partir de la relación entre los individuos y sus dinámicas [9, 10, 11, 12, 13]. Las dinámicas de los resultados obtenidos muestran redes con propiedades de mundo pequeño, principalmente un promedio más bajo de longitud de camino y un valor alto de coeficiente de clustering, lo que se espera de este tipo de redes.

Al realizar la replicación de los trabajos previos, se encuentra que las estrategias de computación evolutiva que presentan propiedades de redes complejas en sus redes subyacentes tienen mejores resultados. Se plantea la hipótesis que la estrategia guiada por una red compleja tiene mejores resultados que un algoritmo tradicional. Por lo anterior, se plantea un modelo del mecanismo de control para una estrategia evolutiva tradicional, donde una red de mundo pequeño guía la reproducción de los individuos, permitiendo que estos solo se puedan reproducir con sus vecinos en la red. La formulación de un modelo de control basado en una red compleja que guíe la interacción de los individuos de la población podría ayudar a optimizar la calidad de la solución final de la iteración. La implementación de este modelo permitirá la realización de un análisis comparativo de su propuesta con la estrategia original. Los resultados encontrados en este trabajo permiten avanzar en la idea de que la presencia de estructuras no triviales en algoritmos de optimización puede influir de forma positiva en su desempeño.

Algoritmos evolutivos y redes complejas

Evolución diferencial

De acuerdo con el trabajo original de Storn [14], la evolución diferencial comienza con una población inicial de individuos generados por una distribución uniforme pseudoaleatoria en un espacio de búsqueda D . En cada generación y para cada individuo x_i , se le asignan tres vectores distintos x_r , x_s y x_t de la población y se genera un nuevo individuo a partir de la ecuación (1), en la que F es una función de escalado que controla la longitud del vector $(x_r - x_s)$. La regla de la ecuación (1) es conocida como DE/rand/1 y tiene una implementación sencilla y un número limitado de parámetros para optimizar, una revisión de literatura sobre aplicaciones se puede ver en el trabajo de Plagianakos [15].

$$x_{off} = x_t + F(x_r - x_s) \quad (1)$$

Otras variantes de la ecuación DE/rand/1 han sido propuestas en la literatura, las cuales se muestran en las ecuaciones (2-6) [16]. Donde x_{off} es el individuo que se quiere recombinar, x_{Best} es el individuo con mejor desempeño en la población, x_u y x_v son dos individuos generados de forma pseudoaleatoria.

$$DE/best/1: x_{off} = x_{best} + F(x_s - x_t) \quad (2)$$

$$DE/cur - to - best/1: x_{off} = x_i + F(x_{best} - x_i) + F(x_s - x_t) \quad (3)$$

$$DE/best/2: x_{off} = x_{best} + F(x_s - x_t) + F(x_u - x_v) \quad (4)$$

$$DE/rand/2: x_{off} = x_r + F(x_s - x_t) + F(x_u - x_v) \quad (5)$$

$$DE/rand - to - best/2: x_{off} = x_t + F(x_{best} - x_i) + F(x_r - x_s) + F(x_u - x_v) \quad (6)$$

Una versión alternativa (ecuación 7) fue propuesta por [17], donde K es el coeficiente de combinación $\in [0,1]$ y $F' = K * F$

$$DE/current - to - rand/1: x_{off} = x_i + K(x_t - x_i) + F'(x_r - x_s) \quad (7)$$

Redes complejas en computación evolutiva

En 2002 Dorogovtsev y Mendes [18] centraron el interés en las propiedades estructurales de redes complejas en comunicaciones, biología, ciencias sociales y economía. Estas redes artificiales gigantes fueron pioneras en su tipo, lo que permitió el estudio de su tipología y evolución, además de los procesos complejos que ocurren en ellas. Dichas redes poseen un amplio conjunto de propiedades de escala. Una de las conclusiones más importan-

tes es que a pesar del gran tamaño de estas redes, la distancia entre la mayoría de sus vértices es corta, una característica conocida como el efecto de mundo pequeño.

Por su parte, las redes de mundo pequeño según Amaral y otros [19] son aquellas que presentan las siguientes propiedades: primero, la distancia más corta promedio entre un par de vértices es proporcional al logaritmo natural del número de nodos y el coeficiente de agrupamiento es grande, lo que indica que los vértices de la red tienden a estar fuertemente conexos.

Existen estrategias de computación evolutiva que permiten modelar las relaciones entre sus individuos en el tiempo por medio de redes complejas, entre las que destacan SOMA [20] y evolución diferencial [21]. El primer trabajo sobre el tema de investigación [8] usa precisamente el algoritmo SOMA y la evolución diferencial para modelar redes complejas. Esta investigación preliminar modela las relaciones descendencia padre e hijo como redes y analiza las propiedades topológicas, entre las que se destacan la longitud promedio de camino más corto con valores pequeños y las distribuciones tipo ley de potencia. El autor demuestra que no en todas las configuraciones ni ejecuciones de los algoritmos se presentan estructuras de redes complejas, por lo que propone utilizar las propiedades de este tipo de redes para el beneficio del algoritmo. El autor siguió publicando trabajos sobre el tema donde plantea más variantes de los algoritmos ya estudiados [9,10,11], usar otras familias de algoritmos de optimización como Lattice Maps [12] y una revisión de lecciones aprendidas [13]. Otros trabajos han estudiado y analizado un grupo de estrategias de algoritmos evolutivos enfocados principalmente en el análisis de centralidad [22].

Se ha hecho una revisión de literatura [23] sobre la relación de computación evolutiva y las redes complejas. El autor estudia trabajos previos en los que se han modelado las redes en los algoritmos de optimización y concluye con una metodología para determinar el grado de distribución para analizar la convergencia prematura, la longitud promedio de los caminos entre los nodos para determinar la existencia de redes complejas, el coeficiente de clustering para analizar el progreso individual en cada iteración y la centralidad entre los nodos como alternativa para encontrar los mejores individuos. Sin embargo, la metodología propuesta carece de implementación y análisis de resultados, es decir se plantea únicamente de forma conceptual.

Se han realizado análisis de las dinámicas evolutivas de las redes complejas [24], donde se indica la posibilidad de disminuir las tasas de error de ciertos algoritmos y su aplicación en las redes de transporte y comunicación. También se ha realizado el proceso inverso, en el que se aplica la computación evolutiva para el análisis de propiedades topológicas de la red [25] y se detectan mejores los mejores individuos de una red [26].

Este trabajo propone como estrategia que para cada individuo de la población $x_{i,G}$, se genere un nuevo individuo $x_{i',G}$ a partir de la diferencia de pesos de dos individuos $x_{r1,G}$ y $x_{r2,G}$ sumada a otro individuo $x_{r3,G}$, todos seleccionados aleatoriamente. Este nuevo individuo $x_{i',G}$ se cruza con el individuo original $x_{i,G}$. Si este nuevo vector llamado $x_{i,G+1}$ es mejor que el individuo original, se reemplaza por el individuo original $x_{i,G}$, en caso contrario se descarta la nueva solución.

En esta implementación se omite la creación de nuevos cromosomas (nodos) de la manera convencional y se actualizan los valores del padre por los valores del hijo, manteniendo el tamaño de la población constante y anotando los cromosomas que intervinieron en la actualización (mediante aristas).

Luego de analizar variantes del modelo [13] se encontró que son muy importantes los parámetros, pues con los valores incorrectos no hay presencia de redes complejas ocultas en los algoritmos evolutivos. Por esta razón se sugiere:

Número de generación: Este es un parámetro sensible, si el número de generaciones es pequeño no se encontraron redes complejas

Tamaño de la población: Cuando el tamaño de la población fue mayor a 20 se encontraron redes complejas

Algoritmo usado: Se obtuvieron redes complejas principalmente de variaciones que se acercaban más a las del tipo enjambre, en otras palabras, cuando emerge un líder.

Metodología

La estrategia de computación evolutiva tradicional es un algoritmo iterativo en el que se tiene una población de individuos que intentan resolver un problema de optimización numérica. Los individuos se mezclan y reproducen entre sí, con el fin de generar nuevos individuos que potencialmente puedan resolver el problema mejor que ellos mismos.

En la estrategia de computación evolutiva tradicional se podrían generar redes a partir de las relaciones entre sus individuos. La relación detectable en esta estrategia es la relación padre e hijo, establecida cuando un individuo sirve para crear o mejorar otro individuo. Esta interacción ocurre en el proceso de reproducción donde, por medio de alguna técnica de selección, dos individuos generan un nuevo individuo recombinaando sus genes. Con este modelamiento es posible crear redes a partir de la relación de descendencia de los individuos.

La generación de redes se pudo realizar exitosamente, pero sus propiedades topológicas no coinciden con la esperadas en las redes complejas. Este resultado y la ventaja competitiva del algoritmo donde emergen redes

complejas subyacentes son la inspiración para el modelo propuesto en este trabajo. Se plantea tomar una red usando el modelo de mundo pequeño planteado por Watts y Strogatz [27] para guiar a la estrategia de computación evolutiva tradicional.

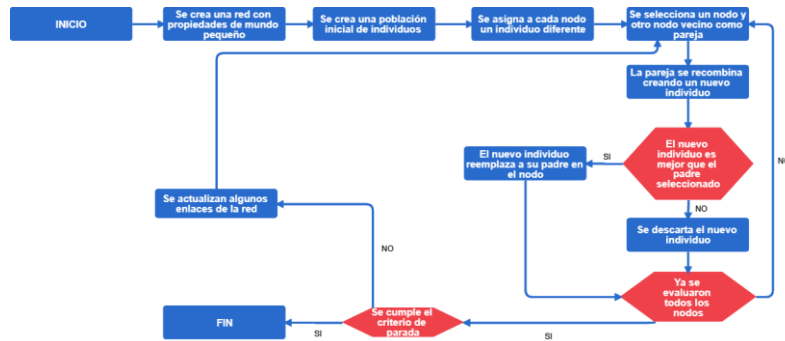
La métrica usada para comparar el modelo propuesto con la estrategia de computación evolutiva tradicional será la comparación del desempeño del mejor individuo encontrado en cada iteración. De esta forma podemos comparar la mejor solución encontrada en cada iteración y ver la evolución del modelo en el tiempo. Para obtener resultados más fiables usamos los promedios del mejor individuo en cada iteración en varios experimentos.

Generación de individuos

Inicialmente se crea una población de individuos para iniciar la estrategia de computación evolutiva y una red de mundo pequeño con el mismo número de nodos que el tamaño de la población de individuos. A continuación, se realiza una asociación pseudoaleatoria de un individuo de la población con un nodo de la red, para proceder a iniciar la iteración del algoritmo, donde se toma cada individuo de la población y se selecciona un vecino de su nodo asociado para reproducirse. De esta forma se guía la reproducción de los individuos por medio de la red de mundo pequeño y la relación entre los nodos. Si el nuevo individuo generado tiene un mejor desempeño que el padre inicialmente seleccionado, se hace un reemplazo directo del hijo por el padre y así se mantiene la uniformidad o tamaño de la población. Este principio de uniformidad es necesario para garantizar que la red siga teniendo la misma estructura y no pierda las propiedades de mundo pequeño.

El mecanismo descrito anteriormente podría presentar una convergencia prematura al conservar la misma red por todas las generaciones. La solución que se plantea es seguir la idea de Watts [27] y hacer un proceso de recableado (*rewiring*). Cada enlace de cada nodo i de la red que conecta a otro nodo i' con cierta probabilidad (30%), rompe el enlace y se une a otro nodo i'' . De esta forma en cada iteración la red cambia ligeramente y le agrega dinamismo. En la Figura 1 se explica el flujo completo del modelo de control.

Figura 1. Mecanismo de control para computación evolutiva guiado por redes complejas

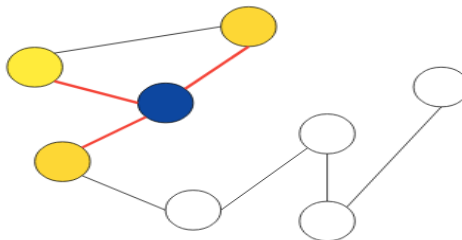


Fuente: Elaboración propia

Selección de vecinos

Un nodo sólo puede seleccionar otro nodo vecino como candidato para reproducirse como se muestra en la Figura 2. En este ejemplo, el nodo azul solo podría seleccionar alguno de sus tres vecinos amarillos para reproducirse. La decisión que debe tomar el nodo es con cuál de sus vecinos reproducirse. Para resolver este problema, se plantean las siguientes variantes de selección de vecinos:

Figura 2. Ejemplo selección de vecinos



Fuente: Elaboración propia

Modelo estándar: Esta variante llamada Small World, consiste en la selección aleatoria de cualquier vecino. Se espera representar con su nombre la intención de que lo importante es la guía de estas redes a los individuos, más que cual vecino selecciona.

Modelo mejor vecino: Esta variante llamada SW Best, y consiste en la selección del vecino por medio del método de ruleta. La intención es favorecer la selección del vecino con mejor desempeño (fitness) con mayor probabilidad.

Modelo de mayor grado: Esta variante llamada SW High Degree, consiste en la selección del vecino con mayor grado. La intención es seleccionar el vecino con mayor relevancia en la red.

Modelo aleatorio: Además de los modelos ya explicados, se implementa para las pruebas una variante de la estrategia evolutiva de nombre Evolutivo Random (que llamaremos No SW). Este es un algoritmo evolutivo donde en cada iteración, cada uno de los individuos selecciona una pareja (método de la ruleta), entre toda la población e ignorando la red, para recombinarse. En caso de que el nuevo individuo generado sea mejor que su padre, se realiza un reemplazo directo. La razón de incluir esta estrategia es medir el impacto de emergencia de la red compleja, en contraparte al modelo propuesto en este trabajo

Funciones de prueba para problemas de optimización

Al igual que en los trabajos previos del estado del arte donde se estudian las redes complejas en algunos algoritmos de computación evolutiva, se seleccionan algunas funciones de prueba para problemas de optimización [28]. Para este trabajo, se seleccionaron las funciones de prueba para problemas de optimización Esfera, Beale, Ackley y Three-hump camel usadas para problemas de un solo objetivo [29] para probar el modelo propuesto. Estas funciones son usadas en matemática aplicada y problemas de optimización para evaluar características como tasa de convergencia, desempeño, robustez, etc.

Configuración de experimentos

En cada experimento se hicieron 50 iteraciones y los resultados presentados son el resultado del promedio de 100 experimentos en cada iteración.

Para cada función de optimización se muestran los resultados de las 50 iteraciones con las estrategias descritas previamente. También se muestra una versión en las últimas 20 iteraciones quitando el peor algoritmo (que hace ruido en este caso), para que se alcance a ver la diferencia de los resultados en las pruebas que convenga. Los parámetros relevantes para las estrategias se muestran en la Tabla 1.

Tabla 1. Configuración de parámetros en las pruebas realizadas

Parámetro	Valor
Tamaño de población	100 individuos
Tasa de mutación	0.1 (10%)
Grado de aburrimiento (criterio de parada)	50 iteraciones

Fuente: Elaboración propia

Resultados

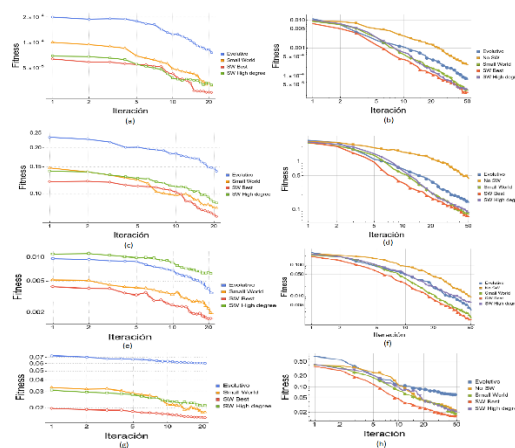
El primer experimento realizado cuando se replicó el modelo de los antecedentes da la idea para implementar el sistema de control guiado por una red de mundo pequeño. En los experimentos realizados se ve claramente como el modelo propuesto tiene un desempeño dominante en todas las iteraciones, aunque a partir de la iteración 25 la dominancia se hace sutil.

La estrategia No SW fue superada por las demás estrategias en todos los experimentos por lo que se concluye que la selección y reemplazo pseudo-aleatorio resulta crear una desventaja en comparación con resto.

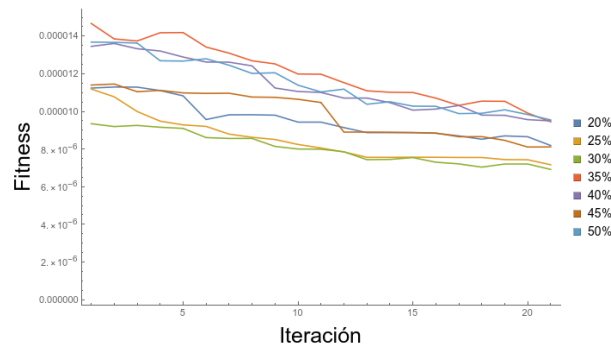
Los experimentos hechos a las funciones de prueba en problemas de optimización evidencian cómo la estrategia evolutiva es superada por la mayoría de las estrategias propuestas en este trabajo en general. En las Figuras 3.b, 3.d, 3.f y 3.h se puede observar como la estrategia original es dominada por las variantes del modelo presentado en este trabajo. Las estrategias Small World y SW Best dominan el resto de las estrategias propuestas incluyendo el modelo base. La estrategia SW Best tiene resultados similares a la variante Small World, pero al final de las iteraciones siempre la supera.

De los resultados experimentales, se puede concluir que las estrategias propuestas superan al modelo evolutivo tradicional para estas funciones de prueba en problemas de optimización. Entre las variantes del modelo propuesto, la estrategia SW Best tuvo mejores resultados y la estrategia SW High Degree tuvo el peor desempeño, como se muestra en la Figura 4.

Figura 3. Experimentos realizados, a) Función esfera en las 20 últimas iteraciones b) Función esfera completa c) Función Ackley en las últimas 20 iteraciones d) Función Ackley completa e) Función camel en las 20 últimas iteraciones f) Función Camel completa g) Función Beale en las 20 últimas iteraciones f) Función Beale completa

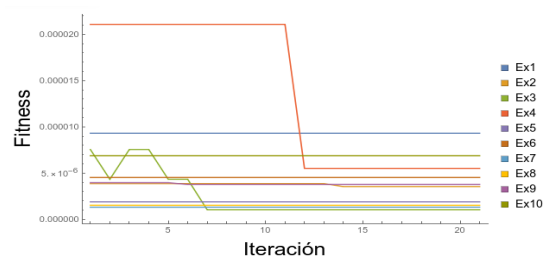


Fuente: Elaboración propia

Figura 4. Variaciones de rewiring en el problema de la esfera con el modelo SWBest

Fuente: Elaboración propia

Los experimentos se realizaron con 30% de probabilidad de *rewiring* de acuerdo con lo sugerido en el modelo base de Watts y Strogatz [27]. Se realizó la prueba modificando ligeramente los valores de probabilidad de *rewiring* en el problema del agente viajero con la mejor estrategia (SW Best). Los resultados obtenidos muestran un mejor desempeño al 30%, siendo consecuente con el modelo original.

Figura 5. 10 experimentos con una misma población, pero diferentes redes de mundo pequeño

Fuente: Elaboración propia

En el modelo propuesto, se hace una asignación aleatoria de un individuo de la población con nodo de la red de mundo pequeño. Se quería descartar que el emparejamiento aleatorio no tuviera algún impacto en la calidad de las soluciones. Para esto, se generó una población inicial para el problema de la esfera usando la estrategia SW Best, y se varió en diferentes experimentos la red inicial. En la Figura 5 se puede observar como la diferencia no es significativa y no tiene mucho impacto en la solución final encontrada.

Tabla 2. Promedio de mejor individuo encontrado de los modelos propuestos en 4 funciones de optimización

	Evolutivo	SW	SW Best	SW HD
Función esfera	0.000075801	0.000030653	0.00002535	0.000031124
Función Ackley	0.140478911	0.080440911	0.070665037	0.086664022
Función Beale	0.06055252	0.017969451	0.015658185	0.021105568
Función Camel	0.003503731	0.001966979	0.001672193	0.006147876

Fuente: Elaboración propia**Tabla 2.** Promedio de mejor individuo encontrado de los modelos propuestos en 4 funciones de optimización

	Evolutivo	SW	SW Best	SW HD
Función esfera	4	2	1	3
Función Ackley	4	2	1	3
Función Beale	4	2	1	3
Función Camel	3	2	1	4
Rankings	R1=15	R2=8	R3=4	R4=13

Fuente: Elaboración propia

Se realizó también la prueba Friedman para algoritmos evolutivos [30]. La prueba consiste obtener el promedio de los mejores individuos en 50 iteraciones de cada modelo en cada función de prueba de optimización usada en este trabajo, los cuales se puede observar en la Tabla 2. Los puntajes son transformados a datos ordinales para poder aplicar la prueba de Friedman siguiendo las siguientes reglas: El dato más bajo es de una fila se le asigna el número 1, el siguiente más bajo se asigna el número 2, luego el número 3 y finalmente el número más alto se asigna el número 4. Al aplicar la prueba con los datos de la Tabla 3 se obtiene un $p = 0.011$, que es lo suficientemente aceptable para rechazar la hipótesis nula y deducir que las distribuciones de los modelos no son iguales. Esta deducción nos permite avanzar en la idea de la diferencia de cada modelo y la variabilidad de cada uno. Todas las implementaciones y pruebas pueden encontrarse en un repositorio [31] donde está la guía de uso y demás configuraciones.

Conclusiones

Es posible obtener redes con características de mundo pequeño por medio de las dinámicas de las relaciones entre los individuos de algunos algoritmos evolutivos. Estas relaciones de descendencia padre/hijo modeladas como grafos se asemejan al modelo Watts–Strogatz.

El modelo de control por redes de mundo pequeño sobrepasa la estrategia original en todos los experimentos realizados. A falta de formalismos y demostraciones teóricas del modelo y sus resultados, no es posible asegurar

que este comportamiento sea el mismo para otros problemas de optimización. Sin embargo, los resultados experimentales dan indicios de buen desempeño para continuar la investigación en esta área.

La alternativa de selección de emparejamiento de vecinos con mayor grado tiende a ser el peor modelo, muchas veces peor que la estrategia original. No se está teniendo en cuenta las propiedades topológicas de la red para guiar el mecanismo de control, razón por la cual se entiende que no aporte información relevante para la evolución del sistema.

En todos los experimentos realizados en este trabajo la alternativa de selección de emparejamiento de vecinos con mejor desempeño (fitness) obtuvo los mejores resultados. Esta selección promueve la competencia por medio del factor de evaluación (el fitness que se quiere mejorar) entre los individuos de la población, lo que podría explicar este comportamiento.

Finalmente, se concluye que la propagación de la información a través de una red de mundo pequeño se hace de forma paralela con todos los individuos de la población en cada iteración. La red de mundo pequeño en este caso sirvió para distribuir de forma simultánea las características de los mejores individuos en todos los individuos de la red., lo cual parece mejorar el desempeño de la población y calidad de la solución final encontrada.

Referencias bibliográficas

1. L. Fogel, A. Owens, and M. Walsh, *Artificial intelligence through simulated evolution*. Chichester, WS, UK: Wiley, 1966.
2. D.B. Fogel, "Evolutionary Computation: Toward a New Philosophy of Machine Intelligence", *IEEE Press*, 1995.
3. M. Raschip, M. Breaban, and C. Croitoru, "Evolutionary Computation in Constraint Satisfaction", vol. 02, 2010.
4. R. Caponetto, "Chaotic sequences to improve the performance of evolutionary algorithms", *IEEE Transactions on Evolutionary Computation*, vol. 7, pp. 289–304, 2003.
5. M.V.D.K. Tasoulis, V.P. Plagianakos, "Clustering in evolutionary algorithms to efficiently compute simultaneously local and global minima", *Proceedings of the IEEE Congress on Evolutionary Computation*, 2005.
6. A. E. Eiben, R. Hinterding, and Z. Michalewicz, "Parameter control in evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2, pp. 124–141, July 1999.

7. I. Zelinka, D. Davendra, V. Snasel, R. Jasek, R. Senkerik, and Z. Oplatkova, "Preliminary investigation on relations between complex networks and evolutionary algorithms dynamics", *2010 International Conference on Computer Information Systems and Industrial Management Applications, CISIM*, pp. 148–153, 2010.
8. S. N. Dorogovtsev and J. F. F. Mendes, *Evolution of Networks: From Biological Nets to the Internet and WWW (Physics)*, New York, NY, USA: Oxford University Press, Inc., 2003.
9. I. Zelinka, D. Davendra, S. Roman, and J. Roman, "Do Evolutionary Algorithm Dynamics Create Complex Network Structures?", *Complex Systems*, vol. 20, n°. 2, pp. 127–140, 2011.
10. I. Zelinka, "Investigation on relationship between complex networks and evolutionary algorithms dynamics", *AIP Conference Proceedings*, vol. 1389, pp.1011–1014, 2011.
11. I. Zelinka, "IWCFTA 2012 Keynote Speech III on Close Relations of Evolutionary Dynamics, Chaos and Complexity", *IEEE Xplore*, 2012..
12. I. Zelinka, D. D. Davendra, M. Chadli, and R. Senkerik, "Evolutionary Dynamics as The Structure of Complex Networks", *Springer, Berlin, Heidelberg*, pp. 215–243, 2013.
13. I. Zelinka, D. Davendra, J. Lampinen, R. Senkerik, and M. Pluhacek, "Evolutionary algorithms dynamics and its hidden complex network structures", *IEEE Congress on Evolutionary Computation, CEC*, 2014.
14. P. K. Storn, "Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces", *Technical Report TR-95-012, ICSI*, 1995.
15. V. M. Plagianakos., and K. Tasoulis, "A review of major application areas of differential evolution", *Advances in differential evolution, vol143 of studies in computational intelligence. Springer, Berlin*, 2008.
16. S. P. Qin AK, "Self-adaptive differential evolution algorithm for numerical optimization.", *Proceedings of the IEEE congress on evolutionary computation*, 2005.
17. J. Lampinen., and I. Zelinka, *Mechanical engineering design optimization by differential evolution, New ideas in optimization*. New York: McGraw-Hill, 1999.

18. S. N. Dorogovtsev and J. F. F. Mendes, "Evolution of networks", *Advances in Physics*, vol. 51, n°. 4, pp. 1079–1187, 2002.
19. L. A. N. Amaral, A. Scala, M. Barthelemy, and H. E. Stanley, "Classes of small-world networks", *Proceedings of the National Academy of Sciences*, vol. 97, n°. 21, pp. 11 149–152, September 2000.
20. I. Zelinka, "SOMA — Self-Organizing Migrating Algorithm.", Berlin, Heidelberg: Springer Berlin Heidelberg, 2004.
21. P. K. Storn R, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces", *J Glob Optim*, 1997.
22. I. Zelinka, L. Tomaszek, and V. Snasel, "On Evaluation of Evolutionary Networks Using New Temporal Centralities Algorithm," *2015 International Conference on Intelligent Networking and Collaborative Systems, IEEE INCoS 2015*, 2015.
23. L. Sheng, F. Chen, and H. Wang, "A Review on complex network dynamics in evolutionary algorithm," *IEEE Xplore*, 2016.
24. Yang Yinghui, Li Jianhua, Shen Di, Nan Mingli, Cui Qiong, "Evolutionary dynamics analysis of complex network with fusion nodes and overlap edges", *Journal of Systems Engineering and Electronics*, 2018
25. C. Pizzuti, "Evolutionary Computation for Community Detection in Networks: A Review" ,*IEEE Transactions on Evolutionary Computation*, 2017
26. Yang Liu, Xi Wang, Jürgen Kurths, "Framework of Evolutionary Algorithm for Investigation of Influential Nodes in Complex Networks", *IEEE Transactions on Evolutionary Computation*, 2019
27. D. J. W. S. H. Strogatz, "Collective dynamics of small-world networks", *Nature*, 1998.
28. T. Back, "Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms", *J Glob Optim*, 1995.
29. C. Vanaret, J.-B. Gotteland, N. Durand, and J.M. Alliot, "Certified global minima for a benchmark of difficult optimization problems," *Mathematics, Computer Science*, 2014.

30. S. Picek, M. Golub., and D. Jakobovic "On the Analysis of Experimental Results in Evolutionary Computation", *Proceedings of the 35th International Convention MIPRO*, 2012
31. J. Triana, V. Bucheli, Redes complejas en computación evolutiva (repositorio)., 2020. Disponible en: <https://github.com/jodatm/RedesComplejasComputacionEvolutiva>