

TÉCNICAS EVOLUTIVAS PARA EL PROBLEMA DEL VIAJANTE

Jessica A. Carballido, Ignacio Ponzoni, Nélide B. Brignole

Grupo de Investigación y Desarrollo en Computación Científica
Departamento de Ingeniería y Ciencias de la Computación
Universidad Nacional del Sur
Av. Alem 1253 – 8000 - Bahía Blanca
ARGENTINA

Grupo de Ingeniería de Sistemas de Procesos
Planta Piloto de Ingeniería Química (PLAPIQUI)
Universidad Nacional del Sur - CONICET
Complejo CRIBABB – Camino La Carrindanga km 7 – CC 717 - Bahía Blanca
ARGENTINA

jcarballido@plapiqui.edu.ar ; ip@cs.uns.edu.ar ; dybrigno@criba.edu.ar

Palabras clave: Problema del Viajante, Algoritmos Evolutivos, MCMP, Criterio de Convergencia.

Resumen. *El Problema del Viajante consiste en encontrar la ruta de mínimo “costo” para un viajante que debe visitar cada ciudad en una lista exactamente una vez y regresar al punto de partida. Nuestra experimentación previa con seis técnicas evolutivas basadas en representación ordinal y de permutaciones, ambas combinadas con los siguientes operadores de cruzamiento: Single Crossover Per Couple (SCPC), Multiple Crossover Per Couple (MCPC) y Multiple Crossover with Multiple Parents (MCMP), evidenció un bajo desempeño de la tercera forma de cruzamiento. En este trabajo se analiza la causa de tal comportamiento en base a una modificación adecuada en el criterio de convergencia. Se diseñó una nueva condición de terminación y se la aplicó con MCMP para varios casos de estudio. De los resultados se concluye que la fuerte presión selectiva del MCMP provoca la convergencia prematura del proceso evolutivo.*

1 INTRODUCCION

Las Técnicas Evolutivas (TEs) incluyen una gran variedad de métodos que establecen una analogía entre las posibles soluciones a un problema y los individuos de una población. Tal como lo explica Holland¹:

"Se pueden encontrar soluciones aproximadas a problemas de gran complejidad computacional mediante un proceso de evolución simulada".

Para realizar tal proceso se definen: una representación para los individuos, una función de evaluación de la calidad (*fitness*) de los mismos basada en la función objetivo del problema, así como también operaciones de selección, cruzamiento y mutación. La selección promueve la reproducción mediante el cruzamiento de aquellos individuos de mejor calidad.

Según Michalewicz², los métodos que conforman las TEs clasifican en: Algoritmos Genéticos (AGs), Algoritmos Evolutivos (AEs), programación evolutiva, estrategias evolutivas y programación genética. Todos estos enfoques se caracterizan por encontrar soluciones que aproximan al óptimo en tiempos de cómputo reducidos. Un AE mantiene las características fundamentales del AG, pero difiere o bien en la representación de los individuos o en las operaciones de cruzamiento y mutación.

En este trabajo se presenta el análisis de los resultados obtenidos a partir de un nuevo criterio de convergencia para AEs aplicados al Problema del Viajante (TSP, por sus siglas en inglés). En la sección 2 se realiza una descripción del TSP y se vincula tal problema con los AEs. Las implementaciones realizadas con anterioridad a este trabajo y sus resultados se explican en la sección 3. Luego, en la sección 4, se describen brevemente los AEs implementados para TSP y se presenta el nuevo criterio de convergencia. A continuación se detallan los casos de estudio junto con los resultados obtenidos y finalmente se resumen las principales conclusiones.

2 EL PROBLEMA DEL VIAJANTE

En el TSP se desea visitar todas las ciudades de un territorio exactamente una vez y regresar al punto de partida. En el caso del TSP asimétrico, este territorio está representado mediante un grafo dirigido cuyas aristas poseen un costo. El objetivo es concretar el recorrido de todos los nodos del grafo de forma tal que su costo total sea mínimo.

El interés en abordar el TSP mediante técnicas evolutivas se fundamenta en dos razones principales. La primera es que el problema es fácil de describir, pero difícil de resolver. Se ha demostrado que es NP-Completo³, es decir que no existe ningún algoritmo capaz de encontrar una solución en tiempo polinomial. La segunda razón es que el TSP es aplicable a una gran variedad de problemas reales de ruteo y scheduling. Por estos motivos, en las últimas décadas surgieron diferentes métodos alternativos para encontrar soluciones que aproximan a la óptima, entre los que se encuentran los AEs^{4,5}.

En Carballido *et al.*⁶ se presentan distintas variantes de AEs basadas en dos representaciones genéticas (permutaciones y decodificadores) para resolver el TSP asimétrico. Dichas variantes surgen de la aplicación de los operadores denominados "Múltiples Cruzamientos por Pareja" y "Múltiples Cruzamientos entre Múltiples Parejas" (MCPC y MCMP, por sus siglas en inglés) presentados en Esquivel *et al.*^{7,8}. Estos nuevos métodos de

recombinación son promisorios por diferentes motivos. MCPC refuerza la explotación de características de buenas soluciones halladas previamente, mientras que MCMP provee un balance satisfactorio entre explotación y exploración. Este equilibrio se logra por dos razones: el espacio de búsqueda se explota más eficientemente con la múltiple aplicación del cruzamiento y el mayor número de muestras provisto por los múltiples padres mejora la exploración.

Sin embargo, en los resultados obtenidos se evidenció un pobre desempeño del MCMP para ambas representaciones, lo cual derivó en la hipótesis de que el mismo presenta problemas de convergencia prematura (Carballido *et al.*⁶). El objetivo de este trabajo es analizar la validez de esta hipótesis mediante la implementación de un nuevo criterio de terminación. Esto permitirá dilucidar si el problema realmente radica en una excesiva presión selectiva por parte del MCMP, o en un número fijo e insuficiente de generaciones para la evolución.

3 ALGORITMOS EVOLUTIVOS PARA EL TSP

3.1 Representaciones genéticas y operadores de cruzamiento

La representación más natural para un tour es la denominada de permutaciones (*path representation*). En este caso las n ciudades que conforman el tour se colocan en una lista ordenada de n elementos, de tal forma que si la ciudad i esta en la posición j , i es visitada en j -ésimo lugar.

Hay básicamente tres métodos de cruzamiento definidos⁹ para esta representación denominados “Cruzamiento con Mapeo Parcial”, “Cruzamiento de Orden” y “Cruzamiento Cíclico” (PMX, OX y CX, respectivamente, por sus siglas en inglés). Para nuestras implementaciones trabajamos con OX¹⁰ debido a que en experimentos realizados¹¹ se demostró que para el TSP tiene una performance un 11% mejor que PMX y 15% mejor que CX.

El OX construye a los hijos eligiendo una subsecuencia del tour de uno de los padres, a la vez que se preserva el orden relativo de las ciudades del otro padre. Esta forma de cruzamiento explota la propiedad de la representación de caminos que dice que lo principal es el orden de las ciudades y no sus posiciones.

Por otra parte, la representación ordinal define al tour con una lista de n ciudades, en la que el elemento i -ésimo es un número en el rango $[1..n-i+1]$. A esta lista se la llama decodificador. La principal ventaja de esta representación es que los operadores clásicos de cruzamiento y mutación no necesitan ser redefinidos.

Independientemente del método de cruzamiento utilizado, la forma general de realizarlo consiste en seleccionar dos padres, aplicar una vez el método y generar dos hijos. A esta forma de cruzamiento se la ha denominado SCPC. Buscando mejores resultados, surgieron nuevas alternativas que dieron lugar, entre otras, a las técnicas MCPC⁶ y MCMP⁷.

3.2 MCMP

En MCMP se genera un pool de parejas pp a partir de la población $p1$, donde cada integrante del mismo se selecciona con un criterio de “Selección Proporcional de *Fitness*” (FPS, por sus siglas en inglés). Esta sub-población tiene la mitad de la cantidad de individuos de $p1$. Se calcula el *fitness* promedio de cada uno de los elementos del pp y de allí se seleccionan dos parejas, aplicando nuevamente FPS. De esta forma se obtienen cuatro padres los cuales se cruzan para obtener 4 hijos, tal como se muestra en la Figura 1.



Figura 1: Combinación de cuatro padres

Estos hijos no pasan directamente a la población sino que, de acuerdo a su *fitness*, se eligen los dos mejores y estos son los insertados. A continuación explicaremos brevemente como se realiza el cruzamiento de acuerdo al tipo de representación utilizada.

DECODIFICADORES (CLÁSICO DE UN PUNTO)

En el caso de emplear decodificadores, es posible representar gráficamente como quedan conformados los hijos. Como se ve en la Figura 2, el cruzamiento clásico de un punto construye a cada hijo asignándole las ciudades a un lado del punto de corte de uno de los padres junto con las del lado opuesto del otro padre. Para cuatro padres, la idea básica es la misma que con dos, pero se tiene en cuenta la combinación mostrada en la Figura 1 para elegir cuales son los que corresponden para armar cada hijo.

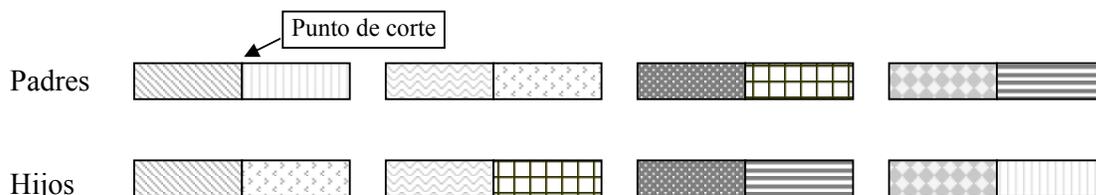


Figura 2: Combinación de cuatro padres para cruzamiento clásico de un punto.

PERMUTACIONES (OX)

OX construye cada hijo eligiendo un subtour de uno de los padres, a la vez que se preserva el orden relativo de las ciudades del otro padre. Un subtour es una sucesión de ciudades que se encuentran entre dos puntos de corte. En el caso de cuatro padres, se procede tomando para $h1$ el subtour de $p1$, y las ciudades de $p2$ que no forman parte del anterior. Para $h2$, el subtour de $p2$ y las ciudades de $p3$. Para $h3$, el subtour de $p3$ y las ciudades de $p4$ y finalmente, para $h4$, el subtour de $p4$ y las ciudades de $p1$.

3.3 Nuevo criterio de convergencia

La condición de terminación aquí propuesta se basa en el concepto de esquemas. Un esquema es una plantilla que permite establecer similitudes entre los cromosomas. Para el caso de representación binaria, un esquema se representa mediante una cadena de símbolos del conjunto $\{0,1,\#\}$, donde $\#$ es un *wildcard*. Por ejemplo, la cadena 011001 es una instancia del esquema 01##0#. Si los posibles valores para los alelos varían entre 1 y N, entonces el conjunto de símbolos para el esquema es $\{1, 2, \dots, N, \#\}$.

De acuerdo con la propiedad enunciada por Radcliffe¹¹, si dos padres son instancias de un mismo esquema, el hijo también será una instancia de ese esquema. En particular, si un esquema acarrea alto *fitness* a sus instancias, la población tenderá a converger sobre los bits definidos en ese esquema. Una vez que converja, todos los vástagos serán instancias de ese esquema. Luego, si el método que resuelve el problema está eficazmente implementado, la solución será una instancia de dicho esquema.

Por esta razón, nuestro criterio de convergencia analiza los genotipos de los individuos hasta que un alto porcentaje de ellos sea instancia de un mismo esquema. A continuación, se explica mediante un algoritmo el criterio de terminación.

Entrada: población p de m individuos con n genes;
porcentaje α de individuos que coinciden en un β por ciento de genes.
Información Auxiliar: vector de frecuencias c de longitud n

- 1- Inicializar i en 1.
- 2- Analizar el i -ésimo alelo de toda la población, es decir desde $p[1, i]$ hasta $p[m, i]$, obteniendo la cantidad c_i de veces que aparece el valor de alelo que más veces se repite en p .
- 3- Incrementar i .
- 4- Si i es mayor que n {lo cual implica ya se recorrieron todos los genes}
 ir al paso 5
 sino
 ir al paso 2
- 5- Contar cuantos elementos de c corresponden a porcentajes mayores que α
- 6- Si esa cantidad representa un porcentaje superior a β
 terminar (se satisfizo el criterio de convergencia)
 sino
 continuar con el AG

Este control se realizó para cada generación. El algoritmo se aplicó en ambas representaciones genéticas con las siguientes contemplaciones:

DECODIFICADORES

En este caso no se utilizaron los alelos del genotipo en forma directa. En cambio, se realizó una transformación intermedia con la cual el decodificador fue convertido al camino correspondiente, considerándose la ciudad número 1 como punto de partida. Los alelos de este nuevo "genotipo" son los que analiza el algoritmo de control de convergencia.

PERMUTACIONES

El procedimiento es equivalente al definido para decodificadores, con la diferencia que el paso de decodificación no es necesario. Sólo se rotaron las ciudades preservando el orden relativo entre las mismas, de forma tal que la ciudad inicial fuese siempre la número 1. Estas modificaciones se introdujeron para considerar iguales a aquellos individuos que representan el mismo tour, pero partiendo de distintas ciudades.

4 EXPERIMENTOS Y RESULTADOS

Se implementó un algoritmo auxiliar de Búsqueda Exhaustiva (BE) para obtener óptimos de referencia en instancias pequeñas del problema. Este algoritmo utiliza la misma función de *fitness* que el AE, aplicándola a todo el espacio de búsqueda.

Los casos de estudio se dividieron en dos grupos: tres grafos (*c1*, *c2* y *c3*) que representan tours de 8 ciudades, y tres grafos (*c4*, *c5*, *c6*) que definen tours de 10 ciudades. Los grafos se representaron mediante matrices con costos aleatorios para las aristas. Para ambos grupos se obtuvieron los óptimos exactos a partir del algoritmo de BE.

En todos los casos se utilizaron las mismas probabilidades de cruzamiento y mutación: 0.7 y 0.25 respectivamente. El tamaño de la población fue de 50 individuos para el primer grupo y de 100, para el segundo.

Las variables que se analizaron son:

- EBest: diferencia entre el valor óptimo conocido o estimado y el mejor valor obtenido, dividida por el valor óptimo.
- GBest: generación en que se obtiene el mejor valor .
- GOpt: generación en que se obtiene el valor óptimo.
- GConv: generación en la cual el algoritmo converge.
- Hit Ratio: cantidad de veces que se obtiene el óptimo.

Los valores reportados son el promedio de los valores obtenidos en 100 corridas de cada uno de los algoritmos. Los resultados para cada uno de los grupos se muestran en las Tablas 1-4.

DECODIFICADORES

Tabla 1: Valores promedio correspondientes a corridas con el primer grupo de grafos

| | <i>c1</i> | <i>c2</i> | <i>c3</i> |
|-----------|-----------|-----------|-----------|
| Hit Ratio | 26% | 40% | 50% |
| GBest | 3,71 | 3,98 | 5,1 |
| GOpt | 3,364 | 4,95 | 6,46 |
| GConv | 11,66 | 11,8 | 19,2 |
| EBest | 0,0111 | 0,0104 | 0,012 |

Tabla 2: Valores promedio correspondientes a corridas con el segundo grupo de grafos

| | <i>c4</i> | <i>c5</i> | <i>c6</i> |
|-----------|-----------|-----------|-----------|
| Hit Ratio | 18% | 20% | 18% |
| GBest | 4,306 | 4,714 | 3,918 |
| GOpt | 4,875 | 6 | 4,111 |
| GConv | 6,94 | 7,14 | 6,24 |
| EBest | 0,013 | 0,02 | 0,0141 |

PERMUTACIONES

Tabla 3: Valores promedio correspondientes a corridas con el primer grupo de grafos

| | <i>c1</i> | <i>c2</i> | <i>c3</i> |
|-----------|-----------|-----------|-----------|
| Hit Ratio | 66% | 38% | 66% |
| GBest | 6,042 | 5,187 | 11,25 |
| GOpt | 6,531 | 6,737 | 14,71 |
| GConv | 19,86 | 15,64 | 30,18 |
| EBest | 0,0004 | 0,0006 | 0,0009 |

Tabla 4: Valores promedio correspondientes a corridas con el segundo grupo de grafos

| | <i>c4</i> | <i>c5</i> | <i>c6</i> |
|-----------|-----------|-----------|-----------|
| Hit Ratio | 30% | 18% | 84% |
| GBest | 12,96 | 7,49 | 8,26 |
| GOpt | 20,14 | 11 | 11,62 |
| GConv | 23,82 | 16,88 | 20,07 |
| EBest | 0,0006 | 0,002 | 0,0008 |

Comparando los resultados de Hit Ratio y EBest para cada grupo, se aprecia que en general la representación con permutaciones logra un mejor desempeño. También se observa, a partir de los valores de GConv, que los AE con decodificadores poseen una mayor velocidad de convergencia. En la Tabla 5 se resumen los promedios obtenidos en los seis casos de estudio:

Tabla 5: Promedios generales

| | <i>Decod.</i> | <i>Permut.</i> |
|-----------|---------------|----------------|
| Hit Ratio | 28,66% | 50,33% |
| Gbest | 4,68 | 8,53 |
| Gopt | 4,96 | 11,79 |
| Gconv | 10,5 | 21,07 |
| Ebest | 0,0134 | 0,0008 |

Como se puede observar, todas las métricas evaluadas evidencian un mejor desempeño del AE con permutaciones y MCMP. No obstante, en ambos casos la convergencia se alcanza con pocas generaciones. Por otro lado, la representación con permutaciones tiene un Hit Ratio aceptable y un muy buen valor promedio de EBest.

4 CONCLUSIONES

En este trabajo se presentó un nuevo criterio de convergencia para AEs aplicados al problema del viajante. Estos algoritmos utilizan representación de caminos y ordinal para los tours, combinados con MCMP. Nuestro objetivo fue corroborar una hipótesis anteriormente planteada sobre la convergencia prematura de este último operador. El desempeño de los AEs fue evaluado mediante su aplicación a distintos casos de estudio correspondientes a grafos de 8 y 10 ciudades.

Los resultados obtenidos revelaron que, en ambas representaciones, los algoritmos convergen en una reducida cantidad de generaciones. Por lo tanto, puede concluirse que la fuerte presión selectiva del MCMP efectivamente deriva en una convergencia prematura del proceso evolutivo. En consecuencia, no se recomienda aplicar MCMP al TSP. De acuerdo con nuestra experimentación (Carballido *et al.*⁵), convendría emplear en cambio la opción MCPC.

Asimismo, se observó que la representación de caminos exhibe un desempeño muy superior a la ordinal, con un excelente acercamiento al valor óptimo. Esto permite inferir que la implementación de caminos con MCMP podría constituir una opción satisfactoria para otras aplicaciones. Por ejemplo, sería útil para metodologías híbridas donde el AE se emplea exclusivamente como estrategia de inicialización, ya que en estos casos es suficiente contar con una “buena” solución aproximada, aún cuando no sea exactamente la óptima.

5 REFERENCIAS

- [1] Holland, John H., *Adaptation in Natural and Artificial Systems*.
- [2] Michalewicz Z., *Genetic Algorithms + Data Structures = Evolutions Programs*, Springer-Verlag, Third, Extended Edition, 1996
- [3] Garey, M. and Jonson, D., *Computers and Interactability*, W. H. Freeman, San Francisco, 1979.
- [4] Greffenstette, J. J., Gopal, R., Rosmaita, B. and Van Gucht, D., *Genetic Algorithm for the TSP*, Laurence Erlbaum Associates, Hillsdale, NJ, 1985.
- [5] Jog, P., Suh J. Y. y Gucht, D. V., *The Effects of Population Size, Heuristic Crossover, and Local Improvement on a Genetic Algorithm for the Traveling Salesman Problem*, *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers, San Mateo, CA, 1989.
- [6] Carballido, J. A., Ponzoni, I. y Brignole, N. B., *MCPC and MCMP Evolutionary Algorithms for the TSP*, V ASAI, XXXII JAIIO, Bs.As., Argentina, septiembre 2003.
- [7] Esquivel, S. C., Leiva A. y Gallard R. H., *Multiple Crossovers Per Couple in Genetic Algorithms*, Grupo de Interés en Sistemas de Computación, UNSL, Argentina.
- [8] Esquivel, S. C., Leiva A. y Gallard R. H., *Multiple Crossovers between Multiple Parents to improve Search in Evolutionary Algorithms*, Proyecto UNSL-338403, UNSL,

Argentina.

- [9] Davis L., Applying Adaptive Algorithms to Epistatic Domains, Proceedings of the International Joint Conference on Artificial Intelligence, pp. 162-164, 1985.
- [10] Oliver, I. M., Smith, D. J. and Holland J. R. C., A Study of Permutation Crossover Operators on the Travelling Salesman Problem, Laurence Erlbaum Associates, Hillsdale, NJ, 1987.
- [11] Radcliffe N. J., Equivalence Class Analysis of Genetic Algorithms, Complex Systems 5, 1991.