

CONCLUSIONES GENERALES

Esta tesis presenta aplicaciones pertenecientes a las áreas ingenieriles, científicas e incluso administrativas (ej. redes, placas de circuitos impresos, planificación y programación de actividades en proyectos de desarrollo de sistemas), con características propias de los problemas de optimización combinatorial NP-Complejos, especialmente relacionados con el problema del viajante de comercio (TSP). Como este problema es un miembro de la familia de los problemas NP-duros, los cuales no poseen un algoritmo que halle una solución en tiempo polinomial. Consecuentemente es posible no encontrar el óptimo global en algunas aplicaciones, o saber si la solución hallada es un óptimo global. En muchos casos puede ser preferible encontrar una muy buena solución en tiempo razonable, aunque esta no sea un óptimo global.

Los algoritmos tradicionales utilizados para solucionar los problemas de optimización combinatorial requieren un gran conocimiento previo de cómo evaluar el fitness (función de costo, superficie de respuesta, etc.). Este es el caso de la búsqueda basada en el gradiente que requiere una función de costo diferenciable. En cambio, los algoritmos evolutivos no necesitan conocer previamente el dominio del problema. Sólo necesita ordenar de acuerdo a un rango, a dos soluciones competidoras, esto es, determinar cuál solución es la más adecuada. Permitiendo de esta manera, que una amplia variedad de problemas de difícil tratamiento en el campo de la ingeniería puedan resolverse por medio de la opción evolutiva. En general, los algoritmos evolutivos son capaces de solucionar problemas que las técnicas numéricas no logran resolver.

Otra ventaja que ofrece la alternativa evolutiva es la capacidad para adaptarse a situaciones cambiantes. En los procedimientos de optimización tradicionales, el cálculo debe iniciarse desde cero si una de las variantes del problema cambia, lo que resulta desde el punto de vista computacional muy costoso. Con un algoritmo evolutivo la población final de soluciones actúa como una reserva de conocimiento que puede aplicarse a ambientes dinámicos. Sin necesidad de iniciar nuevamente el cálculo.

Básicamente los algoritmos evolutivos simulan la evolución de una población de soluciones para un problema dado. Dicho proceso consiste en: la evaluación de todos los individuos de la población, la selección de una nueva población intermedia, y el intercambio de su código genético. Estos algoritmos, son una clase de métodos de búsqueda de propósito general que combinan elementos de la búsqueda estocástica y dirigida; remarcando, así, el balance entre la exploración y la explotación del espacio de búsqueda. Cabe destacar que los algoritmos evolutivos pueden considerarse eficientes en comparación con otros métodos estocásticos de búsqueda ciega, pero en general: si se encuentra un método heurístico para resolver un problema específico este siempre lo hará mucho más eficientemente que un AE. En la práctica, es fundamental para el buen desempeño de un AE, controlar continuamente la diversidad de la población. No es conveniente poseer poca variedad de aptitudes ya que en tal caso todos los individuos tendrían más o menos las mismas posibilidades de sobrevivir, la selección reproduciría la situación anterior y todo el peso de la búsqueda recaería en los operadores genéticos, y acabaría siendo poco más que una búsqueda aleatoria. Además la población sería copada por superindividuos, que sólo son los más aptos en cierto momento, pero no las más aptos absolutos; provocando, en general, una *convergencia prematura* del AE, habitualmente hacia un subóptimo. Uno de los puntos clave en estos algoritmos es la codificación de la solución a un problema en un cromosoma.

Los AGs convencionales involucran la selección de dos individuos padres para crear a lo sumo dos cromosomas hijos. Pero la necesidad de mejorar la performance de los AGs en problemas de alta complejidad, ha llevado a Eiben, con su grupo, y a Esquivel, entre otros, reformular los mecanismos de selección y crossover. Tales cambios han dado origen a la multiplicidad de padres, de hijos y la combinación de ambas.

Muchos son los algoritmos, no evolutivos, para resolver el problema del viajante, algunos de estos proveen una solución óptima global para instancias no triviales de TSP con 7397 ciudades. Para lo cual necesitan 4 años de tiempo de CPU sobre una red de computadoras. En esta tesis se presenta como los algoritmos evolutivos, un método heurístico, es capaz de hallar un conjunto de soluciones promisorias velozmente. Los primeros avances para solucionar el TSP,

por medio de Algoritmos Evolutivos, necesitan desarrollar una apropiada representación para codificar un tour, operadores genéticos válidos para mantener los bloques evitando ilegalidad, y prevención de la convergencia prematura. El problema de los algoritmos evolutivos basados en operadores de crossover está en la necesidad de un mayor tiempo computacional, por lo que resultan muy caros. Mientras que los algoritmos basados en operadores de mutación no escapan de forma eficiente de los óptimos locales. Michalewicz propone un algoritmo evolutivo basado en el operador inver-over. El operador incorpora en una solución el conocimiento provisto por otros individuos, mediante la inversión y la recombinación. La inversión es aplicada a una parte de un único individuo, pero la selección del segmento a invertir depende de otros individuos pertenecientes a la población. Este algoritmo presenta las siguientes ventajas:

Es simple y fácil de implementar (menos de 100 líneas de código).

Los resultados experimentales indican que este operador obtiene mejores resultados que todos los otros operadores evolutivos propuestos para TSP (PMX, CX, OX, ER, EER).

Es bastante rápido y la calidad de sus resultados es muy alta.

Es posible hacer las siguientes observaciones:

Es probablemente el algoritmo evolutivo más rápido para el TSP. Todos los otros algoritmos basados en operadores de crossover proveen resultados mucho peores en un tiempo bastante más prolongado.

Tiene sólo 3 parámetros: el tamaño de la población, la probabilidad p de generar una inversión aleatoria, y el número de iteraciones en la condición de terminación; la mayoría de los otros sistemas evolutivos tienen muchos otros parámetros adicionales.

Introduce un nuevo e interesante operador, que combina características de inversión (o mutación) y crossover.

Con una nueva perspectiva a la naturaleza del operador inver-over, en esta tesis se introduce la característica de multiplicidad al mismo. Permitiendo, de esta manera, la donación de información por parte de otros individuos de la población hasta que eventualmente un mejor vástago sea creado y reemplace a S_i en la población actual. Todos los algoritmos mediante los cuales se ha evaluado el TSP (desde IO-1 hasta IO-5), generan permanentemente hijos factibles. Además, al

evaluar la bondad de las soluciones y el escaso tiempo consumido en hallarlas muy cercanas al óptimo, se deduce que: las opciones que incluyen multirecombinación superan al método original de inver-over, y los valores son mejorados al incrementar el número n_l de las operaciones de inver-over.

Resumiendo, en este trabajo se puede observar como la computación evolutiva, un método heurístico, brinda un grupo de diferentes soluciones a distintas instancias estáticas del problema del viajante de comercio, cuyo modelo responde al de aplicaciones en áreas tales como las redes de telecomunicaciones, la planificación de proyectos de desarrollo de cualquier índole, entre otras. No obstante, el hecho de contar con una batería de soluciones permite una rápida y adecuada adaptación de la aplicación ante cualquier modificación de las condiciones del problema. Por lo cual su utilidad puede extenderse fácil y económicamente a entornos dinámicos. Además, las técnicas de la computación evolutiva son accesibles y de fácil manejo para que el personal administrativo y técnico de una organización pueda resolver una amplia gama de situaciones problemáticas.

En función a la bondad de los resultados obtenidos, en trabajos futuros, se aspira a continuar investigando la aplicación del operador inver-over en un algoritmo evolutivo combinado con técnicas de búsqueda local; tales como tabu search, simulated annealing, entre otras. Ya sea tanto para distintas aplicaciones del problema del viajante de comercio como para las de problemas de scheduling por ejemplo, aquellos casos de scheduling con set-up dependiente de la secuencia, o problemas de flow shop.