

Optimización Mono y Multiobjetivo a través de Métodos de Aproximación de Soluciones en Ambientes Estacionarios y No Estacionarios

Susana Esquivel[†], Victoria Aragón[†], Leticia Cagnina[†]

[†]Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC)
Facultad de Ciencias Físicas, Matemáticas y Naturales
Universidad Nacional de San Luis
Ejército de los Andes 950 - Local 106
(5700) - San Luis - Argentina
Tel: (02652) 420823 / Fax: (02652) 430224
e-mail: {esquivel, vsaragon, lcagnina}@unsl.edu.ar

Resumen

En este artículo se describen en forma breve las direcciones de investigación que en la actualidad se están desarrollando dentro de la línea “Optimización Mono y Multiobjetivo” del Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC). Uno de los objetivos principales de esta línea, es el estudio y desarrollo de métodos de aproximación de soluciones para la optimización mono y multiobjetivo en problemas de optimización combinatoria y en espacios continuos. En particular, el énfasis está puesto en los métodos computacionales de aproximación basados en los paradigmas de inteligencia colectiva y biológicos.

1. Introducción

Se puede definir informalmente el concepto de optimización como el proceso de encontrar y comparar soluciones factibles hasta que ninguna solución mejor pueda encontrarse. Estas soluciones son consideradas buenas o malas en función de algún objetivo. Una porción importante de la investigación y aplicación en el área de optimización considera un único objetivo a optimizar aunque gran parte de los problemas del mundo real involucran más de un objetivo. En los problemas en los cuales existen objetivos conflictivos múltiples el proceso de optimización ya no devuelve una única solución mejor sino un conjunto de soluciones óptimas de compromiso, siendo responsabilidad de quién toma las decisiones elegir cuál de ellas es mejor para el problema que desea resolver.

Los métodos clásicos de optimización, en general, devuelven una solución en cada ejecución lo cual hace que dichos métodos no sean los más aptos para resolver este tipo de problemas [6]. Las heurísticas de búsqueda basadas en poblaciones, que representan conjuntos de soluciones, resultan más adecuadas para resolver problemas de optimización multiobjetivo.

Por otra parte, los problemas pueden cambiar en el tiempo y no conocerse nada acerca del momento en que se producen los cambios. En esta situación ¿es conveniente comenzar desde cero o tratar que el algoritmo de optimización pueda, desde el punto en que se encuentra, adaptarse y seguir con el proceso de optimización reutilizando las buenas soluciones encontradas hasta ese momento?. Es claro que este último tipo de problema, no estacionario, es mucho más complejo de resolver puesto que las heurísticas que suelen usarse en los casos estáticos no devuelven buenos resultados en los dinámicos. Nuevamente aquí, las heurísticas de búsqueda basadas en poblaciones parecen ser más aptas, puesto que existe la posibilidad de encontrar una solución de partida, dentro del conjunto de soluciones que mantienen en la población, que sea un buen punto para continuar la búsqueda cuando se produjo un cambio [2].

2. Tareas Concluidas y en Progreso

En los últimos años el eje de investigación de la línea se enfocó en el estudio de problemas de optimización monoobjetivo usando heurísticas pertenecientes al paradigma de la Computación Evolutiva, especialmente algoritmos genéticos multirecombinados con y sin inserción de conocimiento acerca de la naturaleza del problema. La performance de los algoritmos desarrollados se evaluó sobre problemas de optimización combinatoria duros, tales como distintos problemas de scheduling en ambientes estáticos (las referencias pueden ser consultadas en <http://www.lidic.unsl.edu.ar>) y en la optimización de funciones multimodales dinámicas, donde el objetivo fue seguir la trayectoria del óptimo global, tan cercanamente como sea posible, a medida que éste se desplaza sobre el espacio de búsqueda.

En razón de que los algoritmos evolutivos, conforme la población evoluciona, idealmente convergen hacia el óptimo global o hacia alguna solución muy cercana a él, cuando se produce un cambio la población ha perdido gran parte de la diversidad inicial, característica esta que hace necesario complementar dichos algoritmos con técnicas que permitan mantener la diversidad genética de la población. En tal dirección se implementaron diferentes técnicas:

- Creación de nichos, que promueven la formación y mantenimientos de subpoblaciones que exploran áreas distintas del espacio de búsqueda, los métodos usados fueron *clearing*, *sharing* [16] y *algoritmos de clustering* [10]

- Operadores de macromutación, tales como recrudescencia e inmigrantes aleatorios [12], que insertan diversidad o bien aumentando la probabilidad de mutación o reemplazando un porcentaje de la población por individuos generados aleatoriamente, respectivamente.

Las características de los experimentos diseñados y los resultados de tales enfoques (salvo para el caso de clustering que aun no han sido publicados) pueden ser consultados en [1, 19] disponibles en <http://www.lidic.unsl.edu.ar>.

En la actualidad la actividad se concentra en el estudio y desarrollo de dos heurísticas muy promisorias, tales son *Particle Swarm Optimizer (PSO)* y *Artificial Immune Systems (AIS)*.

La heurística *PSO* que pertenece al paradigma de la *Inteligencia Colectiva* y fue inicialmente propuesta por J. Kennedy y R. Eberhart [15] se basa en el comportamiento de las bandadas (poblaciones) de aves (partículas) cuando buscan alimento. Los *swarms* o bandadas son poblaciones de partículas que representan soluciones potenciales de un problema determinado. La búsqueda hacia nuevas soluciones es guiada por la “presión social”, es decir, se basa en el conocimiento de los demás individuos. Los miembros de la bandada se comunican entre sí para direccionar la búsqueda, emulando las técnicas de exploración que usan los insectos o pájaros.

Una partícula usualmente es representada como un vector, donde cada componente se corresponde con una variable del problema y tiene asociado un vector de *fitness*, indicando la aptitud de la misma. También utiliza un vector de *fitness fbest* asociado a cada una, el cual hace referencia al mejor *fitness* encontrado hasta ese momento por ese individuo.

En cada iteración o *ciclo de vuelo* se evalúa la función objetivo para cada partícula con respecto a su posición corriente. También, en cada ciclo se actualizan las partículas en base a dos valores: *pbest* (explotación), el mejor valor alcanzado por la partícula hasta ese momento, y *gbest* (exploración) el mejor valor encontrado por toda la población.

Inicialmente *PSO* fue propuesto para optimización en espacios continuos. Con respecto a la aplicación de esta heurística para problemas de optimización en espacios discretos poco es el trabajo reportado, hasta el momento, en la bibliografía especializada [4, 17]. Por tal razón, aprovechando la experiencia del grupo en temas de scheduling, parte de la investigación de la línea se derivó a estudiar su aplicabilidad a un problema concreto, el de *Scheduling Total Weighed Tardiness* en ambientes de máquina única. Este problema es considerado un problema NP-Hard o duro [13], ya que no se conoce un algoritmo que pueda resolverlo en tiempo polinomial.

En tal sentido, se trabajó sobre diferentes versiones de *PSO*. Se inició el estudio de esta heurística con una versión hibridizada del *PSO* básico al que se lo denominó *HPSO* [3]. En este algoritmo las partículas se representaron como vectores de números reales (para mantener la eficiencia demostrada del algoritmo cuando trabaja sobre espacios continuos) y se utilizó un operador de mutación dinámica para mantener la diversidad en la población.

Una versión posterior de *HPSO* mejoró la performance del primero, a través de la utilización de vecindarios lógicos. Esta versión denominada *HPSOvecin* [18] logró solucionar una falencia que presentaba *HPSO*: la convergencia prematura hacia óptimos locales. Esto se logró gracias a que cada partícula además de estar influenciada por el mejor valor encontrado por ella misma, está influenciada por el mejor valor alcanzado por alguno de sus vecinos. Ésto produce que cada partícula sea afectada por la mejor performance de un grupo más pequeño de partículas (una del vecindario que comparte), y no por la mejor performance de la mejor partícula de la población como sucedía en *HPSO*.

Trabajos posteriores condujeron a una nueva mejora: la inserción de conocimiento específico del problema permitiendo de este modo que el algoritmo ya no realice una búsqueda ciega dentro del espacio de búsqueda sino que ahora guía a las partículas hacia las regiones más prometedoras del mismo. Este algoritmo *HPSO-kn* es el que revela la mejor performance. La inserción de conocimiento

se realiza a través de la inclusión de 3 semillas provistas por tres buenas heurísticas. Las mismas son introducidas en la población inicial de partículas y de esta manera las influyen. Los resultados obtenidos han sido enviados para su publicación y actualmente está en proceso de revisión.

Otra dirección en la cual se trabajó y se obtuvieron muy buenos resultados es en la aplicación de *HPSOvecin* para el problema de optimización de funciones dinámicas [14]. A diferencia de lo que sucedía cuando se trabajó con algoritmos evolutivos la única técnica de mantenimiento de la diversidad en la población utilizada fue un operador de macromutación dinámico.

Al momento se está trabajando en la adaptación del algoritmo para problemas de optimización multicriterio, en primer instancia, para ambientes estáticos y, posteriormente, si los resultados obtenidos son buenos se extendera para su aplicación en ambientes dinámicos.

Por otro lado, con respecto a la heurística *AIS* recientemente ha comenzado a ser estudiada por parte de los integrantes de la línea. Cuando este modelo computacional *AIS* es comparado con otros bien establecidos modelos de la inteligencia computacional como los algoritmos evolutivos, las colonias de hormigas, las redes neuronales, entre otras, puede decirse que la *AIS* está aun en los primeros años de su “infancia”.

Una definición dada por [5] expresa “*AIS* son sistemas adaptativos inspirados en los principios y modelos de la teoría de la inmunología biológica y funciones inmunológicas observables, que son aplicados a la resolución de problemas”.

En particular parte del grupo está interesado en la aplicación de este modelo en problemas de optimización combinatoria para mono y multiobjetivo. Sólo se reportan en la bibliografía especializada dos trabajos para el problema del viajante [7, 8] y otros pocos para problemas de scheduling [9, 11]. En la última referencia los autores reportaron resultados sobre un problema de optimización de dos objetivos para el caso de Flow Shop Scheduling. Al respecto el grupo todavía no reporta resultados, esencialmente se está encarando la etapa inicial de estudio de la temática.

3. Consideraciones finales

Es importante destacar que al momento están en marcha tres Tesis Doctorales asociadas con las temáticas de esta línea de investigación, dos de ellas con Becas otorgadas por el CONICET en el corriente año.

Referencias

- [1] J. Branke. *Evolutionary Optimization in Dynamic Environments*. Kluwer Academic Publishers, USA, 2002.
- [2] Maurice Clerc. Discrete particle swarm optimization illustrated by the traveling salesman problem. <http://www.mauriceclerc.net>, 2000.
- [3] LeandroÑ. de Castro y Jonathan Timmis. *Artificial Immune Systems: A New Computational Intelligence Approach*. Springer, 2002.
- [4] K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley and Sons, LTD, New York, 1st. edition edition, 2001.

- [5] N. Toma et al. The proposal and evaluation of an adaptive memorizing immune algorithm with two memory mechanisms. *Journal of the Japanese Society for Artificial Intelligence*, 15(6):1097–1106, 2000.
- [6] S. Endoh et al. Immune algorithm for n-tsp. In *IEEE Systems, Man and Cybernetics Conference*, pages 3844–3849, 1998.
- [7] T. Fukuda et al. Immune networks using genetic algorithms for adaptive production scheduling. In *15th IFAC World Congress*, pages 57–60, 1993.
- [8] W. Sheng et al. Clustering with niching genetic k-mean algorithm. In *GECCO(2)*, pages 162–173. Springer, 2004.
- [9] X. Cui et al. Study of population diversity of multiobjective evolutionary algorithm based on immune and entropy principles. In *IEEE Congress on Evolutionary Computation*, pages 1316–1321, 2001.
- [10] J. Grenfenstette. Genetic algorithms for changing environments. In *Second Conference on Paralell Problem Solving from Nature*, pages 137–144. North Holland, 1992.
- [11] M. Pinedo. *Scheduling: Theory, Algorithms and System*. Prentice Hall, 1995.
- [12] S. Esquivel y C. Coello Coello. Particle swarm optimization in non-stationary environments. *Lecture Notes in Artificial Intelligence, Springer*, 3515:757–766, November 2004.
- [13] R. Eberhart y J. Kennedy. A new optimizer using particle swarm theory. In *Proceeding of the Sixth International Symposium on Micromachine and Human Science*, page 39, 1995. Nagoya, Japan.
- [14] D. Goldberg D. y J. Richardson. Genetic algorithm with sharing for multimodal function optimization. In Grenfenstette J., editor, *Second International Conference on Genetic Algorithms*, pages 41–49, New Yersey, 1987. Lawrence Erlbaum.
- [15] X. Hu y R. Eberhart. Swarm intelligence for permutation optimization. a case study: n-queens problem. In *Swarm Intelligence Symposium*, Indianapolis, USA, April 2003.
- [16] L. Cagnina y S. Esquivel. Particle swarm optimization para un problema de optimización combinatoria. In *Proceedings of the X Congreso Argentino de Ciencias de la Computación*, pages 1847–1855, La Matanza, Buenos Aires, Argentina, 2004.
- [17] V. Aragón y S. Esquivel. An evolutionary algorithm to track changes of optimum value locations in dynamic environments. *Journal of Computer Science and Technology*, 4(3):127–133, October 2004.
- [18] V. Aragón y S. Esquivel. Nichos y operadores de macromutación para problemas dinámicos de optimización de funciones multimodales no estacionarias. In *X Congreso Argentino de Ciencias de la Computación*, pages 1799–1810, Buenos Aires, Octubre 2004.
- [19] L. Cagnina y S. Esquivel y R. Gallard. Particle swarm optimization for sequencing problems: a case study. In *Congress on Evolutionary Computation*, pages 536–541, Portland, Oregon, USA, 2004.