

SISTEMAS INTELIGENTES DISTRIBUIDOS PARA RESOLVER PROBLEMAS DE OPTIMIZACIÓN

*Hugo Alfonso, Carlos Bermúdez, Natalia Fernández,
Gabriela Minetti, Carolina Salto, Natalia Stark*

Laboratorio de Investigación en Sistemas Inteligentes (LISI)
Departamento de Informática - Facultad de Ingeniería
Universidad Nacional de La Pampa
Calle 110 Esq. 9
(6360) General Pico – La Pampa – Rep. Argentina
e-mail: {alfonsoh, bermudezc, fernaty, minettig, saltoc, stark}@ing.unlpam.edu.ar
Te. / Fax: (02302) 422780/422372, Ext. 6302

Resumen

En la comunidad científica se observa una importante tendencia por resolver problemas de optimización, a través de la utilización de algoritmos tanto exactos como heurísticos. Esto se debe a que nos enfrentamos con nuevos problemas ingenieriles y recursos computacionales, tales como tipos de máquinas, redes y entornos como Internet.

Los problemas de optimización más atacados son los de: asignación de recursos, balance en la línea de ensamblaje, planificación y programación (scheduling), corte y empaquetado de una o varias dimensiones (cutting/packing), etc. Y ellos cuentan con la particularidad de cubrir una importante cantidad de problemas ingenieriles y productivos en nuestra región, y su aplicación puede producir un favorable impacto económico.

El objetivo principal de esta línea de investigación comprende el diseño y desarrollo de algoritmos en entornos distribuidos para la resolución de problemas de creciente dificultad (NP-duros), utilizando técnicas heurísticas y exactas tales como el recocido simulado, búsqueda tabú, búsqueda local y sus posibles hibridaciones, como así también diferentes técnicas relacionadas con los algoritmos evolutivos. De estos algoritmos se analizará la relación eficacia/eficiencia para obtener patrones del comportamiento de los mismos. La implementación en entornos distribuidos se realizará para que a través del paralelismo de las operaciones se puedan alcanzar soluciones de alta calidad rápidamente.

Palabras Claves: Sistemas Paralelos y Distribuidos, Optimización, Métodos Exactos y Heurísticos, Computación Evolutiva.

Introducción

Uno de los principales frentes de trabajo en el ámbito de la informática ha sido tradicionalmente el diseño de algoritmos cada vez más eficientes para la solución de problemas, tanto de optimización como de búsqueda. En este dominio, el objetivo consiste en obtener algoritmos nuevos que necesiten un menor esfuerzo computacional, y en caracterizar su comportamiento para las clases de problemas que demanda la comunidad científica e industrial en general.

Los problemas de optimización más atacados los podemos encontrar en la OR Library, que desde hace algunas décadas se ha encargado de recopilar series de datos para el testeo de los problemas más relevantes o de mayor aplicación. Entre ellos podemos mencionar problemas de: asignación de recursos, balance en la línea de ensamblaje, planificación y programación (scheduling), corte y empaquetado de una o varias dimensiones (cutting/packing), del vendedor viajero (TSP), ruteo de vehículos, etc.

La investigación en algoritmos, tanto exactos como heurísticos, para resolver problemas de optimización tiene una vigencia inusualmente importante en estos días, ya que nos enfrentamos a nuevos problemas de ingeniería al mismo tiempo que contamos con nuevos recursos computacionales tales como tipos de máquinas, redes y entornos como Internet.

En la actualidad son necesarios algoritmos muy eficientes que puedan dar respuesta en tiempo real a problemas del usuario o del sistema, y que permitan trabajar con los nuevos sistemas operativos y máquinas. Por ello, es muy importante poder caracterizar el funcionamiento de dichos algoritmos, y así conocer sus limitaciones y potencia intrínseca al resolver nuevos problemas no estudiados. Esto supone definir el comportamiento de los algoritmos con relación a distintas clases de problemas, sobre instancias de dimensión creciente; y de esta manera deducir los métodos o modificaciones necesarias que permitan generar un motor de búsqueda eficiente.

Dos de las ramas con más éxito para diseñar algoritmos eficientes en la actualidad son: la hibridación y el paralelismo. La hibridación permite incorporar información del problema en el algoritmo de resolución para trabajar en contacto con sus características diferenciadoras. Esto tiene también relación con la posibilidad de involucrar a varios algoritmos en el proceso de búsqueda de una solución de manera más eficiente, ya que usualmente se obtienen reducciones notables en el esfuerzo cuando se utiliza un algoritmo híbrido. Por otro lado, el uso de múltiples procesadores para resolver en paralelo un problema, permite acelerar la búsqueda en tiempo real y alcanzar nuevos dominios de aplicación antes imposibles. Ello contribuirá a la provisión de, al menos, soluciones quasi-óptimas y resultados oportunos en tiempo.

Actualmente se tiende a resolver los problemas de optimización por medio de algoritmos exactos, heurísticos, búsqueda local (recocido simulado, búsqueda tabú, etc) y sus posibles hibridaciones, como así también diferentes técnicas relacionadas con los algoritmos evolutivos (AEs).

Existen versiones de AEs para la toma de decisión bajo criterios múltiples y diferentes arquitecturas paralelas. Los AEs también son aplicables en situaciones dinámicas donde: el objetivo o las restricciones cambian a través del tiempo (exógena o endógenamente), los ajustes de los parámetros y las medidas de la adaptabilidad son continuamente perturbados, y la topología es irregular, discontinua, multimodal o no puede ser abordada por métodos tradicionales.

Temas de Investigación y Desarrollo

- Revisión de algoritmos evolutivos.
- Análisis del estado del arte de otras técnicas heurísticas y exactas tales como el recocido simulado, búsqueda tabú, ramificación y acotación, búsqueda local y sus posibles hibridaciones.
- Investigación de tendencias actuales, problemas usuales y posible uso de Sistemas de Computación Paralelos y Distribuidos.
- Implementación secuencial y distribuida de los algoritmos diseñados.
- Análisis de las ventajas e inconvenientes de cada variante algorítmica en una base de problemas diversa y amplia.
- Selección de problemas con especial interés en los dominios de la optimización combinatoria y telecomunicaciones.
- Obtención de resultados sobre la forma en que se persigue a la solución en cada tipo de variante y sobre cómo afecta a la escalabilidad observada.
- Modelado de los comportamientos observados usando técnicas estadísticas y analíticas matemáticas.
- Estudio de las implicaciones prácticas que se derivan de este desarrollo.

Resultados Esperados

Se espera diseñar y desarrollar algoritmos en entornos distribuidos para la resolución de problemas de creciente dificultad (escalabilidad), enfatizando el uso de técnicas híbridas para incorporar heurísticas específicas a cada problema que puedan beneficiar la búsqueda de soluciones. Es fundamental comprender el funcionamiento básico de los algoritmos, tanto secuenciales como paralelos, en dichas condiciones de dificultad e identificar los puntos que permitan una mejora en eficiencia o precisión.

Se pondrá especial interés en problemas categorizados como NP-duros tales como: problemas de corte y empaquetado de una o varias dimensiones, y en problemas de asignación de recursos y planificación de operaciones, entre otros.

Bibliografía

Adamidis P. Review of Parallel Genetic Algorithms Bibliography (v1.0). Internal T.R., Aristotle University of Thessaloniki, 1994. (http://www.control.ee.auth.gr/~panos/papers/pgs_review.ps.gz).

Alba, E. Troya, J.M. A Survey of Parallel Distributed Genetic Algorithms, Complexity 4(4):31-52, 1999

Alba E., Troya J. M. An Analysis of Synchronous and Asynchronous Parallel Distributed Genetic Algorithms with Structured and Panmictic Islands. In Zomaya A. Y., Ercal F., Olariu S. (eds.), Proceedings of the BioSP3 Workshop (to appear), IPPS/SPDP, Springer-Verlag, 1999.

Bäck, T., Evolutionary Algorithms in Theory and Practice. Oxford University Press, New York, 1996.

Bruns, R., Scheduling. In Th. Bäck, D. B. Fogel, and Z. Michalewicz, editors. Handbook of Evolutionary Computation, chapter F1.5, pages F1.5:1-F1.5:9. Oxford University Press, New York, and Institute of Physics Publishing, Bristol, 1997.

Coulouris, G., Dollimore, J., Kindberg, T., Distributed Systems. Concepts and Design. Addison-Wesley. Second Edition, 1994.

Foster, I., Designing and Building Parallel Programs, Addison Wesley, 1995.

Garey, M. R., Johnson, D. S., Computers and Intractability. A Guide to the Theory of NP-Completeness. Freeman & Co., San Francisco, CA, 1979.

Ingber, L., Simulated Annealing: Practice versus Theory. Mathl. Comput. Modelling, 18 (11), 29-57, 1993.

Michalewicz Z., Esquivel S., Gallard R., Michalewicz M., Tao G, Trojanowski, K. The Spirit of Evolutionary Algorithms, Special Issue on Evolutionary Computing of the Journal of Computing and Information Technology, University Computing Centre, Zagreb, Croatia, pp1-18, 1999.

Michalewicz, Z., Fogel, D.B.. How to Solve It: Modern Heuristics. Springer, 1999

Ochoa, C., Gallard R., Strategies for Migration Overseeing in Asynchronous Schemes of Parallel Genetic Algorithms, Proceedings del Second International ICSC Symposia on Intelligent Industrial Automation and Soft Computing, pp 300-307, ISBN 3-906454-06-1, Ecole pour les Etudes et la Recherche en Informatique et Electronique (EERIE), Nimes, Francia Septiembre 1997

Pacheco, P., A User's Guide to MPI, 1998.

Reeves C. R. (ed.). Modern Heuristic Techniques for Combinatorial Problems. Blackwell Scientific Publications, Oxford, 1993.

Sprave, J., Linear Neighborhood Evolution Strategy. In A. V. Sebald and L. J. Fogel, editors, Proceedings of the Third Annual Conference on Evolutionary Programming, pages 42-51. World Scientific, Singapore, 1994.

Trojanowski K, Michalewicz Z., Searching for Optima in Non-Stationary Environments", Proceedings of the 1999 Congress on Evolutionary Computation (IEEE). Washington DC, pp 1843-1850.