

Diseño de algoritmos evolutivos auto-adaptativos secuenciales y paralelos

Carolina Salto, Natalia Stark, Carlos Bermudez, Gabriela Minetti, Hugo Alfonso, German Dupuy
 Laboratorio de Investigación en Sistemas Inteligentes (LISI)
 Facultad de Ingeniería - Universidad Nacional de La Pampa
 Calle 110 Esq. 9 (6360) General Pico - La Pampa - Rep. Argentina
 Te. / Fax: (02302) 422780/422372, Int. 6302

Resumen

La contribución de esta línea de investigación es el diseño e implementación de algoritmos evolutivos, tanto secuenciales como paralelos, que puedan guiar el cambio de ciertos parámetros usando información del estado del proceso de búsqueda. El objetivo que se persigue es la reducción de los tiempos incurridos en la tradicional definición de una adecuada configuración de parámetros para el problema que se intenta resolver.

CONTEXTO

Esta línea de investigación se desarrolla en el marco del proyecto de investigación "Resolviendo problemas complejos con técnicas metaheurísticas avanzadas", dirigido por la Dra. Carolina Salto, y llevado a cabo en el Laboratorio de Investigación de Sistemas Inteligentes (LISI), de la Facultad de Ingeniería de la Universidad Nacional de La Pampa. Este proyecto mantiene desde hace varios años una importante vinculación con investigadores de la Universidad Nacional de San Luis (Argentina) y de la Universidad de Málaga (España) con quienes se han realizado varias publicaciones conjuntas.

I. INTRODUCCIÓN

Debido a que los algoritmos genéticos (AGs) tratan con un conjunto de soluciones tentativas, conocidas como poblaciones, tiene sentido pensar en dividir la población en un conjunto de subalgoritmos colaborativos. Actualmente la mayoría de los AGs paralelos hallados en la literatura utilizan alguna clase de disposición espacial de los individuos y entonces se paralelizan los conjuntos resultantes en un conjunto de procesadores.

Existen modelos distribuidos donde múltiples hilos de búsqueda exploran el espacio de soluciones en forma concurrente. Estos modelos se denominan *estrategias paralelas Tipo 3* en [7], [8] y *enfoques con múltiples búsquedas* en [9]. Si cada hilo utiliza un procedimiento de búsqueda diferente (un método distinto o configuración diferente de parámetros, por ejemplo) se obtiene una metaheurística paralela heterogénea (MhPH). La utilización de las MhPH permite una mayor diversidad y una exploración más profunda del espacio de búsqueda, lo cual se espera conduzca a mejores soluciones. Luna et al.[14] presentan una taxonomía de MhPH y hacen una reseña del estado del arte de las mismas. De acuerdo con esta taxonomía, una metaheurística paralela homogénea

(hilos de búsqueda con el mismo procedimiento de búsqueda e igual parametrización) es un subtipo de MhPH. La mayoría de los algoritmos genéticos distribuidos propuestos en la literatura pertenecen a esta categoría [1].

En particular, los AGs distribuidos (dAGs) [2], [20] estructuran la población en pequeñas subpoblaciones (islas) relativamente aisladas unas de otras. Los individuos dentro de una subpoblación particular pueden ocasionalmente migrar a otra subpoblación. Una política de migración deberá definir la topología de las islas, cuando ocurren las migraciones, qué individuos se intercambian, la sincronización entre las subpoblaciones y la forma en la que se integran los individuos recibidos en la población destino [3], [20].

Las ventajas de estos métodos provienen de dividir la población en varias subpoblaciones, cada una ejecutando un AG separado, esto permite una amplia exploración de regiones prometedoras del espacio de búsqueda del problema. Una migración esporádica de individuos produce un intercambio de material genético entre las subpoblaciones, que en realidad es uno de los responsables del equilibrio entre exploración y explotación.

Independientemente del tipo de AG distribuido, es muy común en la literatura establecer en forma experimental los parámetros de la migración, los cuales son responsables de la estructura del algoritmo y del esquema de intercomunicación entre las subpoblaciones (política de migración). En consecuencia, esta situación aumenta la complejidad en el uso de esta técnica ya que se deben considerar parámetros adicionales para controlar política de migración. Esto en general adiciona un tiempo extra de definición de parámetros que los autores raramente reportan en sus estudios. Aunque muchos parámetros regulan la migración, la mayoría de ellos son estándares y ampliamente usados, como una topología de intercomunicación en anillo, enviar uno o pocos individuos, reemplazar el peor individuo de la población receptora, y usar comunicación asíncrona para obtener buenos valores de speedup.

Situación similar se presenta en el diseño e implementación de un AG robusto en cuanto a la definición de los parámetros básicos (tipo de operadores genéticos, probabilidades, tamaño de población, etc.). Esto se debe a la dificultad de determinar a priori cuál es el conjunto de valores más adecuado para el problema que se desea optimizar, para obtener buena calidad de soluciones finales. Hay al menos dos opciones

para este problema. La primera es hacer una experimentación extensiva para determinar, de una manera empírica, cuál operador es el que mejor resultados aporta al problema que se intenta resolver. Dicha tarea implica un tiempo extra de configuración, la cual es computacionalmente costosa. La segunda opción es utilizar un mecanismo adaptativo en el cual el AG selecciona el operador a utilizar. Esta última opción es la seguida en la línea de investigación. Muchas opciones se han propuesto para ajustar los parámetros de un AG tal como probabilidad de mutación [6], probabilidad de recombinación [4], [11], operador de recombinación [21], [18], tamaño de población [10], entre otras.

II. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

En esta sección se describen las distintas líneas de investigación y desarrollo que se encuentran activas.

La definición de un periodo de migración apropiado requiere considerar varios valores (desde bajos a altos) siguiendo un estudio similar a un análisis factorial, realizar la experimentación previa correspondiente, para finalmente poder analizar cuál de esos periodos será usado en la resolución del problema. Esta tarea es computacionalmente costosa y el periodo apropiado depende de las características del problema a resolver. En la línea de investigación se propone una estrategia diferente: un dAG auto-adaptativo que automáticamente define el valor del periodo a un costo bajo. Nuestra técnica la puede usar cualquier dAG para resolver cierto problema logrando minimizar el tiempo de puesta a punto.

El algoritmo resultante se compone de varios sub-algoritmos que utilizan la misma técnica de optimización, un algoritmo genético tradicional, pero considerando distintos periodos de migración, cuyo valor cambia de forma dinámica en relación con la información obtenida durante el proceso de búsqueda (no se define a priori de una manera ad-hoc). El algoritmo resultante ya no es un dAG homogéneo, sino un dAG heterogéneo que no necesita del tradicional preajuste de los parámetros de migración.

La continuación que se propone a esta línea de investigación es el uso de diferentes métricas para la toma de decisiones como así también la incorporación de más parámetros de migración al mecanismo auto-adaptativo del dAG heterogéneo.

Otra tópicos que se trata está orientado también a reducir los tiempos de configuración pero en un AG panmítico utilizando un mecanismo auto-adaptativo. En este caso este mecanismo está orientado a la selección de un operador de recombinación apropiado al problema que se está resolviendo, debido a que no se dispone de una técnica para determinar a priori cuál es el operador más adecuado para el problema en estudio. El nuevo mecanismo adaptativo elige automáticamente entre diferentes formas de operadores de recombinación, usando un criterio simple y poco costoso computacionalmente, basado en estadísticas tradicionales obtenidas del proceso de búsqueda. Esta opción no sólo permite la exploración simultánea del espacio del problema y de diferentes AGs, sino que tiende a aliviar el tiempo necesario

para ajustar los parámetros del algoritmo. En este caso, la acción futura está orientada a perfeccionar el mecanismo auto-adaptativo considerando nuevos criterios e incorporando otros parámetros del AG como pueden ser las probabilidades de cruce. Por otra parte, para fusionar esta línea de investigación con la anterior, se podría considerar este AG auto-adaptativo en un entorno distribuido y así avanzar en la propuesta de dAGs heterogéneos.

III. RESULTADOS OBTENIDOS/ESPERADOS

En esta sección se detallan los resultados obtenidos en la línea de investigación en desarrollo, presentada en la sección anterior.

Los resultados del AG paralelo heterogéneo propuesto, denominado HdAG, fue comparado con los producidos por un AG paralelo homogéneo, denominado HomdAG (el cual realiza la misma clase de búsqueda sobre diferentes conjuntos de individuos) con el fin de mostrar la robustez del algoritmo heterogéneo como un método de búsqueda alternativo [16], [17]. Como estamos interesados en la auto-adaptación del periodo de migración, en el caso de HomdAG se presenta un conjunto de valores para el período de migración que van desde 1 (máximo del acoplamiento entre las islas) a 128 (islas bastante aisladas), con el fin de caracterizar el efecto del período de la migración en la calidad de las soluciones obtenidas.

Para evaluar el desempeño de los algoritmos propuestos, se utilizó el problema de corte máximo en un grafo (MaxCut) [12]. Dado un grafo ponderado no dirigido, este problema consiste en hallar una división de los nodos en dos subconjuntos, tal que se maximice la suma de los pesos de los vértices con un extremo en subconjuntos diferentes. Es un muy conocido problema NP duro, no sólo importante desde el punto de vista teórico, sino con aplicaciones en varios campos [5], [15], lo que hace interesante su estudio.

Como resultado general obtuvimos que el HdAG obtiene la misma calidad de resultados, o aún mejores, que cualquiera de las variantes homogéneas estudiadas. Vale destacar que para algunas instancias del problema en estudio se obtuvieron mejores valores que los reportados en la literatura. Por otra parte, se observa que HdAG reduce en un 75 % el tiempo de cómputo total comparado con un método de búsqueda que requiere un pre-ajuste para determinar el valor adecuado del periodo de migración. Este resultado es interesante, ya que es un problema muy estudiado y porque nuestro verdadero objetivo es proponer un algoritmo que reduzca los tiempos relacionados con el pre-ajuste de parámetros. Los resultados también muestran que es una técnica muy precisa. Por lo tanto, no sólo HdAG es una versión heterogénea simple, sino también un buen algoritmo para resolver el problema MaxCut.

A continuación se detallan los resultados obtenidos en la línea de investigación relacionada con el diseño y desarrollo de AG adaptativos, el cual determina el operador de recombinación a utilizar durante la solución de un problema [19]. En este caso el problema seleccionado es el NK-landscapes [13],

ya que es útil tanto para investigar las propiedades estructurales del espacio de búsqueda como para evaluar el rendimiento de un algoritmo evolutivo. El problema de NK-landscapes es una función de fitness $f : \{0, 1\}^N \rightarrow \mathfrak{R}$ definida sobre cadenas binarias, donde N es el largo de la cadena y K es la cantidad de bits en dicha cadena que interactúan epistáticamente con un bit dado. Los operadores considerados en este trabajo son el crossover uniforme y el de un punto, los cuales presentan distintos niveles de perturbaciones del material genético. El criterio para seleccionar cuál operador aplicar está basado en el uso de una medida simple, tal como el fitness medio de la población el cual brinda información del estado del proceso de búsqueda. Los resultados obtenidos indican, en general, que el algoritmo adaptativo genera buenos resultados en comparación con un AG tradicional utilizando un único operador de cruce (uniforme o cruce de un punto). Analizando el rendimiento del AG adaptativo contra el de un AG con selección aleatoria entre los dos operadores de cruce, podemos ver que nuestra propuesta es capaz de encontrar una mejor calidad de soluciones para el conjunto de instancias resueltas, sin necesidad del tradicional y costoso pre-ajuste relacionado con la selección de los operadores de recombinación adecuados para resolver el problema en cuestión.

IV. FORMACIÓN DE RECURSOS HUMANOS

En cuanto a la formación de recursos humanos cabe mencionar que en el marco de las actividades desarrolladas en el LISI, uno de los investigadores ha concluido la escritura de una tesis doctoral, realizando la correspondiente presentación y defensa para obtener el título de Doctor en Ciencias de la Computación (UNSL). Además, uno de los becarios del proyecto ha presentado su tesis para alcanzar el título de Ingeniero en Sistemas. Es interesante resaltar que en el LISI se trabaja con alumnos avanzados en la carrera Ingeniería en Sistemas en temas relacionados a la resolución de problemas de optimización usando técnicas inteligentes, con el objeto de guiarlos en el desarrollo de sus tesis de grado y, también, de formar futuros investigadores.

REFERENCES

- [1] E. Alba. *Parallel Metaheuristics: A New Class of Algorithms*. Wiley, 2005.
- [2] E. Alba and J.M. Troya. Influence of the migration policy in parallel distributed GAs with structured and panmictic populations. *Applied Intelligence*, 12(3):163–181, 2000.
- [3] T.C. Belding. The distributed genetic algorithms revisited. *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 114–121, 1995.
- [4] Z. Bingul. Adaptive genetic algorithms applied to dynamic multiobjective problems. *Applied Soft Computing*, 7:791–799, 2007.
- [5] S. Burer, R.D.C. Monteiro, and Y. Zhang. Rank-two relaxation heuristics for Max-Cut and other binary quadratic programs. *SIAM Journal on Optimization*, 12(2):503–521, 2002.
- [6] D. CH, Z. YF, and C. WR. Adaptive probabilities of crossover and mutation in genetic algorithms based on cloud model. *Proc. of 2006 IEEE Information Theory Workshop*, pages 710–713, 2006.
- [7] T.G. Crainic and M. Toulouse. Parallel metaheuristics. In T.G. Crainic and G. Laporte, editors, *Fleet Management and Logistics*, pages 205–251. Kluwer Academic Publisher, 2003.
- [8] T.G. Crainic and M. Toulouse. Parallel strategies for metaheuristics. In F.W. Glover and G.A. Kochenberger, editors, *Handbook of Metaheuristics*, pages 475–514. Kluwer Academic Publisher, 2003.
- [9] V. Cung, S.L. Martins, C.C. Ribeiro, and C. Roucairol. Strategies for the parallel implementation of metaheuristics. In C.C. Ribeiro and P. Hansen, editors, *Essays and Surveys in Metaheuristics*, pages 263–308. Kluwer, 2003.
- [10] A.E. Eiben, R. Hinterding, and Z. Michalewicz. Parameter control in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 3:124–141, 1999.
- [11] S.C. Ghosh, B.P. Sinha, and N. Das. Channel assignment using genetic algorithm based on geometric symmetry. *IEEE Transactions on Vehicular Technology*, 52:860–875, 2003.
- [12] R. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85 – 103. 1972.
- [13] S. Kauffman. *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press, 1993.
- [14] F. Luna, E. Alba, and A.J. Nebro. Parallel metaheuristics. A new class of algorithms. In E. Alba, editor, *Parallel heterogeneous metaheuristics*, pages 395–422. Wiley, 2005.
- [15] W.B. McRae and E.R. Davidson. Linear inequalities for density matrices. II. *Journal of Mathematical Physics*, 13(10):1527–1538, 1972.
- [16] C. Salto and E. Alba. Designing heterogeneous distributed GAs by efficient self-adapting the migration period. *Applied Intelligence*, pages 1–9, 2011.
- [17] C. Salto, E. Alba, and F. Luna. Using landscape measures for the online tuning of heterogeneous distributed GAs. *Proceedings of the 13th annual conference companion on Genetic and evolutionary computation (GECCO '11)*, pages 691–694, 2011.
- [18] W.M. Spears. Adapting crossover in evolutionary algorithms. *Proc of the Fourth Annual Conference on Evolutionary Programming*, pages 367–384, 1995.
- [19] N. Stark and C. Salto C. A self-adaptive recombination method in evolutionary algorithms for solving epistatic problems. *XVII Congreso Argentino de Ciencias de la Computación (CACIC 2011)*, pages 1–10, 2011.
- [20] R. Tanese. Distributed genetic algorithms. *Proceedings of the Third International Conference on Genetic Algorithms*, pages 434–439, 1989.
- [21] S. Yang. Adaptive crossover in genetic algorithms using statistics mechanism. *Artificial Life VIII*, pages 182–185, 2002.