

Propuesta metodológica para el diseño de metaheurísticas paralelas en ambientes de cómputo heterogéneo

Carolina Salto¹ and Enrique Alba²

¹ UNLPam - CONICET, Argentina saltoc@ing.unlpam.edu.ar

² Universidad de Málaga, Málaga, España eat@lcc.uma.es

1. Resumen

Los ambientes de computación paralelo y distribuido han ganado popularidad como una forma de aportar la potencia de cómputo necesaria para resolver complejos problemas, representando una estrategia efectiva para la ejecución de algoritmos evolutivos distribuidos (dEAs). La mayoría de los trabajos publicados sobre dEAs asumen que la plataforma paralela subyacente tiene características idénticas (ambiente homogéneo) en cuanto a componentes de hardware y de software. Esta homogeneidad es difícil de encontrar en los laboratorios modernos (Figura ??) por la coexistencia de equipamiento nuevo y viejo, indicando que las plataformas heterogéneas paralelas (diferente rendimiento, construcción y capacidades) serán en el futuro el ambiente dominante en investigación.

La mayoría de la literatura sobre metaheurísticas y ambientes heterogéneos trata sobre la resolución de un problema particular, no hacen referencia a una metodología que ayude a los investigadores en el uso de configuraciones heterogéneas. Esto motiva nuestro trabajo clave (key work) [?], donde se propone un procedimiento metodológico, denominado *Heterogeneous Aware Parallel Algorithm* (HAPA), para guiar el diseño de metaheurísticas, en particular dEA, que se puedan ejecutar sobre software y hardware diverso. Con HAPA se busca que el algoritmo diseñado tenga exactitud y tiempos de ejecución comparables con las metaheurísticas existentes ejecutados sobre plataformas homogéneas, a pesar de incluir hardware antiguo.



Figura 1: HAPA es una metodología para construir algoritmos paralelos eficientes sobre plataformas heterogéneas como ésta.

HAPA consiste en calcular la potencia relativa de los procesadores de una manera *offline*, más un uso *online* de esta información dentro del algoritmo distribuido mientras se está ejecutando, es decir, medir una vez para usar muchas veces. La metodología se divide en tres fases: *i*) calcular numéricamente con datos reales la potencia relativa de cada máquina a fin de caracterizar la plataforma, *ii*) diseñar el algoritmo paralelo con información de la fase previa, y *iii*) validar el dEA propuesto en un problema complejo.

En el trabajo se explican dos posibles instancias de HAPA en el diseño de dEAs. La primera, para determinar la cantidad de generaciones usadas como criterio de parada en cada isla (algoritmo dGA_HAPA), para así hacer frente a las diferentes velocidades de los procesadores subyacentes a cada isla. La segunda, para definir la frecuencia de migración de las islas (algoritmo dGA_HAPA-FM), de modo que todas las islas reciban la misma cantidad de migrantes durante la evolución. En ambos casos, el objetivo es determinar en forma dinámica los valores de los parámetros para hacer frente a las diferencias en el hardware involucrado. Los algoritmos dGA_HAPA y dGA_HAPA-FM se comparan con un dEA tradicional (dGA_hom) ejecutado en un ambiente homogéneo.

Los dEAs generados usando esta nueva propuesta pueden obtener cantidades similares de buenos resultados que un dEA ejecutando en un hardware homogéneo moderno (ver Cuadro ??). Además, se obtiene una reducción en número de evaluaciones de la función objetivo realizadas por la nueva técnica (ver Cuadro ??). Estos resultados refuerzan la idea de la utilidad de HAPA en el diseño de algoritmos de la familia de dEAs competentes.

Concluyendo, con este artículo hemos contribuido a evitar el diseño puramente *ad hoc* de algoritmos para ejecutarse en plataformas heterogéneas, además de generar una nueva línea de investigación en la forma de incorporar conocimiento del hardware en los parámetros del algoritmo.

Referencias

1. C. Salto and E. Alba. Adapting Distributed Evolutionary Algorithms to Heterogeneous Hardware. *Transactions on Computational Collective Intelligence XIX, LNCS vol. 9380*, 103–125, 2015.

Cuadro 1: Tasa de éxito obtenido por dGA_hom y las variantes HAPA para instancias del problema de la mochila

	dGA_hom	dGA_HAPA	dGA_HAPA-FM
KP1-1k	87	63	80
KP1-10k	90	93	93
KP2-1k	70	60	50
KP2-10k	0	0	0
KP3-1k	7	7	13
KP3-10k	27	17	27
KP4-1k	0	0	3
KP4-10k	3	0	0

Cuadro 2: Esfuerzo numérico obtenido por dGA_hom y las variantes HAPA para instancias del problema de la mochila

	dGA_hom	dGA_HAPA	dGA_HAPA-FM
KP1-1k	72635	47739	35991
KP1-10k	49282	69662	42794
KP2-1k	94397	81542	83723
KP2-10k	0	0	0
KP3-1k	190293	63336	67106
KP3-10k	154821	86918	99457
KP4-1k	0	0	61282
KP4-10k	110824	0	0