

# Metaheurísticas multiobjetivo adaptativas

Mirialys Machin Navas<sup>1</sup> y Antonio J. Nebro Urbaneja<sup>2</sup>

<sup>1</sup> Facultad Regional UCI Ciego de Ávila,  
Cuba

<sup>2</sup> Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga,  
España

mmachin@cav.uci.cu, antonio@lcc.uma.es

**Resumen.** La optimización de problemas en los que hay maximizar o minimizar a la vez varias funciones, que usualmente están en conflicto entre sí, usando metaheurísticas, es un campo de investigación cada vez más popular, que ha dado lugar a una disciplina conocida como optimización multiobjetivo. Las metaheurísticas son técnicas no exactas que intentan proporcionar soluciones satisfactorias a problemas complejos de optimización en los que las técnicas exactas no son viables, y se caracterizan por usar una serie de operadores que se aplican de forma estocástica de acuerdo a cierta parametrización. Los valores de estos parámetros suelen ser establecidos al inicio de la ejecución de las técnicas y permanecen invariados hasta que éstas terminan, y recientemente están surgiendo trabajos que sugieren que dichos parámetros se modifiquen de forma adaptativa, según la marcha del algoritmo. En este trabajo se propone estudiar el efecto de usar dos operadores de forma adaptativa en dos metaheurísticas multiobjetivo representativas. Los resultados obtenidos indican que es posible mejorar el rendimiento de los algoritmos usando adaptatividad.

**Palabras clave.** Estrategia adaptativa, metaheurísticas, optimización multiobjetivo.

## Multiobjective Adaptive Metaheuristics

**Abstract.** Solution of Abstract Optimization problems with two or more conflicting functions or objectives by using metaheuristics has attracted attention of researches and become a rapidly developing area known as Multiobjective Optimization. Metaheuristics are non-exact techniques aimed to produce satisfactory solutions to complex optimization problems where exact techniques are not applicable; they are characterized by using some operators that are applied in a stochastic way according to a given parameterization. The settings of these parameters are usually established at the beginning of the execution of

algorithms, and they remain unchanged until the search finishes. Recently, a number of papers studying adaptive modifications of these parameters on the fly have emerged. In this work, we report a study of the effect of using two operators in an adaptive way in two multiobjective metaheuristics representative of the state-of-the-art. The obtained results demonstrate that it is possible to improve the search performance of two chosen algorithms by using the adaptive scheme.

**Keywords.** Adaptive strategy, metaheuristics, multiobjective optimization.

## 1 Introducción

La vida diaria está llena de situaciones donde necesariamente se presentan y se deben resolver problemas de optimización. Mientras estos problemas se manifiestan sin grandes restricciones y complejidades se pueden resolver usando simple lógica o formulaciones matemáticas. Constantemente nos encontramos resolviendo este tipo de problemas, como hallar el camino más corto, aprovechar al máximo el espacio en una habitación, entre otros. Sin embargo el mismo desarrollo ha impuesto la aparición de problemas complejos donde estas técnicas dejan de ser factibles y es necesario entonces hacer uso de las computadoras.

La optimización de dichos problemas usa un conjunto de datos, condiciones y ciertas restricciones que pueden dificultar el proceso de resolución de los mismos usando métodos exactos, debido a la complejidad de los cálculos y a que en muchas ocasiones el tiempo de resolución es muy extenso. Estas situaciones se presentan por ejemplo cuando se intentan optimizar problemas de complejidad exponencial

o NP-duros. Debido a la gran importancia que tienen estos problemas de optimización en la práctica, muchas ingenierías, en especial la Informática, han desarrollado y empleado múltiples métodos para tratar de resolverlos.

Entre estas técnicas están las metaheurísticas, son las que se aplicarán en el presente trabajo. Por otro lado, muchos problemas de optimización del mundo real se caracterizan por requerir optimizar más de una función al mismo tiempo. Estas funciones u objetivos suelen ser contrapuestos (mejorar uno implicar empeorar otro), haciendo que la solución a dichos problemas, denominados multiobjetivo o multicriterio, no sea una única solución sino un conjunto de ellas conocido como conjunto de óptimos de Pareto. La aplicación de metaheurísticas a problemas de optimización multiobjetivo durante los últimos 10 años se ha convertido en un campo de investigación abierto y de gran interés.

Si se profundiza en las metaheurísticas de optimización en general (tanto mono como multiobjetivo) se puede observar que éstas se rigen por unos parámetros que configuran su funcionamiento. Estos parámetros se suelen fijar al inicio de la ejecución de las técnicas y no varían durante el periodo de búsqueda del óptimo de los problemas. Sin embargo, no hay una razón que justifique que dicha configuración inicial sea la más adecuada una vez que los algoritmos se están ejecutando, por lo que han surgido estudios para intentar que los valores de los parámetros que rigen el comportamiento de las metaheurísticas se adapten de forma dinámica durante el proceso de búsqueda con el fin de que ésta sea más efectiva.

## 2 Metodología computacional

En esta sección se describirán los algoritmos trabajados, los diferentes problemas a utilizar y los indicadores de calidad que se han considerado para la realización de este trabajo, así como el análisis estadístico realizado para presentar los resultados con validez.

### 2.1 Algoritmos estudiados

NSGA-II es uno de los algoritmos evolutivos más populares de optimización multiobjetivo. Propuesto por Kalyanmoy Deb [1] y descrito en [2] está basado en la ordenación no-dominada de individuos, consistente en dividir una población en diferentes fronteras, la primera de ella formada por individuos a los que no domina nadie, la segunda de ellas formada por individuos que sólo son dominados por un individuo (perteneciente a la primera frontera), y así sucesivamente. Claramente, el número de fronteras será  $n \leq N$ , donde  $N$  es el tamaño de la población que se quiere ordenar. El número de fronteras será  $n = N$  en el caso de una población en la que exista una ordenación plana de individuos, es decir, una población en la que es cierta la afirmación de que existe un individuo que domina a todos los demás, y si este individuo es eliminado de la población esta afirmación se sigue cumpliendo.

El algoritmo NSGA-II hace uso de un algoritmo de ordenación de individuos denominado ordenación no dominada rápida (Fast NonDominated Sorting), que requiere únicamente del orden de  $O(mn)$  comparaciones para cada individuo, siendo  $M$  el número de funciones objetivos a optimizar y  $N$  el tamaño en número de individuos de la población.

El algoritmo Multi-objective Evolutionary Algorithm Based on Decomposition [3] (MOEA/D) es una metaheurística multi-objetivo que ha mostrado un notable desempeño en la resolución de problemas de optimización complicados.

Se basa en el uso de la descomposición, de tal manera que un número de subproblemas de un solo objetivo se optimizan de forma simultánea, cada uno de estos subproblemas es una agregación de todos los objetivos del MOP.

La diversidad es promovida por el establecimiento de relaciones de vecindad entre los subproblemas, que se definen en función de las distancias entre sus vectores de coeficientes de agregación. MOEA/D y, en particular, su variante con evolución diferencial, MOEA/D-DE [4] ha mostrado un notable rendimiento al resolver MOPs con la resolución de complicados sistemas de Pareto.

## 2.2 Problemas de prueba

Los problemas utilizados para evaluar el rendimiento de las técnicas propuestas constan de ocho problemas del banco de problemas presentado en [4] denominado LZ09, compuesto por nueve MOPs de elevada complejidad, que se denominarán LZ09 F1...LZ09 F9. Todos los problemas son bi-objetivo excepto la instancia LZ09 F6, que se ha omitido en el presente estudio.

Estos problemas se caracterizan por tener conjuntos soluciones Pareto-óptimas muy difíciles de encontrar, ya que se han definido expresamente con esa idea. Esto hace que las metaheurísticas multiobjetivo encuentren muchas dificultades para hallar buenas aproximaciones a los frentes de Pareto de estos problemas.

## 2.3 Indicadores de calidad

Con el objetivo de medir el rendimiento de los algoritmos evaluados, se han considerado los siguientes indicadores de calidad: Epsilon, que mide convergencia al determinar la distancia mínima (en cualquier objetivo) que habría que desplazar cada solución para ser no dominada con respecto a otro frente; Dispersión (Spread) [1], la cual mide la distribución de las soluciones y la distancia con los extremos del verdadero frente; e Hipervolumen (HV), que mide la diversidad del frente y la proximidad al frente óptimo de forma conjunta.

## 2.4 Metodología de las pruebas y análisis estadísticos

La metodología de los experimentos realizados se describe a continuación. Se han llevado a cabo 100 ejecuciones independientes de cada experimento, es decir, de cada algoritmo al resolver cada problema. Una vez aplicados los indicadores de calidad, se han obtenido las medianas y rangos intercuartílicos (IQR), como medidas de localización y dispersión, respectivamente. Para mayor claridad en la interpretación de los resultados mostrados en las tablas, los mejores resultados de cada experimento se marcarán con un fondo gris más oscuro. Adicionalmente, se hace uso de un

segundo tono de gris, más claro, para resaltar los segundos mejores resultados por cada indicador. Por último, para comprobar si las diferencias obtenidas entre los algoritmos son significativas o no desde el punto de vista estadístico, se ha aplicado el test de Wilcoxon para cada par de algoritmos. La prueba de los signos de Wilcoxon es una prueba no paramétrica para comparar la mediana de dos muestras relacionadas y determinar si existen diferencias entre ellas. Se utiliza como alternativa a la prueba t-Student cuando no se puede suponer la normalidad de dichas muestras. Para mostrar los resultados en las tablas, en cada celda cada problema se representará por un símbolo. Hay tres símbolos: “-” indica que no hay diferencias significativas entre los algoritmos, “▼” significa que el algoritmo de la fila ha producido mejores resultados que el algoritmo en la columna con significancia estadística, y “▲” se utiliza cuando el algoritmo en la columna es estadísticamente mejor que el de la fila en el problema considerado.

Tanto NSGA-II como y MOEA/D y sus variantes han sido implementados con jMetal, un framework en Java para optimización multiobjetivo con metaheurísticas [5][6], desarrollado en el Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga. Es código Open Source, que puede obtenerse de esta dirección: <http://www.sourceforge.net/projects/jmetal>.

NSGA-II usa por defecto cruce SBX y mutación polinomial. Dado que se pretende evaluar una versión adaptativa del mismo usando evolución diferencial, se ha implementado una versión que utiliza este operador denominada NSGAIIde. A continuación se han desarrollado dos versiones que usan ambos operadores: una que los utiliza de forma aleatoria (NSGAIIrd) y otra que los usa de forma adaptativa (NSGAIIad). Esta última usa un esquema que calcula la contribución de cada operador (SBX+mutación polinomial, DE+mutación polinomial) de acuerdo con la estrategia planteada en [7].

Con MOEA/D-DE se ha procedido de forma parecida, pero a la inversa: dado que esta metaheurística usa evolución diferencial y mutación polinomial se ha implementado una versión de que usa cruce SBX (MOEADsbx). A

continuación, se han desarrollado dos versiones que usan los operadores de forma aleatoria (MOEADrd) y adaptativa (MOEADad).

## 2.5 Parametrización de los algoritmos

En esta sección se presentarán los parámetros usados en cada una de las variantes implementadas de ambos algoritmos. De cada uno de ellos se presentarán cuatro variantes que serán comparadas entre sí, ya comentadas al final de la sección anterior.

**Tabla 1.** Parametrización usada en NSGA-II y MOEAD y sus variantes  
(L = número de variables del problema)

Parametrización de NSGAI, MOEADsbx	
Tamaño de población	100 individuos
Selección de padres	Torneo binario
Recombinación	SBX, $P_c = 0.9$
Mutación	Polinomial $P_m = 1.0/L$
Parametrización de NSGAIde, MOEAD	
Tamaño de población	100 individuos
Selección de padres	Selección aleatoria
Recombinación	Evolución Diferencial $F = 0.5, CR = 1.0$
Mutación	Polinomial $P_m = 1.0/L$
Parametrización de NSGAIrd, NSGAIad, MOEADrd y MOEADad	
Tamaño de población	100 individuos
Selección de padres	Selección aleatoria
Recombinación	SBX, $P_c = 0.9$ Evolución Diferencial $F = 0.5, CR = 1.0$

Los parámetros usados se detallan en la Tabla 1. Se ha buscado en todos los casos usar valores

de los parámetros de forma que la comparación de los resultados obtenidos por las distintas versiones de los algoritmos sean lo más justa posible. En todos los casos la población es de 100 individuos, que es el valor típico usado en este tipo de estudios. Los índices de distribución del cruce SBX y la mutación polinomial son de 20, y los operadores se aplican con probabilidades 0.9 y 1.0/L (siendo L el número de variables del problema), respectivamente; todos estos valores también son los habitualmente usados. Al aplicar evolución diferencial, los valores de los parámetros CR y F son, respectivamente, 0.5 y 1.0, que son los usados en [4].

## 3 Resultados y discusión

En esta sección se presentarán los resultados de los experimentos realizados con las variantes de los algoritmos NSGA-II y MOEA/D. Se compararán las variantes de cada algoritmo entre sí, presentando las tablas que incluyen los valores de los indicadores calidad obtenidos así como el resultado de aplicar el test de Wilcoxon.

El orden será el siguiente: Primeramente se comentarán los resultados del indicador Epsilon, se seguirá por Spread y se terminará por HV. Hay que comentar que los indicadores más relevantes son: el primero, ya que generalmente se considera que la convergencia es más importante que la diversidad; y el tercero, porque el HV es habitualmente considerado como uno de los indicadores más fiables.

### 3.1 Resultados de NSGA-II y sus variantes

Los resultados obtenidos al aplicar el indicador Epsilon a los frentes producidos por NSGA-II y sus variantes se incluyen en la Tabla 2. El primer hecho a destacar que el algoritmo NSGA-II estándar no obtiene el mejor resultado en ningún caso; sólo usando evolución diferencial destaca en el problema LZ09F1.

**Tabla 2.** Resultados del indicador Epsilon (mediana y rango intercuartílico, IQR) en NSGA-II y variantes

	<b>NSGAI</b>	<b>NSGAIde</b>	<b>NSGAIad</b>	<b>NSGAIrd</b>
LZ09_F1	2.13e-02 <sub>3.5e-03</sub>	1.64e-02 <sub>3.0e-03</sub>	1.71e-02 <sub>2.3e-03</sub>	1.70e-02 <sub>2.3e-03</sub>
LZ09_F2	1.57e-01 <sub>2.0e-02</sub>	1.17e-01 <sub>3.9e-02</sub>	9.88e-02 <sub>1.0e-02</sub>	8.99e-02 <sub>7.4e-03</sub>
LZ09_F3	1.23e-01 <sub>3.3e-02</sub>	8.78e-02 <sub>1.3e-02</sub>	8.03e-02 <sub>2.0e-02</sub>	6.71e-02 <sub>1.4e-02</sub>
LZ09_F4	1.59e-01 <sub>1.2e-02</sub>	8.59e-02 <sub>2.0e-02</sub>	5.84e-02 <sub>1.7e-02</sub>	5.95e-02 <sub>1.7e-02</sub>
LZ09_F5	1.13e-01 <sub>2.6e-02</sub>	8.04e-02 <sub>1.7e-02</sub>	8.57e-02 <sub>2.8e-02</sub>	7.17e-02 <sub>2.5e-02</sub>
LZ09_F7	3.07e-01 <sub>1.3e-01</sub>	2.30e-01 <sub>1.7e-01</sub>	1.32e-01 <sub>1.3e-01</sub>	3.74e-02 <sub>3.6e-02</sub>
LZ09_F8	3.46e-01 <sub>1.6e-01</sub>	3.20e-01 <sub>1.2e-01</sub>	2.76e-01 <sub>7.8e-02</sub>	2.76e-01 <sub>1.1e-01</sub>
LZ09_F9	2.74e-01 <sub>3.7e-02</sub>	1.12e-01 <sub>7.5e-02</sub>	1.46e-01 <sub>6.7e-02</sub>	1.08e-01 <sub>5.9e-02</sub>

Se puede observar con claridad cómo la versión que usa los dos operadores de forma aleatoria es con diferencia la que mayor rendimiento obtiene, consiguiendo los mejores valores del indicador en 5 de los ocho problemas estudiados y el segundo mejor valor en los otros tres. La versión adaptativa, NSGAIad, presenta los mejores valores de Epsilon en dos problemas y el segundo mejor valor en otros tres.

Si se observan los resultados de aplicar el test de Wilcoxon (Tabla 3) se evidencia como las diferencias obtenidas son significativas estadísticamente en la mayor parte de los casos. Como se indicó en la subsección 2.3, cada problema en cada celda está representado por un símbolo (el primero se corresponde con el problema LZ09 F1, el segundo con LZ09 F2, y así sucesivamente). Se puede determinar, entonces, que en los problemas LZ09 F4 y LZ09 F8 las diferencias entre NSGAIad y NSGAIrd no son significativas, lo que confirma a la versión aleatoria como la más destacada del estudio atendiendo al indicador epsilon.

Los resultados obtenidos al aplicar el indicador Spread y el test de Wilcoxon se incluyen en las Tablas 3 y 4, respectivamente. Se puede observar como nuevamente las dos variantes que usan los dos operadores son las que globalmente obtienen los mejores números, siendo la versión que usa los dos operadores de forma aleatoria, NSGAIrd, la más destacada (mejor valor en cuatro problemas, segundo mejor valor en tres).

En este caso, la versión adaptativa NSGAIad obtiene los frentes más diversos en los problemas LZ09 F1 y LZ09 F4 con significancia estadística respecto a la variante aleatoria.

Una vez analizados los resultados de los indicadores Epsilon y Spread, que dan una medida del grado de convergencia y diversidad respectivamente de los frentes obtenidos, la información aportada por el HV debería confirmar los resultados de los dos primeros. Hay que hacer constar que esta idea intuitiva es en realidad errónea, lo que es debido a que la forma de medir convergencia y diversidad del HV difiere a la aplicada por los indicadores Epsilon y Spread, por lo que a veces pueden surgir aparentes contradicciones. Si se presta atención a los valores obtenidos, incluidos en la Tabla 6 se observa que la versión que usa los operadores de forma aleatoria se muestra de nuevo como la más competitiva, obteniendo los mejores valores en seis de los ocho problemas del banco de pruebas; además, el test de Wilcoxon (incluido en la Tabla 7) indica que en la mayoría de los casos las diferencias son estadísticamente significativas.

### 3.2 Resultados de MOEAD y sus variantes

Después de mostrar y analizar los resultados obtenido por NSGA-II y sus variantes, se procede a hacer lo propio con MOEA/D. Al igual que antes, se comienza por el indicador Epsilon, cuyos valores aparecen en la Tabla 8.





**Tabla 9.** Resultados de aplicar el test de Wilcoxon al indicador Epsilon sobre los resultados de MOEAD y sus variantes

	MOEADs								MOEADad								MOEADrd							
MOEAD	▲	—	▲	▲	▲	▲	▲	▼	▼	—	—	—	▼	▲	▼	▼	—	▲	▲	▲	—	▲	▲	
MOEADs								▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	
MOEADd																	—	▲	▲	▲	▲	▲	—	▲

**Tabla 10.** Resultados del indicador Spread (mediana y rango intercuartílico, IQR) en MOEAD y sus variantes

	MOEAD		MOEADs		MOEADad		MOEADrd	
LZ09_F1	2.98e-01	1.2e-02	6.36e-01	7.3e-02	2.93e-01	1.0e-02	2.94e-01	7.6e-03
Z09_F2	6.33e-01	4.4e-01	1.05e+00	7.7e-02	5.90e-01	3.1e-01	6.54e-01	2.6e-01
LZ09_F3	4.88e-01	3.3e-01	7.25e-01	7.9e-02	4.99e-01	2.2e-01	6.00e-01	2.0e-01
LZ09_F4	1.26e+00	1.8e-01	6.87e-01	7.3e-02	8.88e-01	4.3e-01	8.95e-01	3.7e-01
LZ09_F5	5.59e-01	1.7e-01	6.68e-01	6.9e-02	5.52e-01	9.2e-02	5.75e-01	9.1e-02
LZ09_F7	1.13e+00	2.5e-01	1.17e+00	2.1e-01	1.07e+00	2.5e-01	1.13e+00	2.5e-01
LZ09_F8	1.09e+00	1.6e-01	1.22e+00	2.7e-01	1.28e+00	1.1e-01	1.23e+00	1.4e-01
LZ09_F9	4.64e-01	2.4e-01	1.10e+00	9.1e-02	4.57e-01	2.7e-01	6.19e-01	2.9e-01

**Tabla 11.** Resultados de aplicar el test de Wilcoxon al indicador Spread sobre los resultados de MOEAD y sus variantes

	MOEADs								MOEADad								MOEADrd								
MOEAD	▲	▲	▲	▼	▲	▲	▲	▲	▼	▼	—	▼	—	▼	▲	—	▼	—	—	▼	—	—	▲	▲	
MOEADs									▼	▼	▼	▲	▼	▼	▲	▼	▼	▼	▼	▲	▼	▼	—	▼	
MOEADd																		—	▲	▲	—	▲	▲	▼	▲

versiones, en esta sección se va a incluir una discusión sobre los mismos desde una perspectiva global. Se pretende determinar hasta qué punto el uso adaptativo de operadores ha sido efectivo al aplicarlo a los dos algoritmos considerados.

En el caso de NSGA-II, el uso adaptativo de los operadores ha superado claramente a la versión original del algoritmo y a la que sólo usa

evolución diferencial. Sin embargo, el uso combinado de forma aleatoria ha sido más efectivo que el esquema adaptativo, lo que indica que la estrategia adoptada no se ha comportado todo lo bien que se podría desear.

Estos resultados podrían derivarse debido al esquema de contribución empleado, pues al estar ajustado el uso de un determinado operador en dependencia de una probabilidad, podría estarse

**Tabla 12.** Resultados del indicador HV (mediana y rango intercuartílico, IQR) en MOEAD y sus variantes

	MOEAD	MOEADs	MOEADad	MOEADrd
LZ09_F1	6.61e-01 <sub>1.4e-04</sub>	6.19e-01 <sub>1.0e-02</sub>	6.61e-01 <sub>1.1e-04</sub>	6.61e-01 <sub>1.2e-04</sub>
LZ09_F2	6.09e-01 <sub>1.3e-01</sub>	5.52e-01 <sub>4.5e-02</sub>	6.48e-01 <sub>3.4e-02</sub>	6.28e-01 <sub>4.1e-02</sub>
LZ09_F3	6.47e-01 <sub>5.0e-02</sub>	6.17e-01 <sub>1.4e-02</sub>	6.48e-01 <sub>5.1e-03</sub>	6.42e-01 <sub>5.5e-02</sub>
LZ09_F4	6.52e-01 <sub>2.3e-03</sub>	6.18e-01 <sub>4.4e-03</sub>	6.51e-01 <sub>3.5e-03</sub>	6.50e-01 <sub>3.9e-03</sub>
LZ09_F5	6.42e-01 <sub>8.2e-03</sub>	6.23e-01 <sub>6.7e-03</sub>	6.41e-01 <sub>4.2e-03</sub>	6.39e-01 <sub>5.2e-03</sub>
LZ09_F7	6.55e-01 <sub>4.4e-03</sub>	3.85e-01 <sub>1.1e-01</sub>	6.56e-01 <sub>3.2e-03</sub>	6.54e-01 <sub>4.5e-03</sub>
LZ09_F8	5.34e-01 <sub>7.2e-02</sub>	3.91e-01 <sub>7.8e-02</sub>	5.21e-01 <sub>3.9e-02</sub>	5.07e-01 <sub>4.6e-02</sub>
LZ09_F9	3.03e-01 <sub>1.7e-02</sub>	2.21e-01 <sub>7.8e-02</sub>	3.14e-01 <sub>7.8e-03</sub>	2.98e-01 <sub>1.1e-02</sub>

**Tabla 13.** Resultados de aplicar el test de Wilcoxon al indicador HV sobre los resultados de MOEAD y sus variantes

	MOEADs	MOEADad	MOEADrd
MOEAD	▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▼	— ▲ — — ▲ ▼ ▲ — ▲ ▲ ▲ ▲ ▲ ▲ ▲	
MOEADs		▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼	
MOEADd			— ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲

usando más un operador que otro sin ser necesariamente el mejor, pues a medida que incrementa la probabilidad de uso de dicho operador.

### 4 Conclusiones

En este trabajo se ha hecho un estudio sobre el uso adaptativo de operadores. Para ello se han elegido dos algoritmos representativos del área, NSGA-II y MOEA/D, el primero de los cuales es el algoritmo que hoy día se considera como la técnica de referencia, y el segundo es el primero de una nueva generación de técnicas que pueden marcar la tendencia a seguir en los próximos años. Una vez seleccionados los algoritmos sobre los que realizar el estudio, se han implementado tres versiones de cada uno: una que usa un operador de cruce alternativo, DE en el caso de NSGA-II y SBX en el caso de MOEA/D

(ambos comparten el uso de mutación polinomial), y dos que usan estos dos operadores de forma combinada, ya sea aleatoriamente o de forma adaptativa, según un esquema usado en un trabajo de referencia. Se ha planteado un estudio riguroso, siguiendo los patrones que aplican en este tipo de estudios (100 ejecuciones independientes de cada configuración, uso de indicadores de calidad, análisis estadísticos), y usando como banco de prueba ocho problemas de optimización bi-objetivo de elevada dificultad.

Los resultados han indicado que el uso adaptativo de operadores ha sido particularmente eficaz en MOEA/D, mientras que en NSGA-II el uso aleatorio ha sido más efectivo. Teniendo en cuenta que el presente trabajo es la primera aproximación al tema abordado, los resultados obtenidos han sido muy satisfactorios. Se puede concluir entonces que se han dado cumplimiento a los objetivos propuestos al inicio del trabajo.

## Referencias

1. **Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002).** A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182–197.
2. **Gómez, J.C. & Terashima, H. (2012).** Building General Hyper-Heuristics for Multi-Objective Cutting Stock Problems. *Computación y Sistemas*, 16(3), 321–334.
3. **Zhang, Q. & Li, H. (2007).** MOEA/D: A multi-objective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6), 712–731.
4. **Li, H. & Zhang, Q. (2009).** Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II. *IEEE Transactions on Evolutionary Computation*, 13(2), 284–302.
5. **Durillo, J.J., Nebro, A.J., & Alba, E. (2010).** The JMetal framework for multi-objective optimization: Design and architecture. *2010 IEEE Congress on Evolutionary Computation (CEC)*, Barcelona, Spain, 1–8.
6. **Durillo, J.J., Nebro, A.J., Luna, F., Dorronsoro, B., & Alba, E. (2006).** *JMetal: a java framework for developing multi-objective optimization metaheuristics* (ITI-2006-10), Spain: University of Málaga.
7. **Vrugt, J. & Robinson, B. (2007).** Improved evolutionary optimization from genetically adaptive multimethod search. *Proceedings of the National Academy of Sciences of the United States of America*, 104(3), 708–711.



**Mirialys Machín Navas.** Profesora de Programación de la Facultad Regional UCI de Ciego de Ávila. Máster en Ingeniería de Software e Inteligencia Artificial.



**Antonio J. Nebro Urbaneja.** Departamento de Lenguajes y Ciencias de la Computación. Profesor Titular de Universidad de Málaga, España. Doctor en Ciencias de la Computación.

*Article received on 28/09/2012; accepted on 12/01/2013.*