

Equipo Elitista de Algoritmos Evolutivos Multiobjetivo

José Manuel Fernandez Giangreco

Ciencias y Tecnología - Universidad Católica Nuestra Señora de la Asunción
Centro Nacional de Computación - Universidad Nacional de Asunción
Asunción, Paraguay
jmfernandez@cnc.una.py

Benjamín Barán

Ciencias y Tecnología - Universidad Católica Nuestra Señora de la Asunción
Centro Nacional de Computación - Universidad Nacional de Asunción
Asunción, Paraguay
bbaran@cnc.una.py

Resumen

Con el uso cada vez más extendido de Algoritmos Evolutivos para Optimización Multiobjetivo en problemas del mundo real, se hace necesario mejorar su desempeño sacando el máximo provecho de las bondades de cada algoritmo. Para ello, una reconocida alternativa es la incorporación de conceptos de paralelismo. En consecuencia, los algoritmos evolutivos multiobjetivo paralelos se constituyen en un área de creciente interés para aplicaciones prácticas, sobre todo en problemas industriales, financieros y de ingeniería aplicada.

Dado que estos algoritmos difieren en su desempeño según el tipo del problema, el presente trabajo propone un Equipo Elitista de Algoritmos Evolutivos Multiobjetivo que combina diversos algoritmos en el contexto de la Computación Paralela asíncrona. Resultados experimentales demuestran las ventajas (y desventajas) de esta nueva propuesta, al compararla con implementaciones paralelas de diversos algoritmos evolutivos multiobjetivos ya publicados.

Palabras Claves: Algoritmos Evolutivos, Optimización Multiobjetivo, Paralelismo, Equipo de Algoritmos.

Abstract

With the extended use of Evolutionary Algorithms for Multiobjective Optimization in real world problems, it becomes necessary to improve their performance taking advantage of each algorithm virtue. For this purpose, a well-known alternative is the incorporation of parallelism. Therefore, parallel multiobjective evolutionary algorithms are becoming an area of growing interest for practical applications, mainly in industrial, financial and applied engineering problems.

Since these algorithms differ in performance for different kind of problems, an Elitist Team Algorithm combining different Multiobjective Evolutionary Algorithms is proposed in a Parallel Computation context. Experimental results validate this new proposal showing several advantages (and a few disadvantages) when it is compared to parallel implementations of diverse well-known multiobjective evolutionary algorithms.

Keywords: Evolutionary Algorithms, Multiobjective Optimization, Parallelism, Team Algorithm.

1. Introducción

En la búsqueda de soluciones a problemas del mundo real puede ser necesario satisfacer de manera simultánea múltiples objetivos, los cuales suelen ser contradictorios [22]. De existir la posibilidad de combinar los diferentes objetivos y conociendo la mejor manera de hacerlo, se puede simplemente considerar la existencia de un único objetivo a optimizar, resultante de la combinación de todos los objetivos considerados. Sin embargo, lo usual es desconocer de que manera se deben combinar los diferentes objetivos, o esta combinación resulta inadecuada, cuando no imposible. Entonces, se trata de un Problema de Optimización Multiobjetivo (*Multiobjective Optimization Problem - MOP*) [8, 20, 22, 26].

En los problemas de optimización multiobjetivo con objetivos contradictorios no existe una solución única que pueda ser considerada la mejor, sino un conjunto de alternativas representando las mejores relaciones de compromiso entre todos los objetivos, en el sentido que cada solución es mejor que las otras en algún objetivo, pero ninguna es mejor que otra en todos los objetivos simultáneamente [22]. Dicho conjunto es llamado conjunto de soluciones Pareto óptimo y sus correspondientes vectores en el espacio objetivo constituyen el denominado Frente Pareto [20].

Los Algoritmos Evolutivos (*Evolutionary Algorithms - EAs*) [1] han demostrado ser especialmente adecuados para la optimización multiobjetivo [10, 11, 16, 19, 27]. En la actualidad, existe un gran número de Algoritmos Evolutivos para Optimización Multiobjetivo (*MultiObjective Evolutionary Algorithms - MOEAs*) [9]. Con el uso cada vez más extendido de estos algoritmos en problemas reales de optimización, se hace necesario mejorar el desempeño de los mismos [2, 18]. Por lo tanto, para asegurar la aplicabilidad de la técnica de optimización evolutiva multiobjetivo a problemas de complejidad creciente, es necesario mejorar tanto la efectividad como la eficiencia de los métodos evolutivos. Para ello, una alternativa es la incorporación de conceptos de paralelismo al diseño de estos algoritmos [18, 25], así tenemos a los algoritmos evolutivos multiobjetivo paralelos (*parallel Multiobjective Evolutionary Algorithms - pMOEAs*).

Por otra parte se puede considerar a los algoritmos en equipo (*Team Algorithm - TA*), que han demostrado ser una excelente técnica computacional para combinar una variedad de algoritmos corriendo en diferentes procesadores de una red típicamente asíncrona, como las actuales redes de área local conformadas por computadores personales heterogéneos [2, 3, 4, 5, 6, 7].

Ya que se dispone de diversos algoritmos pMOEAs, se sugiere aquí la conformación de un *Team Algorithm of Multiobjective Evolutionary Algorithms (TA-MOEA)* como una alternativa válida para mejorar tanto la efectividad como la eficiencia, al poder explorar diferentes porciones del espacio de búsqueda, posiblemente disjuntas, con características variadas proporcionadas por algoritmos diferentes.

Siendo el área de los algoritmos evolutivos multiobjetivo un área de creciente importancia y reciente interés, este trabajo propone determinar las ventajas y desventajas existentes en el desarrollo y aplicación de un TA-MOEA que combine las mejores características de cada MOEA, conformando un TA paralelo de algoritmos evolutivos multiobjetivo. A fin de determinar la efectividad de la técnica propuesta, inspirado en [20, 26, 30], el presente trabajo realiza una comparación de distintos pMOEAs y el TA-MOEA utilizando diferentes métricas experimentales usualmente utilizadas para medir su desempeño [17, 26].

Entonces, el presente trabajo propone un TA-MOEA que utiliza un criterio de selección elitista de algoritmos, conforme se explicará en detalle en la Sección 4, por el cual se va reemplazando al peor MOEA de un conjunto de algoritmos trabajando en equipo, por otro MOEA de mejor desempeño en el referido equipo de algoritmos. Para determinar el mejor y el peor MOEA de un equipo de algoritmos trabajando en paralelo sobre un mismo problema, se realiza un *ranking* de los pMOEAs, basado en el número de soluciones aportadas al equipo (TA) por cada algoritmo.

El presente trabajo está organizado de la siguiente manera: en la Sección 2 se presenta la definición general de un problema de optimización multiobjetivo. La formulación de los problemas ZDT con sus funciones objetivo es dada en la Sección 3. Los algoritmos en equipo son explicados en la Sección 4. Los resultados experimentales son mostrados en la Sección 5. Por último, las conclusiones y los trabajos futuros son presentados en la Sección 6.

2. Optimización Multiobjetivo

Un Problema de Optimización Multiobjetivo (MOP) general incluye un conjunto de n parámetros (variables de decisión), un conjunto de k funciones objetivo, y un conjunto de m restricciones. Las funciones objetivo y las restricciones son funciones de las variables de decisión. Luego, un MOP puede expresarse como:

$$\begin{array}{ll} \text{Optimizar} & \mathbf{y} = \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})) \\ \text{sujeto a} & \mathbf{e}(\mathbf{x}) = (e_1(\mathbf{x}), e_2(\mathbf{x}), \dots, e_m(\mathbf{x})) \geq \mathbf{0} \\ \text{donde} & \mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbf{X} \\ & \mathbf{y} = (y_1, y_2, \dots, y_k) \in \mathbf{Y} \end{array} \quad (2.1)$$

siendo \mathbf{x} el vector de decisión y \mathbf{y} el vector objetivo. El espacio de decisión se denota por \mathbf{X} , y al espacio objetivo por \mathbf{Y} . Optimizar, dependiendo del problema, puede significar igualmente, minimizar o maximizar. El conjunto de restricciones $\mathbf{e}(\mathbf{x}) \geq \mathbf{0}$ determina el conjunto de soluciones factibles \mathbf{X}_f y su correspondiente conjunto de vectores objetivo factibles \mathbf{Y}_f .

El problema de optimización multiobjetivo consiste en hallar la \mathbf{x} que tenga el “mejor valor” de $\mathbf{f}(\mathbf{x})$. En general, no existe un único “mejor valor”, sino un conjunto de soluciones óptimas. Entonces, un nuevo concepto de optimalidad debe ser establecido para MOPs. Dados dos vectores de decisión $\mathbf{u}, \mathbf{v} \in \mathbf{X}$, se tiene:

$$\begin{array}{ll} \mathbf{f}(\mathbf{u}) = \mathbf{f}(\mathbf{v}) & \text{si y solo si } \forall i \in \{1, 2, \dots, k\}: f_i(\mathbf{u}) = f_i(\mathbf{v}) \\ \mathbf{f}(\mathbf{u}) \leq \mathbf{f}(\mathbf{v}) & \text{si y solo si } \forall i \in \{1, 2, \dots, k\}: f_i(\mathbf{u}) \leq f_i(\mathbf{v}) \\ \mathbf{f}(\mathbf{u}) < \mathbf{f}(\mathbf{v}) & \text{si y solo si } \mathbf{f}(\mathbf{u}) \leq \mathbf{f}(\mathbf{v}) \wedge \mathbf{f}(\mathbf{u}) \neq \mathbf{f}(\mathbf{v}) \end{array} \quad (2.2)$$

En un contexto de minimización, esta situación se expresa con los siguientes símbolos y términos:

$$\begin{array}{ll} \mathbf{u} > \mathbf{v} \text{ (} \mathbf{u} \text{ domina a } \mathbf{v} \text{)} & \text{si y solo si } \mathbf{f}(\mathbf{u}) < \mathbf{f}(\mathbf{v}) \\ \mathbf{v} > \mathbf{u} \text{ (} \mathbf{v} \text{ domina a } \mathbf{u} \text{)} & \text{si y solo si } \mathbf{f}(\mathbf{v}) < \mathbf{f}(\mathbf{u}) \\ \mathbf{u} \sim \mathbf{v} \text{ (} \mathbf{u} \text{ y } \mathbf{v} \text{ no son comparables)} & \text{si y solo si } \mathbf{f}(\mathbf{u}) \not\leq \mathbf{f}(\mathbf{v}) \wedge \mathbf{f}(\mathbf{v}) \not\leq \mathbf{f}(\mathbf{u}) \end{array} \quad (2.3)$$

Dado un vector de decisión $\mathbf{x} \in \mathbf{X}_f$, se dice que \mathbf{x} es no dominado respecto a un conjunto $\mathbf{V} \subseteq \mathbf{X}_f$ si y solo si $\mathbf{x} > \mathbf{v}$ o $\mathbf{x} \sim \mathbf{v}, \forall \mathbf{v} \in \mathbf{V}$. En caso que \mathbf{x} sea no dominado respecto a todo el conjunto \mathbf{X}_f , y solo en ese caso, se dice que \mathbf{x} es una solución Pareto óptima. Por lo tanto, el conjunto Pareto óptimo \mathbf{X}_{true} puede ser definido formalmente como:

$$\mathbf{X}_{true} = \{ \mathbf{x} \in \mathbf{X}_f \mid \mathbf{x} \text{ es no dominado con respecto a todo el conjunto } \mathbf{X}_f \} \quad (2.4)$$

El correspondiente conjunto de vectores objetivo $\mathbf{Y}_{true} = \mathbf{f}(\mathbf{X}_{true})$ constituye el Frente Pareto óptimo [12].

Usualmente, \mathbf{Y}_{true} no puede ser calculado en forma exacta en tiempos razonables por lo que para fines prácticos, se intentará calcular al menos una buena aproximación al Frente Pareto óptimo, conocido en la literatura como \mathbf{Y}_{known} [26].

3. Problemas ZDT

Los Problemas ZDT constituyen un conjunto de seis problemas desarrollados por Zitzler, Deb, y Thiele [30] como un escenario completo de prueba (*test bed*) que contempla distintas posibilidades y dificultades al considerar problemas multiobjetivos [15]. Todos estos problemas del *test bed* son propuestos en un contexto de minimización y están estructurados de igual manera sobre tres funciones \mathbf{f}_1 , \mathbf{g} y \mathbf{h} :

$$\begin{array}{ll} \text{Minimizar} & \mathbf{F}(\mathbf{x}) = (\mathbf{f}_1(x_1), \mathbf{f}_2(\mathbf{x})) \\ \text{sujeto a} & \mathbf{f}_2(\mathbf{x}) = \mathbf{g}(x_2, \dots, x_n) \cdot \mathbf{h}(\mathbf{f}_1(x_1), \mathbf{g}(x_2, \dots, x_n)) \\ \text{donde} & \mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbf{X} \end{array} \quad (3.1)$$

La función \mathbf{f}_1 es una función que depende únicamente de la primera variable de decisión, \mathbf{g} es una función de las $n-1$ variables de decisión restantes y los parámetros de \mathbf{h} son los valores de las funciones \mathbf{f}_1 y \mathbf{g} . Las funciones de prueba difieren en estas tres funciones así como en el número de variables n y en los valores que éstas pueden tomar.

El problema ZDT1 tiene 30 variables en el rango [0,1]. Su frente de Pareto es convexo. Tiene un frente Pareto continuo y una distribución uniforme de soluciones a lo largo del frente, donde $\mathbf{x}=(x_1,\dots,x_n)$, $n=30$ y $x_i \in [0,1]$.

$$f_1(\mathbf{x}_1) = x_1; \mathbf{g}(\mathbf{x}_2, \dots, \mathbf{x}_n) = 1 + 9 \cdot \sum_{i=2}^n \frac{x_i}{(n-1)}; \mathbf{h}(\mathbf{f}_1, \mathbf{g}) = 1 - \sqrt{\frac{f_1}{\mathbf{g}}} \quad (3.2)$$

El problema ZDT2 tiene 30 variables en el rango [0,1]. Su frente de Pareto es no convexo. La distribución de soluciones a lo largo del frente de Pareto es uniforme, donde $\mathbf{x}=(x_1,\dots,x_n)$, $n=30$ y $x_i \in [0,1]$.

$$f_1(\mathbf{x}_1) = x_1; \mathbf{g}(\mathbf{x}_2, \dots, \mathbf{x}_n) = 1 + 9 \cdot \sum_{i=2}^n \frac{x_i}{(n-1)}; \mathbf{h}(\mathbf{f}_1, \mathbf{g}) = 1 - \left(\frac{f_1}{\mathbf{g}}\right)^2 \quad (3.3)$$

El problema ZDT3 tiene 30 variables en el rango [0,1]. Su frente de Pareto es discontinuo. La distribución de soluciones a lo largo del frente es uniforme, donde $\mathbf{x}=(x_1,\dots,x_n)$, $n=30$ y $x_i \in [0,1]$.

$$f_1(\mathbf{x}_1) = x_1; \mathbf{g}(\mathbf{x}_2, \dots, \mathbf{x}_n) = 1 + 9 \cdot \sum_{i=2}^n \frac{x_i}{(n-1)}$$

$$\mathbf{h}(\mathbf{f}_1, \mathbf{g}) = 1 - \sqrt{\frac{f_1}{\mathbf{g}}} - \left(\frac{f_1}{\mathbf{g}}\right) \sin(10\pi f_1) \quad (3.4)$$

El problema ZDT4 tiene 10 variables en el rango [0,1]. Su frente de Pareto es convexo. La complejidad de este problema es que contiene 21^9 frentes Pareto-óptimos locales [30] y, por lo tanto, prueba a los algoritmos evolutivos con relación a su capacidad de lidiar con la multimodalidad. En este caso $\mathbf{x}=(x_1,\dots,x_n)$, $n=10$, $x_1 \in [0,1]$, $x_2, \dots, x_n \in [-5,5]$.

$$f_1(\mathbf{x}_1) = x_1; \mathbf{g}(\mathbf{x}_2, \dots, \mathbf{x}_n) = 1 + 10(n-1) + \sum_{i=2}^n (x_i^2 - 10 \cos(4\pi x_i))$$

$$\mathbf{h}(\mathbf{f}_1, \mathbf{g}) = 1 - \sqrt{\frac{f_1}{\mathbf{g}}} \quad (3.5)$$

El problema ZDT5 utiliza funciones booleanas definidas sobre *strings*. Constituye un caso de problema *deceptivo* ya que la forma de la función auxiliar determina que la mayor parte del espacio de búsqueda se concentra cerca de óptimos locales, mientras que el óptimo global se halla relativamente aislado. Aquí tenemos $\mathbf{x}=(x_1,\dots,x_n)$, $n=11$, $x_1 \in \{0,1\}^{30}$, $x_2, \dots, x_n \in \{0,1\}^5$.

$$f_1(\mathbf{x}_1) = 1 + u(\mathbf{x}_1); \mathbf{g}(\mathbf{x}_2, \dots, \mathbf{x}_n) = \sum_{i=2}^n v(u(\mathbf{x}_i)); \mathbf{h}(\mathbf{f}_1, \mathbf{g}) = \frac{1}{f_1}$$

$$u(\mathbf{x}_i) = \text{"Número de unos en } \mathbf{x}_i \text{"} \quad (3.6)$$

$$v(u(\mathbf{x}_i)) = \begin{cases} 2 + u(\mathbf{x}_i) & \text{si } u(\mathbf{x}_i) < 5 \\ 1 & \text{si } u(\mathbf{x}_i) = 5 \end{cases}$$

El problema ZDT6 tiene 10 variables en el rango [0,1]. Su frente de Pareto es no convexo. La complejidad de este problema está dada por la combinación de la forma no convexa del frente de Pareto y la distribución no uniforme de soluciones a lo largo de él, donde $\mathbf{x}=(x_1,\dots,x_n)$, $n=10$, $x_i \in [0,1]$.

$$f_1(\mathbf{x}_1) = 1 - e^{(-4x_1)} \sin^6(6\pi x_1); \mathbf{g}(\mathbf{x}_2, \dots, \mathbf{x}_n) = 1 + 9 \cdot \left(\sum_{i=2}^n \frac{x_i}{(n-1)}\right)^{\frac{1}{4}}$$

$$\mathbf{h}(\mathbf{f}_1, \mathbf{g}) = 1 - \left(\frac{f_1}{\mathbf{g}}\right)^2 \quad (3.7)$$

4. Algoritmos en Equipo

El desempeño de un algoritmo se mide generalmente por la calidad de los resultados obtenidos y por la rapidez en la respuesta. En este aspecto, los *Team Algorithms* han demostrado ser muy eficientes superando muchas veces en desempeño a los métodos tradicionales [3, 4, 5, 6]. Debido a estas ventajas en combinar diferentes métodos, el presente trabajo propone la combinación de siete Algoritmos Evolutivos Multiobjetivos:

- *Multiobjective Genetic Algorithm* (MOGA) [19].
- *Nondominated Sorting Genetic Algorithm* (NSGA) [24].
- *Niched Pareto Genetic Algorithm* (NPGA) [21].
- *Nondominated Sorting Genetic Algorithm II* (NSGA2) [13].
- *Controlled Elitist NSGA II* (CNSGA2) [14].
- *Strength Pareto Evolutionary Algorithm* (SPEA) [29].
- *Strength Pareto Evolutionary Algorithm II* (SPEA2) [28].

Se utilizaron estos algoritmos por considerarlos de los más representativos entre los MOEAs.

Los *Team Algorithm* han surgido como una novedosa herramienta que no solo mejora los tiempos de respuesta del algoritmo sino que además logra un efecto sinérgico, conforme fuera experimentalmente comprobado [3, 4, 5, 6].

El modelo propuesto en el presente trabajo se denomina *Team Algorithm of pMOEAs* y está formado por un proceso Coordinador y siete procesos paralelos esclavos en los cuales se pueden ejecutar uno de los siete MOEAs disponibles.

El diagrama de flujo del proceso Coordinador se observa en la Figura 1. El proceso Coordinador primeramente crea las estructuras para almacenar los resultados provenientes de los distintos procesos MOEA en su población, luego se agrega a un grupo de trabajo e inicia cada uno de los procesos MOEA con sus parámetros específicos. Esto es implementado utilizando primitivas de comunicación de grupo, proveídas por librerías de paso de mensajes [23], lo que facilita la comunicación entre los distintos procesos utilizados.

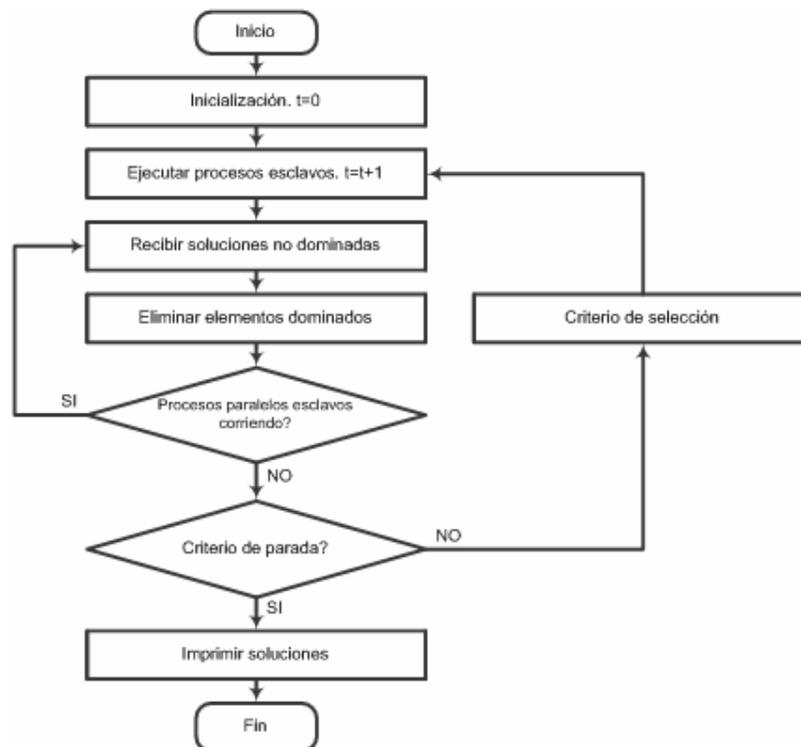


Figura 1. Diagrama de flujo del Coordinador

Durante la evolución, cada uno de los MOEAs que interviene en la búsqueda envía a todos los elementos que componen el equipo de trabajo un porcentaje de las mejores soluciones obtenidas. Cuando el Coordinador recibe estas soluciones las almacena en su población. A fin de mantener sólo las mejores soluciones, se eliminan las soluciones dominadas, también conocidas como soluciones cubiertas.

Cada MOEA se ejecuta independientemente en procesadores posiblemente diferentes, hasta cumplir un criterio local de parada. Cuando la cuenta de MOEAs en ejecución es igual a cero, el Coordinador verifica el criterio global de parada. De no ser alcanzado, se procede a seleccionar nuevos algoritmos esclavos para la siguiente iteración del TA-MOEA.

El criterio utilizado por el proceso Coordinador para la selección de los algoritmos esclavos es establecer un ranking de los algoritmos basado en el número de soluciones no dominadas aportadas a la población del Coordinador. El peor algoritmo es descartado y en su lugar se ubica una copia del mejor algoritmo de la última iteración.

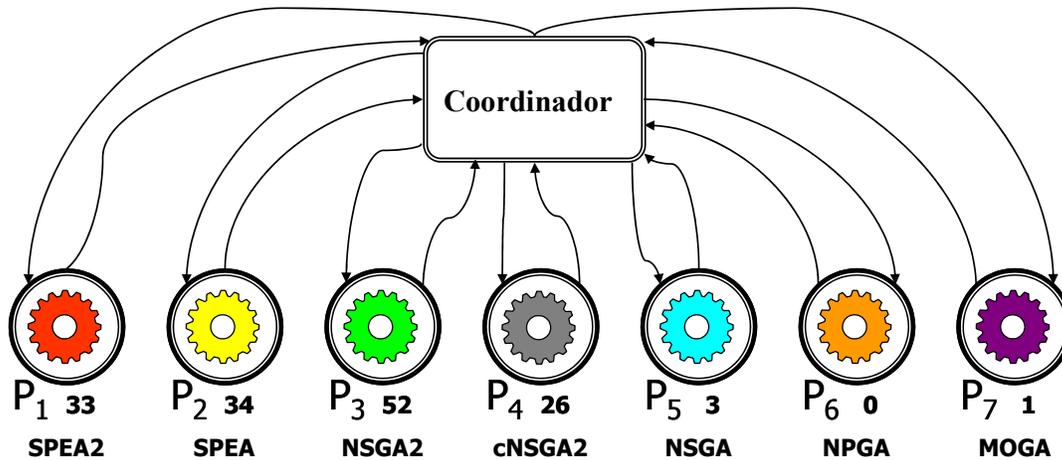


Figura 2. TA-MOEA con los MOEAs clasificados

En la Figura 2 se observan los siete procesos esclavos con los MOEAs correspondientes, clasificados conforme al número de soluciones encontradas. Así, tenemos que el mejor algoritmo es el NSGA2 con 52 soluciones aportadas a la población del Coordinador y el peor algoritmo es el NPGA con 0 soluciones aportadas.

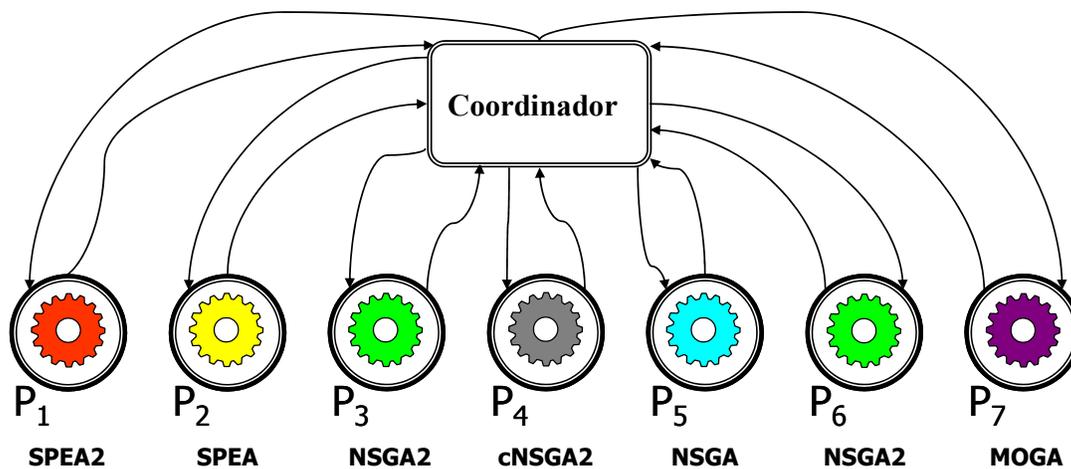


Figura 3. TA-MOEA luego del reemplazo

La Figura 3 presenta el TA-MOEA luego que una copia del NSGA2 reemplaza al NPGA.

Una vez seleccionados los algoritmos, se repite el ciclo hasta que el criterio de parada se cumpla; entonces, el Coordinador imprime todas las soluciones Pareto contenidas en su población y finaliza la ejecución.

Cabe resaltar que a medida que transcurran las iteraciones, el conjunto de procesos tiende a ejecutar el mismo algoritmo, y dicho algoritmo generalmente resulta el mejor MOEA del conjunto de algoritmos utilizados por el TA-MOEA en el problema específico que se está resolviendo. Por lo tanto, otro aporte indirecto que surge del TA-MOEA es la capacidad de identificar cual algoritmo de los combinados es el mejor MOEA para un determinado problema.

5. Resultados Experimentales

Los siete algoritmos elegidos para este trabajo fueron implementados, para la resolución paralela de los seis problemas de prueba presentadas en la Sección 3. Los algoritmos seleccionados se implementaron de acuerdo a la literatura original de referencia. Así, quedaron constituidos 8 algoritmos a ser comparados experimentalmente, 7 de ellos conformados por los pMOEAs enumerados en la Sección 4 y el TA-MOEA propuesto en el presente trabajo.

Característica	Descripción
Tipo de computadora	COW
Tipo de CPU	AMD 800MHz
Memoria	256 MB
Sistema Operativo	Red Hat Linux v7.3
Red de comunicación	Ethernet 100 Mbps
Librería de comunicación	PVM 3.4.5

Tabla 1. Características del ambiente computacional paralelo utilizado

Se utilizaron *Cluster Of Workstations* (COW) en el ambiente computacional que presenta la Tabla 1.

Se realizaron corridas con 1, 2, 4 y 8 procesadores. Las tablas de la Sección 5.3 presentan los resultados promedios de las cuatro corridas.

Para comparar la calidad de los resultados obtenidos por las 8 implementaciones realizadas, se utilizarán en el presente trabajo dos reconocidas métricas ampliamente utilizadas para la comparación de algoritmos evolutivos multiobjetivo [26], definidas a continuación.

5.1. Métricas Utilizadas

La selección de las métricas es una tarea crítica, que debe realizarse cuidadosamente, para evitar obtener resultados poco útiles. Es conveniente recordar que ningún criterio único puede dar una idea acabada del desempeño general de los MOEAs, ya que algunos se enfocan en la efectividad y otros en la eficiencia.

En el presente trabajo se utilizaron dos métricas: la Generación de vectores no dominados, que proporciona la cantidad de soluciones encontradas por el algoritmo y la Generación real de vectores no dominados, que da la cantidad de soluciones encontradas por el algoritmo y que no son dominadas por las soluciones de los demás algoritmos.

Generación de vectores no dominados (GVND). Esta métrica cuenta el número de soluciones en el frente Pareto calculado Y_{known} . Se puede definir mediante la siguiente ecuación:

$$GVND = |Y_{known}^{\Delta}|_c \quad (5.1)$$

donde $| \cdot |_c$ denota cardinalidad.

Generación real de vectores no dominados (GRVND). La métrica denominada generación real de vectores no dominados cuenta la cantidad de elementos en el frente Pareto calculado que en efecto pertenecen al frente Pareto óptimo:

$$GRVND = \left| \left\{ y \mid y \in Y_{known} \wedge y \in Y_{true} \right\} \right|_c \quad (5.2)$$

Para el cálculo de esta métricas que requiere el conocimiento del frente Pareto óptimo real, se utilizó como aproximación un conjunto de soluciones no dominadas con respecto al conjunto unión de todos los resultados obtenidos considerando la totalidad de las ejecuciones realizadas por cada una de las 8 implementaciones arriba mencionadas.

5.3. Resultados experimentales sobre los problemas ZDT

En la Figura 4 se grafica el tiempo en segundos requerido por las implementaciones realizadas para la resolución de los seis problemas ZDT en relación con el número de procesadores utilizados. Como puede apreciarse al aumentar el número de procesadores disminuye el tiempo de ejecución.

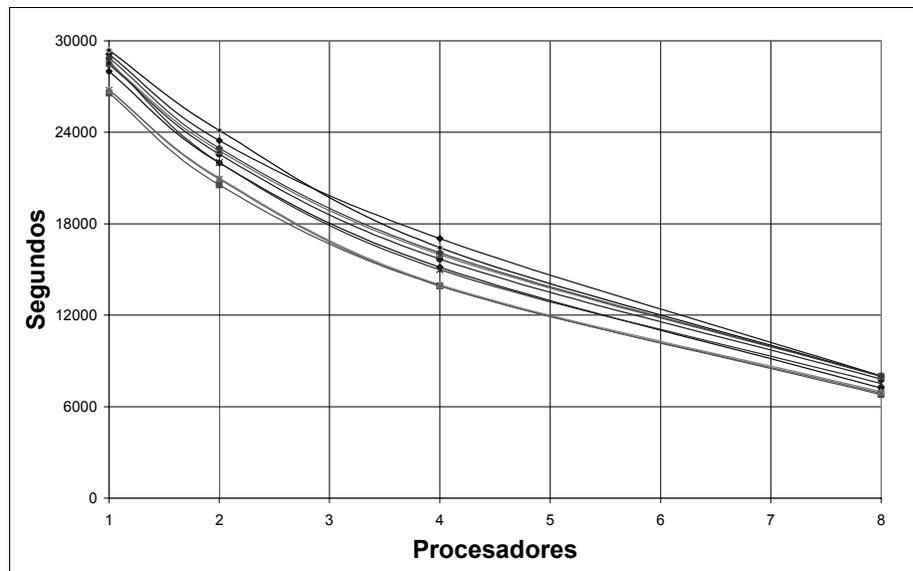


Figura 4. Tiempo total en segundos por número de procesadores

Basados en las métricas utilizadas, se procede a clasificar las 8 implementaciones realizadas, ordenándolas en un ranking (de mejor a peor) para cada problema ZDT.

	Conjunto	ZDT1	Conjunto	ZDT2	Conjunto	ZDT3		
1	NSGA2	1150	1	SPEA	924	1	SPEA2	805
2	TAMOEA	1079,75	2	NSGA2	902	2	TAMOEA	801
3	SPEA	979	3	TAMOEA	901	3	NSGA2	761
4	SPEA2	943	4	SPEA2	860	4	CNSGA2	754,75
5	CNSGA2	795,5	5	CNSGA2	831	5	SPEA	753
6	NSGA	450	6	NPGA	567	6	NPGA	677,25
7	NPGA	429	7	NSGA	508	7	NSGA	550,75
8	MOGA	369,5	8	MOGA	498	8	MOGA	500,5

Tabla 2. Ranking considerando la métrica GVND sobre los problemas ZDT1, ZDT2 y ZDT3

En la Tabla 2 se observa las posiciones en los problemas ZDT1, ZDT2 y ZDT3 utilizando la métrica GVND. Para el problema ZDT1, el mejor fue el NSGA2 quedando en segundo lugar el TA-MOEA. En el problema ZDT2 quedó primero el SPEA mientras el TA-MOEA obtiene el tercer lugar. En el problema ZDT3, el SPEA2 obtuvo el primer lugar y el TA-MOEA quedó segundo.

	Conjunto	ZDT4		Conjunto	ZDT5		Conjunto	ZDT6
1	TAMOEAE	892	1	SPEA2	854	1	SPEA2	690
2	SPEA2	875	2	NSGA2	853	2	NSGA2	670
3	SPEA	834	3	SPEA	850,5	3	CNSGA2	662,75
4	CNSGA2	791,25	4	TAMOEAE	830	4	TAMOEAE	631,5
5	NSGA2	782	5	CNSGA2	821	5	NPGA	582
6	MOGA	601	6	NPGA	580	6	NSGA	578
7	NPGA	590,75	7	NSGA	578	7	SPEA	547
8	NSGA	548	8	MOGA	532,25	8	MOGA	402

Tabla 3. Ranking considerando la métrica GVND sobre los problemas ZDT4, ZDT5 y ZDT6

En la Tabla 3 se observa las posiciones en los problemas ZDT4, ZDT5 y ZDT6 utilizando la métrica GVND. Como vemos para el problema ZDT4, el mejor fue el TA-MOEA. En el problema ZDT5, quedó primero el SPEA2 mientras el TA-MOEA obtiene el cuarto lugar. En el problema ZDT6 el SPEA2 obtiene el primer lugar y el TA-MOEA vuelve a quedar cuarto.

	Conjunto	ZDT1		Conjunto	ZDT2		Conjunto	ZDT3
1	NSGA2	940,25	1	SPEA	866	1	SPEA2	788
2	TAMOEAE	909,25	2	NSGA2	820	2	TAMOEAE	702
3	SPEA2	877,5	3	TAMOEAE	818	3	NSGA2	603
4	SPEA	801,75	4	SPEA2	687	4	SPEA	603
5	CNSGA2	780,75	5	CNSGA2	659	5	CNSGA2	589,25
6	NSGA	3	6	NSGA	6	6	NSGA	58,75
7	NPGA	1	7	NPGA	2	7	NPGA	33,75
8	MOGA	0	8	MOGA	1	8	MOGA	7,25

Tabla 4. Ranking considerando la métrica GRVND sobre los problemas ZDT1, ZDT2 y ZDT3

En la Tabla 4 se observa las posiciones en los problemas ZDT1, ZDT2 y ZDT3 utilizando la métrica GRVND. Para el problema ZDT1, el mejor fue el NSGA2 seguido del TA-MOEA en segunda posición. En el problema ZDT2, quedó primero el SPEA mientras el TA-MOEA obtiene el tercer lugar. En el problema ZDT3 el mejor fue el SPEA2 mientras el TA-MOEA vuelve a quedar segundo.

	Conjunto	ZDT4		Conjunto	ZDT5		Conjunto	ZDT6
1	CNSGA2	701,5	1	SPEA2	632	1	NSGA2	541
2	SPEA	699	2	CNSGA2	594	2	TAMOEAE	501,25
3	TAMOEAE	690	3	TAMOEAE	587	3	SPEA	477
4	SPEA2	657	4	SPEA	583,5	4	SPEA2	459
5	NSGA2	602	5	NSGA2	583	5	CNSGA2	430
6	NSGA	20	6	NSGA	23	6	NPGA	28
7	NPGA	12	7	NPGA	14	7	NSGA	23
8	MOGA	6	8	MOGA	7	8	MOGA	23

Tabla 5. Ranking considerando la métrica GRVND sobre los problemas ZDT4, ZDT5 y ZDT6

En la Tabla 5 se observa las posiciones en los problemas ZDT4, ZDT5 y ZDT6 utilizando la métrica GRVND. Ahora, en el problema ZDT4, el mejor es el CNSGA2, quedando en tercer puesto el TA-MOEA. En el problema ZDT5 quedó primero el SPEA2 mientras el TA-MOEA obtiene el tercer lugar. Por su parte, en el problema ZDT6 el primer lugar lo ocupa el NSGA2, quedando el TA-MOEA en segunda posición.

Generación de vectores no dominados (GVND)			Generación real de vectores no dominados (GRVND)		
	Conjunto	PROMEDIO		Conjunto	PROMEDIO
1	TAMOEA	855,875	1	TAMOEA	701,25
2	NSGA2	853	2	SPEA2	683,416667
3	SPEA2	837,833333	3	NSGA2	681,541667
4	SPEA	814,583333	4	SPEA	671,708333
5	CNSGA2	776,041667	5	CNSGA2	625,75
6	NPGA	571	6	NSGA	22,2916667
7	NSGA	535,458333	7	NPGA	15,125
8	MOGA	483,875	8	MOGA	7,375

Tabla 6. Posiciones promedio según la métrica GVND y la métrica GRVND

Haciendo un promedio global de los 6 problemas tratados, el TA-MOEA resulta claramente superior al considerar ambas métricas. En efecto, considerando la métrica GVND, obtuvo el primer lugar, seguido del NSGA2 y el SPEA2, como se aprecia a la izquierda de la Tabla 6. Por su parte, al considerar la métrica GRVND a la derecha de dicha figura, se observa que el primer lugar de nuevo es ocupado por el TA-MOEA, esta vez seguido del SPEA2 y el NSGA2.

Concluyendo esta sección, se puede afirmar que el TA-MOEA presenta experimentalmente soluciones más robustas ante un conjunto variado de problemas con características y dificultades diferentes, convirtiéndose en una opción válida para la resolución de problemas de optimización multiobjetivo, especialmente al considerar problemas nuevos con características no necesariamente conocidas.

6. Conclusiones y Trabajos Futuros

El presente trabajo propuso por primera vez una combinación de algoritmos evolutivos multiobjetivo en un contexto paralelo asíncrono utilizando un *Team Algorithm* implementado sobre una red de computadores.

A partir de los distintos resultados experimentales obtenidos, se han propuesto conclusiones parciales sobre la implementación del TA-MOEA con criterio de selección elitista, en la resolución de cada uno de los problemas considerados. Con esto, es posible arribar a las siguientes conclusiones finales de este trabajo:

- El diseño e implementación de un equipo de algoritmos evolutivos multiobjetivo paralelos es un problema complejo. Existen varias decisiones que tomar. El rango de éstas va desde el tipo de plataforma paralela en que se realizará la implementación, hasta la determinación de diversos parámetros de configuración y migración de datos.
- Teniendo en cuenta las métricas utilizadas para medir la calidad del conjunto Pareto calculado, se puede establecer que el TA-MOEA ha sido el que logró, en general, un mejor desempeño. Esta diferencia se verifica en toda la gama de problemas resueltos en este trabajo.
- Al utilizar TA-MOEA, se extiende el espacio de búsqueda. La recepción de elementos provenientes de distintos algoritmos introduce información genética en forma aleatoria que es útil para la obtención de mejores soluciones.

En resumen, a partir de las implementaciones consideradas en este trabajo, se recomienda el uso del Equipo Elitista de Algoritmos Evolutivos Multiobjetivo Paralelos (o TA-MOEA) por quedar mejor ubicado en el ranking global, teniendo en cuenta ambas métricas utilizadas, y además, por su capacidad de identificar razonablemente cual es el mejor MOEA para un determinado problema.

De forma a continuar con el trabajo iniciado, los siguientes tópicos son propuestos como trabajos futuros: implementación de TA-MOEA que incorporen criterios de selección más refinados y dinámicos; así como el estudio e implementación de nuevos problemas de ingeniería que sirvan de base comparativa y experimental para el TA-MOEA.

Referencias

- [1] T. Bäck, D. B. Fogel, y Z. Michalewicz, editores. "Handbook of Evolutionary Computation". Institute of Physics Publishing and Oxford University Press, 1997.
- [2] B. Barán, J. Vallejos, R. Ramos, y U. Fernández. "Multi-objective reactive power compensation". En 2001 IEEE/PES Transmission and Distribution Conference and Exposition, volume 1, pg. 97–101. IEEE, 2001.
- [3] B. Barán, E. Kaszkurewicz y D. M. Falcão. "Team Algorithms in Distributed Load Flow Computations". IEE Proceeding on Generation, Transmission and Distribution, Vol. 142, No. 6, pg. 583-588, noviembre 1995. Londres - Gran Bretaña.
- [4] B. Barán, N. Cáceres y E. Chaparro. "Reducción del Tiempo de Búsqueda utilizando una Combinación de Algoritmos Genéticos y Métodos Numéricos". XV International Conference of the Chilean Computer Science Society. Arica - Chile, 1995.
- [5] B. Barán, E. Kaszkurewicz y A. Bhaya. "Parallel Asynchronous Team Algorithms: Convergence and Performance Análisis". IEEE Transactions on Parallel & Distributed Systems, Vol. 7, No. 7, pg. 677-688, julio 1996. Estados Unidos.
- [6] B. Barán, E. Chaparro y N. Cáceres. "A-Teams en la Optimización del Caudal Turbinado de una Represa Hidroeléctrica". IBERAMIA-98, Lisboa-Portugal. 1998
- [7] B. Barán y F. Laufer. "Topological Optimization of Reliable Networks using A-Teams". Systemics, Cybernetics and Informatics SCI'99, Orlando - Florida, Estados Unidos. 1999.
- [8] C. A. Coello Coello y G. Toscano Pulido. "A Micro-Genetic Algorithm for Multiobjective Optimization". En E. Zitzler, K. Deb, L. Thiele, C. A. Coello Coello, y D. Corne, editores, First International Conference on Evolutionary Multi-Criterion Optimization, pg. 126–140. Springer-Verlag. Lecture Notes in Computer Science No. 1993, 2001.
- [9] C. A. Coello Coello y C. E. Mariano Romero. "Evolutionary Algorithms and Multiple Objective Optimization". En M. Ehrgott y X. Gandibleux, editores, Multiple Criteria Optimization: State of the Art Annotated Bibliographic Surveys, pg. 277–331. Kluwer Academic Publishers, Boston, 2002.
- [10] C. A. Coello Coello. "An Updated Survey of GA-Based Multiobjective Optimization Techniques". Technical Report Lania-RD-98-08, Laboratorio Nacional de Informática Avanzada (LANIA), Xalapa, Veracruz, México, December 1998.
- [11] C. A. Coello Coello. "Constraint handling through a multiobjective optimization technique". En A. S. Wu, editor, Proceedings of the 1999 Genetic and Evolutionary Computation Conference. Workshop Program, pg. 117–118, Orlando, Florida, Julio 1999.
- [12] J. Crichigno y B. Barán, "Multiobjective Multicast Routing Algorithm". IEEE ICT'2004, Ceará, Brasil, 2004.
- [13] K. Deb, S. Agrawal, A. Pratab, y T. Meyarivan. "A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II". KanGAL report 200001, Indian Institute of Technology, Kanpur, India, 2000.
- [14] K. Deb y T. Goel. "Controlled Elitist Non-dominated Sorting Genetic Algorithms for Better Convergence". En E. Zitzler, K. Deb, L. Thiele, C. A. Coello Coello, y D. Corne, editores, First International Conference on Evolutionary Multi-Criterion Optimization, pg. 67–81. Springer-Verlag. Lecture Notes in Computer Science No. 1993, 2001.
- [15] K. Deb. "Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems", Technical Report CI-49/98, Dortmund: Department of Computer Science/LS11, University of Dortmund, Germany, 1998.
- [16] K. Deb. "Evolutionary algorithms for multi-criterion optimization in engineering design". En K. Miettinen, M. M. Mäkelä, P. Neittaanmäki, y J. Periaux, editores, Evolutionary Algorithms in Engineering and Computer Science, pg. 135–161, Chichester, UK, 1999. JohnWiley & Sons, Ltd.
- [17] N. M. Duarte, A. E. Ruano, C. M. Fonseca, y P. J. Fleming. "Accelerating Multi-Objective Control System Design Using a Neuro-Genetic Approach". En 2000 Congress on Evolutionary Computation, volume 1, pg. 392–397, Piscataway, New Jersey, Julio 2000. IEEE Service Center.
- [18] S. Duarte y B. Barán. "Multiobjective Network Design Optimisation Using Parallel Evolutionary Algorithms". En XXVII Conferencia Latinoamericana de Informática CLEI-2001, Mérida, Venezuela, 2001.

- [19] C. M. Fonseca y P. J. Fleming. "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization". En S. Forrest, editor, *Proceedings of the Fifth International Conference on Genetic Algorithms*, pg. 416–423, San Mateo, California, 1993. University of Illinois at Urbana-Champaign, Morgan Kaufman Publishers.
- [20] D. E. Goldberg. "Genetic Algorithms in Search, Optimization and Machine Learning". Addison- Wesley Publishing Company, Reading, Massachusetts, 1989.
- [21] J. Horn y N. Nafpliotis. "Multiobjective Optimization using the Niche Pareto Genetic Algorithm". Technical Report IlliGAL Report 93005, University of Illinois at Urbana-Champaign, Urbana, Illinois, USA, 1993.
- [22] K. Miettien. "Some methods for nonlinear multi-objective optimization". En E. Zitzler, K. Deb, L. Thiele, C. A. Coello Coello, y D. Corne, editores, *First International Conference on Evolutionary Multi-Criterion Optimization*. Springer-Verlag. Lecture Notes in Computer Science No. 1993, 2001.
- [23] A. G. et al. "PVM: Parallel Virtual machine - A user's guide and Tutorial for Networked parallel Computing". M.I.T. press, Cambridge, MA, 1994.
- [24] N. Srinivas y K. Deb. "Multiobjective optimization using nondominated sorting in genetic algorithms". Technical report, Department of Mechanical Engineering, Indian Institute of Technology, Kanpur, India, 1993.
- [25] D. A. van Veldhuizen, J. B. Zydallis, y G. B. Lamont. "Issues in Parallelizing Multiobjective Evolutionary Algorithms for Real World Applications". En *Proceedings of the 17th ACM Symposium on Applied Computing*, pg. 595–602, Madrid, Spain, 2002. ACM Press.
- [26] D. A. van Veldhuizen. "Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations". PhD thesis, Department of Electrical and Computer Engineering. Graduate School of Engineering. Air Force Institute of Technology, Wright-Patterson AFB, Ohio, Mayo 1999.
- [27] E. Zitzler, K. Deb, L. Thiele, C. A. Coello Coello, y D. Corne, editores. "Proceedings of the First International conference on EMOO, 2001", Berlin, Germany, Marzo 2001. Springer-Verlag.
- [28] E. Zitzler, M. Laumanns, y L. Thiele. "SPEA2: Improving the Strength Pareto Evolutionary Algorithm", in K. Giannakoglou, D. Tsahalis, J. Periaux, P. Papailou and T. Fogarty (eds.) *EUROGEN 2001, Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems*, pp. 95--100, Athens, Greece, 2002.
- [29] E. Zitzler y L. Thiele. "An Evolutionary Algorithm for Multiobjective Optimization: The Strength Pareto Approach". Technical Report 43, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, Mayo 1998.
- [30] E. Zitzler, K. Deb, y L. Thiele. "Comparison of Multiobjective Evolutionary Algorithms on Test Functions of Different Difficulty". En A. S. Wu, editor, *Proceedings of the 1999 Genetic and Evolutionary Computation Conference. Workshop Program*, pg. 121–122, Orlando, Florida, Julio 1999.