

Algoritmos Evolutivos Multiobjetivo Combinados para la Optimización de la Programación de Bombeo en Sistemas de Suministro de Agua

Aldo Sotelo

Julio Basulado

Pedro Doldán

Benjamín Barán

Centro Nacional de Computación
Universidad Nacional de Asunción
Casilla de Correos 1439
Campus Universitario de San Lorenzo – Paraguay
Tel./Fax: (595)(21) 585 619 y 585 550
<http://www.cnc.una.py>

RESUMEN

La operación de estaciones de bombeo implica altos costos para las empresas de suministro de agua, en consecuencia, resulta importante reducir estos costos con una correcta programación del bombeo. Varios enfoques han sido presentados, demostrándose que pueden lograrse ahorros considerables. El presente trabajo propone su resolución mediante Algoritmos Genéticos Multiobjetivo. En ese contexto, dos reconocidos métodos fueron implementados y comparados: el Non-dominated Sorting Genetic Algorithm (NSGA) y el Strength Pareto Evolutionary Algorithm (SPEA), ambos combinados con un algoritmo heurístico de factibilización. Estos métodos apuntan a minimizar cuatro objetivos: el costo de la energía eléctrica consumida, el costo de mantenimiento de las bombas, la potencia máxima alcanzada (relacionada con el costo del sistema eléctrico y la potencia reservada) y el desnivel en el reservorio entre el inicio y el final del período de optimización. Resultados experimentales demuestran las ventajas de usar SPEA sobre los métodos manuales hoy utilizados y sobre el NSGA.

1. Introducción

La creciente demanda de consumo de agua en las ciudades hace que los sistemas de suministro se tornen cada vez más complejos. Dependiendo de las variables a ser consideradas y de los objetivos a ser tomados en cuenta, la tarea de optimizar la programación de bombeo se vuelve también una tarea complicada y de vital importancia, por lo que diversos autores vienen abordando este tema presentando distintos enfoques. En principio, programaciones del tipo lineal, no lineal, lineal mixta, entera, dinámica, entre otras, fueron aplicadas para optimizar un único objetivo: el costo de la energía eléctrica del bombeo (una revisión completa puede ser encontrada en Ormsbee y Lansey [1]). Por su parte, Lansey y Awumah [2] introdujeron el concepto del número de encendidos de las bombas como medida sustituta para evaluar el costo de mantenimiento de las mismas, siendo este el segundo objetivo considerado en la bibliografía existente. Recientemente, técnicas de Computación Evolutiva fueron introducidas al estudio de la programación de bombeo óptima. En efecto, Mackle et al. [3] realizaron una optimización de simple objetivo (costo de energía eléctrica), Savic et al. [4] propusieron la hibridación de los AG con un método de búsqueda local para la optimización de dos objetivos (costo de energía eléctrica y costo de mantenimiento de bombas), mientras que Schaezen [5] estableció penalidades a la violación de las restricciones del problema presentado como una optimización también de simple objetivo. De esta forma, varias áreas quedaron abiertas a la investigación, quedando demostrada la utilidad de la Computación Evolutiva en el problema de la optimización de la programación de bombeo.

Debido al gran avance obtenido recientemente en el campo de la optimización multiobjetivo [6] y a la probada utilidad de los mismos [4, 7], el presente trabajo propone la resolución de la programación óptima de bombeo como un problema de optimización multiobjetivo que por primera vez trata con 4 objetivos simultáneos.

Este trabajo está organizado de la siguiente forma: en la sección 2 se presenta una descripción del problema de minimización con cuatro objetivos. La sección 3 presenta los métodos de resolución computacional implementados, abarcando dos Algoritmos Evolutivos Multiobjetivo (MOEA: *Multiobjective Evolutionary Algorithm*): el *Non-dominated Sorting Genetic Algorithm* (NSGA) y el *Strength Pareto Evolutionary Algorithm* (SPEA). Además, debido a la particularidad del problema, se introduce la combinación de estos métodos con un algoritmo heurístico de factibilización que corrige las soluciones no factibles obtenidas en cada generación. En la sección 4 son analizados los resultados experimentales considerando un problema paradigma para una estación con 5 bombas. Finalmente, la sección 5 resume las conclusiones del trabajo y las recomendaciones para trabajos futuros.

2. Problema de la programación óptima de bombeo

2.1. Conceptos básicos

La demanda de agua de una población es variable en el tiempo. La cantidad de agua a suministrar para satisfacer dicha demanda deberá, por lo tanto, ser también variable en el tiempo. En general, una estación de bombeo cuenta con un conjunto de bombas de diferentes capacidades que bombean agua a uno o más reservorios. Estas bombas trabajan combinadas para bombear la cantidad de agua necesaria, atendiendo a las restricciones del problema (como la capacidad máxima del reservorio). Por lo tanto, dependiendo del momento, algunas bombas estarán encendidas y otras apagadas.

Programar el bombeo en una estación consiste en establecer la combinación de bombas a utilizarse en cada intervalo de tiempo del horizonte de planificación. Luego, una programación de bombeo es el conjunto de todas las combinaciones de bombas a utilizarse durante cada intervalo del horizonte de planificación [3]. Una programación óptima de bombeo puede definirse entonces, como una programación que cumpla con las restricciones del problema (como atender la demanda), pero que además optimice los objetivos establecidos.

2.2. Modelo hidráulico

Un modelo hidráulico simplificado fue escogido para el presente trabajo. El mismo consiste de:

- una fuente inagotable de agua potable;
- una estación de bombeo constituida por n bombas que impulsan el agua desde la fuente hasta el reservorio;
- un reservorio o tanque de almacenamiento de agua;
- una tubería aductora que conduce el agua desde la estación de bombeo hasta el reservorio.

La hidráulica que existe a partir del reservorio no es considerada. El único dato necesario de esta parte del sistema es la demanda de consumo, la cual es abastecida desde el reservorio. Este modelo hidráulico se presenta en la Figura 1.

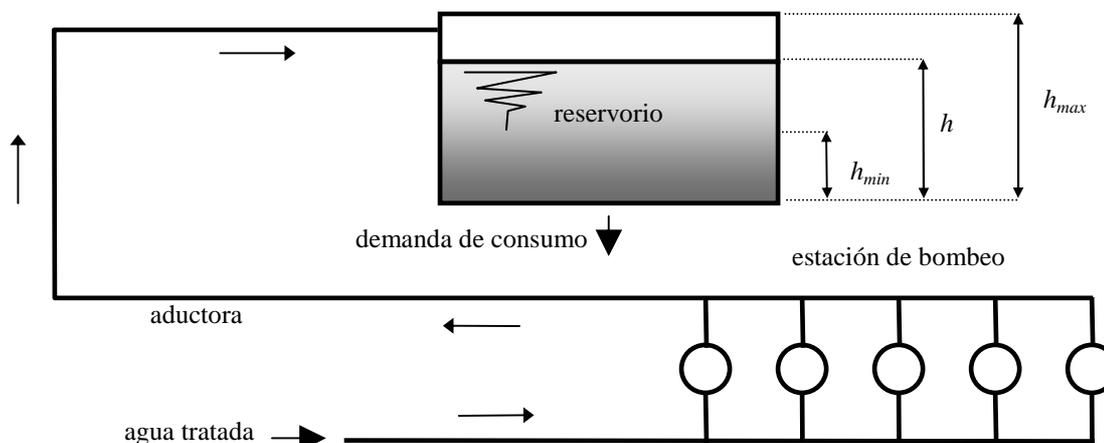


Figura 1: Modelo hidráulico adoptado.

2.3. Período de optimización e intervalos de tiempo

El presente trabajo considera un período de optimización de un día, tomando como dato el patrón de la demanda de consumo histórico para un día promedio. El período de optimización fue dividido en 24 intervalos de 1 hora. Se asume que las bombas pueden ser apagadas o encendidas solo al inicio de cada intervalo. Intervalos de menor duración pueden ser considerados cuando requeridos, a pesar de que resulta usual la utilización de intervalos de 1 hora.

2.4. Objetivos de la programación

La programación de bombeo propone optimizar el costo de bombeo conformado por: el costo de la energía eléctrica y el costo de mantenimiento de las bombas, atendiendo siempre a las restricciones del sistema [4]. De hecho, estos costos de bombeo constituyen dos de los objetivos a ser optimizados en este trabajo, el cual, introduce además otros dos objetivos adicionales: la diferencia de nivel en el reservorio entre el final y el inicio del período de optimización, que es en realidad una restricción introducida como objetivo (mientras que en [4] había sido tratado mediante la aplicación de penalidades), y la potencia máxima alcanzada.

2.4.1. Costo de la energía eléctrica consumida (f_1)

El costo de energía eléctrica consumida se refiere al gasto que significa el consumo de energía eléctrica por parte de las bombas de la estación de bombeo.

Un factor importante en el costo de la energía eléctrica es la estructura tarifaria de la misma. En muchos sistemas de suministro de electricidad el costo de la energía eléctrica no es el mismo en todas las horas del día, dado que existen horas de mayor consumo y otras de menor consumo; sin embargo, las instalaciones deben dimensionarse para el consumo máximo. Esto es conocido como tarifa diferenciada. En este trabajo se considerará la siguiente estructura tarifaria:

- tarifa barata (T_f): de 0 a 17 hs. y de 22 a 24 hs., llamado horario fuera de punta de carga;
- tarifa cara (T_p): de 17 a 22 hs., llamado horario de punta de carga.

Cabe destacar que la influencia de esta diferencia tarifaria dentro de la programación de bombeo es notable, ya que se reducen los costos de consumo de energía eléctrica si la programación de bombeo óptima establece la menor cantidad posible de bombas encendidas durante el horario de punta de carga, cuando la tarifa es mayor, haciendo uso de la capacidad útil del reservorio. También cabe mencionar que no solo es posible utilizar una doble tarifa de energía eléctrica sino que pueden tenerse tres o más tarifas dependiendo del caso. Matemáticamente, el costo de energía eléctrica consumida C_e está dado por la ecuación:

$$C_e = T_f \sum_{i=1}^{17} c(b_i) + T_p \sum_{i=18}^{22} c(b_i) + T_f \sum_{i=23}^{24} c(b_i) \quad (1)$$

donde:

- i intervalo de tiempo del período de optimización, utilizado como unidad
- b_i combinación de bombas en el intervalo i ($b_i \in \mathbf{B}^n$, $\mathbf{B} \in \{0,1\}$, ver ejemplo en 2.8)
- n número de bombas
- $c(b_i)$ electricidad consumida en el intervalo i , con la combinación b_i

2.4.2. Costo de mantenimiento de las bombas (f_2)

Los costos de electricidad no constituyen el único criterio utilizado por los operadores para juzgar la calidad de una programación de. El costo de mantenimiento, en muchos casos, es igual o más importante. En [2] se introduce el concepto del número de encendidos de las bombas como forma de medir el costo de mantenimiento. Es decir, el desgaste de una bomba puede ser medido indirectamente a través del número de veces que la misma ha sido encendida. Un encendido se considera como tal solo en el caso en que la bomba haya estado apagada en el intervalo de tiempo anterior. Si la bomba ya estaba encendida en el intervalo anterior y continúa estándolo o se apaga en el intervalo siguiente, no constituye un encendido. De esta forma, al disminuir el número de encendidos se estará disminuyendo indirectamente dicho costo. Para determinar el número total de encendidos N_e simplemente se cuenta el número de encendidos en cada intervalo y a esto se suma la mitad del número de encendidos entre el primer y el último intervalo; esto último, de manera a considerar los encendidos que hubieran entre el día anterior y el actual, suponiendo cierta periodicidad en las programaciones. Lo explicado anteriormente puede ser expresado como:

$$N_e = \sum_{i=1}^{24} \left\| \max\{0; (b_i - b_{i-1})\} \right\|_1 + \frac{\left\| \max\{0; (b_1 - b_{24})\} \right\|_1}{2} \quad (2)$$

2.4.3. Diferencia de nivel en el reservorio entre el inicio y el final del período de optimización (f₃)

En general, un modelo hidráulico como el presentado en la figura 1 comprende las siguientes restricciones:

- la cantidad de agua que puede ser suministrada por la fuente;
- las condiciones de presión necesarias en la red;
- las características de las válvulas y bombas del sistema;
- la demanda de consumo de agua, teniendo en cuenta tanto la cantidad como la distribución de la misma durante el período de optimización;
- el nivel de agua dentro del reservorio o tanque, el cual considera:
 - a. un nivel mínimo que garantice suficiente presión en las tuberías y que deba mantenerse por motivos de seguridad en caso de incendios u otros imprevistos que demanden gran cantidad de agua en poco tiempo,
 - b. un nivel máximo que evite pérdidas en las tuberías y sea compatible con la capacidad del reservorio,
 - c. un nivel inicial que debe tratar de ser recuperado nuevamente al final del período de optimización.

Para fines del presente trabajo se supondrá que la fuente puede proporcionar toda el agua que sea necesaria en cualquier momento y sin costos adicionales. También se supondrá que las condiciones de presión en las tuberías son satisfechas cualquiera sea el nivel de agua en el tanque, o sea, que los niveles fluctuantes en el tanque proporcionan la suficiente altura (presión mínima) como para que el agua llegue a los usuarios, venciendo las alturas geométricas y las pérdidas en las tuberías (pérdidas por fricción y pérdidas locales) y sin provocar presiones que produzcan pérdidas en la red debido a roturas (presión máxima). El efecto causado debido a la utilización de las válvulas tampoco será tomado en cuenta. La cantidad y distribución de la demanda de consumo será considerada al adoptar una curva de demanda a abastecer.

Los niveles máximo y mínimo de agua en el reservorio serán considerados como restricciones propiamente dichas, de manera que el nivel en el reservorio al final de cada intervalo h_i , deberá ser menor o igual que el nivel máximo h_{max} y mayor o igual que el nivel mínimo h_{min} . Sin embargo, la diferencia de nivel en el reservorio entre el final y el inicio del período de optimización Δh , será planteada como un objetivo adicional a minimizar, esto es:

$$\Delta h = \sum_{i=1}^{24} [q(b_i) - d_i] / S \quad (3)$$

con:

$$h_i = h_{i-1} + [q(b_i) - d_i] / S \quad (4)$$

sujeto a:

$$\begin{aligned} h_i &\leq h_{max} \\ h_i &\geq h_{min} \end{aligned}$$

donde:

S superficie del reservorio
 $q(b_i)$ caudal bombeado en el intervalo i
 d_i demanda de consumo en el intervalo i

2.4.4. Potencia máxima alcanzada (f₄)

Muchas empresas de suministro de energía eléctrica facturan a los usuarios de gran envergadura basándose en la potencia reservada por los mismos y cobrando un elevado costo adicional al máximo pico de potencia que haya sobrepasado dicho valor contratado. Es por tanto de gran importancia, reducir estos picos de potencia para evitar dichos sobre-costos. Como una primera aproximación a dicho problema, en este trabajo se buscará obtener programaciones de bombeo óptimas que utilicen las combinaciones de bombas que minimicen los picos diarios de potencia. Es decir, se introducirá la potencia máxima alcanzada en el día P_{max} como un objetivo a ser minimizado. El cálculo de la misma se realiza utilizando la siguiente ecuación:

$$P_{max} = \max[p(b_i)] \quad (5)$$

donde:
 $p(b_i)$ potencia utilizada en el intervalo i

En conclusión, el presente trabajo toma en consideración las características de las bombas componentes del sistema (sus capacidades de bombeo) de manera a abastecer la demanda de consumo de la población (variable en el tiempo) sin violar los niveles máximo y mínimo del reservorio y tratando de recuperar el nivel inicial al final del día, al mismo tiempo que se minimizan los costos de energía eléctrica consumida y de mantenimiento de las bombas, así como la potencia máxima alcanzada.

2.5. Bombas

En general, el problema podría considerarse sobre la base de un sistema con n bombas centrífugas de velocidad constante. Las mismas pueden ser encendidas o apagadas solamente al inicio de cada hora. Las capacidades de bombeo son consideradas constantes para todas las combinaciones durante los intervalos. Esto porque la fluctuación del nivel en el tanque (variación de altura geométrica) influye muy poco en el caudal bombeado (aproximadamente $\pm 5\%$) cuando se consideran diferencias de altura geométrica muy grandes entre la estación de bombeo y el reservorio, así como tuberías aductoras largas que implican grandes pérdidas de presión. Entonces, cuando las bombas tienen que vencer una gran presión para bombear cierto caudal, una variación pequeña de la altura geométrica, como lo es la fluctuación del nivel en el reservorio, produce también variaciones muy pequeñas en el caudal bombeado, que pueden no ser consideradas. De esta manera, a cada combinación de bombas se puede asignar un volumen de agua bombeado, energía eléctrica consumida y potencia eléctrica utilizada, siempre tomados sobre la base de un intervalo de 1 hora [8 - 10].

Debe también destacarse que las bombas se encuentran asociadas en paralelo, es decir, trabajan a igual presión. El caudal bombeado por una combinación, aunque algunos autores lo consideran como aproximadamente lineal, no es la suma directa de los caudales bombeados por separado por cada una de las bombas componentes de dicha combinación, lo cual no será restringido en este trabajo que lo considera no lineal.

2.6. Reservorio

Debe mencionarse que, por continuidad, el reservorio debe terminar su ciclo diario al mismo nivel con respecto a aquel con el que comenzó el día, de manera a asegurar dos cosas:

- que no fue bombeada agua demás para elevar el nivel final del reservorio con respecto al inicial. Este desnivel es agua acumulada en el tanque que no fue utilizada y por tanto aumenta el costo de bombeo inútilmente;
- que no se utilizó agua ya almacenada en el reservorio, quedando el mismo con un nivel final por debajo del inicial. Este desnivel es agua que deberá ser recuperada luego a través del bombeo y que afectará sustancialmente las condiciones iniciales y la programación del siguiente período.

En síntesis, debe existir un balance o equilibrio de las masas de agua que entran y salen del reservorio entre el inicio y el final del período de optimización. El modelo matemático adoptado es entonces el de balance de masas [1], ya que por las características del problema, es el que más se adecua al sistema estudiado. Luego, el reservorio posee un volumen útil V_{util} correspondiente a una altura útil $h_{util} = h_{max} - h_{min}$ y a una superficie S . Una ventaja adicional de este enfoque es la periodicidad, es decir, poder utilizar una misma programación de bombeo para varios días consecutivos, si la demanda de consumo no varía sustancialmente.

2.7. Demanda de Consumo

Siendo la demanda de consumo un dato de entrada del problema, la misma debe ser obtenida a priori de fuentes confiables, ya que la calidad y la aplicabilidad de los resultados arrojados por el programa dependerán, evidentemente, de cuán reales sean las predicciones de la demanda de consumo. Dicha demanda es obtenida a través del estudio estadístico del comportamiento de la población abastecida por el reservorio durante varios años. Es por ello una condición necesaria el tener bien delimitada la zona de influencia del reservorio. Luego, a través de estos estudios estadísticos se podría establecer la demanda pronosticada para un día determinado de acuerdo a ciertos parámetros. Varios modelos para determinar la demanda de consumo son presentados en [1]. El presente trabajo adopta una curva de demanda similar a la utilizada en [4] (ver Figura 2).

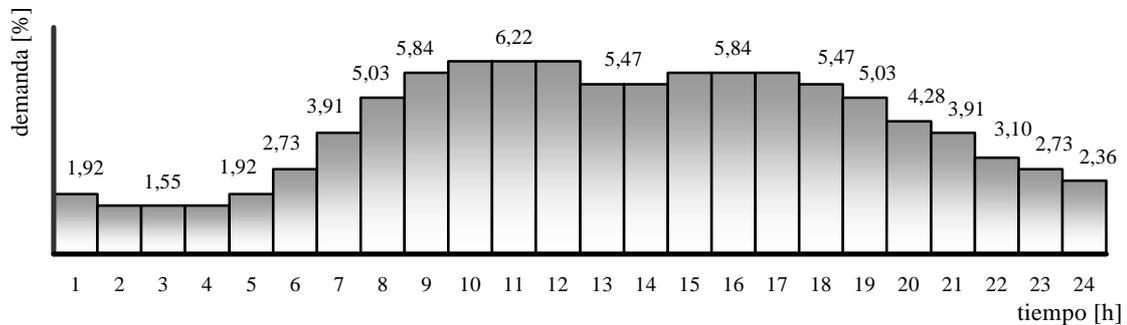


Figura 2: Curva de demanda de consumo adoptada.

2.8. Codificación

El principal requisito para la codificación de un problema es que la misma pueda representar cada solución potencial [11]. Esto se logra mediante una representación adecuada de los parámetros del problema y la unión de los mismos en una cadena o *string*. Los parámetros codificados son llamados genes y la cadena de genes es conocida como cromosoma. En este caso, un cromosoma representa a un individuo o solución posible del problema.

El alfabeto binario fue adoptado para codificar el problema de la programación de bombeo óptima ya que resulta muy apropiado. Cada bomba, durante cada intervalo de tiempo, es representada por un bit de la cadena. Luego, si el valor del bit es cero, la bomba representada por el mismo estará apagada durante este intervalo de tiempo. Si el valor es uno, la bomba estará encendida. Para cada intervalo de tiempo son necesarios n bits para representar todas las combinaciones de bombeo posibles, siendo n el número de bombas de la estación. Para este trabajo se utilizará $n = 5$ conforme muestra la Figura 1. De esta manera, una cadena consistente en $5 \times 24 = 120$ bits, describe una solución completamente (ver Figura 3). Por lo tanto, el número total de soluciones posibles es $2^{120} \cong 10^{36}$, lo cual demuestra que este espacio de búsqueda no podría ser tratado por métodos tradicionales. Sin embargo, debido a las restricciones del problema, muchas de estas soluciones no serán factibles ya que no cumplirán con los niveles máximo y mínimo establecidos en el reservorio. De ahí surge la necesidad de utilizar un algoritmo de factibilización que corrija las soluciones no factibles, de manera que cumplan con las restricciones. Este procedimiento hará que el espacio de búsqueda de soluciones quede reducido al espacio de las soluciones factibles arrojadas por dicho algoritmo.

hora	1					...	24				
bomba	B1	B2	B3	B4	B5	...	B1	B2	B3	B4	B5
bit	0	1	1	0	1	...	1	0	0	0	1
significado	apagada	encendida	encendida	apagada	encendida	...	encendida	apagada	apagada	apagada	encendida

Figura 3: Codificación del problema de la programación óptima de bombeo para $n = 5$ bombas.

2.9. Formulación matemática del problema

$$\text{Minimizar } y = f(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x}), f_4(\mathbf{x}))$$

donde:

f_1 : costo de energía eléctrica consumida; ver ecuación (1).

f_2 : número de encendidos de las bombas; ver ecuación (2).

f_3 : diferencia de nivel en el reservorio; ver ecuación (3).

f_4 : potencia máxima alcanzada; ver ecuación (5).

sujeto a:

$$h_i = h(x_i) \leq h_{max}$$

$$h_i = h(x_i) \geq h_{min}, \text{ para cada intervalo } i$$

h_i : nivel en el reservorio al final del intervalo i ; ver ecuación (4).

donde:

$\mathbf{x} \in X \subseteq B^{120}$ es el vector de decisión, $B = \{0,1\}$;

$y = (y_1, y_2, y_3, y_4) \in Y \subset \mathbb{R}^4$ es el vector objetivo;

El conjunto de las mejores soluciones y es conocido como Frente Pareto FP .

3. Resolución utilizando MOEA

El presente trabajo enfoca la resolución del problema de la programación óptima de bombeo a través de MOEA [6]. Dos métodos fueron implementados: el SPEA y el NSGA.

3.1. SPEA

Un nuevo enfoque basado en el *Strength Pareto Evolutionary Algorithm* fue desarrollado para este trabajo. Este método, muy relacionado con los Algoritmos Genéticos [11], almacena en una población externa P' los mejores individuos (no dominados) encontrados en una población general evolutiva P en cada iteración. La asignación del *fitness* (asociado a la probabilidad de ser seleccionado para una próxima generación) para los individuos de P' y de P se realiza según reglas diferentes, de forma que el mínimo *fitness* asignado en la población externa P' sea mayor que cualquier *fitness* asignado en la población evolutiva P . Esto porque en el SPEA implementado en este trabajo, cuanto mayor sea el *fitness* de un individuo mayor será la probabilidad de que el mismo sea seleccionado. Además, ambas poblaciones se juntan para aplicar los operadores genéticos, manteniéndose constante el tamaño de la población P .

El SPEA preserva la diversidad en la población usando relaciones de dominancia Pareto e incorporando un procedimiento de *clustering* para reducir el conjunto de no dominados sin destruir sus características. En general, el *clustering* particiona un conjunto de m elementos en g grupos de elementos relativamente homogéneos, donde $g < m$, seleccionando un individuo representante por cada uno de los g *clusters*. Así, un número determinado de g individuos puede mantenerse en la población externa preservando las principales características del Frente Pareto [12].

Una importante característica del SPEA es su propiedad de convergencia, una característica no siempre presente en otros MOEAs. Consecuentemente, el algoritmo implementado en este trabajo está basado en el SPEA original presentado en [12] pero difiere del él en los siguientes aspectos:

- *Incorporación de un algoritmo heurístico de factibilización*: el cual corrige las soluciones no factibles obtenidas en cada iteración de forma que cumplan con las restricciones del problema. El algoritmo analiza individuo por individuo a la población evolutiva (cadena de 120 bits) y opera de la siguiente manera:
 - en cada individuo, analiza la factibilidad, intervalo por intervalo (de 5 bits) correspondiente a cada hora, aplicando la restricción (4);
 - si en alguno de los intervalos no se cumple alguna de las restricciones, procede a corregir toda la subcadena de bits desde el inicio hasta el intervalo donde se infringió la restricción. Existen 2 posibilidades a considerar:
 - a) si $h_i < h_{min}$: implica que se necesita mas agua de la que se bombeó, es decir, se deben prender bombas que en la subcadena se encontraban apagadas (ceros). Para ello el programa ubica la posición de estos ceros y los cambia por unos en forma gradual y aleatoria (bombas prendidas) hasta que se cumpla la restricción infringida. El cambio de un bit de 0 a 1 es aceptado solamente si el mismo no origina una violación de la restricción $h_i \leq h_{max}$ en intervalos anteriores al considerado (lo cual puede ocurrir debido a la forma de la demanda de consumo);
 - b) si $h_i > h_{max}$: implica que se necesita menos agua de la que se bombeó, es decir, se deben apagar bombas que en la subcadena se encontraban prendidas (unos). Para ello el programa ubica la posición de estos unos y los cambia por ceros en forma gradual y aleatoria (bombas apagadas) hasta que se cumpla la restricción infringida. El cambio de un bit de 1 a 0 es aceptado solamente si el mismo no origina una violación de la restricción $h_i \geq h_{min}$ en intervalos anteriores al considerado (lo cual puede ocurrir debido a la forma de la demanda de consumo);
 - en caso que un individuo no pueda ser factibilizado el programa emite el mensaje de error correspondiente y propone una solución, por ejemplo, ampliar la capacidad de almacenamiento del reservorio.
- *Doble criterio de parada*: uno de los criterios detiene al algoritmo si un cierto número máximo de generaciones es alcanzado. El otro criterio detiene al algoritmo si no se incorporan nuevas soluciones no dominadas a la población externa del SPEA luego de transcurrido un cierto número de generaciones.

El diagrama de flujo del SPEA propuesto se presenta en la Figura 4.

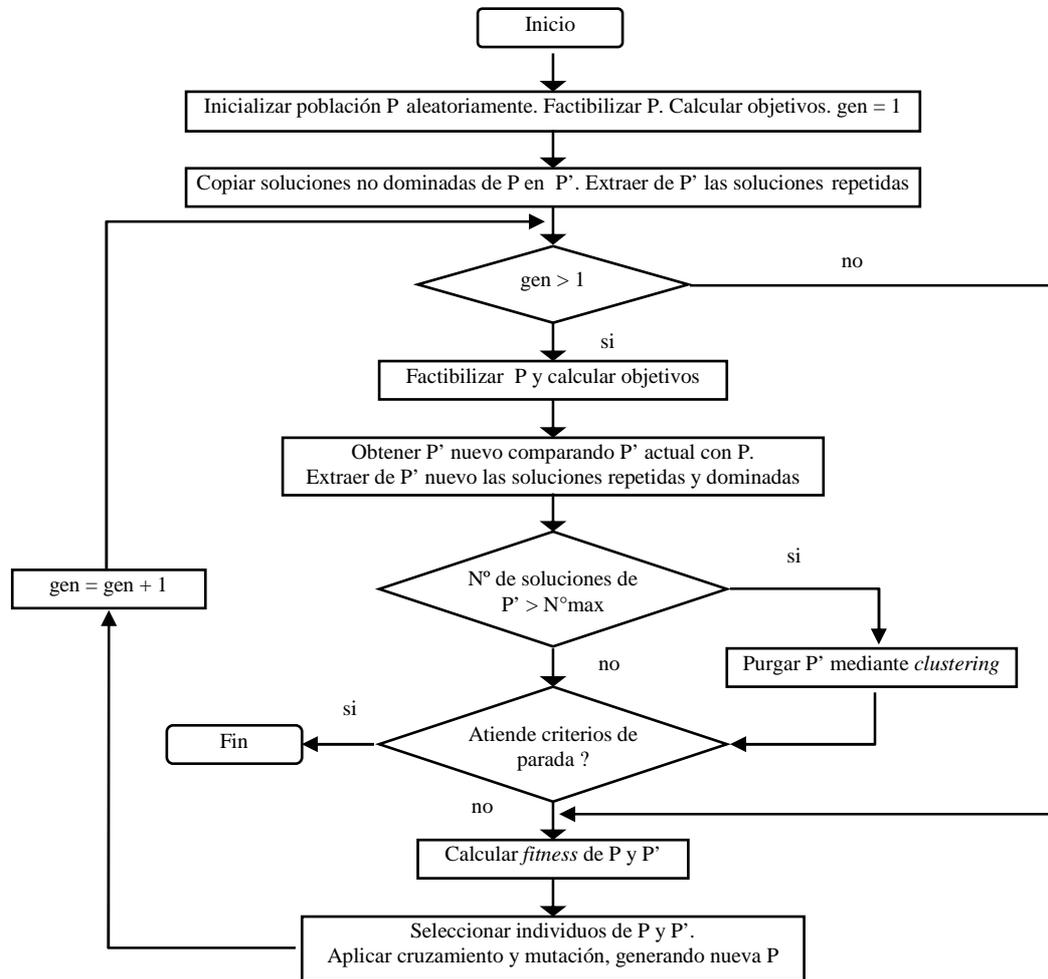


Figura 4: Diagrama de flujo del SPEA propuesto.

3.2. NSGA

Un enfoque basado en el *Nondominated Sorting Genetic Algorithm* también fue desarrollado en este trabajo. El NSGA original presenta dos diferencias principales respecto al SPEA. La primera es que no trabaja con una población externa, solo con la población general evolutiva. La segunda es la forma de asignar el *fitness*. Para ello se clasifica la población en rangos de acuerdo a niveles de dominancia. Varios individuos pueden pertenecer al mismo rango. Luego, se asigna un *fitness* virtual a todos los individuos del mismo rango. Por último, se comparte el *fitness* de los individuos de un mismo rango usando *fitness sharing*. De esta manera, *fitness* decrecientes son asignados a los individuos de forma que el mínimo *fitness* asignado en cierto rango sea ligeramente superior al máximo *fitness* asignado en el rango inmediatamente inferior.

El algoritmo implementado en este trabajo está basado en el NSGA original presentado en [13] pero difiere del él en los siguientes aspectos:

- *Incorporación de un algoritmo heurístico de factibilización*: el mismo utilizado en el SPEA.
- *Porcentajes de cruzamiento y mutación variables*: estos porcentajes varían con el paso de las generaciones, decreciendo linealmente a partir de un valor inicial hasta hacerse cero en la última generación, esto es:

$$Pc = Pc_{inicial} \times (1 - gen / genmax)$$

$$Pm = Pm_{inicial} \times (1 - gen / genmax)$$

El diagrama de flujo del NSGA propuesto se presenta en la Figura 5.

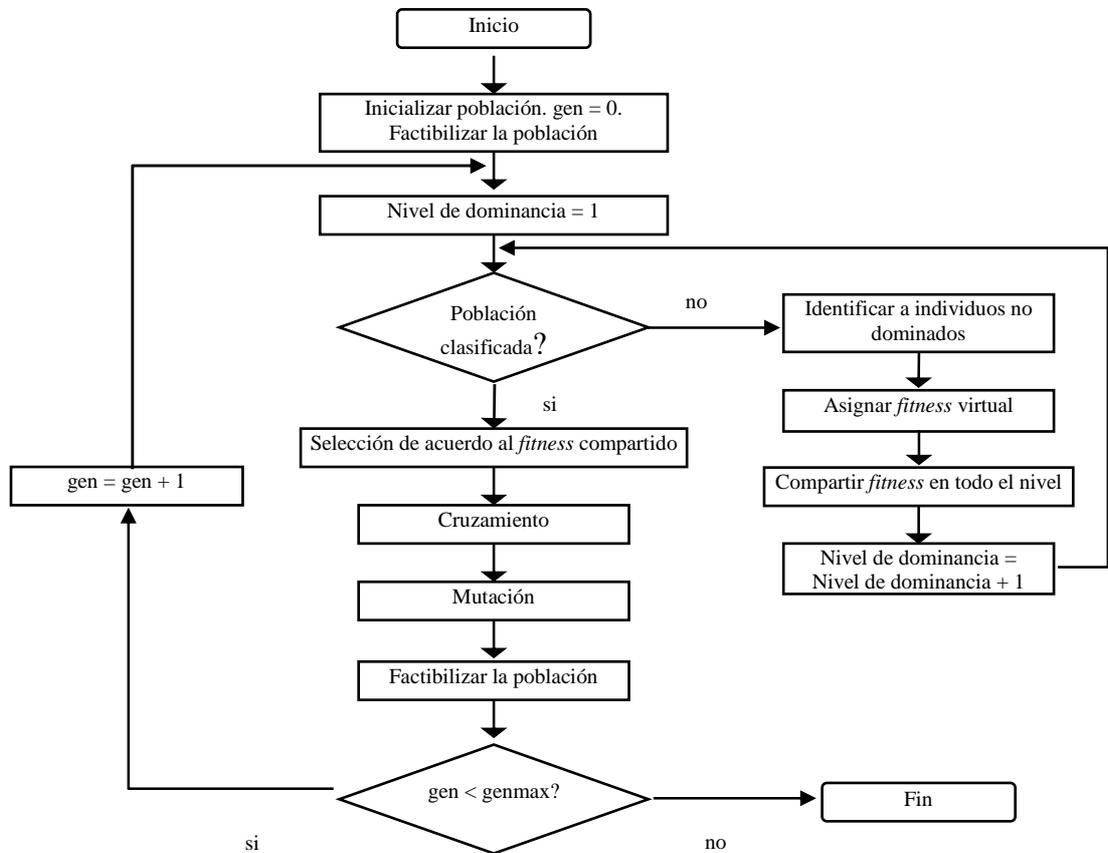


Figura 5: Diagrama de flujo del NSGA propuesto.

4. Experimentación y resultados

4.1. Problema paradigma

Para fines de este trabajo, se presentan los resultados correspondientes a un problema ejemplo con las siguientes características:

- una estación de bombeo con $n = 5$ bombas. Las características de las combinaciones se presentan en la Tabla 1;
- un reservorio con $S = 2600 \text{ m}^2$, $h_{max} = 7 \text{ m}$, $h_{min} = 1 \text{ m}$, $h_{inicial} = 3 \text{ m}$, $V_{util} = 15.600 \text{ m}^3$;
- una curva de demanda de consumo como la de la figura 2. Siendo el caudal total demandado $Q = 54.788 \text{ m}^3/\text{día}$;
- una estructura tarifaria con $T_p = 2T_f$.

Tabla 1: Características de las combinaciones de bombas utilizadas en el problema paradigma.

i	Códi-go	Caudal [m³/h]	Potencia [kW]
0	00000	0	0
1	00001	1800	595
2	00010	828	260
3	00011	2600	855
4	00100	828	260
5	00101	2600	855
6	00110	1650	520
7	00111	3450	1115

id	Códi-go	Caudal [m³/h]	Potencia [kW]
8	01000	1440	445
9	01001	3235	1040
10	01010	2260	705
11	01011	4060	1300
12	01100	2260	705
13	01101	4060	1300
14	01110	3090	965
15	01111	4890	1560

id	Códi-go	Caudal [m³/h]	Potencia [kW]
16	10000	1800	595
17	10001	3600	1190
18	10010	2620	855
19	10011	4420	1450
20	10100	2620	855
21	10101	4420	1450
22	10110	3450	1115
23	10111	5250	1710

id	Códi-go	Caudal [m³/h]	Potencia [kW]
24	11000	3235	1040
25	11001	5035	1635
26	11010	4060	1300
27	11011	5860	1895
28	11100	4060	1300
29	11101	5860	1895
30	11110	4890	1560
31	11111	6690	2155

4.2. Métricas de comparación

Para evaluar los resultados experimentales utilizando el SPEA y el NSGA, fue utilizado un subconjunto de métricas propuestas en [7]. Esto porque una sola métrica no refleja el comportamiento total, la efectividad y la eficiencia de los MOEA. Como estas métricas reflejan la semejanza entre el verdadero conjunto Frente Pareto óptimo $FP_{verdadero}$ y un conjunto Frente Pareto computado $FP_{computado}$, una buena aproximación del $FP_{verdadero}$ es obtenida juntando todos los individuos no dominados de todos los métodos implementados, considerando todas las corridas. En otras palabras $FP_{verdadero}$ es obtenido mediante la unión de las mejores soluciones conocidas en todos los experimentos realizados. El conjunto de métricas comprende las siguientes:

4.2.1. Overall nondominated vector generation (N)

$$N = \left| FP_{computado} \right|_c \quad (6)$$

donde $\left| \cdot \right|_c$ denota cardinalidad.

Esta métrica indica el número de soluciones en $FP_{computado}$. Se espera que un buen conjunto $FP_{computado}$ tenga un gran número de individuos de manera a ofrecer una amplia variedad de opciones al ingeniero.

4.2.2. Overall nondominated vector generation ratio (ONVGR)

$$ONVGR = \frac{N}{\left| FP_{verdadero} \right|_c} \quad (7)$$

Denota la relación entre el número de soluciones en $FP_{computado}$ y el número de soluciones en $FP_{verdadero}$. Como el objetivo es obtener un conjunto $FP_{computado}$ tan similar como sea posible a $FP_{verdadero}$, un valor cercano al 1 sería el deseado.

4.2.3. Error ratio (E)

$$E = \frac{\sum_{i=1}^N e_i}{N} \quad (8)$$

$$e_i = \begin{cases} 0 & \text{si un vector en } FP_{computado} \text{ también está en } FP_{verdadero} \\ 1 & \text{en caso contrario} \end{cases}$$

Esta relación da la proporción de los vectores objetivos en $FP_{computado}$ que no son miembros de $FP_{verdadero}$. Por lo tanto, un Error Ratio E cercano a 1 indica una pobre correspondencia entre $FP_{computado}$ y $FP_{verdadero}$. En efecto, $E = 0$ sería el deseado.

4.2.4. Maximum pareto front error (ME)

$$ME = \max_j \left(\min_i \|y^i - y^j\|_\infty \right) \quad (9)$$

$$y^i \in FP_{verdadero} ; y^j \in FP_{computado}$$

Esta métrica indica la máxima franja de error la cual, cuando es considerada con respecto a $FP_{computado}$, abarca a cada vector en $FP_{verdadero}$. Idealmente, $ME = 0$ sería lo deseado.

4.3. Ambiente computacional

Se utilizaron dos PC para realizar las experimentaciones: una AMD-K6 3D de 350 MHz de velocidad y 128 MB de memoria RAM (denotada como PC1) y una pentium III de 600 MHz de velocidad y 128 MB de memoria RAM (en adelante, PC2). Tanto las corridas del SPEA como las del NSGA, se realizaron en ambas computadoras personales. Para implementar los algoritmos fue utilizado el lenguaje MATLAB versión 5.3, funcionando en un ambiente Windows.

4.4. Resultados experimentales

Se hicieron varias corridas para distintos valores de los parámetros considerados utilizando ambos métodos implementados. Por motivos de espacio, en la Tabla 2 son presentados los resultados de las métricas para dos

corridas típicas del SPEA y del NSGA, aplicados al problema paradigma, así como la solución propuesta por un especialista.

Tabla 2: Cálculo de métricas a partir de los resultados experimentales, para el problema paradigma.

Método	Plataforma	N	$ONVGR$	E	ME
Especialista	Manual	1	0,0017	1	0,0636
SPEA combinado	PC1	790	1,3101	0,79	0,4551
	PC2	438	0,7264	0	0
NSGA combinado	PC1	22	0,0365	1	0,5718
	PC2	73	0,1211	1	0,6097

Para el cálculo de las métricas de la Tabla 2 se utilizó un conjunto Frente Pareto óptimo calculado a partir de la unión de todas las corridas realizadas, obteniéndose un $FP_{verdadero}$ de 603 individuos, como resultado de la dominancia de algunas soluciones de una corrida sobre las soluciones obtenidas en otras corridas. En efecto, la Tabla 2 muestra que el SPEA corriendo en PC1 obtuvo un número mayor de soluciones (790) que las existentes en $FP_{verdadero}$ pero el 79% de ellas no pertenecen a dicho conjunto. Por su parte, el SPEA corriendo en PC2 obtuvo un menor número de soluciones (438) pero todas pertenecen a $FP_{verdadero}$. Por otro lado, el NSGA corriendo en PC1 obtuvo más soluciones (73) que el mismo corriendo en PC2 (22), pero en ninguno de los casos se encontraron soluciones pertenecientes a $FP_{verdadero}$. Finalmente, cabe destacar que la solución propuesta por el especialista tampoco pertenece al $FP_{verdadero}$. Como consecuencia de lo arriba mencionado, el *Error Ratio* E es nulo para el SPEA corriendo en PC2, y toma su valor máximo de 1 para las corridas con el NSGA y para la solución propuesta por el especialista. Lógicamente, el SPEA corriendo en PC2 consigue un valor óptimo de $ME = 0$, mientras que las demás corridas resultan en mayores valores del *Maximum Pareto Front Error*. De entre estas corridas con $ME > 0$, la solución propuesta por el especialista, a pesar de no ser óptima, es la de menor ME , lo que puede ser considerado como un indicio de la experiencia del especialista en la programación de bombeo, aun sin contar con los métodos computacionales aquí propuestos. En conclusión, de los métodos implementados y experimentados, el SPEA corriendo en la plataforma con mejores recursos, se erige como el de mejor desempeño.

Las mejores soluciones obtenidas en todas las corridas realizadas son presentadas en la Figura 6. Debido a la imposibilidad de graficar las soluciones en 4 dimensiones, estas fueron divididas en rangos de diferencia de nivel representadas en ventanas y luego se representó el número de encendidos en cada ventana con símbolos diferentes. De esta forma, gráficos en 2 dimensiones de los costos de energía eléctrica en función de la potencia máxima alcanzada fueron elaborados para cada solución en la ventana correspondiente y con el símbolo adecuado a su N_e . Con esto, se consigue analizar gráficamente las soluciones obtenidas en el espacio objetivo de cuatro dimensiones.

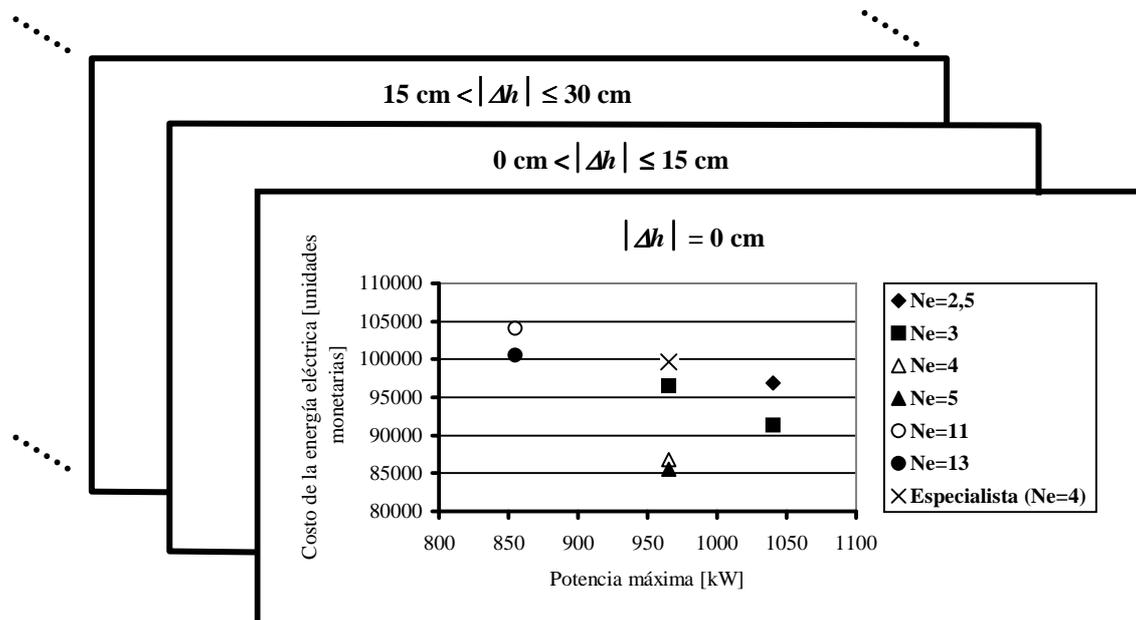


Figura 6: Gráfica de los mejores resultados experimentales. Note que la solución \times del especialista es dominada por la solución Δ que también requiere de 4 encendidos.

La Tabla 3 presenta los valores graficados en la Figura 6 para $|\Delta h| = 0$. Cabe destacar la particularidad de que diferentes programaciones (soluciones) pueden tener iguales valores de los objetivos.

Tabla 3: Resultados presentados en la Figura 6 para $|\Delta h| = 0$.

Nº de encendidos (Ne)	2.5	3	3	4	5	11	13
Costo de la energía eléctrica [unidades monetarias]	96850	96475	91325	86850	85550	104075	100525
Potencia máxima [kW]	1040	965	1040	965	965	855	855
Nº de soluciones por función objetivo	2	2	1	2	6	1	1

5. Conclusiones

El problema de la programación óptima de bombeo es presentado y resuelto en el presente trabajo utilizando dos algoritmos evolutivos multiobjetivos: el SPEA y el NSGA. Para evaluar estas propuestas, se comparan estos algoritmos (corriendo en dos plataformas diferentes) entre sí y con la solución manual calculada por un especialista, utilizando métricas específicas de la optimización multiobjetivo. Analizando estas métricas, resulta evidente que el SPEA propuesto supera ampliamente al NSGA y a la solución planteada por el especialista. De hecho, ninguna solución del NSGA pertenece al Frente Pareto óptimo. En conclusión, el SPEA se presenta como una poderosa herramienta a utilizar en problemas hidráulicos, como la programación óptima de bombeo, pudiendo incrementar sus funcionalidades en la medida de que se disponga de una computadora de alto desempeño.

Para trabajos futuros se espera: utilizar un número mayor de estaciones de bombeo y de reservorios; enfocar el tema de la potencia máxima alcanzada en mayor profundidad, considerando por ejemplo, los costos adicionales en caso de sobrepasarse la potencia reservada; analizar la programación de bombeo semanal o mensualmente; corregir la programación en caso de que la demanda real sea muy distinta a la demanda pronosticada y luego re-alimentar al sistema con una programación corregida.

Referencias

- [1] Ormsbee, L.E. y Lansley, K.E.; "Optimal Control of Water Supply Pumping Systems", *Journal de Planeamiento y Gerenciamiento de Recursos del Agua*, Vol. 120, No. 2, 1994.
- [2] Lansley, K. E. y Awumah, K.; "Optimal Pump Operations Considering Pump Switches", *Journal de Planeamiento y Gerenciamiento de Recursos del Agua*, Vol. 120, No. 1, 1994.
- [3] Mackle, Gunther; Savic, Dragan A. y Walters, Godfrey A.; "Application of Genetic Algorithms to Pump Scheduling for Water Supply", *GALESIA '95*. Publicación de Conferencia 4/4, pp. 400 - 405, Instituto de Ingenieros Eléctricos, Londres, 1995.
- [4] Savic, Dragan A.; Walters, Godfrey A. y Schawb, Martin; "Multiobjective Genetic Algorithms for Pump Scheduling in Water Supply", *ERASMUS*, Escuela de Ingeniería de Exeter, Exeter, Reino Unido, Universidad de Stuttgart, Stuttgart, Alemania, 1997.
- [5] Schaetzen, Werner de; "A Genetic Algorithm Approach for Pump Scheduling in Water Supply Systems", Grupo de Sistemas de Agua, Escuela de Ingeniería, Universidad de Exeter, Reino Unido, 1998.
- [6] Zistler E., Deb K., Thiele L., Coello C. y Corne D.; *Evolutionary Multi-criterion Optimization*, Lecture Notes in Computer Science 1993, First International Conference, EMO 2001, Zurich, Switzerland, ISBN 3-540-41745 Springer-Verlag, 2001.
- [7] Barán B., Vallejos J., Ramos R. y Fernández U.; Reactive Power Compensation using a Multi-objective Evolutionary Algorithm, paper aprobado para publicación en la IEEE Porto Power Tech' 2001.
- [8] Dolqachev, F. M. y Pashkov, N. N.; *Hidráulica y Máquinas Hidráulicas*, Mir, Moscú, Rusia, 1985.

- [9] Mataix, Claudio; *Mecánica de Fluidos y Máquinas Hidráulicas*, Harla, México, 2ª edición, 1982.
- [10] Streeter, Víctor L. y Wylie, Benjamín; *Mecánica de los Fluidos*, McGraw-Hill, México, 6ª edición, 1979.
- [11] Goldberg, David E.; *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison – Wesley, Reading, Massachusetts, 1989.
- [12] Zitzler, Eckart y Thiele, Lothar; “Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach”, *IEEE Transactions on Evolutionary Computation*, Vol. 3, N° 4, 1999.
- [13] Srinivas N. y Deb K.; “Multiobjective Optimization Using Nondominated Sorting Genetic Algorithms”, *Evolutionary Computation*, 2(3): 221-248, Fall 1994.