

Algoritmos Evolutivos para el Diseño de Redes Económicas y Confiables

Susana Duarte F. y Benjamín Barán C.
Centro Nacional de Computación,
Universidad Nacional de Asunción.
San Lorenzo, Paraguay. C.C. 1439.
Tel: +595 21 585550 ; Fax: +595 21 585619
bbaran@cnc.una.py, sduarte@cnc.una.py

Resumen

Este trabajo describe una técnica conocida como “algoritmos evolutivos multiobjetivo”, aplicada al diseño de redes de computadoras. Específicamente propone una versión del Strength Pareto Evolutionary Algorithm (SPEA), implementada sobre computadores personales. La implementación tiene por objetivo el diseño de redes en presencia de objetivos múltiples que pueden presentar conflictos entre ellos. Se pretende encontrar las mejores topologías para la red, a partir de información sobre la localización geográfica de los centros de comunicación, y los costos y confiabilidades de los distintos tipos de tecnologías de conexión disponibles. Esta aproximación es distinta porque elimina las restricciones presentes en implementaciones anteriores y sugiere un tratamiento del problema totalmente innovador y más adecuado a la naturaleza del mismo. Además, se sugiere un criterio de parada que permite reducir los tiempos de ejecución. La herramienta resultante provee al diseñador de todo un conjunto de soluciones válidas, mostrándose útil en el proceso de toma de decisiones. La aproximación sugerida queda validada mediante los resultados obtenidos a partir de su aplicación a un problema de diseño real. Concluimos que el método propuesto es efectivo y conduce a resultados correctos, y es computacionalmente eficiente y flexible permitiendo su aplicación a diversos problemas de diseño reales.

1. Introducción

El problema del diseño óptimo de redes cuando tienen que cumplirse objetivos múltiples es un problema típico del ámbito de la investigación de operaciones, y es relevante y extensible a varios problemas del mundo real, como ser: las telecomunicaciones [1, 14, 22], las redes de computadoras [1, 10, 25], sistemas de drenaje [24] y líneas de distribución de gases u otros líquidos [24]. Este trabajo se enfoca en el diseño óptimo de redes, considerando los objetivos de costo y confiabilidad, y como se establece en [5, 6, 8, 9], tal problema es NP-Hard.

Una forma simple de resolverlo consiste en enumerar todas las topologías de red correspondientes a la distribución geográfica de los nodos, calcular su costo y su confiabilidad y comparar estos valores con los de las demás redes, descartando todas aquellas que conduzcan a valores malos en los objetivos y manteniendo las demás. Este procedimiento da lugar a las siguientes observaciones:

- resulta obvio que existen topologías que no pueden considerarse mejores a las demás, al considerarse simultáneamente ambos objetivos. Esto sucede por existir más de un objetivo a optimizar y por el conflicto inherente a los mismos, i.e. a mayor confiabilidad, mayor costo. Justamente esta cualidad es uno de los principales atractivos del problema;
- el cálculo de confiabilidad de redes dada la confiabilidad de los enlaces que la componen es, a su vez, un problema NP-Hard [4, 5, 6, 8, 9] con un crecimiento exponencial a medida que aumenta el número de los arcos;

- el procedimiento de enumeración exhaustiva de todas las topologías y la evaluación de las mismas no es factible sino para redes de muy pocos nodos; ya que al aumentar el número de nodos la cantidad de topologías a evaluar crece de forma exponencial.

El problema ha sido previamente tratado de varias maneras: con técnicas de perturbación de grafos [23], técnicas enumerativas y técnicas heurísticas. Las técnicas de perturbación de grafos fueron abandonadas ya debido a que no presentan mejores soluciones que las proveídas por los expertos y no justifican el esfuerzo computacional. Las técnicas enumerativas son útiles solo en problemas muy pequeños (hasta apenas una decena de nodos). Y son las técnicas heurísticas las más promisorias en el ámbito. Las aproximaciones heurísticas utilizadas incluyen: búsqueda tabú [15, 11], algoritmos genéticos tradicionales [5, 6, 9, 17, 18, 24 Idem], templado simulado [2, 10, 22], entre otras [1]. Los trabajos de Kumar et al. aplicaban algoritmos genéticos tradicionales al problema de diseño de redes [17, 18] y al problema de la expansión de redes ya existentes [27, 28], considerando diversos objetivos y restricciones. De esta manera proponen el diseño de redes con confiabilidad máxima, dada una restricción de costo. Su aproximación tiene, sin embargo, las siguientes limitaciones:

- consideran siempre la optimización de un único objetivo, mientras los demás se incluían como restricciones;
- requieren que todos los diseños examinados sean factibles, lo que no es fácil de lograr si se considera el problema dual (minimizar costos manteniendo una restricción de confiabilidad);
- la codificación de las redes se realiza como una secuencia arbitraria de nodos, representando la presencia de un nodo con 1 y su ausencia con 0; de este modo la codificación se vuelve una cadena muy larga para redes relativamente pequeñas y, además, es difícil mantener la consonancia de los arcos presentes y ausentes luego de aplicar los operadores genéticos, para ello se deberá utilizar un operador de reparación complejo que además tiende a interferir con los efectos benéficos del cruzamiento;
- mientras que la aproximación se podría expandir para considerar formulaciones alternativas, los artículos solo se refieren a problemas con enlaces de idéntica confiabilidad y costo por unidad; situación que no es realista si se consideran problemas de diseño del mundo real.

De hecho, es una restricción muy simplista considerar que todos los enlaces posibles deben tener el mismo valor de confiabilidad y el mismo costo unitario. Esta es una limitación impuesta por el modelo matemático que se utiliza para el problema, no del problema de diseño mismo. En problemas reales, generalmente existen múltiples opciones para cada enlace, y cada opción tiene su confiabilidad y su costo asociado, así como otros atributos de diseño. Si deseamos redes económicas, es imperativo permitir diseños con enlaces de diferentes tipos y costos. Por ejemplo, en el área de telecomunicaciones, un cable coaxial puede tener protección y aislamiento, así como puede prescindir de él.

En los trabajos posteriores de Dengiz et Smith [6] y Laufer [19] se relaja esta restricción y se presenta una codificación más adecuada, pero el problema resultante sigue tratándose como de un solo objetivo. Cabe mencionar que el hecho de permitir que coexistan en la misma red enlaces de diferentes costos y confiabilidades mejora considerablemente la relevancia de los trabajos para el desarrollo de redes económicas reales, pero complica el cálculo de la confiabilidad de las redes, y aumenta el espacio de búsqueda de forma exponencial.

De todos modos, al considerar al problema como de un solo objetivo, estas aproximaciones también adolecen de limitaciones. Entre ellas la esencial es que permiten la optimización de muy pocos objetivos, y a medida que surgen más objetivos es necesario aplicar algoritmos de reparación y de penalización muy complejos y computacionalmente caros. Además, tienden a explorar una zona limitada del espacio de búsqueda y llegan a una única solución para el problema. Por ende brindan muy poca información que permita flexibilidad a la hora de tomar las decisiones. De esta manera no se explota la cualidad que tienen los algoritmos genéticos de mantener toda una población de soluciones y realizar una búsqueda en paralelo implícita, por lo que la misma cualidad se desperdicia.

Con este trabajo se pretende no solo vencer las limitaciones mencionadas, sino dar los primeros pasos hacia la demostración de que los algoritmos genéticos multiobjetivo tienen un gran potencial para resolver problemas de objetivos múltiples. Para ello se utilizará una implementación del SPEA (*Strength Pareto Evolutionary Algorithm*) [26], que es una versión de algoritmo genético especializada para lidiar con problemas de objetivos múltiples. La hipótesis queda validada con su aplicación a un problema de diseño real previamente utilizado en trabajos anteriores, por lo que puede servir como un excelente parámetro de comparación.

2. Descripción del problema

El problema de diseño de redes consiste en decidir qué enlaces se colocarán entre qué pares de centros de comunicación (nodos), dada la ubicación geográfica de éstos últimos y los diferentes posibles tipos de enlace, con sus características de diseño propios (confiabilidad, costos, velocidad de transmisión, etc.).

Para resolverlo, primero se describe la red en términos matemáticos, como un grafo en que cada centro de comunicación es un nodo del grafo, y las diferentes conexiones entre centros corresponden a los enlaces del grafo. De esta manera cada red es un grafo no dirigido $G = (V, E)$, donde:

- $V = \{1, 2, \dots, v\}$ es el conjunto de nodos o centros de comunicación
- E es el conjunto de enlaces. La cardinalidad de E es también el número de enlaces posibles y puede expresarse como:

$$|E| = L = \frac{|V|(|V| - 1)}{2} \quad (1)$$

Como el problema puede crecer tanto como el diseñador lo desee (pueden existir tantos nodos como se desee, se puede contar con varios tipos de tecnologías de comunicación para cada enlace, se puede escoger optimizar tantos objetivos como se desee, etc.) se deben colocarle límites para hacer factible una implementación real. En el presente trabajo se enuncia el problema como la optimización simultánea de dos objetivos: costo y confiabilidad. Las soluciones propuestas deben cumplir con un requerimiento de confiabilidad mínima. Se asume un enlace bidireccional entre cada par de nodos (no se permite redundancia). Estas restricciones son solo aparentes y no hacen al problema menos general, ya que la adición de nuevos objetivos es un problema trivial, aunque pueda requerir mayor esfuerzo computacional. Además, los enlaces redundantes se pueden tratar como otro tipo de enlace, con su propio costo y su propia confiabilidad [19].

Luego, matemáticamente el problema puede expresarse como:

$$\begin{aligned} \text{Optimizar } \mathbf{y} = \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x})) \\ \text{Sujeto a } e_1(\mathbf{x}) > 0 \end{aligned} \quad (2)$$

donde:

- $\mathbf{x} = (x_1, x_2, \dots, x_l) \in \mathbf{X}$, es el vector de decisión; cada $x_i \in \{0, 1, \dots, k\}$ representa un enlace entre un par de nodos y k es el número de tipos de tecnologías de comunicación disponibles (0 representa la ausencia de la conexión).
- $\mathbf{y} = (y_1, y_2) \in \mathbf{Y}$, es el vector en el espacio objetivo;
- $f_1(\mathbf{x})$ es la confiabilidad correspondiente a la configuración \mathbf{x} ;
- $f_2(\mathbf{x})$ es el costo de la misma configuración \mathbf{x} ;
- La restricción $e_1(\mathbf{x})$ se refiere a la confiabilidad mínima aceptable.
- \mathbf{X} denota el espacio de decisión y el espacio objetivo se conoce como \mathbf{Y} .

Para resolver el problema es necesario asumir ciertas condiciones, que se listan a continuación [5, 6, 8, 9]:

- Los nodos son completamente confiables (aunque se puede simular la caída de un nodo mediante la caída de sus enlaces incidentes).
- El costo y la confiabilidad de cada enlace potencial está dado por su tipo y se conoce de antemano.
- Los enlaces pueden estar en uno solo de dos estados posibles: operacionales o en falla (caídos).
- Los enlaces fallan de manera independiente unos de otros y no se produce la caída de unos como consecuencia de la falla en otros.
- No se consideran reparaciones.

A pesar de que parámetros como velocidad, capacidad y *throughput* son importantes para aplicaciones modernas, los principales objetivos a la hora del diseño de redes siguen siendo la maximización de la confiabilidad y la minimización del costo [4]. Ambas funciones se hallaron en todos los trabajos consultados que se refieren a problemas de optimización de diseño. Algunas veces el problema se declaraba como la minimización de costos sujeta a un requerimiento de confiabilidad [5, 6, 8, 9], mientras que otras veces se daba el problema dual [17, 18]. Sin embargo la naturaleza multiobjetivo del problema no se ha evidenciado previamente.

El concepto de confiabilidad se refiere a la probabilidad de un sistema de tener un comportamiento esperado durante un intervalo de tiempo dado. Luego, la confiabilidad de un sistema depende de su configuración y de la confiabilidad de sus componentes.

Existen varios métodos y métricas para medir la confiabilidad de un sistema. Para esta instancia del problema y para asegurar que exista siempre conexión entre todos los nodos que conforman la red, se utiliza la métrica de confiabilidad conocida como confiabilidad *all-terminal*. Esta métrica exige que la red forme siempre por lo menos un árbol sumidero (*spanning tree*) [4, 3, 17, 18].

El cálculo de confiabilidad se puede realizar de manera exacta solamente para redes de muy pocos nodos, debido al tiempo que consume. Para redes de mediano porte se pueden computar cotas que indiquen entre qué valores queda la confiabilidad de un sistema; pero tal sistema es igualmente impracticable para redes de gran tamaño, a la par que exigen una codificación especial de las redes. Por tales razones no se utilizan métodos de cálculo exacto, ni estimaciones con cotas, sino que se estima el valor de confiabilidad para cada red evaluada mediante simulaciones Monte Carlo [13, 14]. Este método ya ha sido usado con éxito en [6] incluso para redes de hasta 19 nodos.

El cálculo del costo, por su parte es bien sencillo, ya que se define como la sumatoria del costo de colocación de cada enlace. No se considera los costos de los centros de comunicación, pero tal información es fácilmente añadible.

3. Problema de prueba

El problema de test se basa en la expansión de la red de ULAK-NET, inicialmente publicada en [6]. Es una versión simplificada de un problema de diseño de una red real que pretende interconectar varias universidades y centros de investigación de Turkía.

Este problema se ha escogido por las siguientes razones:

- Es el problema de test más grande que se ha encontrado hasta el momento en el curso de las investigaciones;
- Se basa en un problema de diseño real;
- Los resultados obtenidos con aproximaciones previas y diferentes a la planteada en este trabajo están disponibles y pueden utilizarse como parámetros para las comparaciones.

Todos los detalles relativos a la disposición geográfica de los centros de comunicación, la confiabilidad de los diferentes tipos de enlace, así como sus costos puede hallarse en [6].

4. El algoritmo *Strength Pareto Evolutionary Algorithm (SPEA)*

El algoritmo evolutivo escogido para implementación es el *SPEA* propuesto por Zitzler et al en [26]. Este algoritmo, debido a que es muy nuevo, ha sido aplicado a muy pocos problemas de test, aunque representa uno de los algoritmos evolutivos multiobjetivo más promisorios ya que su planteamiento cumple con la hipótesis del teorema de convergencia de algoritmos evolutivos presentado en [27].

SPEA usa una mezcla de técnicas previamente usadas y otras recién establecidas para encontrar soluciones Pareto óptimas diferentes y en paralelo. Las características que comparte con otros algoritmos previamente propuestos son:

- guarda las soluciones no dominadas que ha encontrado hasta el momento en una población externa, de esta manera implementa elitismo;
- usa el concepto de dominancia para asignar un valor de *fitness* a los individuos;
- realiza *clustering* [20] para reducir el número de soluciones no dominadas que tiene almacenadas, sin destruir las características del frente Pareto delineado hasta el momento.

Por otro lado, *SPEA* presenta, de manera exclusiva, los siguientes conceptos:

- combina las tres características arriba mencionadas en un único algoritmos (hasta ahora ellas se habían propuesto siempre aisladas unas de otras);
- el *fitness* de un individuo está determinado a partir de las soluciones que se encuentran en la población de individuos no dominados solamente; de este modo, no es relevante que los miembros de la población general se dominen unos a otros;
- todas las soluciones que forman parte del conjunto externo de soluciones no dominadas participan en la selección;
- se sugiere un nuevo método para inducir a la formación de nichos [7] dentro de la población y así preservar la diversidad genética; este método está basado en el concepto de Pareto y no requiere el conocimiento o determinación previa de algún parámetro de distancia como sí precisan otros algoritmos evolutivos que utilizan el concepto de compartir *fitness* entre los individuos más cercanos.

En la siguiente sub-sección se halla un diagrama de flujo del *SPEA*, indicando los pasos fundamentales del algoritmo. Esencialmente, la diferencia entre el *SPEA* y el algoritmo genético tradicional radica en la asignación de *fitness* y el mantenimiento de la población externa. Los operadores evolutivos de selección, cruzamiento y mutación son los usuales.

4.1 Diagrama de Flujo del *SPEA*

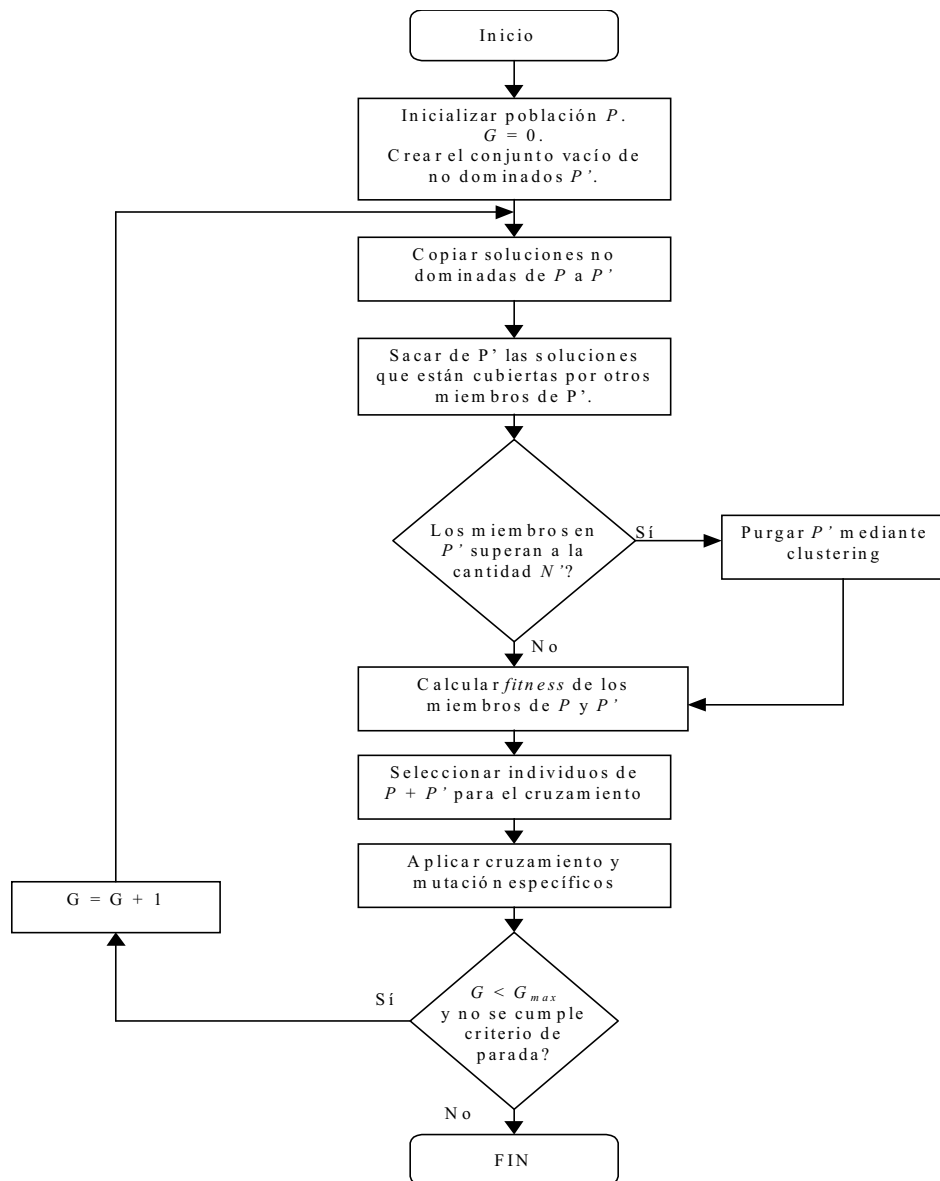


Figura 1. Diagrama de Flujo del *SPEA*.

El diagrama de flujo del algoritmo mostrado en la figura 1 utiliza N' para representar un tamaño máximo de la población externa, mientras que N simboliza el tamaño de la población genética usual. Estos valores se definen previamente y sirven además para la decisión de la aplicación del proceso de *clustering* [20]. La selección se realiza en toda la unión de los dos conjuntos P y P' (conjunto de individuos de la población genética y externa, respectivamente), hasta que se obtiene la cantidad de padres deseada. Para la selección se utilizan torneos binarios con reemplazo. El proceso de *clustering* fue tomado de [20], según sugieren los autores del algoritmo.

4.2 Procedimiento para asignar *fitness* a los individuos

El procedimiento de asignación de *fitness* se realiza en dos fases. Primero, se asigna un rango a los individuos del conjunto P' (conjunto externo de no dominados). Luego, se evalúan a los miembros de P (conjunto de soluciones generales).

A cada solución $i \in P'$ se asigna un valor real $s_i \in [0,1)$, llamado la fuerza o *strength*; s_i es proporcional al número de individuos $j \in P$ para los cuales i domina a j que se denota como $i \succ j$ (es decir, el *strength* es una medida de la cantidad de individuos dominados por i). Sea n el número de individuos en P que están dominados por i , y asumamos que N es el tamaño de P . Luego s_i se define como $s_i = \frac{n}{N+1}$. El *fitness* de i es igual a su *strength*: $F(i) = s_i$.

El *fitness* de un individuo $j \in P$ se calcula sumando los *strengths* de todas las soluciones externas no dominadas $i \in P'$ que dominan a j . Sumamos 1 a ese total para asegurar que los miembros de P' tengan mejor *fitness* que los miembros de P . Así el *fitness* del individuo j se expresa como $f_j = 1 + \sum_{i,i \succ j} s_i$; donde $f_j \in [1, N)$.

La idea detrás de este mecanismo es preferir siempre los individuos que están más cerca del frente Pareto óptimo y al mismo tiempo, distribuirlos en toda la superficie factible. La principal diferencia con otras técnicas usadas para mantener la diversidad genética es que los nichos se definen en términos de la dominancia Pareto, y no en términos de distancia euclidiana entre las representaciones de los individuos.

5. Descripción de la implementación

Para la aplicación del algoritmo evolutivo multiobjetivo escogido, primeramente debió codificarse cada solución posible de la forma $x = (x_1, x_2, \dots, x_L)$, donde cada x_i representa a un enlace y a su tipo, i.e. $x_i \in \{0, 1, \dots, k\}$. De esta manera se obtuvo una cadena de números enteros entre 0 y k , y de longitud L (recordamos que L es la cantidad total de enlaces posibles de la red, y que el valor 0 para un enlace significa la ausencia del mismo). Para determinar qué posición de la cadena así conformada corresponde a qué enlace de la red física se dibuja el grafo correspondiente a la red, se numeran unívocamente cada uno de los nodos y se escribe la matriz de adyacencia del grafo resultante [12]. Como la matriz así obtenida es siempre simétrica, solo se utiliza el triángulo superior a la diagonal de la misma. Por ejemplo, para codificar la red mostrada en la figura 2, se utilizará la matriz de adyacencia mostrada en la tabla 1.

Como ya se expresó, la tabla 1 muestra a una matriz simétrica, y tomando de ella el triángulo superior a la diagonal (sin considerar a esta), leyendo de arriba hacia abajo y de izquierda a derecha, se obtiene la representación final de la red, que denominamos x . En este caso $x = \{1300012022\}$. Nótese que la longitud de todas las cadenas para representar a las diferentes redes posibles tienen siempre la misma longitud L , ya que si un enlace está ausente se lo representa con el valor 0. Los valores diferentes a cero, como ya se expresó, representan diferentes tecnologías de comunicación: fibras ópticas, enlaces de radio, cables coaxiales, etc.

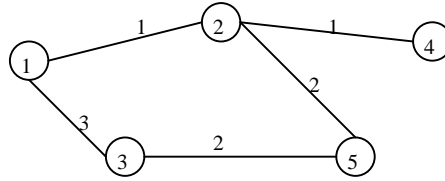


Figura 2. Red a codificar.

	1	2	3	4	5
1	0	1	3	0	0
2	1	0	0	1	2
3	3	0	0	0	2
4	0	1	0	0	2
5	0	2	2	0	0

Tabla 1. Matriz de adyacencia correspondiente a la red a codificar

Superada la fase de codificación conviene concentrarse en el cálculo de los objetivos. El cálculo de confiabilidad se realiza, según se expresó previamente, mediante simulaciones Monte Carlo [8, 13]. Solo se realizaron 10000 replicaciones debido al alto costo computacional de cada simulación. Es necesario comprender que cada simulación implica obtener números al azar para cada uno de los enlaces, dependiendo de éstos números se puede establecer si los enlaces están en falla u operando; luego, con la red así obtenida se verifica la presencia de conexión entre cada par de nodos. De esta manera resulta evidente el costo computacional elevado del cómputo de confiabilidad.

Las soluciones que no superaron el umbral de 0 en el cálculo de confiabilidad, i.e. las redes que no forman al menos un *spanning tree*, no permanecen en la población externa, a pesar de formar parte del conjunto de soluciones porque representan los costos más bajos.

El proceso de hallar las soluciones no dominadas en el conjunto P se basa en el concepto de dominancia [7]. Según este concepto, un individuo i domina a otro j si y solo si se cumple que i es igual o mejor que j en todos los objetivos, y –cumplida la condición anterior– existe al menos un objetivo en el que i es mejor que j . Cada vez que un nuevo individuo no dominado con respecto a P se encuentra, se lo compara con los miembros de P' . Si el individuo representa una nueva solución, se lo inserta en P' ; si llegara a superarse el límite de elementos en P' se aplica el *clustering*.

El cómputo del fitness sigue el proceso previamente descrito en secciones anteriores, mientras que la selección se realiza mediante torneos binarios con reemplazo. La nueva generación se crea mediante cruzamiento de 1 punto.

El operador de mutación toma $m\%$ individuos de la población y cambia un allele de su cromosoma con probabilidad 0.3.

Otros parámetros importantes del algoritmo se listan a continuación:

- Tamaño de la población P (N): 100 individuos.
- Tamaño del conjunto externo de no dominados (N'): 100 individuos.

- Número máximo de generaciones (g_{max}): 5000.
- Probabilidad de cruzamiento (p_c): 1.
- Razón de mutación (r_m): 0.3.
- Porcentaje de la población mutado en cada generación ($m\%$): 5%.

La población inicial del algoritmo se ha generada de manera aleatoria. Para suspender las iteraciones del algoritmo se implementó un criterio de parada (también una contribución original del presente trabajo), que indica que se seguirá iterando siempre y cuando la población externa de dominantes sufra alteraciones en el curso de 100 generaciones.

El siguiente pseudocódigo muestra la versión secuencial implementada.

```

Procedimiento SPEASec()
INICIO
  Leer parámetros de entrada para el SPEA:  $s$ ,  $g_{max}$ ,  $p_c$ ,  $r_m$ ,  $m\%$ 
  Leer parámetros para el problema: costo y confiabilidad de enlaces
  Generar aleatoriamente la población inicial  $P$ 
  Contador_Generaciones = 1.
  MIENTRAS Contador_Generaciones <  $g_{max}$  y no se cumple el criterio de parada
    Computar los valores de los objetivos para cada individuo
    Hallar los individuos no dominados en la población  $P$ 
    Actualizar población externa de no dominados  $P'$ 
    SI las soluciones en  $P'$  superan  $N'$ 
      Aplicar clustering a  $P'$ 
      Calcular el fitness de cada individuo en  $P$  y  $P'$ 
      MIENTRAS no se obtiene la cantidad de padres necesaria
        Usar torneos binarios para seleccionar individuos de  $P + P'$  como padres
        Aplicar operadores de cruzamiento y mutación para generar el nuevo  $P$ 
      FIN MIENTRAS
    Write out the individuals from  $P'$ .
  FIN

```

Pseudocódigo 1. Procedimiento SPEA secuencial para diseño de redes.

6. Resultados experimentales

Los resultados aquí presentados se obtuvieron de 10 corridas sucesivas del algoritmo sobre un computador con procesador AMD K6-2 350 MHz, con 128 MB de memoria RAM. El programa fue enteramente codificado en lenguaje C++, y compilado mediante el compilador de distribución gratuita de GNU. El sistema operativo utilizado ha sido LINUX en la distribución Mandrake 7.0.

La tabla 2 muestra la cantidad de soluciones halladas en cada corrida realizada, así como el tiempo que cada corrida tomó para terminar su ejecución. También se muestra la cantidad total de soluciones no dominadas halladas. Según se explicó anteriormente, el número total de soluciones halladas no corresponde a la suma de la cantidad de soluciones halladas en cada corrida, sino que se obtiene de la aplicación del concepto de dominancia sobre la unión de todos los sub-conjuntos hallados.

Inmediatamente resulta evidente que esta vez, por cada corrida, no se tiene una única solución, sino que se cuenta con todo un conjunto de soluciones factibles, de las cuales se puede escoger la que más se ajuste a los requerimientos de costo y confiabilidad. Así, por ejemplo se ha hallado una red de costo 1188483 y confiabilidad de 0.704, si se desea mayor confiabilidad, digamos de 0.804, se puede obtener una red con esa confiabilidad si se está dispuesto a pagar un poco más (1282711), si se desea confiabilidad aún mayor, 0.999 a 1 deberá pagarse poco más de 2 millones de dólares. Así, se ha dotado al diseñador y al responsable de la toma de decisiones todo un espectro de soluciones entre las cuales puede hacer su elección, con la promesa de que estas soluciones son no dominadas, es decir: difícilmente existe una red de 0.704 de confiabilidad y menor costo al hallado.

Resultados del SPEA para Diseño de Redes		
Corrida	Soluciones dominantes halladas	Tiempo de ejecución
1	41	8.640 horas
2	42	8.712 horas
3	46	8.880 horas
4	46	8.450 horas
5	51	9.003 horas
6	44	8.970 horas
7	44	8.613 horas
8	51	8.257 horas
9	51	8.437 horas
10	57	8.903 horas
Cantidad total de soluciones halladas en todas las corridas		39

Tabla 2. Resultados de 10 corridas del algoritmo propuesto

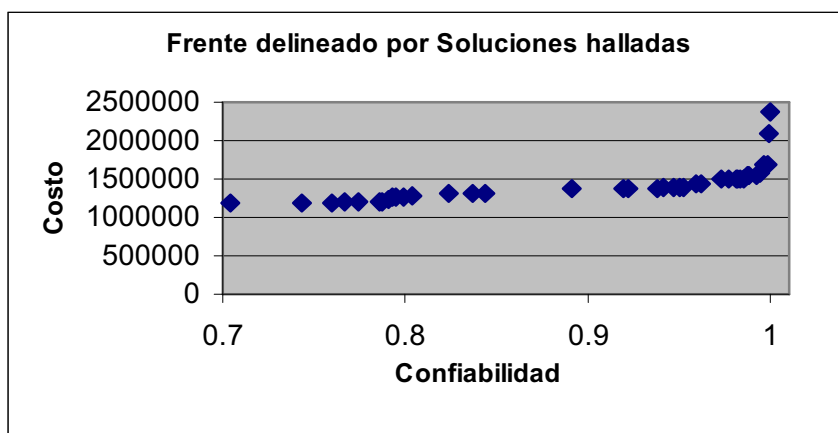


Figura 3. Gráfico que muestra todas las soluciones halladas en las diferentes corridas.

Cabe también observar que el algoritmo ha obtenido mejores resultados que los obtenidos por otras aproximaciones, no solo en cuanto a soluciones halladas, sino también en lo que respecta a calidad de las soluciones. En [Diseño Económico de redes confiables] se presenta como mejor red hallada para el mismo problema, una red de costo muy superior (7000000 \$).

7. Conclusiones y trabajos futuros

Las contribuciones de este trabajo se pueden resumir en lo siguiente:

- La propuesta de utilización de un algoritmo evolutivo multiobjetivo para el problema de diseño de redes. Como se ha dicho, este problema ha sido antes estudiado pero nunca antes se ha evidenciado su naturaleza multiobjetivo. Esta aproximación es útil ya que conduce a todo un conjunto de soluciones óptimas en vez de a una única solución, como lo hacen los trabajos previos.
- Criterio de parada para el algoritmo. Hasta ahora, en todas las implementaciones del mismo no se había utilizado ningún criterio de parada diferente a alcanzar un número máximo de generaciones. Sin embargo, con el criterio de parada escogido se disminuye significativamente el tiempo de ejecución, ya que llegado a un punto en que el algoritmo ya no realiza exploraciones significativas del espacio de búsqueda, ya no se justifica seguir iterando.

Como sugerencia de trabajos futuros se puede citar lo siguiente:

- Realizar la implementación de otros algoritmos evolutivos multiobjetivo, para el mismo problema de modo a concluir que éstos algoritmos son verdaderamente efectivos para la solución del problema planteado.
- Aumentar el número de funciones objetivo a optimizar y verificar el comportamiento consistente del algoritmo.
- Realizar versiones paralelas de la implementación, de modo a mejorar la escalabilidad.

Referencias

- [1] Aggarwal K., Chopra Y. and Bajwa S., "Topological layout of links for optimising the overall reliability in a computer communication system," *Microelectronics and Reliability*, Vol. 22, No. 3 (1982) 347-351.
- [2] Atiqullah M. M. and Rao S. S., "Reliability optimization of communication networks using simulated annealing," *Microelectronics and Reliability*, Vol. 33, No. 9 (1993) 1303-1319.
- [3] Boorstyn R. and Frank H. Large-Scale network topological optimization. *IEEE Trans. on communication*, vol. COM-25, n° 1, pp. 29-47. 1977.
- [4] Colbourn C. J. "Reliability issues in telecommunications network planning". Department of Computer Science, University of Vermont. Burlington. USA.
- [5] Deeter D. L., Smith A. E., "Heuristic optimization of network design considering all-terminal reliability," *Proceedings of the Reliability and Maintainability Symposium* (1997) 194-199.
- [6] Deeter D. L., and Smith A. B. Economic design of reliable networks, *IIE Transactions*, vol. 30, in print. 1999.
- [7] Deb K. Non-linear goal programming using multi-objective genetic algorithms. Technical report TR CI-60/98, University of Dortmund, Germany: Department of Computer Science/XL 1999.
- [8] Dengiz B., Smith A. B. and Altiparmak F. "Local search genetic algorithm for optimal design of reliable networks." *IEEE Transactions on Evolutionary Computation*, Vol. 1, N° 3, Septiembre 1997.
- [9] Dengiz B., Altiparmak F. and Smith A. E., "Efficient optimization of all-terminal reliable networks using an evolutionary approach," *IEEE Transactions on Reliability*, Vol. 46 (1997) 18-26.
- [10] Fetterolf P. C. and Anandalingam G., "Optimal design of LAN-WAN internetworks: an approach using simulated annealing," *Annals of Operations Research*, Vol. 36 (1992) 275-298.
- [11] Glover F., Lee M. and Ryan J., "Least-cost network topology design for a new service: an application of a tabu search," *Annals of Operations Research*, Vol. 33 (1991) 351-362.
- [12] Horowitz E. and Sahni S. *Fundamentals of Data Structures*. Reprint edition. W H Freeman & Co. June 1983.
- [13] Jan R. H. Design of reliable networks. *Computers and operations research*, vol. 20, n° 1, pp. 25-34. 1993.
- [14] Jan R. H., Hwang F. and Chen S., "Topological optimization of a communication network subject to a reliability constraint," *IEEE Transactions on Reliability*, Vol. 42, No. 1 (1993) 63-70.
- [15] Koh S. J. and Lee C. Y., "A tabu search for the survivable fiber optic communication network design," *Computers and Industrial Engineering*, Vol. 28, No. 4 (1995) 689-700.

- [16] Konak A. and Smith A. A hybrid genetic algorithm approach for backbone design of communication networks. Proceedings of the 1999 Congress on Evolutionary Computation, Washington D. C., IEEE, 1999.
- [17] Kumar A., Pthak R. M., Gupta Y. P. and Parsaei H. R., "A genetic algorithm for distributed system topology design," *Computers and Industrial Engineering*, Vol. 28, No. 3 (1995) 659-670.
- [18] Kumar A., Pthak R. M. and Gupta Y. P., "Genetic-algorithm-based reliability optimization for computer network expansion," *IEEE Transactions on Reliability*, Vol. 44, No. 1 (1995) 63-72.
- [19] Laufer F. Optimización topológica de redes confiables empleando A-Teams. Proyecto final. Facultad de Ciencias y Tecnología, Universidad Católica Nuestra Señora de la Asunción. Asunción-Paraguay. Febrero del 2000.
- [20] Morse J. N. Reducing the size of the nondominated set: pruning by clustering. *Comput. Oper. Res.*, vol 7, n° 1 y 2. 1980
- [21] Pierre S. And Elgibaoui A. A tabu search approach for designing computer network topologies with unreliable components. *IEEE Trans. on reliability*, vol. 46, n° 3, pp. 350-359. 1997.
- [22] Pierre S., Hyppolite M. A., Bourjolly J. M. and Dioume O. Topological design of computer communication networks using simulated annealing. *Engineering application of artificial intelligence*, vol. 8, n° 1, pp. 61-69. 1995.
- [23] Tanenbaum A. S. "Computer Networks". Prentice-Hall, Englewood Cliffs, New Jersey 1981.
- [24] Walters G. A. and Smith D. K., "Evolutionary design algorithm for optimal layout of tree networks," *Engineering Optimization*, Vol. 24 (1995) 261-281.
- [25] Wilkov R. S., "Design of computer networks based on a new reliability measure," *Symposium on Computer-Communications Networks and Teletraffic*, Polytechnic Institute of Brooklyn (1972) 371-384.
- [26] Zitzler E. and Thiele L. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE transactions on evolutionary computation*, vol 3, n° 4. November, 1999.
- [27] Van Veldhuisen D. A. *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. Ohio, EE. UU. May, 1999.