

Operadores de Cruce con Múltiples Descendientes para Algoritmos Genéticos con Codificación Real: Estudio Experimental

Francisco Herrera¹, Manuel Lozano¹, Ana M^a Sánchez²

Resumen-- En este trabajo proponemos diferentes operadores de cruce que generan múltiples descendientes a partir de dos padres, seleccionando los dos mejores para sustituir a los padres en la nueva población. Analizando la experimentación realizada, se observa que estos operadores mejoran el comportamiento de los operadores clásicos lográndose un adecuado equilibrio entre diversidad y presión selectiva.

Palabras clave-- Algoritmos Genéticos, Codificación Real, Operador de Cruce.

I. INTRODUCCIÓN

El operador de cruce es un método para compartir información entre cromosomas. Generalmente, se combinan las características de cromosomas padres para generar cromosomas hijos. El operador de cruce juega un papel central en los Algoritmos Genéticos (AGs) [7]. De hecho puede considerarse como una de las características que definen a estos algoritmos y es uno de los componentes a tener en cuenta para mejorar su eficacia. En el caso de los Algoritmos Genéticos con Codificación Real (AGCRs), este operador influye decisivamente sobre el nivel de diversidad en la población, y por ello, es un factor determinante para evitar el problema de la convergencia prematura [3]. Esto explica que el principal esfuerzo en la investigación desarrollada para mejorar los AGCRs se centre en la propuesta y estudio de nuevos operadores de cruce [5].

Normalmente, el operador de cruce se aplica sobre parejas de padres, generando dos hijos para cada una de ellas, los cuales se introducen en la población [14]. Sin embargo, existen operadores de cruce con *múltiples padres* [8, 19, 21, 23], que combinan las características de más de dos padres para generar hijos. Su objetivo es introducir diversidad en la población, ya que mezclan información procedente de diversos padres. También se han presentado

operadores de cruce con *múltiples descendientes* [12, 13, 16], que generan más de dos descendientes por cada grupo de padres. En este caso, un mecanismo de selección de descendientes limita el número de hijos que entran a formar parte de la nueva población. Estos operadores, que están inspirados en la naturaleza, tratan de sacar más beneficio de los padres mediante el muestreo de un mayor número de posibles soluciones resultantes de su recombinación. Todos los descendientes se pueden producir utilizando el mismo mecanismo de generación de descendientes (operadores de cruce *homogéneos*) o mediante distintos mecanismos (operadores de cruce *híbridos*).

La efectividad final del operador de cruce va a depender del equilibrio entre la diversidad asociada a los mecanismos de generación de descendientes y la presión selectiva derivada del mecanismo de selección de descendientes. En este equilibrio intervienen como factores primordiales el número de padres, el número de descendientes y la naturaleza de los mecanismos de generación y selección de descendientes.

Profundizando en estas ideas, en este trabajo, abordamos la mejora del comportamiento de los AGCRs mediante el diseño de operadores de cruce que logren el acertado equilibrio entre la diversidad y la presión selectiva para obtener buenos resultados sobre un amplio abanico de posibles problemas de búsqueda. Para ello, nos centramos en la propuesta y estudio de *operadores de cruce con múltiples descendientes homogéneos*. Limitamos nuestro estudio a operadores de cruce con múltiples descendientes aplicados sobre dos padres y que operan con un mecanismo de selección de descendientes consistente en escoger a los dos hijos con mejor adaptación. En concreto, se trata de determinar el peso de los múltiples descendientes en la evolución, así como el aumento de la presión selectiva que lleva asociado este mecanismo de selección de descendientes, y cómo influyen en la diversidad de la población.

¹Dpto de Ciencias de la Computación e Inteligencia Artificial, Escuela Técnica Superior de Ingeniería Informática, Universidad de Granada, herrera@decsai.ugr.es, lozano@decsai.ugr.es

²Dpto. de Informática, Escuela Superior de Ingeniería Informática, Universidad de Vigo, amlopez@uvigo.es

Esta propuesta va a ser analizada sobre 13 funciones/problemas que son utilizados usualmente para contrastar el rendimiento de los AGCRs.

El trabajo queda estructurado de la siguiente manera: En la Sección II describimos los principales aspectos de la codificación real y se realiza una taxonomía de los principales operadores de cruce para AGCRs. En la Sección III se presenta la propuesta de operadores de cruce con múltiples descendientes. En la Sección IV se muestra el estudio experimental, y el análisis de resultados. Por último, en la Sección V se presentan las conclusiones finales.

II. PRELIMINARES

A. Algoritmos Genéticos con Codificación Real

El uso de la codificación real parece particularmente natural para resolver problemas con variables en espacios de búsqueda continuos [17]. Un cromosoma es un vector de números reales cuyo tamaño es igual a la longitud del vector solución del problema; de esta forma, *cada gen representa una variable del problema*. Los valores de los genes están forzados a permanecer en el intervalo establecido para las variables a las cuales representan, de modo que los operadores genéticos tendrán que preservar este requerimiento.

Debido a la particularidad de su codificación, hay que destacar que con el operador de cruce, a nivel de gen, pueden considerarse distintos tipos de intervalos donde el operador puede generar genes [16,18]:

- Un intervalo de *explotación* cuyos extremos son los genes a combinar. Se denomina así porque contiene los valores situados entre los genes padres y presenta las características de explotación asociadas al operador.

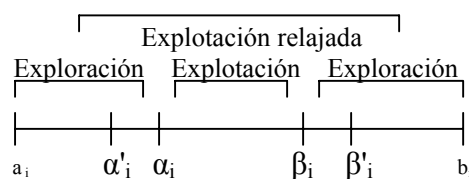
- Dos intervalos de *exploración*, que se definen a partir de cada extremo del intervalo de definición y el gen padre más cercano al mismo. Los valores en estos intervalos siempre están alejados de uno de los genes padres y se consideran zonas de exploración de nuevos valores.

- Intervalos de *explotación relajada*. Son extensiones del intervalo de *explotación* hacia los intervalos de *exploración*.

En la Figura 1 se muestran gráficamente estos intervalos. Los dos genes a combinar son $c_i^1, c_i^2 \in [a_i, b_i]$. Notaremos $\alpha_i = \min\{c_i^1, c_i^2\}$ y $\beta_i = \max\{c_i^1, c_i^2\}$. El intervalo de acción de estos genes, $[a_i, b_i]$, puede dividirse en tres subintervalos: $[a_i, \alpha_i]$, $[\alpha_i, \beta_i]$ y $[\beta_i, b_i]$. También se va a considerar una

región de explotación relajada $[\alpha'_i, \beta'_i]$ con $\alpha'_i \leq \alpha_i$ y $\beta'_i \geq \beta_i$.

Figura 1. Intervalos de pertenencia de un gen resultado de combinar c_i^1 y c_i^2



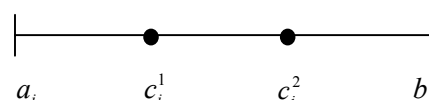
B. Taxonomía de operadores de cruce

Los principales operadores de cruce para AGCRs se pueden clasificar atendiendo a los efectos que provocan sobre los cromosomas, teniendo en cuenta aspectos tales como si se conservan los genes de los padres en los descendientes, si dichos genes son modificados mediante alguna función de agregación, o bien, si los descendientes se generan en base al entorno asociado a los padres.

Se pueden distinguir tres grupos de operadores:

- Operadores de cruce discretos (OCD): Cada gen de un descendiente coincide con un gen de uno de los padres, es decir, los genes de los padres se conservan en los hijos. Ejemplos característicos son: Simple [20], Uniforme [9] y Cruce en dos puntos [9]. En la Figura 2 se observa el efecto de estos operadores en cuanto a los intervalos de generación de los genes.

Figura 2. Efecto de los operadores de cruce discretos



- Operadores de cruce basados en agregación (OCA): Los genes de los descendientes se generan mediante una función de agregación que combina los genes de los padres. Operadores representativos de este grupo son: Aritmético [20], geométrico [21], lineal BGA [22], heurístico [25] y dinámico heurístico [15]. Estos operadores generan genes tanto en el intervalo de explotación como en el de exploración. En la Figura 3 se observa el efecto de estos operadores.

- Operadores de cruce basados en entornos (OCE): Estos operadores generan los genes de los descendientes a partir de valores tomados del entorno asociado a los padres, normalmente a partir de un intervalo cuyos extremos dependen de los valores de los genes de los cromosomas padre. Este tipo de operadores

tienen carácter aleatorio. Los operadores propios de este grupo, BLX- α [10], BLX- α - β [11], SBX- η [4] y FR-d [24]. En la Figura 4 se observa el efecto de estos operadores.

Figura 3. Efecto de los operadores de cruce basados en agregación

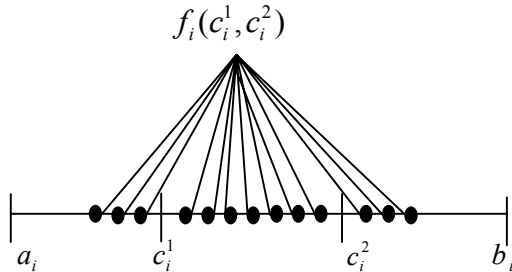
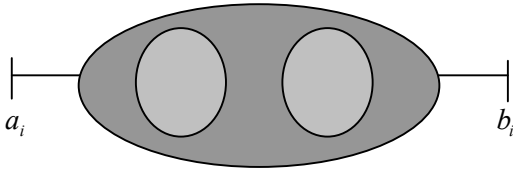


Figura 4. Efecto de los operadores de cruce basados en entornos



III. PROPUESTA DE MÚLTIPLES DESCENDIENTES

En esta sección se propone el incremento del número de descendientes generados por algunos operadores de cruce, dando lugar a lo que hemos definido como operadores de cruce con múltiples descendientes, teniendo en cuenta que seleccionamos los dos mejores para incluirlos en la nueva población.

La exploración del espacio de búsqueda se incrementa con la generación de múltiples descendientes, con lo que se hace posible una mayor diversidad en la población. Sin embargo, la selección de los dos mejores descendientes produce un incremento de la presión selectiva, y puede provocar una rápida convergencia que perjudique a la diversidad.

Dado que los grupos de la taxonomía presentados anteriormente tienen diferentes niveles de exploración y explotación, llevamos a cabo nuestra propuesta con una serie de operadores que representan los distintos grupos de la taxonomía y por lo tanto presentan distintos niveles de exploración y explotación. La Tabla I muestra los operadores que se han utilizado en este trabajo, y cuya descripción puede encontrarse en el Apéndice

A, indicando los grupos de la taxonomía a los que pertenecen. Se han utilizado parámetros que permitan una alta diversidad.

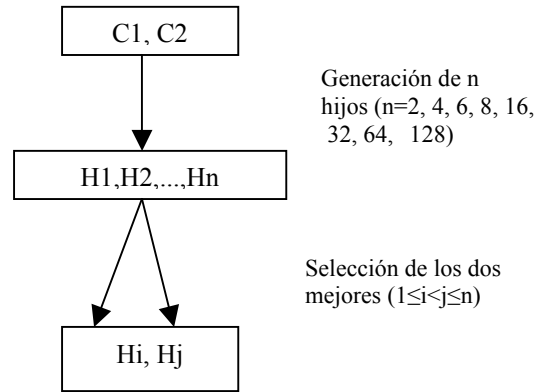
TABLA I. OPERADORES DE CRUCE UTILIZADOS

| Operador de Cruce | Grupo de la Taxonomía |
|-------------------|-----------------------|
| Aritmético | OCA |
| Dos Puntos | OCD |
| BLX- 0.5 | OCE |
| SBX- 1 | |
| FR-0.5 | |

Estos operadores se van a utilizar en la generación de 4, 6, 8, 16, 32, 64 y 128 descendientes, de los cuales se seleccionarán los dos mejores (cruce 2-4-2, 2-6-2, 2-8-2, 2-16-2, 2-32-2, 2-64-2 y 2-128-2 respectivamente) para formar parte de la nueva población.

En la Figura 5 se representa el esquema de funcionamiento de estos operadores en la generación y selección de múltiples descendientes.

Figura 5. Esquema de funcionamiento de los operadores



IV. ESTUDIO EXPERIMENTAL

En esta sección se estudia el comportamiento que presentan los operadores mencionados en la Tabla I en base al número de descendientes que generan.

Para ello en la primera subsección se muestran los experimentos realizados y en la segunda se realiza un análisis de los resultados obtenidos.

A. Experimentos

En todos los experimentos se ha utilizado mutación no uniforme [17], con lo que la probabilidad de que un gen se mute disminuye según avanza la ejecución

del algoritmo. De esta manera los cambios producidos en los genes son menores en las últimas generaciones, produciendo un ajuste local. El procedimiento de selección de la nueva población es el de ordenación lineal [1] asegurando la presencia del mejor individuo (elitismo) [6], y se ha utilizado el método de muestreo universal estocástico [2].

Hemos considerado trece funciones de test frecuentemente utilizadas: Modelo Esférico, Función de Schwefel, Función de Rastrigin Generalizada, Función de Griewangk, Expansión de F10, Función de Rosenbrock Generalizada, Sistema de Ecuaciones Lineales, Problema de modulación de la frecuencia de sonido, Problema de aproximación polinomial, Función de Ackley, Función de Bohachevsky, Función de Watson, y Función de Colville. La formulación de estos problemas se puede encontrar en [18]. Estas funciones varían con respecto a algunas características como continuidad, modalidad o dimensiones, consiguiéndose de esta forma un amplio intervalo de posibles situaciones para la experimentación. La dimensión del espacio de búsqueda es 25.

Los parámetros utilizados para llevar a cabo los experimentos figuran en la Tabla II. Los algoritmos se han ejecutado 30 veces, con valor de semilla distinto para el generador de números aleatorios.

TABLA II. PARÁMETROS DEL ALGORITMO GENÉTICO

| | |
|--------------------------|--------|
| Tamaño de la población | 61 |
| Probabilidad de cruce | 0.6 |
| Probabilidad de mutación | 0.1 |
| Número de evaluaciones | 100000 |

B. Análisis de resultados

En las Tablas III-VII se presenta un resumen de los resultados obtenidos con cada uno de los operadores. Se muestran los porcentajes en que cada operador de cruce, en base al número de descendientes generados, ha obtenido los mejores valores medios (se ha aplicado un t-test con nivel 0.05 de significancia) para cada una de las funciones de evaluación (trece) consideradas.

Las columnas de las Tablas III-VII hacen referencia a lo siguiente:

- **MM / MT**: Mejor media / Mejor test. Porcentaje de funciones de evaluación en las que cada operador de cruce ha obtenido el mejor valor medio en comparación con el resto de operadores, y continúa siendo el mejor tras aplicar el t-test.

- **MM / ST**: Mejor media / Similar test. Porcentaje de funciones de evaluación en las que cada operador

de cruce ha obtenido el mejor valor medio en comparación con el resto de operadores y, tras la aplicación del t-test, hay al menos otro operador de cruce que no presenta diferencias significativas con él.

- **TM**: Total Mejor. Porcentaje de funciones de evaluación en las que cada operador de cruce presenta los mejores resultados sin tener en cuenta la información obtenida mediante la aplicación del t-test. Es la suma de los porcentajes de las dos columnas anteriores.

- **ST / NMM**: Similar test / No mejor media. Porcentaje de funciones de evaluación en las que cada operador de cruce, aunque no obtiene el mejor valor medio, demuestra tener un comportamiento similar al mejor operador de cruce tras la aplicación del t-test.

- **T M/S**: Total Mejor/Similar. Porcentaje total de funciones de evaluación en las que el operador de cruce demuestra tener el mejor comportamiento o similar al mejor. Es la suma de las dos columnas anteriores.

De igual manera, estos resultados se muestran gráficamente en las Figuras 6-10.

TABLA III. ANALISIS DE LOS RESULTADOS OBTENIDOS POR CADA OPERADOR BLX-0.5

| Op. BLX-0.5 | MM /MT | MM /ST | TM | ST / NMM | T M/S |
|-------------|--------|--------|---------------|----------|---------------|
| (2,2,2) | 0% | 0% | 0% | 23.07% | 23.07% |
| (2,4,2) | 0% | 15.38% | 15.38% | 30.76% | 46.14% |
| (2,6,2) | 0% | 7.69% | 7.69% | 46.14% | 53.83% |
| (2,8,2) | 23.07% | 23.07% | 46.14% | 30.76% | 76.90% |
| (2,16,2) | 7.69% | 23.07% | 30.76% | 30.76% | 61.52% |
| (2,32,2) | 7.69% | 15.38% | 23.07% | 23.07% | 46.14% |
| (2,64,2) | 0% | 0% | 0% | 30.76% | 30.76% |
| (2,128,2) | 0% | 0% | 0% | 30.76% | 30.76% |

Figura 6. Gráfico de los resultados obtenidos por cada operador BLX-0.5

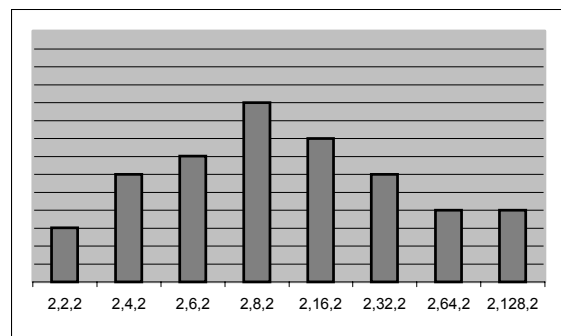


TABLA IV. ANÁLISIS DE LOS RESULTADOS OBTENIDOS POR CADA OPERADOR SBX-1

| Op. SBX 1 | MM /MT | MM /ST | TM | ST / NMM | T M/S |
|-----------|--------|--------|---------------|----------|---------------|
| (2,2,2) | 15.38% | 7.69% | 23.07% | 23.07% | 46.14% |
| (2,4,2) | 23.07% | 7.69% | 30.76% | 46.14% | 76.9% |
| (2,6,2) | 0% | 15.38% | 15.38% | 38.45% | 53.83% |
| (2,8,2) | 0% | 30.76% | 30.76% | 30.76% | 61.52% |
| (2,16,2) | 0% | 0% | 0% | 30.76% | 30.76% |
| (2,32,2) | 0% | 0% | 0% | 23.07% | 23.07% |
| (2,64,2) | 0% | 0% | 0% | 30.76% | 30.76% |
| (2,128,2) | 0% | 7.69% | 7.69% | 0% | 7.69% |

Figura 7. Gráfico de los resultados obtenidos por cada operador SBX-1

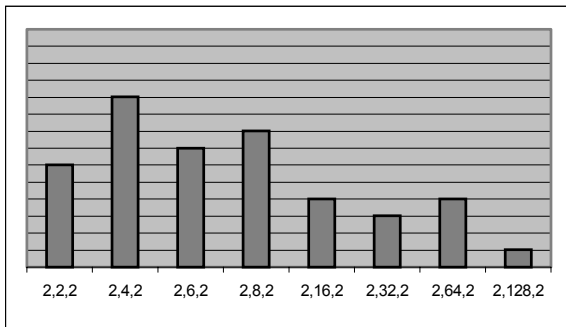


TABLA V. ANALISIS DE LOS RESULTADOS OBTENIDOS POR CADA OPERADOR FR-0.5

| Op. SBX-5 | MM /MT | MM /ST | TM | ST / NMM | T M/S |
|-----------|--------|--------|---------------|----------|---------------|
| (2,2,2) | 0% | 7.69% | 7.69% | 23.07% | 30.76% |
| (2,4,2) | 0% | 15.38% | 15.38% | 61.52% | 76.9% |
| (2,6,2) | 0% | 23.07% | 23.07% | 53.83% | 76.9% |
| (2,8,2) | 7.69% | 30.76% | 38.45% | 61.52% | 99.97% |
| (2,16,2) | 0% | 15.38% | 15.38% | 76.9% | 92.28% |
| (2,32,2) | 0% | 7.69% | 7.69% | 61.52% | 69.21% |
| (2,64,2) | 0% | 23.07% | 23.07% | 30.76% | 53.83% |
| (2,128,2) | 0% | 7.69% | 7.69% | 30.76% | 38.45% |

Figura 8. Gráfico de los resultados obtenidos por cada operador FR-0.5

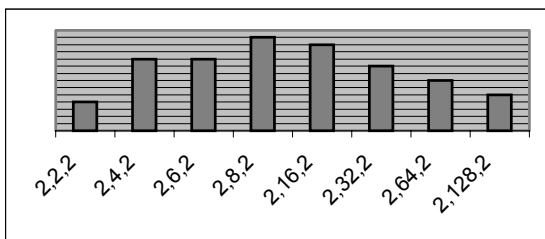


TABLA VI. ANÁLISIS DE LOS RESULTADOS OBTENIDOS POR CADA OPERADOR ARITMETICO

| Op. Aritmético | MM /MT | MM /ST | TM | ST / NMM | TM/S |
|----------------|--------|--------|---------------|----------|---------------|
| (2,2,2) | 69.21% | 23.07% | 92.28% | 7.69% | 100% |
| (2,4,2) | 0% | 7.69% | 7.69% | 15.38% | 23.07% |
| (2,6,2) | 0% | 0% | 0% | 23.07% | 23.07% |
| (2,8,2) | 0% | 0% | 0% | 23.07% | 23.07% |
| (2,16,2) | 0% | 0% | 0% | 0% | 0% |
| (2,32,2) | 0% | 0% | 0% | 0% | 0% |
| (2,64,2) | 0% | 0% | 0% | 0% | 0% |
| (2,128,2) | 0% | 0% | 0% | 0% | 0% |

Figura 9. Gráfico de los resultados obtenidos por cada operador Aritmético

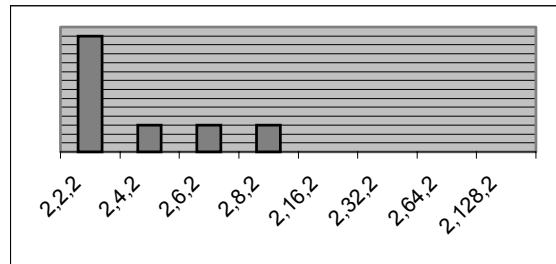
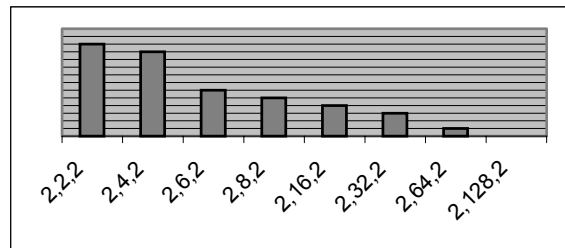


TABLA VII. ANÁLISIS DE LOS RESULTADOS OBTENIDOS POR CADA OPERADOR DE CRUCE EN DOS PUNTOS

| Op. En 2 puntos | MM /MT | MM /ST | TM | ST / NMM | TM/S |
|-----------------|--------|--------|---------------|----------|---------------|
| (2,2,2) | 15.38% | 69.21% | 84.59% | 7.69% | 92.28% |
| (2,4,2) | 0% | 15.38% | 15.38% | 69.21% | 84.59% |
| (2,6,2) | 0% | 0% | 0% | 46.14% | 46.14% |
| (2,8,2) | 0% | 0% | 0% | 38.45% | 38.45% |
| (2,16,2) | 0% | 0% | 0% | 30.76% | 30.76% |
| (2,32,2) | 0% | 0% | 0% | 23.07% | 23.07% |
| (2,64,2) | 0% | 0% | 0% | 7.69% | 7.69% |
| (2,128,2) | 0% | 0% | 0% | 0% | 0% |

Figura 10. Gráfico de los resultados obtenidos por cada operador de cruce en dos puntos



Según los resultados mostrados se puede comprobar que el comportamiento de los operadores de cruce varía dependiendo del número de descendientes que generan, así como del grupo de la taxonomía al que pertenecen y por tanto de las características asociadas a cada operador.

Para los operadores **BLX-0.5** y **FR-0.5**, según las Tablas III y V y las Figuras 6 y 8, se observa como los mejores resultados se consiguen con ocho y dieciseis descendientes. Con el operador **SBX-1** (Tabla IV y Figura 7) los mejores resultados se consiguen con cuatro descendientes, siendo también significativos los resultados alcanzados con seis y ocho descendientes. En general, para estos tres operadores, los mejores resultados se consiguen cuando el número de descendientes oscila entre cuatro y dieciseis.

Estos operadores, pertenecientes al grupo OCE de la taxonomía presentada en la Sección 2, y con los parámetros utilizados, presentan una alta diversidad en la generación de descendientes. El aumento del

número de descendientes y selección de los dos mejores, que conlleva un aumento de la presión selectiva, mejora los resultados obtenidos por el operador (2,2,2) clásico. Sin embargo esto es así solamente hasta cierto número de descendientes. Hasta ocho descendientes se mantiene un buen equilibrio entre exploración y explotación, ya que la presión selectiva no es muy grande y hay bastante diversidad en la población. En cambio, al pasar de ocho descendientes y según se va aumentando este número, también se incrementa la presión selectiva, lo que provoca una pérdida de diversidad.

Para los operadores de cruce **aritmético** y en **2 puntos**, que presentan una mayor explotación y por tanto menor diversidad asociada, el aumento de descendientes y selección de los dos mejores no consigue buenos resultados debido a que se produce una explotación demasiado grande (Tablas VI y VII y Figuras 9 y 10). En los dos casos es el operador clásico (2,2,2) el que logra los mejores resultados, empezando a empeorar los resultados a partir de cuatro descendientes. Este empeoramiento es especialmente significativo para el operador aritmético, mientras que para los operadores de cruce en 2 puntos, la diferencia de resultados entre el operador (2,2,2) y los operadores (2,4,2), (2,6,2) y (2,8,2) no es tan pronunciada. Es significativo el hecho de que un número muy elevado de descendientes, concretamente 64 y 128, no consiguen buenos resultados prácticamente en ningún caso.

VI. CONCLUSIONES

El número de descendientes más idóneo depende del tipo de operador de cruce que se utilice. Centrándonos en el estudio de los operadores aritmético y en dos puntos y según el número de descendientes que generan, desde 2 hijos (operador clásico) hasta 128, se observa que no consiguen mejoras con el aumento del número de descendientes. Para estos dos operadores los mejores resultados los consigue el operador clásico (2,2,2), seguido del (2,4,2), manteniendo buenos resultados hasta los ocho descendientes. A partir de este número de hijos no se consiguen buenos resultados.

En cambio, para los operadores BLX-0.5, SBX-1 y FR-0.5, del grupo OCE de la taxonomía, los resultados van mejorando progresivamente desde el (2,2,2) hasta el (2,8,2). A partir del (2,16,2) y hasta el (2,128,2), los resultados van empeorando, con lo que parece evidente que el generar múltiples descendientes proporciona buenos resultados, pero sólo hasta un cierto número. Si se generan demasiados descendientes, la presión selectiva

puede llegar a ser muy alta y por tanto hacer que se produzca una pérdida de diversidad en la población.

Se puede concluir por tanto, que para estos tres operadores, los mejores resultados se consiguen cuando el número de descendientes oscila entre cuatro y dieciseis, poniéndose de manifiesto que el uso de múltiples descendientes provoca mejoras significativas en el comportamiento de dichos operadores.

Hay que destacar el hecho de que los parámetros que se han utilizado para los operadores BLX, SBX y FR ($\alpha = 0.5$, $\eta = 1$ y $d = 0.5$), que proporcionan una alta diversidad, pueden influir directamente sobre los resultados obtenidos, por lo que puede ser interesante realizar los experimentos con otros valores para dichos parámetros.

Otra posible línea futura es la generación de los múltiples descendientes utilizando distintos mecanismos, dando lugar a operadores de cruce con múltiples descendientes híbridos.

REFERENCIAS

- [1] J.E. Baker, Adaptive Selection Methods for Genetic Algorithms. En: Proc. Of the First Int. Conf. On Genetic Algorithms and their Applications, J.J. Grefenstette (Ed.) L. Erlbaum Associates, Hillsdale, MA, 1987, 14-21.
- [2] J.E. Baker, Reducing bias and inefficiency in the selection algorithm. En: Proc. Of the Second Int. Conf. On Genetic Algorithms and their Applications, J.J. Grefenstette, (Ed.), Hillsdale, NJ: Lawrence Erlbaum, 1987, 14-21.
- [3] H.-G. Beyer, K. Deb, On Self-Adaptive Features in Real Parameter Evolutionary Algorithms. IEEE Transactions on Evolutionary Computation, vol 5, no. 3, 2001, 250-270.
- [4] K. Deb, R.B. Agrawal, Simulated Binary Crossover for Continuous Search Space. Complex Systems, 9, 1995, 115-148.
- [5] K. Deb, Multi-objective optimization using evolutionary algorithms. Chichester: Wiley, 2001.
- [6] K.A. De Jong, An Analysis of the Behavior of a Class of Genetic Adaptive Systems. Tesis Doctoral, Universidad de Michigan, 1975.
- [7] K.A. De Jong, W.M. Spears, A Formal Analysis of the Role of Multi-Point Crossover in Genetic Algorithms. Annals of Mathematics and Artificial Intelligence, 5(1), 1992, 1-26.

- [8] A.E. Eiben, T. Bäck, An Empirical Investigation of Multi-PARENT Recombination Operators in Evolution Strategies. *Evolutionary Computation*, 5(3), 1997, 347-365.
- [9] L.J. Eshelman, A. Caruana, J.D. Schaffer, Biases in the Crossover Landscape. En: Proc. of the Third Int. Conf. on Genetic Algorithms, J. David Schaffer (Ed.), Morgan Kaufmann Publishers, San Mateo, 1989, 86-91.
- [10] L. J. Eshelman, J. D. Schaffer, Real-Coded Genetic Algorithms and Interval-Schemata. En : Foundations of Genetic Algorithms 2, L.D. Whitley (Ed.), Morgan Kaufmann Publishers, San Mateo, 1993, 187-202.
- [11] L.J. Eshelman, K.E. Mathias, J.D. Schaffer, Crossover Operator Biases: Exploiting the Population Distribution. En Proc. of the Seventh Int. Conf. On Genetic Algorithms, T. Bäck (Ed.), Morgan Kaufmann, San Mateo, CA, 1997, 354-361.
- [12] S. Esquivel, R. Gallard, Z. Michalewicz. MPC: Another Approach to Crossover in Genetic Algorithms. *Actas del Primer Congreso de Ciencias de la Computación*, 1995, 141-150.
- [13] S. Esquivel, A. Leiva, R. Gallard. Multiple Crossover per Couple in Genetic Algorithms. Proc. of the 4th IEEE Int. Conf. on Evolutionary Computation (ICEC'97), Indianapolis, USA, 1997, 103-106.
- [14] D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, NEW York, 1989.
- [15] F.Herrera, M. Lozano, J.L. Verdegay, Dynamic and Heuristic Fuzzy Connectives Based Crossover Operators for Controlling the Diversity and Convergence of Real-Coded Genetic Algorithms. *Int. Journal of Intelligent Systems* 11, 1996, 1013-1041.
- [16] F. Herrera, M. Lozano, J.L. Verdegay. Fuzzy Connectives Based Crossover Operators to Model Genetic Algorithms Population Diversity. *Fuzzy Sets and Systems* 92:1, 1997, 21-30.
- [17] F. Herrera, M. Lozano, J.L. Verdegay. Tackling Real-Coded Genetic Algorithms: Operators and tools for the Behaviour Analysis. *Artificial Intelligence Review* 12, 1998, 265-319.
- [18] F. Herrera, M. Lozano, A. Sánchez. A Taxonomy for the Crossover Operator for Real Coded Genetic Algorithms: An Experimental Study. *International Journal of Intelligent Systems*, Vol. 18, 2003, 309-338.
- [19] H. Kita, I. Ono, S. Kobayashi. Multi-parental extension of the unimodal normal distribution crossover for real-coded genetic algorithms. Proc. of the Congress on Evolutionary Computation, 1999. 1581-1587.
- [20] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag, New York, 1992.
- [21] Z. Michalewicz, G. Nazhiyath, and M. Michalewicz, A Note on Usefulness of Geometrical Crossover for Numerical Optimization Problems. En: Proceedings of the 5th Annual Conference on Evolutionary Programming, San Diego, CA, 29 February - 3 March. MIT Press, Cambridge, MA, 1996, 305-312.
- [22] D. Ortiz, C. Hervás, J. Muñoz. Genetic algorithm with crossover based on confidence intervals as an alternative to least squares estimation for nonlinear models. 4th Metaheuristics International Conference, Porto, Portugal, 2001, 343-347.
- [23] D. Schlierkamp-Voosen, H. Mühlenbein, Strategy Adaptation by Competing Subpopulations. En: Paralell Problem Solving from Nature 3, Y. Davidor, H.-P. Schwefel, R. Mämmmer (Eds.), Berlín, Alemania, Springer-Verlag, 1994, 199-208.
- [24] S. Tsutsui, M. Yamamura, T. Higuchi. Multi-parent recombination with simplex crossover in real-coded genetic algorithms. Proc. of the Genetic and Evolutionary Computation Conference (GECCO-99), 1999, 657-664.
- [25] H.M. Voigt, H. Mühlenbein, D. Cvetkovic, Fuzzy Recombination for the Breeder Genetic Algorithm. En : Proc. Of the Sixth Int. Conf. on Genetic Algorithms, L. Eshelman (Ed.), Morgan Kaufmann Publishers, San Francisco, 1995, 104-111.
- [26] A. Wright, Genetic Algorithms for Real Parameter Optimization. En: Foundations of Genetic Algorithms 1, G.J.E. Rawlin (Ed.), Morgan Kaufmann, San Mateo, 1991, 205-218.

APÉNDICE A (OPERADORES DE CRUCE)

Cruce en dos puntos. Aleatoriamente se seleccionan dos puntos de cruce ($i, j \in \{1, 2, \dots, n-1\}$ con $i < j$), y los segmentos de los padres, definidos por ellos, se intercambian para la generación de los dos hijos:

$$H_1 = (c_1^1, c_2^1, \dots, c_i^2, c_{i+1}^2, \dots, c_j^2, c_{j+1}^1, \dots, c_n^1),$$

$$H_2 = (c_1^2, c_2^2, \dots, c_i^1, c_{i+1}^1, \dots, c_j^1, c_{j+1}^2, \dots, c_n^2).$$

Cruce Aritmético. Se generan dos hijos,

$$H_k = (h_1^k, \dots, h_i^k, \dots, h_n^k), \quad k = 1, 2, \text{ donde}$$

$$h_i^1 = \lambda \cdot c_i^1 + (1 - \lambda) \cdot c_i^2$$

$$h_i^2 = \lambda \cdot c_i^2 + (1 - \lambda) \cdot c_i^1,$$

con $\lambda \in [0, 1]$.

BLX- α . Se generan dos hijos:

$$H_k = (h_1^k, \dots, h_i^k, \dots, h_n^k), \quad k = 1, 2, \text{ donde}$$

h_i^k es un número elegido aleatoriamente en el intervalo $[C_{\min} - I\alpha, C_{\max} + I\alpha]$ con una distribución uniforme, donde

$$C_{\max} = \max\{c_i^1, c_i^2\},$$

$$C_{\min} = \min\{c_i^1, c_i^2\},$$

$$I = C_{\max} - C_{\min}.$$

Cruce difuso. Se generan dos hijos,

$$H_k = (h_1^k, \dots, h_i^k, \dots, h_n^k), \quad k = 1, 2. \text{ La}$$

probabilidad de que el i -ésimo gen en un descendiente tenga el valor v_i viene dada por la distribución $p(v_i) \in \{\phi(c_i^1), \phi(c_i^2)\}$ donde $\phi(c_i^1)$ y $\phi(c_i^2)$ son distribuciones triangulares de probabilidad con las siguientes características (se asume que $c_i^1 \leq c_i^2$):

| Distr. Prob. | Valor Mínimo | Valor Modal | Valor Máximo |
|---------------|-----------------------------------|-------------|-----------------------------------|
| $\phi(c_i^1)$ | $c_i^1 - d \cdot c_i^2 - c_i^1 $ | c_i^1 | $c_i^1 + d \cdot c_i^2 - c_i^1 $ |
| $\phi(c_i^2)$ | $c_i^2 - d \cdot c_i^2 - c_i^1 $ | c_i^2 | $c_i^2 + d \cdot c_i^2 - c_i^1 $ |

Cruce por simulación binaria (SBX- η):

Se generan dos hijos,

$$H_k = (h_1^k, \dots, h_i^k, \dots, h_n^k), \quad k = 1, 2, \text{ donde:}$$

$$h_i^1 = \frac{1}{2} \cdot [(1 - \beta_k) \cdot c_i^1 + (1 + \beta_k) \cdot c_i^2] \text{ y}$$

$$h_i^2 = \frac{1}{2} \cdot [(1 + \beta_k) \cdot c_i^1 + (1 - \beta_k) \cdot c_i^2]$$

$\beta_k (\geq 0)$ es una muestra de generador de números aleatorios con densidad:

$$p(\beta) = \begin{cases} \frac{1}{2} \cdot (\eta + 1) \beta^\eta, & \text{si } 0 \leq \beta \leq 1 \\ \frac{1}{2} \cdot (\eta + 1) \frac{1}{\beta^{\eta+2}}, & \text{si } \beta > 1. \end{cases}$$

Esta distribución puede obtenerse fácilmente a partir de una uniforme $u(0,1)$ con una semilla aleatoria usando la transformación:

$$\beta(u) = \begin{cases} (2u)^{\frac{1}{\eta+1}} & \text{si } u(0,1) \leq \frac{1}{2} \\ [2(1-u)]^{-\frac{1}{\eta+1}} & \text{si } u(0,1) > \frac{1}{2}. \end{cases}$$