

ESTUDIO PARA ANALIZAR EL DESEMPEÑO DEL PARÁMETRO CR DE ED (ALGORITMO EVOLUCIÓN DIFERENCIAL) EN SUS VERSIONES ORIGINALES EN PROBLEMAS DE OPTIMIZACIÓN CON RESTRICCIONES

Rubí del Carmen Gómez Ramón
José Angel Pérez Rejón*

Resumen

La Evolución Diferencial (ED) es uno de los algoritmos evolutivos propuesto en 1995 y ha demostrado tener un desempeño muy competitivo en problemas de optimización en espacios restringidos. Con base en una revisión de la literatura especializada se ha notado que no se cuenta con estudios que permitan conocer el comportamiento del parámetro Cr en las variantes de tipo discreto y en sus versiones originales. En este trabajo se propone realizar un estudio del parámetro Cr de las variantes de ED para determinar cuáles proveen el mejor desempeño al resolver un conjunto de funciones de prueba.

Palabras clave

Evolución diferencial, variantes, parámetro Cr, funciones de prueba y medidas estadísticas.

Introducción

Evolución diferencial (ED) es uno de los algoritmos evolutivos que nació en 1994 de los intentos de Price y Storn por resolver el problema polinomial Chebychev. Al siguiente año estos dos autores propusieron ED en [7] para la optimización de funciones no lineales y no diferenciables sobre espacios continuos.

El algoritmo de ED es un método de búsqueda directa y estocástica, que ha demostrado ser efectivo, eficiente y robusto

[7] en una gran variedad de aplicaciones como el aprendizaje de una red neuronal [2], diseño de un filtro-IIR [6], la optimización aerodinámica [8] y otras encontradas en [3]. ED cuenta con un número de características importantes entre ellas se destacan las siguientes: tiene la capacidad de manejar funciones objetivos no diferenciables, no lineales y multimodales, suele converger al óptimo, utiliza pocos parámetros de control, etc. [5].

Descripción del algoritmo de ED

El algoritmo de Evolución Diferencial maneja dos poblaciones diferentes dentro del proceso de optimización: la población primaria y una población secundaria. La primera almacena la población actual, que a su vez se genera al inicio de manera aleatoria, y la población secundaria almacena a los individuos que resultan exitosos en el proceso de selección. La representación gráfica de ambas poblaciones se muestra en la fig. 1.

Metodología

Primer paso del algoritmo de ED, se genera aleatoriamente la población inicial, donde los vectores están uniformemente distribuidos en el espacio de búsqueda dentro de los límites definidos.

Segundo paso: utiliza la mutación y la recombinación para

producir un solo vector hijo (trial) por cada vector de la población. Para la generación del vector hijo se requiere primero llevar a cabo el proceso de mutación, el cual consiste en: a) Seleccionar aleatoriamente tres vectores diferentes entre sí, b) Se restan dos de ellos, a la diferencia se aplica un peso dado por un factor F y por último d) Se suma la diferencia escalada al tercer vector; de esta manera se crea un vector llamado vector de mutación.

La Fig.2 muestra gráficamente el proceso de mutación. El factor escalar $F \in [0,1]$ es un número real positivo que escala el vector de diferencia. La mutación en ED es el principal mecanismo para generar nuevas direcciones de búsqueda.

Tercer paso: se hace la cruce o recombinación entre el vector padre y el vector de mutación con el propósito de generar un vector hijo. $Cr \in [0,1]$ controla la cantidad de variables de decisión (parámetros) que se copian del vector de mutación al vector hijo. La cruce intercambia información entre los vectores participantes con el propósito de buscar una mejor solución en el espacio de búsqueda.

Todos los vectores de la población serán seleccionados sólo

* Docentes de tiempo completo en la Facultad Ciencias de la Información de la Universidad Autónoma del Carmen.

una vez como padre sin depender del valor de la función objetivo (valor de aptitud). Paso 4: Después de obtener el vector hijo, éste se evalúa en la función objetivo del problema y se compara con el vector padre con base en su valor de la función objetivo, y el mejor pasa a formar parte a la población de la siguiente generación. Si el vector padre todavía es mejor, éste es conservado en la siguiente generación.

En ED el proceso de mutación, de recombinación y de selección es repetido hasta que llegue a un criterio de terminación especificado por el usuario, en este caso nos referimos al número máximo de generaciones (iteraciones) Gmax.

Variantes de ED

Las variantes se distinguen por la manera que se lleva a cabo el proceso de mutación y de recombinación. En este trabajo se propone una clasificación de las variantes con respecto al tipo de recombinación (cruza) que utilizan:

a) Recombinación discreta: consiste en copiar directamente al vector hijo las variables de decisión del vector de mutación o del vector padre. Esta recombinación o cruza es utilizada por las siguientes variantes: rand/1/bin, rand/1/exp, best/1/bin y best/1/exp.

La nomenclatura de estas cuatro variantes es la siguiente DE/x/y/z, donde DE indica que estamos trabajando con el Algoritmo de Evolución Diferencial, “x” representa la forma en la que es elegido el vector base, que puede ser aleatoriamente “rand” o aquel vector con el mejor valor de la función objetivo en la generación actual best, “y” es el número de pares de vectores para el cálculo de las diferencias y finalmente “z” representa el tipo de recombinación.

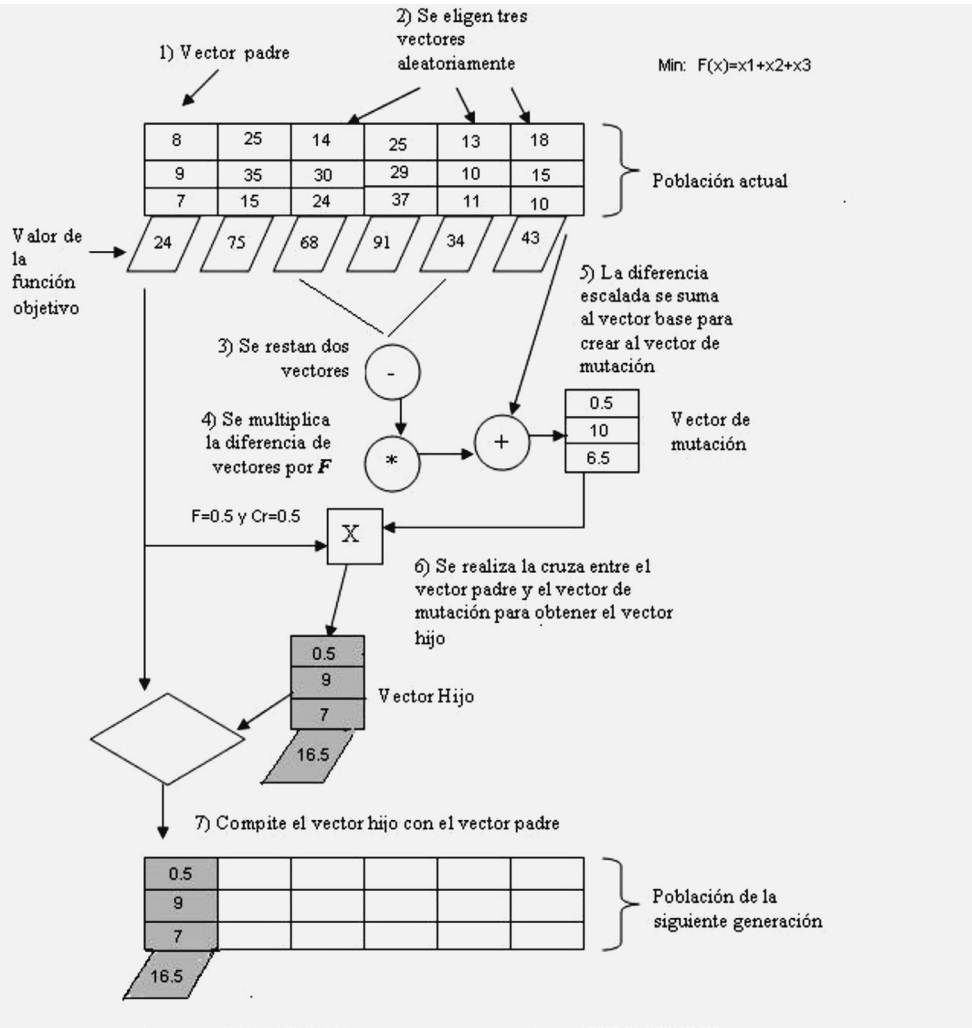


Fig. 1. Diagrama de la versión original de ED.

A su vez la recombinación discreta se clasifica en dos tipos: binomial y exponencial. Si se utiliza la cruce binomial (bin) las variables de decisión del vector de mutación son copiados en el vector hijo si el valor de $rand_j$ es menor al valor Cr , en caso contrario, se copian del vector padre, esta comparación se realiza para todas las variables. En la cruce exponencial (exp) las variables del vector de mutación se copian en el vector hijo mientras que el valor de $rand_j$ sea menor al valor del parámetro Cr , pero una vez que $rand_j$ supere al valor de Cr , la variable actual y el resto de las variables se copian del vector padre. La fig.3 ilustra un ejemplo de la manera como opera la cruce binomial y la cruce exponencial. De estas cuatro variantes, la $rand/1/bin$ es la versión original del algoritmo de Evolución Diferencial y la más popular.

b) Recombinación aritmética: genera al vector hijo mediante una combinación lineal entre el vector padre y el vector base, el cual puede ser elegido aleatoriamente $rand$ o bien, aquel con el mejor valor de la función objetivo en la generación actual $best$. Para esta cruce no se usa el parámetro Cr . Las variantes $current-to-rand$ y $current-to-best$ utilizan este tipo de cruce. El término $current$ indica que al vector actual también conocido como padre se le va sumar las diferencias escaladas. Además introducen un coeficiente real $K \in [0,1]$ responsable del nivel de combinación que ocurre entre el vector padre ($target$) y el vector base. F es el responsable de escalar a los vectores de diferencias. Véase la fig.4.

c) Las variantes $current-to-rand/p/bin$ y la $current-to-rand/p/exp$ manejan al mismo tiempo los dos tipos de recombinación descritos anteriormente (recombinación discreta y la aritmética).

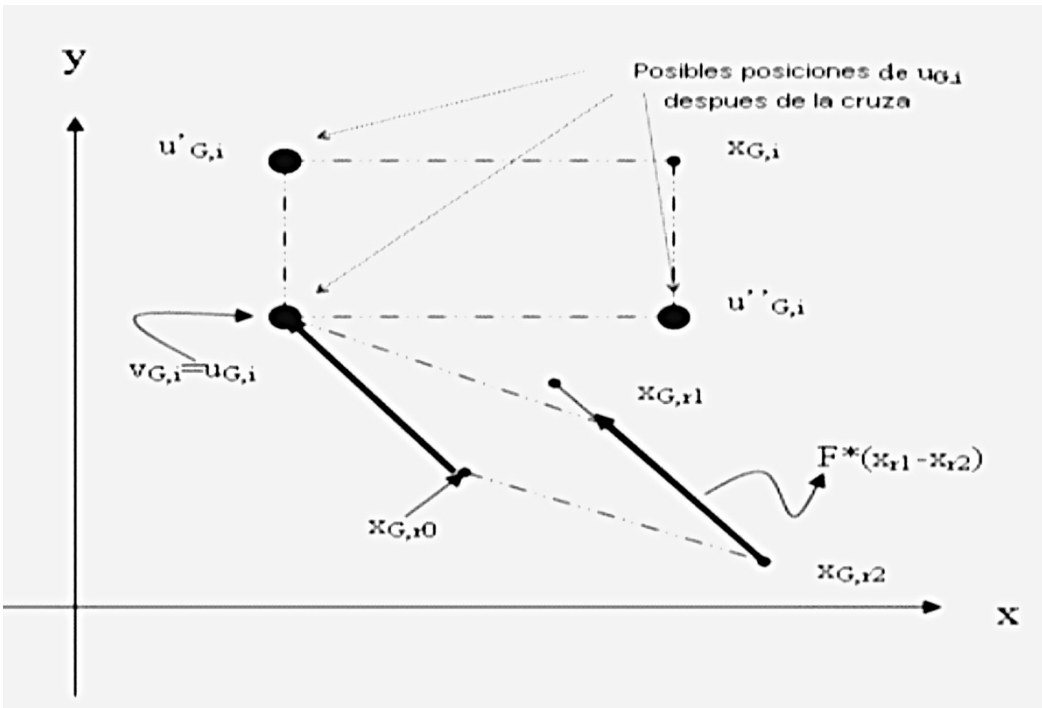
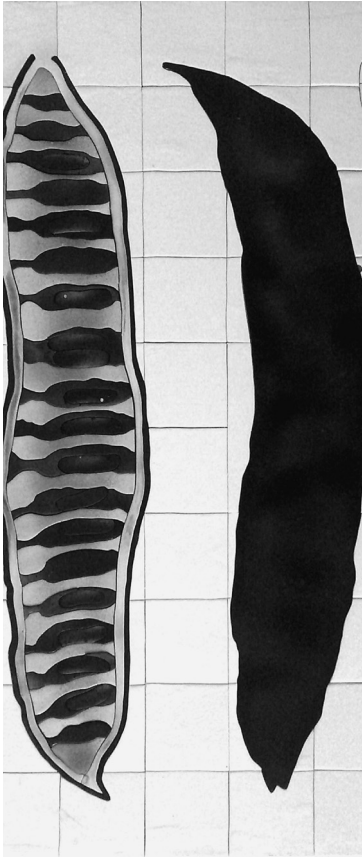


Fig. 2. Esquema del operador de mutación del algoritmo de Evolución Diferencial

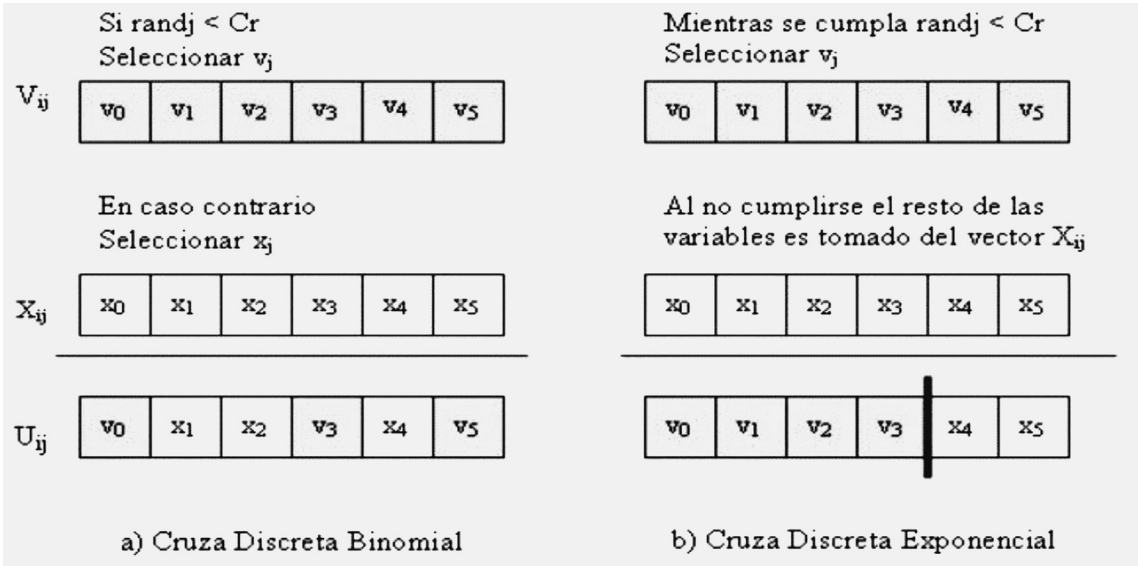


Fig. 3. (a) Cruza binomial (b) Cruza exponencial asumiendo $\text{rand}=0$

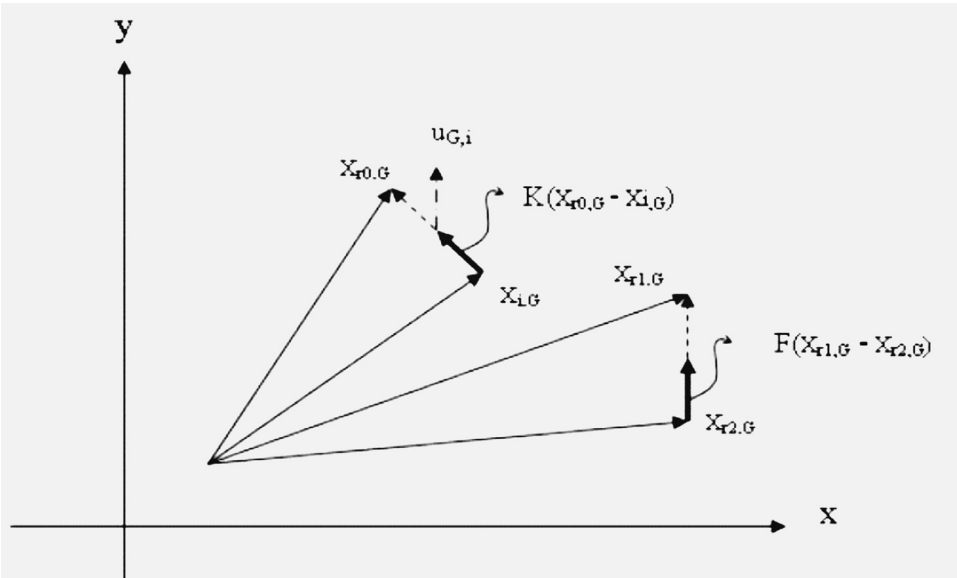


Fig. 4. Esquema del proceso de mutación y de recombinación de la variante current-to-rand

Experimentación

El estudio propuesto en este trabajo consta de analizar el parámetro Cr en 2 variantes de Evolución Diferencial empleando 12 funciones de pruebas estándares encontradas en la literatura especializada para evaluar el desempeño de cada una de ellas y en términos de resultados estadísticos poder determinar cuáles son las que presenta un mejor desempeño general. Las variantes de ED consideradas para la primera fase son: rand/1/bin y

best/1/bin, se fijó el número total de evaluaciones de la función objetivo a 500,040. Los valores de los parámetros son mostrados en la Tabla 1. Por cada variante y función se efectuaron 30 corridas (cada corrida con diferente semilla inicial). Las restricciones de igualdad se transformaron en restricciones de desigualdad con una tolerancia de $\epsilon=0.0001$ y para el manejo de restricciones se usaron las reglas de factibilidad introducida por Deb [1].

Tabla 1. Parámetros de control utilizado por el Algoritmo de ED.

Parámetros	Valor
Np = Tamaño de la población	90
Gmax = Número máximo de generaciones	5556
Cr = Probabilidad de Cruza	1.0
F = Factor escalar	0.9

Las 24 funciones de pruebas que se emplearon, propuestas en [4], tienen diferentes características tales como el tipo de función objetivo (lineal, no lineal, cuadrática, etc.), el tipo y número de restricciones (igualdad y desigualdad), así como también el número de variables de decisión del problema

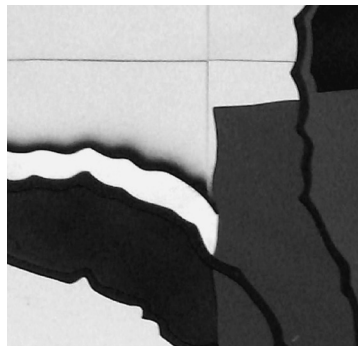
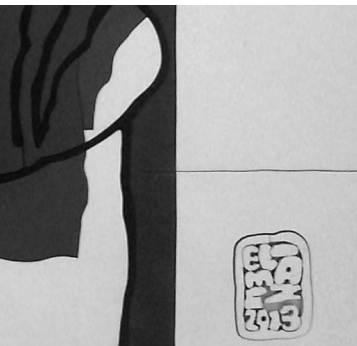
Resultados y análisis

A los resultados obtenidos de las 30 corridas por cada variante-problema se aplicaron medidas estadísticas para determinar cuál variante alcanza de manera consistente el óptimo en la mayor cantidad de problemas.

Las medidas estadísticas que se emplearon son: el mejor (indica la mejor solución encontrada), el peor (la peor solución encontrada), la media, la mediana y la desviación estándar. (Véase la tabla 2).

Tabla 2. Resultados estadísticos correspondientes a las 30 ejecuciones de cada variante para las 12 funciones El valor remarcado en negritas indica el valor óptimo global es decir la mejor solución y las celdas que contienen un guión horizontal indican que no se logró obtener soluciones factibles en las 30 corridas.

funciones	medidas	Variantes		Funciones	medidas	Variantes	
		Rand/1/bin	Best/1/bin			Rand/1/bin	Best/1/bin
G01	Óptimo	-15	-15	G07	Óptimo	24.306	24.306
	Mejor	-15	-		Mejor	24.306	24.306
	Peor	-14.999951	-		Peor	24.306	24.306
	Media	-14.9999946	-		Media	24.306	24.306
	mediana	-14.999998	-		mediana	24.306	24.306
	Desv. estándar	9.97 E.06	-		Desv. estándar	0	0
G02	Óptimo	0.803619	0.803619	G08	Óptimo	0.096	0.096
	Mejor	0.803608	0.751132		Mejor	0.096	0.096
	Peor	0.714626	0.28762		Peor	0.096	0.096
	Media	0.77291897	0.61232043		Media	0.096	0.096
	mediana	0.784534	0.612947		mediana	0.096	0.096
	Desv. estándar	0.02861662	0.08491737		Desv. estándar	4.230 E-17	4.230 E-17
G03	Óptimo	1	1	G09	Óptimo	680.630	680.630
	Mejor	0.40623	1		Mejor	680.630	680.630
	Peor	0.007327	0.825392		Peor	680.630	680.630
	Media	0.148079	0.99077		Media	680.630	680.630
	mediana	0.121313	1		mediana	680.630	680.630
	Desv. estándar	0.11822676	0.04127291		Desv. estándar	1.160 E-13	1.160 E-13
G04	Óptimo	-30665.539	-30665.539	G10	Óptimo	7049.331	7049.331
	Mejor	-30665.5387	-		Mejor	7049.331	7049.331
	Peor	-30665.5387	-		Peor	7049.331	7049.331
	Media	-30665.5387	-		Media	7049.331	7049.331
	mediana	-30665.5387	-		mediana	7049.331	7049.331
	Desv. estándar	2.22 E-11	2.22 E-11		Desv. estándar	4.630 E-12	0.000 E+00
G05	Óptimo	5126.4981	5126.4981	G11	Óptimo	0.750	0.750
	Mejor	5126.49671	-		Mejor	0.750	0.750
	Peor	5126.49671	-		Peor	0.750	0.750
	Media	5126.49671	-		Media	0.750	0.750
	mediana	5126.49671	-		mediana	0.750	0.750
	Desv. estándar	1.85 E-12	-		Desv. estándar	1.130 E-16	1.130 E-16
G06	óptimo	-6961.81388	-	G12	Óptimo	1	1
	mejor	-6961.81388	-		Mejor	1	1
	peor	-6961.81388	-		Peor	0.990	1
	media	-6961.81388	-		Media	0.995	1
	mediana	-6961.81388	-		mediana	0.993	1
	Desv. estándar	0	0		Desv. estándar	3.864 E-03	1.580 E-06



En la tabla 2 se observa que para la función g01 la variante rand/1/bin llega al óptimo global. En la función g02 las dos proporcionan soluciones factibles pero la variante rand/1/bin es la única que llega a la solución óptima, en g03 también las dos variantes generan soluciones factibles, pero la que llegó a la solución óptima es la variante best/1/bin, en g04 las dos llegaron al valor óptimo, en g05 solo se aproxima a la solución óptima la variante rand/1/bin.

Para las funciones g06, g07, g08, g09, g10, g11 y g12 las dos variantes llegan a la solución óptima de manera muy consistente y en la función g07 las dos llegan a converger al óptimo en las 30 corridas (Desviación estándar de 0).

Conclusiones

La variante rand/1/bin llegó al óptimo en 9 funciones (Véase la tabla 2). En dos funciones (g06 y g07) obtuvo valor cero en la desviación estándar. Presentó valores menores de desviaciones estándar

en 3 funciones (g03, g05 y g11). La variante best/1/bin llegó a soluciones óptimas en 7 funciones. En las funciones g06 y g07 obtuvo en la desviación estándar un valor de cero.

Referencia

1. Kalyanmoy Deb. *An Efficient Constraint Handling Method for Genetic Algorithms*. *Computer Methods in Applied Mechanics and Engineering*, 186(2/4):311-338, 2000.
2. Ilonen J., Kamarainen J.K., and Lampinen J. *Differential evolution training algorithm for feed-forward neural networks*. In: *Neural Processing Letters*, 7:93-105, 2003.
3. Price K., Storn R., and Lampinen J. *Differential Evolution - A Practical Approach to Global Optimization*. Springer-Verlag, 2006.
4. J. J. Liang, T. P. Runarsson, E. Mezura Montes, M. Clerc, P. N. Suganthan, C. A. Coello Coello, and K. Deb. *Problem definitions and evaluation criteria for the cec 2006 special session on constrained real-parameter optimization*. [Online] Available: <http://www.ntu.edu.sg/home/EPNsgan,20.8.2007>, September 2006.
5. Kenneth V. Price. *An Introduction to Differential Evolution*. In David Corne, Marco Dorigo, and Fred Glover; editors, *New Ideas in Optimization*, pages 79-108. Mc Graw-Hill, UK, 1999.
6. Rainer Storn. *Differential evolution design of an iir-iter*. In: *Proceedings of IEEE International Conference on Evolutionary Computation ICEC'96*. IEEE Press, pages 268-273, 1996.
7. Rainer Storn and Kenneth Price. *Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces*. Technical Report TR-95-012 International Computer Science Institute, Berkley, 1995.
8. Rogalsky T., Derksen R.W., and Kocabiyik S. *Differential evolution in aerodynamic optimization*. In: *Proceedings of the 46th Annual Conference of the Canadian Aeronautics and Space Institute*, pages 29{36, 1999.