

Tipo de artículo:

Temática:

Recibido: dd/mm/aa | Aceptado: dd/mm/aa | Publicado: dd/mm/aa

Algoritmo basado en Nubes de Partículas para la optimización global de funciones en espacios multidimensionales complejos

Particle Swarm based algorithm for global optimization of functions in multidimensional complex spaces

Gonzalo Nápoles Ruiz^{1*}, Isel Grau García¹, Rafael Bello Pérez¹

¹Laboratorio de Inteligencia Artificial. Centro de Estudios de Informática. Facultad de Matemática, Física y Computación. Universidad Central “Marta Abreu” de Las Villas, Carretera a Camajuaní, km. 5 ½, Santa Clara, Villa Clara, Cuba.

*Autor para la correspondencia: gnapoles@uclv.edu.cu

Resumen

Optimización basada en Nubes de Partículas es una meta-heurística bioinspirada que permite encontrar soluciones prometedoras en espacios de búsqueda complejos. En Optimización basada en Nubes de Partículas estándar es frecuente que la nube de partículas sea atraída por soluciones sub-óptimas, provocando la convergencia prematura del algoritmo y el estancamiento de la bandada. En este artículo se presenta una variante modificada del algoritmo Optimización basada en Nubes de Partículas, que haciendo uso de muestras aleatorias en vecindades variables dispersa la bandada cuando es detectado un estado de estancamiento, ofreciendo de esta manera una alternativa de escape a los óptimos locales. La efectividad y eficiencia del algoritmo propuesto es examinada y resultados experimentales muestran su capacidad para aproximar el mínimo global de nueve funciones *benchmark* reportadas en la literatura.

Palabras clave: convergencia prematura, estancamiento, muestreo aleatorio optimización basada en nubes de partículas, vecindades variables.

Abstract

Particle Swarm Optimization is a bio-inspired meta-heuristic that allows us to find promising solutions in complex search spaces. In standard Particle Swarm Optimization it is frequent for the particle swarm to be attracted by suboptimal solutions, causing the premature convergence of the algorithm and the swarm stagnation. This paper presents a Particle Swarm Optimization modified variant which uses random samples in variable neighbourhoods for dispersing the swarm whenever a stagnation state is detected, offering an escaping alternative from the local optima. The effectiveness and efficacy of the proposed algorithm is discussed and experimental results show its ability to approximate the global minimum of nine benchmark functions reported in the literature.

Keywords: *particle swarm optimization, premature convergence, random samples, stagnation, variable neighbourhoods.*

Introducción

Optimización basada en Nubes de Partículas (PSO por sus siglas en inglés) es una meta-heurística poblacional bioinspirada que simula el comportamiento social observado en grupos o bandadas de individuos biológicos. Esta técnica estocástica parte del principio que la inteligencia no está en los individuos sino en el colectivo, lo que permite la solución de problemas de optimización de forma distribuida sin el control centralizado en un individuo específico. Cada organismo (partícula) ajusta su “vuelo” de acuerdo a su propia experiencia y a la del resto de la bandada, imitando a aquellos que tengan un mejor desempeño. De esta forma la bandada de partículas sobrevuela el espacio de búsqueda localizando regiones prometedoras (Kennedy, 1995; Eberhart, 1995).

Tal vez la característica más atractiva del algoritmo PSO es su rápida convergencia. Sin embargo, en la solución de problemas multimodales complejos la nube de partículas frecuentemente es atraída por puntos estables que no son necesariamente óptimos globales (Kennedy, 2001; Parsopoulos, 2004), provocando un fenómeno conocido como estancamiento (*stagnation*) en el cual las partículas se agrupan alrededor de soluciones sub-óptimas con pocas probabilidades de escapar de tal situación; una vez que la bandada ha convergido prematuramente, las partículas continúan convergiendo dentro de una vecindad extremadamente pequeña del espacio de búsqueda (Evers, 2009) limitando notablemente la capacidad de exploración del algoritmo.

En este artículo se presenta una modificación al algoritmo PSO con coeficiente de restricción llamado Optimización basada en Nubes de Partículas con Muestreo Aleatorio en Vecindades Variables (*Particle Swarm Optimization with Random Sampling in Variable Neighbourhoods*, PSO-RSVN) el cual detecta y trata el estado de estancamiento logrando resultados prometedores en comparación con algoritmos reportados en la literatura.

Metodología Computacional

Optimización basada en Nubes de Partículas

PSO involucra un conjunto de agentes o partículas conocido como bandada (*swarm*) que recorre el espacio de soluciones tratando de localizar regiones prometedoras. Las partículas se interpretan como posibles soluciones del problema de optimización y son representadas como puntos n-dimensionales en el espacio de búsqueda. En el caso del PSO estándar, cada partícula (X_i) tiene su propia velocidad (V_i) acotada por un valor máximo ($V_{\text{máx}}$), una memoria de la mejor posición que ha obtenido y conocimiento de la mejor solución encontrada en su vecindad. En el proceso de búsqueda las partículas ajustan su posición de acuerdo a las ecuaciones:

$$V_i^{(k+1)} = V_i^{(k)} + c_1 r_1 (X_{\text{pbest}_i} - X_i^{(k)}) + c_2 r_2 (X_{\text{gbest}} - X_i^{(k)}) \quad (1)$$

$$X_i^{(k+1)} = X_i^{(k)} + V_i^{(k+1)} \quad (2)$$

donde k indexa la generación actual, c_1 y c_2 son constantes positivas, r_1 y r_2 son números aleatorios con distribución uniforme en el intervalo $[0, 1]$, mientras que X_{pbest_i} y X_{gbest} representan la mejor posición previa para la partícula X_i y la mejor posición encontrada por la bandada de manera global respectivamente.

El primer parámetro que modifica al algoritmo PSO es el peso de inercia (*inertia weight*, ω) adicionado por (Shi, 1998) para reemplazar a $V_{\text{máx}}$. La incorporación de este parámetro garantiza el balance entre las capacidades de

búsqueda local y global, pues un peso con valores altos facilitará la exploración, mientras que un peso bajo facilitará la explotación o intensificación. Sin embargo, la selección errónea del valor de este parámetro puede afectar la velocidad de convergencia del algoritmo, por lo que se recomienda su ajuste de forma dinámica como se muestra en la siguiente ecuación:

$$\omega_k = \omega_{\text{máx}} - \frac{\omega_{\text{máx}} - \omega_{\text{mín}}}{N} k \quad (3)$$

donde k es el ciclo actual, N se corresponde con el número de generaciones, mientras que $\omega_{\text{mín}}$ y $\omega_{\text{máx}}$ coinciden con los extremos del intervalo $[\omega_{\text{mín}}, \omega_{\text{máx}}]$ sobre el cual estará definido el k -ésimo peso de inercia.

El otro parámetro que modifica al algoritmo PSO es el coeficiente de restricción (*constriction coefficient*, Υ) introducido por (Clerc, 2002). Este parámetro garantiza que el algoritmo converja, evitando la explosión de la nube de partículas y puede ser expresado en función de c_1 y c_2 de la siguiente manera:

$$\Upsilon = \frac{2}{|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}|} \text{ y } \varphi = c_1 + c_2, \varphi > 4 \quad (4)$$

Un estudio comparativo de los dos métodos ha demostrado que el algoritmo PSO con coeficiente de restricción es en realidad un caso especial del algoritmo con peso de inercia, donde los valores de los parámetros se han determinado analíticamente (Eberhart, 2000). Las ecuaciones (5) y (6) muestran como se aplican ambos factores a la variante estándar de PSO:

$$V_i^{(k+1)} = \omega_{k+1} V_i^{(k)} + c_1 r_1 (X_{pbest_i} - X_i^{(k)}) + c_2 r_2 (X_{gbest} - X_i^{(k)}) \quad (5)$$

$$X_i^{(k+1)} = X_i^{(k)} + \Upsilon V_i^{(k+1)} \quad (6)$$

Aunque PSO con coeficiente de restricción (*constricted PSO*) es significativamente superior al PSO clásico (Bratton, 2007) no ofrece ningún mecanismo para detectar y tratar el estancamiento de la nube de partículas, lo que pudiera incidir negativamente en el rendimiento del algoritmo.

Detección del estado de estancamiento

El primer paso para tratar la convergencia prematura que sufre el algoritmo PSO es detectar el estado de estancamiento. Cuando PSO cae en un óptimo local todos los individuos se agrupan alrededor de esta solución perdiéndose la diversidad entre las partículas de la bandada. Esta situación provoca que a medida que avanza el ciclo evolutivo sea más difícil encontrar mejores soluciones. En (Van den Bergh, 2002) se discuten varias alternativas para detectar este estado:

- Análisis de clúster:** un porcentaje de las partículas se encuentran a una cierta distancia Euclídea de la mejor partícula global.
- Función objetivo sin progreso:** la función objetivo no sufre mejoras de manera significativa en p iteraciones del ciclo generacional.
- Máximo radio de la bandada:** la partícula con mayor distancia Euclídea con respecto a la mejor partícula global encontrada es menor que un umbral α .

En este artículo se utiliza c) para detectar la convergencia prematura del algoritmo y se define formalmente a continuación:

$$\rho(k) = \frac{\max_{1 \leq i \leq |\Omega|} \|X_i^{(k)} - X_{gbest}\|}{|\sigma_{m\acute{a}x} - \sigma_{m\acute{i}n}|} \quad (7)$$

donde Ω representa la nube de partículas, $\|\cdot\|$ denota la norma Euclideana sobre \mathbb{R}^n , mientras que $\sigma_{m\acute{i}n}$ y $\sigma_{m\acute{a}x}$ son los extremos sobre los cuales está definida cada dimensión de la partícula X_i (asumiendo el mismo dominio para todas la dimensiones). De esta manera queda normalizado el umbral de estancamiento para cada generación k .

Tratamiento del estado de estancamiento

Una vez que se detectan señales de convergencia prematura es necesario adoptar alguna medida para escapar de tal situación, por ejemplo:

- a) Mover la posición de la mejor partícula globalmente encontrada.
- b) Reagrupar las partículas de la bandada (aplicando operadores genéticos con el objetivo de diversificar la población, re-inicializando la bandada, etc.).

Aunque ambas estrategias han reportado resultados satisfactorios en la solución de problemas de optimización global presentan algunos inconvenientes. En el primer caso no es trivial encontrar una mejor partícula, e incluso una vez encontrada, la población se encuentra pobremente diversificada. En el segundo caso la diversificación aumenta las probabilidades de escapar del óptimo local pero si no se reagrupa adecuadamente la bandada, las partículas podrían converger hacia la misma solución o alejarse indiscriminadamente de las zonas prometedoras encontradas. Para atenuar estos inconvenientes en este trabajo se asume una perspectiva híbrida que consiste en diversificar la población tratando de mover la posición de la mejor partícula global.

La Búsqueda en Vecindades Variables (*Variable Neighborhood Search*, VNS) es una simple y efectiva metaheurística para problemas combinatorios y de optimización global (Hansen, 2001) que consiste en el cambio sistemático de la vecindad en el proceso de búsqueda. Inspirado en esta idea, se presenta un procedimiento llamado Muestreo Aleatorio en Vecindades Variables (*Random Sampling in Variable Neighborhoods*, RSVN) que tiene como objetivo dispersar la bandada cuando se detecta el estado de estancamiento. La idea principal de este procedimiento es reestructurar la nube de partículas a partir de la selección de muestras aleatorias uniformemente distribuidas en m vecindades generadas alrededor del punto n -dimensional X_{gbest} . Las ecuaciones (8), (9) y (10) formalizan matemáticamente la manera de generar el conjunto de muestras en cada vecindad.

$$\beta_{j,d}^- = \begin{cases} \tilde{\beta}_{j,d}^-, \tilde{\beta}_{j,d}^- \geq \sigma_{m\acute{i}n} \\ \sigma_{m\acute{i}n}, \tilde{\beta}_{j,d}^- < \sigma_{m\acute{i}n} \end{cases} \quad \text{and} \quad \tilde{\beta}_{j,d}^- = X_{gbest}_d - \xi_j |\sigma_{m\acute{a}x} - \sigma_{m\acute{i}n}| \quad (8)$$

$$\beta_{j,d}^+ = \begin{cases} \tilde{\beta}_{j,d}^+, \tilde{\beta}_{j,d}^+ \leq \sigma_{m\acute{a}x} \\ \sigma_{m\acute{a}x}, \tilde{\beta}_{j,d}^+ > \sigma_{m\acute{a}x} \end{cases} \quad \text{and} \quad \tilde{\beta}_{j,d}^+ = X_{gbest}_d + \xi_j |\sigma_{m\acute{a}x} - \sigma_{m\acute{i}n}| \quad (9)$$

$$X_t \in \Psi_j \mid X_{t,d} \sim U(\beta_{j,d}^-, \beta_{j,d}^+), \quad t = 1, \dots, |\Psi_j|; \quad j = 1, \dots, m; \quad d = 1, \dots, n \quad (10)$$

donde ξ_j es un valor fraccional en el intervalo (0, 1] que denota la proporción de la vecindad j con respecto al tamaño del espacio de búsqueda, d indexa la dimensión de la partícula, mientras que Ψ_j representa el j -ésimo conjunto de muestras uniformemente distribuidas en el dominio que define el intervalo $[\beta_{j,d}^-, \beta_{j,d}^+]$.

Después de recolectadas las muestras, se lleva a cabo un proceso de selección de las partículas que conformarán la nueva bandada como se muestra a continuación:

$$\Omega = (\Phi_1 \subseteq \Psi_1) \cup (\Phi_2 \subseteq \Psi_2) \cup \dots \cup (\Phi_m \subseteq \Psi_m) = \bigcup_{j=1}^m (\Phi_j \subseteq \Psi_j) \quad (11)$$

Sujeto a:

$$i) \quad |\widehat{\Omega}| = |\bigcup_{j=1}^m (\Phi_j \subseteq \Psi_j)| \quad (12)$$

$$ii) \quad \|X_i - X_{gbest}\| > \alpha |\sigma_{m\acute{a}x} - \sigma_{m\acute{i}n}|, \forall X_i \in \Omega \quad (13)$$

donde $\widehat{\Omega}$ representa la bandada de partículas inicial, Φ_j es un sub-conjunto suficientemente bueno de partículas respecto al conjunto de muestras Ψ_j , mientras que α corresponde al umbral permitido al máximo radio de la bandada. En este procedimiento cada partícula $X_i \in \Omega$ es una candidata para sustituir a la mejor partícula global X_{gbest} , lo que complementa el proceso de dispersión de la bandada.

Resultados y Discusión

La Tabla 1 describe nueve funciones de pruebas tomadas de (Suganthan, 2005) útiles para comparar el desempeño del algoritmo propuesto con varios enfoques reportados en la literatura. Estas funciones se caracterizan por tener varios óptimos locales, especialmente cuando aumenta la complejidad de la función, o sea, la dimensionalidad del espacio de búsqueda. Los primeros siete problemas son escalables e incluyen funciones unimodales, multimodales y ruidosas, mientras que los últimos dos problemas (definidos en \mathbb{R}^2) son fuertemente multimodales.

Tabla 1. Funciones de pruebas utilizadas en este trabajo. La última columna (F_{min}) corresponde con el mínimo global.

ID	Función	Intervalo	F_{min}
F ₁	$\sum_{i=1}^n x_i^2$	[-100,100]	0.0000000
F ₂	$\sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	[-5.12,5.12]	0.0000000
F ₃	$\left(\frac{1}{4000}\right) \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[-600,600]	0.0000000
F ₄	$\sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	[-2.048,2.048]	0.0000000
F ₅	$\sum_{i=1}^n i x_i^4 + rand[0,1]$	[-1.28,1.28]	0.0000000
F ₆	$-\sum_{i=1}^n (x_i \sin(\sqrt{ x_i }))$	[-500,500]	-420.968n
F ₇	$20 + e - 20e^{-0.2 \left(\left(\frac{1}{n}\right) \sum_{i=1}^n x_i^2\right)^{\frac{1}{2}}} - e^{(1/n) \sum_{i=1}^n \cos(2\pi x_i)}$	[-32.0,32.0]	0.0000000
F ₈	$\sum_{i=1}^5 \cos((i+1)x_1 + i) \sum_{i=1}^5 \cos((i+1)x_2 + i)$	[-10.0,10.0]	-186.7309
F ₉	$(x_2 + x_1^2 - 11)^2 + (x_1 + x_2^2 - 7)^2 + x_1$	[-5.0,5.0]	-3.783960

Comportamiento del algoritmo PSO-RSVN en espacios de búsqueda complejos

En esta sección se compara la efectividad y eficiencia del algoritmo PSO-RSVN frente a cinco enfoques evaluados en (Plant, 2008; Idoumghar, 2011): *Quadratic Interpolation based PSO* (QIPSO), *Gaussian Mutation based PSO* (GMPSO), *Attraction-Repulsion based PSO* (ATREPSO), *Hybrid PSO and Simulated Annealing* (HPSO-SA), y *Particle Swarm Optimization* (PSO) estándar.

En cada simulación se utilizan 30 partículas, 300000 evaluaciones de la función objetivo en un espacio de búsqueda de dimensionalidad 20. Además, se utilizan dos vecindades con factor de vecindad $\xi_1 = 0.05$ y $\xi_2 = 0.1$ respectivamente y una tolerancia al máximo radio de la bandada $\alpha = 1.0E-5$. La Tabla 2 resume la media y la desviación estándar de cada algoritmo para 30 simulaciones independientes.

Tabla 2. Comparación en el desempeño de PSO-RSVN frente a otros algoritmos tomados de (Plant, 2008; Idoumghar, 2011).

ID	PSO	QIPSO	ATREPSO	GMPSO	HPSO-SA	PSO-RSVN
F ₁	1.1677E-45	0.00000000	4.0002E-17	7.2635E-17	5.3656E-32	0.00000000
	5.2223E-46	0.00000000	0.00024600	6.1888E-17	2.9849E-31	0.00000000
F ₂	22.3391580	11.9468880	19.4259790	20.0791850	0.00000000	0.00000000
	15.9320420	9.16152600	14.3490460	13.7002020	0.00000000	0.00000000
F ₃	0.03164600	0.01158000	0.02515800	0.02446200	3.3225E-20	0.00000000
	0.02532200	0.01285000	0.02814000	0.03930400	2.6841E-20	0.00000000
F ₄	22.1917250	8.93901100	19.4908200	14.1595470	0.22704810	2.6354E-16
	1.6155E+04	3.10635900	3.9643E+04	4.3354E+04	0.24399780	1.4309E-15
F ₅	8.68160200	0.45110900	8.04661700	7.16067500	0.00201990	4.3721E-06
	9.00153400	0.32862300	8.86238500	7.66580200	0.00065030	3.9298E-06
F ₆	-6178.5598	-6355.5866	-6183.6776	-6047.6708	-8379.6600	-8375.7098
	4.8933E+02	477.532584	469.611104	482.926738	2.2042E-19	21.6237825
F ₇	3.4839E-18	2.4618E-24	0.01849300	1.4749E-18	7.4354E-16	4.4408E-16
	8.3595E-19	0.01442500	0.01474700	1.1537E-08	1.0938E-15	0.00000000
F ₈	-186.73094	-186.73094	-186.73094	-186.73094	-186.73094	-186.73090
	1.4241E-05	0.00000000	1.4241E-05	1.5258E-05	8.6674E-14	0.00000000
F ₉	-3.3314880	-3.7839610	-3.7514580	-3.4602330	-3.7839600	-3.7839600
	1.24329000	0.19039400	0.17446000	0.45782000	3.1600E-15	0.00000000

Analizando los resultados mostrados en la Tabla 2 es posible observar que para las funciones F₁, F₂, F₃, F₈ y F₉ PSO-RSVN encuentra el óptimo global de manera satisfactoria mientras que para F₄ y F₅ proporciona estimaciones aceptables. Para la función F₆, HPSO-SA localiza buenas aproximaciones, sin embargo, en un segundo lugar PSO-RSVN encuentra las mejores soluciones con respecto al resto de los algoritmos. Finalmente, para F₇ es QIPSO el método que mejor resultado reporta seguido por PSO-RSVN, lo que reafirma el desempeño del algoritmo propuesto. La Figura 1 ilustra el comportamiento del máximo radio de la bandada $\rho(k)$ en la optimización de la función F₄ para los algoritmos PSO-RSVN y PSO con coeficiente de restricción; la variabilidad que introduce el mecanismo de dispersión propuesto garantiza la exploración de nuevas áreas del espacio de búsqueda, incrementando la posibilidad de escapar de soluciones sub-óptimas.

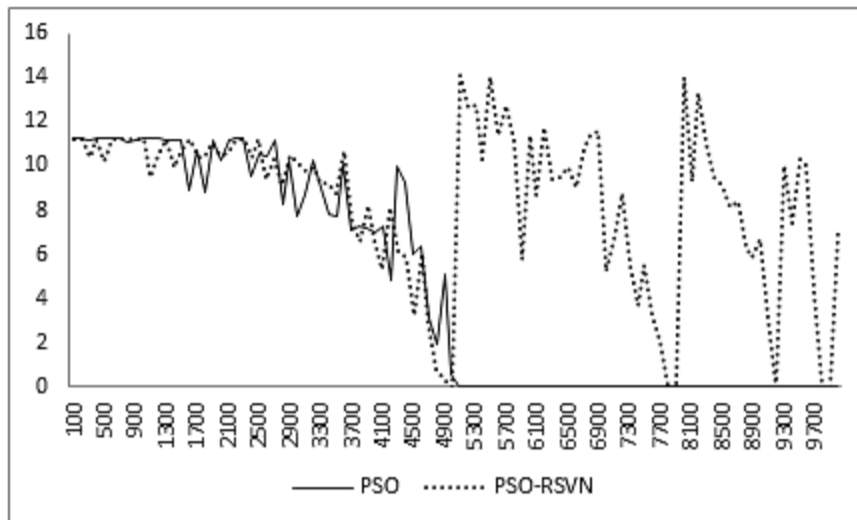


Figura 1. Comportamiento del máximo radio de la bandada a lo largo del ciclo evolutivo para PSO y PSO-RSVN.

Aunque PSO-RSVN, de manera general, ofrece resultados superiores a los enfoques indagados para las funciones de pruebas descritas en la Tabla 1, en un trabajo futuro incorporaremos un análisis estadístico más riguroso para explorar diferencias significativas entre los enfoques examinados y estudiaremos el rendimiento del algoritmo discutido para otras funciones de pruebas.

Conclusiones

En este trabajo se propone una variante modificada del algoritmo PSO con coeficiente de restricción, que es capaz de detectar y tratar el estado de estancamiento que se produce como consecuencia de la convergencia prematura del algoritmo en problemas de optimización complejos. Esta modificación detecta el estado de estancamiento comparando el máximo radio de la bandada con un umbral pre-establecido y posteriormente dispersa la misma en varias vecindades realizando un muestreo aleatorio en cada una de ellas, haciendo énfasis en la vecindad de la mejor partícula encontrada hasta el momento. Adicionalmente fue evaluado el desempeño del algoritmo en espacios de búsqueda complejos comparándolo con otras variantes de PSO mediante nueve funciones de prueba reconocidas en la literatura, ofreciendo resultados superiores en la mayoría de los casos. En un trabajo futuro se propone incorporar pruebas estadísticas para estudiar diferencias significativas entre los enfoques examinados.

Referencias

- CLERC, M.; KENNEDY, J. *The Particle Swarm - Explosion, Stability, and Convergence in a Multidimensional complex space*, IEEE Transactions on Evolutionary Computation, 2002, 6 (1): 58-73.
- EBERHART, R.; KENNEDY, J. *A New Optimizer Using Particle Swarm Theory*. En: Proceedings of the Sixth International Symposium on Micro machine and Human Science. Japón: IEEE Service Center, 1995, p. 39-43.
- EBERHART, R.; SHI, Y. *Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization*. En: Proceedings of the 2000 IEEE Congress on Evolutionary Computation. Indianapolis, IN: IEEE Computer Society, 2000, p. 84-88.
- EVERS, G.; GHALIA, M. B.: *Regrouping Particle Swarm Optimization: A new Global Optimization Algorithm with Improved Performance Consistency Across Benchmarks*. En: IEEE International Conference on Systems, Man and Cybernetics, San Antonio, TX: IEEE Computer Society, 2009, p. 3901-3908.

- HANSEN, P.; MLADENOVIĆ, N. *Variable Neighborhood Search: Principles and Applications*. European Journal of Operations Research, 2001, 130(3): 449-467.
- IDOUMGHAR, L.; MELKEMI, M. *et al. Hybrid PSO-SA Type Algorithms for Multimodal Function Optimization and Reducing Energy Consumption in Embedded Systems*. Applied Computational Intelligence and Soft Computing, 2011, 2011, 12 páginas.
- KENNEDY, J.; EBERHART, R. *Particle Swarm Optimization*. En: Proceedings of the 1995 IEEE International Conference on Neural Networks. Australia: IEEE Computer Society, 1995, p. 1942-1948.
- KENNEDY, J.; RUSSELL, C.E. *Swarm Intelligence*. Morgan Kaufmann Publishers, 2001.
- PARSOPOULOS, K. E.; VRAHATIS, M.N. *UPSO-A Unified Particle Swarm Optimization Scheme*. Lecture Notes in Computer Science, Springer, 2004, p. 868-873.
- SUGANTHAN, P.N.; HANSEN, N. *et al. Problem Definition and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization*. Technical Report 2005005, 2005.
- PANT, M.; THANGARAJ, R.; ABRAHAM, A. *Particle Swarm Based Metaheuristic for Function Optimization and Engineering Applications*. En: Proceedings of the 7th Computer Information Systems and Industrial Management Applications (CISIM '08). Ostrava: IEEE Computer Society, 2008, p. 84-90.
- SHI, Y.; EBERHART, R. *A Modified Particle Swarm Optimizer*. En: Proceedings of the IEEE Congress on Evolutionary Computation (CEC '98). IEEE Computer Society, 1998, p. 69-73.
- VAN DEN BERGH, F. *An Analysis of Particle Swarm Optimizers*. PhD Thesis, University of Pretoria, Pretoria, 2002.