



PSO con operador de clearing adaptativo para fomentar la diversidad

PSO with adaptive clearing operator to promote diversity

Eduardo Samaniego¹, Kenya Guerrero¹, José Luis Tubay¹, Yaima Trujillo², Amilkar Puris³, Pavel Novoa³

¹Universidad Técnica Estatal de Quevedo, Facultad de Ciencias de la Ingeniería. Carrera de Ingeniería en Telemática. Campus "Ingeniero Manuel Haz Álvarez", Av. Quito, km 1.5 vía a Santo Domingo de los Tsáchilas. EC.120301. Quevedo, Ecuador.

esamaniego@uteq.edu.ec; kguerrero@uteq.edu.ec; jtubay@uteq.edu.ec

²Universidad Técnica Estatal de Quevedo, Unidad de Admisión y Nivelación. Campus "Ingeniero Manuel Haz Álvarez", Av. Quito, km 1.5 vía a Santo Domingo de los Tsáchilas. EC.120301. Quevedo, Ecuador. yaimatr@gmail.com

³Universidad Técnica Estatal de Quevedo, Facultad de Ciencias de la Ingeniería. Carrera de Ingeniería en Sistemas. Campus "Ingeniero Manuel Haz Álvarez", Av. Quito, km 1.5 vía a Santo Domingo de los Tsáchilas. EC.120301. Quevedo, Ecuador.

apuris@uteq.edu.ec; pnovoa@uteq.edu.ec

Rec.: 6.04.2016. Acept.: 8.07.2016.

Publicado el 30 de diciembre de 2016

Resumen

En el presente trabajo se aplica una variante adaptativa del operador de Clearing para fomentar la diversidad de la Meta-heurística Optimización basada en Enjambre de Partículas (PSO). El objetivo consiste en determinar si la utilización de dicho operador mejora los niveles de exploración del algoritmo PSO incrementando la diversidad de la población y por consiguiente obteniendo mejores soluciones. Para ello, se propone una forma adaptativa para controlar la distancia permitida entre soluciones y de esta forma perturbar soluciones que se encuentren muy cercanas a otra con mejor calidad. Esta idea permite estudiar diferentes alternativas de como perturbar las soluciones seleccionadas, presentando en el trabajo dos enfoques diferentes: basado en la posición actual y en la mejor posición histórica. Además, se detalla un estudio para determinar cuando se aplicará el operador de Clearing para obtener la máxima contribución al proceso de búsqueda aplicado por PSO. La propuesta fue probada en un conjunto de 20 funciones multimodales, donde los resultados mostraron que el operador de clearing mejora la exploración de la Meta-heurística PSO, obteniéndose resultados significativamente superiores en cuanto a calidad que el modelo PSO original.

Palabras clave: exploración, diversidad, población metaheurística.

Abstract

In this paper an adaptive Clearing operator is applied to the Particle Swarm Optimization Meta-heuristic (PSO) in order to encourage diversity in the population. The goal is to determine whether the use of the Clearing operator improves levels of exploration PSO algorithm increasing diversity of the population without adding too much computational cost to the search process. Thus, an adaptive way is proposed to control the allowable distance between solutions and disturb the solutions that are very close to another with better quality. This idea allows to study different alternatives as to disturb the selected solutions, presenting at work two different approaches based on the current position and in the best historical position. In addition a study is detailed to determine when to be applied Clearing operator for maximum contribution to the search process implemented by PSO. The proposal was tested in a set of 20 multi-modal functions, where the results showed that the operator clearing improves exploration of meta-heuristic PSO, yielding significantly superior results in quality that Original PSO model.

Key words: exploration, diversity, population metaheuristic.

Introducción

El área de optimización actualmente recibe mucho interés pues aparece en gran cantidad de problemas científicos e industriales. A veces abordar estos problemas de manera tradicional es algo impracticable, dada la complejidad computacional del proceso de optimización. Los procedimientos meta heurísticos constituyen una posible variante de soluciones en estos casos, brindándonos soluciones aproximadas con una calidad relativamente buena.

Las Meta-heurísticas poblacionales y específicamente las Bioinspiradas, son de las más estudiadas y empleadas en la actualidad, pues en general permiten una mejor exploración del espacio de búsqueda, al tener más de un agente involucrado en el proceso de búsqueda heurística; entre las conocidas están los Algoritmos Genéticos (AG) (Goldberg, 1989), la Optimización basada en Colonias de Hormigas (ACO) (Dorigo *et al.*, 2004) y la Optimización basada Enjambre de Partículas (PSO) (Kennedy, 2011).

Sin embargo, un problema presente en la búsqueda realizada por estas Meta-heurísticas, es la convergencia prematura, donde la población puede converger a zona muy específica del espacio de soluciones, disminuyendo la capacidad de poder encontrar soluciones en zonas no exploradas.

Algunas han sido las propuestas presentadas en la comunidad científica para fomentar la diversidad en las Meta-heurísticas poblacionales; “DeJong (1975) introduce el concepto de Niching, donde un conjunto de subpoblaciones evolucionan de manera independiente y a través de un mecanismo de intercambio, se comparte información entre ellas”. Otra propuesta, es el operador de Crowding (Lozano *et al.*, 2008), donde el individuo más similar de la población es eliminado cuando otro nuevo es introducido a la subpoblación. Por otra parte “Schaffer *et al.*, (1991) proponen una técnica de cruzamiento restringido (Restricted Mating), donde dos individuos se cruzan si son parcialmente diferentes”. Otra propuesta para fomentar la diversidad en la población, inspirado en el ecosistema natural, donde cada individuo esta forzado a compartir su calidad con sus vecinos es el operador de Sharing (Goldberg *et al.*, 1987). Por otra parte, “Deepti *et al.*, (2012) realizaron una revisión detallada de las propuestas más novedosas para fomentar la diversidad”.

En este trabajo se estudia una variante de la estrategia Clearing (Petrowski, 1996) para fomentar la diversidad en la Meta-heurística PSO. Se propone una forma adaptativa para calcular la distancia entre partículas y se presenta un estudio experimental con un conjunto de 20 funciones multimodales de prueba.

Materiales y métodos

Optimización basada en Bandas de Pájaros

Esta meta heurística se inspira en el comportamiento social de varios grupos de organismos como son; las bandadas de pájaros, bancos de peces y enjambre de insectos. El principio fundamental de PSO es la inteligencia colectiva, donde el conocimiento no se encuentra en cada individuo sino en la interacción del colectivo. En este esquema, la interacción se desarrolla de manera directa, donde cada organismo se orienta a partir de su propia experiencia y la del resto del grupo.

Desde su surgimiento, PSO ha mostrado ser eficiente para resolver problemas de optimización continua de un único objetivo, aunque también se han presentado variantes para problemas discretos. Una explicación más precisa del modelo se describe a continuación:

* Se crea una población inicial, donde el valor de cada dimensión del individuo se mueve aleatoriamente entre un rango mínimo y un rango máximo dados.

* Cada individuo representa una posición en el espacio de búsqueda, y se desplaza a una determinada velocidad que se calcula a partir de su velocidad en la iteración anterior k , su posición actual (x^k), la mejor posición por la que ha pasado ($pBest$) y la posición de la partícula mejor posicionada en la población ($gBest$). La ecuación 1 describe la forma de calcular la velocidad, donde w representa la inercia de la partícula para controlar el nivel de explotación o exploración (Eberhart *et al.*, 1998) y φ_1, φ_2 representan la dependencia cognitiva o social respectivamente, de la nueva velocidad (Beielstein *et al.*, 2002).

$$v^{k+1} = w * v^k + \varphi_1 * U(0,1) * (pBest - x^k) + \varphi_2 * U(0,1) * (gBest - x^k) \quad (1)$$

* La nueva posición de cada individuo en la iteración $k+1$ (x^k) viene determinada por la posición en que está situado actualmente x^k , más el desplazamiento que supone el movimiento de cada partícula con velocidad v^{k+1} . Dicha posición se calcula como:

$$x^{k+1} = x^k + v^{k+1} \quad (2)$$

* Una vez se tiene la nueva posición de cada individuo en el espacio de búsqueda, se calcula el nuevo valor de fitness para esa posición, y se actualizan los valores de ($pBest$) y ($gBest$).

* Se producen una serie de iteraciones en las que se modifican la velocidad y la posición de cada individuo hasta llegar a la condición de terminación del algoritmo. Se puede definir como: alcanzar el número máximo de iteraciones dadas al inicio de la ejecución del algoritmo, llegar a un número de iteraciones que no produzca una mejora significativa en la mejor solución encontrada, entre otras.

La propuesta original de PSO ha tenido numerosas

modificaciones, entre las más recientes se tienen:

* “Ostadmohammadi *et al.* (2013) presentan un operador de colisión para mantener delimitados los territorios en la población y evita las agrupaciones de las partículas o convergencia prematura, propiciando la exploración del espacio de búsqueda”.

* “Kang *et al.* (2016) proponen una mutación adaptativa de la inercia de movimiento de cada partícula utilizando el valor de la diversidad promedio del enjambre en cada iteración, de esta forma en la nueva generación las partículas son impulsadas a otras zonas del espacio de búsqueda”.

* “Wanga *et al.* (2016) proponen una estrategia llamada torneo topológico dinámico, la cual difiere de la propuesta original en que cada partícula es guiada por varias soluciones mejores, elegidas en la población actual. Este mecanismo favorece no solo a la mejor solución sino también a otras partículas con buena posición”.

Operación de limpieza o Clearing

El método clearing propuesto en Petrowski (1996) utiliza el concepto de nicho (subregiones del espacio de búsqueda), y plantea que dos individuos van a pertenecer a un mismo nicho si la distancia entre ellos es menor que una cuota establecida. El autor recomienda el uso de la distancias hamiltoniana para problemas binarios y la distancia euclidiana para problemas continuos. La idea fundamental de este operador es, limitar un poco la competencia dentro de las subregiones o nichos perturbando algunos de los individuos que pertenecen al mismo nicho. Esta estrategia elitista mejoró considerablemente el rendimiento de los algoritmos genéticos en el estudio de funciones multimodales. La Figura 1 presenta el esquema general del procedimiento de clearing propuesto por Petrowski. Los parámetros Sigma, y Kappa representan el radio de clearing y la capacidad de cada nicho respectivamente, la función SortFitness (P) ordena la población de partículas por la calidad.

```
function Clearing(Sigma, Kappa)
begin
  SortFitness(P)
  for i = 0 to n - 1
    if Fitness(P[i]) > 0
      nbWinners := 1
      for j = i + 1 to n - 1
        if Fitness(P[j]) > 0 and
           Distance(P[i], P[j]) < Sigma
          if nbWinners < Kappa
            nbWinners := nbWinners + 1
          else
            Fitness(P[j]) := 0.0
          endif
        endif
      endfor
    endif
  endfor
end
```

Figura 1. Esquema general del operador de Clearing

Técnicas y procedimiento

Para analizar los resultados, se emplearon un conjunto de técnicas estadísticas no paramétricas de contrastación de hipótesis horizontales: Friedman (Friedman, 1937) para determinar si existen diferencias significativas entre el grupo de resultados, el test de Holm (1979) para comparaciones múltiples entre más de dos resultados y prueba de signos de Wilcoxon (1945) para comparaciones en pares.

Las variables independientes responden a las diferentes alternativas de aplicación del operador de clearing, a la Meta-heurística PSO, así como la configuración de los parámetros involucrados para determinar la repercusión de estas variables en la calidad de solución.

Resultados y discusión

En esta sección se aplica una modificación del operador de clearing a la Meta-heurística PSO como alternativa para aumentar la diversidad. Un análisis similar fue realizado por Martínez *et al.* (2012), donde las partículas cercanas eran remplazadas por otra creada de manera aleatoria. Esa propuesta no conservaba información de las partículas remplazadas, por ende se perdía conocimiento. En esta ocasión se presenta otra forma de aplicar el operador de clearing que consiste en:

- * Ordenar todas las partículas de la población inicial en función de su calidad.
- * Comparar cada partícula del enjambre con sus sucesores, en forma secuencial, y las partículas que se encuentren a una distancia menor que la cuota establecida (Sigma) son perturbados (cambiando su posición).

La obtención de nuevas coordenadas de las partículas perturbadas se realizan de dos formas:

- * Reseteando de manera aleatoria la posición actual de la partícula y manteniendo su mejor posición.
- * Reseteando de manera aleatoria la mejor posición por la que ha transitado la partícula y manteniendo su posición actual.

El operador de Clearing puede aplicarse indistintamente en diferentes momentos de la ejecución del algoritmo para prevenir o atacar la convergencia prematura de la población. En este trabajo se utiliza solo para prevenir la convergencia prematura (aplicarlo cada cierto número de iteraciones). Para determinar la separación entre las partículas se utilizó la distancia euclidiana definida como:

$$D_{P1,P2} = \sqrt{\sum_{i=1}^n (x_{p1,i} - x_{p2,i})^2} \quad (3)$$

El valor de Sigma utilizado en el operador de Clearing, se calcula en la iteración que se aplique, y se determina a partir del dominio de la función de estudio y el valor de la iteración actual del algoritmo. Se define como:

$$\text{Sigma} = \begin{cases} \frac{|min + max|}{4}, & \text{si } j < 15\%M \\ \frac{|min + max|}{8}, & \text{si } 15\%M \leq j < 30\%M \\ \frac{|min + max|}{16}, & \text{si } 30\%M \leq j < 60\%M \\ \frac{|min + max|}{50}, & \text{si } 60\%M \leq j < 80\%M \\ \frac{|min + max|}{100}, & \text{si } j \geq 80\%M \end{cases} \quad (4)$$

Donde M representa el valor máximo de iteraciones a ejecutarse y j el valor actual de la misma. El min y max, determinan los extremos del dominio de la función en estudio. En la ecuación se observa que el valor de sigma decrece a medida que se ejecuta el algoritmo PSO, de esta forma se puede controlar el nivel de exploración del algoritmo.

Se analizó la influencia de aplicar el operador de Clearing cada cierto número de iteraciones (1, 50, 100) del algoritmo PSO y se estudian las dos formas de perturbación de las partículas. Como escenario de prueba fue considerado la minimización de 20 funciones continuas multimodales de dimensión 10, presentadas en una sección especial organizada por la IEEE en el Congress on Evolutionary Computation (CEC 2005) (Suganthan et al., 2005). Se trabajó con una población de 40 partículas, valor fijo de inercia $w=0.80$ y $\phi_1=\phi_2=1.90$ para el cálculo de la velocidad y 10000 iteraciones del algoritmo.

La figura 2 muestra los resultados alcanzados en el proceso de minimización (ordenadas) de cada una de las 20 funciones de estudio (coordenadas). En este caso, se aplicó un reinicio aleatorio de la posición actual de la partícula k (x^k), si su distancia a otra partícula con mejor calidad (Ecuación 3),

no cumple con el umbral permitido (Ecuación 4). Se puede observar que los resultados más prometedores se alcanzan por la variante ClearingPSO (50), donde se aplica el operador cada 50 iteraciones del algoritmo PSO. El test de Friedman ejecutado con estos datos obtuvo un valor p de $4.067E-06$ donde evidencia claramente que existe diferencia significativa en el grupo (valor $p < 0.05$). Seguidamente los resultados del test de Holm (Cuadro 1) aseguran que el algoritmo de control ClearingPSO (50) (menor valor medio) obtiene resultados significativamente mejores que las otras dos variantes en estudio, porque el valor p de cada comparación es menor que su respectivo α/i que determina el valor utilizado para el cumplimiento de la hipótesis y depende del caso que se esté comparando (i) y del valor de confianza ($\alpha=0.05$) para este caso en particular. Por su parte Z representa el estadístico utilizado en la comparación entre el algoritmo de control (R_i) y el i-ésimo algoritmo (R_j).

Por otra parte, la Figura 3 muestra un estudio similar al determinado en la Figura 2, en este caso se realizó un reinicio aleatorio de la mejor posición por la que ha transitado la partícula k (*pbest*). En este caso, sigue teniendo mejor comportamiento la variante que aplica el Clearing cada 50 iteraciones del PSO.

Los resultados del test de Friedman muestran que existe diferencia significativa en el grupo con un valor $p = 5.805E-08$ y el test de Holm (Cuadro 2) determina que el algoritmo de control ClearingPSO (50) también obtiene resultados significativamente mejores que las otras variantes. A continuación se comparan los resultados de ClearingPSO (50) para cada alternativa de reinicio con la variante pura de PSO utilizando el test de Wilcoxon (Cuadro 3). Se puede apreciar la hipótesis de igualdad, es rechazada en cada caso de estudio ($p < 0.05$). Analizando los rangos positivos y negativos se puede detectar que aplicar el operador de Clearing cada 50 iteraciones en cada una de sus variantes de reinicio proporciona resultados significativamente mejores que los que se obtiene con la versión pura del algoritmo PSO.

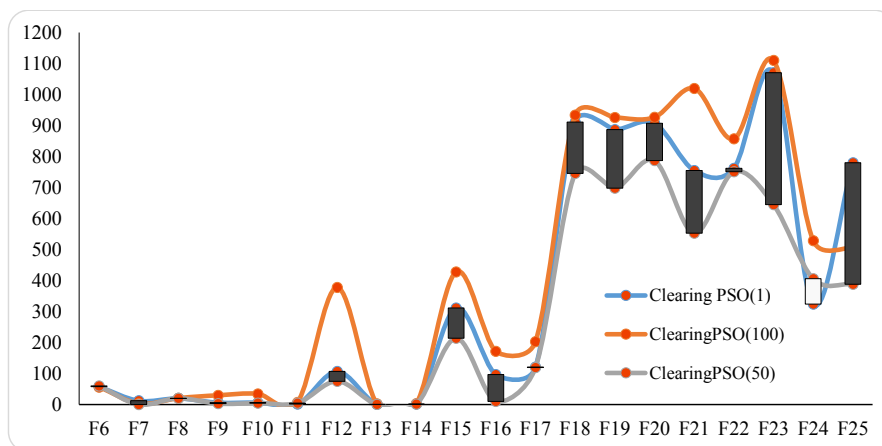


Figura 2. Operador de Clearing con reinicio aleatorio de la posición actual

Cuadro 1. Resultados del test de Holm para el primer caso de estudio

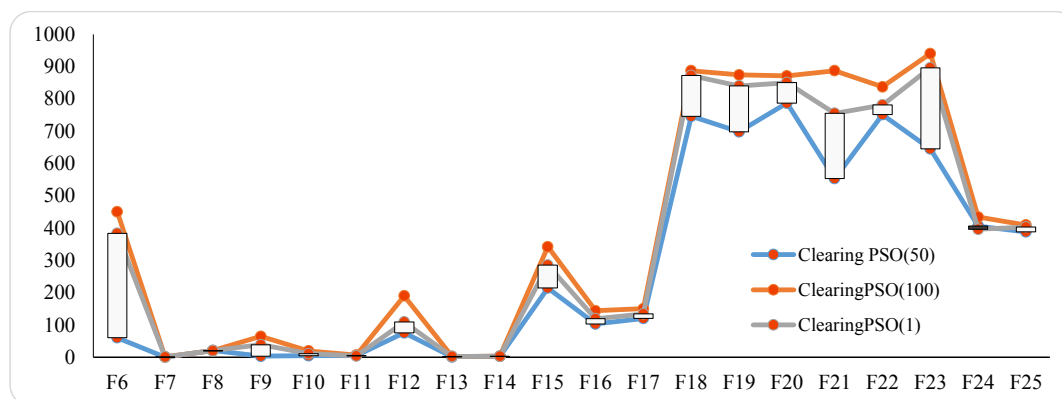
Algoritmo	$Z=(R_0-R_i)/0.5$	Valor p	α/i	Hipótesis
ClearingPSO(100)	4.98	6.34E-7	0.025	Rechazada
ClearingPSO(1)	2.37	0.017	0.050	Rechazada

Cuadro 2. Resultados del test de Holm para el segundo caso de estudio

Algoritmo	Z	Valor p	α/i	Hipótesis
ClearingPSO(100)	5.77	7.87E-9	0.025	Rechazada
ClearingPSO(1)	2.76	0.005	0.050	Rechazada

Cuadro 3. Resultados de la prueba de Wilcoxon para cada caso de estudio

Comparación	R ⁺	R ⁻	Valor p	Hipótesis
ClearingPSO(50) vs PSO	210.00	0	0.002	Rechazada
ClearingPSO(50) vs PSO	117.50	92.50	0.043	Rechazada

**Figura 3. Operador de Clearing con reinicio aleatorio de la mejor posición**

Conclusiones

Los mejores resultados se alcanzaron aplicando el Clearing cada 50 iteraciones de PSO independiente de la alternativa de reinicio. Además, se constató que mejora la exploración del espacio de búsqueda y por ende la calidad de las soluciones.

Bibliografía

Beielstein, T., Parsopoulos, KE., and Vrahatis, MN. (2002). Tuning PSO parameters through sensitivity analysis. University of Dortmund: Technical Report of the Collaborative Research.

- Deepti, G., and Shabina, G. (2012). An Overview of methods maintaining Diversity in Genetic Algorithms. *International Journal of Emerging Technology and Advanced Engineering*, 2:56-60.
- DeJong, KA. (1975). An analysis of the behavior of a class of genetic. Ann: Ph.D. dissertation, Univ. of Michigan.
- Dorigo, M., and Stützle, T. (2004). *Ant Colony Optimization*. MIT Press. Cambridge,
- Schaffer, L., and Eshelman, J. (1991). Preventing premature convergence in genetic algorithms by preventing incest. In Schaffer, JD. (Ed.). *Fourth International Conference on Genetic*. Morgan Kaufmann, San Mateo, CA.
- Eberhart, Y., and Stützle, T. (1998). A modified particle swarm optimizer. *World Congress on Computational Intelligence, the IEEE International Conference on Evolutionary Computation*. NJ, USA: Anchorage, AK or Piscataway.

- Friedman, M. (1937). Individual comparisons by ranking methods. *Journal of the American Statistical Association*, 32: 675–701.
- Goldberg, DE., and Richardson, J. (1987). Genetic algorithms with sharing for multimodal function optimization. In: Hillsdale, NJ: Lawrence E. (Eds), *Second International Conference on Genetic algorithms and their application*.
- Goldberg, DE. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Boston, MA, USA: Addison-Wesley Longman Publishing Co.
- Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2): 65–70.
- Kang, HT., and Liang, FH. (2016). Adaptive mutation particle swarm algorithm with dynamic nonlinear changed inertia weight. *Optik*, 127: 8036–8042.
- Kennedy, J. (2011). Particle swarm optimization. In *Encyclopedia of machine learning*, Springer US.
- Lozano, M., Herrera, F. and Cano, JR. (2008). Algorithms, Replacement strategies to preserve useful diversity in steady-state genetic. *Information Sciences*, 178(23): 4421-4433.
- Martínez, C., Puris, A., y Bello, R. (2012). Mecanismo de limpieza del espacio de soluciones para fomentar la diversidad de la meta heurística PSO. *Contribuciones a las Ciencias Sociales (CCCCSS)*: 2-23.
- Ostadmohammadi Arani, B., Mirzabeygi, P., and Shariat Panahi, M. (2013). An improved PSO algorithm with a territorial diversity-preserving scheme and enhanced exploration–exploitation balance. *Swarm and Evolutionary Computation*, 11: 1–15.
- Petrowski, A. (1996). A clearing procedure as a niching method for genetic algorithms. *Proceedings of IEEE Evolutionary Computation*. Nagoya, Japan.
- Wanga, L., Yang, B., and Orchard, J. (2016). Particle swarm optimization using dynamic tournament topology. *Applied Soft Computing*, 48:584–596.
- Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics*, 1: 80-83