SUPERNOVA CON SECUENCIAS DE BAJA DISCREPACIA

Eddy Mesa
Universidad Nacional de Colombia
A.A. 1027 Medellín, Colombia
ejmesad@unal.edu.co

Juan David Velásquez
Universidad Nacional de Colombia
A.A. 1027 Medellín, Colombia
jdvelasq@unal.edu.co

Gloria Patricia Jaramillo
Universidad Nacional de Colombia
A.A. 1027 Medellín, Colombia
jdvelasq@unal.edu.co

RESUMEN

Las secuencias de baja discrepancia son una buena alternativa para remplazar los números pseudo-aleatorios cuando los métodos depende de la generación computacional de números aleatorios como en la aproximación de integrales, simulación o optimización global. Supernova es una metaheurística que utiliza la distribución uniforme para hallar la primera población. En este artículo se presenta, prueba y analiza una nueva versión de supernova que utiliza LDS para iniciar el algoritmo. El método se implementó para dos secuencias diferentes, Halton y Sobol; y ambas versiones se probaron con dos funciones benchmark conocidas. El desempeño de la nueva versión fue mejor que la versión original en términos de las llamadas a la función objetivo y la calidad de mínimo encontrada.

PALABRAS CLAVES: Optimización no lineal, Metaheurísticas, números quasi-aleatorios

ABSTRACT

Low discrepancy sequences (LDS) are a good alternative to replace pseudo-random number when methods depend of computational random number generation like integral approximation, simulation or global optimization. Supernova is a metaheuristics that use uniform random distribution to find the initial population. In this paper we present, test and analyze a new version of supernova that use LDS to start the algorithm. The method was implemented with two different sequences, Halton and Sobol; and both versions were tested for two known benchmark functions. The performance of new version was better than original one in terms of calls to goal function and quality of the minimum.

KEY WORDS: Nonlinear optimization, Metaheuristics, quasirandom numbers.

1 Introducción

En la vida real los problemas de optimización son complejos y se asocian a problemas no polinomiales conocidos como NP-Hard. Para resolverlos, se han desarrollado metaheurísticas que son procesos heurísticos inspirados mayormente en fenómenos naturales que optimizan en si mismos como: mejor adaptación en la evolución que inspira las estrategias evolutivas, grano de mejor calidad en el temple de materiales de donde resulta el temple simulado, etc. Entre sus ventajas, las metaheurísticas, no necesitan la información analítica de las derivadas por ser métodos de búsqueda directa, y aunque no garantizan la optimalidad, proveen soluciones factibles en un tiempo razonable. Se han desarrollado muchas que obedecen a distintos tipos de fenómenos y se pueden agrupar como: evolutivas, de aprendizaje y de procesos físicos (Talbi, 2009; Yang, 2010; Zilinskas & Törn, 1989).

Supernova es una metaheurística inspirada en la culminación de las estrellas en supernovas y todavía está en fase de desarrollo. Aún así, muestra gran potencial para los problemas no lineales continuos con una topología descendente logrando mejor calidad de los mínimos en relación a otras metaheurísticas como evolución diferencial (DE del inglés), con una menor cantidad de llamadas a la función objetivo para algunas. Básicamente, supernova tiene tres pasos básicos: Inicia con una explosión de partículas, la interacción entre ellas(exploración) y un reinicio o re-explosión sobre el mejor punto encontrado(exploración). En estos pasos se combina una exploración de la región de búsqueda y una explotación del conocimiento encontrado (Mesa, 2010).

En la explosión inicial de Supernova, como en el inicio de muchas otras metaheurísticas, se utiliza un muestreo uniformemente aleatorio inicial. Este muestreo se hace con el fin de conocer la superficie en la mayor parte de sus secciones. A pesar de ser un muestreo uniforme, la distribución uniforme tiene altas discrepancias entre los puntos encontrados y deja espacios en blanco de los cuales no se tiene información, lo cual es un problema para la mayoría de metaheurísticas pues quedan espacios en los cuales no se logra explorar de manera apropiada. Se han desarrollado, desde casi el mismo inicio de los métodos heurísticos, estrategias para solucionar el problema de la distribución de las soluciones; algunas de ellas son: distribución uniforme estratificada (Brooks, 1959), piscina de individuos (Michalewicz & Schmidt, 2002), secuencias de baja discrepancia (Xuan Hoai, Quang Uy, McKay, & Minh Tuan, 2007), entre otras. Las secuencias de baja discrepancia (LDS del inglés), reemplazan el muestreo uniforme en algunos casos, como aproximación de integrales, y se ha aplicado exitosamente para hallar poblaciones iniciales de metaheurísticas como enjambre de partículas (PSO del inglés). Además de la baja discrepancia, que logra un mejor muestreo inicial, se tiene la ventaja de que no varían y son proporcionales (Gentle, 2003; Knuth, 1998).

En este momento, Supernova utiliza la función de distribución uniforme para hallar los puntos iniciales presentando los espacios en blanco de los cuales no se puede extraer información(Mesa, 2010). El problema se puede resolver por las diversas maneras ya diseñadas, pero se escogió las LDS por que, al ser determinísticas no presenta variación entre corridas. En este caso, supernova sería totalmente determinístico. En este artículo se describe la implementación y prueba de una versión de Supernova con inicio con secuencias de baja discrepancia.

En la Sección 2, materiales y métodos, se describen las secuencias de baja discrepancia se que utilizarán, el algoritmo de supernova, las implementaciones y el diseño de experimentos. En la Sección 3, se describen los resultados y se analizan en la Sección 4. Finalmente se concluye sobre en trabajo en la Sección 5.

2 Materiales y Métodos

2.1 Secuencias de baja discrepancia

Las LDS se conocen también como números quasi-aleatorios, y se caracterizan por tener pequeñas discrepancias. Mientras que la discrepancia mínima para una distribución aleatoria es cerca de $\log(\log N)^{\frac{1}{2}}$ que es mucho mas grande que la discrepancia de las secuencias, que es cercana a $\frac{\log(N)^s}{N}$ lo que logra una distribución mas pareja sobre el espacio de búsqueda y proporcional al número de puntos. Si bien no se garantiza la no existencia de espacios significativos, es menos probable que en una distribución pseudo-aleatoria (Gentle, 2003; Knuth, 1998).Además, las LDS no varían en cada iteración.

Existen varios tipos de distribuciones que cumplen con las características para ser LDS, las mas comunes son: Halton, Sobol, Faure y Van der Corput. En este caso se tomará la más antigua Halton que es una generalización de Van der Corput y Sobol que es la que muestra desempeño superior para PSO (Xuan Hoai et al., 2007). Estas secuencias no se programaron, se tomaron del paquete ya desarrollado en lenguaje R llamado fOptions (Wuertz & et al, 2012) en su más reciente versión. En este caso el programa trae por defecto las secuencias normalizadas es decir puntos entre 0 y 1 para un hipercubo.

2.2 Supernova, versión original.

Supernova, inspirada en las estrellas supernovas, utiliza para su funcionamiento un sistema de ecuaciones abreviado del fenómeno físico real. Particularmente, el impulso y la atracción gravitatoria de las partículas; la combinación de estos dos fuerzas, se utiliza como regla de búsqueda. El algoritmo se diseñó para optimizar problemas no lineales y sin restricciones (Mesa, 2010).

Supernova sigue la secuencia de la Figura 1. Para una solución inicial, se explota sobre el área de búsqueda un número K de partículas con una fuerza λ; es decir en una hiperesfera de radio λ se explotan K puntos uniformamente distribuidos. Estos puntos interactúan de acuerdo a la gravedad y el impulso así:

$$\mathbf{s}_k = \mathbf{x}_k + \mathbf{p}_k + \mathbf{g}_k \qquad (2)$$

donde $\mathbf{x}_k$ son cada una de las K partículas inicialmente explotadas a partir de la function uniforme, $\mathbf{p}_k = F m_k^{-1} \mathbf{d}_k$ es el impulso y $\mathbf{g}_k = \sum_{i=1|i\neq k}^{K} G \cdot m_k^{-1} \cdot m_i^{-1} \cdot \left[1 - r_{ik} \cdot \left(\sum_{j=1|j\neq i}^{K} r_{jk}\right)^{-1}\right] \cdot \mathbf{u}_{ik}$ la fuerza de gravedad asociada a la masa m y la distancia r análogamente con el modelo físico de las supernovas.

La masa es la característica por la cual se describe la fuerza que tiene cada uno de los elementos, es decir que brinda información sobre la superficie explorada. En este caso, se toma un escalamiento de la función objetivo; se escala para que el valor de cualquier función sea positivo y mayor que 0 que son las mismas características para la masa del fenómeno físico.

La distancia r, se toma como un promedio ponderado de la distancia euclidiana entre las partículas para evitar divisiones por 0 y problemas computacionales para conjuntos de partículas muy cercanas entre sí.

La interacción particular se repite hasta llegar a un número de iteraciones T o un equilibro que se nota como un criterio de parada; éste consiste en parar cuando para un porcentaje ε de iteraciones en las que no se halla un mejor mínimo. Una vez llega a este punto de equilibrio se reduce λ a una razón de α y sobre el mejor punto encontrado se vuelve a explotar el

```
Datos de entrada:
Función (f(x)), punto de inicio (x0) y límites (LB; UB)
Parámetros: K, T, R, α, λ, ε, F, G
Para r desde 1 hasta R
    s_k = Explosión(x0, LB, UB)
    t = 2
    Para t desde 2 hasta T o criterio de parada
        p_k ← x_k; x_k ← s_k
        n_k = Selección(x_k, n_k)
        s_k = Iinteracción(f(x), x_k, p_k)
        t = t + 1
    Fin del para
    t = 1; r = r + 1
Fin del para
```

algoritmo, reiniciando así el proceso de búsqueda. Estos reinicios se realizarán hasta llegar a un número de reinicios R.

Figura 1. Secuencia que sigue la metaheurísticas de supernova

2.3 Supernova implementada con LDS.

Para implementar el inicio con LDS es necesario hacer algunos ajustes de la versión original. Las LDS utilizadas forman un hipercubo normalizado, en el algoritmo original de supernova se hallaban los primeros puntos a partir de un punto que explotaba sobre una hiperesfera de radio λ para el caso del inicio con LDS, se tomará el hipercubo de lado 2λ y para los reinicios se centrará en el mejor punto encontrado. Las otras secciones del algoritmo permanecerán como están en el algoritmo original. Para la primera iteración, no se toma en cuenta el punto de inicio sino las K soluciones encontradas con la LDS correspondiente para poder garantizar la completa evaluación de la región. Por esta misma razón, se tomó el λ inicial como la semisuma de la distancia entre el borde superior menos el borde inferior.

2.4 Diseño de experimentos.

Se implementó el algoritmo en lenguaje R (R Development Core Team, 2011) en dos versiones H-snova que corresponde a la versión inicializada con la secuencia Halton y S-snova correspontiente a la versión inicializada con la secuencia Sobol.

Se probó el funcionamiento de la nueva versión de supernova en dos funciones benchmark conocidas como son la esfera f_1 (unimodal, separable)y ackley f_2 (multimodal, no separable) descritas en la Tabla 1 incialmente para dos dimensiones, se trasladaron las funciones de manera que no fueran simétricas y para comprobar la robustés para funciones no simétricas, además de observar el comportamiento del algoritmo para diferentes dimensiones. Luego de estas pruebas internas de la nueva versión, para la función esférica se comparó con el comportamiento de la versión original para 30 dimensiones.

Tabla 1. Funciones benchmark

f	Equation	Range	N	$f(\mathbf{x}_{opt})$
f_1	$=\sum_{i=1}^{N} x_i^2$	$[-500, 500]^N$	30	0.0
f_2	$=-20\exp\left(-0.2\sqrt{\frac{1}{N}\sum_{i=30}^{N} x_i^2}\right)-\exp\left(\frac{1}{N}\sum_{i=1}^{N}\cos(2\pi x_1)\right)+20+\mathrm{e}^1$	$[-32{,}32]^N$	30	0.0

3 Resultados

Inicialmente se parametrizó para la nueva versión, con estos nuevos parámetros se corrieron las diferentes pruebas. En esta nueva versión solo una corrida es necesaria dado que las LDS generan la misma malla inicial que era la parte aleatoria del algoritmo de búsqueda. Los parámetros encontrados se muestran en la Tabla 2.

Tabla 2. Parámetros utilizados para las funciones

Parámetro	K	T	R	F	G	λ_0	α	ε
f_1	50	100	50	4	4	500	1.5	0.15
f_2	50	100	50	4	4	32	1.2	0.15

Inicialmente se corrió el algoritmo para las dos funciones de prueba y se reporta el mínimo encontrado f^* para comparar la calidad del punto encontrado y las llamadas a la función objetivo (F.C.) para ver el esfuerzo necesario para llegar a estos resultados. También se calculó la distancia euclidiana del punto encontrado al mínimo real d^* para observar que tan cercano están los puntos encontrados del mínimo, los resultados se muestran en la Tabla 3. Para las dos versiones con LDS, la secuencia Halton (H-snova) y la secuencia Sobol (S-snova) se corrio el algoritmo para las funciones simétricas y trasladadas para mostrar si hay problemas con la simetría. Además las dos funciones propuestas fueron trasladadas linealmente para observar el comportamiento ante mínimos no simétricos y cerca del borde; para ello se modificó la función sumándole la distancia que se muestra en la Tabla 3 bajo las columnas Δx_1 y Δx_2.

Tabla 3. Resultados para dos dimensiones y traslado para las diferentes secuencias

	Δx_1	Δx_2	H-snova			S-snova		
			f^*	d^*	F.C.	f^*	d^*	F.C.
f_1	0	0	7.38×10^{-34}	2.72×10^{-17}	224	$\mathbf{3.50 \times 10^{-38}}$	$\mathbf{1.87 \times 10^{-19}}$	**214**
f_2	0	0	0	3.53×10^{-16}	214	**0**	$\mathbf{2.54 \times 10^{-16}}$	**224**
f_1	450	450	**0**	**0**	**224**	**0**	**0**	**224**
f_2	-20	-20	0	0	226	**0**	**0**	**214**

Las dimensiones de un problema de optimización son un grado de dificultad y aumento en el esfuerzo para los métodos de solución, en la Tabla 4 se reportan los datos para las dos versiones de supernova con LDS conservando el formato anterior, valor de la función objetivo, distancia del mínimo encontrado al mínimo real de la función de prueba y las llamadas a la función objetivo para diferentes dimenciones (10,20 y 30).

Tabla 4. Resultados para diferentes dimensiones de las funciones de prueba

Dimensiones	Función	H-snova			S-snova		
		f^*	d^*	F.C.	f^*	d^*	F.C.
10	f_1	6.16×10^{-30}	6.16×10^{-30}	214	$\mathbf{1.10 \times 10^{-46}}$	$\mathbf{1.05 \times 10^{-23}}$	**224**
	f_2	0	1.19×10^{-15}	226	**0**	**0**	**224**
20	f_1	$\mathbf{1.32 \times 10^{-29}}$	$\mathbf{1.32 \times 10^{-29}}$	**214**	2.04×10^{-29}	4.52×10^{-15}	224
	f_2	0	6.72×10^{-16}	226	**0**	**0**	**224**
30	f_1	$\mathbf{1.96 \times 10^{-29}}$	$\mathbf{1.96 \times 10^{-29}}$	**214**	7.70×10^{-28}	2.77×10^{-14}	224
	f_2	3.55×10^{-15}	3.98×10^{-14}	226	**0**	**0**	**224**

Finalmente se comparó el dempeño de la nueva versión con LDS y la versión original con la distribución uniforme en el inicio, en este caso se comparó para 10, 20 y 30 dimensiones en la Tabla 5 se reporta el mínimo encontrado, las llamadas a la función objetivo y en el caso snova se reporta la desviación estándar pues el mínimo encontrado es el promedio de 50 corridas independientes. Los parametros utilizados en relación a la version original solo difieren en los parámetros $\lambda = 600$, $F = 1$ y $G = 1$. Al final de la tabla como método ADE, se reportan los mejores resultados reportados en (Brest, Greiner, Boskovic, Mernik, & Zumer, 2006), en este caso solo están disponibles para 30 dimensiones.

Tabla 5. Comparación del desempeño entre las nuevas versiones y la versión original de supernova

Método		f_1			f_2		
		Dimensiones			Dimensiones		
		10	20	30	10	20	30
H-snova	f^*	$6.16 \text{x} 10^{-30}$	$\mathbf{1.32 \text{x} 10^{-29}}$	$\mathbf{1.96 \text{ x } 10^{-29}}$	0	0	$3.55 \text{ x } 10^{-15}$
	F.C.	214	**214**	**214**	226	226	226
S-snova	f^*	$\mathbf{1.10 \text{x} 10^{-46}}$	2.04×10^{-29}	$7.7 \text{ x } 10^{-28}$	**0**	**0**	**0**
	F.C.	**224**	224	224	**224**	**224**	**224**
snova	f^*	$2.29 \text{ x } 10^{-32}$	$6.0 \text{ x } 10^{-32}$	$6.3 \text{ x } 10^{-29}$	$9.95 \text{ x } 10^{-16}$	$2.70 \text{ x } 10^{-15}$	$4.7 \text{ x } 10^{-15}$
	σ	$4.65 \text{ x } 10^{-32}$	$1.03 \text{ x } 10^{-31}$	$1.0 \text{ x } 10^{-28}$	$1.63 \text{ x } 10^{-15}$	$2.36 \text{ x } 10^{-15}$	$2.3 \text{ x } 10^{-15}$
	F.C.	1000	1000	972	1000	1000	972
ADE	f^*	N.A.	N.A.	$1{,}1 \text{ x } 10^{-28}$	N.A.	N.A.	$7{,}7 \text{ x } 10^{-15}$
	σ	N.A.	N.A.	$1{,}0 \text{ x } 10^{-28}$	N.A.	N.A.	$1{,}4 \text{ x } 10^{-15}$
	F.C.	N.A.	N.A.	1500	N.A.	N.A.	1500

4 Discusión

En la sección anterior se reportan los resultados para las pruebas hechas para la nueva versión de Supernova. En la etapa de parametrización, se observó que no había mucha variación respecto a los parámetros hallados de le versión anterior (Mesa, 2010). Aunque en esta versión la incidencia de parámetros como λ fue más clara, mostrando una relación fuerte del parámetro con el tamaño del área de búsqueda.

Una vez se hicieron las pruebas para determinar los parámetros, se corrieron las pruebas para las dos funciones elegidas. Se Inició con dos dimensiones para observar el comportamiento mas de cerca, inicialmente se tuvo que implementar un nuevo indicador, la distancia entre el mínimo real y mínimo hallado, pues el 0 encontrado en la función de ackley f_2 no siempre correspondía al punto, sino una aproximación muy cercana. Con esta medida se tiene una idea mas clara de la calidad de respuesta encontrada, en especial para una función multimodal que puede tener mínimos de magnitud similar en puntos lejanos.

Respecto a los resultados se puede observar en la Tabla 3 que para mínimos no simétricos de las funciones de prueba elegidas se obtienen resultados similares y que la versión que utiliza la sequencia Sobol presenta resultados consistentemente mejores para las cuatro situaciones planteadas.

En la Tabla 4 se toca el problema dimensional, a mayores dimensiones se tiene una mayor cantidad de espacio a explorar por lo tanto es cada vez mas complejo encontrar mejores respuestas. Se espera en estos casos que los resultados encontrados por el algoritmo presenten una tendencia lineal ascendente con una pendiente cercana a 1 en el caso de la función esférica f_1 fue lo encontrado, pero en la función de Ackley f_2 permanece constante en las dimensiones lo cual corresponde a lo encontrado en el algoritmo original (Mesa, 2010). También es importante notar que la versión que utiliza Halton presenta mejores resultados a partir de 20 dimensiones para la función esférica f_1.

Finalmente, en la Tabla 4 se tiene la comparación con la versión original, en ambos casos la nueva versión presenta mejores resultados, y el hallazgo mas importante es la disminución en el número de iteraciones obteniendo resultados similares. Esto implica un menor tiempo de proceso, además que con la nueva versión no son necesarias las corridas, por ser determinístico.

5 Conclusiones

En este artículo se presenta una nueva versión del algoritmo Supernova inicializando con LDS, se probaron dos secuencias diferentes, Halton y Sobol y se variaron las funciones de prueba trasladandolas a asimétricas y se concentró en el comportamiento al variar las dimensiones. Los resultados numéricos muestran que la simetría de la función y el desplazamiento a los bordes del mínimo no afectan el proceso de búsqueda, que la versión con Sobol es superior para problemas con dimensiones hasta 20 y Halton presenta mejores resultados para una de las funciones de prueba para 20 y 30 dimensiones; que la variación de parámetros para los diferentes problemas fue casi nula. Y que para las funciones de prueba elegidas funciona mejor que la versión original. Estos resultados son preliminares y es necesario probar con más funciones de distinto tipo validando los resultados con otras heurísticas, además la nueva versión, al no tener el componente aleatorio, permite un mayor aprendizaje en el ajuste de parámetros que puede ser una ventaja importante para plantear formas de autoadaptación de parámetros. Es decir la nueva versión presentada muestra ventajas prometedoras para su implementación final.

Referencias

Brest, J., Greiner, S., Boskovic, B., Mernik, M., & Zumer, V. (2006). Self-Adapting Control Parameters in Differential Evolution: A Comparative Study on Numerical Benchmark Problems. *IEEE Transactions on Evolutionary Computation*, *10*(6), 646-657.

Brooks, S. H. (1959). A Comparison of Maximum-Seeking Methods. *Operations Research*, *7*(4), 430-457.

Gentle, J. E. *Random number generation and Monte Carlo methods*. (J. Chambers, W. Eddy, W. Härdle, S. Sheater, & L. Tierney, Eds.)*Statistics and Computing* (Vol. 4, p. xvi, 381). Springer. 2003.

Knuth, D. E. *The Art of Computer Programming: Seminumerical Algorithms*. *Massachusetts AddisionWesley* (Vol. 2). Addison-Wesley. 1998.

Mesa, E. *Supernova : un algoritmo novedoso de optimización global*. National University. 2010. (http://www.bdigital.unal.edu.co/2035/)

Michalewicz, Z., & Schmidt, M. Evolutionary Optimization. Kluwer Academic Press. 2002.

R Development Core Team. R: A Language and Environment for Statistical Computing. Vienna, Austria. 2011. (http://www.r-project.org/)

Talbi, E. G. *Metaheuristics: from design to implementation*. Wiley. 2009.

Wuertz, D., & et al. fOptions packages. 2012. (http://cran.r-project.org/web/packages/fOptions/index.html)

Xuan Hoai, N., Quang Uy, N., McKay, R. I., & Minh Tuan, P. (2007). Initializing PSO with randomized low-discrepancy sequences. *GECCO* (pp. 173-181).

Yang, X.-S. *Engineering Optimization An Introduction with Metaheuristics Applications*. *Engineering Optimization*. Hoboken: Wiley. 2010.

Zilinskas, A., & Törn, A. Global Optimization. In G. Goos & J. Hartmanis (Eds.), *Lecture Notes in Computer Science vol 350* (p. 255). Berlin: Springer. 1989.