INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

EVOLUTIONARY COMPUTING

PRÁCTICA 6: PARTICLE SWARM OPTIMIZATION

ALUMNO: ORTEGA VICTORIANO IVAN

PROF.: JORGE LUIS ROSAS TRIGUEROS

FECHA DE REALIZACIÓN DE LA PRÁCTICA: 17/04/2018

FECHA DE ENTREGA DEL REPORTE: 24/04/2018

MARCO TEÓRICO.

La Optimización por Enjambre de Partículas (Particle Swarm Optimization, PSO) son algoritmos metaheurísticos inspirados en la naturaleza y basados en las poblaciones, originalmente acreditados a Eberhart, Kennedy y Shi. Estos algoritmos imitan el comportamiento social de acuerdo con una medida de calidad (función de aptitud). La improvisación es realizada a través de mover las partículas alrededor del espacio de búsqueda por medio de un conjunto de simples expresiones matemáticas que modelan algunas comunicaciones entre partículas. Estas expresiones matemáticas en su más simple y básica forma sugieren el movimiento de cada partícula hacia su mejor posición experimentada y la mejor posición del enjambre hasta el momento, junto con algunas perturbaciones aleatorias. Hay una abundancia de diferentes variaciones utilizando diferentes reglas de actualización. [1]

A pesar de ser generalmente conocido y utilizado como una técnica de optimización, PSO tiene sus raíces en el renderizado de imágenes y en tecnología para animación por computadora, donde Reeves [2] definió e implementó un sistema de partículas como un conjunto de individuos autónomos trabajando en conjunto para formar la apariencia de un objeto difuso, como una nube o una explosión. La idea fue en un inicio generar un conjunto de puntos y asignarles un vector de velocidad inicial a cada uno de ellos. Utilizando estos vectores de velocidad, cada partícula cambiaba su posición iterativamente mientras los vectores de velocidad eran ajustados por algunos factores aleatorios. [1]

Reynolds [3] agregó la noción de comunicación entre objetos al sistema de partículas de Reeves al introducir un algoritmo de parvada, en donde los individuos fueran capaces de seguir algunas reglas de parvada básicas, tales como tratar de igualar las velocidades de los demás. Dicho sistema permitió el

modelado de comportamiento de grupos más complejos de una forma más sencilla y apegada a la naturaleza. [1]

Kennedy y Eberhart [4] mientras intentaban “simular gráficamente la elegante pero impredecible coreografía de una parvada de aves” encontraron el potencial de las capacidades de optimización de una parvada de aves. Durante el curso de refinamiento y simplificación de su paradigma, discutieron que el comportamiento de una población de agentes que ellos sugirieron seguía los cinco principios de la inteligencia de enjambre articulada por Millonas [5]. El primero es el principio de proximidad: la población debe ser capaz de llevar a cabo simples comunicaciones en espacio y tiempo. El segundo es el principio de calidad: la población debe ser capaz de responder a factores de calidad en el entorno. El tercero es el principio de respuestas diversas: la población no debe comprometer sus actividades a lo largo de canales excesivamente estrechos. El cuarto es el principio de estabilidad: la población no debe cambiar su modo de comportamiento cada que el entorno cambie. El quinto es el principio de adaptabilidad: la población debe ser capaz de cambiar su modo de comportamiento cuando vale la pena el precio computacional. También mencionan que llaman comprometedoramente a sus partículas sin masa de miembros de población sin volumen para hacer que el uso de conceptos como la velocidad y la aceleración sean más sensibles. Por lo tanto, se acuñó el término optimización de enjambre de partículas. [1]

MATERIAL Y EQUIPO.

- Equipo de Cómputo
- Python3

DESARROLLO DE LA PRÁCTICA.

Con el código proporcionado, empezamos por modificar los valores de la influencia del mejor valor encontrado de cada partícula y la influencia del líder global, estableciendo la influencia individual a 0 y la influencia del líder global a 0.1. El comportamiento observado en las figuras 1 a la 3, es que todas las partículas tienden a seguir al líder y converger a un solo punto en menos iteraciones. Esto es hasta cierto punto bueno, sin embargo, cuando no se conoce con certeza el comportamiento de una función y esta tiene demasiados mínimos locales, si se inicia en una posición lejana al mínimo global puede que, si todas las partículas tienen una mayor influencia por el líder que por su propio mejor resultado, estas queden atoradas en un mínimo local, como se verá más adelante.

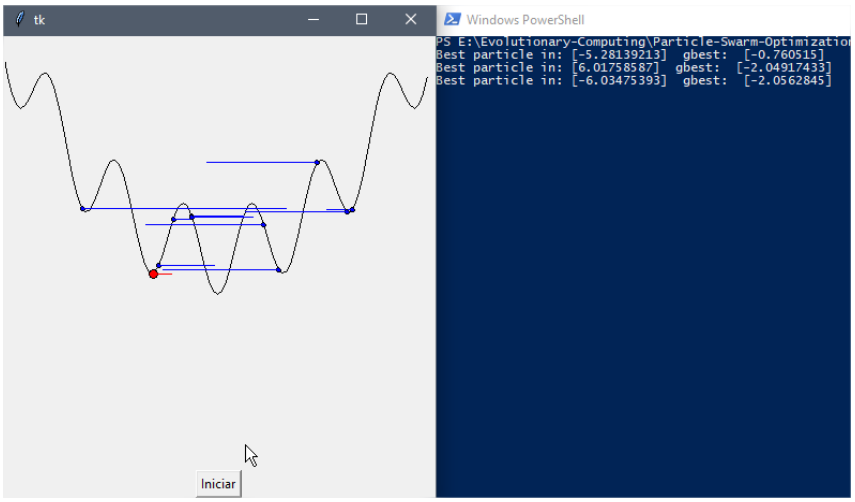


Figura 1. Convergencia de las partículas a un punto (posición del líder).

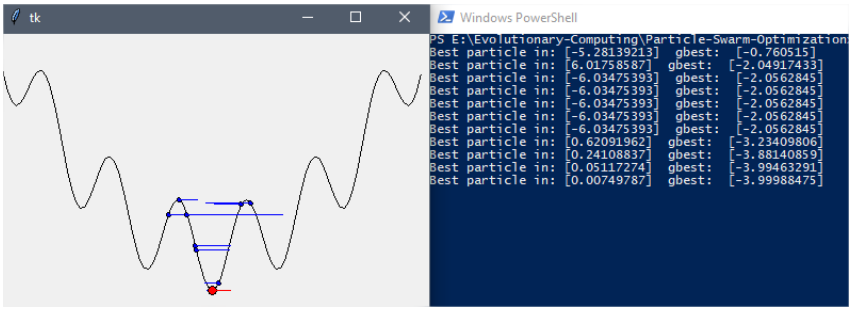


Figura 2. Convergencia de las partículas a un punto (posición del líder).

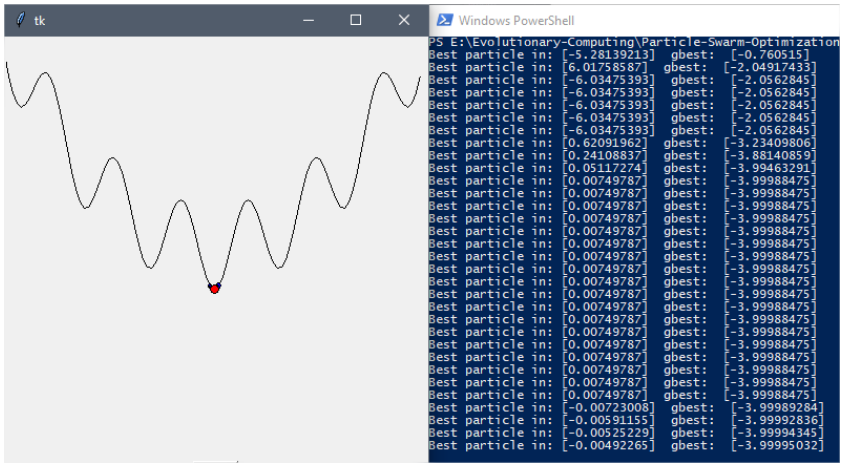


Figura 3. Convergencia de las partículas a un punto (posición del líder).

Como se pudo apreciar en las figuras 1 a 3, las partículas tienden a converger a un solo punto en un número menor de iteraciones.

El segundo experimento consistió en poner la influencia individual a un valor de 0.1 y la influencia del líder global a 0.5, pero en la inicialización de la posición de

las partículas, incrementarla en 35. El comportamiento se puede observar en la figura 4.

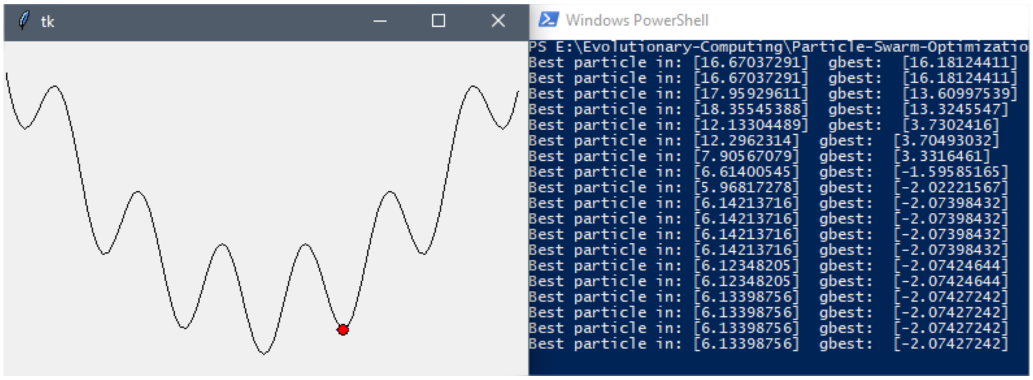


Figura 4. Las partículas convergen a un mínimo local, pero no al global.

Como se comentó anteriormente, dado que en este caso la posición de las partículas estaba lejana al mínimo global, estas convergieron a un mínimo local, en vez de converger al mínimo global.

El siguiente experimento en laboratorio fue poner valor de influencia individual mayor a la influencia del líder global. Además de modificar el valor del peso inercial a 0. El comportamiento, se puede observar en la figura 5.

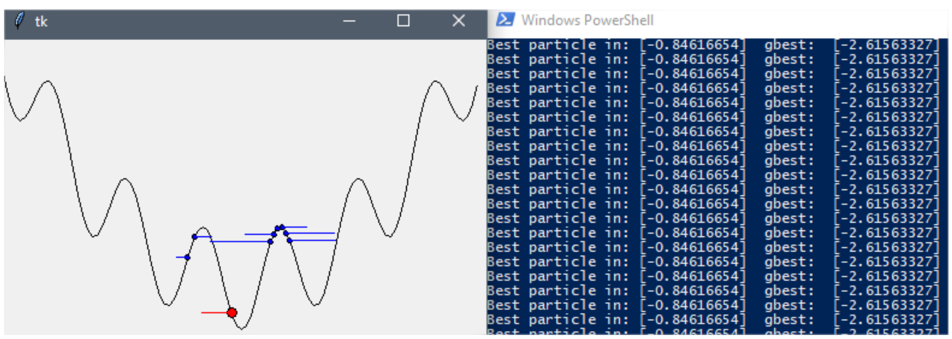
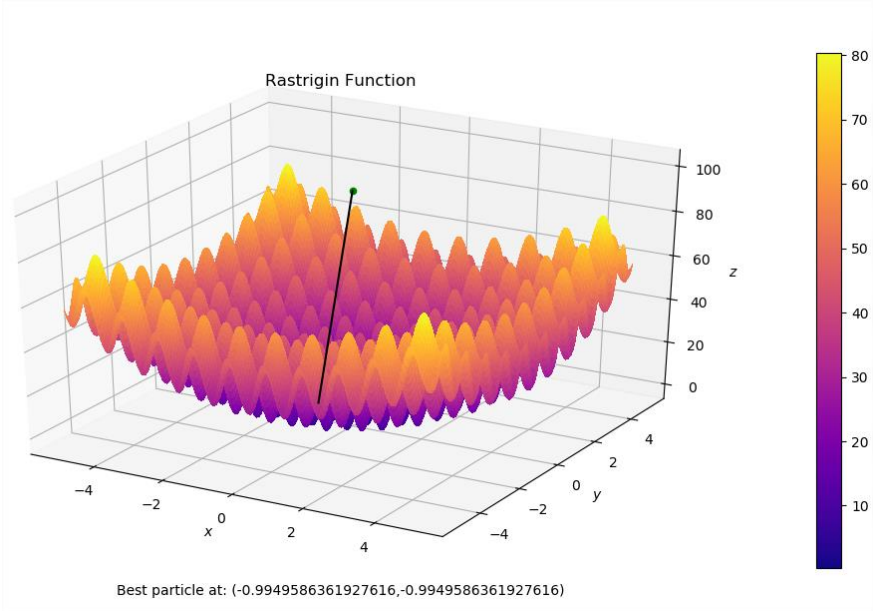


Figura 5. La búsqueda del mínimo queda estancada debido a la modificación del peso inercial.

En este último experimento se observaron 2 cosas, la primera de ellas y más sencilla es que las partículas tardan más en converger a un sol punto, la segunda se trata de la modificación del peso inercial. Al hacer cero el paso inercial impedimos que las partículas no puedan abandonar su porción del espacio de búsqueda, lo cuál sería equivalente a hacer una búsqueda local de un mínimo. En cambio, dándolo un mayor valor al peso, el ajuste de las velocidades sería tal, que permitiría a las partículas explorar regiones no vistas del espacio de búsqueda, aunque posiblemente se salieran del mismo. Por lo anterior, este factor es importante a la hora de analizar a que resultados queremos llegar.

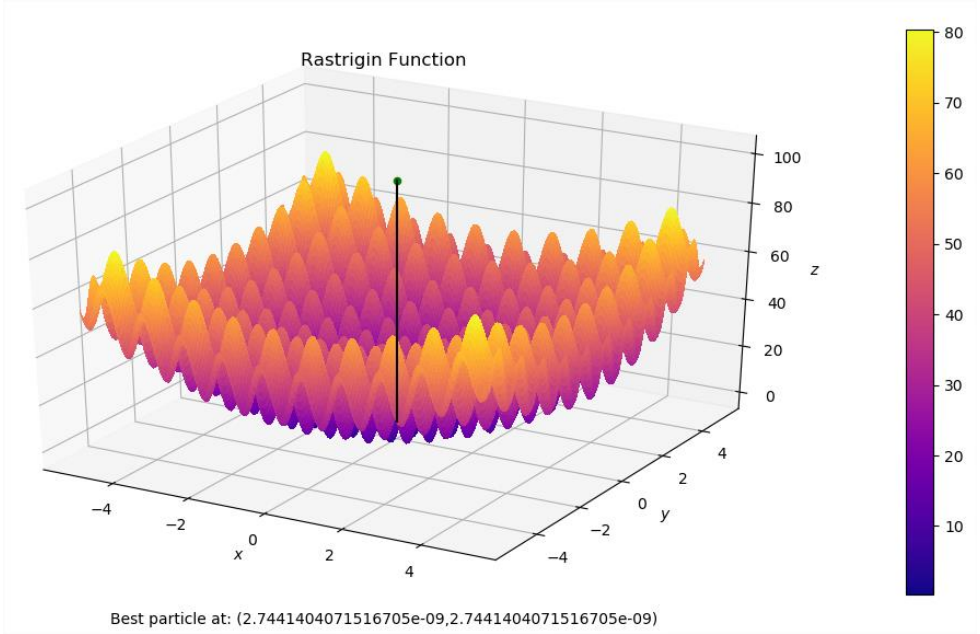
Finalmente, se dejó como tarea hacer lo mismo con las funciones de Rastrigin y Ackley, pero esta vez en su forma de 2 variables.

Empezando por Rastrigin, los resultados se muestran de la figura 6 a la 8, mientras que los de Ackley de la figura 9 a la 11. Las modificaciones al código, entre otras cosas, serán aclaradas en las conclusiones. Para ambos problemas, se utilizaron 1000 iteraciones para converger a un posible mínimo global.




```
Windows PowerShell
Iteration:  984
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  985
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  986
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  987
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  988
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  989
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  990
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  991
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  992
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  993
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  994
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  995
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  996
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  997
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  998
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  999
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
Iteration:  1000
Best particle in: [-9.94958636e-01  5.96282119e-10]  gbest:  0.9949590570932898
```

Figura 6. Primer resultado para la función de Rastrigin.

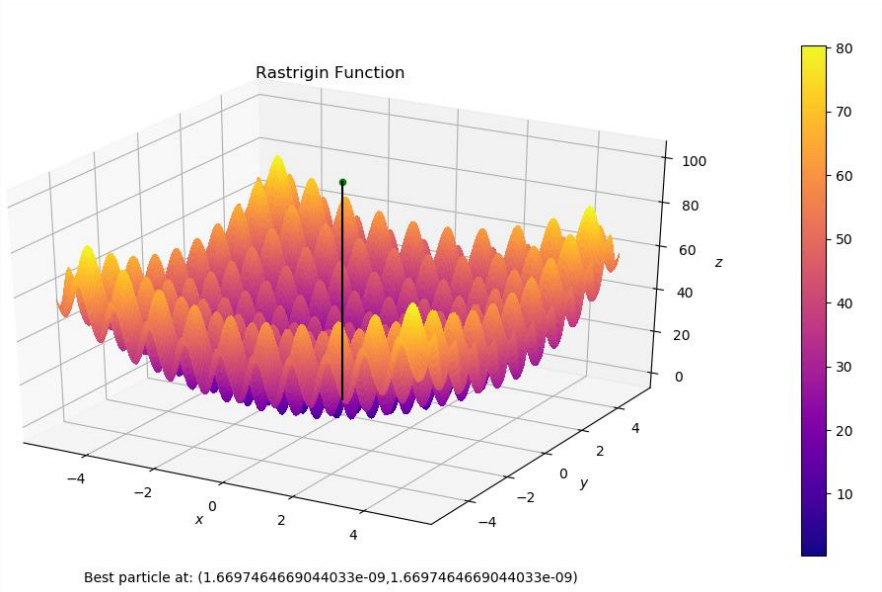


Best particle at: (2.7441404071516705e-09,2.7441404071516705e-09)

Windows PowerShell

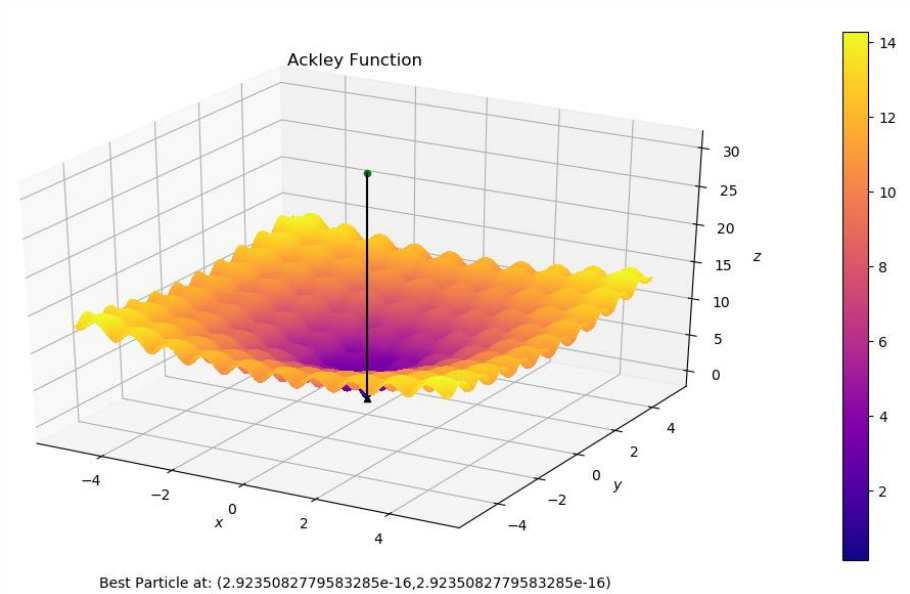
```
Iteration:  984
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  985
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  986
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  987
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  988
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  989
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  990
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  991
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  992
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  993
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  994
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  995
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  996
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  997
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  998
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  999
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
Iteration:  1000
Best particle in: [ 2.74414041e-09 -6.50211734e-10]  gbest:  0.0
```

Figura 7. Segundo resultado para la función de Rastrigin.



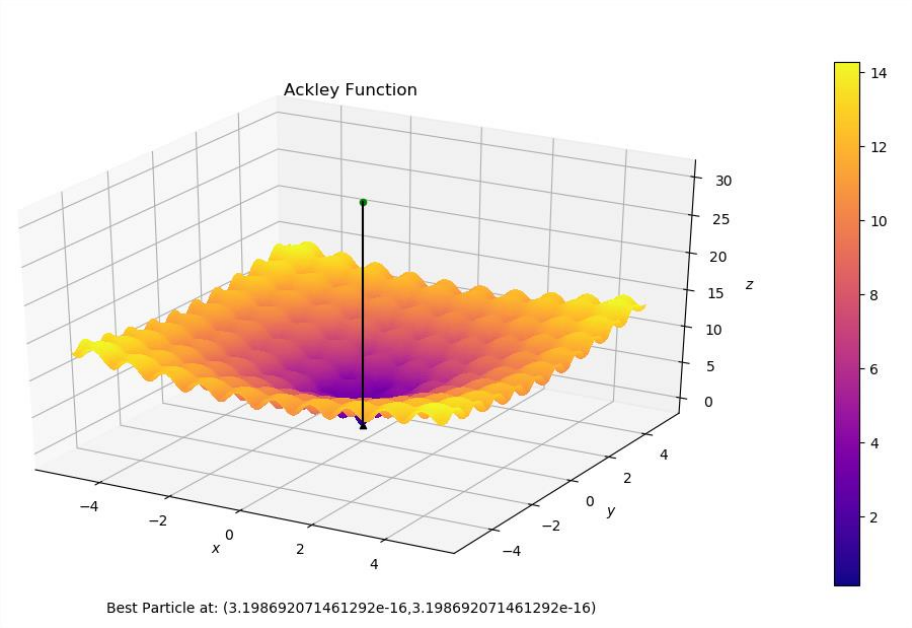

```
Windows PowerShell
Iteration:  984
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  985
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  986
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  987
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  988
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  989
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  990
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  991
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  992
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  993
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  994
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  995
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  996
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  997
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  998
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  999
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
Iteration:  1000
Best particle in: [1.66974647e-09 1.83756465e-09]  gbest:  0.0
```

Figura 8. Tercer resultado para la función de Rastrigin.



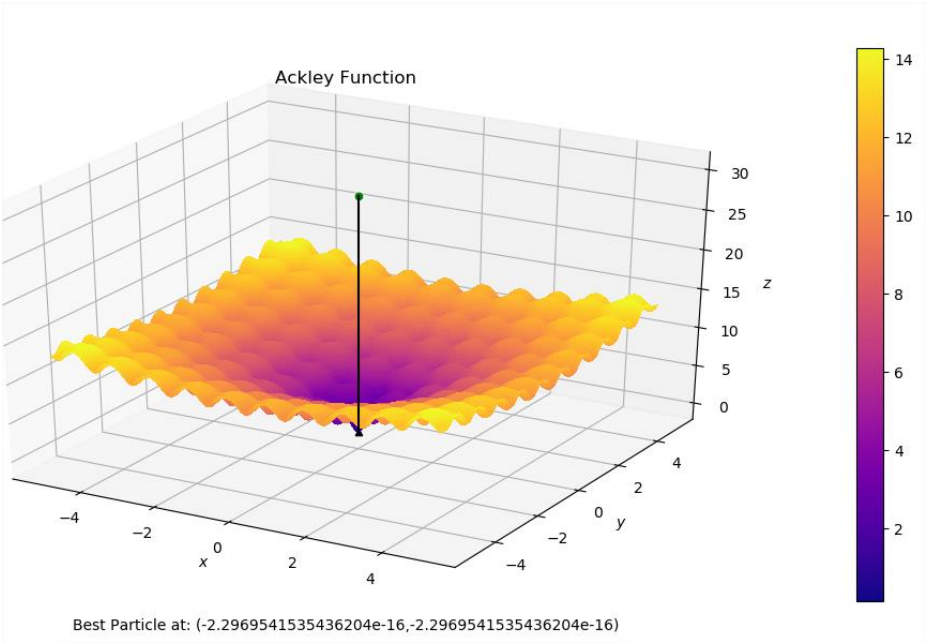

```
Windows PowerShell
Iteration:  984
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  985
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  986
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  987
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  988
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  989
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  990
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  991
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  992
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  993
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  994
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  995
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  996
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  997
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  998
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  999
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
Iteration:  1000
Best particle in: [ 2.92350828e-16 -9.39569514e-17]  gbest:  0.0
```

Figura 9. Primer resultado para la función de Ackley.




```
Windows PowerShell
Iteration:  984
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  985
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  986
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  987
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  988
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  989
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  990
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  991
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  992
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  993
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  994
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  995
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  996
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  997
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  998
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  999
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
Iteration:  1000
Best particle in: [3.19869207e-16 1.79206780e-16]  gbest:  0.0
```

Figura 10. Segundo resultado para la función de Ackley.




```
Windows PowerShell
Iteration:  984
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  985
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  986
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  987
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  988
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  989
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  990
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  991
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  992
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  993
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  994
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  995
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  996
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  997
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  998
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  999
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
Iteration:  1000
Best particle in: [-2.29695415e-16 -2.00102852e-16]  gbest:  0.0
```

Figura 11. Tercer resultado para la función de Ackley.

CONCLUSIONES.

Comparado a los algoritmos genéticos, considero que la optimización por enjambre de partículas tiene un mejor desempeño en la búsqueda de mínimos globales, más aún, considerando el caso de funciones en espacios de n-dimensiones.

Al investigar sobre la historia de PSO, no me imaginé que sus origines de dieron en el área de gráficos por computadora al implementar sistemas de partículas para el modelado de objetos difusos (de bajo brillo) como eran las nubes o explosiones y que más tarde, al agregar nuevos conceptos como la comunicación entre partículas, se permitiría modelar el comportamiento de sistemas más complejos.

Esta heurística, junto con los algoritmos genéticos, se me hacen muy intuitivos en cuanto a la comprensión de cómo funcionan y que es lo que hacen, ya que están basados en aspectos de la naturaleza que directa o indirectamente vemos día con día.

En cuanto a la implementación del programa para las funciones de Ackley y Rastrigin, modifiqué el número de dimensiones con las que trabajarían las partículas, además de cambiar los pesos inerciales. En el caso de Rastrigin, el peso inercial con el que obtuve mejores resultados fue de 0.9, ya que, para valores menores, la convergencia al mínimo global era más tardada en cuanto al número de iteraciones. En el caso de la función de Ackley, con 0.7 fue suficiente. En cuanto a los límites de búsqueda, se utilizaron los indicados en [6] y [7].

Referencias

[1] A. Kaveh, Advances in Metaheuristic Algorithms for Optimal Design of Structures, Nueva York: Springer, 2017.

[2] W. T. Reeves, «ACM Digital Library,» 25 Julio 1983. [En línea]. Available: https://dl.acm.org/citation.cfm?id=801167.

[3] C. W. Reynolds, «ACM Digital Library,» 4 Julio 1987. [En línea]. Available: https://dl.acm.org/citation.cfm?id=37406.

[4] J. Kennedy y R. Eberhart, «IEEE Xplore Digital Library,» 27 Noviembre 1995. [En línea]. Available: https://ieeexplore.ieee.org/document/488968/.

[5] M. M. Millonas, «Cornell University Library,» 11 Junio 1993. [En línea]. Available: https://arxiv.org/abs/adap-org/9306002.

[6] S. F. University, «Virtual Library of Simulation Experiments,» [En línea]. Available: https://www.sfu.ca/~ssurjano/ackley.html.

[7] S. F. University, «Virtual Library of Simulation Experiments,» [En línea]. Available: https://www.sfu.ca/~ssurjano/rastr.html.