

Artificial Bee Colony for optimizing mathematical functions

David Gómez

UPM

da.gomez@alumnos.upm.es

1. Resumen

El algoritmo Artificial Bee Colony (ABC) es un algoritmo metaheurístico que se inspira en la forma en que las abejas reales buscan y eligen las fuentes de alimento. Se ha utilizado con éxito para resolver una amplia variedad de problemas de optimización, incluyendo la optimización de funciones.

El algoritmo ABC es un algoritmo muy popular con una gran capacidad exploratoria, pero varios investigadores señalan que el algoritmo carece de un balance entre la exploración y la explotación. Para solventar este problema, se han realizado muchas variantes del algoritmo ABC.

Este trabajo parte del artículo presentado en la asignatura «*A labor division artificial bee colony algorithm based on behavioral development*» por Yingcong Wang et al. A continuación, se describen las herramientas y la estructura del trabajo, además se aportan algunas de las referencias clave utilizadas en este.

2. Herramientas y recursos

El entorno de trabajo principal es Unity3D, que además de proporcionar un motor para simulaciones 2D y 3D también puede ejecutar código en C#. Con Unity3D es fácil observar el comportamiento de los diferentes agentes durante la ejecución del algoritmo, de forma que se pueden contrastar los resultados con las simulaciones.

Para medir el rendimiento del tipo de algoritmos como el trabajado en el artículo se hace uso de un conjunto de funciones de «benchmark» sobre las que se ejecutan. Estas funciones están disponibles online y las hay multimodales, dinámicas...

Citation Geko es una herramienta de mucha utilidad para encontrar referencias y artículos relacionados. Facilita mucho la tarea de revisión bibliográfica.

3. Estructura del trabajo

El trabajo realizado cuenta con varias partes. Primero se realiza un estudio del estado del arte en el área de los algoritmos ABC y sus modificaciones. En concreto son interesantes las modificaciones del algoritmo que implementan varias estrategias de búsqueda (como en el

artículo de Wang et al.) más que aquellas que modifican la ecuación de búsqueda o realizan un híbrido entre distintos algoritmos de SI.

Más adelante se realiza un estudio comparando las dos modificaciones de este tipo más competentes encontradas en la literatura, centrándose en el tiempo de convergencia y la media y desviación típica de los resultados de un número de ejecuciones por determinar sobre las funciones de evaluación.

Los esfuerzos del trabajo consisten en detallar cual es el impacto de los distintos métodos de búsqueda utilizados sobre el algoritmo original. Además, si el trabajo avanza de la forma esperada, se realizarán modificaciones en alguno de los algoritmos que se sospeche que puedan mejorar el rendimiento de este. Este algoritmo modificado se comparará con los dos anteriores sobre las mismas métricas.

Referencias clave

- [1] Wenjian Luo, Xin Lin, Changhe Li, Shengxiang Yang, Yuhui Shi. *Benchmark Functions for CEC 2022 Competition on Seeking Multiple Optima in Dynamic Environments*. Neural and Evolutionary Computing (2022) <https://doi.org/10.48550/arXiv.2201.00523>
- [2] Yingcong Wang, Jiao, Junhui Liu, Renbin Xiao. *A labor division artificial bee colony algorithm based on behavioural development*. Information Sciences 606 (2022) pp.152-172.
- [3] D. Ustun, A. Toktas, U. Erkan y A. Akdagil. *Modified artificial bee colony algorithm with differential evolution to enhance precision and convergence performance*. Expert Systems with Applications 198 (2022) 116930. 10.1016/j.eswa.2022.116930
- [4] Xiaoyu Song, Ming Zhao, Shuangyun Xing. *A multi-strategy fusion artificial bee colony algorithm with small population*. Expert Systems with Applications 142 (2020) 112921.
- [5] Hui Wang et al. *Improving artificial Bee colony algorithm using a new neighborhood selection mechanism*. Information Sciences 527 (2020) pp. 227-240.