



TECNOLÓGICO
NACIONAL DE MÉXICO



Instituto Tecnológico Nacional de México, campus Culiacán

Ingeniería en Sistemas Computacionales

Trabajo Bonus Evolución Diferencial.

Protocolo de investigación para Tópicos de Inteligencia
Artificial

Integrantes del proyecto:

Payan Urquidez Rafael Alberto

Quiñonez Ramirez Nestor de Jesus

Docente: Mora Felix Zuriel Dathan

Repositorio:

<https://github.com/Norkat/Topicos-de-Inteligencia-Artificial>

Culiacán, Sinaloa a 5 Noviembre del 2025

Evolución Diferencial.....	3
Mecanismo.....	3
Mutación.....	3
Cruce.....	3
Selección.....	3
Aplicaciones.....	3
Ventajas y Desventajas.....	3
Bibliografía.....	4

Evolución Diferencial

La evolución diferencial (ED) es un algoritmo de optimización evolutiva propuesto por R. Storn y K. Price en 1997. Perteneció a la familia de los algoritmos evolutivos, y su objetivo principal es encontrar la solución óptima de un problema minimizando o maximizando una función objetivo. A diferencia de otros métodos basados en poblaciones, la ED se caracteriza por su simplicidad y su capacidad para manejar espacios de búsqueda continuos sin requerir información derivada (como gradientes).

Mecanismo

El algoritmo trabaja con una población de posibles soluciones, representadas por vectores de valores reales. En cada iteración (o generación), los individuos se modifican mediante tres operadores principales: mutación, cruce (recombinación) y selección.

Mutación

Se genera un vector mutante sumando a un individuo base la diferencia ponderada entre otros dos individuos aleatorios.

Cruce

Se combinan los valores del vector mutante con el individuo actual para formar un vector de prueba.

Selección

Se evalúa la función objetivo, y el mejor entre el individuo actual y el de prueba pasa a la siguiente generación.

Este proceso se repite hasta alcanzar un número máximo de iteraciones o una solución suficientemente buena.

Aplicaciones

La evolución diferencial se utiliza ampliamente en problemas de optimización continua, ajuste de parámetros, control automático, entrenamiento de redes neuronales, diseño de sistemas de ingeniería y procesamiento de señales, entre otros campos. Su capacidad para explorar espacios de búsqueda complejos la hace ideal para funciones no lineales, multimodales o con muchas variables.

Ventajas y Desventajas

Entre sus ventajas destacan su implementación sencilla, pocos parámetros de ajuste, buena convergencia en problemas no convexos y robustez ante ruidos o irregularidades en la función objetivo. Sin embargo, sus desventajas incluyen una alta dependencia de la correcta elección de parámetros (como el factor de escala y la tasa de cruce), posible lentitud en la convergencia en problemas de gran dimensión y la necesidad de múltiples evaluaciones de la función objetivo, lo que puede ser costoso computacionalmente.

Bibliografía

Storn, R. & Price, K. (1997). *Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces*. Journal of Global Optimization, 11(4), 341–359.

Das, S., & Suganthan, P. N. (2011). *Differential Evolution: A Survey of the State-of-the-Art*. IEEE Transactions on Evolutionary Computation, 15(1), 4–31.

Blum, C., & Roli, A. (2003). *Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison*. ACM Computing Surveys, 35(3), 268–308.