



TECNOLÓGICO
NACIONAL DE MÉXICO®



TECNOLOGICO NACIONAL DE MEXICO.
INSTITUTO TECNOLOGICO DE CULIACAN.

EVOLUCIÓN DIFERENCIAL DE TRABAJO ADICIONAL

TOPICOS DE INTELIGENCIA ARTIFICIAL 12-13.

DOCENTE: MORA FELIX ZURIEL DATHAN.

CLAVE DE LA MATERIA: GTD2102

NOMBRE DE LOS ALUMNOS:

- BORBÓN SÁNCHEZ EDGAR - 16171301
- MILLÁN LÓPEZ ANA KAREN - 20170985

UNIDAD: 3.

ACTIVIDAD: TAREA 2 BONUS I - EVOLUCIÓN DIFERENCIAL.

CULIACÁN, SINALOA.

Contenido

Definición.....	3
Mecanismo	3
Aplicaciones en el Mundo Real	4
Análisis del método	4
Ventajas	4
Desventajas	5
Conclusiones.....	5
Referencias	5
Repositorio	5

Definición

La **Evolución Diferencial (ED)** es un algoritmo de **optimización metaheurístico** que pertenece a la familia de la computación evolutiva. Su objetivo principal es encontrar la solución óptima (es decir, el valor máximo o mínimo) para un problema matemático complejo, especialmente en espacios de búsqueda continuos.

Fue desarrollado por *Rainer Storn* y *Kenneth Price* en la década de 1990 como una heurística simple y eficiente para la optimización global (**Storn & Price, 1997**). Su principal característica, y de donde toma su nombre, es el uso de las **diferencias entre vectores** de la población actual para **mutar** y crear nuevas soluciones, guiando así la búsqueda hacia mejores resultados (**Price, Storn, & Lampinen, 2005**).

Mecanismo

El algoritmo de **Evolución Diferencial** funciona manteniendo una población de soluciones candidatas y mejorándolas iterativamente a través de un ciclo de mutación, recombinación y selección. Este enfoque poblacional es característico de la **computación evolutiva** (**Coello, 2019**).

El proceso general sigue estos pasos clave:

1. **Inicialización:** Se crea una población inicial de **N** vectores (soluciones) de forma aleatoria dentro de los límites del problema.
2. **Mutación:** Por cada vector de la población, se crea un **vector donante**. Esto se logra seleccionando aleatoriamente otros tres vectores distintos de la población (x_a , x_b y x_c) y calculando un nuevo vector v usando la siguiente fórmula fundamental:

$$v = x_a + F * (x_b - x_c)$$

El factor de mutación (F) escala la diferencia vectorial, controlando la amplitud del paso de búsqueda (**Storn & Price, 1997**).

3. **Recombinación (Cruce):** Se crea un **vector de prueba** mezclando los componentes del **vector donante** con los del **vector objetivo** original. La **tasa de cruce (CR)** controla la probabilidad de heredar componentes del vector mutado, fomentando la diversidad en la población (**Price, Storn, & Lampinen, 2005**).
4. **Selección:** Se compara el **vector de prueba** con el **vector objetivo** original. Si el nuevo vector es mejor según la función de costo, reemplaza al original en la siguiente generación, en caso contrario, se descarta (**Storn & Price, 1997**).

5. **Iteración:** Los pasos 2 a 4 se repiten hasta que se cumple un criterio de parada, como alcanzar un número máximo de generaciones.

Aplicaciones en el Mundo Real

La Evolución Diferencial es muy eficaz en problemas de optimización no lineales y no diferenciables. Algunas de sus aplicaciones incluyen:

- **Ingeniería:** Diseño óptimo de estructuras, antenas y, de forma muy común, en la sintonización de controladores (como los **PID**) para procesos industriales, donde se busca la respuesta más estable y rápida (**Price, Storn, & Lampinen, 2005**).
- **Inteligencia Artificial:** Entrenamiento de redes neuronales, ajuste de hiperparámetros en **Machine Learning** y optimización de funciones en robótica (**Coello, 2019**).
- **Finanzas y Economía:** Modelado de mercados y optimización de carteras de inversión (**Price, Storn, & Lampinen, 2005**).
- **Ciencias:** Calibración de parámetros en modelos biológicos o químicos.
- **Logística:** Optimización de rutas y planificación de la producción.

Análisis del método

Este método tiene fortalezas y debilidades claras en comparación con otros algoritmos de optimización.

Ventajas

- **Potente y Robusto:** Es conocido por su capacidad para encontrar el óptimo global en problemas complejos (**Price, Storn, & Lampinen, 2005**).
- **Simple de Implementar:** El algoritmo base es conceptualmente sencillo y requiere pocos parámetros de control (**Storn & Price, 1997**).
- **Pocas Presuposiciones:** No requiere que la función a optimizar sea continua o diferenciable, lo que amplía su rango de aplicación (**Price, Storn, & Lampinen, 2005**).
- **Independencia de evaluaciones:** Las evaluaciones de las soluciones en cada generación son independientes y se pueden realizar en paralelo.

Desventajas

- **Sensible a los Parámetros:** El rendimiento del algoritmo depende mucho de la elección correcta de F y CR . Este es un desafío conocido, y mucha investigación actual se enfoca en crear variantes del algoritmo que autoajustan estos parámetros durante la ejecución para reducir la dependencia del usuario (*Price, Storn, & Lampinen, 2005*).
- **Convergencia Lenta:** En problemas de muy alta dimensionalidad, puede ser más lento que los métodos basados en gradiente (*Price, Storn, & Lampinen, 2005*).
- **No ideal para problemas discretos:** Está diseñado principalmente para espacios de búsqueda continuos, aunque se han propuesto adaptaciones (*Price, Storn, & Lampinen, 2005*).

Conclusiones

Basado en la investigación, la **Evolución Diferencial** es una herramienta de optimización **notablemente poderosa y versátil**. Su principal fortaleza radica en un equilibrio casi perfecto entre **simplicidad y eficacia**. A diferencia de otros algoritmos, la **ED** explora el espacio de soluciones usando una idea elegante: las diferencias vectoriales de la propia población. Esto demuestra una profunda comprensión del equilibrio entre **exploración** (buscar en nuevas áreas) y **explotación** (refinar soluciones buenas), un concepto clave en la computación evolutiva. Su éxito en tantos dominios demuestra que es un pilar fundamental de la optimización global.

Referencias

- Coello, C. A. (2019). *Computación Evolutiva*. Academia Mexicana de Computación, A.C.
- Price, K. V., Storn, R. M., & Lampinen, J. A. (2005). *Differential Evolution*. Springer.
- Storn, R., & Price, K. (1997). Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *ResearchGate*, 15.

Repositorio

<https://github.com/AnaKMLopez/ITCAKML-Topicos-IA/tree/main>