

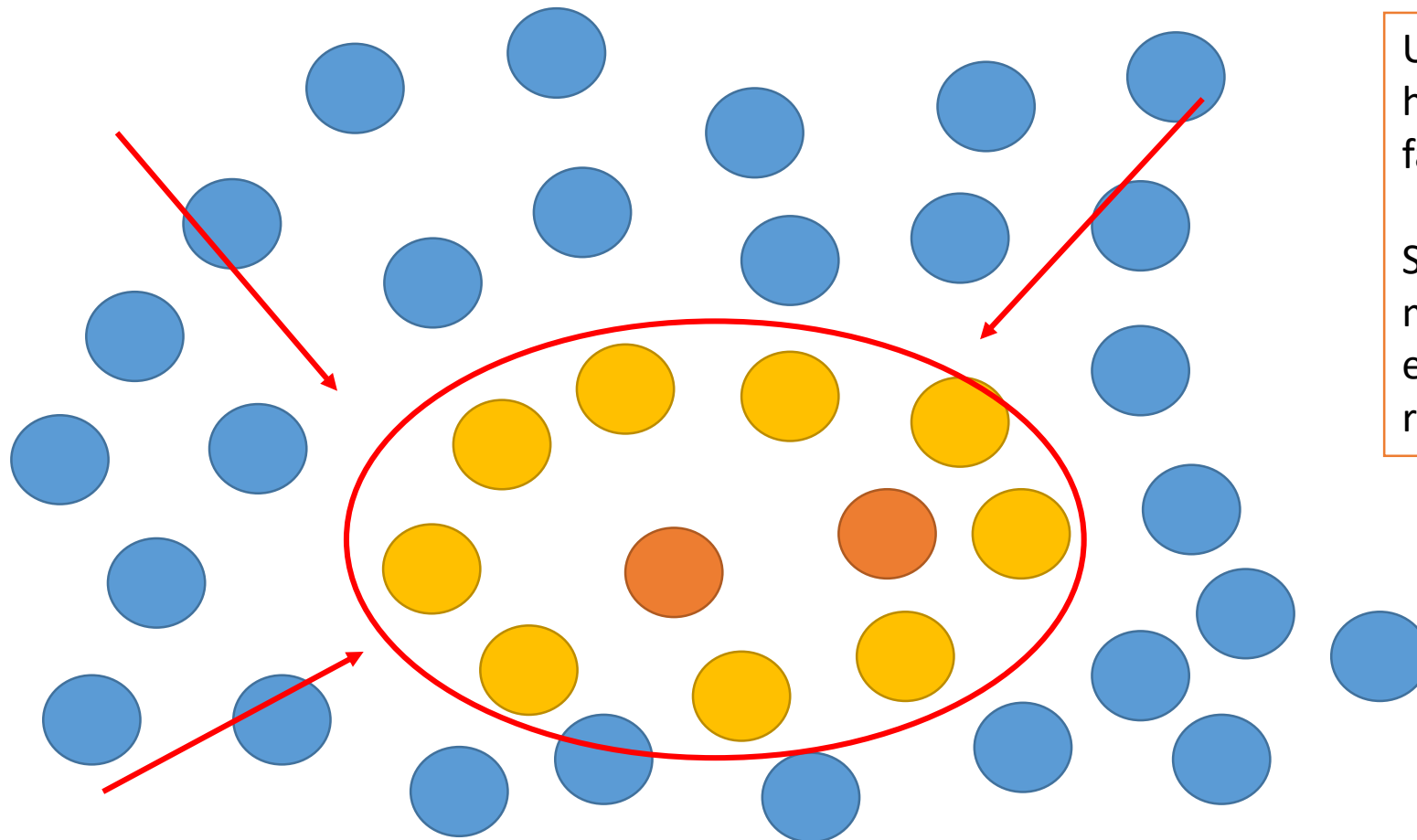
Algoritmos de optimización metaheurísticos

Algoritmos de optimización metaheurísticos

- No se requiere representar explícitamente el grafo de estados-acciones.
- Requiere describir el **espacio de soluciones** y definir operadores que permitan explorar este espacio.
- **Solución candidata**: cada elemento del espacio de soluciones.
- **Soluciones factibles**: soluciones candidatas que satisfacen las restricciones del problema.

Espacio de soluciones

Conjunto donde existen todas las posibles soluciones a un problema



Una buena heurística guiará el algoritmo hacia las regiones de interés (soluciones factibles).

Se desea un algoritmo que busque las mejores soluciones posibles. Por ejemplo, encontrar la ruta más corta o la ruta más rápida de una ciudad a otra.

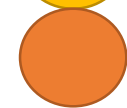
Solución candidata



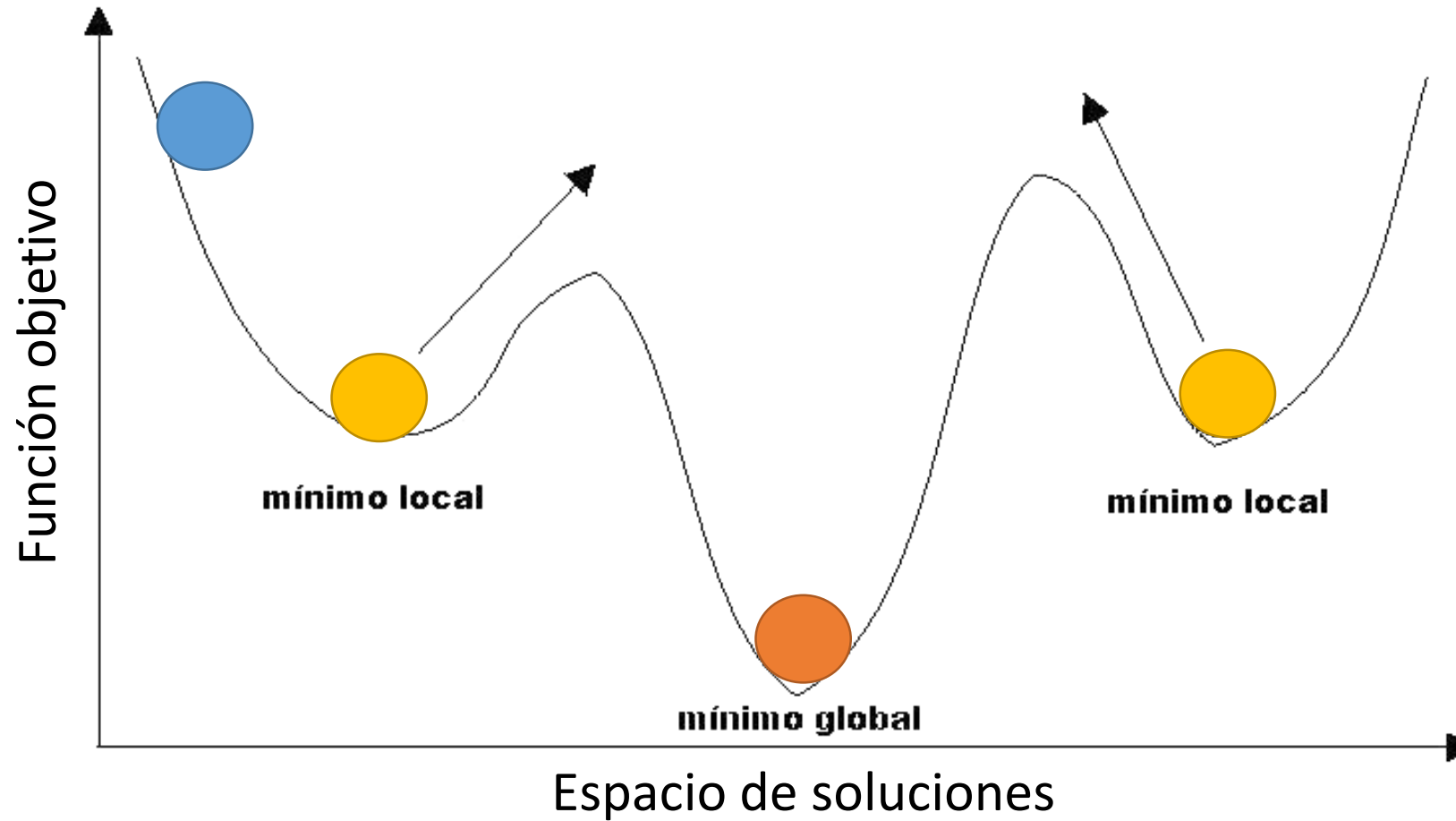
Solución factible



Solución óptima



Optimización de funciones



Algoritmos metaheurísticos se inspiran de varias fuentes

1. Algoritmos Culturales
2. Optimización de Partículas
3. Sociedad Anarquista
4. Competitivo Imperialista
5. Búsqueda de Armonía
6. Rueda Chirriante
7. Optimización social cognitiva

Algoritmos metaheurísticos inspirados en procesos físicos

1. Templado Simulado
2. Gran Explosión Gran Colapso
3. Gravitacional
4. Similar a Electromagnetismo
5. Optimización de fuerza central
6. Dinámica de Formación de Ríos
7. Colisión de partículas

Algoritmos metaheurísticos inspirados en la biología

1. Sistema Inmune Artificial
2. Algoritmos Genéticos
3. Colonia de Hormigas
4. Enjambre de Abejas
5. Algoritmo de Luciérnagas
6. Algoritmo de Murciélagos
7. Búsqueda del Cardumen

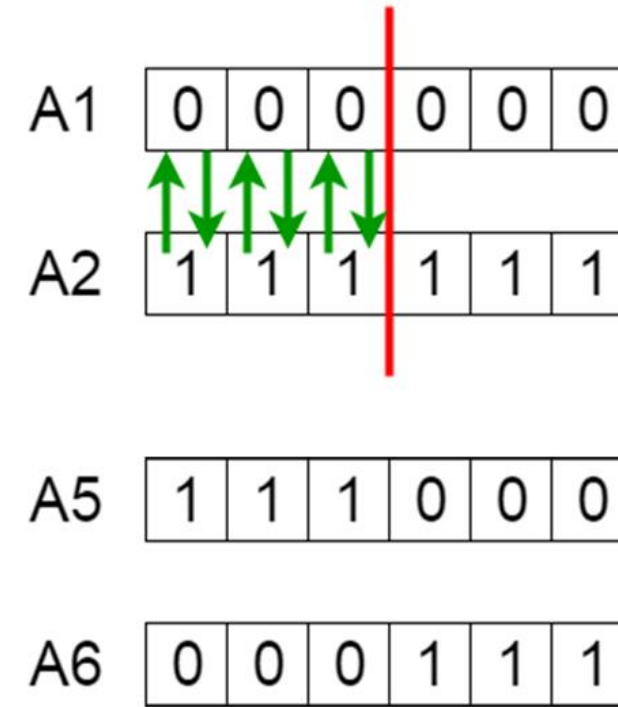
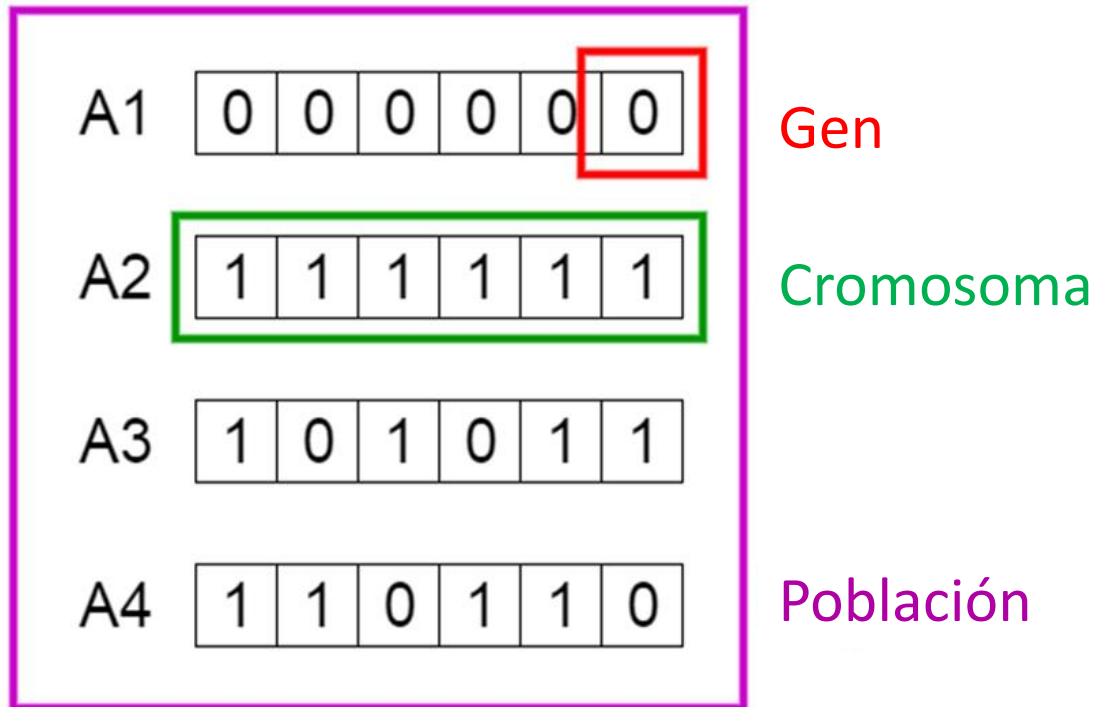
Algoritmos metaheurísticos inspirados en la biología

1. Sistema Inmune Artificial
2. Algoritmos Genéticos
3. Colonia de Hormigas
4. Enjambre de Abejas
5. Algoritmo de Luciérnagas
6. Algoritmo de Murciélagos
7. Búsqueda del Cardumen

Algoritmos genéticos

- Un algoritmo genético es una heurística de búsqueda inspirada en la teoría de evolución natural de Charles Darwin.
- Este algoritmo refleja el **proceso de selección natural** donde se seleccionan los individuos más aptos para la reproducción con el fin de producir descendencia de la próxima generación.
- Primeras investigaciones por John Holland en los años 60.
- Popularizados por David Goldberg en los años 80.

Algoritmos genéticos



Noción de selección natural

- Comienza con la selección de los individuos más aptos de una población.
- Producen descendientes que heredan las características de los padres. Si los padres tienen una mejor “aptitud”, sus hijos serán mejores que los padres y tendrán una mejor oportunidad de sobrevivir.
- Este proceso continúa iterando y, al final, se encontrará una generación con los individuos más aptos.

Noción de selección natural

- Esta noción se puede aplicar para un problema de búsqueda.
- Consideramos un conjunto de soluciones para un problema y seleccionamos el conjunto de las mejores entre ellas.

Algoritmos genéticos

Se consideran cinco fases en un algoritmo genético:

1. Población inicial
2. Función de aptitud
3. Selección
4. Cruce
5. Mutación

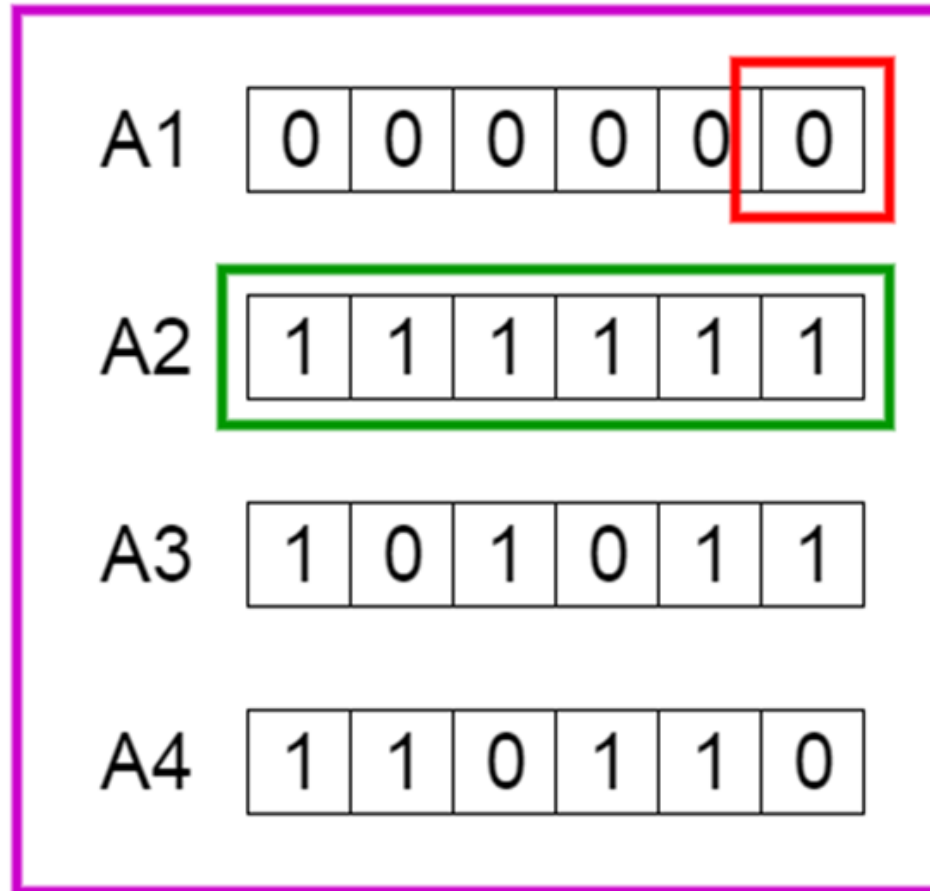
Población inicial

El proceso comienza con un conjunto de individuos que se llama **Población**. Cada individuo es una solución al problema que desea resolver.

Un individuo se caracteriza por un conjunto de parámetros (variables) conocidos como **Genes**. Los genes se unen en una cadena para formar un **Cromosoma** (solución).

En un algoritmo genético, el conjunto de genes de un individuo se representa usando una cadena. Por lo general, se utilizan valores binarios (cadena de 1s y 0s).

Población inicial



Gen

Cromosoma

Población

Función de aptitud

La función de aptitud (*fitness function*) determina qué tan apto está un individuo (la capacidad de un individuo para competir con otros individuos).

Se le da un puntaje de aptitud (o fitness) a cada individuo. La probabilidad de que un individuo sea seleccionado para la reproducción, se basa en su puntaje de aptitud.

Selección

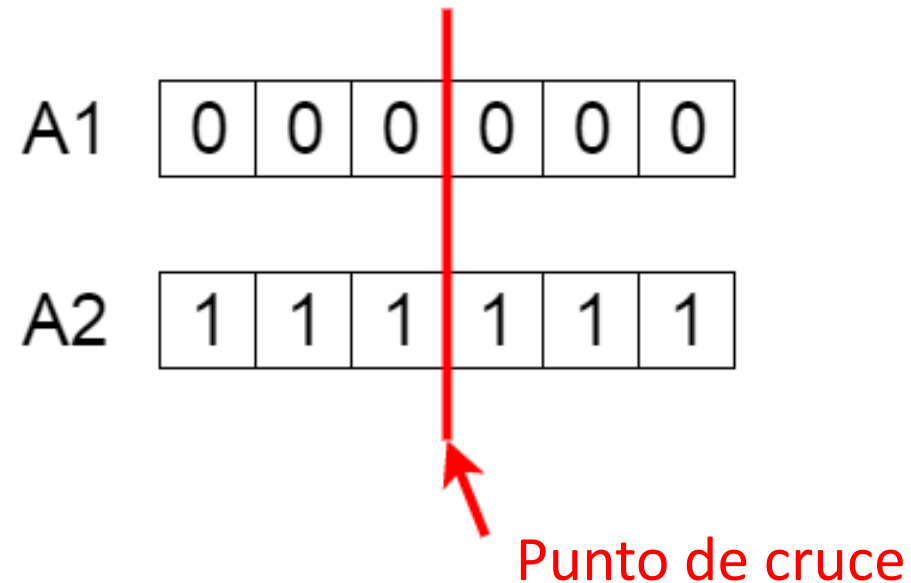
Seleccionar a los individuos más aptos y dejarlos pasar sus genes a la próxima generación.

Se seleccionan dos pares de individuos (padres) en función de sus puntajes de aptitud. Los individuos con buen puntaje tienen más posibilidades de ser seleccionados para la reproducción.

Cruce

El cruce (*crossover*) es la fase más significativa en un algoritmo genético. Para cada pareja de padres que se van a unir, se elige un punto de cruce al azar dentro de los genes.

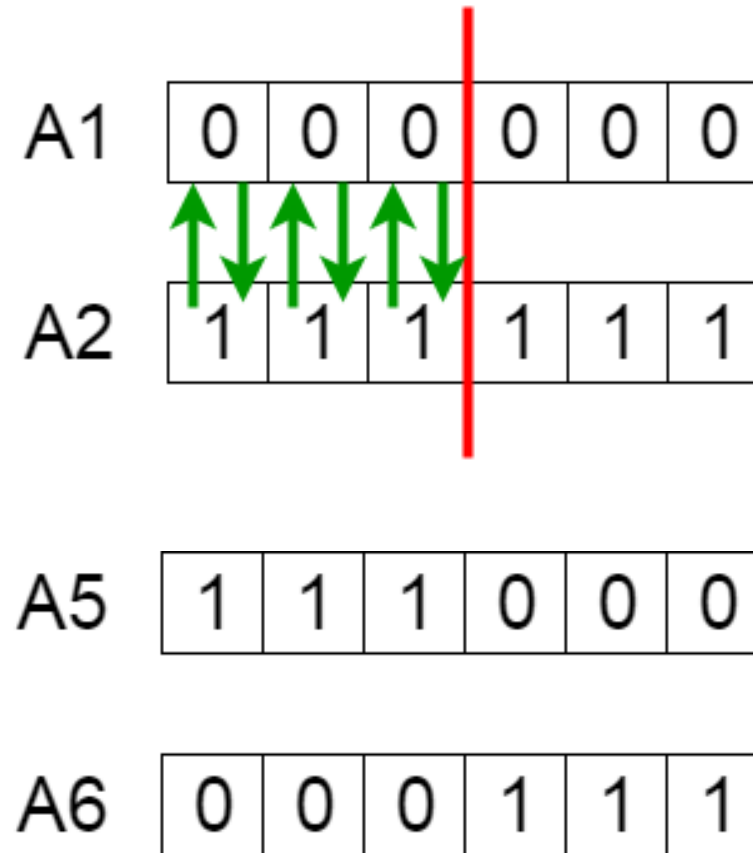
Por ejemplo, considerar que el punto de cruce es 3 como se muestra a continuación:



Cruce

La **descendencia** se crea intercambiando los genes de los padres entre ellos hasta que se alcanza el punto de cruce.

Los nuevos descendientes se agregan a la población.



Mutación

En ciertos nuevos descendientes formados, algunos de sus genes pueden estar sujetos a una **mutación** con una baja probabilidad aleatoria. Esto implica que algunos de los bits en la cadena de bits se pueden voltear.

Antes de la mutación

A5

1	1	1	0	0	0
---	---	---	---	---	---

Después de la mutación

A5

1	1	0	1	1	0
---	---	---	---	---	---

La mutación ocurre para mantener la diversidad dentro de la población y prevenir la **convergencia prematura**.

Terminación

El algoritmo termina si la población ha convergido (no produce descendencia que sea significativamente diferente de la generación anterior). Luego se dice que el algoritmo genético ha proporcionado un conjunto de soluciones al problema.

La población tiene un tamaño fijo. A medida que se forman nuevas generaciones, los individuos con menor aptitud “mueren”, lo que proporciona espacio para nuevos descendientes.

La secuencia de fases se repite para producir individuos en cada nueva generación que son mejores que la generación anterior.

Flujo del algoritmo

INICIO

Generar la población inicial

Calcular aptitud

REPETIR

Selección

Cruce

Mutación

Calcular aptitud

HASTA que la población haya convergido

PARAR