

# Optimizing a Transportation System Using Metaheuristics Approaches (EGD/GA/ACO): A Forest Vehicle Routing Case Study

Hossein Havaeji, Thien-My Dao, Tony Wong

Mechanical Engineering Department, école de technologie supérieure, Montreal, Canada.

**DOI:** [10.4236/wjet.2024.121009](https://doi.org/10.4236/wjet.2024.121009)

Alvarez Lucía, Germani Emiliano, Padilla Gonzalo

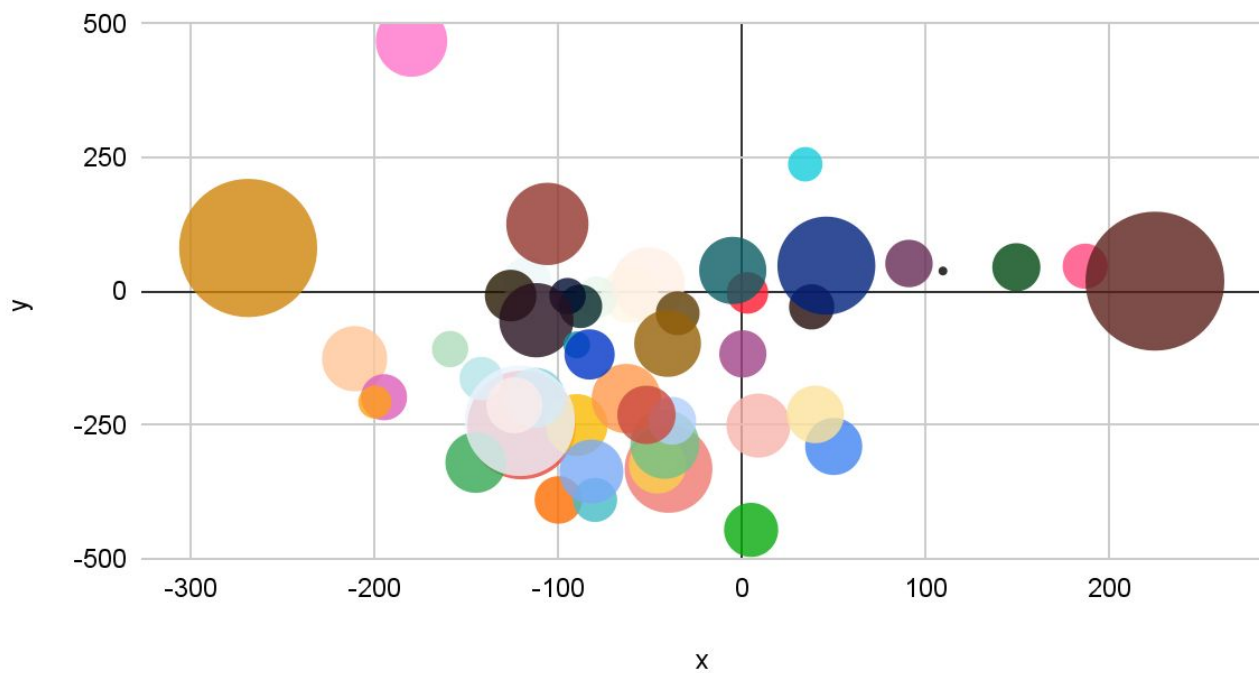
# Problema

Se cuenta con un depósito central y 50 clientes que deben recibir mercancía, utilizando 9 vehículos de capacidad limitada ( $50 \text{ m}^3$  cada uno) y velocidad promedio de 65 km/h. Cada cliente debe ser atendido exactamente una vez, respetando las restricciones de capacidad y tiempos de servicio.

**El objetivo es diseñar rutas óptimas para la flota que minimicen simultáneamente el costo total de transporte (costos fijos, por kilómetro, tiempos de ruta y conductor, esperas y servicios) y la distancia total recorrida, asegurando eficiencia en la logística.**

# Problema

Ubicación y demanda de clientes



# Forma de la solución

No. of vehicle	ACO Route	Length (km)	Capacity used by each vehicle (m <sup>3</sup> )
1	[9, 1]	111.38	42.782
2	[19, 3, 49, 36]	193.93	30.83
3	[20, 10, 5, 4, 18, 13, 48]	169.3	48.38
4	[33,44]	76.35	49.611
5	[30, 38, 37]	53.74	51.384
6	[2, 6, 11, 12, 15, 17, 24, 27, 28, 32, 35]	288.11	42.297
7	[29, 47, 25]	97.01	12.403
8	[43, 46, 34, 42, 45, 7, 16]	355.13	44.913
9	[26, 31, 8, 14, 21, 22, 23, 39, 40, 41]	334.86	25.158
Total Distance: 1679.81			

# Metodología

Formulación matemática del problema: Modelado VRP (Vehicle Routing Problem)

- **Componentes de costo:** Distancia, tiempo de ruta, conductor, servicio, espera.
- **Restricciones:** capacidad máxima de vehículos, conexión de rutas, tiempos de viaje, clientes atendidos una sola vez.

Parámetros de Simulación

- 50 clientes
- 9 vehículos con 50m<sup>3</sup> de capacidad
- Velocidad promedio 65 km/h
- Costos incluidos en la función objetivo

# Metodología

## Aplicación de Metaheurísticas

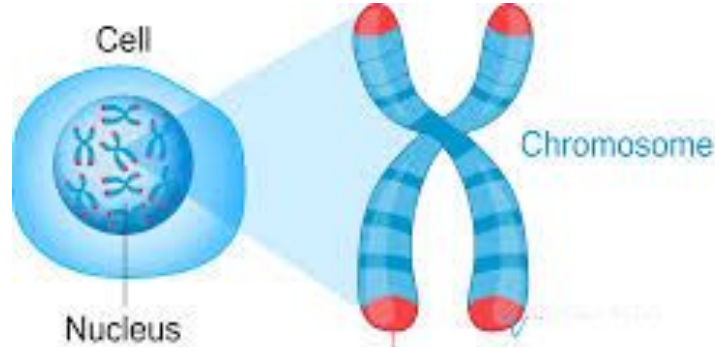
- Genetic Algorithm
- Ant Colony Optimization
- Extended Great Deluge

## Ejecución y Comparación

Se comparan:

- Costo total de transporte
- Distancia total recorrida
- Tiempo de cómputo
- Convergencia

# Genetic Algorithm (GA): una metaheurística evolutiva



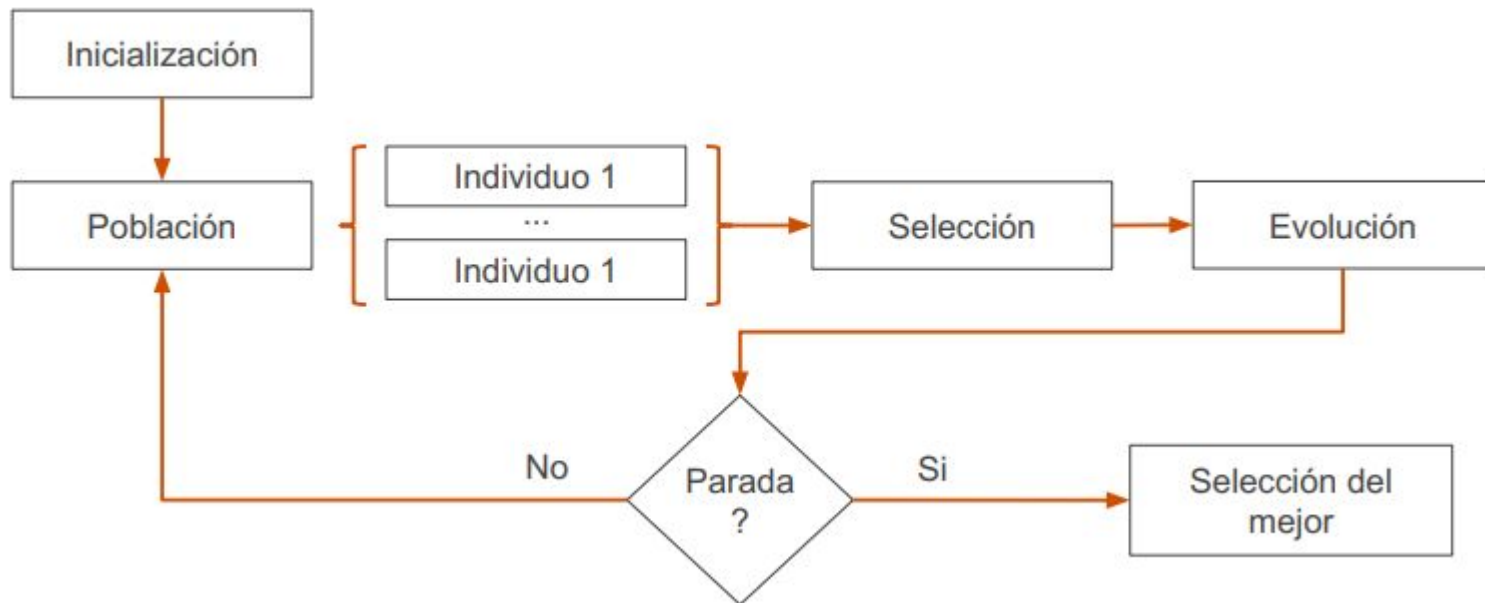
Imita la selección natural de Darwin: las soluciones más aptas tienen más probabilidades de reproducirse.

**Población** = conjunto de soluciones.

**Cromosomas** = representación de cada solución.

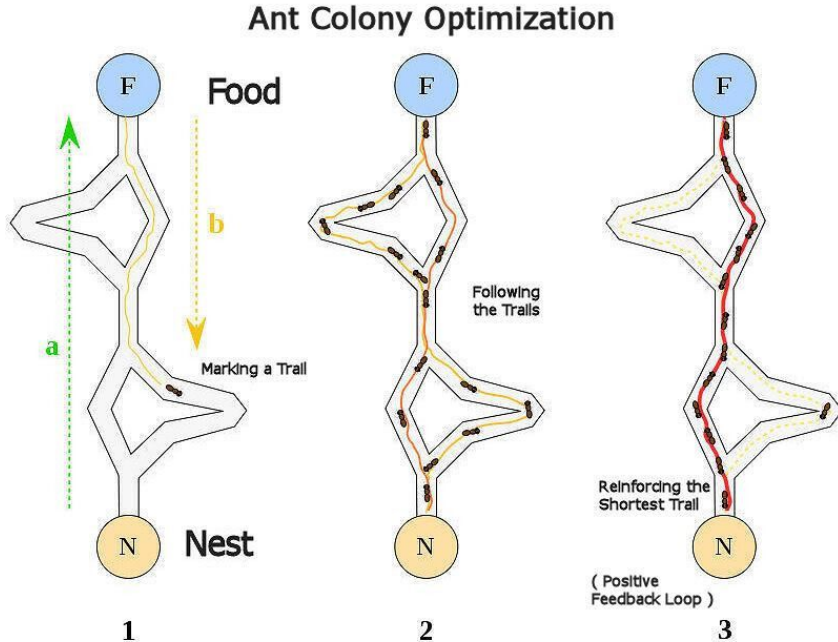
**Fitness** = calidad de cada solución.

# Pasos del algoritmo GA





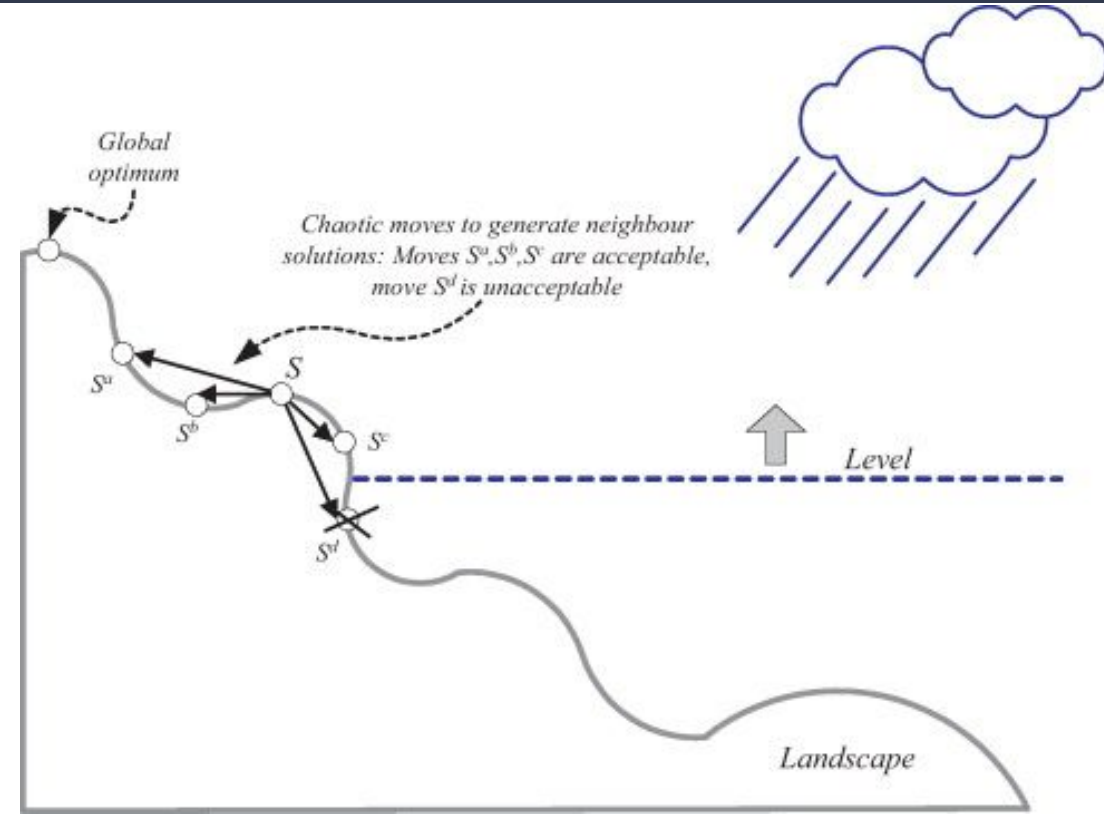
# Ant Colony Optimization (ACO)



[http://en.wikipedia.org/wiki/Ant\\_colony\\_optimization](http://en.wikipedia.org/wiki/Ant_colony_optimization)

- Construcción de soluciones:  
Cada hormiga considera la longitud de la arista y el nivel de feromona.
- Actualización de feromonas:  
Se evaporan las feromonas para reducir el efecto en caminos antiguos y se depositan nuevas en aquellos caminos que construyeron mejores soluciones.

# Great Deluge (GD)



Metaheurística inspirada en **el nivel de agua que sube**.

Se empieza con una **solución inicial** y un **nivel de agua** por debajo de su calidad.

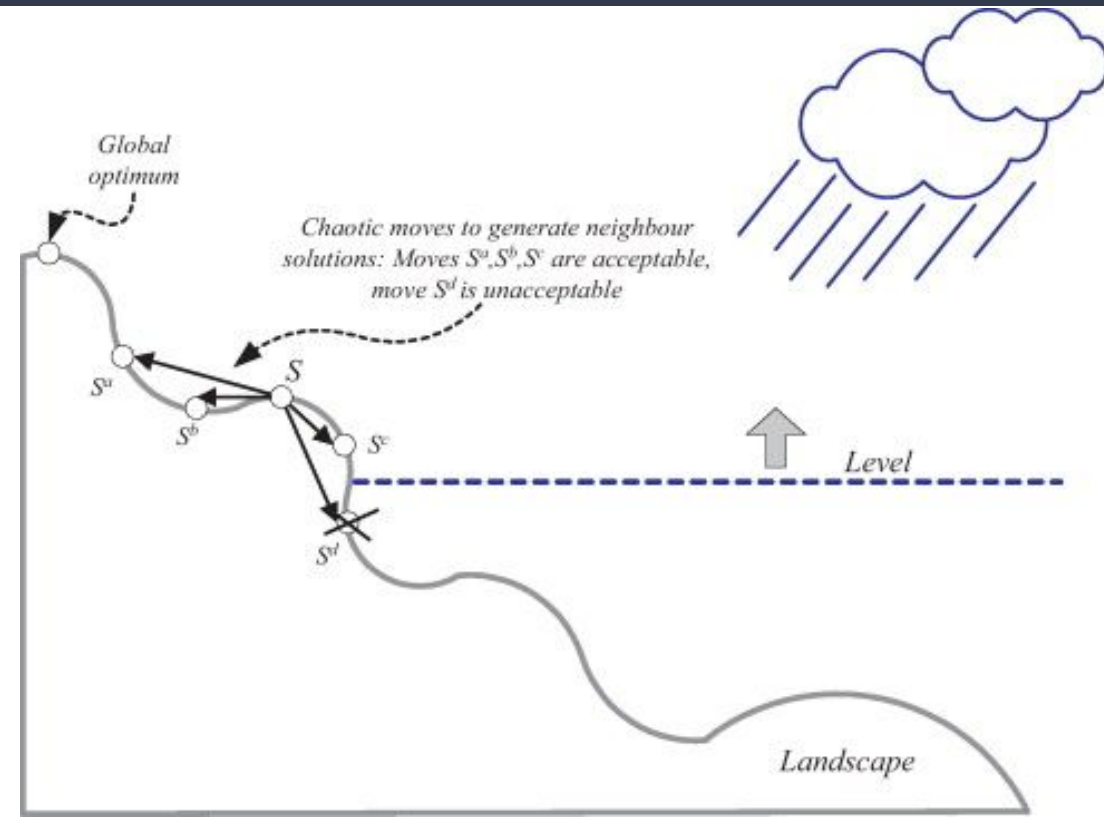
Se **aceptan soluciones** si su calidad  $\geq$  nivel de agua.

El **nivel de agua aumenta progresivamente**, restringiendo la aceptación de soluciones peores.

Permite **exploración inicial** y **explotación posterior**.

Principal uso: **problemas combinatorios** como timetabling, scheduling, rutas.

# Extended Great Deluge (EGD)



Mejora del GD con **ajuste dinámico del nivel de agua**.

Nivel de agua se adapta según:

- Mejor solución encontrada.
- Mejora reciente del costo.
- Estancamiento en la búsqueda.

Permite **aceptar temporalmente soluciones peores** para escapar de mínimos locales.

Flexible en la **generación de candidatos**: puede ser aleatoria o guiada por heurística.

Mantiene **equilibrio entre exploración y explotación**.

Aplicaciones: scheduling, timetabling, optimización combinatoria compleja.

# Función de costo

$$\begin{aligned} \text{TheMin}C_t = MC_f &+ \sum_{i=1}^{L+1} \sum_{j=1}^{L+1} \sum_{k=1}^m C_{ijk} D_{ij} X_{ijk} \\ &+ (C_{vt} + C_{dt}) \sum_{i=1}^{L+1} \sum_{j=1}^{L+1} \sum_{k=1}^m X_{ijk} (T_{ij} + W_{jk} + S_j) \end{aligned}$$

$MC_f$  = El costo fijo total del vehículo

$C_f$  = El coste fijo unitario del vehículo, que cubre la carga y descarga

# Función de costo

$$\sum_{i=1}^{L+1} \sum_{j=1}^{L+1} \sum_{k=1}^m C_{ijk} D_{ij} X_{ijk}$$

Suma total del costo de la distancia

$C_{ijk}$  = El coste unitario de transporte por kilómetro del vehículo k de i a j

$D_{ij}$  = La distancia entre dos ubicaciones i y j

$X_{ijk}$  = Vehículo k va de i a j

# Función de costo

$$(C_{vt} + C_{dt}) \sum_{i=1}^{L+1} \sum_{j=1}^{L+1} \sum_{k=1}^m X_{ijk} (T_{ij} + W_{jk} + S_j)$$

Suma del costo total del tiempo de ruta del vehículo y el costo total del tiempo de trabajo del conductor

$C_{vt}$  = El costo del tiempo de ruta unitario del vehículo

$C_{dt}$  = El costo del tiempo de trabajo del conductor unitario

$W_{jk}$  = Tiempo de espera del vehículo k en casa del cliente j

$S_j$  = El tiempo de servicio al cliente j

# Función Tiempo

$$T_{ij} = D_{ij}/V \quad i \in [1, L + 1]; j \in [1, L + 1]$$

$$W_{ik} = \max (T_i^s - T_{ik} \cdot 0) \quad i \in [1, L + 1]; k \in [1, m]$$

$T_{ij}$  = Tiempo de llegada del vehículo k al cliente i

$W_{ik}$  = Tiempo de espera del vehículo k en casa del cliente i

$S_i$  = Tiempo de servicio al cliente i

# Resultados

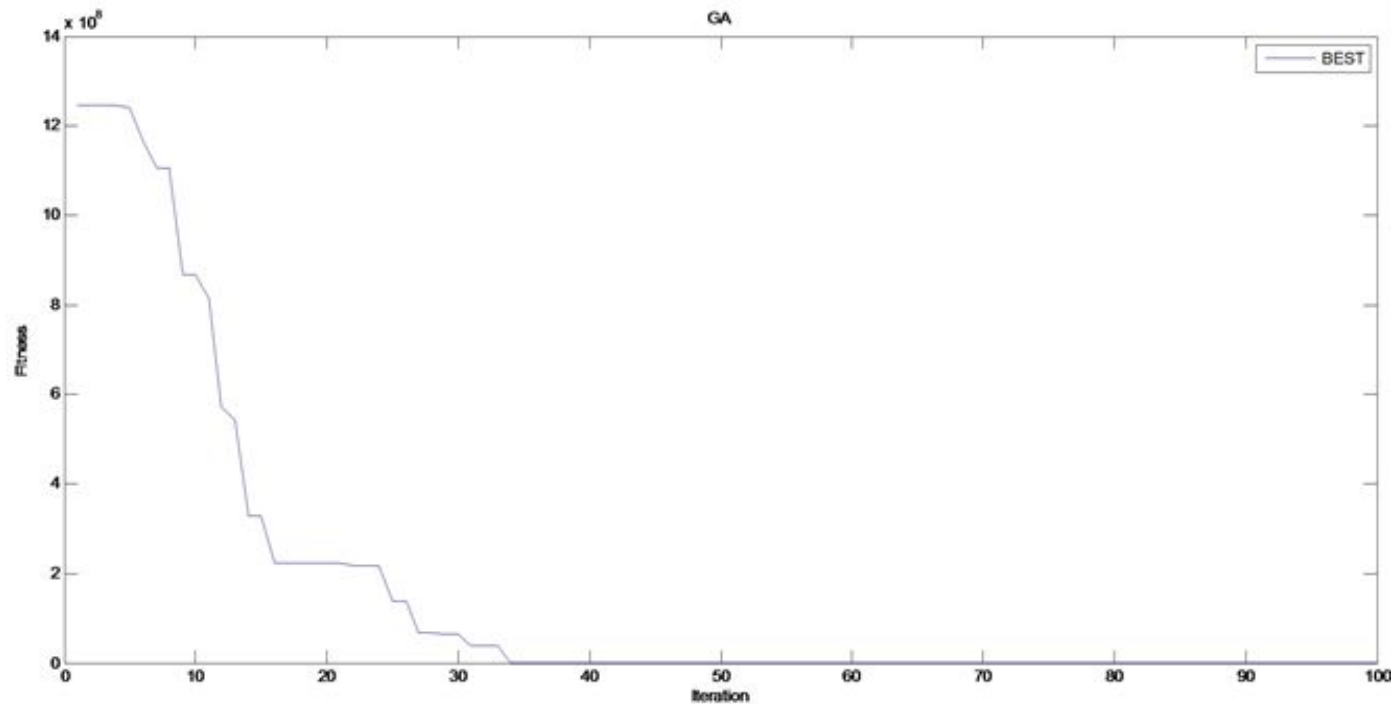
**Genetic Algorithm (GA):** alcanzó el menor costo (~19.901 USD), menor distancia (~1.093 km), con convergencia rápida (~100 iteraciones) y tiempo de cómputo razonable (~1.540 s / ~25 min).

**Extended Great Deluge (EGD):** mayor costo (~48.930 USD) y distancia mayor (~4383 km) , convergencia muy lenta (~200 iteraciones), y tiempo de cómputo más alto (~3.110 s / ~51 min).

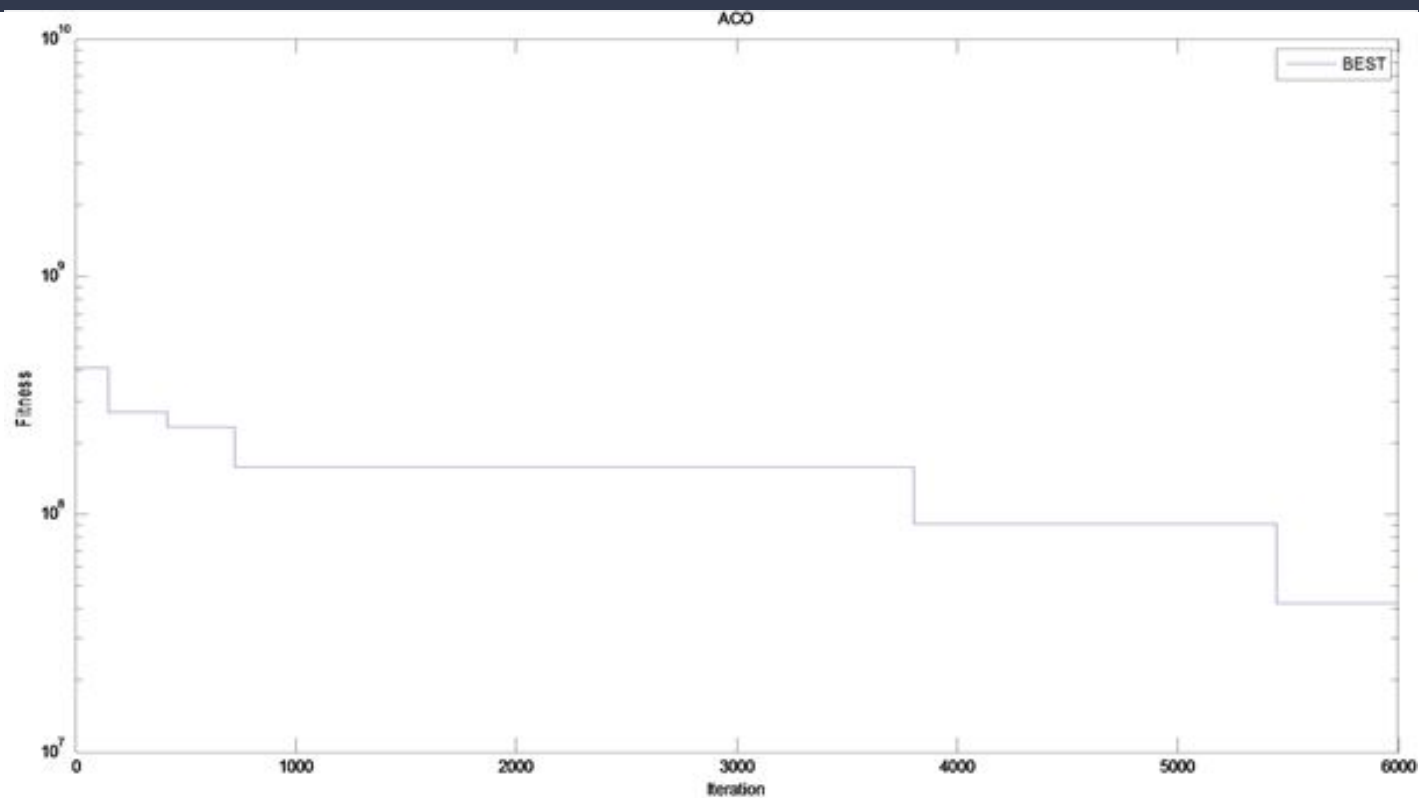
**Ant Colony Optimization (ACO):** costo excesivo (~42 millones USD), distancia mayor (~1.679 km), convergencia muy lenta (~6000 iteraciones) y tiempo de cómputo alto (~1.889 s / ~31 min).



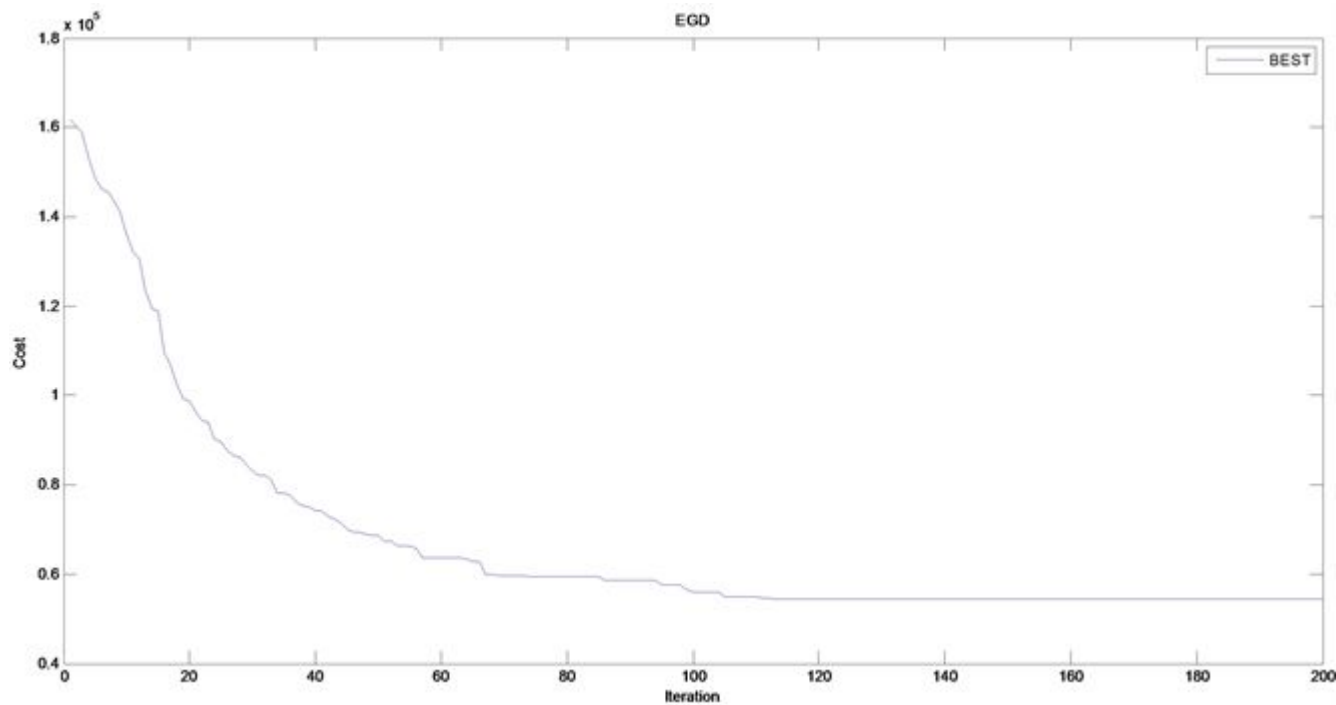
# Resultados / GA



# Resultados / ACO



# Resultados / EGD



## Comparación de los Algoritmos

Número de camión	Ruta GA	Longitud GA (km)	Ruta ACO	Longitud del ACO (km)	Ruta EGD	Longitud EGD (km)
1	[22, 43, 17, 46, 47, 2 12]	147.82	[9, 1]	111.38	[30, 50, 34, 45, 35, 36]	538.46
2	[29, 36, 35, 50, 34, 33, 32]	78.77	[19, 3, 49, 36]	193.93	[31, 38, 18, 12, 3]	584.24
3	[31, 20, 7, 49, 9, 10]	150.38	[20, 10, 5, 4, 18, 13, 48]	169.3	[32, 41, 44]	1057.49
4	[23, 41, 44]	166.11	[33, 44]	76.35	[39, 2, 4, 6, 49, 1]	1009.63
5	[38, 19, 4, 5, 6, 48, 40, 30]	137.71	[30, 38, 37]	53.74	[33, 42, 43]	687.35
6	[45, 15, 1, 8, 14]	147.39	[2, 6, 11, 12, 15, 17, 24, 27, 28, 32, 35]	288.11	[21, 27, 28, 16, 17, 46, 47, 19, 20]	702.53
7	[13, 3, 28, 25, 24, 26, 21]	110.90	[29, 47, 25]	97.01	[10, 9, 8, 7, 5, 15, 40]	884.95
8	[39, 11, 18, 16, 27, 42]	121.87	[43, 46, 34, 42, 45, 7, 16]	355.13	[29, 23, 22, 26, 24, 25]	325.77
9	[37]	32.48	[26, 31, 8, 14, 21, 22, 23, 39, 40, 41]	334.86	[37, 11, 48, 13, 14]	560.24
Distancia total	1093.46		1679.81		6350.66	
Tiempo de cálculo por computadora (seg)	1540.7213		1889.60		3110.01	
Mejor costo (\$)	19901.1417		42345519.77		48930.6899	

# Conclusión

Vimos que en el caso de ruteo de vehículos forestales, el AG encontró soluciones con menor costo y distancia que otros métodos como ACO o EGD, demostrando su capacidad para adaptarse a problemas reales y de gran escala.

**Extended Great Deluge (EGD)** no tuvo tan buen rendimiento como (GA) ya que es propenso a quedarse estancado en un mínimo local, una vez que el nivel del agua ya es muy alto.

**Ant Colony Optimization (ACO):** Es un algoritmo que necesita muchas especificaciones para funcionar mejor, a veces requiere muchas hormigas para converger, requiere mucho poder de cómputo para calcular probabilidades y actualizar los niveles de feromonas, suele estancarse en caminos explorados al inicio muchas veces.