

Caso 03 - TSP con Deadlines: ruta mínima con entregas antes de caducar

Enunciado

Un camión sale del depósito (punto 0) y debe visitar todos los clientes exactamente una vez.

Cada cliente i tiene un tiempo límite (deadline). Si el camión llega después del deadline, el material se considera caducado.

Objetivo: encontrar el orden de visita que minimiza la distancia total, cumpliendo todos los deadlines (o minimizando penalización por tardanza).

Datos y parámetros

- Velocidad: 1 unidad de distancia = 1 minuto (para convertir distancia en tiempo)
- Tiempo de servicio: columna servicio_min (minutos) al llegar a cada cliente
- Restricción: $\text{llegada}_i \leq \text{deadline}_i$

Dataset de muestra (pequeño)

Este dataset es intencionalmente pequeño para poder validar a mano la evaluación de soluciones.

| punto | tipo | x | y | deadline_min | servicio_min |
|-------|----------|---|---|--------------|--------------|
| 0 | deposito | 0 | 0 | 0 | 0 |
| 1 | cliente | 2 | 7 | 18 | 2 |
| 2 | cliente | 6 | 4 | 22 | 2 |
| 3 | cliente | 8 | 9 | 35 | 3 |
| 4 | cliente | 3 | 1 | 16 | 2 |
| 5 | cliente | 9 | 2 | 28 | 2 |
| 6 | cliente | 5 | 8 | 30 | 2 |

Pistas para un Algoritmo Genético

Representación (cromosoma)

- Permutación de clientes $[1..n]$.

Fitness (evaluación)

- Simula la ruta acumulando tiempo: $t += \text{viaje} + \text{servicio}$.
- Coste base: distancia_total.
- Penalización: $\text{tardanza_total} = \text{suma}(\max(0, \text{llegada}_i - \text{deadline}_i))$.
- Fitness final típica: $\text{distancia_total} + \lambda * \text{tardanza_total}$ (λ grande).

Mutación (ejemplos)

- Inversión 2-opt para reducir distancia sin romper demasiado la factibilidad.
- Move (inserción): sacar un cliente de una posición y reinsertarlo en otra.

Cruce (ejemplos)

- OX para mantener orden relativo.
- Edge recombination (avanzado): intenta preservar aristas frecuentes.

Selección (ejemplos)

- Torneo + elitismo. En problemas con restricciones, elitismo ayuda a no perder soluciones factibles.