

Análisis de ACO sobre Instancias TSPLIB

Este informe resume los resultados de ACO con grid search sobre 9 instancias TSPLIB.

☐ Explicación de los Parámetros del Algoritmo ACO

En el experimento se ajustaron distintos parámetros del algoritmo de colonia de hormigas (ACO) para resolver problemas de rutas (TSP). Cada uno de estos parámetros tiene un efecto directo sobre el comportamiento del algoritmo. A continuación se explica qué hace cada uno y cómo afectó los resultados observados:

☐ **n_ants** (Número de hormigas)

Qué significa: Es la cantidad de soluciones simultáneas que el algoritmo construye en cada iteración.

Cómo influye: Más hormigas permiten explorar más caminos posibles, pero también aumentan el tiempo de cómputo.

Lo que observamos: - En problemas pequeños, usar pocas hormigas fue suficiente para encontrar buenas rutas. - En problemas más grandes, aumentar este valor ayudó a mejorar la calidad de las soluciones, aunque con mayor tiempo de ejecución.

☐ **alpha** (Importancia de la feromona)

Qué significa: Controla cuánto influye el rastro de feromonas (experiencia previa) en la elección de caminos.

Cómo influye: Valores altos hacen que las hormigas se guíen más por los caminos ya marcados por otras, lo que puede llevar a soluciones estables pero también a estancarse.

Lo que observamos: - Un valor medio ($\alpha = 1.0$) fue generalmente el más efectivo. - Valores bajos ($\alpha = 0.5$) dieron buenos resultados en problemas pequeños, al favorecer la exploración.

☐ **beta** (Importancia de la visibilidad)

Qué significa: Mide cuánto influye la distancia entre ciudades en la decisión de las hormigas. A mayor beta, más peso se da a las ciudades más cercanas.

Cómo influye: Valores altos favorecen decisiones 'cortoplacistas' (ir al nodo más cercano), mientras que valores bajos permiten rutas más globales.

Lo que observamos: - En la mayoría de las instancias, un valor alto ($\beta = 2.0$) mejoró los resultados, ya que dirigió a las hormigas por rutas más cortas de forma más efectiva.

☐ **ro** (Tasa de evaporación)

Qué significa: Define qué tan rápido se 'evapora' la feromona acumulada en los caminos.

Cómo influye: Un valor bajo permite que las feromonas duren más tiempo, ayudando a mantener el conocimiento colectivo. Un valor alto promueve olvidar caminos rápidamente y explorar más.

Lo que observamos: - Una evaporación moderada ($\rho = 0.3$) fue útil para balancear exploración y explotación. - En instancias grandes, una evaporación baja ($\rho = 0.1$) ayudó a mantener las rutas buenas por más tiempo y evitó perder el progreso.

☐ Conclusión general

El éxito del algoritmo ACO depende de encontrar un equilibrio entre explorar nuevas rutas (exploración) y reforzar las mejores encontradas (explotación). Ajustar los parámetros correctamente puede marcar la diferencia entre una solución mediocre y una ruta casi óptima. Los resultados muestran que no hay una única combinación ganadora, pero ciertos patrones se repiten: - beta alto suele ser beneficioso. - alpha medio o bajo ayuda a explorar mejor en problemas pequeños. - ro bajo mantiene la estabilidad en problemas grandes. - n_ants debe adaptarse al tamaño del problema.

☐ Conclusiones y Recomendaciones por Instancia

☐ **BERLIN52**

- ☐ GAP promedio general: 16.19%
- ☐ Mejor GAP alcanzado: 0.07% con parámetros:
- n_ants: 20

- alpha: 1.0
- beta: 2.0
- ro: 0.1
- ☐ ☐ Tiempo medio de ejecución: 75.62 segundos

Recomendación: Usar esta configuración balanceada para berlin52.

☐ BURMA14

- ☐ GAP promedio general: 0.57%
- ☐ Mejor GAP alcanzado: 0.00% con parámetros:
- n_ants: 10
- alpha: 0.5
- beta: 2.0
- ro: 0.1
- ☐ ☐ Tiempo medio de ejecución: 18.52 segundos

Recomendación: Usar esta configuración balanceada para burma14.

☐ CH150

- ☐ GAP promedio general: 31.65%
- ☐ Mejor GAP alcanzado: 3.29% con parámetros:
- n_ants: 20
- alpha: 1.0
- beta: 2.0
- ro: 0.1
- ☐ ☐ Tiempo medio de ejecución: 280.27 segundos

Recomendación: Usar esta configuración balanceada para ch150.

☐ KROA100

- ☐ GAP promedio general: 27.60%
- ☐ Mejor GAP alcanzado: 6.16% con parámetros:
- n_ants: 10
- alpha: 1.0
- beta: 1.0
- ro: 0.3
- ☐ ☐ Tiempo medio de ejecución: 164.17 segundos

Recomendación: Usar esta configuración balanceada para kroA100.

☐ PR439

- ☐ GAP promedio general: 48.64%
- ☐ Mejor GAP alcanzado: 11.27% con parámetros:
- n_ants: 20
- alpha: 1.0
- beta: 2.0
- ro: 0.1
- ☐ ☐ Tiempo medio de ejecución: 1710.91 segundos

Recomendación: Usar esta configuración balanceada para pr439.

☐ ST70

- ☐ GAP promedio general: 21.61%
- ☐ Mejor GAP alcanzado: 6.07% con parámetros:
- n_ants: 20
- alpha: 1.0
- beta: 2.0
- ro: 0.3
- ☐ ☐ Tiempo medio de ejecución: 108.98 segundos

Recomendación: Usar esta configuración balanceada para st70.

☐ TSP225

- ☐ GAP promedio general: 40.09%
- ☐ Mejor GAP alcanzado: 8.45% con parámetros:
- n_ants: 20
- alpha: 1.0
- beta: 2.0
- ro: 0.3
- ☐ ☐ Tiempo medio de ejecución: 492.28 segundos

Recomendación: Usar esta configuración balanceada para tsp225.

☐ **ULYSSES16**

- ☐ GAP promedio general: 0.55%
- ☐ Mejor GAP alcanzado: 0.00% con parámetros:
- n_ants: 10
- alpha: 1.0
- beta: 1.0
- ro: 0.1
- ☐☐ Tiempo medio de ejecución: 21.36 segundos

Recomendación: Usar esta configuración balanceada para ulysses16.

☐ **ULYSSES22**

- ☐ GAP promedio general: 2.28%
- ☐ Mejor GAP alcanzado: 0.00% con parámetros:
- n_ants: 30
- alpha: 1.0
- beta: 1.0
- ro: 0.3
- ☐☐ Tiempo medio de ejecución: 29.27 segundos

Recomendación: Usar esta configuración balanceada para ulysses22.