

# Proyecto Etapa III Implementación de Metaheurísticas:

Optimización en logística para la empresa LogistiCo

**Curso:** ISIS-3302  
Departamento de Ingeniería de Sistemas  
Universidad de los Andes

## 1 Introducción

Este documento presenta la tercera etapa del proyecto de optimización en logística para LogistiCo. Esta etapa se enfoca en la implementación de un método metaheurístico para resolver exclusivamente el problema base del CVRP (Capacitated Vehicle Routing Problem) y comparar los resultados con las soluciones obtenidas mediante Pyomo en la fase anterior. El objetivo es analizar las ventajas y desventajas de cada enfoque en términos de calidad de solución, tiempo de ejecución y escalabilidad.

## 2 Evaluación de los entregables

Para la entrega correspondiente a esta etapa, los estudiantes deberán presentar un documento en formato PDF o *Jupyter Notebook* que incluya el desarrollo completo de la implementación de un método metaheurístico para resolver el CVRP base, así como la comparación con las soluciones obtenidas mediante Pyomo. Los entregables se evaluarán conforme a los siguientes criterios:

### 1. Implementación de Metaheurísticas (35 %):

- Correcta implementación de un algoritmo metaheurístico para resolver el CVRP base.
- Adecuada parametrización y calibración del algoritmo implementado.
- Correcto manejo de las restricciones del problema.

---

## **2. Comparación con Soluciones Previas (25 %):**

- Análisis comparativo detallado entre las soluciones obtenidas mediante la metaheurística y las obtenidas con Pyomo.
- Evaluación de calidad de solución (valor de función objetivo), tiempo de ejecución y recursos computacionales.
- Discusión sobre ventajas y desventajas de cada enfoque.

## **3. Análisis de Escalabilidad (20 %):**

- Evaluación de la escalabilidad del método metaheurístico implementado para instancias de mayor tamaño.
- Análisis de la degradación de rendimiento a medida que aumenta el tamaño del problema.
- Conclusiones sobre la aplicabilidad de cada enfoque para problemas reales de gran escala.

## **4. Visualización y Evaluación (10 %):**

- Visualizaciones adecuadas para comparar las rutas obtenidas por el método metaheurístico y Pyomo.
- Gráficos de convergencia y evolución de las soluciones para el método metaheurístico.

## **5. Documentación y Calidad del Código (10 %):**

- Código limpio, bien estructurado y adecuadamente comentado.
- Modularidad y reutilización de componentes.
- Manejo efectivo de excepciones y casos límite.
- Documentación clara que permita comprender la implementación.

---

## 3 Instrucciones

### 1. Ámbito del problema y conjuntos de datos

- **Problema a resolver:** Se abordará el *Capacitated Vehicle Routing Problem (CVRP)* estándar con una flota homogénea. Todas las variantes adicionales (ventanas de tiempo, múltiples depósitos, reabastecimiento, etc.) **quedan fuera del alcance** de esta fase.

- **Instancias a ejecutar:**

Proyecto\_Caso\_Base, Proyecto\_A\_Caso2, Proyecto\_A\_Caso3

Cada instancia corresponde a un subconjunto de clientes con distinta escala (pequeña, media y grande, respectivamente) y se almacena en su propio directorio de datos.

- **Archivos requeridos por instancia:**
  - `clients.csv`: lista de clientes y demandas.
  - `vehicles.csv`: se **omite** la columna `Capacity`
  - `depots.csv`: utilice únicamente el depósito definido en `Proyecto_Caso_Base`; ignore posibles depósitos adicionales.
- **Objetivo:** comparar la *escalabilidad* del método metaheurístico frente al modelo exacto (Pyomo) cuando crece el número de clientes.

### 2. Organización del repositorio

- Conserve la estructura de directorios establecida en fases anteriores.
- Actualice el `README.md` detallando:
  - Descripción del metaheurístico seleccionado.
  - Cómo reproducir los experimentos (rutas de datos, comandos, semillas, etc.).

---

### 3. Implementación del método metaheurístico

- Seleccione y programe **un** método entre: *Algoritmo Genético*, *Búsqueda Tabú*, *Recocido Simulado*, *Optimización por Colonia de Hormigas*, *Búsqueda Local Iterativa* u otro (con aprobación previa).
- Desarrolle operadores específicos para CVRP:
  - Representación de soluciones (cromosoma, lista de rutas, etc.).
  - Operadores de cruce, mutación y/o movimiento tabú.
  - Heurística de reparación o penalización para mantener la factibilidad (respeto de capacidad  $Q$  y unicidad de visita).

### 4. Calibración y experimentación

- Diseñe un plan experimental para sintonizar los parámetros del algoritmo.
- Ejecute al menos **3 corridas independientes** por configuración, usando semillas distintas.
- Registre tiempo de CPU, valor de la función objetivo y estadísticas relevantes (mejor, peor, media, desviación estándar).

### 5. Análisis comparativo

- Compare metaheurístico vs. Pyomo en:
  1. Calidad de la solución (Valor de la función objetivo obtenida en ambos casos).
  2. Tiempo de ejecución.
  3. Uso de memoria.
  4. Comportamiento al escalar (Casos 2 y 3).
- Analice diferencias cualitativas en las rutas (longitud, número de vehículos, balance de carga).
- Discuta ventajas y desventajas de ambos enfoques.

---

## 6. Visualización de resultados

- Grafique la evolución de la mejor solución (*curva de convergencia*) para el metaheurístico.
- Trace las rutas finales superpuestas en el plano para metaheurístico y Pyomo.
- Presente histogramas o cajas de distribución de cargas y longitudes de ruta.

## 7. Archivo de verificación

- Genere un archivo por metaheurístico:  
`verificacion_metaheuristica_X_<Número de Caso>.csv` ( $X$  = nombre del método).
- Use las mismas columnas que en la fase anterior para permitir comparación directa (cliente, secuencia de visita, distancia acumulada, etc.).

## 8. Entregables

- Código fuente y repositorio actualizado.
- Archivos `verificacion_metaheuristica_X_<Número de Caso>.csv` para las tres instancias.
- Visualizaciones comparativas y gráficos de convergencia.
- Informe técnico con análisis de resultados, discusión de escalabilidad y conclusiones.

# 4 Consideraciones para la Implementación de Metaheurísticas

## 4.1 Algoritmos Recomendados

Se recomienda seleccionar alguno de los siguientes algoritmos metaheurísticos para implementar:

- **Algoritmo Genético (GA):**

- 
- Desarrollar una representación adecuada para rutas de vehículos
  - Implementar operadores de cruce específicos para CVRP.
  - Diseñar operadores de mutación que preserven la factibilidad
  - Implementar mecanismos de reparación para soluciones no factibles
- **Búsqueda Tabú (TS):**
    - Definir movimientos específicos para el CVRP (e.g., 2-opt, intercambio de clientes)
    - Implementar estructuras de memoria a corto y largo plazo
    - Desarrollar criterios de aspiración
    - Diseñar estrategias de diversificación e intensificación
  - **Optimización por Colonia de Hormigas (ACO):**
    - Diseñar estrategias de construcción de rutas
    - Implementar actualización de feromonas específica para CVRP
    - Definir heurísticas locales para guiar la búsqueda
    - Desarrollar mecanismos de balance entre exploración y explotación

## 4.2 Estructura del Código

Se recomienda seguir una estructura modular para la implementación de los algoritmos:

- **Módulo de representación:** Clases y estructuras para representar soluciones.
- **Módulo de evaluación:** Funciones para calcular la función objetivo y verificar restricciones.
- **Módulo de operadores:** Implementación de operadores específicos para cada metaheurística.
- **Módulo de algoritmo:** Implementación principal de cada metaheurística.
- **Módulo de experimentación:** Código para ejecutar experimentos, recopilar estadísticas y generar gráficos.
- **Módulo de visualización:** Funciones para visualizar soluciones y resultados comparativos.

---

## 5 Análisis Comparativo

Para realizar un análisis comparativo completo entre las soluciones obtenidas mediante Pyomo y las generadas por los métodos metaheurísticos, se sugiere considerar los siguientes aspectos:

### 5.1 Métricas de Calidad

Evaluar la calidad de las soluciones utilizando las siguientes métricas:

- **Valor de la función objetivo:** Comparar el costo total obtenido por cada método.
- **GAP de optimalidad:** Calcular la diferencia porcentual respecto a la solución óptima (o mejor conocida).
- **Factibilidad:** Verificar el cumplimiento de todas las restricciones.
- **Distribución de carga:** Analizar el balance de carga entre vehículos.
- **Distancia total recorrida:** Comparar la eficiencia de las rutas.

### 5.2 Métricas de Rendimiento

Evaluar el rendimiento computacional utilizando:

- **Tiempo de ejecución:** Medir el tiempo total requerido por cada método.
- **Memoria utilizada:** Cuantificar los recursos de memoria requeridos.
- **Escalabilidad:** Analizar cómo varía el rendimiento al aumentar el tamaño del problema.
- **Convergencia:** Estudiar la velocidad de convergencia de los métodos metaheurísticos.
- **Robustez:** Evaluar la consistencia de los resultados en múltiples ejecuciones.

---

## 5.3 Visualización Comparativa

Desarrollar las siguientes visualizaciones para facilitar la comparación:

- **Mapas comparativos:** Mostrar las rutas generadas por cada método.
- **Gráficos de convergencia:(Solo para las Metaheurísticas)** Visualizar la evolución de la función objetivo.
- **Gráficos de tiempo vs. calidad:** Analizar el compromiso entre tiempo y calidad.

## 6 Consideraciones Adicionales

### 6.1 Medidas de Complejidad

Es importante discutir las siguientes consideraciones de complejidad:

- **Complejidad teórica:** Analizar la complejidad algorítmica de los métodos implementados.
- **Complejidad práctica:** Evaluar el comportamiento real en términos de tiempo y memoria.
- **Escalabilidad:** Determinar límites prácticos para cada enfoque.
- **Complejidad de implementación:** Discutir el esfuerzo requerido para implementar y mantener cada método.

### 6.2 Calibración de Parámetros

Abordar los siguientes aspectos relacionados con la calibración:

- **Metodología de calibración:** Describir el enfoque utilizado (e.g., diseño de experimentos, optimización de parámetros).
- **Sensibilidad:** Analizar la sensibilidad de los resultados a cambios en los parámetros.
- **Parámetros críticos:** Identificar los parámetros que más impactan el rendimiento.
- **Configuraciones recomendadas:** Proporcionar valores recomendados para diferentes escenarios.



---

## 7 Estructura del Informe Final

El informe final debe seguir la siguiente estructura:

### 1. Descripción del Problema

- Formulación matemática del CVRP
- Características de la instancia base
- Restricciones y consideraciones

### 2. Método Implementado

- Descripción detallada del método metaheurístico implementado
- Estrategias de representación y operadores
- Proceso de calibración de parámetros

### 3. Resultados Experimentales

- Configuración experimental
- Presentación de resultados por método
- Análisis comparativo con soluciones Pyomo
- Visualizaciones y gráficos relevantes

### 4. Análisis de Escalabilidad

- Rendimiento en instancias de diferentes tamaños
- Límites prácticos de aplicabilidad
- Estrategias para mejorar la escalabilidad

### 5. Discusión

- Ventajas y desventajas de cada enfoque
- Recomendaciones para diferentes escenarios
- Lecciones aprendidas y desafíos encontrados

### 6. Conclusiones

- Resumen de hallazgos principales
- Respuestas a preguntas estratégicas
- Direcciones futuras de investigación

---

## 8 Consideraciones para la Entrega

- **Fecha límite:** La entrega debe realizarse a través de Bloque Neón antes del **Martes 27 de Mayo**
- **Formato:** El informe debe entregarse en formato PDF o como Jupyter Notebook.
- **Código fuente:** Todo el código fuente debe estar adecuadamente comentado y organizado.
- **Verificación:** Incluir los archivos de verificación para cada método implementado.
- **Grupos:** Esta actividad se puede realizar en grupos de hasta 3 integrantes.
- **Importante:** Adjuntar los nombres de los integrantes del equipo junto con los archivos de la entrega.

## 9 Criterios de Evaluación Detallados

### 1. Implementación de Metaheurísticas (35 %):

- Correcta implementación del algoritmo metaheurístico (15 %)
- Adecuada parametrización y calibración de los algoritmos (10 %)
- Correcto manejo de las restricciones del problema (10 %)

### 2. Comparación con Soluciones Previas (25 %):

- Análisis comparativo detallado entre metaheurísticas y Pyomo (10 %)
- Evaluación de calidad de solución, tiempo y recursos (10 %)
- Discusión sobre ventajas y desventajas de cada enfoque (5 %)

### 3. Análisis de Escalabilidad (20 %):

- Evaluación de escalabilidad para instancias mayores (10 %)
- Análisis de degradación de rendimiento (5 %)
- Conclusiones sobre aplicabilidad para problemas reales (5 %)

---

#### 4. Visualización y Evaluación (10 %):

- Visualizaciones comparativas de rutas (5 %)
- Gráficos de convergencia y evolución (5 %)

#### 5. Documentación y Calidad del Código (10 %):

- Código limpio y bien comentado (4 %)
- Modularidad y reutilización (3 %)
- Manejo de excepciones y casos límite (3 %)

## 10 Bibliografía Recomendada

Se recomienda consultar las siguientes referencias:

1. Gendreau, M., & Potvin, J. Y. (Eds.). (2010). Handbook of metaheuristics (Vol. 146). Springer Science & Business Media.
2. Toth, P., & Vigo, D. (Eds.). (2014). Vehicle routing: problems, methods, and applications. Society for Industrial and Applied Mathematics.
3. Laporte, G. (2009). Fifty years of vehicle routing. *Transportation Science*, 43(4), 408-416.
4. Golden, B. L., Raghavan, S., & Wasil, E. A. (Eds.). (2008). The vehicle routing problem: latest advances and new challenges (Vol. 43). Springer Science & Business Media.
5. Cordeau, J. F., Gendreau, M., Laporte, G., Potvin, J. Y., & Semet, F. (2002). A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, 53(5), 512-522.
6. Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, 31(12), 1985-2002.
7. Cordeau, J. F., & Laporte, G. (2005). Tabu search heuristics for the vehicle routing problem. In *Metaheuristic Optimization via Memory and Evolution* (pp. 145-163). Springer.

- 
8. Bell, J. E., & McMullen, P. R. (2004). Ant colony optimization techniques for the vehicle routing problem. *Advanced Engineering Informatics*, 18(1), 41-48.
  9. Osman, I. H. (1993). Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research*, 41(4), 421-451.
  10. Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2013). Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. *European Journal of Operational Research*, 231(1), 1-21.