
Manual de Usuario

Proyecto: Sistema de Optimización de Rutas de Transporte

Integrantes : Andres Felipe Carrasquilla Gutierrez
Yohan Stevens Piñarte Diaz
Jhon Franklin Sandoval Segura



Manual de Usuario Optimización de Rutas

0.3
Pág. 2

Contenido

<u>CONTENIDO</u>	2
<u>1 INTRODUCCIÓN</u>	4
<u>1.1 Propósito</u>	4
<u>1.2 Alcance</u>	4
<u>2 DESCRIPCIÓN GENERAL</u>	4
<u>2.1 Funcionalidad del producto</u>	4
<u>2.2 Requisitos del sistema</u>	4
<u>3.1 GUÍA DE FUNCIONES Y PROCEDIMIENTOS</u>	8
<u>3.1 Área principal</u>	8
<u>3.2 Configuración del problema</u>	8
<u>3.3 Selección de rutas oficiales</u>	8
<u>3.4 Gestión de demanda por paradero</u>	8
<u>3.5 Ejecución de optimización</u>	8
<u>3.6 Visualización paso a paso</u>	9
<u>4 APÉNDICES</u>	10
<u>4.1 Glosario de términos</u>	8

1 Introducción

Este manual explica cómo usar el sistema de optimización de rutas de buses universitarios. Cubre la interfaz web, configuración de parámetros, selección de rutas, gestión de demanda y análisis de resultados optimizados. Está dirigido a planificadores de rutas y administradores de transporte universitario.

1.1 Propósito

Este manual está dirigido a:

- Planificadores de transporte: Configurar y optimizar rutas de buses
- Administradores universitarios: Analizar eficiencia de rutas existentes
- Coordinadores de transporte: Planificar rutas considerando demanda estudiantil

Qué se espera que aprendas:

- Configurar parámetros de optimización
- Seleccionar rutas oficiales a combinar
- Establecer demanda estudiantil por paradero
- Ejecutar algoritmos de optimización
- Interpretar métricas y resultados
- Analizar el proceso de optimización paso a paso

1.2 Alcance

- *El Sistema de Optimización de Rutas Unillanos es una herramienta web que permite optimizar rutas de buses universitarios usando diferentes algoritmos metaheurísticos (Greedy, Algoritmo Genético, Recocido Simulado). Genera rutas eficientes considerando distancia, capacidad del bus y demanda estudiantil.*

2 Descripción general

2.1 Funcionalidad del producto

- Optimización multi-método: Tres algoritmos de optimización (Greedy, GA, SA)
- Visualización en mapa: Muestra rutas oficiales y optimizadas en tiempo real
- Gestión de demanda: Configuración flexible de estudiantes por paradero
- Métricas en tiempo real: Distancia, tiempo, capacidad, fitness
- Análisis paso a paso: Detalle completo del proceso de optimización
- Rutas oficiales integradas: Base de datos de rutas existentes del boletín

2.2 Requisitos del sistema

- Navegador web moderno (Chrome, Firefox, Safari, Edge)
- Conexión a Internet para cargar mapas y ejecutar optimizaciones
- Resolución recomendada: 1280x720 o superior
- No requiere creación de cuenta - acceso directo

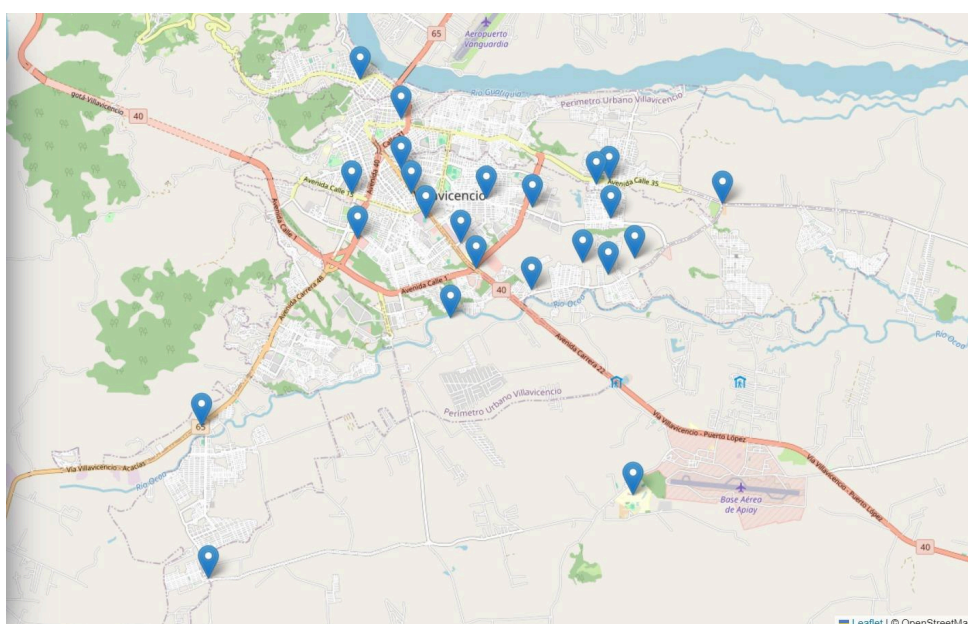
3 GUÍA DE FUNCIONES Y PROCEDIMIENTOS

3.1 Área principal

Propósito: Interfaz central con mapa interactivo y panel de control.

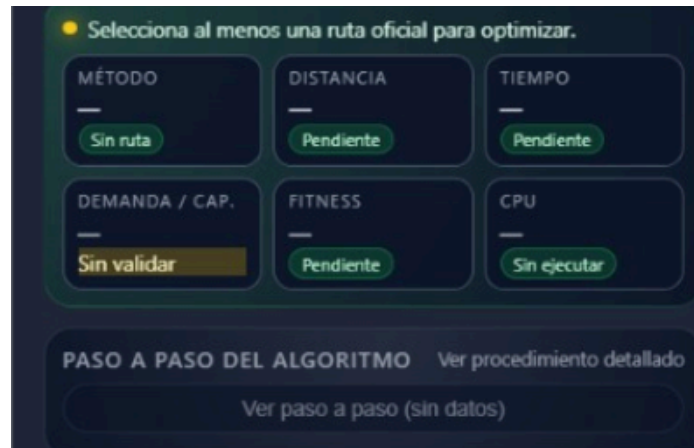
Componentes:

Mapa interactivo: Muestra paraderos y rutas



Panel de control lateral: Configuración y resultados

DEMANDA POR PARADERO	N° de estudiantes en cada punto
(VALLES_CAROLINA_20SUR)	
Las Gaviotas (Calle 20 Sur) (GAVIOTAS_20SUR)	0
Acapulco (Calle 20 Sur) (ACAPULCO_20SUR)	0
Marco Antonio Pinilla (MARCO_ANTONIO_PINILLA)	0
Santa Catalina (SANTA_CATALINA)	0
Porfía (PORFIA)	0



3.2 Configuración del problema

Propósito: Definir método de optimización y capacidad del bus.

Procedimiento:

En "Configuración del problema", selecciona el método:

Greedy: *Rápido para soluciones básicas*

Genetic Algorithm: *Balance entre velocidad y calidad*

Simulated Annealing: *Mayor precisión, más tiempo*

Define la capacidad del bus (estudiantes)

Haz clic en "Optimizar ruta" para ejecutar

3.3 Selección de rutas oficiales

Propósito: Elegir qué rutas oficiales incluir en la optimización.

Procedimiento:

En "Rutas oficiales (boletín)", marca las rutas a combinar

Usa "Marcar todo" / "Desmarcar todo" para selección rápida

Las rutas seleccionadas se muestran inmediatamente en el mapa

Nota: Debes seleccionar al menos una ruta para poder optimizarla.

3.4 Gestión de demanda por paradero

Propósito: Establecer número de estudiantes en cada paradero.

Procedimiento:

En "Demanda por paradero", busca el paradero deseado

Ingresa el número de estudiantes esperados

Repite para todos los paraderos relevantes

Los paraderos con demanda 0 no se considerarán

Recomendación: Actualiza la demanda según horarios pico y valle.

3.5 Ejecución de optimización

Propósito: Ejecutar el algoritmo seleccionado y generar rutas optimizadas.

Procedimiento:

1. Verifica que todos los parámetros estén configurados
2. Haz clic en "Optimizar ruta"
3. Espera el procesamiento (segundos a minutos según método)
4. Las nuevas rutas aparecen en verde en el mapa
5. Revisa las métricas en el panel de resultados

Estado durante ejecución:

- Verde: Listo/Optimizado
- Amarillo: Procesando
- Rojo: Error

3.6 Visualización paso a paso

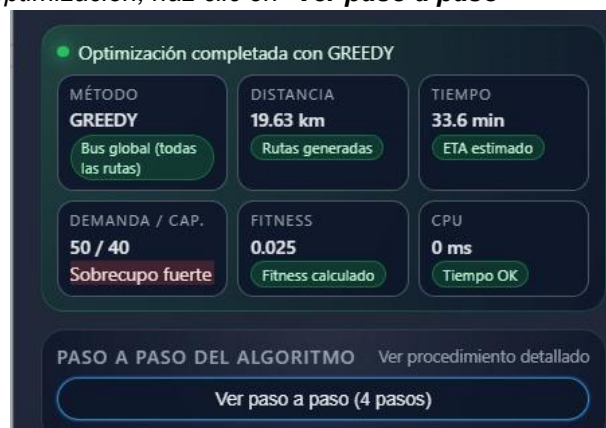
Propósito: Analizar el proceso detallado de optimización.

Procedimiento:

Después de Entrar en la página y seleccionar el algoritmo y ,las rutas a analizar seleccionamos **Optimizar ruta**

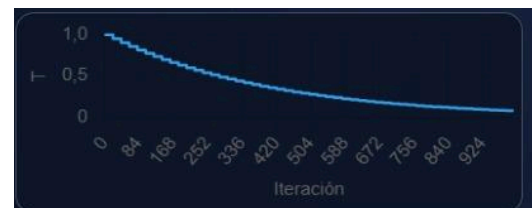
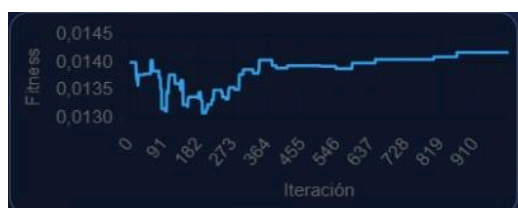


Después de una optimización, haz clic en **"Ver paso a paso"**



Se abre una ventana con:

Gráficas de evolución del algoritmo



Lista detallada de cada paso

- Fórmulas y cálculos utilizados
- Explicaciones del proceso

```
[SA · iter 0] initial_greedy (mejor)
UNILLANOS → VALLES_CAROLINA_20SUR → ACAPULCO_20SUR → MARCO_ANTONIO_PINILLA → KIRPA_20SUR → CAMPANARIO
→ TERMINAL → COVISAN → HATO_GRANDE → CENTRO → PARQUE_ESTUDIANTES → POSTOBON → VIVA → VILLACENTRO →
LOS_MARACOS → GRAMA → LA_ROCHELA → PORFIA → LA_MADRID → UNILLANOS
dist=50.45 km · obj=70.45 · fitness=0.0140 · T=1.000

Solución inicial Greedy. Obj = 70.447 km-equivalentes.

CÁLCULOS DETALLADOS

Distancia total:
d(UNILLANOS,VALLES_CAROLINA_20SUR) = 4.957 + d(VALLES_CAROLINA_20SUR,ACAPULCO_20SUR) = 0.631 +
d(ACAPULCO_20SUR,MARCO_ANTONIO_PINILLA) = 1.166 + d(MARCO_ANTONIO_PINILLA,KIRPA_20SUR) = 1.039 + d(KIRPA_20SUR,CAMPANARIO) = 2.301
+ d(CAMPANARIO,TERMINAL) = 4.241 + d(TERMINAL,COVISAN) = 1.900 + d(COVISAN,HATO_GRANDE) = 1.754 + d(HATO_GRANDE,CENTRO) = 1.112 +
d(CENTRO,PARQUE_ESTUDIANTES) = 0.599 + d(PARQUE_ESTUDIANTES,POSTOBON) = 0.648 + d(POSTOBON,VIVA) = 1.587 + d(VIVA,VILLACENTRO) =
1.015 + d(VILLACENTRO,LOS_MARACOS) = 1.408 + d(LOS_MARACOS,GRAMA) = 1.236 + d(GRAMA,LA_ROCHELA) = 5.707 + d(LA_ROCHELA,PORFIA) =
6.077 + d(PORFIA,LA_MADRID) = 3.410 + d(LA_MADRID,UNILLANOS) = 9.660 = 50.447 km

Demanda:
dem(VALLES_CAROLINA_20SUR) = 0 + dem(ACAPULCO_20SUR) = 0 + dem(MARCO_ANTONIO_PINILLA) = 0 + dem(KIRPA_20SUR) = 0 + dem(CAMPANARIO)
= 0 + dem(TERMINAL) = 0 + dem(COVISAN) = 0 + dem(HATO_GRANDE) = 0 + dem(CENTRO) = 0 + dem(PARQUE_ESTUDIANTES) = 0 + dem(POSTOBON) =
0 + dem(VIVA) = 50 + dem(VILLACENTRO) = 0 + dem(LOS_MARACOS) = 0 + dem(GRAMA) = 0 + dem(LA_ROCHELA) = 0 + dem(PORFIA) = 0 +
dem(LA_MADRID) = 0 = 50

Exceso de capacidad:
max(0, 50 - 40) = 10

Penalización:
10 * 2.0 = 20.000 km

Función objetivo:
50.447 + 20.000 = 70.447 km

Fitness:
1 / (1 + 70.447) = 0.013996

Δ de costo (SA):
Paso inicial: se parte de la solución Greedy.
No hay vecino que comparar, por lo tanto Δ = 0.

Prob. de aceptación (SA):
T0 = 1.0000.
La solución inicial se acepta por definición (probabilidad = 1).
```

Uso recomendado:

Administradores: Para validar y justificar decisiones

4 Apéndices

4.1 Glosario de términos

A

Algoritmo Genético (GA): Método de optimización inspirado en la evolución natural que usa selección, cruce y mutación.

C

Capacidad: Número máximo de estudiantes que puede transportar un bus.

D

Demanda: Número de estudiantes que requieren transporte en un paradero específico.

Distancia objetivo: Distancia total incluyendo penalizaciones por sobrecupo.

F

Fitness: Medida de calidad de una solución (mayor valor indica mejor solución).

G

Greedy: Algoritmo que toma decisiones óptimas locales en cada paso.

M

Metaheurística: Técnica avanzada de optimización para problemas complejos.

O

Optimización: Proceso de encontrar la mejor solución según criterios definidos.

P

Paradero: Punto específico de recogida o dejada de estudiantes.

Penalización: Kilómetros virtuales añadidos por exceder la capacidad del bus.

R

Recocido Simulado (SA): Algoritmo inspirado en procesos térmicos que permite soluciones temporalmente peores.

Ruta optimizada: Trayecto mejorado calculado por los algoritmos.

Ruta oficial: Ruta existente publicada en el boletín universitario.

S

Sobrecupo: Situación donde la demanda excede la capacidad del bus.

U

Unillanos: Universidad de los Llanos, sede Barcelona.