Justificación Detallado de Algoritmos Implementados en el Sistema de Transporte

Introducción

El sistema desarrollado integra diversos algoritmos fundamentales de teoría de grafos con el fin de optimizar, analizar y visualizar la red de transporte urbano (TransMilenio, SITP y Metro). Cada algoritmo fue seleccionado debido a su idoneidad para modelar fenómenos específicos de movilidad: rutas mínimas, congestión, conectividad estructural y asignación de recursos. A continuación se presenta una explicación detallada del porqué y cómo funciona cada uno.

1. Algoritmo de Dijkstra: Rutas Más Rápidas

Motivación

El problema a resolver es:

"¿Cuál es la ruta que toma menos tiempo entre dos estaciones?"

La red se modela como un grafo donde las estaciones son nodos y las conexiones entre ellas son aristas ponderadas por el tiempo estimado de desplazamiento. Dado que los pesos son positivos, el algoritmo óptimo es Dijkstra.

Funcionamiento

1. Se inicializan todas las distancias como infinitas, excepto el nodo origen.
2. Se utiliza una cola de prioridad para elegir el nodo con menor distancia acumulada.
3. Para cada vecino, se relaja la arista actualizando la distancia mínima conocida.
4. Cuando se alcanza el destino, la ruta obtenida es óptima.

Ventajas

- Alta eficiencia: $O(E \log V)$.
- Ideal para rutas en tiempo real.
- Puede adaptarse a múltiples criterios (tiempo, distancia, transbordos).

2. Edmonds–Karp: Análisis de Flujos Máximos y Congestión

Motivación

Este algoritmo resuelve el problema:

> *"¿Cuál es la máxima cantidad de pasajeros que puede moverse entre dos estaciones sin saturar la red?"*

Cada arista representa una capacidad (pasajeros por hora), lo que permite detectar cuellos de botella y evaluar congestión en horas pico.

Funcionamiento

Edmonds–Karp es una implementación de Ford–Fulkerson utilizando BFS:

1. La red se modela con capacidades en cada arista.
2. Se busca un camino aumentante usando BFS.
3. Se envía el flujo máximo posible por ese camino.
4. Se actualiza la red residual.
5. El proceso continúa hasta que no existan caminos aumentantes.

El resultado incluye el flujo máximo total y las aristas saturadas (cuellos de botella).

Ventajas

- Modelo realista de congestión.
- Permite análisis del tipo "what if".
- Identifica tramos críticos para priorizar mejoras.

3. Kruskal: Árbol de Recubrimiento Mínimo (MST)

Motivación

El objetivo del MST es:

> *"¿Cuál es la manera más eficiente de conectar todas las estaciones usando la mínima distancia posible?"*

Esto permite analizar conectividad estructural, planificar expansiones y detectar redundancias en la red.

Funcionamiento

1. Se ordenan todas las aristas por peso ascendente.
2. Se recorren en orden, añadiendo cada arista si no forma un ciclo.
3. Para verificar ciclos se usa la estructura **Union–Find (DSU)**.
4. Se detiene cuando hay $V - 1$ aristas en el árbol.

Ventajas

- Complejidad: $O(E \log E)$.
- Excelente rendimiento para grafos grandes y dispersos.
- Útil para evaluar estructura y conectividad del sistema.

4. Algoritmo Greedy de Coloreado: Asignación de Recursos

Motivación

El coloreado de grafos responde a:

> *"¿Cuál es el mínimo número de recursos (frecuencias, equipos, horarios) para que estaciones adyacentes no compartan el mismo recurso?"*

Dado que el problema óptimo es NP-duro, se utiliza una heurística rápida y eficiente.

Funcionamiento

1. Se ordenan los nodos por grado (algoritmo de Welsh–Powell).
2. Para cada nodo se bloquean los colores usados por sus vecinos.
3. Se asigna el color válido más pequeño disponible.
4. Se continúa hasta colorear todo el grafo.

Ventajas

- Extremadamente rápido en grafos grandes.
- Permite visualizar clusters y zonas conflictivas.
- Apoya la planificación de frecuencias y recursos.

5. Comparación General

Algoritmo	**Problema**	**Tipo**	**Complejidad**	**Uso**
Dijkstra	Camino mínimo	Optimización	$O(E \log V)$	Rutas rápidas
Edmonds–Karp	Flujo máximo	Optimización	$O(VE^2)$	Congestión
Kruskal	MST	Estructural	$O(E \log E)$	Conectividad
Greedy Coloring	Coloreado	Heurístico	$O(V^2)$	Recursos

6. Conclusión

Los algoritmos seleccionados permiten modelar de manera integral el funcionamiento de una red de transporte urbano. Cada uno contribuye a resolver un aspecto diferente del problema: rutas óptimas, capacidad máxima, análisis estructural y asignación de recursos. En conjunto, ofrecen una solución sólida, eficiente y fundamentada matemáticamente para la planificación y análisis del sistema.