



ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE CIENCIAS

MODELOS INTEGRADOS DE OPTIMIZACIÓN DE TRANSPORTE PÚBLICO

PROBLEMA DE ASIGNACIÓN DE VEHÍCULOS INTEGRANDO LA CALENDARIZACIÓN DE VIAJES

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO
MATEMÁTICO**

MAURICIO ESTEBAN YÉPEZ BAÑO

mauricio.yepez@epn.edu.ec

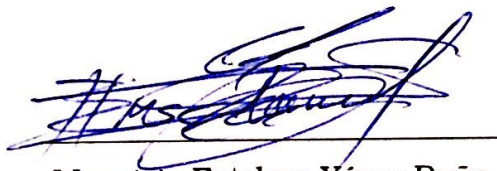
DIRECTOR: LUIS MIGUEL TORRES CARVAJAL

luis.torres@epn.edu.ec

DMQ, SEPTIEMBRE 2022

CERTIFICACIONES

Yo, MAURICIO ESTEBAN YÉPEZ BAÑO, declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.



Mauricio Esteban Yépez Baño

Certifico que el presente trabajo de integración curricular fue desarrollado por Mauricio Esteban Yépez Baño, bajo mi supervisión.



Luis Miguel Torres Carvajal

DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el(los) producto(s) resultante(s) del mismo, es(son) público(s) y estará(n) a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

Mauricio Esteban Yépez Baño

Luis Miguel Torres Carvajal

Agradecimientos

A mi familia, en especial mis hermanos y primos, quienes han sido un pilar fundamental en mi formación y por quienes he podido culminar esta etapa de mi vida.

A mi tutor, Luis Miguel Torres, quien me permitió iniciar en esta maravillosa rama de las matemáticas aplicadas y cuyos consejos me han hecho un mejor profesional. Por su guía, soporte, paciencia y esfuerzo en la realización y revisión de este trabajo.

A mis buenos amigos, Mesías, Henry, Christopher, David, Danny, entre otros, quienes en nada aportaron, pero cuya presencia me ha brindado experiencias que nunca se olvidarán y que sin su amistad este recorrido no hubiera sido el mismo.

Dedicatoria

A mi madre, mi mayor motivación en la vida, cuyo aliento y apoyo en los momentos difíciles me han permitido y motivado a seguir aprendiendo y mejorando, y sin los cuales nunca hubiera alcanzado mi proficiencia en inglés. Por no haberse rendido nunca. Por brindarme el amor y paciencia que me mantuvieron de pie a lo largo de mi trayecto universitario. Por brindarme una serie de valores y aptitudes que apreciaré y seguiré a lo largo de mi vida.

Gracias por luchar cada día para hacer de mí una mejor persona.

RESUMEN

Este trabajo se enfoca en explorar el problema de asignación de vehículos, el cual constituye una etapa fundamental en la planificación de los sistemas de transporte público. Se abordan conceptos, tanto generales como específicos, para la formulación del problema de asignación de vehículos, así como su relación con otras fases de la planificación de un sistema de transporte público. De forma específica, se analiza la relación con el problema de calendarización de viajes. Se revisan dos modelos de programación lineal entera obtenidos de la literatura, los cuales son implementados utilizando el API Python del solver Gurobi, considerando siempre una flota homogénea de vehículos y un sólo depósito. El primer modelo usa un enfoque de problema de flujo para asignar vehículos a rutas de costo mínimo. El segundo modelo trata el problema integrado de calendarización de viajes y asignación de vehículos. Los modelos fueron probados sobre seis instancias: tres redes de transporte público pequeñas, una mediana y dos grandes. Se reportan los resultados de los experimentos computacionales, las conclusiones correspondientes y las perspectivas de trabajo futuro.

Palabras clave: Problema de asignación de vehículos, calendarización de viajes, flota homogénea, optimización del transporte público, programación lineal entera, investigación de operaciones.

ABSTRACT

This work focuses on exploring the vehicle scheduling problem, which is an essential step in public transportation planning systems. General and specific concepts for formulating the vehicle scheduling problem are studied, as well as its relation to other phases in public transportation planning. Specifically, the relation with the timetabling problem is analysed. Two integer linear programming models for the problem are introduced and implemented using the Python API of the solver Gurobi, considering a homogeneous fleet and a single depot. The first model uses a flow approach to assign vehicles to minimum cost routes. The second one addresses the integrated timetabling and vehicle scheduling problems. These models have been tested on six instances: 3 small public transportation networks, 1 middle-sized, and 2 large ones. The results of the computational experiments, the corresponding conclusions, and the perspectives for future work are presented.

Keywords: Vehicle Scheduling problem, Timetabling, homogeneous fleet, optimization of public transportation systems, linear integer programming, Operations Research.

Índice general

1. Introducción	1
2. Marco Teórico	5
2.1. Estado del arte	5
2.2. Preliminares	7
2.3. Calendarización de viajes	11
2.4. Asignación de Vehículos	12
2.5. Integración de Calendarización de Viajes y Asignación de Vehículos	22
3. Formulación de Modelos para el VSP	25
3.1. Modelización como problema de flujo de costo mínimo	25
3.2. Modelo integrado TimVeh	28
4. Resultados Computacionales	33
4.1. Descripción de las Instancias	34
4.1.1. Small	35
4.1.2. Toy0	36
4.1.3. Toy1	37
4.1.4. Toy2	38
4.1.5. Grid	39

4.1.6. Athens	41
4.2. Minimización del tamaño de la flota	43
4.3. Resultados del Modelo Integrado	46
4.3.1. Configuración de parámetros del solver	46
4.3.2. Solución de las instancias	48
4.4. Comparación entre componentes de la función objetivo	50
5. Conclusiones	53
A. Redes evento-actividad para las redes de transporte público	57
A.1. Small	57
A.2. Toy0	58
A.3. Toy	58
A.4. Toy2	59
A.5. Grid	60
A.6. Athens	61
Bibliografía	62

Índice de figuras

1.1. Orden usual de las etapas de planificación de un sistema de transporte público: diseño de la red, asignación de pasajeros, planificación de líneas, enrutamiento de pasajeros, calendarización de viajes, asignación de vehículos y asignación de personal. Fuente: Imagen adaptada de [25].	3
2.1. Red de transporte público. Fuente: Adaptado de [25].	8
2.2. Ejemplo de plan de líneas simple, el cual consiste de dos líneas. Línea 1: camino de líneas punteadas (l_1); línea 2: camino de líneas entrecortadas (l_2); cada una con frecuencia 1. Fuente: [25].	9
2.3. EAN para el plan de líneas de la Figura 2.2. Los eventos de llegada son de color gris y los eventos de salida son de color blanco. Los arcos sólidos representan actividades de conducción, los arcos punteados representan actividades de espera y los arcos entrecortados son actividades de transferencia. Fuente: [25].	11
2.4. Red de transporte público con plan de 2 líneas. Fuente: [25].	15
2.5. Viajes programados a cubrir. Fuente: Adaptado de [25].	16

2.6. Viajes compatibles y viajes vacíos. Las relaciones de compatibilidad entre viajes se representan por medio de arcos de líneas entrecortadas. De entre todos estos, se han seleccionado aquellos en color negro como viajes vacíos en las rutas de servicio de las unidades de la flota. Fuente: Adaptado de [25].	18
2.7. Terminales de 4 ciudades entre las cuales se realizan viajes. Fuente: Basado en los datos de [18].	19
2.8. Líneas en una ciudad con tipografía de grilla. Fuente: [10].	19
4.1. PTN de la instancia Small.	35
4.2. PTN de la instancia Toy0.	36
4.3. PTN de la instancia Toy1.	37
4.4. PTN de la instancia Toy2.	38
4.5. PTN de la instancia Grid.	40
4.6. PTN de la instancia Athens.	42
A.1. EAN de la instancia Small.	57
A.2. EAN de la instancia Toy0.	58
A.3. EAN de la instancia Toy.	58
A.4. EAN de la instancia Toy2.	59
A.5. EAN de la instancia Grid.	60
A.6. EAN de la instancia Athens.	61

Capítulo 1

Introducción

El incremento permanente de la población en las ciudades hace que tener un sistema de transporte público sea fundamental para garantizar la adecuada movilidad de las personas y permitir que sus actividades se desarrollen con normalidad. El diseño y operación eficiente de estos sistemas constituyen retos inmensos para las grandes y medianas ciudades, donde las poblaciones crecen a un ritmo acelerado. Adicionalmente, este constante incremento de individuos ha causado un deterioro progresivo en la calidad de los servicios de transporte público. Una de las primeras ideas que pueden surgir para solucionar el problema de movilidad en una ciudad es aumentar la capacidad de la red vial, el número de vehículos, etc. Sin embargo, el aumento de estos elementos tiene asociado un costo económico que usualmente es muy elevado, y produce además un impacto negativo sobre el medio ambiente, por lo que no siempre es una solución viable ni deseable para hacer frente al ritmo de crecimiento de las poblaciones. Esto motiva el estudio de estrategias para optimizar el diseño y la operación de sistemas de transporte público que se enfoquen en proveer soluciones adecuadas de movilidad urbana a un costo razonable y minimizando la afectación al medio ambiente.

Diseñar un sistema de transporte público significa abordar una serie de problemas de resolución compleja como la planificación de líneas, la calendarización de viajes, la asignación de vehículos y personal, entre otros. En general, cada uno de los problemas anteriores son resueltos de

forma secuencial, y se los considera como fases dentro del diseño del sistema de transporte público. El objetivo de cada fase puede depender de diferentes enfoques. Por ejemplo, el diseño de las líneas y la calendarización de los viajes pretende usualmente mejorar (maximizar) la comodidad y calidad del servicio para los pasajeros, al reducir los tiempos de viaje promedio y el número de transferencias promedio en el sistema. Por otro lado, en la asignación de vehículos y la asignación de personal se pretende minimizar los costos asociados en los que incurre el operador del sistema, de modo que el uso de los recursos disponibles sea el más eficiente posible.

La planificación del transporte público ha sido un área ampliamente estudiada desde hace algunas décadas. El proceso de planificación se divide en distintos subproblemas consecutivos (ver Figura 1.1). En particular, el problema de asignación de vehículos (VSP), estudiado extensivamente en los últimos 40-50 años (ver [8, 4]), se enfoca en determinar rutas para los vehículos de la flota de transporte público, con el objetivo de cubrir los viajes programados dentro de un horizonte de tiempo minimizando los costos operacionales. Estas rutas pueden incluir desplazamientos vacíos para la conexión entre dos viajes consecutivos. Se busca asignar varios viajes a los vehículos de modo que se cumpla que cada viaje se cubre una sola vez y cada vehículo tiene asignado un conjunto secuencial de viajes que son factibles de recorrer.

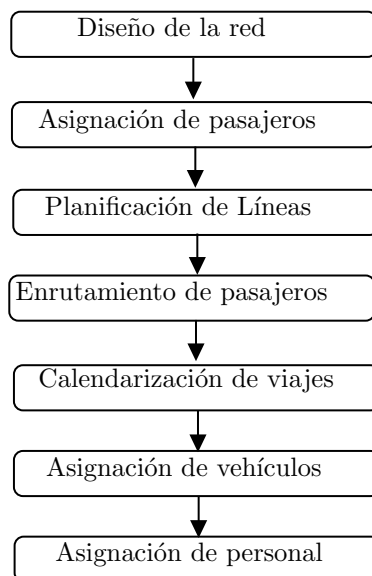


Figura 1.1: Orden usual de las etapas de planificación de un sistema de transporte público: diseño de la red, asignación de pasajeros, planificación de líneas, enrutamiento de pasajeros, calendarización de viajes, asignación de vehículos y asignación de personal. **Fuente:** Imagen adaptada de [25].

Existen en la literatura numerosos trabajos enfocados en el estudio de modelos de optimización para cada una de las fases del diseño del sistema de transporte público por separado, para lo cual se puede revisar [25]. Sin embargo, considerar los problemas por separado puede dar lugar a soluciones subóptimas tanto en lo que respecta a los recursos utilizados como a la comodidad del pasajero, pues cada fase depende de la anterior y por tanto una solución que parezca óptima en una fase puede que dé lugar a decisiones poco adecuadas en la fase siguiente. Es por esto que en los últimos años se han desarrollado trabajos enfocados en resolver de manera integrada dos o más fases de la planificación. Aquí, en particular, nos centraremos en una manera de resolver simultáneamente las fases de calendarización de viajes y asignación de vehículos, TTVS por sus siglas en inglés (TimeTabling y Vehicle Scheduling), ya que como se establece en [5] estas son las fases de mayor importancia en el transporte público.

En este trabajo se busca estudiar modelos de programación lineal entera para el problema de asignación de flotas de vehículos e implementarlos computacionalmente. De manera específica, se estudiará el pro-

blema de asignación de flotas de vehículos dentro la planificación de los sistemas de transportación pública, con y sin integración al problema de calendarización de viajes. Se formularán dos modelos de programación lineal entera para el problema de asignación de flotas de vehículos y se implementarán los modelos computacionalmente, utilizando como herramienta de programación el API Python del solver Gurobi. El primer modelo a implementar considerará todas las fases anteriores ya terminadas, y se enfocará en resolver exclusivamente el problema de asignación de vehículos. Por otra parte, el segundo modelo se enfocará en la solución simultánea de las fases de calendarización de viajes y asignación de vehículos. Finalmente, se probarán los modelos sobre distintas instancias y se analizarán los resultados.

El presente trabajo se estructura como sigue. En el Capítulo 2 se realiza una revisión de literatura sobre el problema de asignación de vehículos (VSP) y las nociones preliminares y conceptos necesarios para definir y tratar los problemas de calendarización de viajes (TT) como un problema periódico de calendarización de eventos, PESP por sus siglas en inglés (Periodic Event Schedule Problem), y de asignación de vehículos. En el Capítulo 3 se presenta un modelo para la resolución del VSP considerando las demás fases fijas y posteriormente un modelo TTVSP para la integración de las dos fases de calendarización de horarios y asignación de vehículos, combinando la periodicidad de los horarios y la aperiodicidad de los viajes. El Capítulo 4 se dedica a presentar los resultados de experimentos computacionales sobre instancias para ambos modelos, además de reportar su desempeño en cada una. Las instancias consideradas son variadas en tamaño y forma. Finalmente, en el Capítulo 5 se presentan conclusiones y desafíos futuros a enfrentar en trabajos posteriores.

Capítulo 2

Marco Teórico

2.1. Estado del arte

Existen diferentes enfoques para la resolución del problema de asignación de vehículos para los cuales se han propuesto distintos modelos. [4, 8] resumen el trabajo que ha existido sobre el problema, describen algunas variantes del mismo y presentan varios modelos para resolver el VSP.

Dentro de la clasificación de problemas de asignación de vehículos se pueden destacar dos principales: si existen uno o más tipos de vehículos y si se consideran uno o múltiples depósitos para la salida de vehículos.

Cuando se tiene un solo depósito y un solo tipo de vehículo el VSP es relativamente fácil de resolver en el sentido de que se conocen algoritmos que encuentran una solución óptima en tiempo polinomial. Este resultado se presenta en [1] y junto a [14] se indica que para el caso de varios depósitos el VSP es NP-difícil.

Considerando un sólo depósito, uno de los primeros acercamientos a la solución del problema fue presentado en [23], en donde la modelización se basa en el Teorema de Dilworth para conjuntos parcialmente ordenados. Por otro lado, [19] formula el problema como uno de asignación sobre un grafo bipartito, en donde tanto la reubicación y los costos correspondientes son considerados, mientras que en [20] una formula-

ción de flujo en redes es propuesta. Adicionalmente, en [21] se describe un VSP más realista, en cuya solución se utilizan técnicas de generación de columnas.

El caso de varios depósitos y varios tipos de vehículos también se presentan en [4] pero no se tratan aquí.

La calendarización de viajes, problema en el cual se establecen horarios para los distintos viajes programados en un sistema de transporte público también se ha estudiado de manera extensa. El caso que se aborda en este trabajo se refiere a horarios periódicos. Sin embargo, [16] ofrece una compilación de trabajos previos sobre la calendarización de viajes periódica y aperiódica. La calendarización es usualmente modelada por el problema periódico de asignación de eventos (PESP), el cual es NP-difícil ([28]).

En cuanto a la integración de los problemas de calendarización de viajes y asignación de vehículos (TTVSP), existen varios trabajos publicados que consideran variaciones respecto a la periodicidad de los problemas individuales. [6] propone un modelo de flujomultiproducto para el problema integrado, además de un método metaheurístico para resolverlo. [25] presenta un modelo de programación lineal entera y para resolver las fases de forma simultánea propone dos enfoques heurísticos. El más notable es el enfoque iterativo de re-optimización el cual consiste en, iterativamente, resolver una fase del problema de transporte público mientras las variables correspondientes a la otra fase se fijan. Esta manera de resolver el problema heurísticamente se conoce como el *eigen-model*, según se menciona en [26]. Recientemente, en [3] se propone un método heurístico que permite determinar una asignación de vehículos únicamente con un plan de líneas dado y a partir de esta se procede a calcular la calendarización respectiva. La implementación de una heurística específica para el problema integrado TTVSP está fuera del alcance de este trabajo de investigación.

2.2. Preliminares

Para un estudio más extendido sobre problemas combinatorios, grafos y algoritmos véase [27, 13].

Definición 1. Un grafo no dirigido $G = (V, E)$ se compone de un conjunto finito V de nodos o vértices y un conjunto $E \subset \{\{i, j\} : i, j \in V\}$ de aristas. Dada una arista $e := \{i, j\} \in E$, decimos que los nodos i y j son los extremos de e , y que e conecta a i con j .

Si se dota de una dirección a las aristas se tiene lo siguiente.

Definición 2. Un grafo dirigido o digrafo $D = (V, A)$ se compone de un conjunto finito V de nodos o vértices y un conjunto $A \subset V \times V$ de arcos. Dado un arco $a := (i, j) \in A$, decimos que i es el extremo inicial de a , j su extremo final, y que a conecta i con j .

Definición 3. Un sendero en un grafo no dirigido G es una sucesión alternada

$$P := (v_1, e_1, v_2, e_2, \dots, e_{n-1}, v_n)$$

de nodos $v_1, v_2, \dots, v_n \in V$ y aristas $e_1, e_2, \dots, e_{n-1} \in E$ que satisface que $e_i = \{v_i, v_{i+1}\}$, para todo $i \in \{1, \dots, n-1\}$.

Debido a que las aristas determinan un sendero, es común escribir $P = (e_1, e_2, \dots, e_{n-1})$. Se emplea $v \in P$ y $e \in P$ para indicar que un nodo v y una arista e pertenecen a un sendero P .

Definición 4. Un camino en un grafo dirigido D es una sucesión alternada

$$P := (v_1, a_1, v_2, a_2, \dots, a_{n-1}, v_n)$$

de nodos $v_1, v_2, \dots, v_n \in V$ y arcos $a_1, a_2, \dots, a_{n-1} \in A$ que satisface que $a_i = (v_i, v_{i+1})$.

Como los arcos determinan un camino, es común escribir $P = (a_1, a_2, \dots, a_{n-1})$. Se emplea $v \in P$ y $a \in P$ para indicar que un nodo v y un arco a pertenecen a un camino P . Además, un camino que no contiene nodos repetidos se llama simple.

Definición 5. Dados un grafo dirigido $D = (V, A)$ y un vector $\ell \in \mathbb{R}^A$ que asocia a cada arco $a \in A$ una longitud ℓ_a , la longitud de un camino $P := (v_1, a_1, v_2, a_2, \dots, a_{n-1}, v_n)$ se define como

$$\ell_P := \sum_{i=1}^{n-1} \ell_{a_i}.$$

Una definición similar se adopta para la longitud de un sendero en un grafo no dirigido.

Observación. De aquí en adelante la palabra *camino* se utilizará indistintamente para un sendero o para un camino, mientras del contexto se pueda deducir significado.

Definición 6. Un camino $s - t$ es un camino $P := (s, a_1, v_2, a_2, \dots, a_{n-1}, t)$ que empieza en un nodo $s \in V$ y termina en un nodo $t \in V$. Un camino más corto $s - t$ es un camino $s - t$ de longitud ℓ_P mínima.

Definición 7. Una red de transporte público (Public Transportation Network, PTN) es un grafo (V, E) cuyos nodos corresponden a estaciones o paradas del sistema, y cuyas aristas indican conexiones directas de transporte entre estaciones. Las aristas pueden o no tener dirección, es decir, de acuerdo a la aplicación, una red de transporte público puede referirse a un grafo o a un digrafo.

Un ejemplo de PTN puede encontrarse en la figura siguiente.

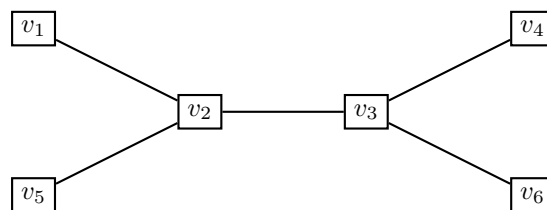


Figura 2.1: Red de transporte público. **Fuente:** Adaptado de [25].

Definición 8. Una matriz origen-destino [origin-destination matrix, OD-matrix] asociada a una red de transporte público (V, E) es una matriz cuadrada $C \in \mathbb{Z}^{V \times V}$ cuyos elementos $C_{u,v}$ indican la cantidad de pasajeros que viajan desde la estación u hasta la estación v del sistema dentro de un horizonte de tiempo.

La expresión $(u, v) \in C$ indicará que $C_{u,v} > 0$.

Definición 9. Una línea l es un camino simple en la red de transporte público que debe ser operada de inicio a fin por un mismo vehículo.

La operación de una línea conlleva el gasto de recursos económicos entre los cuales podemos destacar el salario del personal que opera la línea, el cual se considera fijo, y el costo del combustible que los vehículos utilizan en cada operación de la línea, el cual es variable. La suma de todos los valores en los que se incurra por utilizar la línea se conoce como el costo de la línea, el cual se denota por cost_l . La distancia total que implica recorrer la línea de inicio a fin se conoce como la longitud de la línea y se nota por $\text{length}_l = \sum_{e \in l} \text{length}_e$. Si se considera la planificación periódica del sistema de transporte público, cada línea l está conectada con una cantidad de viajes que se realizan dentro de un intervalo de tiempo dado, esta cantidad de viajes es llamada frecuencia f_l . Un concepto de líneas es un conjunto $\mathcal{L} = (\mathcal{L}^0, f)$ que asocia una frecuencia f_l a la que se opera cada línea $l \in \mathcal{L}^0$.

Definición 10. Para un concepto de líneas (\mathcal{L}^0, f) , el conjunto $\mathcal{L}' = \{l \in \mathcal{L}^0 \mid f_l > 0\}$ se conoce como plan de líneas.

Para la PTN de la figura 2.1, un plan de líneas puede ser el siguiente

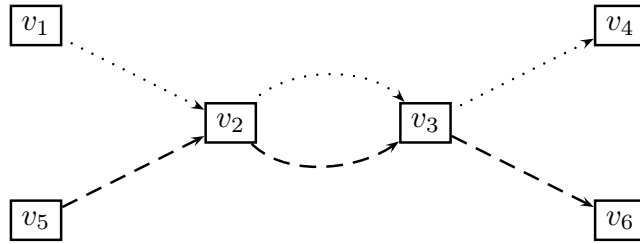


Figura 2.2: Ejemplo de plan de líneas simple, el cual consiste de dos líneas. Línea 1: camino de líneas punteadas (l_1); línea 2: camino de líneas entrecortadas (l_2); cada una con frecuencia 1. **Fuente:** [25].

Definición 11. Sea una (V, E) una PTN y \mathcal{L} un plan de líneas. Una Red Evento-Actividad (EAN), $\mathcal{N}^0 = (\mathcal{E}^0, \mathcal{A}^0)$ es un grafo dirigido con nodos \mathcal{E}^0 , llamados eventos, y arcos \mathcal{A}^0 , llamados actividades. Los nodos representan llegadas y salidas de vehículos en las estaciones mientras que los arcos

representan tres tipos de actividades que los pasajeros realizan durante su viaje: actividades de conducción, que corresponden al desplazamiento en un vehículo entre dos estaciones, actividades de espera por la llegada de los vehículos en las estaciones, y actividades de transferencia entre líneas. Formalmente, la EAN se construye de la siguiente forma:

$$\begin{aligned}
\mathcal{E}^0 &= \mathcal{E}_{arr} \cup \mathcal{E}_{dep} \text{ con} \\
\mathcal{E}_{arr} &= \{(v, \mathbf{arr}, l) : v \in l \cap V, l \in \mathcal{L}\} \\
\mathcal{E}_{dep} &= \{(v, \mathbf{dep}, l) : v \in l \cap V, l \in \mathcal{L}\} \\
\mathcal{A}^0 &= \mathcal{A}_{drive} \cup \mathcal{A}_{wait} \cup \mathcal{A}_{trans} \text{ con} \\
\mathcal{A}_{drive} &= \{((v_1, \mathbf{dep}, l), (v_2, \mathbf{arr}, l)) : e = (v_1, v_2) \in l \cap E, l \in \mathcal{L}\}, \\
\mathcal{A}_{wait} &= \{((v, \mathbf{arr}, l), (v, \mathbf{dep}, l)) : v \in l \cap V, l \in \mathcal{L}\}, \\
\mathcal{A}_{trans} &= \{((v, \mathbf{arr}, l_1), (v, \mathbf{dep}, l_2)) : v \in l : l_1 \cap l_2 \cap V, l_1, l_2 \in \mathcal{L}, l_1 \neq l_2\}
\end{aligned}$$

En la definición anterior se ha asumido que las frecuencias de cada línea del plan son todas iguales a uno. Cuando se tienen líneas con frecuencias superiores a 1, las mismas pueden ser reemplazadas por varias copias de líneas con frecuencias unitarias.

Para cada actividad $a \in \mathcal{A}^0$ se dan cotas inferiores y superiores para su duración. Esto es, cada actividad $a \in \mathcal{A}^0$ tiene asociada una duración d_a que cumple $0 \leq L_a \leq d_a \leq U_a$. Estas cotas vienen asociadas a la red de transporte público subyacente, es decir para la red de transporte público existen tiempos de viaje mínimos y máximos sobre las aristas de la red: $0 < L_e^{drive} \leq U_e^{drive}, e \in E$; tiempos de espera mínimos y máximos de los vehículos en cada estación de la red: $0 < L_v^{wait} \leq U_v^{wait}, v \in V$; y tiempos de trasbordo mínimos y máximos de los pasajeros en cada estación de la red: $0 < L_v^{trans} \leq U_v^{trans}, v \in V$.

Para el plan de líneas de la Figura 2.2, la correspondiente EAN es

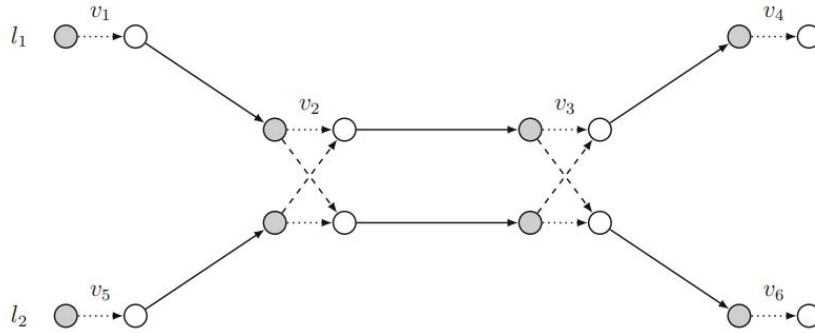


Figura 2.3: EAN para el plan de líneas de la Figura 2.2. Los eventos de llegada son de color gris y los eventos de salida son de color blanco. Los arcos sólidos representan actividades de conducción, los arcos punteados representan actividades de espera y los arcos entrecortados son actividades de transferencia. **Fuente:** [25].

2.3. Calendarización de viajes

En la calendarización de viajes (Timetabling) se asignan horarios fijos para cada uno de los viajes programados en el sistema, los mismos que consisten de tiempos exactos de llegada y salida a cada una de las estaciones.

Se distingue entre planificaciones de tipo periódica y aperiódica. En la primera los horarios se calculan para un periodo fijo de tiempo y luego se repiten múltiples veces a lo largo del día, o de una determinada franja horaria (por ejemplo, durante las horas pico o las horas valle). En cambio, en el caso aperiódico, los horarios se calculan para todo el horizonte de planificación (usualmente un día). Para este estudio consideraremos una planificación periódica.

El problema de calendarización de viajes se modela sobre la *red Evento-Actividad* (EAN), la cual se deriva de la PTN y el plan de líneas.

Definición 12. Sean $\mathcal{N}^0 = (\mathcal{E}^0, \mathcal{A}^0)$ una EAN con cotas $0 \leq L_a \leq U_a, a \in \mathcal{A}^0$ y T la longitud del periodo a considerar. Un horario periódico factible

$$\pi \in \{0, 1, \dots, T - 1\}^{\mathcal{E}^0}$$

asigna tiempos periódicos π_i a cada evento $i \in \mathcal{E}^0$ de modo que la duración de la actividad $a \in \mathcal{A}^0$ esté dentro del intervalo $[L_a, U_a]$ especificado por sus cotas inferior y superior, es decir, que se cumpla

$$L_a \leq (\pi_j - \pi_i - L_a) \bmod T + L_a \leq U_a,$$

para todas las actividades $a = (i, j) \in \mathcal{A}^0$.

El problema de la calendarización de viajes consiste en hallar un horario factible que minimice el tiempo de viaje ponderado con los pesos de los pasajeros sobre cada actividad

$$\min_{\{0,1,\dots,T-1\}^{\mathcal{E}^0}} \sum_{a=(i,j) \in \mathcal{A}^0} w_a \cdot ((\pi_j - \pi_i - L_a) \bmod T + L_a),$$

donde w_a es el número de pasajeros que participan en la actividad $a \in \mathcal{A}^0$, el cual se determina durante la fase de enrutamiento de pasajeros.

Mayores detalles sobre la calendarización de viajes y su asociación con fases previas como el enrutamiento de pasajeros pueden encontrarse en [25] y [28].

2.4. Asignación de Vehículos

El objetivo del problema de asignación de vehículos (VSP) es determinar rutas para los vehículos de la flota de transporte público para cubrir los viajes programados dentro de un horizonte de planificación y procurando minimizar los costos operativos asociados.

Algunos tipos de problemas de asignación de vehículos que se pueden encontrar en la literatura son los siguientes:

- **Periódicos [2].** En este tipo de VSP se obtienen soluciones para un período de tiempo fijo $T \in \mathbb{N}$ y esta solución se repite a lo largo de todo el horizonte de planificación (e.g. todo el día de servicio).
- **Aperiódicos [22, 17, 29].** Aquí la solución se la obtiene directamente para todo el horizonte de planificación.

- **Un depósito [23, 19, 20].** Se asume que todos los vehículos salen de y llegan a un único lugar de almacenamiento de la flota. Este caso es interesante pues para el mismo pueden obtenerse algoritmos de resolución polinomial.
- **Múltiples depósitos [1, 14, 11].** Aquí los vehículos inician sus viajes desde distintos lugares y pueden terminarlos en otros distintos. Un claro ejemplo de este tipo de problema es el enrutamiento de autobuses para cubrir viajes de larga distancia en distintas ciudades.
- **Un tipo de vehículo [1, 20].** En este caso toda la flota es homogénea, es decir, cada vehículo de la flota es idéntico, y por lo tanto es posible asignar cualquier viaje a cualquier vehículo.
- **Varios tipos de vehículos [9, 7, 15].** Aquí, la flota ya no es homogénea, es decir, es posible que cierto tipo de vehículo no pueda cubrir un determinado viaje, por lo que debe considerarse un criterio de compatibilidad entre viajes y vehículos al realizar la asignación.

En este trabajo se tratará el caso en el que se cumplen los siguientes supuestos:

1. El horario para la calendarización de los viajes es periódico.
2. Existe un sólo depósito.
3. La flota es homogénea.

Debido a que se quiere incorporar las reubicaciones de vehículos entre viajes y una evaluación de costos más sofisticada, abordaremos el caso aperiódico del VSP.

Además, se supone que la red de transporte público está ya diseñada, el plan de líneas y el enrutamiento de pasajeros es fijo y, en el primer modelo, que se han determinado los horarios periódicos para todos los viajes programados. Adicionalmente, supondremos que cada línea del plan tiene frecuencia igual a 1. (Las líneas con frecuencias mayores han sido reemplazadas por múltiples líneas con frecuencias unitarias.

Definición 13. Sean π un horario periódico con longitud de periodo T para el plan de líneas \mathcal{L} y \mathcal{P} el conjunto de periodos a considerar dentro del

horizonte de tiempo. Un viaje $t = (p, l)$ es la operación número p de la línea l . El conjunto de todos los viajes a realizar es $\mathcal{T} = \{(p, l) \mid p \in \mathcal{P}, l \in \mathcal{L}\}$.

Observación. Es fácil ver que por la definición anterior un viaje está determinado por

- La línea $l \in \mathcal{L}$ a la que corresponde;
- el periodo $p \in \mathcal{P}$ en que inicia;
- su tiempo de inicio s_{pl} ; y
- su tiempo de finalización e_{pl} .

Considerando que el horario de calendarización de los viajes es periódico se tienen ciertas particularidades respecto a los tiempos de inicio y fin de los viajes.

Proposición 1. Sean $\text{first}(l)$ y $\text{last}(l)$ el primer y último eventos que corresponden a la línea l en la EAN, respectivamente. Entonces para el viaje (p, l) se tiene que:

$$\begin{aligned} s_{p,l} \quad \text{mód } T &= \pi_{\text{first}(l)} \quad \text{y} \\ e_{p,l} \quad \text{mód } T &= \pi_{\text{last}(l)}. \end{aligned}$$

Esto quiere decir que un viaje $t = (p, l), p \in \mathcal{P}, l \in \mathcal{L}$, siempre iniciará al tiempo $\pi_{\text{first}(l)}$ en cada periodo en que se repita el horario periódico π . En particular, el tiempo exacto de inicio de cada viaje puede ser calculado como

$$s_{p,l} = p \cdot T + \pi_{\text{first}(l)}.$$

Notar, por otra parte, que el tiempo de finalización de un viaje no puede asociarse de la misma manera con $\pi_{\text{last}(l)}$, debido a que el viaje puede durar más que la longitud del periodo del horario. Por este motivo, el tiempo exacto de finalización se calcula a partir del tiempo de inicio y la duración del viaje como

$$e_{p,l} = p \cdot T + \pi_{\text{first}(l)} + d_l.$$

Ejemplo 1. Consideremos la siguiente red de transporte público con 6 estaciones y dos líneas, cada una de las cuales tiene frecuencia unitaria [25]. En la fase de calendarización de viajes se ha establecido un horario periódico con duración del período igual a $T = 60$. Para los eventos de la primera línea se tiene que $\pi_{\text{first}(l_1)} = 20$ y $\pi_{\text{last}(l_1)} = 0$, mientras que para la segunda se tiene que $\pi_{\text{first}(l_2)} = 0$ y $\pi_{\text{last}(l_2)} = 20$.

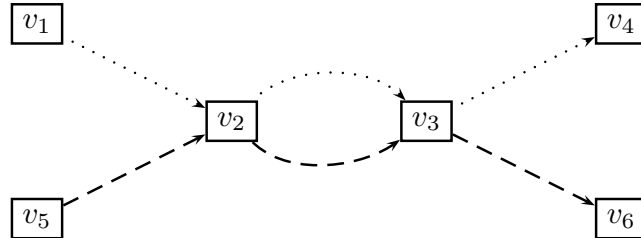


Figura 2.4: Red de transporte público con plan de 2 líneas. **Fuente:** [25].

Los viajes a cubrir para un horizonte de planificación que consiste de 4 períodos están dados en la Figura 2.5.

Se observa que los viajes de la línea uno inician a los 20 minutos de iniciado cada periodo, lo cual concuerda con la condición $\pi_{\text{first}(l_1)} = 20$. De la misma manera, se tiene que terminan al final de cada periodo el cual coincide con el inicio del periodo siguiente: $\pi_{\text{last}(l_1)} = 0$. Ocurre algo similar para la segunda línea.

Para poder definir el concepto fundamental de compatibilidad entre viajes se necesitan introducir previamente otros términos y notaciones.

Definición 14. La longitud de un camino más corto P_{v_1, v_2} entre dos estaciones v_1, v_2 de una PTN se conoce como la distancia entre v_1 y v_2 y se notará por dist_{v_1, v_2} . El tiempo necesario para recorrer P_{v_1, v_2} será notado como time_{v_1, v_2} .

En este sentido, D_{l_1, l_2} denotará la distancia entre la última estación v'_1 de la línea l_1 y la primera estación v_2 de la línea l_2 , es decir, $D_{l_1, l_2} = \text{dist}_{v'_1, v_2}$. De manera similar, $L_{l_1, l_2} = \text{time}_{v'_1, v_2} + L_{\text{turn}}$ denotará el tiempo necesario entre la finalización de la operación de la línea l_1 y el inicio de la operación de la línea l_2 , donde L_{turn} representa el tiempo mínimo requerido para preparar a un vehículo antes de iniciar un viaje.

De forma análoga, se escribirá $D_{\text{dep},l}$ para representar la distancia del depósito hacia la primera estación de la línea $l \in \mathcal{L}$ y $L_{\text{dep},l}$ el tiempo necesario para realizar este recorrido; así como $D_{l,\text{dep}}$, $L_{l,\text{dep}}$ serán la distancia y tiempo necesarios para ir de la última estación v' de la línea l hacia el depósito.

Observación. Así definida, $\text{dist}_{\bullet,\bullet}$ satisface la desigualdad triangular.

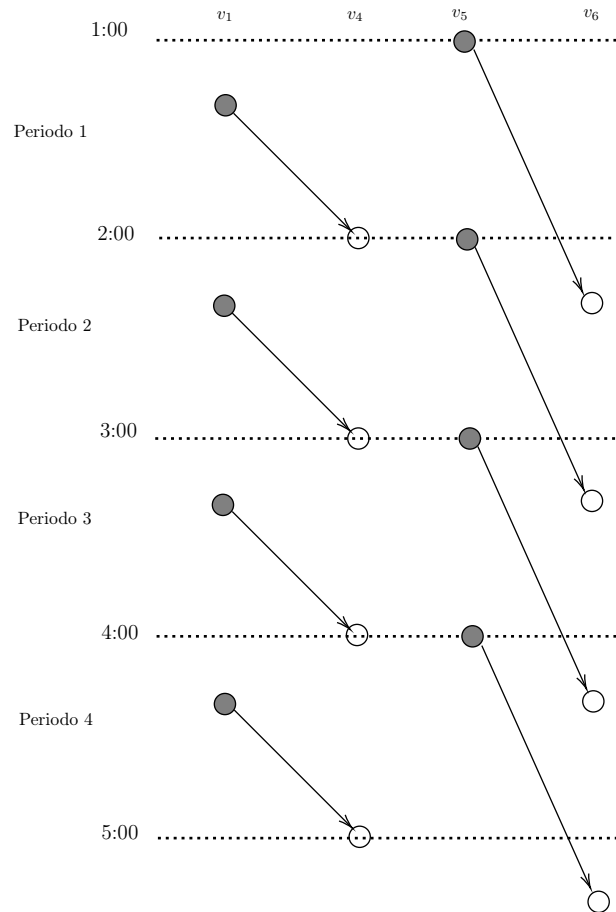


Figura 2.5: Viajes programados a cubrir. **Fuente:** Adaptado de [25].

Para poder definir los costos operativos necesitamos introducir la siguiente notación:

Definición 15. Para un viaje (p, l) , su longitud se representa por $\text{length}_{p,l} = \text{length}_l$ y su duración por $d_{p,l} = d_l = e_{p,l} - s_{p,l}$. El recorrido que un vehículo realiza para conectar dos viajes consecutivos se llama viaje de conexión y se lo representa por $c = ((p_i, l_i), (p_j, l_j))$, con $(p_i, l_i), (p_j, l_j) \in \mathcal{T}$. La longitud de un viaje de conexión se notará como $\text{length}_c = D_{l_i, l_j}$ y su duración $d_c = s_{p_j, l_j} - e_{p_i, l_i}$.

De forma similar, se llaman viajes de depósito a los recorridos que un vehículo realiza desde el depósito hasta la primera estación del primer viaje en su ruta de servicio, cuya longitud y duración son $length_{dep,l} = D_{dep,l}$ y $d_{dep,l} = L_{dep,l}$, respectivamente; y a los recorridos desde la última estación del último viaje de su ruta de servicio hacia el depósito con longitud y duración $length_{l,dep} = D_{l,dep}$, $d_{l,dep} = L_{l,dep}$, respectivamente.

Observación. Los viajes de conexión y de depósito son llamados *viajes vacíos* ya que estos no son utilizados por los pasajeros.

Definición 16. El viaje $(p_1, l_1) \in \mathcal{T}$ es compatible con el viaje $(p_2, l_2) \in \mathcal{T}$ si

$$s_{p_2,l_2} - e_{p_1,l_1} \geq L_{l_1,l_2}.$$

Notaremos esto por:

$$(p_1, l_1) \vdash (p_2, l_2).$$

Observación. La relación \vdash es antisimétrica.

Ejemplo 2. Para la PTN del Ejemplo 1, los posibles viajes compatibles se representan a continuación. Así, por ejemplo, el viaje $t_1 = (1, l_1)$ es compatible con el último viaje de la línea dos $t_2 = (4, l_2)$ pues existe tiempo suficiente para que un vehículo que completa t_1 llegue a cubrir t_2 . Sin embargo, $t_3 = (1, l_2)$ no es compatible con el viaje $t_4 = (2, l_2)$ pues cuando un vehículo termina t_3 , t_2 empezó hace 20 minutos.

Observación. La Definición 16 constituye el criterio más simple para definir la compatibilidad entre viajes. A menudo, esta definición puede y debe extenderse para incluir las características particulares de cada aplicación, como se indica en los dos ejemplos siguientes.

Ejemplo 3. Supongamos que se requiere realizar la asignación de vehículos para que una flota de transportes interprovinciales cubra los viajes programados entre cuatro ciudades A, B, C, D en el horizonte de tiempo de una semana. En la Figura 2.7 se representan las terminales de cada ciudad como nodos de un grafo y los viajes programados como arcos bidireccionales.

Si un vehículo está disponible en la mañana en el terminal A, no es prudente enviarlo sin pasajeros a cubrir un viaje en otra ciudad C en la

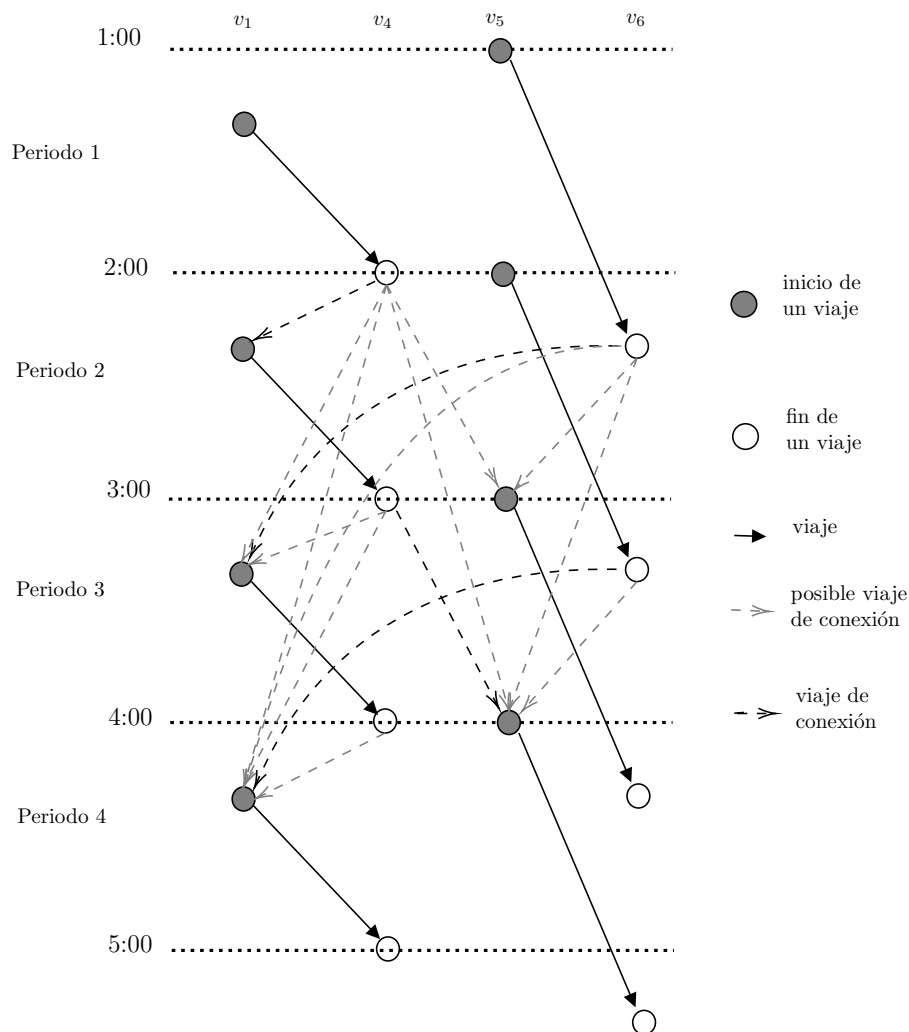


Figura 2.6: Viajes compatibles y viajes vacíos. Las relaciones de compatibilidad entre viajes se representan por medio de arcos de líneas entrecortadas. De entre todos estos, se han seleccionado aquellos en color negro como viajes vacíos en las rutas de servicio de las unidades de la flota. **Fuente:** Adaptado de [25].

noche, incluso así los tiempos de conexión lo permitan, pues desplazar un vehículo vacío entre dos ciudades es demasiado costoso. Por lo tanto, en este caso el criterio de compatibilidad entre viajes debe especificar, adicionalmente, que la última estación del primer viaje debe coincidir con la primera estación del segundo viaje.

Ejemplo 4. Ahora, supongamos una ciudad con una topografía de grilla rectangular. El sistema de transporte público emplea buses de una flota homogénea, y cada punto de la grilla corresponde a una estación v_{ij} , con $1 \leq i, j \leq 5$ del sistema. Las líneas se especifican en la figura 2.8.

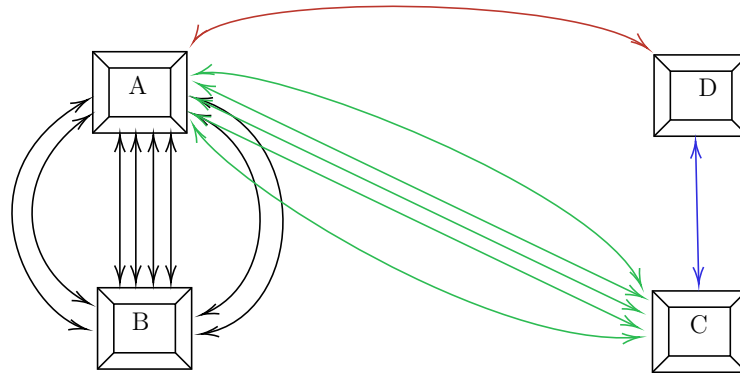


Figura 2.7: Terminales de 4 ciudades entre las cuales se realizan viajes. **Fuente:** Basado en los datos de [18].

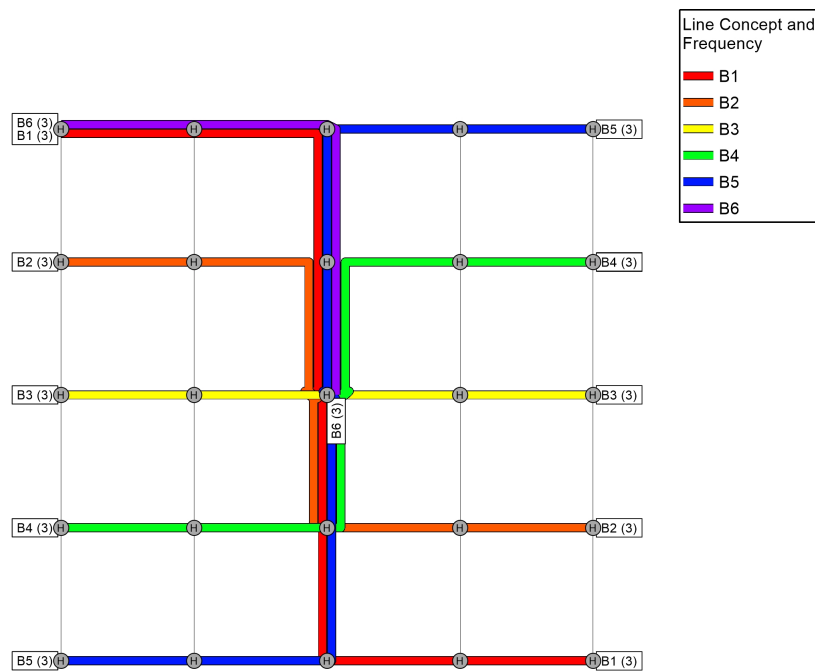


Figura 2.8: Líneas en una ciudad con tipografía de grilla. **Fuente:** [10].

Si un vehículo termina el servicio de la línea B1 en la estación v_{55} , podría a continuación trasladarse a la estación v_{11} para servir a un viaje de la línea B6 que empiece suficientemente tarde. Sin embargo, eso implicaría que cruce la ciudad entera sin pasajeros, lo cual no es conveniente. En este caso, la definición de compatibilidad entre viajes puede requerir que la distancia entre la última estación del primer viaje y la primera estación del segundo viaje, no supere una cota preestablecida.

Las soluciones del VSP consisten de rutas de servicio para las unidades de la flota.

Definición 17. Una ruta de servicio $r = (t_1, \dots, t_n)$ es una sucesión de viajes operados por un mismo vehículo y tal que dos viajes consecutivos son compatibles. Dada una ruta de servicio $r = (t_1, \dots, t_n)$, decimos que el viaje t_i pertenece a r , para todo $1 \leq i \leq n$. De igual forma decimos que el viaje de conexión $c := (t_i, t_{i+1})$ pertenece a r , para todo $1 \leq i < n$.

Definición 18. Una asignación vehicular \mathcal{R} es un conjunto de rutas de servicio cuya cardinalidad es menor o igual que el tamaño de la flota disponible. (Observar que, como la flota es homogénea, cada ruta de servicio puede ser asignada a cualquier unidad de la flota). \mathcal{R} es factible si cada viaje es cubierto exactamente por una de las rutas de servicio.

Definición 19. Sea $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_5)^t$ un vector de costos unitarios, donde

- γ_1 son los costos unitarios asociados a la duración de los viajes,
- γ_2 son los costos unitarios asociados a la distancia de los viajes,
- γ_3 son los costos unitarios asociados a la duración de los viajes vacíos,
- γ_4 son los costos unitarios asociados a la distancia de los viajes vacíos,
- y γ_5 son los costos fijos por el uso de cada vehículo.

Los costos operativos del sistema de transporte se definen de la siguiente manera:

$$\begin{aligned} \text{cost}(\mathcal{R}) = & \sum_{r \in \mathcal{R}} \left(\sum_{t \in r} (\gamma_1 \cdot d_t + \gamma_2 \cdot \text{length}_t) \right. \\ & + \sum_{c \in r} (\gamma_3 \cdot d_c + \gamma_4 \cdot \text{length}_c) \\ & \left. + \gamma_3 \cdot (d_{\text{dep}, l_r} + d_{l'_r, \text{dep}}) + \gamma_4 \cdot (\text{length}_{\text{dep}, l_r} + \text{length}_{l'_r, \text{dep}}) + \gamma_5 \right), \end{aligned}$$

donde l_r es la primera estación del primer viaje cubierto por la ruta de servicio r y l'_r es la última estación del último viaje cubierto por la ruta de servicio r .

Observar que la función de costos operativos puede escribirse en la siguiente forma equivalente:

$$\begin{aligned} \text{cost}(\mathcal{R}) = & \sum_{r \in \mathcal{R}} \left(\sum_{t=(p,l) \in r} (\gamma_1 \cdot (e_{p,l} - s_{p,l}) + \gamma_2 \cdot \text{length}_t) \right. \\ & + \sum_{c=((p,l),(p',l')) \in r} (\gamma_3 \cdot (s_{p',l'} - e_{p,l}) + \gamma_4 \cdot D_{l,l'}) \\ & \left. + \gamma_3 \cdot (L_{\text{dep}, l_r} + L_{l'_r, \text{dep}}) + \gamma_4 \cdot (D_{\text{dep}, l_r} + D_{l'_r, \text{dep}}) + \gamma_5 \right) \end{aligned}$$

El Problema de Asignación Vehicular (Vehicle Scheduling Problem, VSP) puede entonces ser formulado de la manera siguiente:

VSP. Dados una PTN (V, E) , un plan de líneas \mathcal{L} con su correspondiente calendarización de viajes π , un conjunto \mathcal{P} de periodos, duraciones $L_{l_1, l_2}, L_{\text{dep}, l}, L_{l, \text{dep}}, l, l_1, l_2 \in \mathcal{L}$ estimadas para los viajes vacíos potenciales, así como sus correspondientes longitudes $D_{l_1, l_2}, D_{\text{dep}, l}, D_{l, \text{dep}}, l, l_1, l_2 \in \mathcal{L}$, y un vector de costos unitarios $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_5)^t$, hallar una asignación vehicular factible \mathcal{R} que minimice la función de costos operativos del sistema.

El VSP puede ser resuelto de manera eficiente cuando la flota es homogénea y se tiene un solo depósito en el sistema, pues en este caso el problema puede ser reducido a un problema de flujo de costo mínimo, como veremos en el siguiente capítulo. Por otra parte, notar que en la

formulación del VSP se asume que fases anteriores como la planificación de líneas y calendarización de viajes ya han sido resueltas. Como se mencionó antes, este enfoque secuencial da lugar a soluciones subóptimas que, para nuestro caso, pueden resultar en costos excesivos de operación, por ejemplo, una cantidad elevada de vehículos.

La observación anterior motiva la idea de resolver simultáneamente diversas fases de planificación. En particular, en la siguiente sección se explora la posibilidad de determinar la calendarización de los viajes conjuntamente con las rutas vehiculares, flexibilizando la compatibilidad inicial entre viajes, con el objetivo de aumentar la cantidad de posibles rutas de servicio factibles.

2.5. Integración de Calendarización de Viajes y Asignación de Vehículos

En el problema de asignación vehicular descrito en la sección anterior, la compatibilidad entre viajes se determina a partir de tiempos de inicio y fin de los viajes que se consideran preestablecidos y fijos, es decir, la calendarización de los viajes ha sido calculada previamente, procurando que el tiempo promedio de viaje para los pasajeros sea mínimo. Sin embargo, este enfoque exclusivo en el beneficio del pasajero puede dar lugar a que los costos operativos para las fases posteriores sean muy altos, ya que los horarios calculados no pueden ser modificados. Puede ocurrir, por ejemplo, que dos viajes lleguen a ser compatibles si uno de ellos se inicia dos minutos después de lo establecido originalmente en el horario. Esto puede permitir que un vehículo pueda servir ambos viajes, reduciendo el costo de un vehículo adicional. De esta manera, si hay flexibilidad en la determinación de los tiempos de inicio y fin de los viajes, existe la posibilidad de alcanzar una mayor reducción en los costos de asignación vehicular significativamente. Este problema integrado de calendarización de viajes y asignación vehicular es presentado a continuación.

(Problema integrado de calendarización de viajes y asignación de vehículos TimVeh) Considerar una red evento-actividad $\mathcal{N}^0 = (\mathcal{E}^0, \mathcal{A}^0)$ para un conjunto de líneas \mathcal{L} , con cotas L_a, U_a para la duración de cada

actividad $a \in \mathcal{A}^0$, y pesos por pasajeros $w = (w_a)_{a \in \mathcal{A}^0}$ para las actividades de la red. Sean además T la longitud del periodo de planificación, \mathcal{P} el conjunto de periodos, $L_{l_1, l_2}, L_{\text{dep}, l}, L_{l, \text{dep}}, l, l_1, l_2 \in \mathcal{L}$ las duraciones mínimas para los viajes vacíos potenciales, y $D_{l_1, l_2}, D_{\text{dep}, l}, D_{l, \text{dep}}, l, l_1, l_2 \in \mathcal{L}$ sus correspondientes longitudes. Se pide encontrar una calendarización de viajes periódica y una asignación vehicular factible \mathcal{R} , tales que el tiempo de viaje promedio de los pasajeros y los costos operativos del sistema se minimicen.

A diferencia de la integración de otras fases previas, la integración de la calendarización de viajes y la asignación de vehículos debe compaginar diferentes periodos de planificación. Mientras el horario que se busca es periódico para un periodo de planificación de longitud T , la asignación de vehículos se realiza para todos periodos a considerar, es decir, es aperiódica.

Como se indicó con anterioridad, para la modelización se asume que todas las líneas tienen frecuencia 1. Si ocurriese que existen líneas con frecuencias mayores, se puede considerar cada repetición de la línea como una línea distinta.

Observar que el problema integrado TimVeh es un problema de optimización bi-criterio pues se minimiza la suma ponderada de β veces el tiempo promedio de viaje y γ veces los costos de operación del sistema. Al variar los parámetros (β, γ) se obtienen diferentes soluciones del problema. Por ejemplo, un alto valor de β respecto a γ favorecerá a tener menores tiempos de viaje, mientras que un alto valor de γ favorecerá a las soluciones con costos operativos más bajos. Al utilizar $\beta = 0$ ($\gamma = 0$) se obtiene una cota inferior para los costos operativos (tiempo de viaje). En particular, la solución que utiliza $\gamma = 0$ arroja el mismo tiempo de viaje que al resolver únicamente el problema de calendarización de viajes, ya que para un número ilimitado de vehículos existe una solución factible del VSP que es asignar un vehículo a cada viaje.

Es importante notar que, como la fase de planificación de líneas se considera fija, los costos basados en la distancia de los viajes, i.e.,

$$\gamma_2 \cdot \text{length}_t, t \in r, r \in \mathcal{R},$$

que son parte de los costos operativos, están fijos y en consecuencia pueden omitirse de la función objetivo.

Capítulo 3

Formulación de Modelos para el VSP

En este capítulo presentaremos dos modelos de programación lineal entera para el problema de asignación vehicular (VSP). En el caso de flota homogénea con un solo depósito: un primer modelo basado en un flujo de costo mínimo, y un modelo integrado de asignación de vehículos y calendarización de viajes.

3.1. Modelización como problema de flujo de costo mínimo

Cuando la flota de transporte es homogénea y está ubicada en un único depósito, el VSP puede reducirse a un problema de flujo de costo mínimo, el cual es conocido por admitir algoritmos de solución polinomiales. Para ello, se construye una red $D = (V, A)$ con los siguientes elementos.

- Un conjunto de nodos $V = \mathcal{T} \cup \{s, t\}$, junto a un nodo fuente s y un nodo sumidero t que representan al depósito. Asociados a cada viaje $i = (p_i, l_i)$ están una hora de inicio s_i , una hora de finalización e_i , una estación de salida $\text{first}(l_i)$ y una estación de llegada $\text{last}(l_i)$.

Para la formulación de este primer modelo, simplificaremos un poco la notación introducida en el capítulo anterior y designaremos a los viajes omitiendo la información del período y la línea, es decir,

escribiremos $i \in \mathcal{T}$ para referirnos a $(p_i, l_i) \in \mathcal{T}$.

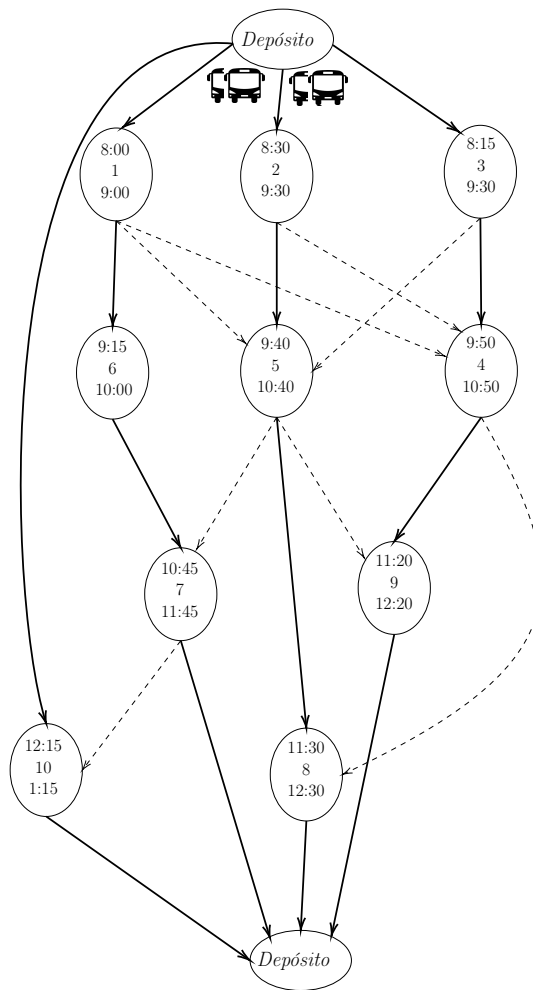
- Un conjunto de arcos A que refleja las relaciones de compatibilidad entre viajes definidas en el capítulo anterior (ver Definición 16 y observaciones subsiguientes), y que está construido como sigue

$$A = \{(i, j) \mid i \vdash j, i, j \in \mathcal{T}\} \cup \{(s, i) \mid i \in \mathcal{T}\} \cup \{(i, t) \mid i \in \mathcal{T}\}.$$

Recordar que la relación de compatibilidad entre viajes se define a partir de los tiempos de salida y llegada de los viajes, pero puede extenderse para incluir criterios como la distancia entre la estación de llegada del primer viaje y la estación de salida del segundo viaje.

- Un vector de costos $c \in \mathbb{R}^A$ sobre los arcos de D . Cuando un arco representa un par de viajes compatibles, su costo refleja alguna medida de la inconveniencia de cubrir ambos viajes de manera consecutiva con un mismo vehículo, por ejemplo, el tiempo muerto de espera entre la finalización del primer viaje y el inicio del segundo viaje, o la distancia a recorrer desde la estación de llegada del primer viaje a la estación de salida del segundo viaje. Los costos de los arcos que salen del depósito s reflejan costos fijos asociados al uso de un vehículo dentro de la operación del sistema. Los costos de los arcos que retornan al depósito t generalmente se fijan iguales a cero.

De esta manera, el problema puede formularse como el siguiente programa lineal entero:



$$\text{mín} \sum_{(i,j) \in A} c_{ij} x_{ij}$$

$$\text{s.t.} \sum_{i:(i,j) \in A} x_{ij} - \sum_{i:(j,i) \in A} x_{ji} = 0, \quad \forall j \in \mathcal{T},$$

$$\sum_{i:(i,j) \in A} x_{ij} = 1, \quad \forall j \in \mathcal{T},$$

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A.$$

Donde las variables binarias x_{ij} son variables de selección de arcos que indican lo siguiente:

$$x_{ij} = \begin{cases} 1, & \text{si el arco } (i, j) \text{ forma parte de alguna ruta en la solución} \\ 0, & \text{caso contrario.} \end{cases}$$

El modelo consiste en seleccionar un conjunto de caminos desde s hasta t , de tal forma que cada nodo $j \in \mathcal{T}$ pertenezca exactamente a un camino, y que la suma de los costos de los caminos sea mínima. Cada camino de s a t representa una ruta de servicio para un vehículo en la PTN (ver Definición 17). La primera restricción del modelo es una condición estándar de flujo que, como D es acíclico, garantiza que la solución consista de caminos de s a t ; mientras que la segunda asegura que cada viaje sea cubierto por exactamente un camino. Luego, la solución óptima del modelo consiste de un conjunto de rutas de costo mínimo que cubren todos los viajes programados. Es conocido que este modelo puede resolverse de manera eficiente, ya que es equivalente a un modelo de flujo de costo mínimo.

3.2. Modelo integrado TimVeh

El modelo anterior tiene el problema de que los horarios de los viajes están prefijados en los valores que fueron calculados en la fase de calendarización de viajes, generalmente tomando en cuenta de manera exclusiva el beneficio del pasajero. Esto puede elevar demasiado los costos de la asignación vehicular. Por ejemplo, podrían existir dos viajes para los cuales hiciera falta solamente un minuto para que sean compatibles. Aún así, el modelo de flujo asignará estos viajes a dos vehículos distintos. Sin embargo, si se flexibiliza este horario y se permite que el segundo viaje inicie un minuto después de lo establecido, un mismo vehículo podría tomar ambos viajes, reduciendo los costos de operación. Esta es precisamente la motivación para formular un modelo que integre la calendarización de viajes y la asignación de vehículos. La idea principal de los mismos es permitir que los horarios puedan moverse de manera relativamente libre, de modo que el inicio y fin los viajes se acomoden a los objetivos de la fase de asignación vehicular, por ejemplo, reducir el número de vehículos requeridos para la operación del sistema o reducir el tiempo muerto en que un vehículo permanece moviéndose sin pasajeros para conectar dos viajes consecutivos.

Presentaremos en esta sección el modelo TimVeh, propuesto en [25] para la solución integrada de la asignación de vehículos y calendarización

de viajes. Este modelo se formula sobre una Red Evento-Actividad (EAN, ver Definición 11 en el capítulo anterior), y emplea las siguientes variables de decisión:

- Variables enteras $\pi_i \in \{0, \dots, T - 1\}, i \in \mathcal{E}^0$ que indican los horarios (dentro del período de planificación) en los que se programan los eventos de la EAN.
- Variables enteras $z_a \in \mathbb{Z}, a \in \mathcal{A}^0$, que se requieren para modelizar la periodicidad del horario.
- Variables enteras positivas $d_l \in \mathbb{N}, l \in \mathcal{L}$, que indican la duración de cada viaje.
- Variables enteras $s_{p,l}, e_{p,l} \in \mathbb{N}, p \in \mathcal{P}, l \in \mathcal{L}$, que indican los horarios de inicio y fin de cada viaje.
- Variables binarias $x_{(p_1, l_1), (p_2, l_2)} \in \{0, 1\}, p_1, p_2 \in \mathcal{P}, l_1, l_2 \in \mathcal{L}$, entre pares de viajes (p_1, l_1) y $(p_2, l_2) \in \mathcal{T}$, que toman el valor de uno, si los viajes son cubiertos de manera consecutiva por alguna ruta de servicio, o cero caso contrario.
- Variables binarias $x_{\text{dep},(p,l)}, x_{(p,l),\text{dep}} \in \{0, 1\}$, con $p \in \mathcal{P}, l \in \mathcal{L}$, que toman el valor de uno si y solamente si el viaje asociado es el primero o último viaje de una ruta de servicio, respectivamente.
- Variables enteras $y_{(p_1, l_1), (p_2, l_2)} \in \mathbb{Z}$, con $p_1, p_2 \in \mathcal{P}, l_1, l_2 \in \mathcal{L}$, que indican la duración de los viajes de conexión entre pares de viajes consecutivos dentro de una ruta de servicio.

Empleando estas variables, el modelo TimVeh puede ser formulado como el siguiente programa lineal entero (los parámetros están descritos en las Definiciones 14 y 15 del capítulo anterior):

$$(\text{TimVeh}) \text{ m\u00edn } \beta \cdot \sum_{a=(i,j) \in \mathcal{A}^0} w_a \cdot (\pi_j - \pi_i + z_a \cdot T) \quad (3.1)$$

$$+ \gamma_1 \cdot \sum_{l \in \mathcal{L}} d_l \quad (3.2)$$

$$+ \gamma_3 \cdot \sum_{p_1 \in \mathcal{P}} \sum_{l_1 \in \mathcal{L}} \left(\sum_{p_2 \in \mathcal{P}} \sum_{l_2 \in \mathcal{L}} y_{(p_1, l_1), (p_2, l_2)} \right. \\ \left. + x_{\text{dep}, (p_1, l_1)} \cdot L_{\text{dep}, l_1} + x_{(p_1, l_1), \text{dep}} \cdot L_{l_1, \text{dep}} \right) \quad (3.3)$$

$$+ \gamma_4 \cdot \sum_{p_1 \in \mathcal{P}} \sum_{l_1 \in \mathcal{L}} \left(\sum_{p_2 \in \mathcal{P}} \sum_{l_2 \in \mathcal{L}} (x_{(p_1, l_1), (p_2, l_2)} \cdot D_{l_1, l_2}) \right. \\ \left. + x_{\text{dep}, (p_1, l_1)} \cdot D_{\text{dep}, l_1} + x_{(p_1, l_1), \text{dep}} \cdot D_{l_1, \text{dep}} \right) \quad (3.4)$$

$$+ \gamma_5 \cdot \sum_{p \in \mathcal{P}} \sum_{l \in \mathcal{L}} x_{\text{dep}, (p, l)} \quad (3.5)$$

$$\text{s.t. } \pi_j - \pi_i + z_a \cdot T \leq U_a, \quad \forall a = (i, j) \in \mathcal{A}^0, \quad (3.6)$$

$$\pi_j - \pi_i + z_a \cdot T \geq L_a, \quad \forall a = (i, j) \in \mathcal{A}^0, \quad (3.7)$$

$$d_l = \sum_{a=(i,j) \in \mathcal{A}^0(l)} (\pi_j - \pi_i + z_a \cdot T), \quad \forall l \in \mathcal{L}, \quad (3.8)$$

$$s_{p,l} = p \cdot T + \pi_{\text{first}(l)}, \quad \forall p \in \mathcal{P}, l \in \mathcal{L}, \quad (3.9)$$

$$e_{p,l} = p \cdot T + \pi_{\text{first}(l)} + d_l, \quad \forall p \in \mathcal{P}, l \in \mathcal{L}, \quad (3.10)$$

$$s_{p_2, l_2} - e_{p_1, l_1} \geq x_{(p_1, l_1), (p_2, l_2)} \cdot L_{l_1, l_2} \\ - M \cdot (1 - x_{(p_1, l_1), (p_2, l_2)}), \quad \forall p_1, p_2 \in \mathcal{P}, l_1, l_2 \in \mathcal{L}, \quad (3.11)$$

$$1 = \sum_{p_1 \in \mathcal{P}} \sum_{l_1 \in \mathcal{L}} x_{(p_1, l_1), (p_2, l_2)} \\ + x_{\text{dep}, (p_2, l_2)}, \quad \forall p_2 \in \mathcal{P}, l_2 \in \mathcal{L}, \quad (3.12)$$

$$1 = \sum_{p_2 \in \mathcal{P}} \sum_{l_2 \in \mathcal{L}} x_{(p_1, l_1), (p_2, l_2)} \\ + x_{(p_1, l_1), \text{dep}}, \quad \forall p_1 \in \mathcal{P}, l_1 \in \mathcal{L}, \quad (3.13)$$

$$y_{(p_1, l_1), (p_2, l_2)} \geq s_{p_2, l_2} - e_{p_1, l_1} \\ - M' \cdot (1 - x_{(p_1, l_1), (p_2, l_2)}), \quad \forall p_1, p_2 \in \mathcal{P}, l_1, l_2 \in \mathcal{L}, \quad (3.14)$$

$$\begin{aligned}
\pi_i &\in \{0, \dots, T-1\}, \quad i \in \mathcal{E}^0, \\
z_a &\in \mathbb{Z}, \quad a \in \mathcal{A}^0, \\
d_l &\in \mathbb{N}, \quad l \in \mathcal{L}, \\
s_{p,l}, e_{p,l} &\in \mathbb{N}, \quad p \in \mathcal{P}, l \in \mathcal{L}, \\
x_{(p_1, l_1), (p_2, l_2)} &\in \{0, 1\}, \quad p_1, p_2 \in \mathcal{P}, l_1, l_2 \in \mathcal{L}, \\
x_{\text{dep}, (p, l)}, x_{(p, l), \text{dep}} &\in \{0, 1\}, \quad p \in \mathcal{P}, l \in \mathcal{L}, \\
y_{(p_1, l_1), (p_2, l_2)} &\in \mathbb{Z}_0^+, \quad p_1, p_2 \in \mathcal{P}, l_1, l_2 \in \mathcal{L}.
\end{aligned}$$

La función objetivo consiste de cinco componentes ponderadas, que miden: el tiempo de viaje total de los pasajeros (asumiendo cargas de pasajeros w_a sobre los arcos de la EAN que han sido calculadas previamente durante la fase de enrutamiento de pasajeros); la duración de los viajes; la duración de viajes vacíos, incluyendo los viajes de salida y retorno del depósito; la longitud de viajes vacíos; y el número de vehículos.

La primera familia de restricciones (3.6) y (3.7) contiene las restricciones estándar para el problema de calendarización de viajes. Estas restricciones establecen que los horarios determinados para los eventos deben asegurar que la duración de cada actividad no sobrepase las cotas establecidas.

La restricción (3.8) relaciona la duración d_l de una línea l con los horarios determinados para los eventos de esa línea.

Las siguientes dos familias de restricciones (3.9) y (3.10) determinan el tiempo al que deben iniciar y terminar cada uno de los viajes en función de la duración de la línea correspondiente y de los horarios de sus eventos inicial y final.

La familia de restricciones (3.11) asegura que si un viaje (p_2, l_2) se realiza inmediatamente después de un viaje (p_1, l_1) por el mismo vehículo, el tiempo entre los respectivos fin e inicio de estos viajes debe ser mayor o igual a L_{l_1, l_2} . El lado derecho de estas restricciones puede incrementarse para incorporar cualquier tiempo adicional que sea necesario para preparar un vehículo antes de iniciar un viaje. Para la constante M puede

elegirse cualquier valor que sea suficientemente grande de modo que si un viaje (p_2, l_2) no se realiza justo después de un viaje (p_1, l_1) la restricción resultante sea trivial. En [25] se propone utilizar

$$M \geq \max \mathcal{P} \cdot T + \max_{a \in \mathcal{A}^0} U_a \cdot \max_{l \in \mathcal{L}} |e \in l|.$$

Las dos familias de restricciones siguientes (3.12) y (3.13) modelan las rutas de servicio. Todo viaje se realiza justo después (antes) de otro, o es el primer (último) viaje en una ruta de servicio. Notar que estas restricciones capturan la misma lógica que las restricciones equivalentes del primer modelo. En efecto, las variables binarias x pueden verse como variables de selección de arcos de un grafo dirigido que contiene un nodo (p, l) por cada viaje y un nodo adicional dep que representa al depósito. Entonces, estas restricciones requieren que los arcos seleccionados formen circuitos que empiezan y terminan en el depósito (rutas de servicio), tal que cada viaje sea cubierto exactamente por un circuito.

Las últimas restricciones (3.14) fijan los valores de las variables que miden la duración de los viajes de conexión, en función de los horarios de inicio y fin de los viajes que son conectados. Aquí, de nuevo, debe elegirse un valor suficientemente grande para la constante M' . Por ejemplo, puede demostrarse que. $M' = M$ funciona de forma adecuada.

Obs. Notar que se puede restringir el número de vehículos de la flota a N al añadir la restricción

$$\sum_{p \in \mathcal{P}} \sum_{l \in \mathcal{L}} x_{dep, (p, l)} \leq N$$

Capítulo 4

Resultados Computacionales

En este capítulo se reportan los resultados de la aplicación de los dos modelos implementados, el modelo basado en flujo de costo mínimo y el modelo integrado, a 6 instancias de redes de transporte público.

Se resuelven los problemas de asignación de vehículos para un horizonte de planificación de 16 periodos, equivalente a un día de operación del sistema. Para determinar la compatibilidad entre viajes, al criterio de compatibilidad básico (ver Definición 16 del Capítulo 2) se le añade la regla de que dos viajes son compatibles si el tiempo que toma ir desde la estación final del primero hasta la estación inicial del otro no es muy grande. De forma específica, se admiten únicamente aquellos viajes de conexión cuyas duraciones se encuentren en el cuartil más bajo de entre todos los viajes de conexión posibles. Esto es, si para una red de transporte hay cuatro posibles viajes de conexión con duraciones de 0, 2, 4 y 8 minutos, solo se admitirán dentro de las rutas de servicio los viajes que duran 0 o 2 minutos. El plan de líneas, así como el enrutamiento de pasajeros se resolvieron previamente de forma individual en otras componentes del Proyecto de MODELOS INTEGRADOS DE OPTIMIZACIÓN DE TRANSPORTE PÚBLICO y se consideran fijos. Se calcula la calendarización de viajes y la asignación de vehículos primero de manera secuencial y luego de forma integrada. En el primer caso, se toma la calendarización de viajes calculada también en otra componente del proyecto.

Los experimentos fueron ejecutados en un computador con sistema

operativo Linux CentOS 7 con procesador de 4 núcleos Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz y 32 GB de memoria RAM. Para la solución de los modelos de programación lineal entera se utilizó el algoritmo branch-and-cut implementado en el solver Gurobi 9.5.1 ([12]).

4.1. Descripción de las Instancias

En esta sección se describen las instancias sobre las que se han probado los dos modelos estudiados en este trabajo. Las instancias *Toy*, *Grid* y *Athens* se obtuvieron del software de computación libre para transporte LinTim ([24]). Las instancias *Toy0*, *Toy2* fueron creadas en el Proyecto de MODELOS INTEGRADOS DE OPTIMIZACIÓN DE TRANSPORTE PÚBLICO. Finalmente, la instancia *Small* está basada en la instancia empleada para ilustrar la Definición 7 y sus viajes correspondientes. La instancia *Athens* tiene una longitud de periodo T de 600 unidades de tiempo, cada una equivalente a una décima de minuto. Todas las demás instancias trabajan con un periodo de 60 minutos, es decir, una hora.

En todas las instancias se ha resuelto previamente la fase de planificación de líneas, por lo que se cuenta con información de las líneas que operan en el sistema, con sus respectivas frecuencias. Para el primer modelo también se ejecutó la fase de calendarización de viajes y se cuenta con información de la hora de inicio y finalización de cada viaje. Para el segundo modelo, se ha ejecutado la fase de enrutamiento de pasajeros, de donde se conoce el número de pasajeros que se desplazan por cada arco de la red evento-actividad (es decir, la carga de los arcos de la EAN).

Las líneas consideradas son no dirigidas, por tanto al transformarlas a líneas dirigidas se obtienen dos copias de cada línea, una en sentido de *ida* y otra en sentido de *regreso*. Además, las líneas con frecuencias mayores a 1 se tratan como copias individuales de una misma línea, cada una con frecuencia igual a 1.

Los datos de todas las instancias, así como todos los archivos auxiliares generados durante las pruebas computacionales y el código fuente correspondiente a la implementación de los modelos forman parte de este trabajo y se entregan en soporte magnético.

Describimos a continuación las características de cada instancia.

4.1.1. Small

Esta instancia tiene asociada la red de transporte público de 6 estaciones y 5 aristas que ha sido utilizada como ejemplo en capítulos anteriores, la diferencia es que ahora la segunda línea tiene frecuencia 2. Es decir, se tiene una línea de frecuencia 1 $l_1^{(1)}$ y una línea de frecuencia 2 $l_2^{(2)}$. Esta última se trata como dos copias de una misma línea, cada una con frecuencia 1: $l_{2:1}^{(1)}$ y $l_{2:2}^{(1)}$. Así, se tienen 3 líneas; al darle a cada una dirección de ida y vuelta se termina con un total de 6 líneas a considerar en el modelo.

La Figura 4.1 demuestra un esquema de la PTN asociada a esta instancia. Los pesos sobre las aristas corresponden al tiempo que toma ir de una estación a la siguiente dentro de la red de transporte público.

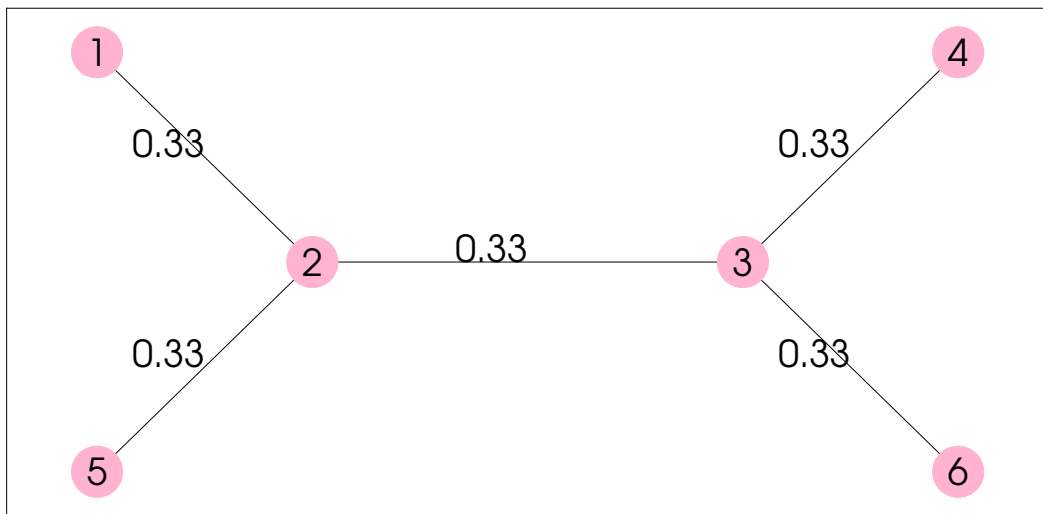


Figura 4.1: PTN de la instancia Small.

La Tabla 4.1 indica la frecuencia de cada línea, y las estaciones visitadas por la misma.

Línea	Estaciones	Frecuencia
1	1,2,3,4	1
2,	5,2,3,6	2

Tabla 4.1: Líneas de la instancia Small.

4.1.2. Toy0

Esta instancia posee una distribución de estaciones menos usual. Sus 8 estaciones se conectan por 7 aristas. Consta de dos líneas, la primera de frecuencia 4 y la segunda de frecuencia 2. Una vez replicadas las frecuencias y considerados los sentidos, se tienen en total 12 líneas en el modelo.

La Figura 4.2 muestra la PTN asociada a esta instancia, mientras que la Tabla 4.2 indica información acerca de la frecuencia y las estaciones visitadas por cada línea.

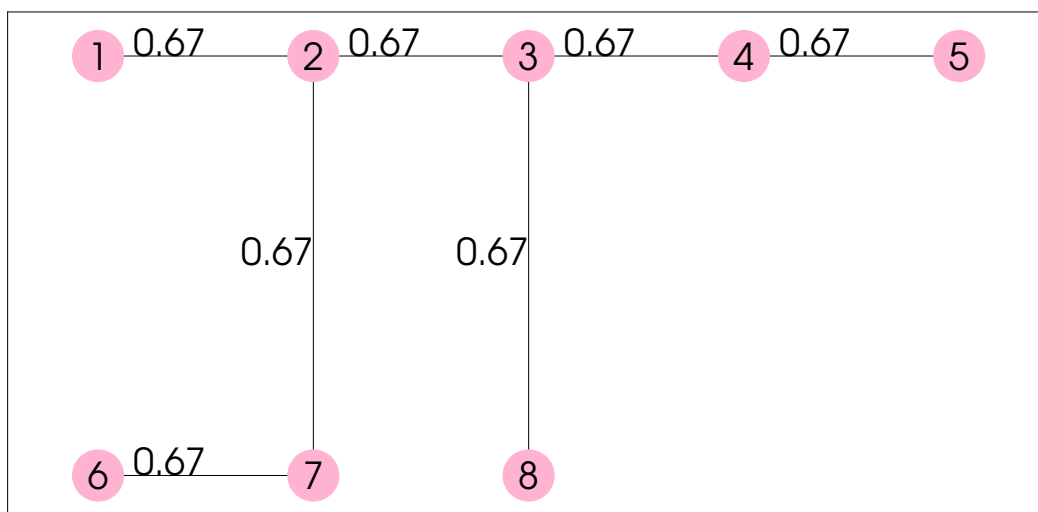


Figura 4.2: PTN de la instancia Toy0.

Línea	Estaciones	Frecuencia
1	1, 2, 3, 4, 5	4
2	2, 3, 6, 7, 8	2

Tabla 4.2: Líneas de la instancia Toy0.

4.1.3. Toy1

Toy1 es otra red pequeña. Consiste de 8 estaciones y 8 aristas. Tiene asociadas 6 líneas de frecuencias 3, 2, 4, 1, 1, 3, respectivamente. Las mismas dan origen a un total de 28 líneas en el modelo, una vez consideradas las copias debido a las frecuencias y al sentido de circulación.

La Figura 4.3 muestra la PTN asociada a esta instancia, mientras que la Tabla 4.3 indica información acerca de la frecuencia y las estaciones visitadas por cada línea.

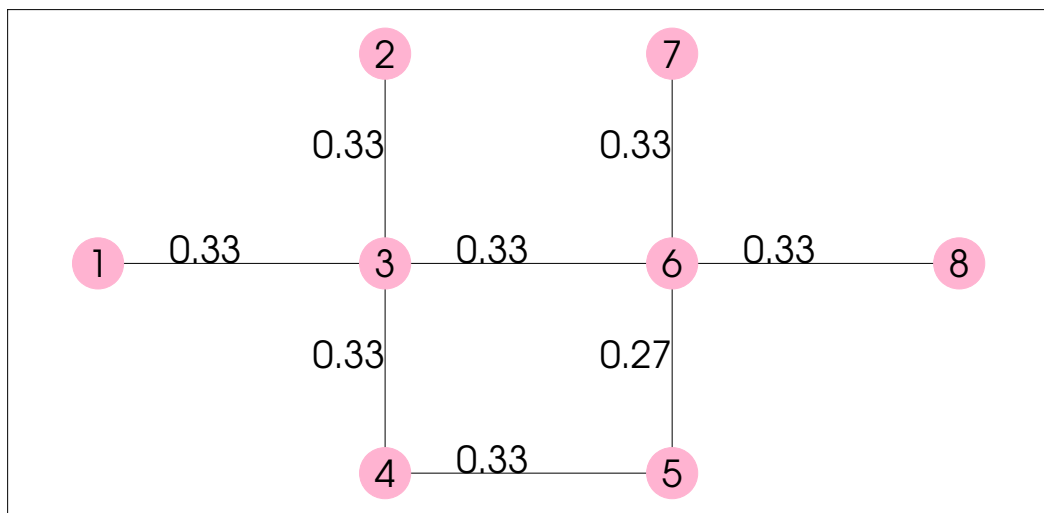


Figura 4.3: PTN de la instancia Toy1.

Línea	Estaciones	Frecuencia
2	8, 2, 3, 6	3
3	5, 6	2
4	3, 5, 6	4
5	2, 3, 4	1
6	3, 4, 5, 6	1
8	1, 3, 4, 5, 6, 7	3

Tabla 4.3: Líneas de la instancia Toy1.

4.1.4. Toy2

Toy2 es una red que tiende a ser mediana. Consiste de 10 estaciones y 9 aristas. Tiene asociadas 7 líneas de frecuencias 2, 4, 1, 7, 3, 5, 11, respectivamente. Estas corresponden a un total de 66 líneas en el modelo, una vez contabilizadas las copias por frecuencias y sentido de circulación.

La Figura 4.4 muestra la PTN asociada a esta instancia, mientras que la Tabla 4.4 indica información acerca de la frecuencia y las estaciones visitadas por cada línea.

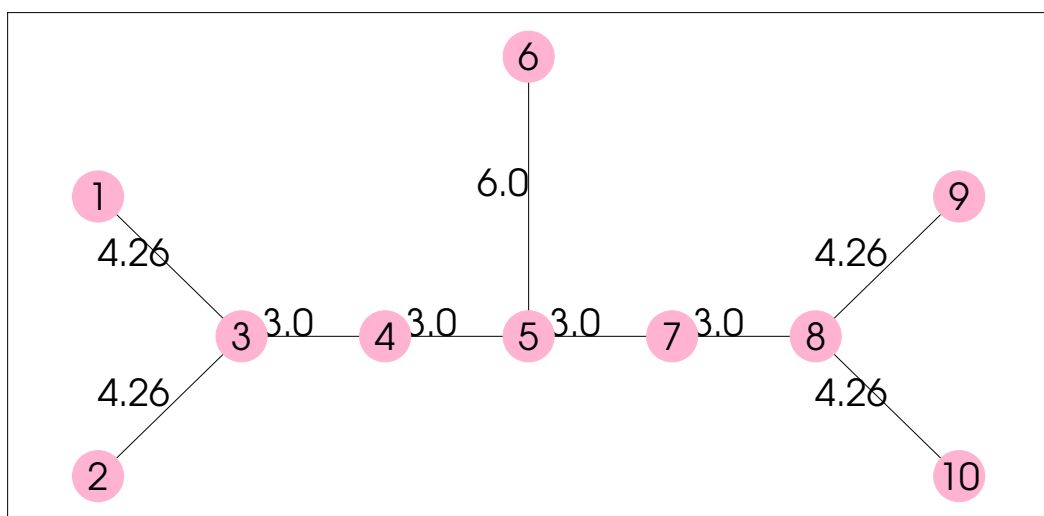


Figura 4.4: PTN de la instancia Toy2.

Línea	Estaciones	Frecuencia
2	2, 3, 4, 5, 7, 8, 9	2
3	1, 3, 4, 5, 6	4
4	5, 6, 7, 8, 10	1
6	5, 6	7
7	1, 2, 3	3
8	8, 9, 10	5
9	3, 4, 5, 7, 8	11

Tabla 4.4: Líneas de la instancia Toy2.

4.1.5. Grid

La instancia corresponde a una red de transporte con topología de una grilla de 5×5 nodos, por tanto tiene 25 estaciones y 40 aristas. Las 11 líneas asociadas tienen frecuencias de 1, 2, 3, 5, 7, 8, 9, 11, 19, 23, 25, respectivamente. El modelo incluye un total de 34 líneas una vez consideradas las copias por frecuencias y sentido de circulación.

La Figura 4.5 muestra la PTN asociada a esta instancia, mientras que la Tabla 4.5 indica información acerca de la frecuencia y las estaciones visitadas por cada línea.

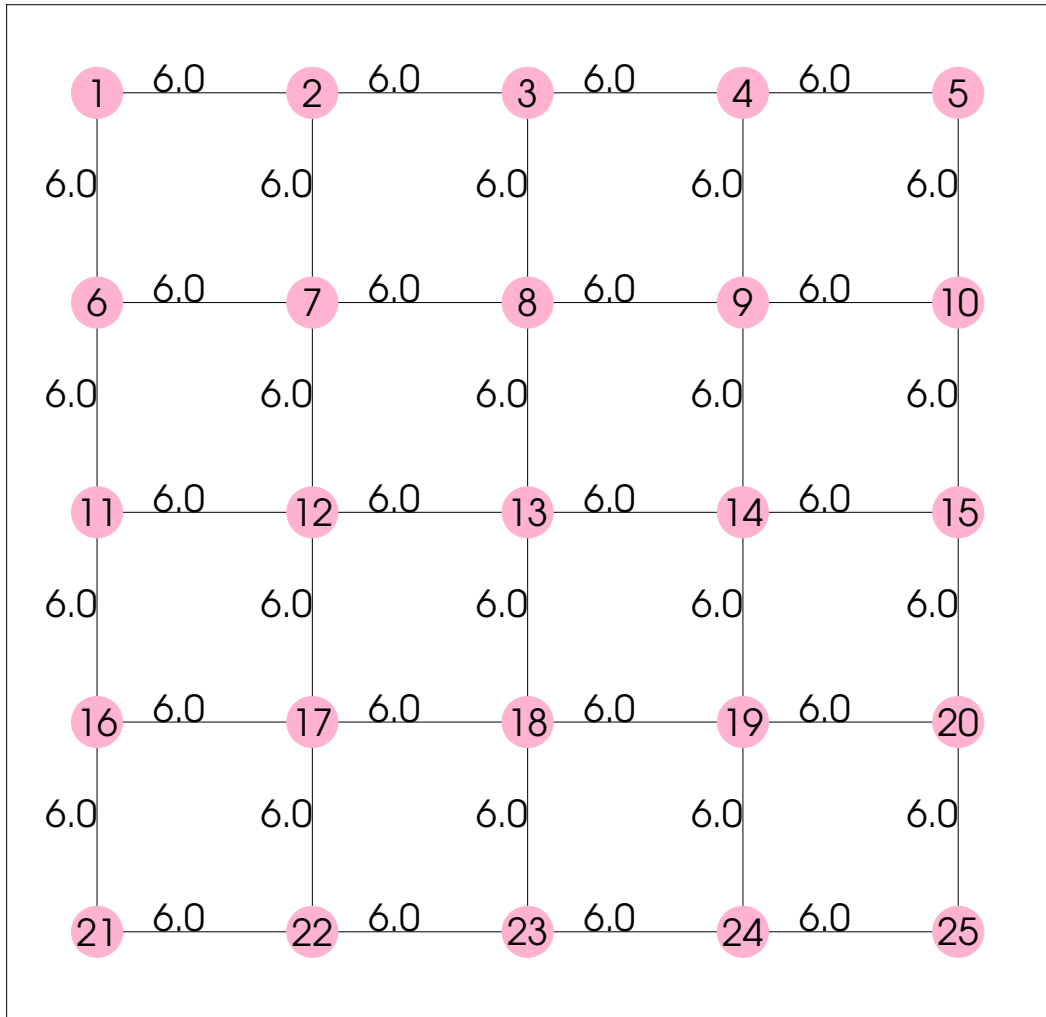


Figura 4.5: PTN de la instancia Grid.

Línea	Estaciones	Frecuencia
1	1, 2, 3, 4, 5, 6, 11, 12, 13, 18, 19, 20	1
2	13, 14, 15, 16, 17, 18, 21, 22, 23, 24, 25	1
3	6, 7, 8, 9, 10, 11, 12, 13, 18, 19, 20	1
5	3, 4, 5, 8, 11, 12, 13	1
7	6, 7, 8, 9, 10, 12, 13	3
8	18, 19, 20, 23, 24, 25	1
9	16, 17, 18, 23, 24, 25	2
11	12, 13, 14, 15, 16, 17	1
19	1, 2, 3, 8, 13	3
23	18, 13	2
25	17, 12	1

Tabla 4.5: Líneas de la instancia Grid.

4.1.6. Athens

Esta instancia real fue creada a partir de información de la red de metro de la ciudad de Atenas. Cuenta con 51 y 52 aristas. Luego de tratar las frecuencias de cada línea y sus orientaciones, se obtienen en total 96 líneas en el modelo.

La Figura 4.6 muestra la PTN asociada a esta instancia, mientras que la Tabla 4.6 indica información acerca de la frecuencia y las estaciones visitadas por cada línea.

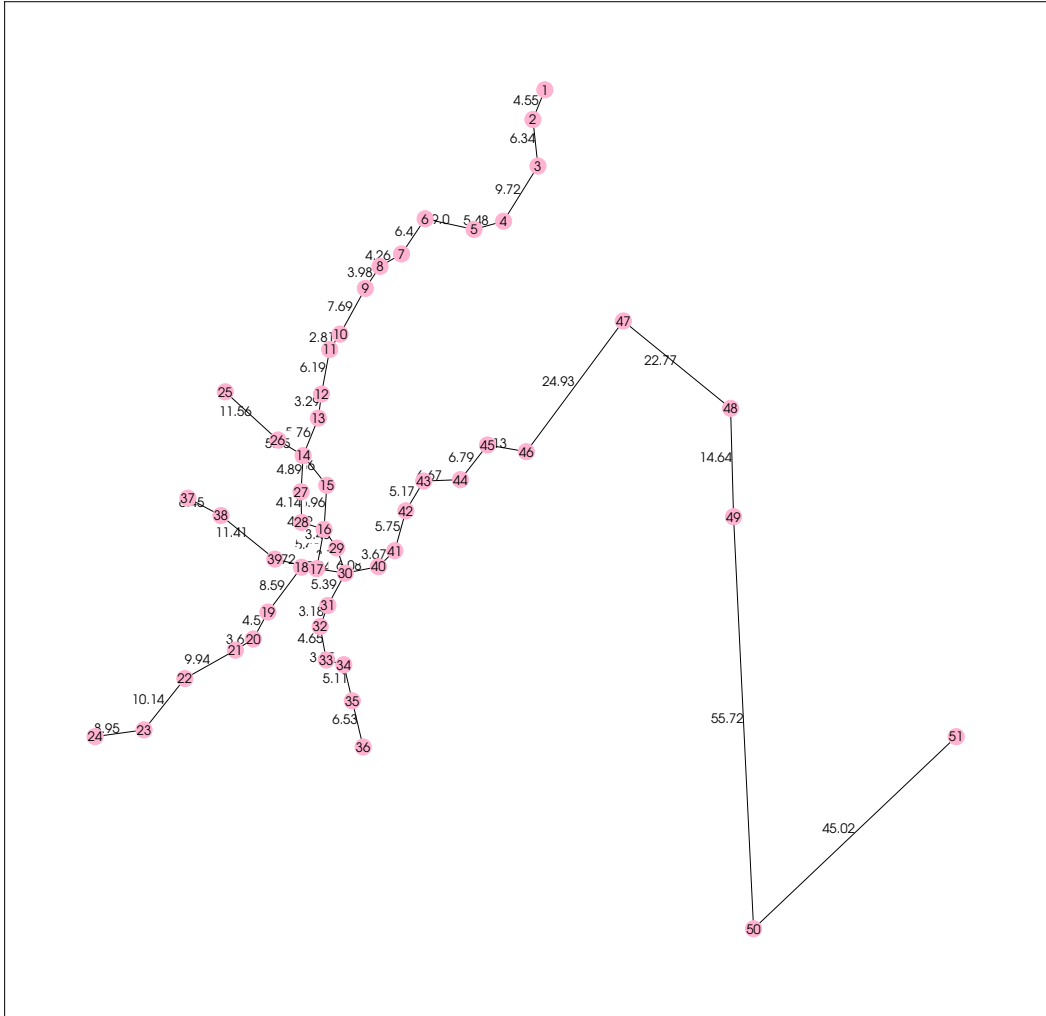


Figura 4.6: PTN de la instancia Athens.

Línea	Estaciones	Frecuencia
1	1, 2, 3, 4, 5, 6	2
2	1, 2, 3, 4, 5, 6, 7	1
11	5, 6, 7, 8, 9, 10, 11, 12, 13, 14	1
17	6, 7, 8, 9, 10, 11, 12, 13, 14	2
23	7, 8, 9, 10, 11, 12, 13, 14	3
29	10, 11, 12, 13, 14, 15, 16	1
31	10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20	1
37	16, 17, 18, 19, 20, 21, 22, 23	2
38	16, 17, 18, 19, 20, 21, 22, 23, 24	1
41	14, 16, 25, 26, 27, 28	2
45	14, 16, 26, 27, 28, 29, 30	1
48	14, 16, 27, 28, 29, 30	6
49	32, 33, 34, 35, 14, 16, 27, 28, 29, 30, 31	3
51	32, 33, 34, 35, 16, 29, 30, 31	2
53	32, 33, 34, 35, 30, 31	1
54	32, 33, 34, 35, 36, 30, 31	2
55	37, 38, 39, 40, 41, 42, 43, 44, 45, 17, 30	7
57	40, 41, 42, 43, 44, 45, 30	1
58	40, 41, 42, 43, 44, 45, 46, 47, 30	7
59	47, 48, 49, 50, 51	2

Tabla 4.6: Líneas de la instancia Athens.

4.2. Minimización del tamaño de la flota

En este experimento, comparamos los resultados obtenidos por los dos modelos cuando la función objetivo se enfoca únicamente en minimizar el número de vehículos que se requieren para cubrir todos los viajes. Con esta finalidad, utilizamos los siguientes parámetros de costos:

- En el caso del primer modelo, el valor de c_{ij} es 1 si $i = s$ y 0 en caso contrario, es decir, se cubren los viajes con el menor número posible de rutas de servicio.
- Para el modelo integrado se estudiaron dos posibles configuraciones:

en la primera, se toman en cuenta en la función objetivo el tiempo total de viaje de los pasajeros, ponderado por el parámetro $\beta = 1$, y el número de vehículos, ponderado con el parámetro $\gamma_5 = 100$; es decir, cada vehículo extra tiene el mismo peso que 100 minutos adicionales en el tiempo de viaje total de los pasajeros. En la segunda configuración, no se contabiliza el tiempo de los pasajeros, por lo que se fija $\beta = 0$. En ambas configuraciones se fija a cero el valor de los parámetros $\gamma_i, i \leq 4$.

Los resultados obtenidos de este experimento están indicados en la Tabla 4.2. Cada fila de la tabla corresponde a los resultados de un modelo/configuración sobre una instancia: M1 indica el primer modelo, M2 el segundo modelo en su primera configuración, y M3 el segundo modelo en su segunda configuración. Se reporta lo siguiente:

- Cantidad de viajes a realizar (NT).
- Número de restricciones en el modelo (NC).
- Número de variables en el modelo (NV).
- Número de vehículos empleados en la solución (NVH).
- Valor de la función objetivo obtenido (Obj).
- Mejor cota obtenida (BB).
- Tiempo de cómputo en segundos (RT).
- Gap obtenido (GAP).
- Número de nodos explorados en el algoritmo de branch-and-cut en la última optimización (ND).

Debido a que M1 se resuelve de forma eficiente, no fue necesario especificar parámetros de tiempo límite ni de tolerancia del gap en el solver. Para M2 se utilizó un tiempo límite de 2 horas y un margen del 1% de tolerancia del gap, mientras que para M3 el tiempo límite fue de 8 horas, con 0% de tolerancia para el gap¹.

¹Cabe mencionar que los parámetros predeterminados del solver no brindan solución alguna para el modelo integrado. En consecuencia, para tener alguna solución factible se utilizó la modificación de los parámetros del solver expuesta en la sección siguiente.

Instancia	Modelo	NT	NC	NV	NVH	Obj	BB	RT	GAP	ND
Small	M1	96	192	1425	4	4	4	0.02	0.00%	0
	M2	96	5771	5818	3	14160	14022	14.31	0.97%	4300
	M3	96	5771	5688	3	300	300	0.57	0.00%	1
Toy0	M1	192	384	5109	12	12	12	0.03	0.00%	1
	M2	192	22413	22532	9	1152	1052	7200.04	8.68%	444795
	M3	192	22413	21956	9	900	700	28800.03	22.22%	811644
Toy	M1	448	896	36703	17	17	17	0.56	0.00%	1
	M2	448	156161	156372	13	71700	70500	7200.03	1.67%	11432
	M3	448	156161	154370	12	1200	2.97E-09	28800.02	100.00%	13329
Toy2	M1	1056	2112	139172	41	41	41	1.90	0.00%	1
	M2	1056	607807	608334	40	168422	164420	7200.15	2.38%	1421
	M3	1056	607807	599440	36	3600	8.09E-10	28800.14	100.00%	3732
Grid	M1	544	1088	30364	40	40	40	0.38	0.00%	1
	M2	544	139411	139838	33	227580	225364	3925.73	0.97%	1570
	M3	544	139411	137158	27	2700	1300	28800.03	51.85%	4430
Athens	M1	1536	3072	272545	56	56	56	5.99	0.00%	1
	M2	—	—	—	—	—	8.05E+07	7200.01	—	—
	M3	1536	1191013	1172942	89	8900	100	28800.17	98.88%	9

Tabla 4.7: Comparación del número de vehículos utilizados bajo un enfoque secuencial vs integrado.

4.3. Resultados del Modelo Integrado

4.3.1. Configuración de parámetros del solver

Debido a que 8 horas en la máquina virtual del supercomputador QUINDE del Centro de Modelización Matemática no son suficientes para resolver hasta la optimalidad incluso las instancias pequeñas con la configuración por defecto del solver Gurobi, y en algunos casos ni siquiera alcanzan para encontrar una solución factible, se modificaron los valores por defecto de algunos parámetros estándar del solver, como indicaremos a continuación. La lista completa de todos los parámetros del solver puede revisarse en [12].

Los parámetros ajustados fueron los siguientes:

- MIPFocus. Controla las estrategias de solución de alto nivel según los objetivos a alcanzar. El valor predeterminado de este parámetro (0) se enfoca en tener un balance entre encontrar soluciones factibles y probar la optimalidad de estas. Un valor de 1 indica que el solver debe enfocarse en encontrar soluciones factibles de forma más rápida. El valor de 2 es útil si se piensa que el solver no tiene problema en hallar soluciones factibles y se quiere poner más atención en probar optimalidad. Un valor de 3 enfoca al solver en mejorar la cota del modelo, esto es útil si la cota se mueve poco o nada.
- Cuts. Este parámetro determina qué tan agresivo es el uso de planos cortantes en la solución del modelo. Un valor de 0 desactiva la generación de planos cortantes, un valor de 3 fija la generación en el nivel más agresivo.
- Heuristics. Determina la cantidad de tiempo que se utiliza en aplicar heurísticas para generar soluciones factibles, como un porcentaje del tiempo total de cálculo. El valor predeterminado (0.05) indica que debe destinarse alrededor del 5% del tiempo de cálculo a la aplicación de heurísticas. Valores más grandes pueden producir mejores soluciones factibles, pero sacrifican la velocidad de progreso en la cota. Los valores pueden ir de 0 a 1.

Para calibrar los parámetros se realizaron varias pruebas sobre la instancia Toy2, con un tiempo límite de ejecución de una hora, y utilizando en la función objetivo coeficientes de costos con valores iguales a $\beta = \gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = 1$ y $\gamma_5 = 100$. El modelo a resolver tiene 1056 viajes, 607807 restricciones y 599440 variables.

Como se mencionó, las pruebas con el valor predeterminado de los parámetros no permitieron obtener ninguna solución factible. Por lo tanto, se probaron distintas opciones. Los valores de $MIPFocus = 2$ y $MIPFocus = 3$ no produjeron ningún cambio a los resultados. Esto es, los experimentos llevados a cabo para los valores 2 y 3 del parámetro $MIPFocus$ no hallaron ninguna solución factible para ninguna combinación con los distintos valores de $Cuts$. Los únicos experimentos en donde se pudieron obtener soluciones factibles fueron aquellos realizados con el parámetro $MIPFocus = 1$. A continuación se compara el desempeño del solver bajo las distintas combinaciones de valores probadas para los parámetros Heuristics y Cuts, fijando $MIPFocus = 1$. Como criterio para medir el desempeño se utilizó el gap obtenido al finalizar el tiempo límite de cálculo, es decir, valores menores indican un mejor desempeño.

Heuristics	Cuts			
	0	1	2	3
0.3	56.71832548	55.80836285	54.81777421	53.44736429
0.5	54.26657119	57.7179217	42.68409678	
0.75	20.73993858	15.93890723	17.22497773	20.13784361
0.9	19.28368179	19.53120012	20.6839482	99.7009
1	16.62263441	19.36151205	19.53120012	20.27072786

Tabla 4.8: Desempeño del modelo para distintas configuraciones de *Heuristics* y *Cuts*. En todos los casos se fijó $MIPFocus = 1$.

Como se observa en los cuadros precedentes, los parámetros $MIPFocus = 1$, $Cuts = 1$ y $Heuristics = 0.75$ permiten obtener los mejores resultados en las pruebas de calibración de los parámetros. Esta es la configuración que se utilizará en todos los experimentos.

4.3.2. Solución de las instancias

Las ponderaciones para la función objetivo utilizadas en los experimentos fueron $\beta = 1$ para el tiempo de viaje total de los pasajeros, $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = 1$ para las componentes de los costos de la asignación de vehículos concernientes a la longitud y duración de los viajes y viajes de conexión. Al parámetro γ_5 se le asignó un valor de 100, con el objetivo de lograr que el modelo ponga más énfasis en disminuir la cantidad de vehículos.

El número máximo de vehículos disponibles se fijó para cada instancia en un valor igual a la mitad del total de viajes, aunque en todas las instancias las soluciones obtenidas requieren muchos menos vehículos.

Se utilizó la configuración de parámetros descrita en la sección anterior y se ejecutaron 3 experimentos para cada instancia, bajo las condiciones que se indican a continuación.

MA. Se establece un tiempo límite de cálculo de 8 horas y se busca alcanzar la optimalidad (Parámetro $MIPGap = 0$).

MB. Se establece un tiempo límite de 2 horas de ejecución y se busca alcanzar la optimalidad (Parámetro $MIPGap = 0$).

MC. Se establece un tiempo límite de 2 horas de ejecución y se busca alcanzar una brecha de optimalidad del 1% (Parámetro $MIPGap = 0.01$).

Los resultados obtenidos pueden apreciarse en la Tabla [4.3.2](#).

Cada fila de la tabla corresponde a los resultados de un modelo/configuración sobre una instancia: MA indica la primera configuración del modelo integrado, MB representa a modelo integrado con su segunda configuración, y MC es modelo integrado con su tercera configuración. Se reporta lo siguiente: cantidad de viajes a realizar (NT), número de restricciones en el modelo (NC), número de variables en el modelo (NV), número de vehículos empleados en la solución (NVH), valor de la función objetivo obtenido (Obj), mejor cota obtenida (BB), tiempo de cómputo en segundos (RT), gap obtenido (GAP) y número de nodos explorados en el algoritmo de branch-and-cut en la última optimización (ND).

Instancia	Experimento	NT	NC	NV	NVH	Obj	BB	RT	GAP	ND
Small	MA	96	5771	5688	3	15266	15213	28800.01	0.35%	12182746
	MB	96	5771	5688	3	15266	15119	7200.01	0.96%	2746197
	MC	96	5771	5688	3	15266	15102	7200.01	1.07%	2779440
Toy0	MA	192	22413	21956	9	3420	3208	28800.02	6.20%	313988
	MB	192	22413	21956	9	3420	3090	7200.02	9.65%	73108
	MC	192	22413	21956	9	3420	3109	7200.07	9.09%	85111
Toy	MA	448	156161	154370	13	76730	75503.81	28800.19	1.60%	42859
	MB	448	156161	154370	14	77621.8	75326.55	7200.12	2.96%	5405
	MC	448	156161	154370	14	77896.8	75270.78	7200.03	3.37%	5426
Toy2	MA	1056	607807	599440	37	184606.8	170808.8	28800.11	7.47%	5298
	MB	1056	607807	599440	38	188709.1	167501.6	7200.12	11.24%	2534
	MC	1056	607807	599440	41	191762.5	167435.3	7200.61	12.69%	1815
Grid	MA	544	139411	137158	31	235358	232196	28800.13	1.34%	56076
	MB	544	139411	137158	37	237521	232165.1	7200.02	2.25%	5576
	MC	544	139411	137158	30	235217	232275	7200.04	1.25%	5272
Athens	MA	1536	1191013	1172942	768	80748948	80551233	28800.79	0.24%	855
	MB	1536	1191013	1172942	768	80837673	80551375	7200.25	0.35%	18
	MC	1536	1191013	1172942	768	81020130	80551473	4675.24	0.58%	1

Tabla 4.9: Resultados de la aplicación del modelo integrado.

Cabe recalcar que existen soluciones factibles para la instancia Athens que requieren de muchos. En efecto, en una prueba se encontró la siguiente solución.

NVH	Obj	NT	NC	NV	BB	RT	GAP	ND
116	81216254.71	1536	1191013	1172942	80551430.91	4036.02	0.82 %	1

Sin embargo, las soluciones óptimas obtenidas pueden ocupar muchos vehículos pero reducir el valor de las otras componentes de la función objetivo, por ejemplo el tiempo que los vehículos permanecen vacíos. Una mayor cantidad de vehículos se puede compensar con viajes más rápidos o con vehículos operados más tiempo. En la práctica, el enfoque a utilizar deberá ajustarse a de los objetivos de la optimización. Por ejemplo, se puede ajustar a la función objetivo de modo que se dé énfasis a la minimización de los viajes vacíos en longitud y/o duración. Esto se prueba en la siguiente sección.

4.4. Comparación entre componentes de la función objetivo

En esta sección se comparan los valores obtenidos para las componentes individuales de la función objetivo en las soluciones de las diferentes instancias, bajo distintas configuraciones de la función objetivo. Las configuraciones utilizadas en los experimentos fueron las siguientes:

- (T1) $\beta = \gamma_i = 0, i \leq 4$ y $\gamma_5 = 100$, 8 horas de cómputo y 0% de tolerancia para la brecha de optimalidad.
- (T2) $\beta = \gamma_i = 1, i \leq 4$ y $\gamma_5 = 100$, 2 horas de cómputo y 1% de tolerancia para la brecha de optimalidad.
- (T3) $\beta = \gamma_1 = \gamma_2 = \gamma_5 = 0$ y $\gamma_3 = \gamma_4 = 100$, 2 horas de cómputo y 1% de tolerancia para la brecha de optimalidad.

La Tabla 4.10 reporta, en este orden, la cantidad de viajes a realizar (NT), el número de restricciones en el modelo (NC), el número de variables en el modelo (NV), el número requerido de vehículos (NVH), el tiempo

total de viaje de los pasajeros (TTV), la duración total de los viajes (DT), la duración de los viajes vacíos (DET), la distancia total de los viajes vacíos (LET), el número de vehículos empleados en la solución (NVH), el valor de la función objetivo obtenido (Obj), la mejor cota obtenida (BB), el tiempo de cómputo en segundos (RT), el gap obtenido (GAP) y número de nodos explorados en el algoritmo de branch-and-cut en la última optimización (ND).

Instancia	Experimento	NT	NC	NV	NVH	TTV	DT	DET	LET	BB	RT	GAP	ND
Small	T1	96	5771	5688	3	14310	100	1230	0	300	0.57	0.00%	1
	T2	96	5771	5688	3	13860	116	990	0	15114	360.34	1.00%	116108
	T3	96	5771	5688	48	19770	113	480	0	48000	11.57	0.00%	1238
Toy0	T1	192	22413	21956	9	281	368	2617	0	700	28800.03	22.22%	811644
	T2	192	22413	21956	9	270	420	1830	0	3137	7200.04	8.27%	70716
	T3	192	22413	21956	96	400	580	960	0	100	7200.05	99.90%	134216
Toy	T1	448	156161	154370	12	72530	353	5708	219	2.97E-09	28800.02	100.00%	13329
	T2	448	156161	154370	15	70400	499	6075	62	75183.4	7200.03	4.27%	5559
	T3	448	156161	154370	224	91500	558	2240	0	0	7200.16	100.00%	32894
Toy2	T1	1056	607807	599440	36	192610	907	19321	502.68	8.09E-10	28800.14	100.00%	3732
	T2	1056	607807	599440	56	164420	1265	31275	78.1	167080.2	7200.31	17.55%	704
	T3	1056	607807	599440	528	356250	1871	7610	122.12	0	7200.09	100.00%	216
Grid	T1	544	139411	137158	27	227155	1118	7624	318	1300	28800.03	51.85%	4430
	T2	544	139411	137158	31	224280	1446	6206	120	232198	7200.06	1.26%	5834
	T3	544	139411	137158	272	256201	1559	2720	0	100	7200.03	99.96%	23918
Athens	T1	1536	1191013	1172942	89	92639083	15550	567257	1381.992	100	28800.17	98.88%	9
	T2	1536	1191013	1172942	123	80596510	13173	654718	1532.713	80551213	3694.26	0.89%	1
	T3	1536	1191013	1172942	97	81289522	13216	1.1E+09	1532.713	0	7200.18	100.00%	1

Tabla 4.10: Valores de las componentes individuales de la función objetivo, bajo distintos parámetros de ponderación.

Capítulo 5

Conclusiones

- Luego de analizar la Tabla 4.2 se evidencian varios puntos de manera general.
 - En primer lugar, el modelo M2 disminuye en promedio el número de vehículos obtenido por el modelo M1 en un 19%. El caso de menor diferencia es en la instancia Toy2, en donde dos horas con el modelo M2 disminuyeron únicamente en un vehículo la solución de M1.
 - En cuanto al modelo M3, se tiene una disminución promedio del número de vehículos obtenidos por el modelo M1 de un 25%.
 - Al comparar los modelos M2 y M3 se observa que en las instancias Small, Toy0 y Toy no hay diferencia marcada en el tamaño de la flota obtenido por los modelos. Sin embargo, en las instancias Toy2 y Grid existe una reducción más significativa del número de vehículos necesarios. Debido a que no se obtuvieron soluciones del modelo M2 para la instancia Athens no se pueden establecer comparaciones con M3.
- De forma general, es evidente la mejora que se obtiene al tratar dos fases conjuntas en el proceso de planificación de sistemas de transporte público; por ejemplo, en este trabajo se evidenció que el tratamiento integrado de los problemas de calendarización de viajes y asignación de vehículos puede conducir a una disminución de los

costos del uso de las unidades de la flota. Esto debido a que se trabajó con una configuración de parámetros en la función objetivo que ordenaban al modelo tener más atención en el número de vehículos a utilizar.

- Los experimentos que se enfocaron en aumentar los parámetros $\gamma_{3,4}$ muestran cómo, aunque se aumente el tamaño de la flota, los tiempos y/o duraciones de los viajes vacíos disminuyen, pues el modelo minimiza el valor asociado a estos parámetros. Además, se evidencia que los valores de tiempo de viaje aumentan si se da un mayor enfoque a un parámetro o a otro.
- Pese a los beneficios en las soluciones que la parte integrada pueda ofrecer, la dificultad computacional de los problemas integrados se incrementa substancialmente respecto a sus contrapartes individuales, lo que resulta en la imposibilidad de resolver instancias grandes de manera directa, incluso empleando los solvers computacionales más actualizados.
- Pese a que se encontró una configuración del modelo integrado que parece adecuada para la función objetivo probada, esto no es generalizable para distintas variaciones de los parámetros (β, γ) . En efecto, la Tabla 4.3.2 muestra un buen desempeño del modelo integrado en todas las instancias para una función objetivo y configuración del solver específicas; sin embargo, la misma configuración para otra función objetivo tiene menor calidad en el desempeño (ver Tabla 4.2 y Tabla 4.10).
- La Tabla 4.10 evidencia cómo los cambios en los parámetros de ponderación afectan al resultado del modelo. En cada una de las instancias se observa que cada componente varía según el modelo que se esté utilizando y el enfoque que está teniendo dicho modelo en la función objetivo. Por ejemplo, el segundo modelo alcanza la menor cantidad de tiempo total de viaje de los pasajeros pues se agrega un costo $\beta > 0$ a la cantidad de tiempo que un pasajero pasa en el sistema y por tanto no se desprecia. El modelo 3 por el contrario aumenta los valores de tiempo total de viaje de los pasajeros pero da soluciones con duraciones de viajes y viajes vacíos mucho más

cortas que los otros modelos. Esto enfatiza que se pueden obtener diversidad de soluciones enfocándose en los objetivos que se tenga en la optimización.

- La integración de más fases de planificación, como por ejemplo agregar la planificación de líneas, está fuera del alcance de este trabajo de investigación. Se intuye que la complejidad computacional de los modelos se incrementará aún más, y se requerirá por tanto del uso de métodos heurísticos específicos para dicho problema, lo cual puede tratarse en trabajos de investigación futuros.
- Las instancias presentadas, aunque pequeñas, ilustraron bien cómo se puede comportar el modelo en distintas redes. Por ejemplo, aunque la instancia Toy2 fue más pequeña que la instancia Grid en cuanto al número de estaciones, el modelo integrado tuvo mayor dificultad en resolver Toy2.
- Con el fin de examinar en mayor detalle la particularidad de la instancia Athens, se comparan la cantidad de pasajeros y el número de líneas de esta instancia y la segunda instancia más grande (instancia Grid).

Característica	Grid	Athens	Razón
Líneas	34	96	2.82
Cargas	37 380	5 546 956	148.39
Variables	137 158	1 172 942	8.55

Tabla 5.1: Comparación de algunos atributos de las instancias Grid y Athens.

Como se puede observar, las líneas a tratar y el número de cargas sobre la red se incrementan substancialmente de la instancia Grid a Athens. Esto explica por qué se tiene mayor número de variables en la segunda instancia y cuán complejo es resolverla de forma óptima.

- En trabajos futuros, se puede analizar una posible alteración del orden secuencial en que se pueden resolver las fases de la planificación del sistema de transporte público. Por ejemplo, en [29] se propone obtener una asignación de vehículos directamente del plan

de líneas, mientras que en [26] se consideran diversas maneras en como pueden interactuar las distintas fases.

- Otra posible línea de trabajo futuro consiste en explorar métodos heurísticos específicos para el problema integrado. Esto debido a que durante los experimentos se evidenció cómo el solver encuentra más difícil hallar una solución del modelo integrado a medida que la instancia crece. De manera particular, el modelo sin modificaciones empieza a ser intratable de forma directa para la instancia Grid. Más aún, se torna más difícil resolver la instancia Athens que, aunque es la más grande tratada en este trabajo, aún es pequeña en comparación con instancias de sistemas de transporte de otras ciudades.
- Se pueden correr diversidad de experimentos para distintas configuraciones de los parámetros del solver como se hizo para los parámetros $(\beta, \gamma) = (1, 1, 1, 1, 1, 100)$ a fin encontrar una calibración adecuada del modelo. Esto demandará mayor tiempo y por eso no se lo implementó en el marco del presente trabajo.
- Se puede ser más meticulosos a la hora de construir la red Evento-Actividad. Una posible línea de trabajo consiste en desarrollar métodos para analizar qué arcos de la red son innecesarios. Esto a fin de reducir el tamaño de las instancias y mejorar el desempeño del modelo.

Anexo A

Redes evento-actividad para las redes de transporte público

En este capítulo se presentan las redes Evento-Actividad de las distintas instancias. Solo se agregan las líneas no dirigidas y se eliminan arcos de transferencia innecesarios, por ejemplo, del tipo $((v, arr, l_1), (v, dep, l_1^<))$ o $((v, arr, l_1^{(1)}), (v, dep, l_1^{(2)}))$, esto para tener una mejor visualización de la red y sus distintos arcos.

A.1. Small

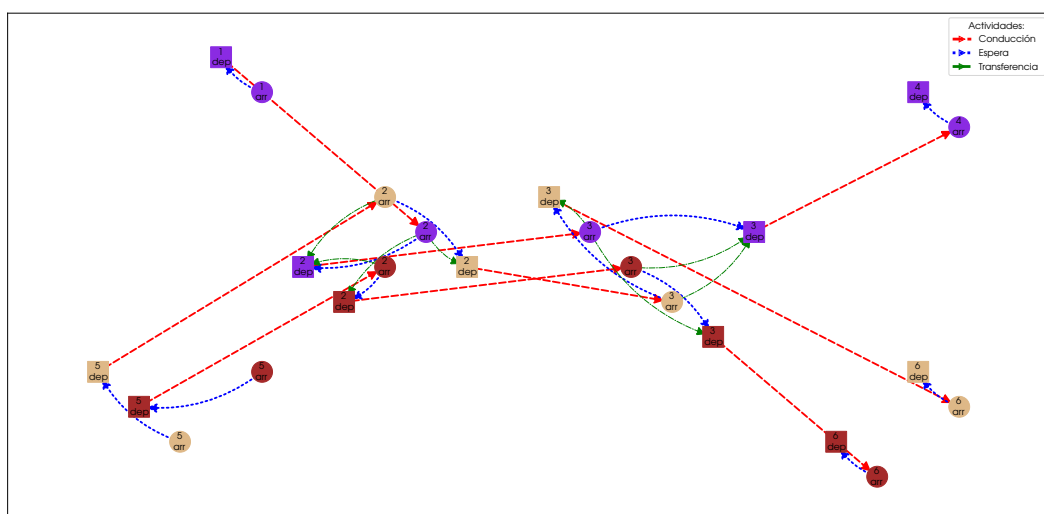


Figura A.1: EAN de la instancia Small.

A.2. Toy0

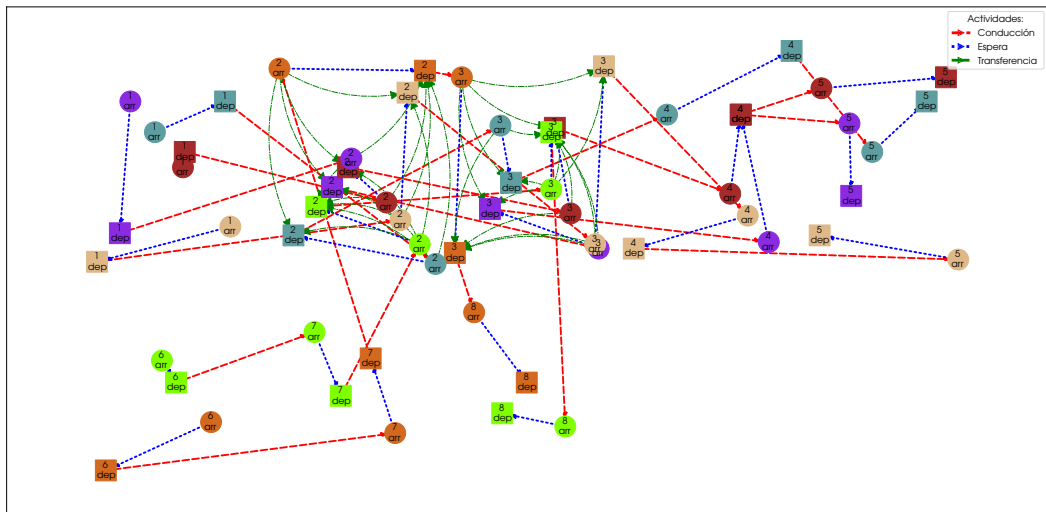


Figura A.2: EAN de la instancia Toy0.

A.3. Toy

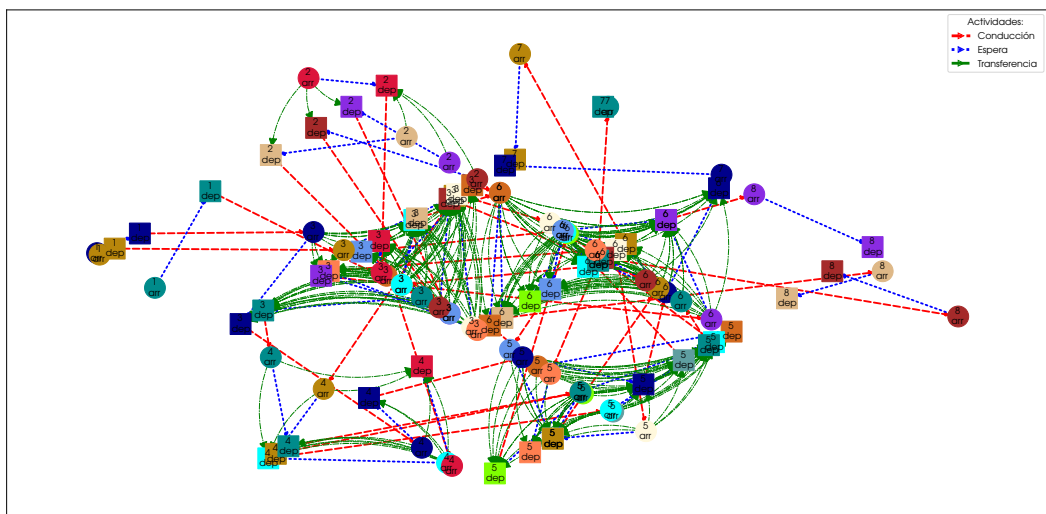


Figura A.3: EAN de la instancia Toy.

A.4. Toy2

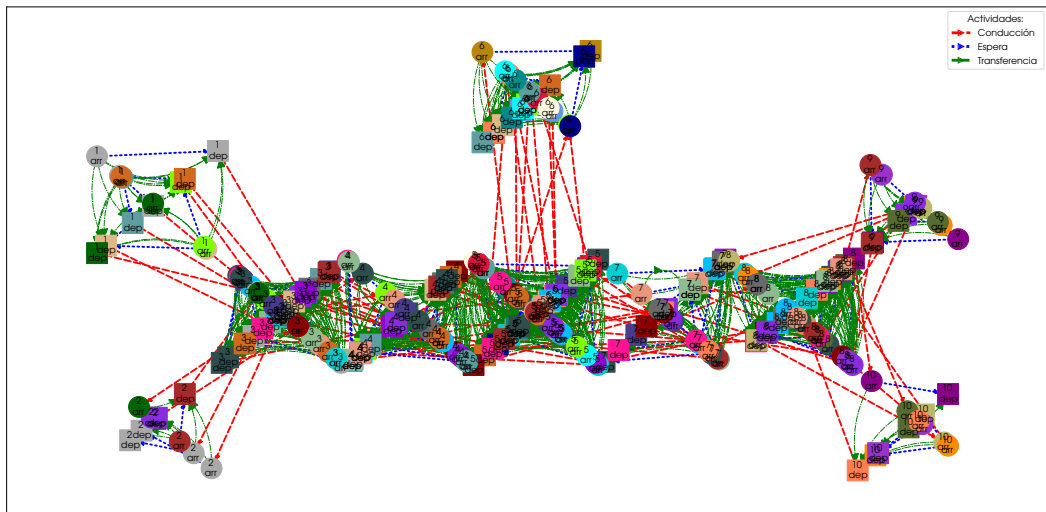


Figura A.4: EAN de la instancia Toy2.

A.5. Grid

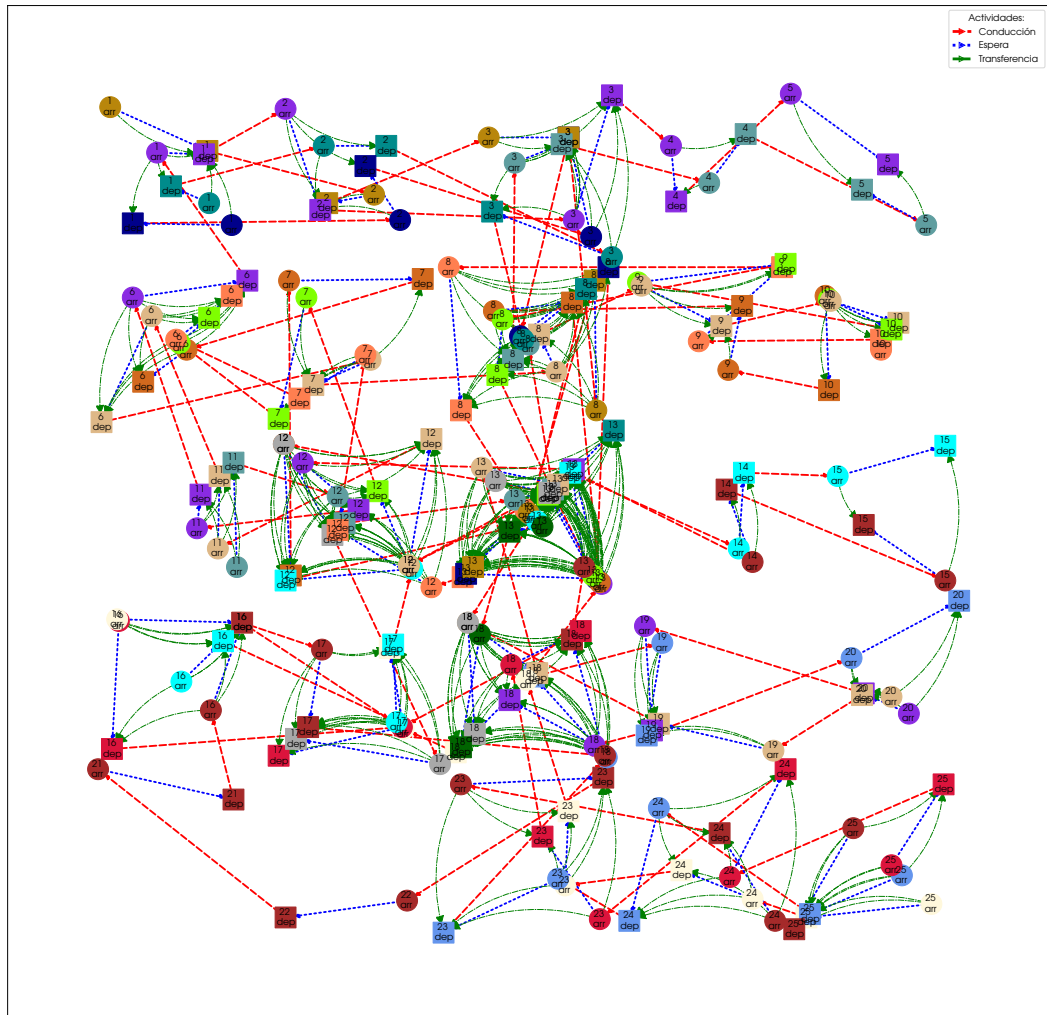


Figura A.5: EAN de la instancia Grid.

A.6. Athens

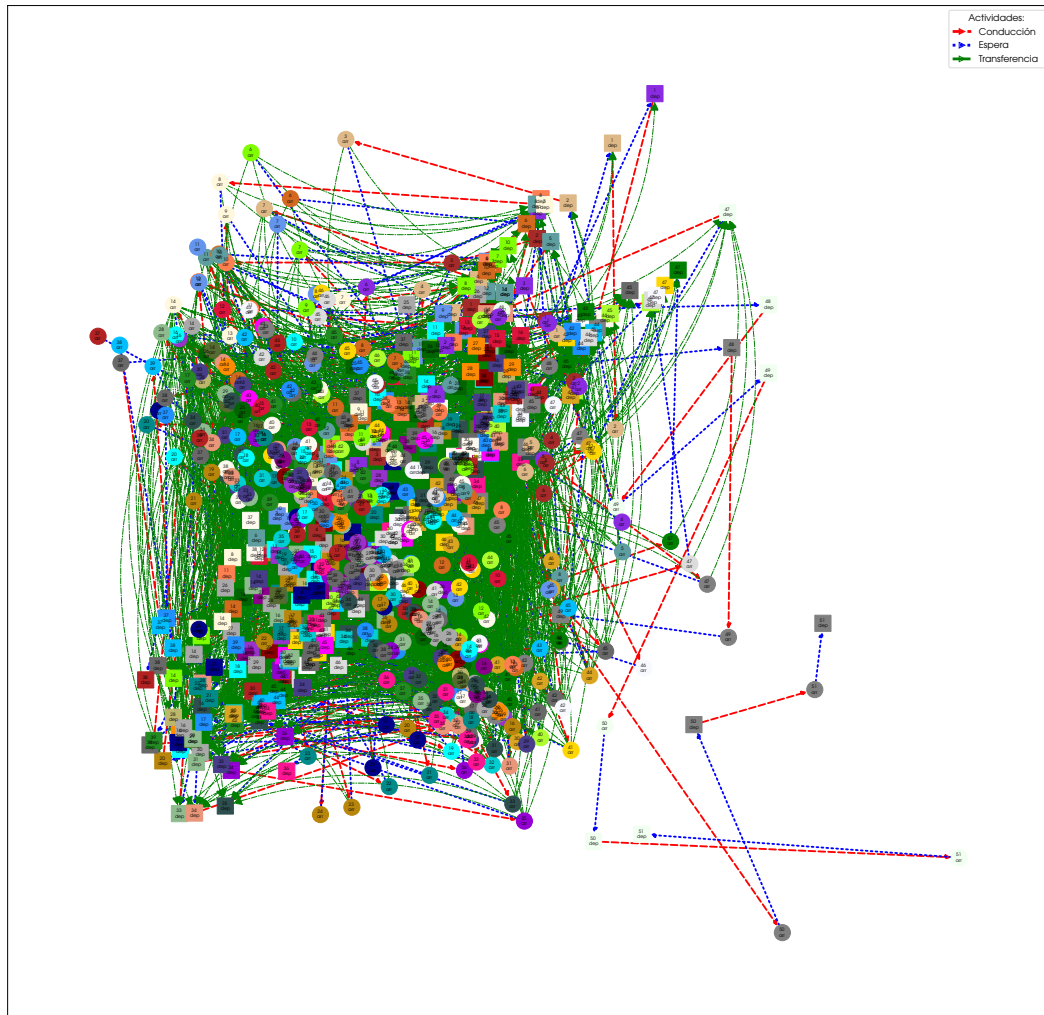


Figura A.6: EAN de la instancia Athens.

Referencias bibliográficas

- [1] Alan A Bertossi, Paolo Carraresi, and Giorgio Gallo. On some matching problems arising in vehicle scheduling models. *Networks*, 17(3):271–281, 1987.
- [2] Ralf Borndörfer, Marika Karbstein, Christian Liebchen, and Niels Lindner. A simple way to compute the number of vehicles that are required to operate a periodic timetable. In *18th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.
- [3] Paul Bouman, Alexander Schiewe, and Philine Schiewe. A New Sequential Approach to Periodic Vehicle Scheduling and Timetabling. In Dennis Huisman and Christos D. Zaroliagis, editors, *20th Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2020)*, volume 85 of *OpenAccess Series in Informatics (OASICs)*, pages 6:1–6:16, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.
- [4] Stefan Bunte and Natalia Kliewer. An overview on vehicle scheduling models. *Public Transport*, 1(4):299–317, 2009.
- [5] Samuela Carosi, Antonio Frangioni, Laura Galli, Leopoldo Girardi, and Giuliano Vallese. A matheuristic for integrated timetabling and vehicle scheduling. *Transportation Research Part B: Methodological*, 127:99–124, 2019.

- [6] Samuela Carosi, Antonio Frangioni, Laura Galli, Leopoldo Girardi, and Giuliano Vallese. A matheuristic for integrated timetabling and vehicle scheduling. *Transportation Research Part B: Methodological*, 127:99–124, 2019.
- [7] Anabela Costa, Isabel Branco, and Jose M Pinto Paixão. Vehicle scheduling problem with multiple type of vehicles and a single depot. In *Computer-aided transit scheduling*, pages 115–129. Springer, 1995.
- [8] Joachim R Daduna and José M Pinto Paixão. Vehicle scheduling for public mass transit—an overview. *Computer-aided transit scheduling*, pages 76–90, 1995.
- [9] A El-Azm. The minimum fleet size problem and its applications to bus scheduling. from the book computer scheduling of public transport 2. 1985.
- [10] FOR2083. Public transport networks, collection of open source public transport networks by dfg research unit. <https://github.com/FOR2083/PublicTransportNetworks>, 2019.
- [11] MA Forbes, JN Holt, and AM Watts. An exact algorithm for multiple depot bus scheduling. *European Journal of Operational Research*, 72(1):115–124, 1994.
- [12] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2022.
- [13] Bernhard H Korte, Jens Vygen, B Korte, and J Vygen. *Combinatorial optimization*, volume 1. Springer, 2011.
- [14] WHK Lam and MGH Bell. Advanced modeling for transit operations and service planning.(new book). *OR/MS Today*, 29(6):62–63, 2002.
- [15] Andreas Löbel. *Optimale vehicle scheduling in public transit*. PhD thesis, 1997.
- [16] Richard M Lusby, Jesper Larsen, Matthias Ehrgott, and David Ryan. Railway track allocation: models and methods. *OR spectrum*, 33(4):843–883, 2011.

- [17] Gábor Maróti. Operations research models for railway rolling stock planning, 2006.
- [18] Miika Niemi. Fleet size formula. <https://github.com/MiikaNiemi/fleet-size-formula>, 2021.
- [19] Clifford S Orloff. Route constrained fleet scheduling. *Transportation Science*, 10(2):149–168, 1976.
- [20] Samuel Raff. Routing and scheduling of vehicles and crews: The state of the art. *Computers & Operations Research*, 10(2):63–211, 1983.
- [21] Markus Reuther and Thomas Schlechte. Optimization of rolling stock rotations. In *Handbook of Optimization in the Railway Industry*, pages 213–241. Springer, 2018.
- [22] Markus Reuther and Thomas Schlechte. Optimization of rolling stock rotations. In *Handbook of Optimization in the Railway Industry*, pages 213–241. Springer, 2018.
- [23] J.L. Saha. An algorithm for bus scheduling problems. *Journal of the Operational Research Society*, 21(4):463–474, 1970.
- [24] Alexander Schiewe, Sebastian Albert, Philine Schiewe, Anita Schöbel, and Felix Spühler. Documentation for lintim 2020.12. *Homepage*. see <https://gitlab.rlp.net/lintim/OpenLinTim/-/tree/master/datasets>, 2020.
- [25] Philine Schiewe et al. *Integrated optimization in public transport planning*. Springer, 2020.
- [26] Anita Schöbel. An eigenmodel for iterative line planning, timetabling and vehicle scheduling in public transportation. *Transportation Research Part C: Emerging Technologies*, 74:348–365, 2017.
- [27] Alexander Schrijver et al. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer, 2003.
- [28] Paolo Serafini and Walter Ukovich. A mathematical model for periodic scheduling problems. *SIAM Journal on Discrete Mathematics*, 2(4):550–581, 1989.

- [29] Rolf N. van Lieshout and Paul C. Bouman. Vehicle Scheduling Based on a Line Plan. In Ralf Borndörfer and Sabine Storandt, editors, *18th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2018)*, volume 65 of *OpenAccess Series in Informatics (OASICs)*, pages 15:1–15:14, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.