Transit Network Timetabling Problem: A Case Study of the Transport Operator in Cinu (Colombia)

Andrés Rueda López[a,*], Juan Sebastian Agudelo Gomez[b]

[a] *Universidad de Los Andes, Bogotá, Colombia*
[b] *Universidad de Los Andes, Bogotá, Colombia*

Abstract

Keywords: Networks, Optimization, Transportation, Public transport systems, Transit Network Timetabling Problem

1. Introduction

Public transport systems (PTS) play an important role in the development of urban populations and have a significant influence on the quality of life for residents. In recent years, the need to develop more efficient PTS has grown, as it has become evident that the economic and environmental performance of cities can be improved by efficiently connecting resources and mass population mobility [1].

One of the main concerns in developing PTS and making accurate operational decisions is measuring the temporal and spatial coverage of the systems for the population. While spatial coverage or accessibility refers to ensuring that a large number of people have access to a transit station within a short distance, temporal coverage means that PTS are available when customers need the service [2].

The increasing demand for PTS, driven by the rapid growth of urban populations, implies that PTS planning becomes exponentially more complex. If this systems are unable to adapt rappidly and in an efficient way to those changes, it can result in an increase of waiting times or PTS congestions [3]. For this reason, it is essential that Public Transport Companies (PTC) allocate some of their resources to develop smart planning strategies that can quickly and accurately adapt to the needs of users.

A robust plan is the base for a PTC to satisfy the demand, give users the best possible experience, and guarantee a fair working policy for their drivers. A common transit operations planning process consists of four stages: first, designing the route network system; second, developing a timetable for each route; third, scheduling vehicles to match the timetable trips; and finally, crew scheduling [4].

This paper focuses on the second stage of the robust planning process by designing a bus timetable for one route of the *Transport Operator of Cinu (Colombia)*, dealing with what is known as a *Transit Network Timetabling Problem* (TNTP). This involves scheduling a timetable to operate over pre-planned routes, where all vehicles depart from and arrive at a single depot. This well-know problem can typically be solved in two possible ways, periodic timetabling represented as $trips/hour$ and non-periodic timetabling where each trip has a fixed departure time [5].

The proposed solution should benefit the company by either reducing operational costs or increasing revenue through an efficient and replicable implementation across different routes. However, because the company provides a public service, it is equally important that the resulting solution guarantees temporal accessibility for users. Since users always have a preferred departure time, aligning the service with this preference in demand behavior will increase user satisfaction and should also be considered in the problem solution [6]. The common strategies to achieve a solution for this kind of problems involve exact and/or approximate methods.

Exact methods often formulate timetabling as an optimization problem. For complex combinatorial problems, classic optimization techniques such as column generation are commonly used. These methods enable the creation of variables without the need to explore the entire solution space directly, facilitating the generation of efficient schedules or shifts [7]. Additionally, given the mixed-integer nature of these problems, computational acceleration techniques such as branch-and-cut can be applied, utilizing both branching and cutting to enhance solution efficiency [8]. It has also been shown that these techniques are beneficial when applied to scheduling problems formulated in networks with resource constraints [9].

Approximation methods based on heuristics offer an alternative for reducing the complexity of solving transit network timetable problems (TNTP). Heuristics often frame TNTP as a graph problem, where nodes represent feasible departure times and arcs indicate possible sequences of these times, transforming the solution into a shortest path problem. In [10], a multi-objective label-correcting algorithm is proposed to minimize both user waiting time (service metric) and the number of empty seats (operations metric) by comparing near-optimal solutions using a Pareto strategy.

*Corresponding author.

In recent years, *Reinforcement Learning* (RL) has emerged as a method to solve timetabling problems in real-time. RL treats timetabling as a Markov Decision Process (MDP), where future decisions depend only on the current state [11]. RL agents interact with simulated environments, receiving rewards for actions like departure timing. In [12] is proposed a reward function that aims to optimize bus utilization while minimizing waiting times, using a Deep Q-Network (DQN). However, RL agents may select infeasible actions, requiring a masking step to ensure that only valid actions are considered [13].

When dealing with large-scale scenarios, formulating and solving optimization problems can become challenging. In these cases, it is common to adopt a standard formulation. Most bus scheduling formulations aim to minimize user waiting times while adhering to resource constraints [14]. One formulation that follows this for a single-direction bus line is presented in [15]. ~~as follows~~:

$$(Q) \quad \cancel{\min \sum_{p\in\mathcal{P}} w_p} \tag{1}$$

$$\cancel{x_1} = \cancel{D_f} \tag{2}$$

$$\cancel{x_k} = \cancel{D_l} \tag{3}$$

$$\cancel{x_k} \leq \cancel{x_{k+1}} \tag{4}$$

$$\cancel{V_{1k}} = \cancel{x_k} \tag{5}$$

$$\cancel{V_{mk}} = \cancel{V_{m-1,k}} + \cancel{\lambda_{m-1,j}} \tag{6}$$

$$\cancel{\tau_p} = \min_k(\cancel{V_{Bpk}} > \cancel{A_p}) \tag{7}$$

$$\cancel{w_p} = \cancel{V_{B_p\tau_p} - A_p} \tag{8}$$

~~Equation (1) minimizes passenger waiting times throughout the day. Equations (2) and (3) ensure the first and last trips are dispatched according to the schedule. Equation (4) maintains the order of trips. Equation (5) defines the dispatch time of the first trip. Equation (6) guarantees accurate arrival times at each stop. Equation (7) assigns passengers to buses, and Equation (8) calculates the waiting time for each passenger.~~

There are several extensions to the standard formulation ~~discussed earlier~~ presented in [15], such as the one presented by [14], where the author introduces additional constraints, including vehicle availability for successive trips, vehicle capacities, and permitted dispatching headway variations.

This paper aims to ~~explores the application of the standard formulation proposed by [15]~~ model and optimize one bus line of the Transport Operator of Cinu (Colombia), using real-world data provided by the company. ~~Beyond adapting the model to specific case constrains~~, This article introduces a novel reformulation, replacing the MIP approach from authors like [15] with a network-based model. This reformulation leverages the computational efficiency of network optimization, particularly for integer problems, aiming to deliver scalable solutions that can be easily extended to larger transport networks, thus avoiding computational limitations [16].

2. Problem statement

Consider a route in the system represented as a directed graph $G = (\mathcal{N}, \mathcal{A}, \mathcal{T})$, where $\mathcal{N} = \{s, 1, 2, \ldots, e \ldots, 1', 2', \ldots\}$ is the set of nodes, $\mathcal{A} \subseteq \{(i, j) \mid i \in \mathcal{N}, j \in \mathcal{N}, i \neq j\}$ is the set of arcs, and $\mathcal{T} = \{0, \varphi, \ldots, t_{\max}\}$ is the set of time intervals, each of length φ. Let $P = \{p_1, p_2, \ldots, p_m\}$ be the set of buses available for the route, each with an identical capacity k. Every bus $p \in P$ must begin the route at the node $s \in \mathcal{N}$, and it may depart at any time $t \in \mathcal{T}$.

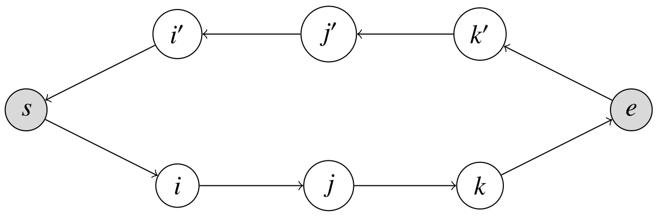

Figure 1: General form of the route

A path Ω is defined as an ordered sequence of nodes, representing the order that each bus $p \in P$ must follow while traveling along $\mathcal{G}$. Each bus $p \in P$ must return to node $s \in \mathcal{N}$ by the end of the time horizon, under the assumption that the parking spot is only available at this node. This implies that every bus that begins the route must complete the entire cycle in $\mathcal{G}$ before finishing.

Let $y_{(i,j),p,t}$ be a binary variable that indicates whether arc (i, j) is traversed by bus $p \in P$ at time $t \in \mathcal{T}$. Specifically, if $y_{(i,j),p,t} = 1$, it means that at time t, bus p is at node i and has begun its journey towards node j, allowing passengers at node i to board. For simplicity, the boarding time of passengers is considered negligible.

The traversal time of arc (i, j) is assumed to be deterministic and provided, which is then converted into time units φ and denoted by $\tau_{i,j}$. Additionally, each node $i \in \mathcal{N}$ has an associated parameter $w_{i,t}$, which is assumed as a deterministic value and is an estimation of the number of passengers arriving at node i at time $t \in \mathcal{T}$. By defining $W_{i,t}$ as the number of people at node $i \in \mathcal{N}$ at time $t \in \mathcal{T}$, it is possible to model the number of people for each arc at each time instant as:

$$W_{i,t} = w_{i,t} - \left[\sum_{p\in P}(k - s_{i,p,t})y_{(i,j),p,t}\right], \quad \forall (i, j) \in \mathcal{A},\ t \in \mathcal{T} \mid t = 0 \tag{9}$$

$$W_{i,t} = W_{i,(t-1)} + w_{i,t} - \left[\sum_{p\in P}(k - s_{i,p,t})y_{(i,j),p,t}\right], \quad \forall (i, j) \in \mathcal{A},\ t \in \mathcal{T} \mid t > 0 \tag{10}$$

Where $s_{i,p,t}$ represents the amount of people in the bus $p \in P$ at the time $t \in \mathcal{T}$ in the node i. To calculate the

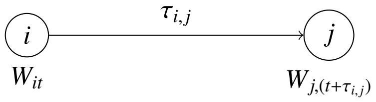

Figure 2: Atributes for each section of the route

charge of each bus at each time, let u be a parameter that indicates the proportion of people in the bus that get off in each node, assuming that every node, is indeferent for people.

$$s_{i,p,t} = \lfloor s_{(i-1),p,(t-\tau_{i,i-1})}(1 - uy_{(i,j),p,t}) + (k - W_{i,t})\rfloor \quad (11)$$

Because the amount of people that arrives to each node is out of control for the decision maker, what can be achieved is the amount of people that is waiting in each node. For this reason, the metric in the system, denoted by z, represents the total number of people that is waiting for transportation and can be calculated as follows:

$$Q = \sum_{(i)\in\mathcal{N}} \sum_{t\in\mathcal{T}} W_{i,t} \quad (12)$$

For sumarize the timetabling problem, it can be writen as:

$$\arg\min_{y_{(i,j),p,t}} \sum_{(i)\in\mathcal{N}} \sum_{t\in\mathcal{T}} W_{i,t} \quad (13)$$

In the following section, the problem is translated into an optimization model that aims to create a structure that can be implemented using the data provided by the company.

2.1. Optimization Model

An optimization model is proposed as an approximation to the previous problem. This approximation aims to generalize to any route. The approach preserves the linearity of the problem constraints by linearizing specific conditions, thereby enabling the use of accessible linear solvers to solve the M.I.P.

2.1.1. Sets

- $\mathcal{N}$: Set of nodes (or stations), $\mathcal{N} = \{0, 1, 2, \dots, e\}$.
- $\mathcal{P}$: Set of buses, $\mathcal{P} = \{p_1, p_2, \dots, p_m\}$.
- $\mathcal{T}$: Set of time intervals, $\mathcal{T} = \{0, \varphi, 2\varphi, \dots, t_{\max}\}$.
- $\mathcal{A}$: Set of arcs, $\mathcal{A} \subseteq \{(i, j) | i \in \mathcal{N}, j \in |, i \neq j\}$.

2.1.2. Parameters

- $d_{i,t}$: Number of people arriving at node $i \in \mathcal{N}$ at time $t \in \mathcal{T}$ [Units].
- $\tau_{i,j}$: Travel time from node $i \in \mathcal{N}$ to node $j \in \mathcal{N}$ [Time units φ].
- k: Capacity of any bus [Units].
- u: Rate of decline at any node, $u \in [0, 1]$.
- I^0: Initial charge of any bus [Units], with $I^0 = 0$.
- $W_{i,t}$: Initial amount of people at node $i \in \mathcal{N}$ at time $t \in T$.
- M: Big number.
- K: Max number of cycles that can be made by any bus.

2.1.3. Decision Variables

- $x_{i,j,t,p} = \begin{cases} 1 & \text{if the bus } p \in \mathcal{P} \text{ starts the route along the arc } (i, j) \in \mathcal{A} \\ & \text{at time } t \in \mathcal{T}, \\ 0 & \text{otherwise.} \end{cases}$
- $y_{i,j,t,p} = \begin{cases} 1 & \text{if the bus } p \in \mathcal{P} \text{ is travelling along the arc } (i, j) \in \mathcal{A} \\ & \text{at time } t \in \mathcal{T}, \\ 0 & \text{otherwise.} \end{cases}$
- $z_{i,t,p} = \begin{cases} 1 & \text{if the bus } p \in \mathcal{P} \text{ is at node } i \in \mathcal{N} \\ & \text{at time } t \in \mathcal{T}, \\ 0 & \text{otherwise.} \end{cases}$
- $w_{i,t,p}$: Amount of demand atended in the node $i \in \mathcal{N}$ at time $t \in \mathcal{T}$ by the bus $p \in \mathcal{P}$.
- $o_{i,t,p}$: Amount of people decending in the node $i \in \mathcal{N}$ at time $t \in \mathcal{T}$ from the bus $p \in \mathcal{P}$.
- $s_{t,p}$: Amount of people at time $t \in \mathcal{T}$ in the bus $p \in \mathcal{P}$.
- $W_{i,t}$: Non attended demand in the node $i \in \mathcal{N}$ at time $t \in T$.

2.1.4. Constraints

$$\min \sum_{i\in\mathcal{N}} \sum_{t\in\mathcal{T}} W_{i,t} \quad (1)$$

s.t

$$\sum_{t\in T} x_{i,j,t,p} \leq K, \quad \forall (i, j) \in \mathcal{A}, p \in P | i = 0, j = i + 1 \quad (2)$$

$$\sum_{t\in T} x_{i,j,t,p} = 0, \quad \forall p \in P(i, j) \in \mathcal{A} | i = 0, j = i + 1 \quad (3)$$

$$\begin{aligned} y_{i,j,\delta,p} \leq x_{i,j,t,p}, \quad & \forall\, (i, j) \in \mathcal{A},\ t \in \mathcal{T},\ p \in \mathcal{P}, \\ & \delta \in \{t, t + 1, \dots, \min(t + \tau_{i,j}, |\mathcal{T}| + 1)\} \end{aligned} \quad (4)$$

Constraints (2)-(4) define the conditions for the initiation of the route and the activation of the arcs.

$$z_{i,t,p} + z_{j,t,p} \geq y_{i,j,t,}, \quad \forall (i, j) \in \mathcal{T}, t \in \mathcal{T}, p \in \mathcal{P} \quad (5)$$

$$\sum_{t\in\mathcal{T}} z_{i,t,p} \leq 1, \quad \forall p \in \mathcal{P}, i \in \mathcal{N} | i \neq 0 \quad (6)$$

$$\sum_{n\in\mathcal{N}} z_{i,t,p} \leq 1, \quad \forall t \in \mathcal{T}, p \in \mathcal{P} \quad (7)$$

$$z_{i,t,p} = z_{i,t-1,p} - \sum_{j\in\mathcal{N}} x_{i,j,t-1,p}, \forall p \in \mathcal{P}, t \in \mathcal{T} | i = 0 \quad (8)$$

$$z_{i,t,p} = \sum_{(k,i)\in\mathcal{A}} y_{k,i,t-1,p}, \quad \forall t \in \mathcal{T}, p \in \mathcal{P} \quad (9)$$

$$z_{i,t,p} = 1, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} | i = 0, t = 0 \tag{10}$$

$$z_{i,t,p} = 0, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} | t = 0 \tag{11}$$

$$\sum_{t \in \mathcal{T}} z_{i,t,p} = 1, \quad \forall i \in \mathcal{N}, p \in \mathcal{P} | i = e \tag{12}$$

$$\sum_{(i,j) \in \mathcal{A}} x_{i,j,t,p} \leq z_{i,t,p}, \quad \forall t \in \mathcal{T}, p \in \mathcal{P} \tag{13}$$

Constraints (6) through (13) define the relationships between the presence variable $z_{i,t,p}$ and the variables $x_{i,j,t,p}$ and $y_{i,j,t,p}$. These constraints model the presence of the bus at each node within the graph.

$$s_{t,p} = s^0 + \sum_{n \in \mathcal{N}} w_{i,t,p} - \sum_{i \in \mathcal{N}} o_{i,t,p}, \quad \forall t \in \mathcal{T}, p \in \mathcal{P} | t = 0 \tag{14}$$

$$s_{t,p} = s_{t-1,p} + \sum_{n \in \mathcal{N}} w_{i,t,p} - \sum_{i \in \mathcal{N}} o_{i,t,p}, \quad \forall t \in \mathcal{T}, p \in \mathcal{P} | t > 0 \tag{15}$$

$$s_{t,p} \leq k, \quad \forall t \in \mathcal{T}, p \in \mathcal{P} \tag{16}$$

Constraints (14) and (16) model the bus charge along the time horizon for each bus.

$$W_{i,t} = d_{i,t} - \sum_{p \in P} w_{i,t,p}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T} | t = 0 \tag{17}$$

$$W_{i,t} = W_{i,t-1} + d_{i,t} - \sum_{p \in P} w_{i,t,p}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T} | t > 0 \tag{18}$$

$$w_{i,t,p} \leq W_{(i,t)}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} \tag{19}$$

$$W_{i,t,p} \leq M z_{i,t,p}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} \tag{20}$$

$$\sum_{n \in \mathcal{N}} w_{i,t,p} \leq k - s^0, \quad \forall t \in \mathcal{T}, p \in \mathcal{P} | t = 0 \tag{21}$$

$$\sum_{n \in \mathcal{N}} w_{i,t,p} \leq k - s_{t-1}, \quad \forall t \in \mathcal{T}, p \in \mathcal{P} | t = 0 \tag{22}$$

Constraints (17) through (22) model the non attended and new demand for each node.

$$o_{i,t,p} \leq s_{t-1,p}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} | t > 1 \tag{23}$$

$$o_{i,t,p} \leq k z_{i,t,p}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} | t > 1 \tag{24}$$

$$o_{i,t,p} \leq k(1 - z_{i,t,p}), \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} | i = 0, i = e, t > 1 \tag{25}$$

$$o_{i,t,p} = 0, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} | t = 0 \tag{26}$$

$$o_{i,t,p} \leq u_i s_{t-1,p} \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} \tag{27}$$

Where constraints (23) through (27) model the descent of the people from the buses in each node.

$$x_{i,j,t,p} \in [0,1], \quad \forall (i,j) \in \mathcal{A}, t \in \mathcal{T}, p \in \mathcal{P} \tag{28}$$

$$y_{i,j,t,p} \in [0,1], \quad \forall (i,j) \in \mathcal{A}, t \in \mathcal{T}, p \in \mathcal{P} \tag{29}$$

$$z_{i,t,p} \in [0,1], \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} \tag{30}$$

$$w_{i,t,p} \in \mathbb{Z}^+ \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} \tag{31}$$

$$s_{t,p} \in \mathbb{Z}^+ \quad \forall t \in \mathcal{T}, p \in \mathcal{P} \tag{32}$$

$$o_{i,t,p} \in \mathbb{Z}^+ \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} \tag{33}$$

$$W_{i,t} \in \mathbb{Z}^+ \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, p \in \mathcal{P} \tag{34}$$

(28) through (34) express the nature of the variables.

3. Implementation for a Basic Instance

A small instance was built to test the previous model in a simple case. For this case, the parameters were generated artificially.

3.0.1. Graph

The graph illustrated below is structured with the node s serving as the starting point of the route, leading sequentially through a series of connected nodes, culminating at node 6, which is designated as the endpoint of the route. This layout provides a clear pathway from the origin to the destination, highlighting the directional flow between nodes.

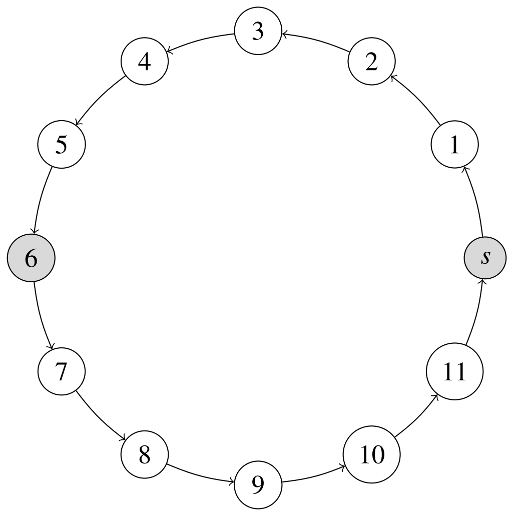

Figure 3: Example graph

For the temporal distance between each pair of connected nodes (i, j) within the set of arcs $\mathcal{A}$ is uniformly defined as $\tau_{i,j} = 1$, ensuring that transition times are consistent across all connections.

The demand at each node during different times is generated by summing up individual normal distributions, capturing the temporal variations and characteristics of demand effectively. This modeling approach allows us to reflect the fluctuating nature of demand as an approximation (see the ilustration for details).

Furthermore, the proportion of descent at each node is determined by a stochastic process, where each value is a uniformly distributed random number between 0 and 1. This method introduces an element of randomness into the model, influencing the dynamics and outcomes of the system in a probabilistically varied manner. Finally, the model was tested for 3 buses with equal capacity ($k = 200$) in set of 50 time intervals.

3.0.2. Results

The computational experiment was performed on a device named IIIND-PIRYBKR3. The system features a 12th Generation Intel(R) Core(TM) i7-1265U processor with a base clock speed of 1.80 GHz. The machine is equipped with 16 GB of installed RAM (15.7 GB usable) and runs a 64-bit operating system based on the x64 architecture.

The optimization model was implemented and solved using the PuLP library, a linear programming package in

Python (3.11.5 version). The problem was solved using the CBC (Coin-or branch and cut) solver, a free and open-source mixed-integer programming solver. This approach was executed on a computational setup described previously, ensuring adequate power for processing complex optimization tasks.

The solution process took approximately 49 minutes and 51 seconds, indicating the scale and complexity of the model. The solver successfully found an optimal solution, achieving an objective value of 18,147.0, which represents the total non attended demand in the graph.

Table 1: Bus Departure Times

Bus	Departure Time	Arrival Time
Bus 1	7	19
Bus 2	15	22
Bus 3	21	33

In Table 1, the departure times from the initial node and the subsequent returns to these node are presented. Given the simplicity of the assumptions, these times significantly influence the deterministic nature of the problem, thereby determining the total solution.

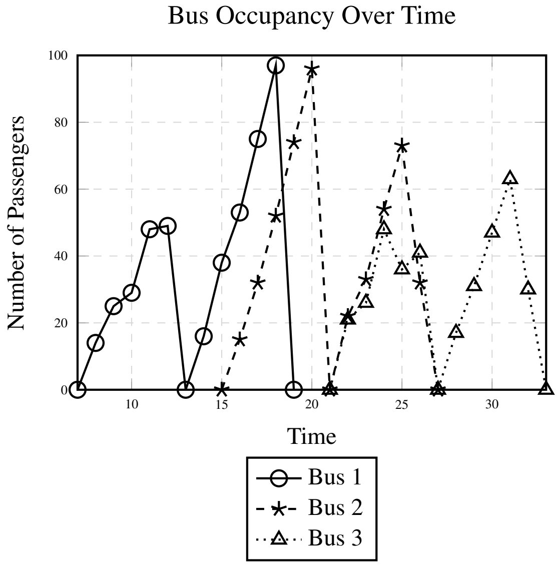

Figure 4: Bus occupancy over the time

Figure 4 illustrates the occupancy levels of each bus from the departure to the return times. Notably, the occupancy is zero at the beginning, midpoint, and end of the route, corresponding to the terminal nodes. This pattern highlights the nodes where each bus journey either starts or concludes.

Finally, the reader can view an animation of the travel process along the graph for each bus in the supplementary online materials, accessible via the digital repository. This animation provides a dynamic representation of the bus occupancy data over time, illustrating the fluctuating passenger counts as the buses progress through their routes.link

References

[1] J. Doe, Public transport systems and urban mobility, Urban Mobility Journal 10 (3) (2021) 123–135. doi:10.1000/xyz123.

[2] S. A. Mamun, N. E. Lownes, J. P. Osleeb, K. Bertolaccini, A method to define public transit opportunity space, Journal of Transport Geography 28 (2013) 144–154. doi:10.1016/j.jtrangeo.2012.12.007.

[3] M. A. Saif, M. M. Zefreh, A. Torok, Public transport accessibility: A literature review, Periodica Polytechnica Transportation Engineering 47 (1) (2019) 36–43. doi:10.3311/PPtr.12072.

[4] A. Ceder, Public transit planning and operation : modeling, practice and behavior, CRC Press, 2015.

[5] J. Mesa, F. Ortega, M. Pozo, Locating optimal timetables and vehicle schedules in a transit line, Annals of Operations Research 222 (2013) 439–455.

[6] A. de Palma, R. Lindsey, Optimal timetables for public transportation, Transportation Research Part B: Methodological 35 (2001) 789–813.

[7] G. Desaulniers, J. Desrosiers, M. M. Solomon (Eds.), Column Generation, Springer Science & Business Media, 2005.

[8] L. A. Wolsey, Integer Programming, 1st Edition, Wiley, 1998.

[9] F. B. Talbot, J. H. Patterson, An efficient integer programming algorithm with network cuts for solving resource-constrained scheduling problems, Management Science 24 (11) (1978) 1163.

[10] S. Hassold, A. Ceder, Multiobjective approach to creating bus timetables with multiple vehicle types, Transportation Research Record 2276 (2012) 56–62.

[11] R. Bellman, A markovian decision process, Journal of Mathematics and Mechanics 6 (1957) 679–684.

[12] G. Ai, X. Zuo, G. Chen, B. Wu, Deep reinforcement learning based dynamic optimization of bus timetable, Applied Soft Computing 131 (109752) (2022).

[13] Y. Liu, X. Zuo, G. Ai, Y. Liu, A reinforcement learning-based approach for online bus scheduling, Knowledge-Based Systems 271 (110584) (2023).

[14] K. Gkiotsalitis, Bus scheduling considering trip-varying travel times, vehicle availability and capacity, IET Intelligent Transport Systems 14 (10) (2020) 1239–1246. doi:10.1049/iet-its.2019.0725.

[15] Y. Wang, D. Zhang, L. Hu, et al., A data-driven and optimal bus scheduling model with time-dependent traffic and demand, IEEE Transactions on Intelligent Transportation Systems 18 (9) (2017) 2443–2452. doi:10.1109/TITS.2017.2686015.

[16] R. K. Ahuja, T. L. Magnanti, J. B. Orlin, Network Flows: Theory, Algorithms, and Applications, Prentice Hall, Upper Saddle River, New Jersey, 1993.