ASSESSING METRO NETWORKS EFFICIENCY USING MAX-PLUS ALGEBRA

COMPLEXITY72H

Sami Aboumad[*1,2], Firas Blibeche[†3,4,5], Kishore Hari[‡6], Siwei Liu[§7], Sara Linde Neven[¶8], Aarathi Parameswaran[‖9,10,11], and Ebrahim Patel[**12]

[1]Université Paris-Saclay, Orsay, FR
[2]CENTAI Institute, Torino, IT
[3]Sorbonne Université, Paris, FR
[4]Politecnico di Torino, Torino, IT
[5]Department of Physics and Chemistry - Emilio Segrè, University of Palermo, Palermo, IT
[6]Center for Theoretical and Biological Physics, Northeastern University, Boston, USA
[7]Department of Engineering, University of Cambridge, Cambridge, UK
[8]Institute for Biodiversity and Ecosystem Dynamics, University of Amsterdam, Amsterdam, NED
[9]University of Bonn, Bonn, DE
[10]Institute for Energy and Climate Research (IEK-STE), Jülich, DE
[11]Institute for Theoretical Physics, University of Cologne, Köln, DE
[12]University of Greenwich, London, UK

October 7, 2024

ABSTRACT

A goal for designing public transport networks such as the metro or railway is to ensure a feasible commute for the public. Various variables are considered in such designs, including the number of lines to be added, the number of trains on each line, and the number of stops and intersections to optimize the commute time, structure of the network, passenger wait time, and access to various connections. As geography and the populations of most cities are dynamically changing, city transportation networks must be frequently optimized. Here, we present an approach to perform such an approach using max-plus algebra. Max-plus algebra is an alternative algebraic formulation that can convert non-linear equations involving max operations as linear. The fundamental assumption makes sure that passengers have access to all lines at the intersection points by setting the departure time to be the maximum of the arrival time of all trains coming to the stations. Using its convenient algebraic properties, such as eigenvalues and eigenvectors, we estimate the average waiting time for departures at each station. We then optimize the departure time by various methods. We demonstrate the applicability of our methods to the metro networks of Milan and Amsterdam, reducing the average waiting times significantly. We also perform delay propagation analysis to identify the key stations in the network where delays must be swiftly dealt with. We further find that random transportation networks generated using different methods have weaker performance as compared to real-world networks.

Keywords max-plus algebra · transportation networks · network optimization

[*]aboumads@gmail.com
[†]firasblibeche@gmail.com
[‡]a.hari@northeastern.edu
[§]sl2049@cam.ac.uk
[¶]s.l.neven@uva.nl
[‖]a.parameswaran@fz-juelich.de
[**]e.patel@greenwich.ac.uk

1 Introduction

Transportation networks have been a significant area of interest for network science, owing to the nature of cities being complex systems (Barthelemy [2016]) that are dynamic and expanding rapidly that requires the transportation systems within cities to accommodate for this growth. The movement of people and goods through transportation systems is a factor that influences mobility, development, economic activity and livability of cities. More specifically, public transport has a growing relevance with the push for a more sustainable means of transport for larger urban populations and as a competitor to the automobile industry. There is a need for attention particularly to the optimization of public transportation networks due to the growing demand of having more efficient and reliable public transport with the expansion of cities and increase in congestion within these networks. In such dense networks that are highly interconnected, a single failure or delay can have a cascading effect to the entire network (Goverde [2007]) and this warrants a study of the propagation of such delays and minimizing their effects. A recent study by Wu et al. [2023] has performed delay propagation for the Chinese High Speed Railway using max-plus automata applied to a double layered network. In this paper, we attempt this by adding the initial delays to different stations to see how the delays will influence the punctuality of the whole system.

The objective of this paper is to demonstrate a method of optimizing subway networks using max-plus algebra to efficiently model such networks. The focus of our paper is subway networks of cities, as subway systems are physical networks and closed systems that are usually a reflection and representative of the cities themselves (Derrible [2012]).

Max-plus algebra is a mathematical system used to model discrete events with the operations being maximization and addition as opposed to conventional addition and multiplication. The use of max-plus algebra to optimize railway networks has been described in the early 2000s by Goverde [2007], and later by Heidergott et al. [2014],and is typically suited for schedules that are periodic and where events are interdependent. The system is modelled as a linear dynamic system and the computation is reduced to solving an eigenvalue problem. Goverde [2007] described the framework to perform a stability analysis on railway timetables to study its realisability and robustness and they performed a case study on the Dutch railway system to illustrate its potential. We adopt this framework to analyse the efficiency and robustness of subway systems of Amsterdam and Milan. Optimization methods for transport networks have been developed abundantly, and specific to transportation networks a technique that was popularised was that of capacity analysis. The goal of capacity analysis is to determine the maximum number of trains that would be able to operate on given infrastructure during a specific time interval, and this can be quantified (Abril et al. [2008]). The advantage of using max-plus algebra for this is that it can be extended to larger networks with ease and given the simplicity of the algorithm, it is computationally inexpensive.

Studies that compare different transport networks have also been performed (Derrible [2012], where they look at the topological properties of 33 different large subway networks across the world and compute certain characteristics such as betweenness centrality, connectivity and directness of the networks. They give an insight into topological properties and similarities and patterns that arose across different networks and provide a means to quantify and compare these topological properties. In the following analysis, we will compare the performance of Milan, Amsterdam, and random metro networks.

2 Methods

2.1 Metro Network Modelling

In this study, we use graph theory and a max-plus algebraic model to analyze the efficiency of metro networks.

For each metro network, we have the following criteria to design a timetable:

1. The trains should depart as frequently as possible, the total number of trains being fixed.
2. The departure frequency should be consistent across all tracks, creating a timetable with regular departure intervals.
3. Trains arriving at a station should wait for each other to allow passengers to transfer between lines if needed.
4. Trains should leave the station as soon as possible. In other words, the trains will depart as soon as all the trains from the other lines arrive at this station. At that time, the transfer between all lines will be finished. (For simplicity, we assume the time needed for the passengers to walk between platforms is 0.)

In our model, we abstract a metro network by setting the terminal stations and the branching stations (or transfer stations) as nodes and setting the subway rail in between to be the edges. Note that the stations that are not transfer

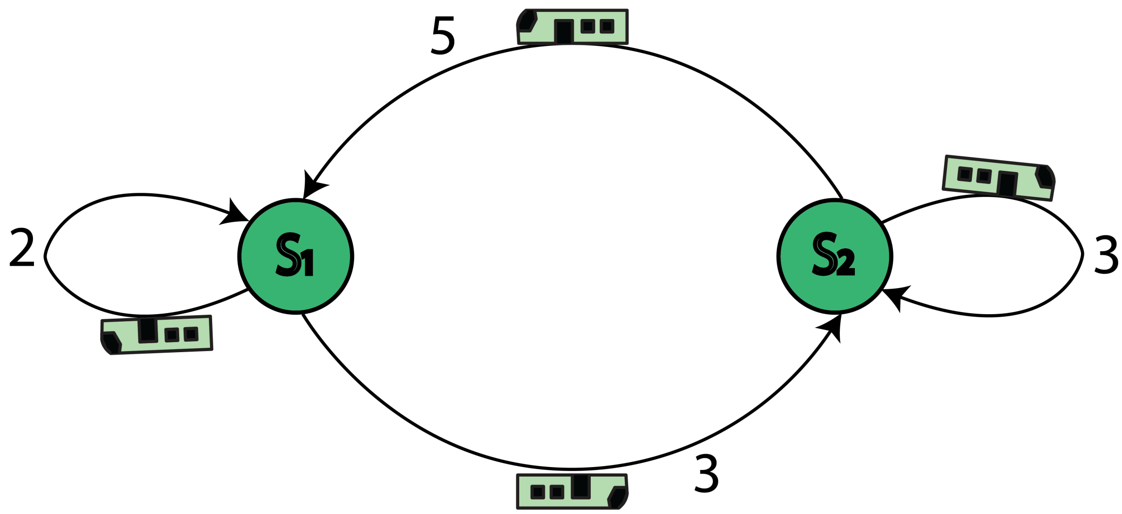

Figure 1: An illustrative railway network comprising two stations (S_1 and S_2) and 4 metro lines. One inner loop represents the travel between two stations. The self-loops represent travel through stations that can be neglected as they have no effect on the travel time.

stations (that is, no junction between different lines) will not be modelled since they do not influence the frequency of the train's departure.

2.1.1 Model Illustration & Max-Plus Algebra Model Construction

After abstracting a metro network, we will model the metro departure times at different stations using max-plus algebra. Let us start with a small example from Heidergott et al. [2014], which proves to be illustrative; the example network is shown in Figure 1, and it consists of two stations and a total of four trains, with one on each track. There are two outgoing tracks (hence two trains) from station S_1. Due to criteria 3 and 4, these two trains depart at the same time. That is, one train does not depart before the other train arrives, and once both of these two trains arrive at this station (to allow transfer), these two trains will depart simultaneously. Note that we assume that the transfer time for passengers to walk between platforms is 0. We denote the k^{th} common departure time at S_1 as $x_1(k)$. Similarly, $x_2(k)$ is the k^{th} common departure time of the two trains at S_2. Together, the departure times are written as the vector $\mathbf{x}(k) \in \mathbb{R}^2$ in this two-station setting. Thus, the initial departure times will be given by $\mathbf{x}(0)$, and the departure times of trains after that will be at $\mathbf{x}(1), \mathbf{x}(2), \ldots$. Because of the criteria given above, it follows that $x_1(k+1)$ satisfies the following two inequalities.

$$x_1(k+1) \geq x_1(k) + a_{11},$$
$$x_1(k+1) \geq x_2(k) + a_{12},$$

where a_{ij} is the travel time from station S_j to S_i. Applying this model to our setting, we have

$$x_1(k+1) \geq \max\left(x_1(k) + 2, x_2(k) + 5\right).$$

Due to criteria 1 and 4, this departure at S_1 has to occur as soon as both trains arrive at this station. Hence, the inequality will be an equality. In a similar manner, we obtain the equation for departure times at S_2. Thus, overall, we obtain

$$x_1(k+1) = \max\left(x_1(k) + 2, x_2(k) + 5\right),$$
$$x_2(k+1) = \max\left(x_1(k) + 3, x_2(k) + 3\right).$$

Then, generalizing our model to the setting with n stations, we have

$$x_i(k+1) = \max_{j=1,2,\ldots,n} \left(a_{ij} + x_j(k)\right), \quad i = 1, 2, \ldots, n. \tag{1}$$

Let $\oplus$ denote the maximum operator (over many terms) while $\otimes$ denotes the addition between terms. For $i = 1, \ldots, n$, we can rewrite Eq. (1) into

$$x_i(k+1) = \bigoplus_{j=1}^{n} \left(a_{ij} \otimes x_j(k)\right).$$

Another way to write this is

$$x_i(k+1) = (a_{i1} \otimes x_1(k)) \oplus (a_{i2} \otimes x_2(k)) \oplus \cdots \oplus (a_{in} \otimes x_n(k)))$$

where $\oplus$ denotes the maximum operator between two terms. (Note that $\oplus$ is smaller than $\bigoplus$ because the latter represents maximisation over more than two terms.)

Further, we may represent Eq. (1) in matrix-vector form as follows:

$$\mathbf{x}(k+1) = A \otimes \mathbf{x}(k) \tag{2}$$

where

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}. \tag{3}$$

Here, $\otimes$ for matrix calculation in max-plus algebra follows the traditional term-to-term corresponding rules of linear algebra. The only difference is that the two terms that are multiplied in linear algebra will be summed in this case. Moreover, the total sum in linear algebra is replaced by the maximisation $\bigoplus$ in this setting. More specifically, we express Eq. (2) as

$$\mathbf{x}(k+1) = \begin{bmatrix} \bigoplus_{j=1}^{n} a_{1j} + x_j(k) \\ \bigoplus_{j=1}^{n} a_{2j} + x_j(k) \\ \vdots \\ \bigoplus_{j=1}^{n} a_{nj} + x_j(k) \end{bmatrix}.$$

Based on Eq. (2), we have that $\mathbf{x}(1) = A \otimes \mathbf{x}(0)$. Moreover, since addition and multiplication in max-plus algebra have the associativity property, we can derive the following.

$$\begin{aligned} \mathbf{x}(2) &= A \otimes \mathbf{x}(1) \\ &= A \otimes (A \otimes \mathbf{x}(0)) \\ &= (A \otimes A) \otimes \mathbf{x}(0) \\ &= A^{\otimes 2} \otimes \mathbf{x}(0). \end{aligned}$$

We use $A^{\otimes 2}$ to denote $A \otimes A$, where the symbol $\otimes$ in the exponent indicates the matrix power in the max-plus algebra. To calculate a matrix power, first consider $A \otimes B$, where

$$B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{bmatrix}.$$

We get

$$A \otimes B = \begin{bmatrix} \bigoplus_{j=1}^{n} a_{1j} + b_{j1} & \bigoplus_{j=1}^{n} a_{1j} + b_{j2} & \cdots & \bigoplus_{j=1}^{n} a_{1j} + b_{jn} \\ \bigoplus_{j=1}^{n} a_{2j} + b_{j1} & \bigoplus_{j=1}^{n} a_{2j} + b_{j2} & \cdots & \bigoplus_{j=1}^{n} a_{2j} + b_{jn} \\ \vdots & \vdots & \vdots & \vdots \\ \bigoplus_{j=1}^{n} a_{nj} + b_{j1} & \bigoplus_{j=1}^{n} a_{nj} + b_{j2} & \cdots & \bigoplus_{j=1}^{n} a_{nj} + b_{jn} \end{bmatrix}. \tag{4}$$

Then, we can calculate matrix powers $A^{\otimes 2}, A^{\otimes 3}, \ldots$, and, subsequently, we get $\mathbf{x}(k) = A^{\otimes k} \otimes \mathbf{x}(0)$, $(k = 0, 1, \ldots)$.

2.1.2 Eigenvalue and Eigenvector in Max-Plus Algebra

The matrix in max-plus algebra also has an eigenvalue and eigenvectors. Given a square matrix A of size n, if there exists a constant λ and n-dimensional vector $\mathbf{v}$ with not all of its components $-\infty$ such that

$$A \otimes \mathbf{v} = \lambda \otimes \mathbf{v},$$

then λ and $\mathbf{v}$ will be the eigenvalue and eigenvector of the matrix A. Note that $\lambda \otimes \mathbf{v}$ is an n-dimensional vector whose i^{th} element equals $\lambda \otimes v_i = \lambda + v_i$.

Now consider Eq. (2). If we let $\mathbf{x}(0)$ be an eigenvector of A corresponding to the eigenvalue λ, then the solution of (2) will be

$$\mathbf{x}(1) = A \otimes \mathbf{x}(0) = \lambda \otimes \mathbf{x}(0),$$
$$\mathbf{x}(2) = A \otimes \mathbf{x}(1) = A \otimes (\lambda \otimes \mathbf{x}(0)) = \lambda^{\otimes 2} \otimes \mathbf{x}(0).$$

Hence, in general,

$$\mathbf{x}(k) = \lambda^{\otimes k} \otimes \mathbf{x}(0), \quad k = 0, 1, 2, \ldots$$

Note that the numerical evaluation of $\lambda^{\otimes k}$ in max-plus algebra is equal to $k \times \lambda$.

Like in conventional linear algebra, the eigenvectors in max-plus algebra are determined up to a multiplicative factor; the eigenvectors are also not unique. Thus, it can be shown that, if the same constant number is added to all elements of $\mathbf{x}(0)$, then the resulting vector is still an eigenvector.

The significance of an eigenvector is seen when we set the initial condition, that is, the initial departure times $\mathbf{x}(0)$ of trains at different stations in the system described in Eq. (2) or Figure 1 to be an eigenvector. It can easily be shown that the subsequent departure times $\mathbf{x}(k)$ ($k > 0$) for all the stations in that system will be equally spaced by the eigenvalue λ of A, leading to a 'regular' timetable. On the other hand, suppose the initial departure time of the stations in this system is not an eigenvector of A. Then, after some transient time, even though the average difference between consecutive departure times at all stations will be λ, the difference between consecutive departure times in at least one station will not be homogeneous. This is a consequence of the system entering a 'periodic regime' of period $\rho > 1$ after the transient time, where we define a periodic regime as the ordered set $\mathbf{x}(k), \mathbf{x}(k+1), \mathbf{x}(k+2), \ldots$ for some $k \geq 0$ such that

$$\mathbf{x}(k + \rho) = \mu \otimes \mathbf{x}(k),$$

where $\mu/\rho = \lambda$. It is well-known that all max-plus systems of the type (2) enter a periodic regime after some transient time (Heidergott et al. [2014]).

2.1.3 Adding Trains to a Metro Line

Ideally, we want to lower the average time between departures (corresponding to increasing the frequency of trains). An obvious step seems to be to add an extra train. However, in this case, we need to determine which track this added train should run.

Consider the setting in Figure 2, and suppose the extra train is placed on the track from S_1 to S_2, just outside station S_1. Suppose also that a train has already left in the direction of S_2 and is in front of the newly added train. If the newly added train is the k^{th} train in this direction, the train in front is the $(k-1)^{th}$ train, and this model describing the earliest departure times will be given by

$$x_1(k+1) = \max\left(x_1(k) + 2, x_2(k) + 5\right),$$
$$x_2(k+1) = \max\left(x_1(k-1) + 3, x_2(k) + 3\right).$$

We want to change the above equations to an equation with a first-order recurrence relation. Thus, we introduce an auxiliary variable x_3 such that $x_3(k+1) = x_1(k)$ in the following way:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} = \begin{bmatrix} 2 & 5 & -\infty \\ -\infty & 3 & 3 \\ 0 & -\infty & -\infty \end{bmatrix} \otimes \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix}.$$

In order to understand the auxiliary variable x_3 better, we may assume that there is a virtual station S_3 situated on the track from S_1 to S_2, just outside S_1, with the travel time between S_1 and S_3 being 0, and the travel time from S_3 to S_2 being the original travel time from S_1 to S_2 (which is 3 is the example in Figure 1). Note that in this setting, our assumption of there being one train on each track still holds.

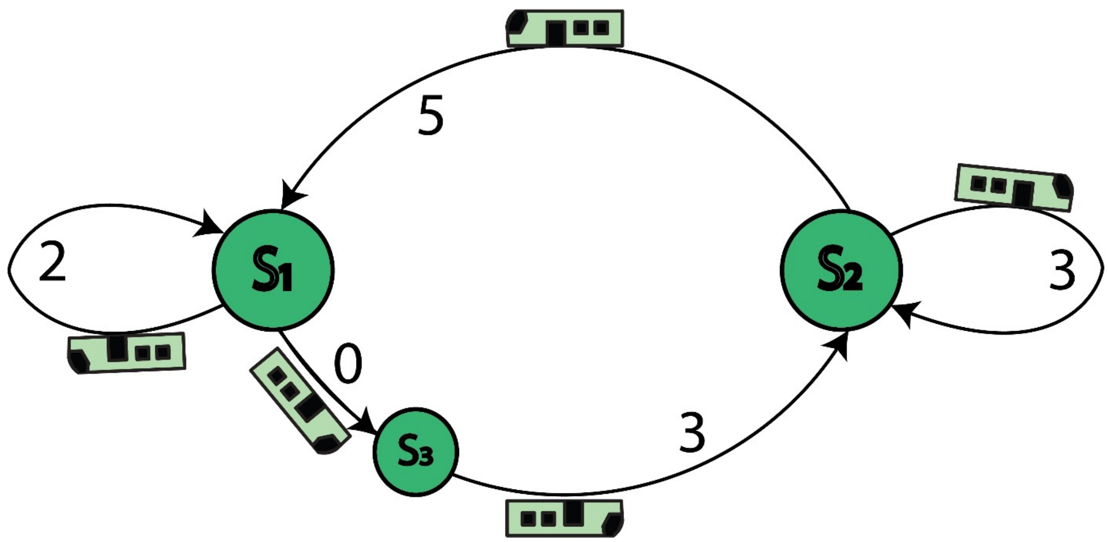

Figure 2: An illustrative metro network with two subway stations (denoted by S_1 and S_2) with an additional train.

We could place the auxiliary station in other places. For example, if S_3 was situated just before S_2, still on the track from S_1 to S_2, then the equations become

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} = \begin{bmatrix} 2 & 5 & -\infty \\ -\infty & 3 & 0 \\ 3 & -\infty & -\infty \end{bmatrix} \otimes \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix}.$$

Alternatively, S_3 can also be placed just after S_2 on the track towards S_1. Each of these three settings above indeed speeds up the network's efficiency to the same extent. Thus, the eigenvalues of the matrices corresponding to these three settings are identical (and equal to 3). Note, however, that the eigenvectors for these three settings are different.

In fact, the eigenvalue $\lambda = 3$ is also the travel time on the outer circuit (self-loop) at S_2, meaning this circuit becomes the main restriction on the efficiency of the whole system. This is a consequence of the following theorem, for which we define the average weight of a circuit to be the total travel time on this circuit divided by the number of tracks in it (including the virtual track with distance 0). Note that the number of tracks is precisely the number of trains on this circuit. Theorem 2.1 implies that the eigenvalue of the matrix A describing this metro system is the maximal average weight among all circuits in this metro system. That is,

Theorem 2.1 *Let $A \in \mathbb{R}^{n\times n}_{\max}$ be irreducible. Then A has a unique finite eigenvalue, denoted $\lambda = \lambda(A)$. Moreover, this eigenvalue is equal to the maximal average weight of all circuits in $\mathcal{G}(A)$. Let c denote a circuit of $\mathcal{G}(A)$. Denote the set of all circuits of $\mathcal{G}(A)$ by $\mathcal{C}(A)$. Then*

$$\lambda(A) = \max_{c\in\mathcal{C}(A)} \frac{|c|_w}{|c|_l},$$

where $|c|_w$ is the total weight of circuit c and $|c|_l$ is the total length of the circuit c. (Heidergott et al. [2014])

A circuit c is called critical if its average weight is maximal, i.e., $\lambda = |c|_w/|c|_l$. Denote by $\mathcal{G}(A)$ the graph whose adjacency matrix is A; we call $\mathcal{G}(A)$ the communication graph of A. The critical graph of A, denoted by $\mathcal{G}^{cr}(A)$, is the graph consisting of those nodes and edges that belong to critical circuits in $\mathcal{G}(A)$.

Viewed as a graph, a metro network of the type in Fig. 4 has adjacency matrix A, where entry a_{ij} denotes the travel time from node j to node i; if there is no track between the two nodes, the travel time is ∞. Thus, an adjacency matrix is irreducible if there is a path - a sequence of directed edges - between any pair of nodes in its communication graph.

By construction, the matrix A corresponding to most practical metro systems is irreducible since it should be possible to travel from any station to any other station in the network.

THEOREM 2.1 IS OUR MAIN BASIS FOR AN ALGORITHM TO IMPROVE THE EFFICIENCY OF THE METRO NETWORK.

Consider the matrix A_λ with all its elements to be

$$[A_\lambda]_{ij} = a_{ij} - \lambda,$$

i.e., A_λ is obtained by subtracting the eigenvalue from each element of A. Now define

$$\begin{aligned} A_\lambda^\star &= \bigoplus_{k\geq 0} A_\lambda^{\otimes k} && (5) \\ &= E \oplus A_\lambda \oplus A_\lambda^{\otimes 2} \oplus A_\lambda^{\otimes 3} \oplus \cdots && (6) \end{aligned}$$

where E is the identity matrix (having 0 on the diagonal and $-\infty$ elsewhere). Then, the eigenvectors of A may be obtained with the following lemma:

Lemma 2.2 *Let the communication graph $\mathcal{G}(A)$ of matrix A have finite maximal average circuit weight λ. Then λ is an eigenvalue of A, and the column $[A_\lambda^\star]_{\cdot\eta}$ is an eigenvector of A associated with λ, for any node η in $\mathcal{G}^{cr}(A)$. (Heidergott et al. [2014])*

Note that, although Eq. (6) is a max-plus sum to infinity, it does have a limit. It is often possible to compute $A_\lambda^\star$ by taking the sum of a finite number of powers of A_λ (Heidergott et al. [2014]).

2.2 Splitting a Station

Another way to optimize the critical circuit is to split a station. We first identify the node in the critical circuit with the maximum in-degree. We then replace this node with two nodes. We then iterate through each of the original node's incoming and outgoing connections and randomly assign them to one of the two new nodes, thereby splitting the connections of the original node to the new nodes evenly on average (as in Figure 3).

We then update the adjacency matrix corresponding to the split network and perform the interdeparture time calculation and optimization by iteratively splitting stations.

2.2.1 Delay Analysis

In reality, the departure times of trains should follow some scheduled timetable, which describes the planned departure times of the trains at different stations. Now assume the $(k+1)^{th}$ planned departure times for all stations are given by the vector $\mathbf{d}(k+1) = [d_1(k+1), \ldots, d_n(k+1)]^\top$. We require $\mathbf{d}(k+1) \geq \mathbf{x}(k+1)$, which is equivalent to

$$\mathbf{d}(k+1) \geq A \otimes \mathbf{x}(k). \qquad (7)$$

It is always possible to find the largest solution to such an inequality Heidergott et al. [2014], where the largest solution is named by the principal solution and is denoted as $\mathbf{x}^*(A, \mathbf{b})$.

Theorem 2.3 *For $A \in \mathbb{R}_{\max}^{m\times n}$ and $\mathbf{b} \in \mathbb{R}^m$ it holds that*

$$[\mathbf{x}^*(A,\mathbf{b})]_j = \min\{b_i - a_{ij} : i = 1, 2, \ldots, m\}$$

for $j = 1, 2, \ldots, n$.

We can show that $\mathbf{x}(k)^*(A, \mathbf{d}(k+1))$ is the principal solution to (7) and gives the latest departure times of trains from previous stations such that the train can still meet the next scheduled time $\mathbf{d}(k+1)$.

If a timetable requires that a train is not allowed to depart before its scheduled departure time, that is,

$$\mathbf{x}(k+1) = A \otimes \mathbf{x}(k) \oplus \mathbf{d}(k+1).$$

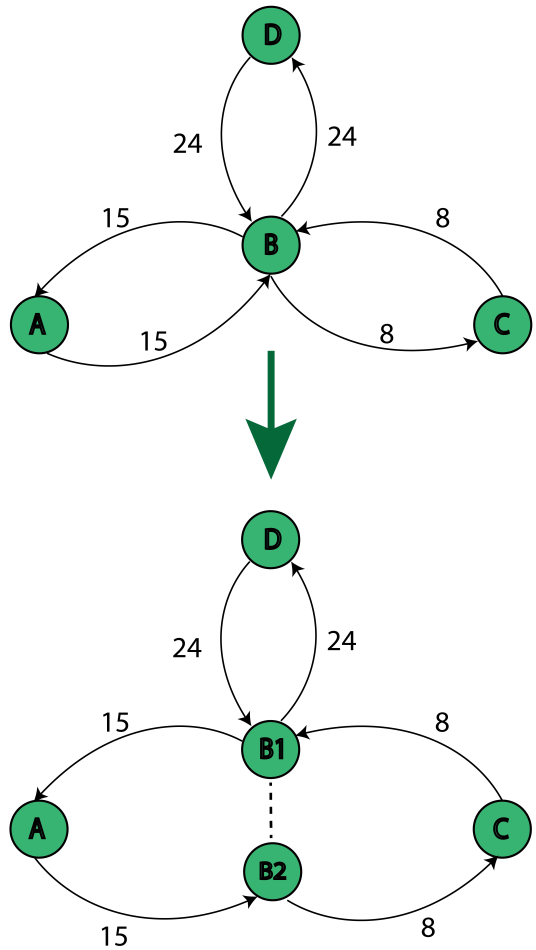

Figure 3: An illustration of splitting a station. The critical circuit in the figure is between B and A, among which B has the maximum indegree. We therefore split B into B1 and B2.

This equation can be further written as

$$\begin{aligned}\mathbf{x}(k+1) &= A \otimes (A \otimes \mathbf{x}(k-1) \oplus \mathbf{d}(k)) \oplus \mathbf{d}(k+1)\\ &= A^{\otimes 2} \otimes \mathbf{x}(k-1) \oplus A \otimes \mathbf{d}(k) \oplus \mathbf{d}(k+1)\\ &= A^{\otimes 2} \otimes \mathbf{x}(k-1) \oplus \mathbf{d}(k+1)\end{aligned}$$

since $\mathbf{d}(k+1) \geq A \otimes \mathbf{d}(k)$ for a feasible timetable. Following the same way, we conclude that

$$\mathbf{x}(k+1) = A^{\otimes(k+1)} \otimes \mathbf{x}(0) \oplus \mathbf{d}(k+1). \tag{8}$$

In the case without no initial delays, $\mathbf{x}(0) = \mathbf{d}(0)$ and hence Eq. (8) can be written as

$$\mathbf{x}(k+1) = A^{\otimes(k+1)} \otimes \mathbf{d}(0) \oplus \mathbf{d}(k+1) = \mathbf{d}(k+1).$$

Let a first train be delayed so that for a unique and certain j, $x_j(0) > d_j(0)$, and $x_i(0) = d_i(0)$ for $i \neq j$. If this initial delay causes a delay for the $(k+1)^{\text{th}}$ train departing from node i, then

$$\bigoplus_{l=1}^{n} \left[A^{\otimes(k+1)}\right]_{il} \otimes x_l(0) > d_i(k+1).$$

Since $x_i(0) = d_i(0)$ for $i \neq j$, the delay of $(k+1)^{\text{th}}$ departure at node i is caused by the initial delay of node j. Hence,

$$\left[A^{\otimes(k+1)}\right]_{ij} \otimes x_j(0) > d_i(k+1). \tag{9}$$

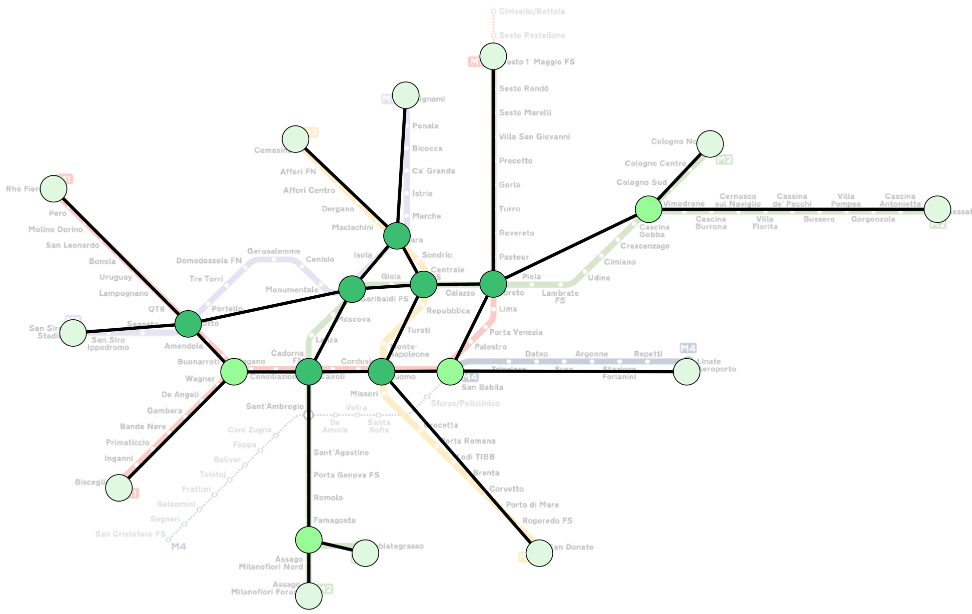

Figure 4: Abstracted version of the metro network of Milan.

Given the initial delay at node j, if the initial delay is out of the system after k_j^* steps (that is, all the trains at all stations can departure on time after k_j^* steps), then

$$k_j^* = \min\left\{k \in \mathbb{N} : \left[A^{\otimes(k+1)}\right]_{ij} \otimes x_j(0) \leq d_i(k+1) \text{ for } i = 1, \ldots, n\right\}.$$

In the case of a tight timetable, $k_j^* = +\infty$, i.e. the delay never goes out of the system (there will always be some trains delayed in this system).

2.2.2 Stability Analysis in the Delayed Network

The critical circuit is the slowest in the network. The mean cycle time of this circuit determines the minimum possible cycle time for the entire system.

In practice, we do not set the planned inter-departure time as the mean cycle time of the critical circuit since this timetable cannot handle train delays properly. A delay on a critical circuit will never die out. Hence, buffer times should be incorporated into the timetable to increase the stability and robustness of the system.

Suppose we desire a periodic timetable with period $T \in \mathbb{R}$. Hence, the planned inter-departure times are

$$\mathbf{d}(k) = T^{\otimes k} \otimes \mathbf{d}(0) \tag{10}$$

where $\mathbf{d}(0)$ is the departure schedule for the first trains.

A scheduled train service system is stable if any initial delay settles in a finite time. Consider system in Eq. (8) and (10). This system is stable if and only if $\lambda < T$, where λ is the eigenvalue defined of A Heidergott et al. [2014].

The traffic rate $\rho = \lambda/T$. It is an indicator denoting the trade-off between robustness and the maximum performance $(T = \lambda)$ under the ideal case. A stable system requires $0 \leq \rho < 1$, whereas $\rho = 1$ corresponds to the saturated case, where any delay cannot go out of the system Heidergott et al. [2014].

Theorem 2.4 *The stability margin Δ with respect to a set of parameters is the maximum amount of time that can be added to all these parameters (travel times, waiting times, ...) simultaneously such that the corresponding system is still operable with the given period T; that is, the eigenvalue of the new system matrix with the maximum Δ added equals T. The stability margin Δ for the first-order system described in Eq. (8) is equal to $\Delta = T - \lambda$, where λ is the eigenvalue of A Heidergott et al. [2014].*

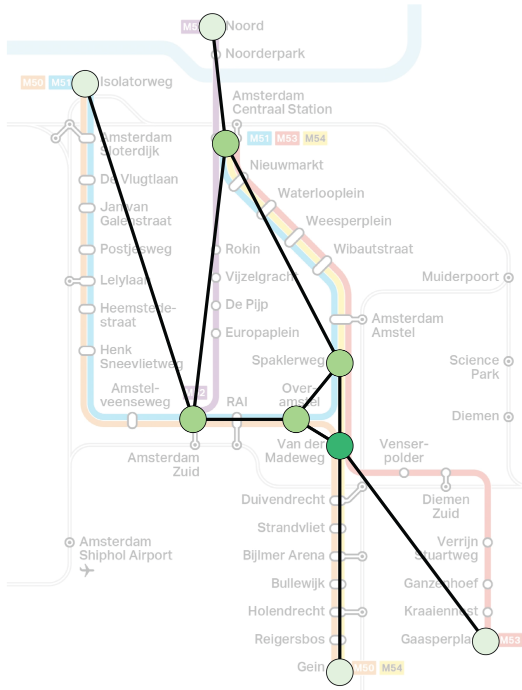

Figure 5: Abstracted version of the metro network of Amsterdam

Theorem 2.5 *Consider the max-plus linear system Eq. (8). The entry r_{ji} of recovery matrix R is the maximum delay of $x_i(0)$ such that $x_j(k)$ is not delayed for any $k \geq 1$. The elements of the recovery matrix R are given by*

$$r_{ji} = d_j - d_i - \left[A \otimes T^{\otimes -1}\right]^+_{ji}$$

where $T \in \mathbb{R}$ and the vector $\mathbf{d} = (d_1, d_2, \ldots, d_n)^\top$ equals the vector $\mathbf{d}(0)$ in Eq. (10). $\left[A \otimes T^{\otimes -1}\right]^+_{ji}$ refers to the $(j,i)^{th}$ element of the matrix

$$\left[A \otimes T^{\otimes -1}\right]^+ = \bigoplus_{k=1}^{\infty} \left[A \otimes T^{\otimes -1}\right]^{\otimes k}.$$

If in the graph of A, no path exists from node i to j, then $r_{ji} = +\infty$.

Note that the analysis of this recovery matrix assumes that only one train is initially delayed. A more general setup is the following. The vector of the latest departure times such that the trains can meet the timetable at their subsequent departures is given by the principal solution $\mathbf{x}^*(A, \mathbf{d})$.

2.3 Metro Networks Data Processing

To analyze the public transportation networks, we needed to identify the critical stations in each network such as terminal stations, intersections, and branching points, and the corresponding average travel times between the critical stations. Initially, we attempted to obtain the networks holistically from open data initiatives of the corresponding cities. While the volume of data in the standard GTFS format for public transportation networks is impressive, there exist multiple inconsistencies within and across datasets corresponding to each city which make it frustratingly impossible to programmatically acquire the networks and timetables in a short time. We hence chose to analyze the metro networks

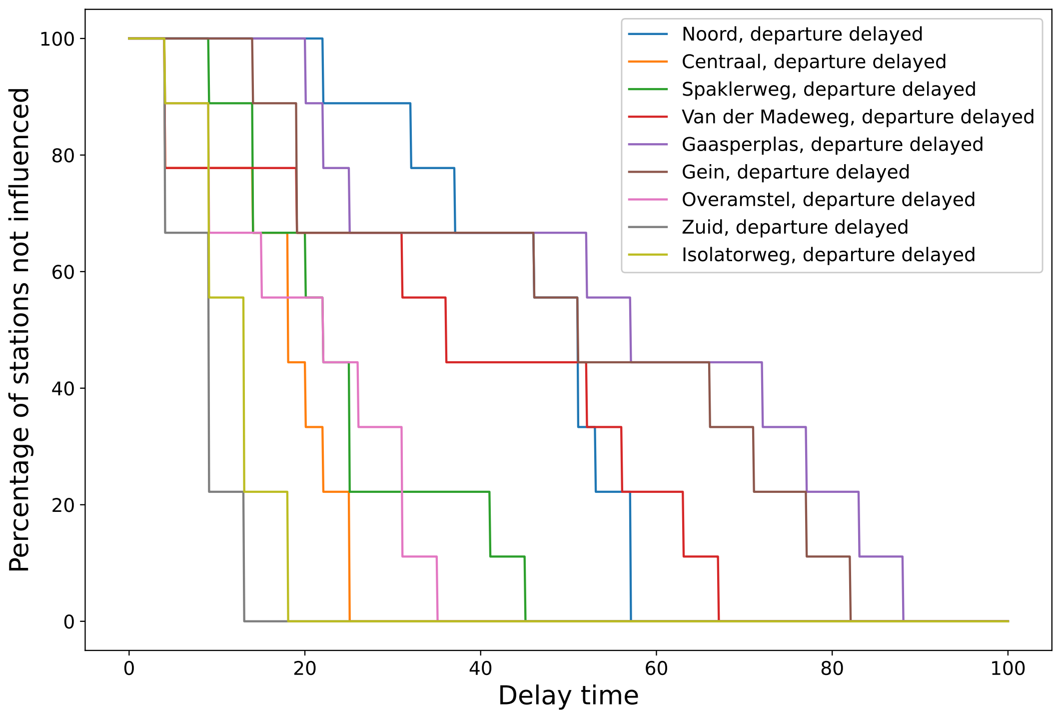

Figure 6: Delay analysis for Amsterdam when $\rho = 0.8$ before the train number is optimized.

of Amsterdam (5 lines and 39 stations) and Milan (5 lines and 113 stations) for the manageable sizes of their reduced networks. Each of these metro networks contains multiple lines with multiple stations in each line. We reduced the metro networks keeping only the stations that are either terminal stations of a line, intersection points between multiple lines, or branching points within the same line (Figure 4, 5). We then obtained the inter-station transit times from the websites of the corresponding metro networks and Google Maps. For ease of analysis, we initially assume that the trains corresponding to different lines from the intersection stations depart at the same time, relaxing the assumption later. The reduced network data has been provided in the GitHub repository.

2.4 Critical Circuit Identification for Adding trains and Splitting Stations

For optimizing lambda, we identify the critical circuits in each network and add a train or split a station as described below. For a given metro network, we use the *simple_cycles* function from the *networkx* package to identify all circuits. We then calculate the weight of each circuit by adding the transition times corresponding to each path in the circuit and dividing the sum by the total number of nodes in the circuit. We then identify the circuit with the highest weight as the critical circuit and modify it to reduce the inter-departure time.

sectionResults

In this section, we apply our model from the previous section to the real metro systems in Milan and Amsterdam to analyze their theoretical optimal efficiency given the total number of trains and design an optimal timetable for them. Note that the matrices corresponding to these metro systems are fundamental. Amsterdam Metro has 5 lines and 39 stations. Milan metro has 5 lines and 113 stations.

We first demonstrate our subway line graph with the example of Milan Metro Line 1 (MML1). Sesto S. Giovanni, Rho Fieramilano, and Bisceglie stations are terminals of MML1, and hence, they are denoted by nodes in Figure 4. Note that MML1 has two branches on the west of Pagano station; therefore, we also treat Pagano as a transfer station in our model (although we cannot transfer to other lines at Pagano). Moreover, Lotto, Cadorna, Duomo, San Babila, and Loreto are transfer stations of MML1 (with other lines in Milan). Hence, they are all denoted by nodes in Figure 4.

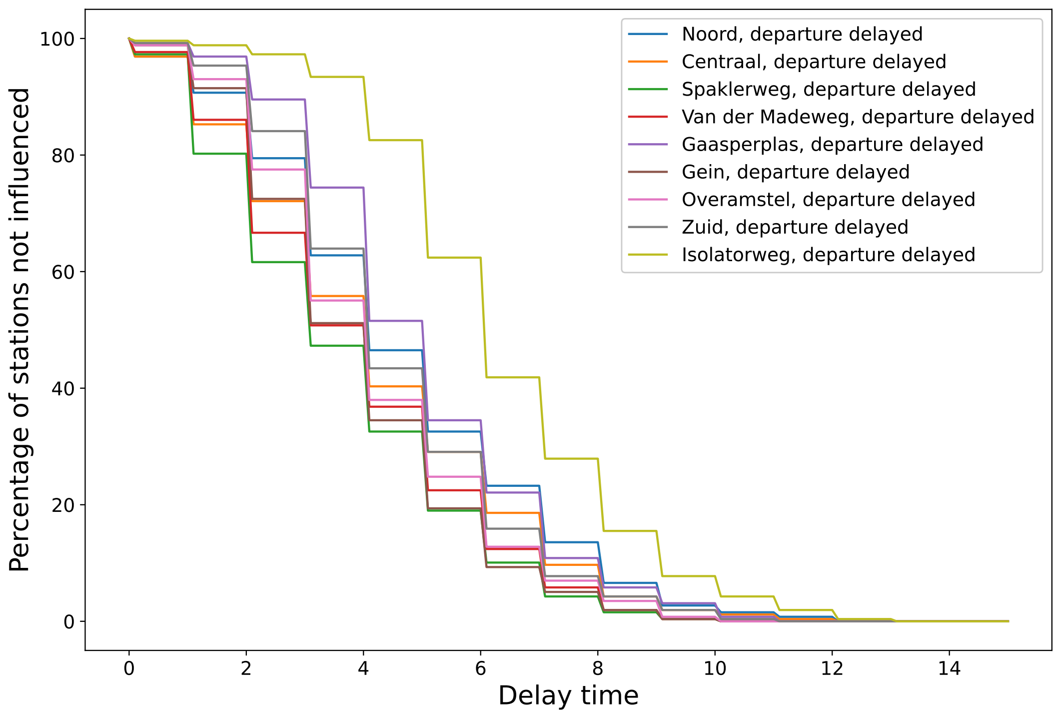

Figure 7: Delay analysis for Amsterdam when $\rho = 0.8$ after the train number is optimized.

Following the same convention, we draw the subway system networks of Milan and Amsterdam in Figure 4 and 5, respectively. The reduced networks thus have 9 nodes and 10 edges for Amsterdam and 23 nodes and 27 edges for Milan. We then calculated the eigenvalues of the corresponding max-plus algebraic transition matrix, representing the average inter-departure time. If only one train is operating in between two nodes in each subway line (before adding trains), the λ for Milan and Amsterdam are 23 and 18, respectively, showing that the travel time between the transfer stations in Amsterdam is faster, and exhibits higher efficiency in Amsterdam's subway system.

3 Results

In this section, we apply our model from the previous section to the real metro systems in Milan and Amsterdam to analyze their theoretical optimal efficiency given the total number of trains and design an optimal timetable for them. Note that the matrices corresponding to these metro systems are fundamental. Amsterdam Metro has 5 lines and 39 stations. Milan metro has 5 lines and 113 stations.

We will first demonstrate our subway line graph with the example of Milan Metro Line 1 (MML1). Sesto S. Giovanni, Rho Fieramilano, and Bisceglie stations are terminals of MML1, and hence, they are denoted by nodes in Figure 4. Note that MML1 has two branches on the west of Pagano station; therefore, we also treat Pagano as a transfer station in our model (although we cannot transfer to other lines at Pagano). Moreover, Lotto, Cadorna, Duomo, San Babila, and Loreto are transfer stations of MML1 (with other lines in Milan). Hence, they are all denoted by nodes in Figure 4. Following the same convention, we draw the subway system networks of Milan and Amsterdam in Figure 4 and 5, respectively. The reduced networks thus have 9 nodes and 10 edges for Amsterdam and 23 nodes and 27 edges for Milan. We then calculated the eigenvalues of the corresponding max-plus algebraic transition matrix, representing the average inter-departure time. If only one train is operating in between two nodes in each subway line (before adding trains), the λ for Milan and Amsterdam are 23 and 18, respectively, showing that the travel time between the transfer stations in Amsterdam is faster, and exhibits higher efficiency in Amsterdam's subway system.

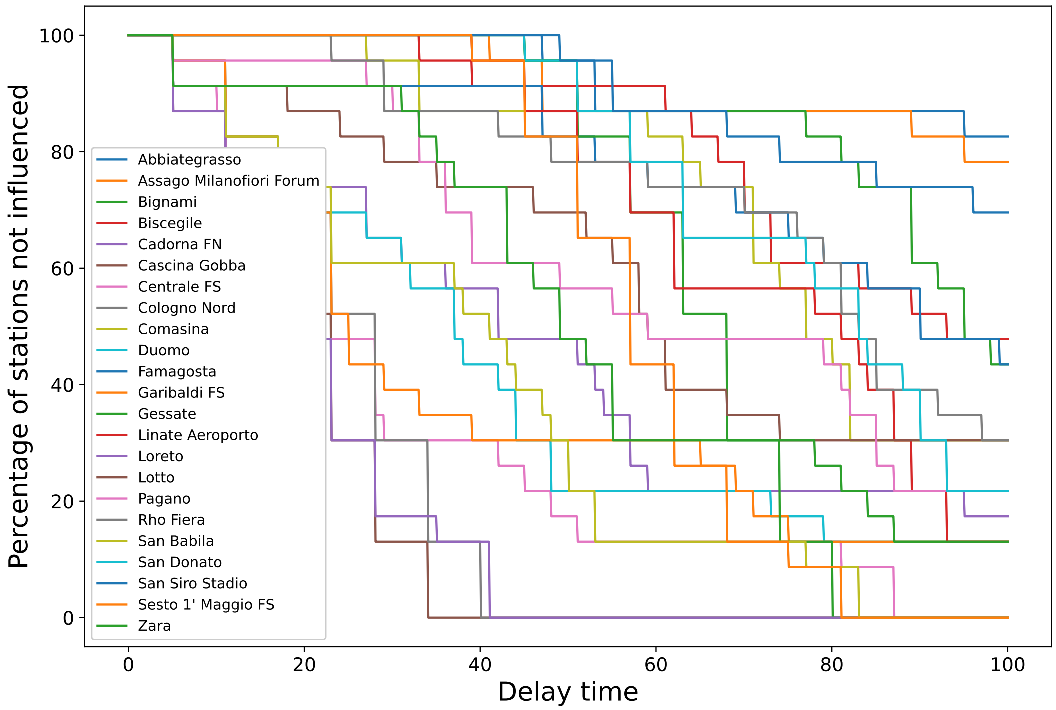

Figure 8: Delay analysis for Milan when $\rho = 0.8$ before the train number is optimized.

4 Delay and Stability Analysis

In all the following stability analyses, we keep $\rho = 0.8$ always fixed. Moreover, we will only delay the departure time of the first train at one station (node) each time. The delay time is 0-100 minutes before adding trains and 0-20 minutes after adding trains. Then, we will see that given the initial delay at a certain station, the departure time of how many percent of stations will never be influenced by that delay.

Firstly, we analyze the robustness of the Amsterdam metro when we set $\rho = \lambda/T = 0.8$ in Figure 6. When there are no additional trains added on each rail, which means on the rail between each pair of connected nodes, there is only one train operating in that direction. In this case, the inter-departure time $\lambda = 18$. For the given ρ, $T = 22.5$. From Figure 6, we can see that the delay of the stations located in the city centre and having more lines interchanging will have a larger influence on the stability of the whole metro system. For example, there are three major subway lines interchanging at Zuid, and a delay of 15 minutes in Zuid will induce a delay in all the stations in the Amsterdam metro system. However, the delay of Noord, which is the terminal of M52 without interchanging with other stations, for 15 minutes will have absolutely no influence on the departure times of all the other stations. The same result applies to Gaasperpias station.

Then, we will analyze the robustness of Amsterdam metro after adding trains to its subway lines when $\rho = \lambda/T = 0.8$. In this case, $\lambda = 2$. For the given ρ, $T = 2.5$. These results are shown in Figure 7. Obviously, we may find that when the number of trains is increased, the maximum allowed delay time (for all stations) that will not influence other stations is significantly decreased. The reason behind this phenomenon is that when the number of trains is increased, following Theorem 2.1, the $|c|_l$ is increased. Hence, when ρ is fixed, both λ and T will decrease. Therefore, the maximum allowed delay time will be decreased for all stations. Moreover, our previous finding still holds, that is, the delay of stations located in the city centre will have a larger influence on the whole system (see Spaklerweg station), and the delay from suburb stations will have a trivial influence on the punctuality of the system (see Isolatorweg station).

We also applied the same analysis to the Milan Metro system. Milan's metro system is more complicated than Amsterdam's. We can see that for the larger metro system, the delay propagation is slower than in Amsterdam. Without adding trains (one train on each line), $\lambda = 23$ and $T = 28.75$ in this setting. Comparing Figure 6 and 8, we can see

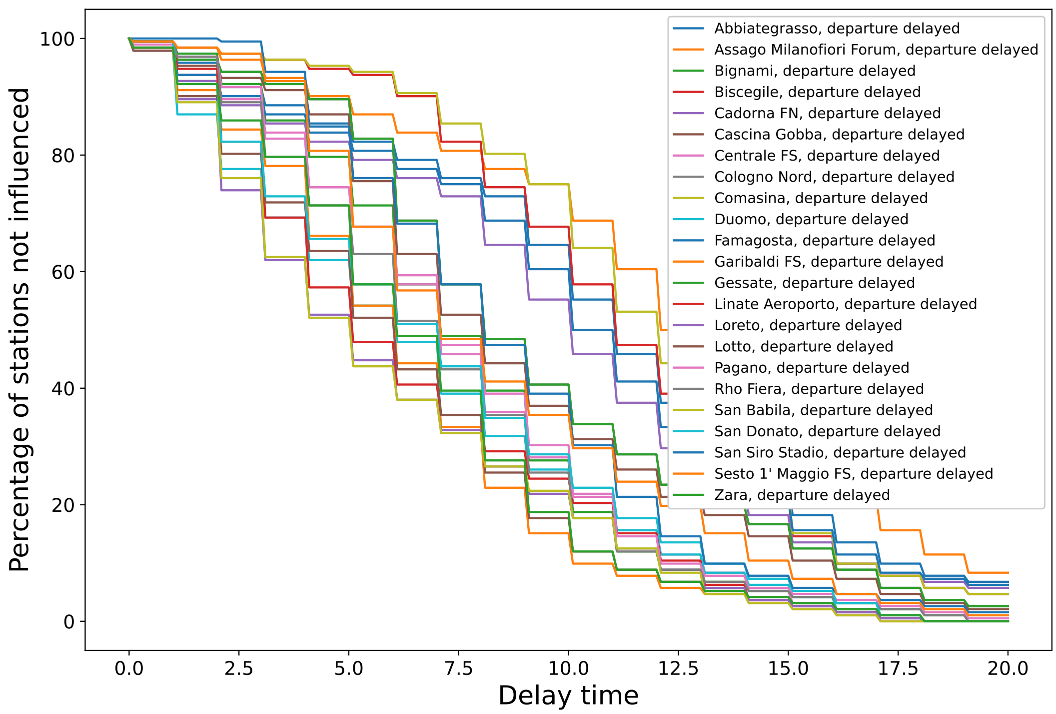

Figure 9: Delay analysis for Milan when $\rho = 0.8$ after the train number is optimized.

that given the delay of 100 minutes, no matter whether this station is in the city centre or suburb, the punctuality of all stations in Amsterdam will be influenced. However, if this happens in the suburban station in Milan (such as Assago Milanofiori Forum station), 80% of other stations in Milan will not be influenced. The same result is also applied to the optimized metro system. After adding trains, $\lambda = 2$ and $T = 2.5$ in Milan. Having added trains, for Amsterdam, a 12-minute delay at any station will influence the punctuality of the whole system (see Figure 7). However, a 20-minute delay at Assago Milanofiori Forum station in Milan will still not influence the punctuality of 10% of stations in that system (see Figure 9). Moreover, there is a gap seen in Figure 9, which is a gap separating the suburb stations and city centre stations, showing the greater influence of the delay on city centre stations than on suburb stations.

5 Optimizing the inter-departure times

To optimize the inter-departure times, we added more trains to the network and asked how many trains must be added to obtain lower average inter-departure times (λ). For each network, we determined the "critical circuit" that represents the most congested part of the network and added a train to that circuit. We iterate this process of calculating the critical circuit and adding the network until the required desired λ is achieved. To achieve $\lambda = 2$, 170 trains had to be added to the Milan Metro system, forming a system with 192 trains (Figure 10 A). Moreover, 62 trains are added to the Amsterdam metro system, forming a system with 101 trains (Figure 10B). These numbers translate to 1.7 trains per station on average for Milan, and 2.6 trains per station for Amsterdam, suggesting that Milan network has a higher potential for optimization than Amsterdam.

We further compare these results with a randomly generated transition network with 5 nodes and uniformly sampled travel times between the range of 5 to 15 minutes inspired by Milan and Amsterdam's transit times. We find that the random network requires the addition of 130 trains (Figure 10C).

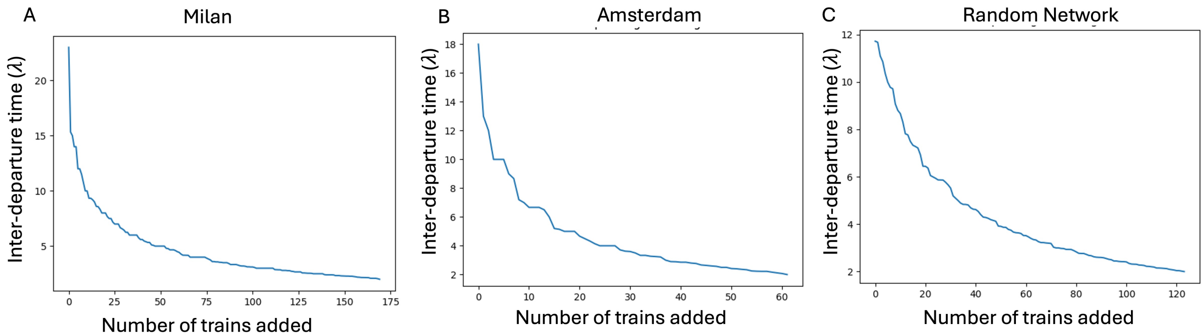

Figure 10: Optimizing inter-departure times for **A** Milan, **B** Amsterdam and **C** Random transit networks. The figures depict the inter-departure time (y axis) against the number of trains added (x axis) at each iteration.

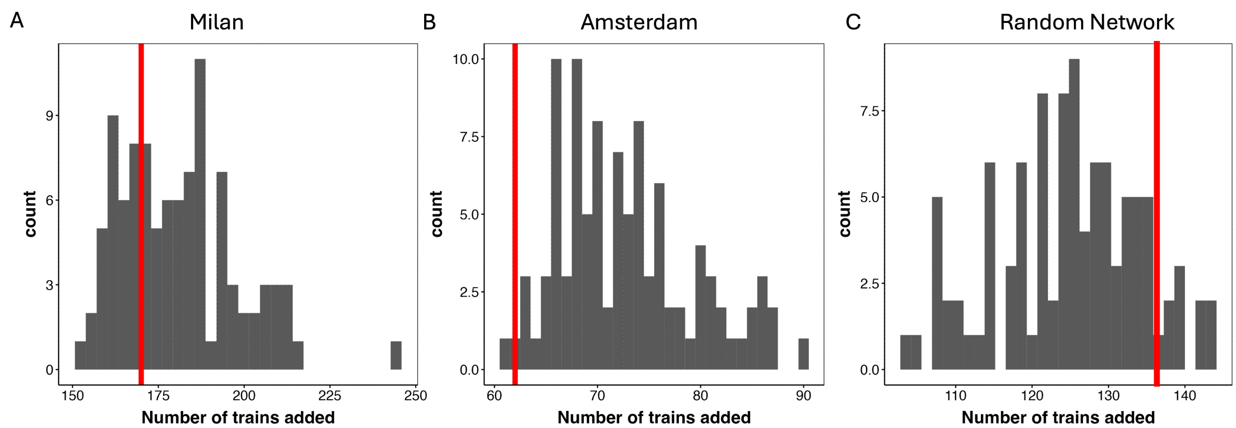

Figure 11: Estimating the optimization potential of **A** Milan, **B** Amsterdam and **C** Random transit networks against their weight-randomized counterparts. The figures depict the histogram of the maximum number of trains to be added for different weight-randomized networks. The red lines represent the number of trains for the corresponding original network.

5.1 Weight randomization reveals the optimization potential of transport networks

We further randomized the networks of Milan and Amsterdam by shuffling the weights among themselves, to ask if theoretically there exists a better network configuration that has better performance as compared to the real network.

For each such "weight-randomized" network, we calculated the inter-departure times and optimized them by adding more trains. The initial inter-departure times do not vary upon randomization. However, we find that the number of trains to be added to achieve $\lambda = 2$ varies widely for the weight-randomized networks. Interestingly, most of the weight-random networks required a larger number of trains as compared to the original for Amsterdam (percentile 5%), while for the Milan network, we found more weight-randomized networks with lesser train addition requirement (percentile 20%) for achieving the same optimal average inter-departure time. This suggests that the Milan network may have a higher capacity for optimization than the Amsterdam network. The corresponding percentile for the 5-node random network was much higher (>80%) suggesting that the number of nodes in the network may not be a significant factor in determining the optimization potential of a transport network.

We further analyzed the effect of the distribution used to determine the weights of the random network. We find, on average, that using a gamma distribution allows for better optimization potential, that is the mean number of trains to be added is much lower as compared to other distributions, as showed in Figure 12.

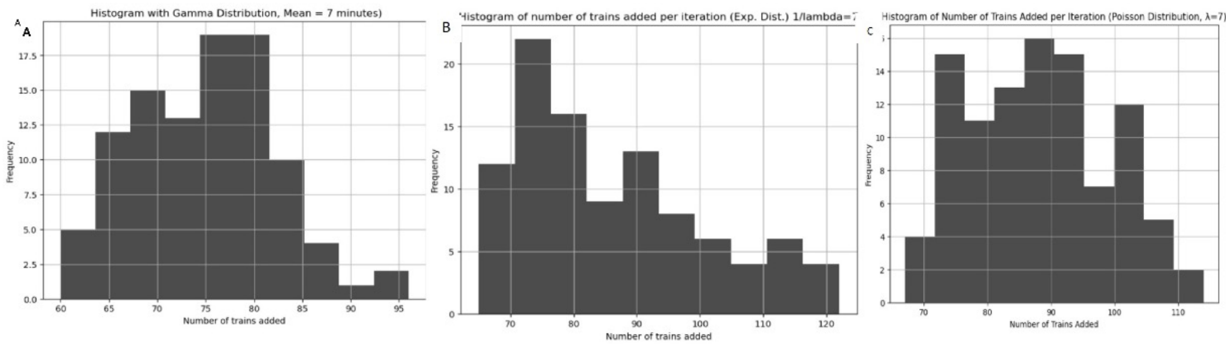

Figure 12: Weight randomization analysis performed on different random networks with the original weights sampled from **A** Gamma distribution, **B** Exponential distribution and **C** Poisson distribution

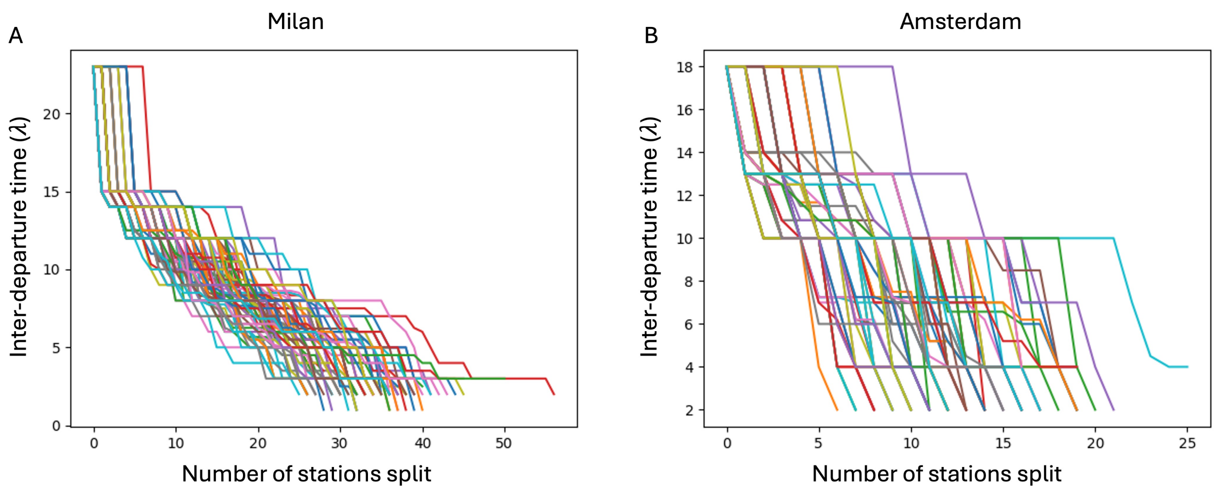

Figure 13: Optimizing the inter-departure time by splitting stations for **A** Milan and **B** Amsterdam. Each trajectory represents one iteration of splitting stations to achieve an average inter-departure time of 2.

5.2 Splitting stations achieves faster optimization

Another way to optimize the inter-departure times can be to split some stations into individual platforms. Briefly, we split the station in the critical circuit with the maximum in-degree into two new nodes and randomly assigned the incoming and outgoing connections of the original node to the new nodes (Figure 3).

Unlike adding trains, splitting stations has an element of randomness to it, which is reflected in the trajectories in Figure 13. Each line corresponds to one iteration of sequential station splitting to achieve the optimal inter-departure time. The number of splits required in Milan's case is higher than Amsterdam, which might be a consequence of the difference in the number of nodes in the original network. Milan's trajectories are more uniform as compared to Amsterdam, the implications of which are currently unclear. However, we do notice that splitting leads to disconnected components in the network.

6 Conclusion

In this paper, we use max-plus algebra to analyze the efficiency and stability of the subway systems in Amsterdam and Milan. We show their inter-departure times without adding extra trains, which reveals that the travel time between nodes in Amsterdam is shorter than in Milan. Then, we add trains to both of these two subway systems to make the inter-departure time λ to be 2. The results show that the Amsterdam Metro system needs more trains than the Milan Metro system to achieve the same level of inter-departure time.

Then, we analyze the robustness and stability of the Milan and Amsterdam metro systems against delay. Our results show that for both systems, the influence of the delay in suburban stations (without interchanging with many other subway lines) is much smaller than in the city center stations, where many subway lines intersect.

Moreover, in Milan, that is the larger metro system, the influence of the delay at a certain station will be smaller on the whole system than in a small metro system. Given the delay at a certain station, a smaller percentage of subway stations in Milan are influenced than in Amsterdam.

In addition, when ρ is fixed, after adding trains to the system, the system's robustness to delay will decrease. With more trains added, the inter-departure time λ and planned departure interval T will decrease simultaneously, which makes it easier for a constant time delay at one station to affect other stations.

Finally, we simulate the number of trains to be added to a random metro network to attain the same level of inter-departure time with Milan and Amsterdam metro system. The results show that the current metro system in Milan and Amsterdam are more efficient in terms of the train number than the random metro network.

7 Acknowledgements

This work is the output of the Complexity72h workshop, held at the Universidad Carlos III de Madrid in Leganés, Spain, 24-28 June 2024. https://www.complexity72h.com

References

Marc Barthelemy. *The Structure and Dynamics of Cities: Urban Data Analysis and Theoretical Modeling*. Cambridge University Press, 2016. URL `https://api.semanticscholar.org/CorpusID:29873735`.

Rob M. P. Goverde. Railway timetable stability analysis using max-plus system theory. *Transportation Research Part B: Methodological*, 41(2):179–201, February 2007. doi:10.1016/j.trb.2006.02.003. URL `https://doi.org/10.1016/j.trb.2006.02.003`.

X. T. Wu, M. K. Yang, W. B. Lian, M. Zhou, H. W. Wang, and H. R. Dong. Cascading delays for the high-speed rail network under different emergencies: A double layer network approach. *IEEE/CAA Journal of Automatica Sinica*, 10(10):2014–2025, October 2023. doi:10.1109/JAS.2022.105530.

Sybil Derrible. Network centrality of metro systems. *PLOS ONE*, 7(7):e40575, July 2012. doi:10.1371/journal.pone.0040575. URL `https://doi.org/10.1371/journal.pone.0040575`.

Bernd Heidergott, Geert Jan Olsder, and Jacob Van Der Woude. *Max Plus at work: modeling and analysis of synchronized systems: a course on Max-Plus algebra and its applications*, volume 48. Princeton University Press, 2014.

M. Abril, F. Barber, L. Ingolotti, M. A. Salido, P. Tormos, and A. Lova. An assessment of railway capacity. *Transportation Research Part E: Logistics and Transportation Review*, 44(5):774–806, September 2008. doi:10.1016/j.tre.2007.04.001. URL `https://doi.org/10.1016/j.tre.2007.04.001`.