Master of Science Thesis

An Exploratory Study in Max-Plus Linear Parameter Varying Systems

with Application to an Urban Railway Line

Ben Zwerink Arbonés

February 28, 2019

Delft Center for Systems and Control

Delft University of Technology

An Exploratory Study in Max-Plus Linear Parameter Varying Systems
with Application to an Urban Railway Line

Master of Science Thesis

For obtaining the degree of Master of Science in Mechanical Engineering at Delft University of Technology

Ben Zwerink Arbonés

February 28, 2019

Faculty of Mechanical, Maritime and Materials Engineering · Delft University of Technology

Delft University of Technology

Copyright © Delft Center for Systems and Control
All rights reserved.

Delft University Of Technology
Delft Center for Systems and Control

The undersigned hereby certify that they have read and recommend to the Faculty of Mechanical, Maritime and Materials Engineering for acceptance a thesis entitled **"An Exploratory Study in Max-Plus Linear Parameter Varying Systems"** by **Ben Zwerink Arbonés** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: February 28, 2019

Supervisor:

dr.ir. A.J.J. van den Boom

Readers:

dr. K. Batselier

A. Gupta

Abstract

Max-plus algebra is an algebra that is entirely based on the mathematical operations $\oplus = \max(\cdot)$ and $\otimes = +$, hence the name *max-plus* algebra. It can be used to describe Discrete Event Systems (DES) that require scheduling, such as a printer or train network. Max-plus algebra is studied because of its interesting properties, which make some non-linear systems in conventional algebra linear in max-plus algebra. These systems are called Max-Plus Linear (MPL) systems. This exploratory study introduces Max-Plus Linear Parameter Varying (MP-LPV) systems, systems that are not entirely linear in max-plus algebra but not that non-linear either, like LPV systems in conventional algebra. An urban railway line will be taken as an example of an MP-LPV system. Urban railway lines often operate relatively freely, with a passenger-dependent variable dwell time which can be modelled in max-plus algebra as a linear variable dependency on the arrival and departure times. It will be shown that such an MP-LPV system of an urban railway line can be rewritten to a set of linear inequalities, which can be used in an optimization framework to optimize for both minimal total passenger travel time and minimal absolute operation time. Some study cases will be shown in with it can be observed that these systems compute very rapidly, which makes a possible practical implementation interesting. Finally, some algebraic analysis on the MP-LPV system of an urban railway line will be done, such as on the definition of stability. But future work is still necessary on further analysis on the general class of MP-LPV systems.

Acknowledgements

I am very glad that you are reading this page, since it means that my master thesis is officially finished! Working on this thesis has been quite the adventure, but a very nice one indeed. It was the first time I every committed to such a large project on my own, which has given me many insights that I've never had before. Of course, the thesis in its current form would never have been possible without the help of quite many people...

First of all, I would like to show my gratitude to my supervisor Ton, whom without I could have never done a thesis on the subject of trains, my childhood dream! I want to thank him for all the knowledge, help and interesting talks about music during the past year. Being able to research a part of max-plus algebra that had never been studied before made me realize how hard it must be to keep on publishing new articles about the topic, once every now and then for the past 20 years.

Secondly, I would like to thank my parents for always supporting me, with special thanks to Núria for reading this whole paper to correct my mistakes, and Esther, for all her love, help and support during this master's programme.

And finally, a special thanks to all my friends with whom I've spent countless hours of writing, discussing, coffee-drinking and laughing. You are too many to mention, but you all know who you are!

I hope you will all enjoy this work!

Delft, University of Technology
February 28, 2019

Ben Zwerink Arbonés

Contents

Chapter 1

Introduction

Many processes around us can be described by mathematical models. These models describe a (simplified) system, for example in terms of a relation between what goes in and out of the system. Most of these systems are described by (conventional) algebra, which makes use of elementary operations like $+$ and $\times$ and has been studied extensively through the past thousands of years. Examples of systems that are described with conventional algebra are the rise of temperature in a room or the amount of current in an electrical wire.

But many systems can also be described in terms of different algebra. Take for instance the timetable of a train service or the scheduling of the order of tasks that a printer has to complete. These two systems have in common that they require an event-wise scheduling, and that one process can only start after the other one has finished. They (might) also repeat themselves after a specific cycle-time. These event-based systems are called Discrete Event Systems (DES) and can be described with event-wise algebra. The algebra that will be used in this paper is max-plus algebra, an algebra that is entirely based on two elementary operations, namely the *maximum* of two elements and the *summation* of two elements. These operations can be seen as the equivalence of summation and multiplication in regular algebra, respectively.

Many research has been done already on max-plus algebra (Heidergott et al., 2014). Most DES systems are non-linear when modelled in conventional algebra, but a specific class of systems that can be modelled by only using the max- and plus-operation are linear in max-plus algebra. Those systems are called Max-Plus Linear (MPL) systems and can be used to describe above-mentioned systems like printer queuing and train scheduling. A more recent addition to MPL systems are the so-called Switching Max-Plus Linear (SMPL) systems (van den Boom & De Schutter, 2006), where the relations between states and matrices are still linear in max-plus algebra, but where the systems matrices can switch between different modes. As an addition to the field of max-plus systems, this paper will introduce a new class of systems that use max-plus algebra: so-called Max-Plus Linear Parameter Varying (MP-LPV) systems.

While (S)MPL's were originally meant to be linear in max-plus algebra, MP-LPV systems are not: the system matrices can vary dependent on the states of the system, like regular

LPV systems in conventional algebra (Tóth et al., 2009). This means that there is an implicit relation between the left and right side of the equations which is hard to model in max-plus algebra. This paper will try to define this problem while also modelling such a system by using an urban railway line with variable dwell time as a case study.

Urban railway networks are a special case of railway networks that are seen as complicated systems to optimize (Wang, Ning, et al., 2015; Li & Lo, 2014; Sun et al., 2014), often resulting in a non-linear programming problem. While regular railway networks can be modelled (and synchronized) by using a timetable, urban railway networks often work with irregular intervals without timetabling. Research in the past years has mainly been conducted on energy saving, minimal passenger travel time, stop-skipping (i.e skipping stops that are less popular) by allowing these dynamics intervals, but up to the knowledge of the author none of these researches have been done within max-plus algebra. Different from regular railway networks (Kersbergen et al., 2016), urban railway lines contain variable dwell times that will not result into (S)MPL systems with constant matrices but in systems that will contain non-linear and implicit relations within max-plus algebra. The question is whether they can be modelled, controlled and analysed as MP-LPV systems and more importantly, if it has any use to do so.

The research goals are therefore defined as following:

- How can a Max-Plus Linear Parameter Varying (MP-LPV) system be defined and what are its properties?
- Is it possible to model an urban railway line with variable dwell times in max-plus algebra? And more importantly, can it modelled as an MP-LPV system?
- Can we control these systems with respect to an objective function in order to find a global optimum?

To answer these questions, this thesis will build up as following: in chapter 2, max-plus algebra and the concept of MP-LPV will be introduced to the reader. In chapter 3, urban railway modelling (in max-plus algebra) will be explained after which it will be presented as an MP-LPV. Chapter 4 will introduce control with respect to minimizing the total schedule of a predefined amount of trains and with respect to minimizing total passenger travel time, while chapter 5 will consist of some case studies showing to show some results. In chapter 6 MP-LPV systems will be analysed according to max-plus algebra theory. Finally, in chapter 7, there will be conclusions and suggestions for future work on this topic.

Chapter 2

Introduction to max-plus algebra

This chapter will mainly follow the theory and notation of Heidergott et al. (2014).

2-1 Max-plus algebra

Max-plus algebra is an algebra that focuses mainly on the use of two elementary operations, namely the *maximum* of two elements and the *summation* of two elements. Hence, the name *max-plus* algebra. Mathematically, for the relation between max-plus algebra and conventional algebra, we denote:

$$a \oplus b = \max(a, b) \tag{2-1}$$

$$a \otimes b = a + b \tag{2-2}$$

It is important to notice the order in max-plus algebra, whereas $\otimes$ (read: o-times) always has priority over $\oplus$ (read: o-plus). This is in line with regular algebra, where $\times$ has priority over $+$.

Conventional algebra also has two essential 'neutral' numbers, namely 0 and 1. A summation and multiplication with these numbers respectively will not have an influence on the outcome of a computation. In max-plus algebra, we will have to introduce a new '0' and '1' in order to introduce a neutral computation. These new numbers will be $\epsilon = -\infty$ and $e = 0$ respectively, because:

$$a \oplus \epsilon = \max(a, \epsilon) = \max(a, -\infty) = a \tag{2-3}$$

$$a \otimes e = a + e = a + 0 = a \tag{2-4}$$

More complex systems will require more complex notation. Therefore, we can introduce max-plus computations for matrices. If we define $A \in \mathbb{R}_{\max}^{n \times n}$ and $B \in \mathbb{R}_{\max}^{n \times n}$, then:

$$[A \oplus B]_{ij} = a_{ij} \oplus b_{ij} \tag{2-5}$$

$$[A \otimes B]_{ij} = \bigoplus_{k=1}^{l} a_{ik} \otimes b_{kj} \tag{2-6}$$

We can also denote the max-plus power function as:

$$A^{\otimes k} = \underbrace{A \otimes A \otimes ... \otimes A}_{\text{k times}} \tag{2-7}$$

An brief example of these matrix computations is given in the following Example 2-1:

Example 1.1 *Given* $A = \begin{pmatrix} e & \epsilon \\ 3 & 2 \end{pmatrix}$ *and* $B = \begin{pmatrix} -1 & 11 \\ 1 & \epsilon \end{pmatrix}$, *we find that:*

$$\begin{aligned}
A \oplus B &= \begin{pmatrix} \max(e, -1) & \max(\epsilon, 11) \\ \max(3, 1) & \max(2, \epsilon) \end{pmatrix} = \begin{pmatrix} e & 11 \\ 3 & 2 \end{pmatrix} \\
A \otimes B &= \begin{pmatrix} \max(e-1, \epsilon+1) & \max(e+11, \epsilon+\epsilon) \\ \max(3-1, 2+1) & \max(3+11, 2+\epsilon) \end{pmatrix} = \begin{pmatrix} -1 & 11 \\ 3 & 14 \end{pmatrix} \\
A^{\otimes 2} &= A \otimes A = \begin{pmatrix} e & \epsilon \\ 5 & 4 \end{pmatrix}
\end{aligned}$$

Equal to the scalar cases in (2-3) and (2-4), there also exists a max-plus version of the conventional identity-matrix I, denoted as the matrix E, and of the zero matrix 0, denoted as $\mathcal{E}$:

$$E = \begin{pmatrix} e & \epsilon & \cdots & \epsilon \\ \epsilon & e & \cdots & \epsilon \\ \vdots & \vdots & \ddots & \vdots \\ \epsilon & \epsilon & \cdots & e \end{pmatrix} \tag{2-8}$$

$$\mathcal{E} = \begin{pmatrix} \epsilon & \epsilon & \cdots & \epsilon \\ \epsilon & \epsilon & \cdots & \epsilon \\ \vdots & \vdots & \ddots & \vdots \\ \epsilon & \epsilon & \cdots & \epsilon \end{pmatrix} \tag{2-9}$$

such that:

$$A \oplus \mathcal{E} = A \tag{2-10}$$

$$A \otimes E = A \tag{2-11}$$

It will also be defined that $A^{\otimes e} \equiv E$ for $A, E \in \mathbb{R}_{\max}^{n \times n}$.

2-2 Max-plus linear systems

Max-plus algebra has many advantages over conventional algebra. The most beneficial one can be found in Max-Plus Linear (MPL) systems, defined as in (2-12) (De Schutter & van den Boom, 2008; Baccelli et al., 1992). These systems would be non-linear in conventional algebra, due to the max-operator, but are linear in max-plus algebra.

$$\begin{aligned} x(k) &= A \otimes x(k-1) \oplus B \otimes u(k) \\ y(k) &= C \otimes x(k) \end{aligned} \tag{2-12}$$

In line with conventional algebra, max-plus algebra also has eigenvalues and eigenvectors. Let $A \in \mathbb{R}_{\max}^{n \times n}$. If there exist $\lambda \in \mathbb{R}_{\max}$ and $v \in \mathbb{R}_{\max}^{n}$ (with $v \neq \mathcal{E}_{n \times 1}$) such that $A \otimes v = \lambda \otimes v$, then λ is a max-plus eigenvalue of A and v is a corresponding max-plus eigenvector. Every square matrix with entries in $\mathbb{R}_{\max}$ has at least one eigenvalue, but in contrast to conventional algebra the number of eigenvalues is in general lower than the dimension n. Furthermore, if a matrix is *irreducible*, then it has only one eigenvalue. The definition of irreducibility can be found in graph theory (Heidergott et al., 2014): a matrix A is irreducible when its graph $\mathcal{G}(A) = (\mathcal{N}, \mathcal{D})$, with node set $\mathcal{N}$ and arc set $\mathcal{D}$, is *strongly connected*. A graph $\mathcal{G}$ is strongly connected when for any two nodes $i, j \in \mathcal{N}$, node j is *reachable* from node i. A node j is reachable from node i if there exists a path between them.

The eigenvalue λ of the matrix A has some interesting properties. For instance, a finite eigenvalue of a matrix A is the *average weight* of some circuit in $\mathcal{G}(A)$. The average weight is defined as the *weight* of the circuit divided by the amount of arcs that connected the nodes of that circuit. A good example of a circuit would be a (circular) urban rail line, with every station being a node and the arcs being the connection between the stations. The values assigned to these arcs would be the travel times and the total weight of the circuit would be the summation of all these travel times, i.e. the time it takes to complete the circuit.

We can now also define a *normalized* matrix of A, namely A_λ (2-13). The original matrix A had eigenvalue (and thus average circuit weight) λ, so by subtracting λ it is clear that A_λ has average circuit weight (and thus eigenvalue) e.

$$[A_\lambda]_{ij} = a_{ij} - \lambda \tag{2-13}$$

Now first, we consider the max-plus linear system in (2-14)

$$x = A \otimes x \oplus b \tag{2-14}$$

with $A \in \mathbb{R}_{\max}^{n \times n}$ and dimension n. According to (Baccelli et al., 1992) (Theorem 2.1 in (Kersbergen et al., 2016)), the unique solution to this max-plus linear system is (2-15)

$$x = A^* \otimes b \tag{2-15}$$

where we can write A^* as (2-16):

$$A^* = E \oplus A \oplus A^2 \oplus ... \oplus A^\infty = \bigoplus_{p=0}^{\infty} A^p \tag{2-16}$$

In the case that there are no circuits with positive weights in the graph of A, $\mathcal{G}(A)$, Theorem 2.2 from (Kersbergen et al., 2016) shows that we can limit the sum to the dimension n of the matrix A, such that:

$$A^* = E \oplus A \oplus A^2 \oplus ... \oplus A^{n-1} \tag{2-17}$$

Now, the concept of implicit and explicit MPL systems can be introduced. Imagine the autonomous system in (2-18), without any control signal $u(k)$:

$$x(k) = A_0 \otimes x(k) \oplus A_1 \otimes x(k) \tag{2-18}$$

Since we have $x(k)$ on both the left and right side of the equation, this is called an *implicit* MPL system. The system in (2-12) was *explicit*, since it only contained past entries of the states $x(k)$ on the right side of the equation. But, by using (2-15) and (2-17), it can be found that the implicit relation in (2-18) can be rewritten to an explicit relation under certain conditions, as will be shown in the following example:

Example 1.2 *Consider the implicit MPL system of (2-18), with dimension $n = 4$. Given $A_0 = \begin{pmatrix} \epsilon & \epsilon & \epsilon & \epsilon \\ \epsilon & \epsilon & 3 & \epsilon \\ 4 & \epsilon & \epsilon & \epsilon \\ \epsilon & 5 & \epsilon & \epsilon \end{pmatrix}$ and $A_1 = \begin{pmatrix} 1 & \epsilon & \epsilon & \epsilon \\ 2 & 1 & \epsilon & \epsilon \\ 3 & 2 & 1 & \epsilon \\ 4 & 3 & 2 & 1 \end{pmatrix}$. Since the graph of A_0 does not contain any circuits, and thus also no circuits with positive weights, we can use (2-17) to find that:*

$$A^{\otimes n-1} = A^{\otimes 3} = \begin{pmatrix} \epsilon & \epsilon & \epsilon & \epsilon \\ \epsilon & \epsilon & \epsilon & \epsilon \\ \epsilon & \epsilon & \epsilon & \epsilon \\ 12 & \epsilon & \epsilon & \epsilon \end{pmatrix}$$

$$A^{\otimes 2} = \begin{pmatrix} \epsilon & \epsilon & \epsilon & \epsilon \\ 7 & \epsilon & \epsilon & \epsilon \\ \epsilon & \epsilon & \epsilon & \epsilon \\ \epsilon & \epsilon & 8 & \epsilon \end{pmatrix}$$

$$A_0^* = \bigoplus_{p=0}^{n-1} A^p = E \oplus A \oplus A^2 \oplus A^3 = \begin{pmatrix} e & \epsilon & \epsilon & \epsilon \\ 7 & 0 & 3 & \epsilon \\ 4 & \epsilon & 0 & \epsilon \\ 12 & 5 & 8 & 0 \end{pmatrix}$$

According to (2-15), if we take $b = A_1 \otimes x(k-1)$, we finally obtain the explicit relation:

$$x(k) = A_0^* \otimes b = A_0^* \otimes A_1 \otimes x(k-1) = A \otimes x(k-1)$$
$$A = A_0^* \otimes A_1 = \begin{pmatrix} e & \epsilon & \epsilon & \epsilon \\ 7 & 0 & 3 & \epsilon \\ 4 & \epsilon & 0 & \epsilon \\ 12 & 5 & 8 & 0 \end{pmatrix} \otimes \begin{pmatrix} 1 & \epsilon & \epsilon & \epsilon \\ 2 & 1 & \epsilon & \epsilon \\ 3 & 2 & 1 & \epsilon \\ 4 & 3 & 2 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \epsilon & \epsilon & \epsilon \\ 8 & 5 & 4 & \epsilon \\ 5 & 2 & 1 & \epsilon \\ 13 & 10 & 9 & 1 \end{pmatrix}$$

Combining (2-13) and (2-16) will result into the following theory: let the communication graph $\mathcal{G}(A)$ of matrix $A \in \mathbb{R}_{\max}^{n \times n}$ have finite maximal average circuit weight λ. Then, the scalar λ is an eigenvalue of A, and the column $[A_\lambda^*]_{\cdot\eta}$ is an eigenvector of A associated with λ, for any node η in the *critical graph* $\mathcal{G}^c(A)$ (2-19). Here, a critical graph is defined as a graph which contains all nodes and arcs that are contained within the *critical circuits* of $\mathcal{G}(A)$. A critical circuit is a circuit with maximal average weight.

$$A \otimes [A_\lambda^*]_{\cdot\eta} = \lambda \otimes [A_\lambda^*]_{\cdot\eta} \tag{2-19}$$

However, all this theory is based on graph theory. To find the eigenvalue λ of matrix A without graph theory, one can also use other methods (Heidergott et al., 2014; De Schutter & van den Boom, 2008). One of these lies within the cyclic behaviour of irreducible matrices: if $A \in \mathbb{R}_{\max}^{n \times n}$ is irreducible, then for a certain k and cycle c, (2-20) holds and the eigenvalue λ can be found.

$$A^{\otimes^{k+c}} = \lambda^{\otimes^c} \otimes A^{\otimes^k} \tag{2-20}$$

Example 1.3 *Given the irreducible matrix* $A = \begin{pmatrix} 0 & e & 2 \\ 2 & 0 & 4 \\ 1 & 2 & 3 \end{pmatrix}$ *with cyclicity* $c = 1$*, we find that:*

$$A^{\otimes^2} = \begin{pmatrix} 3 & 4 & 5 \\ 5 & 6 & 7 \\ 4 & 5 & 6 \end{pmatrix}, \quad A^{\otimes^3} = \begin{pmatrix} 6 & 7 & 8 \\ 8 & 9 & 10 \\ 7 & 8 & 9 \end{pmatrix}, \quad A^{\otimes^4} = \begin{pmatrix} 9 & 10 & 11 \\ 11 & 12 & 13 \\ 10 & 11 & 12 \end{pmatrix}, \quad \ldots$$

$$\textit{and thus that:} \quad A^{\otimes^{k+1}} = 3 \otimes A^{\otimes^k} \quad \text{for} \quad k = 2, 3, \ldots$$

2-3 Max-plus linear parameter varying systems

Up to the authors knowledge, this section is committed to an entirely new class of max-plus systems: so-called Max-Plus Linear Parameter Varying (MP-LPV) systems. Within max-plus algebra, regular max-plus linear systems or the more extended Switching Max-Plus Linear (SMPL) systems (van den Boom & De Schutter, 2006) are often used to describe discrete-event scheduling processes. These models are per definition linear in max-plus algebra, since the elemental operation $a \oplus b = \max(a, b)$ is linear in max-plus algebra. For this exploratory study, it will be investigated whether MP-LPV has the potential to be used and controlled like MPL and SMPL as well.

In conventional algebra, LPV systems are a special case of non-linear systems (Tóth et al., 2009), defined in discrete time as:

$$
\begin{aligned}
x[k+1] &= A[p]x[k] + B[p]u[k] \\
y[k] &= C[p]x[k] + D[p]u[k]
\end{aligned}
\tag{2-21}
$$

where $A[p]$, $B[p]$, $C[p]$ and $D[p]$ are system matrices dependent on the parameter $p = p[k]$. This parameter is - as the name Linear Parameter Varying suggests - a linear combination of one or multiple states of the system (Mohammadpour & Scherer, 2012). For instance, the difference between two scaled time-steps of the state x: $p[k] = ax[k] - bx[k-1]$ with a and b constants. Since the system-matrices are state-dependent, the system is in fact non-linear. But due to the special structure, there are still several ways to analyse and control such systems Tóth et al. (2009); Mohammadpour and Scherer (2012). The last citation appropriately describes LPV systems as 'a "middle ground" between linear and nonlinear dynamics'.

LPV systems have been introduced in order to systematically compute gain-scheduled control, and also to be able to systematically interpret the results and properties of non-linear systems (Gáspár et al., 2017). Before LPV systems were modelled as such, non-linear plants were often linearized around multiple operation points, after which interpolation or switching between the multiple controller gains was necessary. But these linearized points only guarantee stability and robustness around the operation points and not around the interpolated values between them. LPV control changed this and introduced a more valid robustness criterion for such non-linear systems. As mentioned before, 'non-linear' is a strong definition: these systems can often be modelled with a dependency on a linear operation.

LPV systems deal with time-varying matrices within a dynamical system. Within max-plus algebra, systems with variable system matrices are often modelled as SMPL's, which means that the matrices vary per event but that they are still constant. Under certain conditions, even with (bounded) uncertainty present in (S)MPL systems, it can be proven that these systems are stable (van den Boom & De Schutter, 2011).

But dealing with a situation where these matrices vary with dependency on the states is a new field of research. It has been observed that this class of systems appears when we try to model a train network with dependency on the time it spends at a platform, i.e. the dwell time. The amount of dwell time depends on the amount of passengers, which on itself depends on the arrival and departure times of the trains and of course on the time of the day. Since this means that arrival and departure times of trains are partly dependent on the arrival and departure times of these same trains, we can observe a self-loop that can best be described as a Max-Plus Linear Parameter Varying system of the form:

$$x(k) = A(p) \otimes x(k-1) \oplus B(p) \otimes u(k) \tag{2-22}$$

In (2-22), the matrices $A(p)$ and $B(p)$ are dependent on the varying parameter p. Similarly to the conventional algebra system description in (2-21), this parameter can be modelled as a linear relation between the states of the system. However, an important notice will be made here: in this thesis, this dependency will be modelled as a linear relation in *conventional algebra* and not in max-plus algebra. An example of such a conventional algebra-expression for the parameter p will be shown in section 3-5. Furthermore, this paper will focus on the situation where only $A(q)$ has dependency on the parameter p, and where $B(q) = B$ is a constant matrix. Furtermore, an implicit relation will be modelled instead of the explicit relation in (2-22). This means that $x(k)$ on the left side of the equation is equal to itself multiplied with a system matrix, in this case the LPV matrix $A_0(p)$. This general form of explicit linear parameter varying max-plus systems can be written as:

$$x(k) = A_0(p) \otimes x(k) \oplus A_1 \otimes x(k-1) \oplus B \otimes u(k) \tag{2-23}$$

where by using the Kleene star as defined in (2-15) and (2-17), it can be shown that (2-23) is equal to (2-22) when $A(p) = A_0^*(p) \otimes A_1$. Observe the difference between the two expressions: the right side of the equation is both dependent on the past (i.e., $k-1$) and present (i.e., k), where the latter makes this expression implicit. We have seen in (2-12) that an explicit max-plus linear system is only dependent on the past states.

Since the matrix $A(q)$ is state-dependent, it is not naturally to transform (2-23) back to (2-22) by using the Kleene star. Observe that $A(q)$ itself introduces yet another implicit relation between the left and right side of the equation in (2-23), since it is also dependent on scaled differences between the past and current cycles. In the regional railway network of (Kersbergen et al., 2016), only (switching) constant matrices appear and therefore the expression can be rewritten to an explicit model like in (2-12), where the left side only contains past entries $x(k-i)$ with $i > 0$. The implicit relation implies that current states are directly involved into the current outcome, and that a Kleene star product does not solve the implicit relation. It will be shown from section 3-6 onward that this problem can be solved iteratively.

From now on, the class of max-plus systems in (2-23) will be called Max-Plus Linear Parameter Varying (MP-LPV) in the rest of this paper, since the matrix $A_0(p)$ will depend on scaled (and past) versions of the the state $x(k)$. The case of MP-LPV will investigated by modelling an urban railway line in max-plus algebra. Different to most regional railway networks (Kersbergen et al., 2016), many urban railway networks in large cities operate relatively freely, because they arrive and leave whenever they are able and allowed to. If the arrival and departure times of a train k are based only on others trains and on the variable dwell time at stations, and thus not on some fixed scheduled arrival time $r_a(k)$ and departure time $r_d(k)$, then such a system can be seen as a MP-LPV system.

Since the MP-LPV system in this thesis is both implicit and p is expressed in conventional algebra, this thesis can be seen as an exploratory study into MP-LPV. This also means that the problem is all but fully described and that more research is needed into the stability of these systems. Stability will be briefly touched upon in chapter 6.

In the next chapter, it will be shown how the situation of MP-LPV occurs. It will summarize the use of max-plus notation to model an urban train line. The use of max-plus algebra is limited to what has been explained in this chapter so much more in-depth knowledge is not required for now.

Chapter 3

Modelling of an urban railway network in max-plus algebra

In this chapter, the modelling of a railway network in max-plus algebra will be discussed. Railway networks can be modelled in many different ways, but the use of max-plus algebra is a very convenient one since many non-linearities become linear in max-plus algebra. Take for instance the modelling of the Dutch railway networks in Kersbergen et al. (2016). One of the things that this master thesis tries to add to the previous work, is the addition of a variable dwell time, which is the case for many urban railways systems. Adding such a variation introduces a non-linearity in max-plus algebra, because the dwell times are now dependent on the states (arrival and departure times) of the system.

This problem will become more clear by the end of this chapter. First, the difference with the previous work done by Kersbergen et al. (2016) will be explained. Then, the constraints of an (urban) railway network will be mentioned. Finally, the constraints can be used to model the urban railway network in max-plus algebra.

3-1 Difference with previous studies

As mentioned before, the largest difference between this master thesis and the study by Kersbergen et al. (2016) will be the addition of a variable dwell time, dependent on the states of the system itself. Variable dwell times (or 'Passenger-demand scheduling' in some papers) are already considered and solved by (Wang, Ning, et al., 2015), (Sun et al., 2014) and (Niu et al., 2015) for instance, but all of these studies do not work within the max-plus framework. Often, the problem becomes a non-linear or non-convex optimization problem. For on-line computations, it is of large importance that the optimization problem can be solved quick and efficiently, which is often possible with discrete-event algebra like max-plus algebra. Furthermore, it will be interesting to be able to extrude some information out of the max-plus system, for instance on the (max-plus) eigenvalue of the system, and see if this has any useful physical interpretation.

Another difference with Kersbergen et al. (2016) is the switch from a cycle-counter to a product-counter k (van den Boom, T., unpublished work, 2019). The reason for this is the switch from a regional train *network* to an urban train *line*. Due to the nature of most urban train lines, trains can not overtake each other and therefore their order remains the same for the whole duration of the day. Furthermore, controlling only one line instead of multiple train lines makes it unnecessary to consider multiple trains at once. A product-counter simply makes the model more insightful, because every state $x(k)$ will be a direct representation of the arrival and departure times of train k.

3-2 Constraints

The modelling of a train network will be done following the procedure in Kersbergen et al. (2016), in which a train network has been documented in max-plus notation. In his work, six different constraint are mentioned that have to be taken into account:

- Running time constraints
- Dwell time constraints
- Headway constraints
- Timetable constraints
- Coupling constraints
- Connection constraints

A satisfying observation is that not all of these constraints have to be modelled for an urban train line. Many constraints like the transfer from one train to another will be neglected in this study, so they don't have to be taken into account. In the following subsections, the constraints will be described in further detail. Notation-wise, the following things should be noted: in this thesis, the event-counter k is considered to be a product-counter instead of a cycle-counter. This means that k is the train index, and the departure times $d_j(k)$ and $a_j(k)$ represent the arrival and departure times at station j.

3-2-1 Running time constraints

A running time constraint simply deals with the time between arrival and departure of a train between the stations $j-1$ and j as:

$$a_j(k) \geq d_{j-1}(k) + \tau_{\mathrm{r,min},j-1}(k) \tag{3-1}$$

In other words; a train k can not arrive at the station j before this minimal traversing time from station $j-1$ to j.

3-2-2 Dwell time constraints

The dwell time constraint (also called continuity constraint) is an inequality that tries to connect the departure and arrival time at station j to each other in the following simple manner:

$$d_j(k) \geq a_j(k) + \tau_{\mathrm{d},j}(k) \tag{3-2}$$

where $\tau_{\mathrm{d},j}(k)$ is the variable dwell time. One goal of this thesis is to make this dwell time dependent on the amount of passengers on the platform, which on itself is dependent on the amount of trains in service and the time of the day. This will introduce a non-linear term in the max-plus algebra. Later in chapter, it will be explained how this dwell time can be modelled as a variable dependent on the departure and arrival times of previous and current trains k.

3-2-3 Headway constraints

To keep a sufficient (time-)distance between two trains, headway constraints are necessary. To make sure that two trains do not approach each other to closely, these margins are necessary to maintain safety. Mathematically, we can write that:

$$d_j(k) \geq d_{j+1}(k-1) + \tau_{\mathrm{h,d},j}(k) \tag{3-3}$$
$$a_j(k) \geq a_{j+1}(k-1) + \tau_{\mathrm{h,a},j}(k) \tag{3-4}$$

In order to have some kind of structure in the metro service - and often because of the absence of timetable constraints -, these constraints can be seen as some kind of minimal distance between two trains, to maintain some margins. More on this can be found in section 3-3.

3-2-4 Timetable constraints

A timetable constraint is simply a guarantee that trains cannot depart before the actual departure time according to a predefined timetable. Sometimes, trains are also not allowed to enter a station before the timetable says they can. Mathematically, we can denote that:

$$d_j(k) \geq r_{\mathrm{d},j}(k) \tag{3-5}$$
$$a_j(k) \geq r_{\mathrm{a},j}(k) \tag{3-6}$$

However, in the case of an urban train line without a timetable - like the urban train line in this study-, these constraints can be neglected. This can be done by setting both $r_{\mathrm{a},j}(k)$ and $r_{\mathrm{d},j}(k)$ to ϵ, such that these inequalities are always true.

Another option is to maintain these constraints as a way to control the network. By replacing $r_{\mathrm{a},j}(k)$ and $r_{\mathrm{d},j}(k)$ with $u_{\mathrm{a},j}(k)$ and $u_{\mathrm{d},j}(k)$, we find that the railway network is only controlled when the control inputs are sufficiently large. Note that because of the $\geq$-sign, this way of incorporating control does only allow to delay the schedule.

3-2-5 Coupling and connection constraints

In train networks, it can occur that two trains have to be coupled to each other. This could be modelled as following:

$$d_j(k) \geq d_{j-1}(k) \tag{3-7}$$
$$d_{j-1}(k) \geq d_j(k) \tag{3-8}$$
$$a_j(k) \geq a_{j-1}(k) \tag{3-9}$$
$$a_{j-1}(k) \geq a_j(k) \tag{3-10}$$

However, for urban train lines like a metro, coupling does normally not take place. Therefore, we can and will neglect these constraints. The same will be done with the connection constraints. These normally guaranty that passengers are able to transfer from one train to another. In practice, this means that one train cannot leave if the other train did not arrive yet. Besides that, it is necessary to have some extra *connection time* $\tau_{c,j}(k)$ at station j for the passengers to change trains.

Even though this is an important part for both trains and metro's, and any public transport in general, this paper will start with the modelling and control of a single isolated urban rail line. This means that there is no need for any transfer constraint yet, as there is no other line to transfer to. Furthermore, to transform this constraint from a cycle- to a product-counter, it would be necessary to have a much finer definition of the product-counter k. For instance, are all train lines defined as different counters k_1, k_2, etc., or differently? It has been chosen to not answer this question for now and just neglect this constraint.

3-3 Modelling the max-plus equations

If the arrival and departure times of train k at station j are defined as $a_j(k)$ and $d_j(k)$ respectively, then we can write out the max-plus equations. But before this is done, there are some assumptions that can be made to ease the modelling for urban rail transportation compared to regional rail transportation. The assumptions are as following:

1. There is only one train track (in every direction); trains can not overtake each other.
2. There can only be one train waiting on a station; a subsequent train should wait in the tunnel if it is closely following its predecessor.
3. There is a maximum velocity in every segment of the urban train line; trains can not run faster between two stations than within a certain minimum running time.
4. It is assumed that all passengers arrive between the departure of a train and the arrival of its successor; no passengers arrive during the dwell time.
5. In the uncontrolled case, trains leave as soon as all conditions are satisfied. In the controlled case, trains may be delayed to improve the overall performance of the urban railway line.

Assumption 1 makes most urban train lines different from its regional counterparts, because having trains run over multiple tracks makes the model different: by having one track per direction only, trains always have the same order so there is no need to consider different orders of trains. Also, in regional rail modelling, there should always be enough headway time τ_{h} between two trains - even when they are running over different tracks - to make sure that they do not interfere once they come back together again. But in the urban rail case, it is enough to guarantee some minimum safety headway time between two trains. This brings us to assumption number 2: due to physical limitations, it is often not possible to stall two trains in one stop. Therefore, we can define a minimum headway time $\tau_{\mathrm{h,min}}$ between a departing train k and a soon-to-arrive train $k+1$ on station j which guarantees that these trains will not physically touch each other. Assumption 3 is also based on physical limitations: even if a train can run faster to make up for delays, it is not allowed to do so because of the facilities it uses; tunnels and train tracks are often older than the trains and should be used responsibly to guarantee safety. However, this assumption eases our model because a train can not arrive before a certain minimum running time $\tau_{\mathrm{r,min}}$. Assumption 4 can be done by assuming that most passengers arrive before the train arrives, and thus that relatively no passengers arrive during the dwell time of train k. This assumption is very important for the MP-LPV model of the urban railway line, because it reduces the implicit complexity of the matrix $A_0(q)$. More on this in section 3-6. Finally, assumption 5 allows us to write the urban railway line as a max-plus system with an equality constraint. It also indicates that the max-plus system can be rewritten to linear inequality constraints as long as one of the inequalities is an equality, the so called extended linear complementarity problem (ELCP) (De Schutter & van den Boom, 2001).

Last but not least, a train can only depart from station j after a certain dwell time τ_{d} to allow passengers to alight and board the train. This leaves us with the following equations:

$$a_j(k) \geq d_{j-1}(k) + \tau_{\mathrm{r,min},j-1} \tag{3-11}$$
$$a_j(k) \geq d_j(k-1) + \tau_{\mathrm{h,min}} \tag{3-12}$$
$$d_j(k) \geq a_j(k) + \tau_{\mathrm{d},j}(k) \tag{3-13}$$

where $\tau_{\mathrm{r,min},j-1}(k)$ is the minimum running time between station $j-1$ and station j and $\tau_{\mathrm{d},j}(k)$ is the dwell time of train k at station j. Because the dwell time can change over time, dependent on the amount of passengers, it is variable over time and per station. We can observe that these constraints can be written as a maximum expression as following:

$$a_j(k) = \max(d_{j-1}(k) + \tau_{\mathrm{r,min},j-1}, d_j(k-1) + \tau_{\mathrm{h,min}}) \tag{3-14}$$
$$d_j(k) = a_j(k) + \tau_{\mathrm{d},j}(k) \tag{3-15}$$

and then be translated to a max-plus notation, in the following way:

$$a_j(k) = d_{j-1}(k) \otimes \tau_{\mathrm{r,min},j-1} \oplus d_j(k-1) \otimes \tau_{\mathrm{h,min}} \tag{3-16}$$
$$d_j(k) = a_j(k) \otimes \tau_{\mathrm{d},j}(k) \tag{3-17}$$

If we denote a state $x(k)$ being all the arrival and departure times of train k together, then we can easily rewrite the set of equations to a max-plus system.

$$x(k) = \left(\begin{array}{c} a_1(k) \\ a_2(k) \\ \vdots \\ a_J(k) \\ \hline d_1(k) \\ d_2(k) \\ \vdots \\ d_J(k) \end{array}\right) = \left(\begin{array}{c} d_0(k) \otimes \tau_{\mathrm{r,min,0}} \oplus d_1(k-1) \otimes \tau_{\mathrm{h,min}} \\ d_1(k) \otimes \tau_{\mathrm{r,min,1}} \oplus d_2(k-1) \otimes \tau_{\mathrm{h,min}} \\ \vdots \\ d_{J-1}(k) \otimes \tau_{\mathrm{r,min},J-1} \oplus d_J(k-1) \otimes \tau_{\mathrm{h,min}} \\ \hline a_1(k) \otimes \tau_{\mathrm{d},1}(k) \\ a_2(k) \otimes \tau_{\mathrm{d},2}(k) \\ \vdots \\ a_J(k) \otimes \tau_{\mathrm{d},J}(k) \end{array}\right) \tag{3-18}$$

Now it can be observed that the arrival and departure times of train k depend on both the previous train $k-1$ and on itself. If we consider all the unnecessary variables in this system as ϵ, as for instance $d_0(k)$, then we can rewrite the equation into the following max-plus state-space equation:

$$\begin{aligned} x(k) = {} & \underbrace{\left(\begin{array}{c|c} \mathcal{E} & \begin{matrix} \epsilon & \epsilon & \cdots & \epsilon & \epsilon \\ \tau_{\mathrm{r,min,1}} & \epsilon & \cdots & \epsilon & \epsilon \\ \epsilon & \tau_{\mathrm{r,min,2}} & \cdots & \epsilon & \epsilon \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \epsilon & \epsilon & \cdots & \tau_{\mathrm{r,min}\,.J-1} & \epsilon \end{matrix} \\ \hline \begin{matrix} \tau_{\mathrm{d},1}(k) & \epsilon & \cdots & \epsilon \\ \epsilon & \tau_{\mathrm{d},2}(k) & \cdots & \epsilon \\ \vdots & \vdots & \ddots & \vdots \\ \epsilon & \epsilon & \cdots & \tau_{\mathrm{d},J}(k) \end{matrix} & \mathcal{E} \end{array}\right)}_{A_0(p)} \otimes x(k) \\ & \oplus \underbrace{\left(\begin{array}{c|c} \mathcal{E} & \begin{matrix} \tau_{\mathrm{h,min}} & \epsilon & \cdots & \epsilon \\ \epsilon & \tau_{\mathrm{h,min}} & \cdots & \epsilon \\ \vdots & \vdots & \ddots & \vdots \\ \epsilon & \epsilon & \cdots & \tau_{\mathrm{h,min}} \end{matrix} \\ \hline \mathcal{E} & \mathcal{E} \end{array}\right)}_{A_1} \otimes x(k-1) \end{aligned} \tag{3-19}$$

As can be seen from (3-19), not only the states $x(k)$ but also the dwell times $\tau_{\mathrm{d},j}(k)$ are dependent on the train-counter k. The exact relation is yet unknown and will be called p, which means that we have a dependency on p in the A_0-matrix. Therefore, we can conclude that we have an autonomous Max-Plus Linear Parameter Varying (MP-LPV) system from the form:

$$x(k) = A_0(p) \otimes x(k) \oplus A_1 \otimes x(k-1) \tag{3-20}$$

where the lack of linearity comes from the $A_0(p)$ matrix. This can be seen from (3-27) and the derivation that led to it.

The system is autonomous because it contains no control input $u(k)$. Else, it would look like the following equation:

$$x(k) = A_0(p) \otimes x(k) \oplus A_1 \otimes x(k-1) \oplus u(k) \tag{3-21}$$

In this form, it is possible to regulate the metro network with the help of a control signal $u(k)$. This would for example be an amount of time to delay the arrival or departure time of train k, if necessary. This will be discussed in more detail in section 4-4.

But before we get into controlling this system, the modelling of passengers will be discussed in 3-4.

3-4 Passenger-dependent demand

Wang et al. have performed research on both Origin-Destination-independent and Origin-Destination-dependent rail scheduling (Wang, Ning, et al., 2015) and passenger-demands-oriented train scheduling (Wang, Tang, et al., 2015), both for urban rail. In both papers, the optimization process is done with respect to minimizing both total travel time and energy consumption. The first paper looks at a time-driven model while the second one looks at an event-driven model, according to the author mainly for computational efficiency and to better model time-varying origin-destination passenger demand. This section is entirely based on the formulation in those papers, except for some notation.

Regarding OD-independent passenger modeling, (Wang, Ning, et al., 2015) does the following assumptions: first, passengers arrive at a constant rate λ_j at station j. Second, the number of passengers that exit a train at station j is a fixed proportion ρ_j of the number of passengers that are on board of the train when it arrives at station j. Third, the number of passenger on platforms and in the trains are approximated by real numbers. The first assumption is done because for short headway times, people arrive randomly over the time interval between two trains and therefore can be seen as a uniform distribution. Assumption two can be done by analysing historical data and by estimating a constant average value from this data. The third assumption is done because integer numbers simplify their optimization process, and the error by rounding up or down is relatively small with large number of passengers. However, it will not simplify the optimization process in this paper since there will be no Mixed-Integer Linear Programming (MILP) problem. Therefore, all but the third assumption will be used for the modelling of passengers in this paper.

The passenger demand characteristics are formulated as following: the number of passengers remaining on the platform after train $k-1$ departed from station j is $w_j^{\text{remain}}(k-1)$, such that we can write:

$$w_j^{\text{wait}}(k) = w_j^{\text{remain}}(k-1) + \lambda_j(d_j(k) - d_j(k-1)) \tag{3-22}$$

with $d_j(k)$ and $d_j(k-1)$ the departures of trains k and $k-1$ respectively. However, since we decided in assumption 5 of section 3-3 that we can ignore the amount of arriving passengers during the dwell time, we can rewrite (3-22) into:

$$w_j^{\text{wait}}(k) = w_j^{\text{remain}}(k-1) + \lambda_j(a_j(k) - d_j(k-1)) \tag{3-23}$$

If the number of passengers on trains k and $k-1$ are $n_j(k)$ and $n_j(k-1)$ respectively, then the remaining capacity of the train after departure from station j is:

$$n_j^{\text{remain}}(k) = C_{\text{max}}(k) - n_{j-1}(k) \cdot (1 - \rho_j) \tag{3-24}$$

with $C_{\text{max}}(k)$ being the effective maximal capacity of train k. The number of passengers at station j that can board is now:

$$n_j^{\text{board}}(k) = \min(n_j^{\text{remain}}(k), w_j^{\text{wait}}(k)) \tag{3-25}$$

and, in case of a full train, the remaining passengers on the platform can be found to be $w_j^{\text{remain}}(k) = w_j^{\text{wait}}(k) - n_j^{\text{board}}(k)$. Furthermore, the number of passengers on train k after departing from station j is:

$$n_j(k) = n_{j-1}(k) \cdot (1 - \rho_j) + n_j^{\text{board}}(k) \tag{3-26}$$

which is simply the amount of people on the train after departing station $j-1$, times the proportion of people that stayed the train and the amount of new people that has boarded.

The dwell time is being modelled as a variable, which means that boarding and alighting passengers are being taken into account. This can be done either with a linear (3-27) or a non-linear (3-28) model. Here, $\tau_{\text{d,min},j}(k)$ is the minimum dwell time, $\alpha_{1,d}$, $\alpha_{2,d}$, $\alpha_{3,d}$ and $\alpha_{4,d}$ are coefficients that can be estimated from historical data, and n^{door} is the amount of doors of the train.

$$\tau_{\text{d,min},j}(k) = \alpha_{1,d} + \alpha_{2,d} \cdot n_{j-1}(k) \cdot \rho_j + \alpha_{3,d} \cdot n_j^{\text{board}}(k) \tag{3-27}$$

$$\tau_{\text{d,min},j}(k) = \alpha_{1,d} + \alpha_{2,d} \cdot n_j^{\text{alight}}(k) + \alpha_{3,d} \cdot n_j^{\text{board}}(k) + \alpha_{4,d} \cdot \left(\frac{w_j^{\text{wait}}(k)}{n^{\text{door}}}\right)^3 \cdot n_j^{\text{board}}(k) \tag{3-28}$$

If $\tilde{\tau}_{\text{min}}$ is the minimum dwell time set by the operator and $\tau_{\text{d,max},j}(k)$ is the maximum 'allowed' dwell time for train k at station j, then:

$$\max(\tilde{\tau}_{\text{min}}, \tau_{\text{d,min},j}(k)) \leq \tau_{\text{d},j}(k) \leq \tau_{\text{d,max},j}(k) \tag{3-29}$$

For OD-dependent rail scheduling, some slight alterations are done to the passenger modelling procedure. In OD-independent rail scheduling, passengers are considered to have no specific destination and they are therefore modelled as percentages of the total on-board passengers that leave at a certain stop. However, OD-data is available for operators, which means that the number of passengers with a certain trajectory can be estimated over time. The largest difference with the OD-independent rail scheduling problem is therefore the more specifically defined amount of passengers that are waiting on a platform; the number of waiting passengers $w_j^{\text{wait}}(k)$ with destination m for instance is:

$$w_{j,m}^{\text{wait}}(k) = w_{j,m}(k-1) + \lambda_{j,m}(d_j(k) - d_j(k-1)) \tag{3-30}$$

where every term is now defined in terms of destination m as well. The sum of all destinations m now gives the total amount of waiting passengers for train k at station j:

$$w_j^{\text{wait}}(k) = \sum_{m=j+1}^{J} w_{j,m}^{\text{wait}}(k) \tag{3-31}$$

3-5 The max-plus linear parameter varying system

Now that we have defined passenger-dependent demand, we can incorporate this into the MP-LPV that was defined in the end of section 3-3. First of all, from (3-23), we find that the amount for passengers $w_{\text{wait}}(k)$ waiting at station j is dependent on the arriving time $a_j(k)$ of train k and departure time $d_j(k-1)$ of train $k+1$. We also found in (3-27) that the minimum dwell time $\tau_{d,\min,j}(k)$ for train k at station j is based on the amount of passengers $n_{j-1}(k)$ that were on board of train k since station $j-1$ and on the amount of passengers $n_j^{\text{board}}(k)$ that board train k at station j. And because $n_{j-1}(k)$ is calculated from $n_{j-2}(k)$ and $n_j^{\text{board}}(k-1)$ (3-26), and $n_j^{\text{board}}(k)$ is calculated from $w_{\text{wait}}(k)$, we find that recursively all these variables depend somehow on arriving and departure times of previous- and current trains $\{k, k-1, k-2, ...\}$ at stations $\{j, j-1, j-2, ...\}$. If we further assume that all trains have infinite capacity $C_{\max}(k)$ such that there is always room for more passengers in train k, then the min-operation in (3-25) is unnecessary and the amount of boarding passengers is just the amount of waiting passengers:

$$n_j^{\text{board}}(k) = w_j^{\text{wait}}(k) \tag{3-32}$$

Also, there are no remaining passengers $w_j^{\text{remain}}(k)$ on the platform any more, because all of them fitted in the train. Therefore:

$$w_j^{\text{wait}}(k) = \lambda_j(a_j(k) - d_j(k-1)) \tag{3-33}$$

This results into the following expression for the minimal dwell time:

$$\begin{aligned}\tau_{\text{d},\min,j}(k) &= \alpha_{1,d} + \alpha_{2,d} \cdot n_{j-1}(k) \cdot \rho_j + \alpha_{3,d} \cdot n_j^{\text{board}}(k) \\ &= \alpha_{1,d} + \alpha_{2,d} \cdot \Big(n_{j-2}(k) \cdot (1-\rho_{j-1}) + n_{j-1}^{\text{board}}(k)\Big) \cdot \rho_j + \alpha_{3,d} \cdot n_j^{\text{board}}(k)\end{aligned}$$

$$
\begin{aligned}
&= \alpha_{1,d} + \alpha_{2,d} \cdot \Big(n_{j-3}(k) \cdot (1-\rho_{j-2})(1-\rho_{j-1}) + n_{j-2}^{\text{board}}(k) \cdot (1-\rho_{j-1}) + n_{j-1}^{\text{board}}(k) \Big) \cdot \rho_j \\
&\quad + \alpha_{3,d} \cdot n_j^{\text{board}}(k) \\
&\qquad \vdots \\
&= \alpha_{1,d} + \alpha_{2,d} \cdot \Big(\underbrace{n_0(k)}_{0} \cdot (1-\rho_1)(1-\rho_2)(...)(1-\rho_{j-1}) + n_1^{\text{board}}(k) \cdot (1-\rho_2)(1-\rho_3)(...)(1-\rho_{j-1}) \\
&\quad + ... + n_{j-1}^{\text{board}}(k) \Big) \cdot \rho_j + \alpha_{3,d} \cdot n_j^{\text{board}}(k) \\
&= \alpha_{1,d} + \alpha_{2,d} \cdot \Big(n_1^{\text{board}}(k) \cdot (1-\rho_2)(1-\rho_3)(...)(1-\rho_{j-1}) + ... + n_{j-1}^{\text{board}}(k) \Big) \cdot \rho_j \\
&\quad + \alpha_{3,d} \cdot n_j^{\text{board}}(k)
\end{aligned}
$$

$$
= \alpha_{1,d} + \begin{pmatrix} \alpha_{3,d} \\ \alpha_{2,d}\rho_j \\ \alpha_{2,d}(1-\rho_{j-1})\rho_j \\ \vdots \\ \alpha_{2,d}(1-\rho_2)(1-\rho_3)(...)(1-\rho_{j-1})\rho_j \end{pmatrix}^T \cdot \begin{pmatrix} n_j^{\text{board}}(k) \\ n_{j-1}^{\text{board}}(k) \\ n_{j-2}^{\text{board}}(k) \\ \vdots \\ n_1^{\text{board}}(k) \end{pmatrix}
$$

The dwell time $\tau_{\text{d,min},j}(k)$ for train k at station j has now recursively been reduced to a formula that shows dependency only on the proportion of passengers ρ_j that leave the train at station j and the amount of passengers that board train k at station j (and past stations). If we then use what was found in (3-32) and (3-33), we find that:

$$
\tau_{\text{d,min},j}(k) = \alpha_{1,d} + \begin{pmatrix} \alpha_{3,d} \\ \alpha_{2,d}\rho_j \\ \alpha_{2,d}(1-\rho_{j-1})\rho_j \\ \vdots \\ \alpha_{2,d}(1-\rho_2)(1-\rho_3)(...)(1-\rho_{j-1})\rho_j \end{pmatrix}^T \cdot \begin{pmatrix} w_j^{\text{wait}}(k) \\ w_{j-1}^{\text{wait}}(k) \\ w_{j-2}^{\text{wait}}(k) \\ \vdots \\ w_1^{\text{wait}}(k) \end{pmatrix} \tag{3-34}
$$

The length of the expression for the dwell time $\tau_{\text{d,min},j}(k)$ logically depends on the station counter j; if j is large, then the expression that determines the dwell time will also be large because all previous j's have to be taken into account. For $j = 1$ for example, the expression reduces to $\tau_{\text{d,min},1}(k) = \alpha_{1,d} + \alpha_{3,d} \cdot w_1^{\text{wait}}(k)$. Proceeding with this for $j = \{1, ..., J\}$ gives us the following:

$$\tau_{\text{d,min},1}(k) = \alpha_{1,d} + \alpha_{3,d} \cdot w_1^{\text{wait}}(k) = \alpha_{1,d} + \underbrace{\begin{pmatrix} \alpha_{3,d} \\ 0 \\ \vdots \\ 0 \end{pmatrix}^T}_{\Phi_1} \cdot \begin{pmatrix} w_1^{\text{wait}}(k) \\ w_2^{\text{wait}}(k) \\ \vdots \\ w_J^{\text{wait}}(k) \end{pmatrix}$$

$$\tau_{\text{d,min},2}(k) = \alpha_{1,d} + \alpha_{3,d} \cdot w_2^{\text{wait}}(k) + \alpha_{2,d} \cdot \rho_2 \cdot w_2^{\text{wait}}(k) = \alpha_{1,d} + \underbrace{\begin{pmatrix} \alpha_{2,d} \cdot \rho_2 \\ \alpha_{3,d} \\ \vdots \\ 0 \end{pmatrix}^T}_{\Phi_2} \cdot \begin{pmatrix} w_1^{\text{wait}}(k) \\ w_2^{\text{wait}}(k) \\ \vdots \\ w_J^{\text{wait}}(k) \end{pmatrix}$$

$$\vdots$$

$$\begin{aligned} \tau_{\text{d,min},J}(k) &= \alpha_{1,d} + \alpha_{3,d} \cdot w_J^{\text{wait}}(k) + ... + \alpha_{2,d} \cdot \rho_J(1-\rho_{J-1})(...)(1-\rho_3)(1-\rho_2) \cdot w_1^{\text{wait}}(k) \\ &= \alpha_{1,d} + \underbrace{\begin{pmatrix} \alpha_{2,d} \cdot \rho_J(1-\rho_{J-1})(...)(1-\rho_3)(1-\rho_2) \\ \alpha_{2,d} \cdot \rho_{J-1}(1-\rho_{J-2})(...)(1-\rho_2) \\ \vdots \\ \alpha_{3,d} \end{pmatrix}^T}_{\Phi_J} \cdot \begin{pmatrix} w_1^{\text{wait}}(k) \\ w_2^{\text{wait}}(k) \\ \vdots \\ w_J^{\text{wait}}(k) \end{pmatrix} \end{aligned}$$

Now, the structure of $w_j^{\text{wait}}(k)$ can be exploited. As mentioned before, it is dependent on the arriving time of train k and the departure time of train $k-1$ (3-33). In (3-18), state vector $x(k)$ was defined as the collection of all arrival and departure times $a_j(k)$ and $d_j(k)$, such that $a_j(k) = x_j(k)$ and $d_j(k) = x_{j+J}(k)$. Therefore, we can define a vector $\boldsymbol{w}^{\text{wait}}(k)$ as following:

$$\boldsymbol{w}^{\text{wait}}(k) = \begin{pmatrix} w_1^{\text{wait}}(k) \\ w_2^{\text{wait}}(k) \\ \vdots \\ w_J^{\text{wait}}(k) \end{pmatrix} \tag{3-35}$$

and if $\boldsymbol{I} \in \mathbb{R}^{J\times J}$ is the the unity matrix with only ones on the diagonal, the zero matrix is $\mathbf{0} \in \mathbb{R}^{J\times J}$, the unity vector is $\bar{\mathbf{1}} \in \mathbb{R}^{1\times J}$, and the zeroes vector is $\bar{\mathbf{0}} \in \mathbb{R}^{1\times J}$, it can be written that:

$$\boldsymbol{w}^{\text{wait}}(k) = \boldsymbol{\lambda}\left(\begin{bmatrix}\boldsymbol{I} & \mathbf{0}\end{bmatrix} x(k) - \begin{bmatrix}\mathbf{0} & \boldsymbol{I}\end{bmatrix} x(k-1)\right) = \begin{bmatrix}\boldsymbol{\lambda} & \mathbf{0}\end{bmatrix} x(k) - \begin{bmatrix}\mathbf{0} & \boldsymbol{\lambda}\end{bmatrix} x(k-1) \tag{3-36}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^{J\times J}$ is a diagonal matrix containing all λ_j.

Finally, everything can be substituted in (3-19) by defining:

$$
\begin{aligned}
\boldsymbol{\tau}_{\mathrm{d,min}}(k) = \begin{pmatrix} \tau_{\mathrm{d,min},1}(k) \\ \tau_{\mathrm{d,min},2}(k) \\ \vdots \\ \tau_{\mathrm{d,min},J}(k) \end{pmatrix} &= \boldsymbol{\alpha}_{1,d} + \begin{pmatrix} \Phi_1 \\ \Phi_2 \\ \vdots \\ \Phi_J \end{pmatrix} \boldsymbol{w}^{\mathrm{wait}}(k) \\
&= \boldsymbol{\alpha}_{1,d} + \begin{pmatrix} \Phi_1 \\ \Phi_2 \\ \vdots \\ \Phi_J \end{pmatrix} \begin{bmatrix} \boldsymbol{\lambda} & \mathbf{0} \end{bmatrix} x(k) - \begin{pmatrix} \Phi_1 \\ \Phi_2 \\ \vdots \\ \Phi_J \end{pmatrix} \begin{bmatrix} \mathbf{0} & \boldsymbol{\lambda} \end{bmatrix} x(k-1) \\
&= \boldsymbol{\alpha}_{1,d} + \begin{bmatrix} \Psi & \mathbf{0} \end{bmatrix} x(k) - \begin{bmatrix} \mathbf{0} & \Psi \end{bmatrix} x(k-1)
\end{aligned}
\tag{3-37}
$$

where $\boldsymbol{\alpha}_{1,d} \in \mathbb{R}^{1\times J}$ is simply a vector filled with $\alpha_{1,d}$ and where $\Psi = \Phi\boldsymbol{\lambda}$. If we now forget about the upper boundary of (3-29), we see that $\tau_{\mathrm{d},j}(k) \geq \max(\tilde{\tau}_{\min}, \tau_{\mathrm{d,min},j}(k))$ or in max-plus algebra that $\tau_{\mathrm{d},j}(k) = \tilde{\tau}_{\min} \oplus \tau_{\mathrm{d,min},j}(k)$, which means that we can reinterpret (3-16) and write:

$$
a_j(k) = d_{j-1}(k) \otimes \tau_{\mathrm{r,min},j-1} \oplus d_j(k-1) \otimes \tau_{\mathrm{h,min}} \tag{3-38}
$$
$$
d_j(k) = a_j(k) \otimes (\tilde{\tau}_{\min} \oplus \tau_{\mathrm{d,min},j}(k)) = a_j(k) \otimes \tilde{\tau}_{\min} \oplus a_j(k) \otimes \tau_{\mathrm{d,min},j}(k) \tag{3-39}
$$

and thus that the max-plus system becomes:

$$
\begin{aligned}
x(k) = &\underbrace{\left(\begin{array}{cccc|ccccc}
 & & & & \epsilon & \epsilon & \cdots & \epsilon & \epsilon \\
 & & & & \tau_{\mathrm{r,min},1} & \epsilon & \cdots & \epsilon & \epsilon \\
 & \mathcal{E} & & & \epsilon & \tau_{\mathrm{r,min},2} & \cdots & \epsilon & \epsilon \\
 & & & & \vdots & \vdots & \ddots & \vdots & \vdots \\
 & & & & \epsilon & \epsilon & \cdots & \tau_{\mathrm{r,min}\,.J-1} & \epsilon \\
\hline
\tilde{\tau}_{\min} \oplus \tau_{\mathrm{d,min},1}(k) & \epsilon & \cdots & \epsilon & & & & & \\
\epsilon & \tilde{\tau}_{\min} \oplus \tau_{\mathrm{d,min},2}(k) & \cdots & \epsilon & & & \mathcal{E} & & \\
\vdots & \vdots & \ddots & \vdots & & & & & \\
\epsilon & \epsilon & \cdots & \tilde{\tau}_{\min} \oplus \tau_{\mathrm{d,min},J}(k) & & & & &
\end{array}\right)}_{A_0(p)} \otimes x(k) \\
&\oplus \underbrace{\left(\begin{array}{c|cccc}
 & \tau_{\mathrm{h,min}} & \epsilon & \cdots & \epsilon \\
\mathcal{E} & \epsilon & \tau_{\mathrm{h,min}} & \cdots & \epsilon \\
 & \vdots & \vdots & \ddots & \vdots \\
 & \epsilon & \epsilon & \cdots & \tau_{\mathrm{h,min}} \\
\hline
\mathcal{E} & & \mathcal{E} & &
\end{array}\right)}_{A_1} \otimes x(k-1)
\end{aligned}
\tag{3-40}
$$

We can observe that sub-matrix $A_0^{[2,1]}(p) = \mathrm{diag}^{\oplus}(\tilde{\tau}_{\min}) \oplus \mathrm{diag}^{\oplus}(p(k))$, where the $\mathrm{diag}^{\oplus}(\cdot)$

operator makes a max-plus matrix with the arguments a on the diagonal and ϵ everywhere else, and where $p(k)$ can be defined as a vector with the dwell times:

$$p(k) = \boldsymbol{\tau}_{\mathrm{d,min}}(k) = \boldsymbol{\alpha}_{1,d} + \begin{bmatrix}\Psi & \mathbf{0}\end{bmatrix} x(k) - \begin{bmatrix}\mathbf{0} & \Psi\end{bmatrix} x(k-1) \tag{3-41}$$

As can be seen from (3-41), $p(k)$ is defined in conventional algebra and clearly shows the (scaled) relation between $x(k)$ and $x(k-1)$, and how $A_0(p)$ depends on it. To make a clearer distinction between the part of $A_0(p)$ that is dependent on $x(k)$ and $x(k+1)$ and the part that is not, we can split $A_0(p)$ into $\tilde{A}_0$ and $\hat{A}_0(p)$, where $A_0(p) = \hat{A}_0(p) \oplus \tilde{A}_0$. Then:

$$\begin{aligned} x(k) = & \underbrace{\left(\begin{array}{c|c} \mathcal{E} & \begin{matrix} \epsilon & \epsilon & \cdots & \epsilon & \epsilon \\ \tau_{\mathrm{r,min},1} & \epsilon & \cdots & \epsilon & \epsilon \\ \epsilon & \tau_{\mathrm{r,min},2} & \cdots & \epsilon & \epsilon \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \epsilon & \epsilon & \cdots & \tau_{\mathrm{r,min}\,.J-1} & \epsilon \end{matrix} \\ \hline \begin{matrix} \tilde{\tau}_{\mathrm{min}} & \epsilon & \cdots & \epsilon \\ \epsilon & \tilde{\tau}_{\mathrm{min}} & \cdots & \epsilon \\ \vdots & \vdots & \ddots & \vdots \\ \epsilon & \epsilon & \cdots & \tilde{\tau}_{\mathrm{min}} \end{matrix} & \mathcal{E} \end{array}\right)}_{\tilde{A}_0} \otimes x(k) \\ & \oplus \underbrace{\left(\begin{array}{c|c} \mathcal{E} & \mathcal{E} \\ \hline \begin{matrix} \tau_{\mathrm{d,min},1}(k) & \epsilon & \cdots & \epsilon \\ \epsilon & \tau_{\mathrm{d,min},2}(k) & \cdots & \epsilon \\ \vdots & \vdots & \ddots & \vdots \\ \epsilon & \epsilon & \cdots & \tau_{\mathrm{d,min},J}(k) \end{matrix} & \mathcal{E} \end{array}\right)}_{\hat{A}_0(p)} \otimes x(k) \oplus \underbrace{\left(\begin{array}{c|c} \mathcal{E} & \begin{matrix} \tau_{\mathrm{h,min}} & \epsilon & \cdots & \epsilon \\ \epsilon & \tau_{\mathrm{h,min}} & \cdots & \epsilon \\ \vdots & \vdots & \ddots & \vdots \\ \epsilon & \epsilon & \cdots & \tau_{\mathrm{h,min}} \end{matrix} \\ \hline \mathcal{E} & \mathcal{E} \end{array}\right)}_{A_1} \otimes x(k-1) \end{aligned} \tag{3-42}$$

It can be observed that $\tilde{A}_0$ and A_1 are constant matrices, independent of the varying train counter and the changing arrival and departure times. $\hat{A}_0(p)$ [1] is the matrix that will depend on the arrival and departure times of train k and $k-1$ according to (3-41).

3-6 Exploring the structure of $\hat{A}_0(p)$

The structure of the max-plus system (still without control) has been defined in the previous section. It has become clear from (3-37) that one of the limiting factors is the dependency of $\hat{A}_0(p)$ on the schedule of the current and previous trains $x(k)$ and $x(k-1)$. However, in

[1] Note: when $\hat{A}_0(p)$ is written, in fact it should be $\hat{A}_0(p(k)))$. But for convenience and readability, the dependency of $p(k)$ on k and $k-1$ is assumed automatically.

order to find the schedule of the current train $x(k)$, $\hat{A}_0(p)$ has to be known. This tendency is therefore a vicious circle and not directly solvable.

The first problem is the implicit relation between the left and right side of the equation. In (Kersbergen et al., 2016), the problem is solved by going from an implicit to an explicit formulation by using the so-called Kleene star A^*. In order to do so, the Kleene star is computed out of A, as in (2-17). However, in order to compute the Kleene star of a matrix A, that matrix A has to be constant. Because the A-matrix in this thesis is varying for every train k, it can not be used in the same way. This directly implicates the second problem: because of the dependency on $x(k)$ and $x(k-1)$, no matter what we do, $\hat{A}_0(p)$ will remain implicit. One way to tackle this problem is by solving (3-42) for every k iteratively. This means that the calculation has to be done step-by-step, until the scheduling problem $x(k)$ is solved and $\hat{A}_0(p)$ is constant. The following procedure can be applied:

```
For every x(k):

    Calculate an initial x_i(k) based on known information x(k-1) and A_1
    Determine the dwell times tau_d(k) based on initial x_i(k) and known x(k-1)
    Find the A_0_i(p) matrix according to the dwell times tau_d(k)
    Calculate the next iterative schedule x_i+1(k) based on A_0_i(p), x_i(k) and x(k-1)

    Repeat until x_i+1(k) = x_i(k)
        Return  x(k)       = x_i+1(k)
                A_0(p)     = A_0_i(p)
                Index(k)   = i
```

Because of all the max- and plus-operations, it is guaranteed that the next iteration is always 'later' than or equal to the current one. Ideally, there will be a moment where the next iteration $x^{i+1}(k)$ is strictly equal to the current iteration $x^i(k)$, even if this means that initially the train has to be delayed at some stations. Apparently, that delay is necessary in order to fulfil all the max-plus constraints and the train can not run faster, because it would violate one or more of these constraints (see Lemma 1 and **Proof** underneath). The initial condition $x^i(k)$ is always valid, because it consists of a combination of the prior knowledge $x(k-1)$, which is always known at the start of $x(k)$, and the matrix A_1, which is a collection of the minimal travelling times between the stations. Therefore, train k is never allowed to run faster than the initial schedule $x^i(k)$, and $x^i(k)$ can now be used to determine the corresponding dwell times. The script will stop when the arriving and departure times in $x^i(k)$ do not change any more, resulting into a static equation in which $A_i(p)$ becomes constant (i.e. $\hat{A}_0^{i+1}(p) = \hat{A}_0^i(p) \rightarrow \hat{A}_0(p)$) and $x^{i+1}(k) = x^i(k) \rightarrow x(k)$. The train schedule $x(k)$ can then be used to define the initial schedule $x^i(k+1)$ for the next train, and this goes on until the schedule of all K trains is determined. Last but not least, for every train k, the index-number i is returned to show how many iterations it took to reach for a constant solution.

Lemma 1 (Existence of x(k))

Let $x(k-1)$ be an initial feasible schedule. Then under assumptions 4 and 5 in section 3-3, it is guaranteed that $x(k)$ exists.

Proof *Since $x(k-1)$ exists and is feasible, we can use the relations in (3-11) to see that we can always compute $x(k)$, even if $p(k) = \boldsymbol{\tau}_{d,\min}(k)$ is variable. Since a train k has to leave as soon as it is allowed to, we have that the maximum of the inequalities is an equality constraint. Also, since the arrival time $a_j(k)$ is only based on constant constraints, it can always be found as long as either the departure time $d_j(k-1)$ of train $k-1$ or $d_{j-1}(k)$ of train k is known. The first one is always finite since $x(k-1)$ exists and is feasible. This means that $a_j(k)$ can* *only become infinitely large is $d_{j-1}(k)$ was also infinitely large. But $d_{j-1}(k)$ can only become infinitely large if the variable dwell time is infinitely large. The variable dwell times are given as a vector in (3-41). Since all the elements of Ψ are finite and even smaller than 1, we know that the variable dwell times can only become infinitely large if one of the arrival or departure times in $x(k)$ is infinitely large.*

Assume that $a_1(k)$, the arrival time of station $j = 1$ exists because train simply has to start from this station. Now, $\tau_{\mathrm{d,min},1}(k) = \alpha_{1,d} + \Psi_{1,1}a_1(k) - \Psi_{1,1}d_1(k-1)$ can be computed and is finite, therefore $d_1(k)$ also exists. From station 1 on, $d_{j-1}(k)$ and $a_j(k)$ are always known and finite, the latter one because it is just a summation of a constant with a existing variable. Therefore, and since the structure of Ψ is lower-triangular in such a way that it only takes into account past arriving times, all the arrival and departure times of $x(k)$ can be computed and therefore, $x(k)$ exists. ■

Because of assumption 4 in section 3-3, there can not exist a self-loop where the departure time is dependent on the departure time itself. This could potentially become unstable if $\Psi_{j,j}$ is sufficiently large. For simplification, imagine that $p(k) = \boldsymbol{\tau}_{\mathrm{d,min}}(k)$ in (3-41) would not be dependent on the arrival times but on the departure times. We would find that for $j = 1$ that $\tau_{\mathrm{d,min},1}(k) = \alpha_{1,d} + \Psi_{1,1}d_1(k) - \Psi_{1,1}d_1(k-1)$, and thus that $d_1(k) \geq a_1(k) + \alpha_{1,d} + \Psi_{1,1}d_1(k) - \Psi_{1,1}d_1(k-1)$, resulting into:

$$\begin{aligned} d_1(k) &\geq a_1(k) + \alpha_{1,d} + \Psi_{1,1}d_1(k) - \Psi_{1,1}d_1(k-1) \\ (1-\Psi_{1,1})d_1(k) &\geq a_1(k) + \alpha_{1,d} - \Psi_{1,1}d_1(k-1) \\ d_1(k) &\geq \frac{1}{1-\Psi_{1,1}}a_1(k) + \frac{1}{1-\Psi_{1,1}}\alpha_{1,d} - \frac{1}{1-\Psi_{1,1}}\Psi_{1,1}d_1(k-1) \end{aligned}$$

where if $\Psi_{1,1}$ becomes too large, i.e. goes to 1, then the departure time $d_1(k)$ would go to infinity since $\frac{1}{1-\Psi_{1,1}} \to \frac{1}{1-1} = \frac{1}{0} = \infty$. This is why assumption 4 helps to ease the model; once a train k arrives at a station j, its dwell time is fixed.

Lemma 1 shows us that $x(k)$ exists. Furthermore, it can be shown that if $x(k)$ exists, that it is also unique according to Lemma 2.

Lemma 2 (Uniqueness of x(k))

Let $\hat{x}(k)$ be an existing schedule of train k, with the equality $\hat{x}(k) = \hat{A}_0(\hat{p}) \otimes \hat{x}(k) \oplus \tilde{A}_0 \otimes \hat{x}(k) \oplus A_1 \otimes x(k-1)$ where $\tilde{A}_0$, A_1 and $x(k-1)$ are all known. Then we know that $\hat{x}(k)$ is a unique solution.

Proof *This proof will be done by contradiction. Imagine a time schedule $\tilde{x}(k)$ for train k that is quicker then the 'optimal' free-run $\hat{x}(k)$, because $\tilde{x}(k) = \hat{x}(k) - \gamma$, where γ is a arbitrarily chosen positive vector because all its entries $\{\gamma_1, ..., \gamma_{2J}\}$ are positive but not necessarily equal to each other. Then, according to (3-42), we find find that:*

$$\begin{aligned} \tilde{x}(k) &= \hat{A}_0(\tilde{p}) \otimes \tilde{x}(k) \oplus \tilde{A}_0 \otimes \tilde{x}(k) \oplus A_1 \otimes x(k-1) \\ &= \hat{A}_0(\tilde{p}) \otimes (\hat{x}(k) - \gamma) \oplus \tilde{A}_0 \otimes (\hat{x}(k) - \gamma) \oplus A_1 \otimes x(k-1) \end{aligned}$$

with $\hat{A}_0(\tilde{p})$ being defined according to (3-37):

$$\begin{aligned}
\tilde{\boldsymbol{\tau}}_{\mathrm{d,min}}(k) &= \boldsymbol{\alpha}_{1,d} + \begin{pmatrix}\Phi_1\\ \Phi_2\\ \vdots\\ \Phi_J\end{pmatrix}\begin{bmatrix}\boldsymbol{\lambda} & \mathbf{0}\end{bmatrix}(\hat{x}(k)-\gamma) - \begin{pmatrix}\Phi_1\\ \Phi_2\\ \vdots\\ \Phi_J\end{pmatrix}\begin{bmatrix}\mathbf{0} & \boldsymbol{\lambda}\end{bmatrix}x(k-1)\\
&= \boldsymbol{\alpha}_{1,d} + \begin{pmatrix}\Phi_1\\ \Phi_2\\ \vdots\\ \Phi_J\end{pmatrix}\begin{bmatrix}\boldsymbol{\lambda} & \mathbf{0}\end{bmatrix}\hat{x}(k) - \begin{pmatrix}\Phi_1\\ \Phi_2\\ \vdots\\ \Phi_J\end{pmatrix}\begin{bmatrix}\mathbf{0} & \boldsymbol{\lambda}\end{bmatrix}x(k-1) - \begin{pmatrix}\Phi_1\\ \Phi_2\\ \vdots\\ \Phi_J\end{pmatrix}\begin{bmatrix}\boldsymbol{\lambda} & \mathbf{0}\end{bmatrix}\gamma)\\
&= \boldsymbol{\tau}_{\mathrm{d,min}}(k) - \begin{pmatrix}\Phi_1\\ \Phi_2\\ \vdots\\ \Phi_J\end{pmatrix}\begin{bmatrix}\boldsymbol{\lambda} & \mathbf{0}\end{bmatrix}\gamma)
\end{aligned}$$

and so:

$$\hat{A}_0(\tilde{p}) = \left(\begin{array}{cccc|c}
\multicolumn{4}{c|}{\mathcal{E}} & \mathcal{E}\\
\hline
\tau_{\mathrm{d,min},1}(k) - (\Phi\begin{bmatrix}\boldsymbol{\lambda} & \mathbf{0}\end{bmatrix}\gamma)_1 & \epsilon & \cdots & \epsilon & \\
\epsilon & \tau_{\mathrm{d,min},2}(k) - (\Phi\begin{bmatrix}\boldsymbol{\lambda} & \mathbf{0}\end{bmatrix}\gamma)_2 & \cdots & \epsilon & \mathcal{E}\\
\vdots & \vdots & \ddots & \vdots & \\
\epsilon & \epsilon & \cdots & \tau_{\mathrm{d,min},J}(k) - (\Phi\begin{bmatrix}\boldsymbol{\lambda} & \mathbf{0}\end{bmatrix}\gamma)_J &
\end{array}\right)$$

Now, we transform the first equation back to the max of the three subcomponents to find that:

$$\begin{aligned}
\tilde{x}(k) &= \hat{A}_0(\tilde{p})\otimes(\hat{x}(k)-\gamma)\oplus\tilde{A}_0\otimes(\hat{x}(k)-\gamma)\oplus A_1\otimes x(k-1)\\
&= \max(\hat{A}_0(\tilde{p})\otimes(\hat{x}(k)-\gamma), \tilde{A}_0\otimes(\hat{x}(k)-\gamma), A_1\otimes x(k-1))\\
&\geq \begin{cases}\hat{A}_0(\tilde{p})\otimes(\hat{x}(k)-\gamma)\\ \tilde{A}_0\otimes(\hat{x}(k)-\gamma)\\ A_1\otimes x(k-1))\end{cases}
\end{aligned}$$

If we look at the first inequality, we see that:

$$\begin{aligned}
\tilde{x}(k) &\geq \hat{A}_0(\tilde{p})\otimes(\hat{x}(k)-\gamma)\\
\hat{x}(k)-\gamma &\geq \hat{A}_0(\tilde{p})\otimes(\hat{x}(k)-\gamma)
\end{aligned}$$

To show the logic behind this proof, it is best to split $\hat{x}(k)$ and $\tilde{x}(k)$ into an upper and lower half, both halves of a vector being $\mathbb{R}^{J\times 1}$. If we write that $\Psi = \Phi\boldsymbol{\lambda}$, we find that:

$$
\begin{aligned}
\hat{x}_l(k) &\geq \tilde{\boldsymbol{\tau}}_{d,\min}(k) - \begin{bmatrix}\Psi & \mathbf{0}\end{bmatrix}\gamma + \hat{x}_u(k) + \begin{bmatrix}\mathbf{0} & I\end{bmatrix}\gamma \\
&\geq \underbrace{\tilde{\boldsymbol{\tau}}_{d,\min}(k) + \hat{x}_u(k)}_{\hat{x}_l(k)} - \begin{bmatrix}\Psi & \mathbf{0}\end{bmatrix}\gamma + \begin{bmatrix}\mathbf{0} & I\end{bmatrix}\gamma \\
&\geq \hat{x}_l(k) + \begin{bmatrix}-\boldsymbol{\Psi} & I\end{bmatrix}\gamma \\
0 &\geq \begin{bmatrix}-\boldsymbol{\Psi} & I\end{bmatrix}\gamma
\end{aligned}
$$

Similarly, the same kind of argumentation can be done for the second inequality, since:

$$
\begin{aligned}
\tilde{x}(k) &\geq \tilde{A}_0 \otimes (\hat{x}(k) - \gamma) \\
\hat{x}(k) - \gamma &\geq \tilde{A}_0 \otimes (\hat{x}(k) - \gamma) \\
\hat{x}(k) - \gamma &\geq \begin{pmatrix} \epsilon \\ \tau_{\mathrm{r,min},1} + \hat{x}_{J+1}(k) - \gamma_{J+1} \\ \vdots \\ \tau_{\mathrm{r,min},J-1} + \hat{x}_{2J-1}(k) - \gamma_{2J-1} \\ \tilde{\tau}_{\min} + \hat{x}_1(k) - \gamma_1 \\ \vdots \\ \tilde{\tau}_{\min} + \hat{x}_J(k) - \gamma_J \end{pmatrix} \\
\hat{x}(k) - \gamma &\geq \hat{x}(k) - \left(\begin{array}{c|cccc} & 0 & 0 & \cdots & 0 \\ & 1 & 0 & \cdots & 0 \\ 0 & \vdots & \ddots & \ddots & \vdots \\ & 0 & \cdots & 1 & 0 \\ \hline I & & & 0 & \end{array}\right)\gamma \\
0 &\geq \left(\begin{array}{c|cccc} & 0 & 0 & \cdots & 0 \\ & -1 & 0 & \cdots & 0 \\ I & \vdots & \ddots & \ddots & \vdots \\ & 0 & \cdots & -1 & 0 \\ \hline -I & & & I & \end{array}\right)\gamma
\end{aligned}
$$

For the last inequality, we simply find that:

$$
\begin{aligned}
\tilde{x}(k) &\geq A_1 \otimes x(k-1) \\
\hat{x}(k) - \gamma &\geq \underbrace{A_1 \otimes x(k-1)}_{\hat{x}(k)} \\
0 &\geq \gamma
\end{aligned}
$$

By simply combining all these constraints into one big matrix, we find that:

$$0 \geq \underbrace{\left(\begin{array}{c|cccc} -\Psi & & I & & \\ \hdashline & 0 & 0 & \cdots & 0 \\ & -1 & 0 & \cdots & 0 \\ I & \vdots & \ddots & \ddots & \vdots \\ & 0 & \cdots & -1 & 0 \\ \hline -I & & I & & \\ \hdashline I & & 0 & & \\ \hline 0 & & I & & \end{array}\right)}_{V} \gamma$$

and since γ is strictly positive and $V^T V$ is positive definite for every value possible in the matrix Ψ (because $0 < \Psi_{i,j} \leq 1$), the right side of this equation will always be larger than 0, therefore not allowing the inequality to be true. Furthermore, we can observe that the easiest satisfaction of this inequality would be $\gamma = 0$, which would indicate that $\tilde{x}(k) = \hat{x}(k)$ and therefore that $\hat{x}(k)$ is a unique solution. ■

3-7 Expanding for multiple train cycles

The arrival and departure times of train k can be found according to (3-42), but for optimization purposes it is necessary to consider multiple trains cycles at once in order to anticipate on the control action that might have to be taken in the future. Notice that up until now, control action is still not taken into account yet.

If we consider the same strategy as in Kersbergen et al. (2016), then multiple trains cycles can be combined into one vector $\breve{x}(k) = \begin{bmatrix} x^T(k) & x^T(k+1) & \ldots & x^T(k+M-1) \end{bmatrix}^T$. Here, M is the total amount of train cycles that should be considered at once, the so-called planning or prediction horizon. We then find the following equation:

$$\breve{x}(k) = \breve{A}(\breve{p}) \otimes \breve{x}(k) \oplus \breve{A}_0 \otimes \breve{x}(k) \oplus \breve{A}_1 \otimes x(k-1) \tag{3-43}$$

where $\breve{A}(\breve{p})$ [2], $\breve{A}_0$ and $\breve{A}_1$ are defined as:

[2]Note: with $\breve{p}$, the dependency on the collection of vectors $\{p(k), p(k+1), ..., p(k+M-1)\}$ is meant.

$$\breve{A}(\breve{p}) = \begin{bmatrix} \hat{A}_0(p(k)) & \mathcal{E} & \cdots & \mathcal{E} \\ A_1 \otimes \hat{A}_0(p(k)) & \hat{A}_0(p(k+1)) & \cdots & \mathcal{E} \\ \vdots & \vdots & \ddots & \vdots \\ A_1^{\otimes M-1} \otimes \hat{A}_0(p(k)) & A_1^{\otimes M-2} \otimes \hat{A}_0(p(k+1)) & \cdots & \hat{A}_0(p(k+M-1)) \end{bmatrix} \tag{3-44}$$

$$\breve{A}_0 = \begin{bmatrix} \tilde{A}_0 & \mathcal{E} & \cdots & \mathcal{E} \\ A_1 \otimes \tilde{A}_0 & \tilde{A}_0 & \cdots & \mathcal{E} \\ \vdots & \vdots & \ddots & \vdots \\ A_1^{\otimes M-1} \otimes \tilde{A}_0 & A_1^{\otimes M-2} \otimes \tilde{A}_0 & \cdots & \tilde{A}_0 \end{bmatrix} \tag{3-45}$$

$$\breve{A}_1 = \begin{bmatrix} A_1 \\ A_1^{\otimes 2} \\ \vdots \\ A_1^{\otimes M-1} \end{bmatrix} \tag{3-46}$$

A similar iterative procedure as before can now be applied to find the matrix $\breve{A}(\breve{p})$:

```
For every x_breve(k):

    Calculate an initial x_breve_i(k) based on known information x(k-1) and A_breve_1
    Determine the dwell times tau_d_breve(k) based on initial x_breve_i(k) and known x(k-1)
    Find the A_breve_i(p) matrix according to the dwell times tau_breve_d(k)
    Calculate the next iterative schedule x_breve_i+1(k) based on A_breve_i(p), x_breve_i(k) and x(k
        -1)

    Repeat until x_breve_i+1(k) = x_breve_i(k)
        Return   x_breve(k)      = x__breve_i+1(k)
                 A_breve(p)      = A_breve_i(p)
                 Index(k)        = i
```

To be able to model disturbances, it is also wise to add some control variable $\breve{d}(k)$ to this formulation. This can be done by considering the original inequalities and by adding $d_{a,j}(k)$ to delay the arrival time and $d_{d,j}(k)$ to delay the departure time, where $d_{a,j}(k)$ and $d_{d,j}(k)$ should not be confused with the departure times $d_j(k)$.

$$a_j(k) \geq d_{j-1}(k) + \tau_{\text{r,min},j-1} \tag{3-47}$$
$$a_j(k) \geq d_j(k-1) + \tau_{\text{h,min}} \tag{3-48}$$
$$a_j(k) \geq d_{a,j}(k) \tag{3-49}$$
$$d_j(k) \geq a_j(k) + \tau_{\text{d},j}(k) \tag{3-50}$$
$$d_j(k) \geq d_{d,j}(k) \tag{3-51}$$

and therefore also in the single-cycle max-plus system that we saw before:

$$x(k) = \hat{A}_0(p) \otimes x(k) \oplus \tilde{A}_0 \otimes x(k) \oplus A_1 \otimes x(k-1) \oplus B \otimes d(k) \tag{3-52}$$

where $d(k) = \begin{pmatrix} d_{a,1}(k) & \cdots & d_{a,J}(k) & d_{d,1}(k) & \cdots & d_{d,J}(k) \end{pmatrix}$ and $B = E$. Scaling this up to multiple cycles at once (Section 3-7) results into (3-53):

$$\breve{x}(k) = \breve{A}(\breve{p}) \otimes \breve{x}(k) \oplus \breve{A}_0 \otimes \breve{x}(k) \oplus \breve{A}_1 \otimes x(k-1) \oplus \breve{B} \otimes \breve{d}(k) \tag{3-53}$$

where $\breve{B}$ and $\breve{d}(k)$ are defined as:

$$\breve{B} = \begin{bmatrix} B & \mathcal{E} & \cdots & \mathcal{E} \\ A_1 \otimes B & B & \cdots & \mathcal{E} \\ \vdots & \vdots & \ddots & \vdots \\ A_1^{\otimes M-1} \otimes B & A_1^{\otimes M-2} \otimes B & \cdots & B \end{bmatrix} = \begin{bmatrix} E & \mathcal{E} & \cdots & \mathcal{E} \\ A_1 & E & \cdots & \mathcal{E} \\ \vdots & \vdots & \ddots & \vdots \\ A_1^{\otimes M-1} & A_1^{\otimes M-2} & \cdots & E \end{bmatrix} \tag{3-54}$$

$$\breve{d}(k) = \begin{pmatrix} d(k) \\ d(k+1) \\ \vdots \\ d(k+M-1) \end{pmatrix} \tag{3-55}$$

Now that the max-plus model for an urban train line is found, we can move on to the next chapter. In the next chapter, the addition of control will be discussed. Furthermore, a direct translation to an optimization problem is suggested.

Chapter 4

Controlling an urban train line

An important part for a model of an urban rail network is the ability to control it. There are several ways to define control action. One way for example is to use the amount of passengers to define the ideal frequency for trains, which results into an optimal headway time $\tau_{\mathrm{h}}(k)$ and consequently into optimal departure and arrival times. However, the max-plus framework only allows delaying the departure and arrival times, because the $\oplus$-operator always takes the maximum of the control signal $u(k)$ and the time vectors $x(k)$ and $x(k-1)$. This is why, if there is need for an earlier departure, the current max-plus formulation does not allow that.

Therefore, it might be more interesting to solve an optimization problem over a certain prediction horizon and to find an ideal train schedule in this way. In order to do so, it is interesting to see if the max-plus framework can be rewritten back to linear inequalities, in order to find the optimal control law $\breve{u}(k)$. Such an action would be a relaxation of the problem and therefore it does not necessarily have to be that the best schedule for the relaxed problem works as well for the max-plus framework. Within the framework of linear inequalities, optimization can be used to find the best possible schedule $\breve{x}_{\mathrm{opt}}(k)$. If this ideal $\breve{x}_{\mathrm{opt}}(k)$ gives the same cost in the max-plus framework as it gives in the linear inequalities framework, then the transformation is valid. In short, we would like to replace:

$$\begin{aligned} &\min_{\breve{x}(k)} \quad J_{\mathrm{obj}} \\ &\text{s.t.} \quad \breve{x}(k) = \breve{A}(\breve{p}) \otimes \breve{x}(k) \oplus \breve{A}_0 \otimes \breve{x}(k) \oplus \breve{A}_1 \otimes x(k-1) \oplus \breve{B} \otimes \breve{d}(k) \end{aligned} \tag{4-1}$$

with

$$\begin{aligned} &\min_{\breve{x}(k)} \quad J_{\mathrm{obj}} \\ &\text{s.t.} \quad F_0 \cdot \breve{x}(k) + F_1 \cdot x(k-1) \leq g_0 + G_d \cdot \breve{d}(k) \end{aligned} \tag{4-2}$$

without any different results. To achieve this result, this chapter builds up as following: in section 4-1, it will be shown that an uncontrolled case of an urban railway line can become unstable over time and thus that control is necessary. In section 4-2, the max-plus model will be expressed in a set of linear inequalities. In section 4-3, the many options for an objective function will be discussed. Finally, in section 4-4 the addition of a control signal to the original MP-LPV equations will be discussed, and also the conditions are discussed which allow us to do so.

4-1 Uncontrolled case study of an urban railway line

In this section, it will be shown that there are cases where control is absolutely necessary in order to maintain a stable rail line.

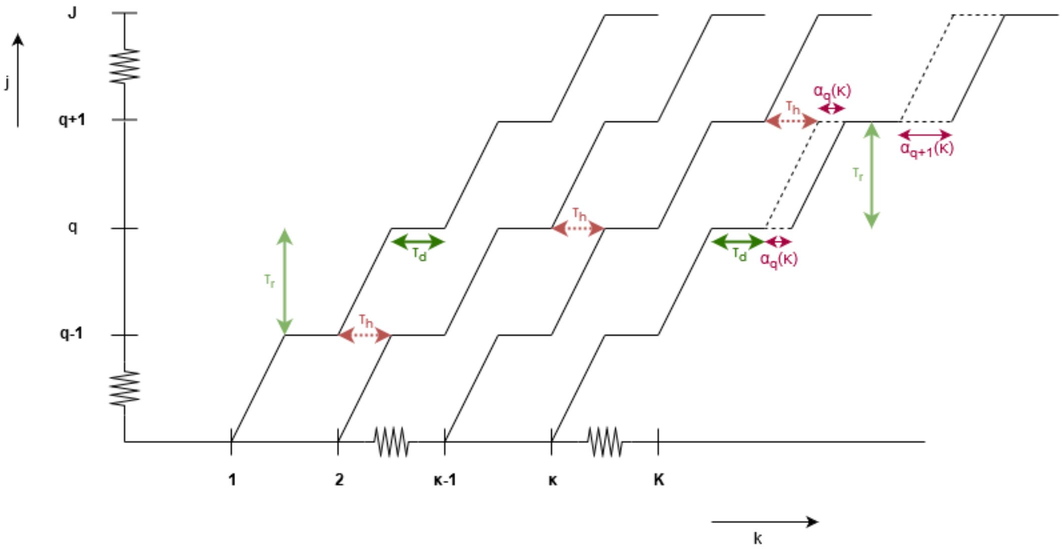


Figure 4-1: Example of the schedule an urban train line, where all headway-, dwell- and running times are constant and where trains $\{1, ..., \kappa - 1\}$ run without delays. Train κ is the first train with delay from station q onward.

Imagine the situation in 4-1, where all trains $k \in \{1, ..., K\}$ run from station 1 towards station J. Here, J can be taken to be a large integer value. We assume that all headway- and running times are constant. This means that we assume that the travel time between two stations is always the same, and also that the amount of arriving passengers per second λ is equal at every station j. Because of this, the headway times between trains will always stay the same. Furthermore, all dwell times are constant up until $k = \kappa$.

The goal is to observe if a sudden increase of dwell time at a station q will introduce an insuperable delay for train κ in the long run. If this can be shown, there is a good reason to introduce control and make sure that such a situation does not occur. Therefore, let us introduce the difference in time between (the arrival times of) two trains at station j as:

$$\Delta_j(k) = x_j(k) - x_j(k-1) \tag{4-3}$$

Consequently, $\Delta_J(k) = x_J(k) - x_J(k-1)$ and $\Delta_q(k) = x_q(k) - x_q(k-1)$. To see if such a delay becomes insuperable, the limit of the difference can be taken for train κ between the differences at the last station of the line ($\Delta_J(\kappa)$) and the first station where the delay started ($\Delta_q(\kappa)$) such that:

$$\lim_{J\to\infty} (\Delta_J(\kappa) - \Delta_q(\kappa)) = \infty \tag{4-4}$$

If this condition is true, control is necessary. This brings us to the following Theorem:

Theorem 3 (Divergence of $\lim_{J\to\infty}(\Delta_J(\kappa) - \Delta_q)(\kappa))$)
Let $\Delta_J(\kappa) - \Delta_q(\kappa)$ be the difference in headway between station J and station q, for train κ. If there is a disturbance $\alpha_q(\kappa)$ for train κ at station q, then $\lim_{J\to\infty}(\Delta_J(\kappa) - \Delta_q(\kappa)) = \infty$.

Underneath, a proof will follow to show that this limit occurs. The **Proof** will be done in conventional algebra, since working with minus-signs is not convenient in max-plus algebra. But since max-plus algebra always takes the maximum arrival and departure times for any train k, we can assume that the results in conventional algebra will also hold in max-plus algebra, since ∞ is always the maximum of two or more values if one of them is also ∞. That means, that if ∞ is reached in conventional algebra, it will also be reached in max-plus algebra. However, by doing this proof in conventional algebra, it can not be shown when the limit in (4-4) will be reached (i.e. if it is already 'before' $J \to \infty$) in max-plus algebra.

Proof *Since the delay occurs for train κ at station q, only train κ and $\kappa-1$ will be considered because every train up until then had a constant undisturbed schedule, so the operation in (4-3) will result into 0. If we consider $\Delta_j(\kappa) = x_j(\kappa) - x_j(\kappa-1)$, it can be written that:*

$$\begin{aligned} \Delta_{q+1}(\kappa) &= x_{q+1}(\kappa) - x_{q+1}(\kappa-1) \\ \Delta_q(\kappa) &= x_q(\kappa) - x_q(\kappa-1) \end{aligned} \tag{4-5}$$

and that:

$$\begin{aligned} \Delta_{q+1}(\kappa) - \Delta_q(\kappa) &= x_{q+1}(\kappa) - x_{q+1}(\kappa-1) - x_q(\kappa) + x_q(\kappa-1) \\ &= \underbrace{x_{q+1}(\kappa) - x_q(\kappa)}_{\tau_{\mathrm{d},q}(\kappa)+\tau_{\mathrm{r}}} - \underbrace{\big(x_{q+1}(\kappa-1) - x_q(\kappa-1)\big)}_{\tau_{\mathrm{d},q}(\kappa-1)+\tau_{\mathrm{r}}} \\ &= \underbrace{\tau_{\mathrm{d},q}(\kappa)}_{\tau_{\mathrm{d}}+\alpha_q(\kappa)} - \underbrace{\tau_{\mathrm{d},q}(\kappa-1)}_{\tau_{\mathrm{d}}} \\ &= \alpha_q(\kappa) \end{aligned} \tag{4-6}$$

where $\alpha_q(\kappa)$ is the extra dwell time caused by an increased amount of passengers at station q for train κ (see Figure 4-1). Similarly, we find that:

$$\begin{aligned}\Delta_{q+2}(\kappa) - \Delta_{q+1}(\kappa) &= \alpha_{q+1}(\kappa)\\ \Delta_{q+3}(\kappa) - \Delta_{q+2}(\kappa) &= \alpha_{q+2}(\kappa)\\ &\vdots\\ \Delta_{J}(\kappa) - \Delta_{J-1}(\kappa) &= \alpha_{J-1}(\kappa)\end{aligned}$$

Then, by summation, it is easy to see that:

$$\begin{array}{r}\Delta_{q+1}(\kappa) - \Delta_{q}(\kappa) = \alpha_{q}(\kappa)\\ \Delta_{q+2}(\kappa) - \Delta_{q+1}(\kappa) = \alpha_{q+1}(\kappa)\\ \Delta_{q+3}(\kappa) - \Delta_{q+2}(\kappa) = \alpha_{q+2}(\kappa)\\ \Delta_{J}(\kappa) - \Delta_{J-1}(\kappa) = \alpha_{J-1}(\kappa)\\ \hline \Delta_{J}(\kappa) - \Delta_{q}(\kappa) = \alpha_{q}(\kappa) + \cdots + \alpha_{J-1}(\kappa)\end{array} + \tag{4-7}$$

For the second part of the proof, it is important to rewrite all α's that occur at stations after station q. Consider $\tau_{h,q}(\kappa)$, the difference between the arrival time of train κ and the departure time of train $\kappa - 1$ at station q, and $\tau_{h,q+1}(\kappa)$:

$$\begin{aligned}\tau_{\mathrm{h},q}(\kappa) &= x_q(\kappa) - (x_q(\kappa-1) + \tau_{\mathrm{d}})\\ \tau_{\mathrm{h},q+1}(\kappa) &= x_{q+1}(\kappa) - (x_{q+1}(\kappa-1) + \tau_{\mathrm{d}})\end{aligned}$$

The dwell time at a station j is defined as the amount of passengers per second multiplied with the headway time between two trains, i.e. $\tau_{d,j}(k) = \lambda \cdot \tau_{h,j}(k)$. Furthermore, by writing out the expression for $\tau_{h,q+1}(\kappa)$, it can be found that:

$$\begin{aligned}\tau_{\mathrm{h},q+1}(\kappa) &= x_{q+1}(\kappa) - (x_{q+1}(\kappa-1) + \tau_{\mathrm{d}})\\ &= \underbrace{x_{q+1}(\kappa)}_{x_q(\kappa)+\tau_{\mathrm{d}}+\alpha_q(\kappa)+\tau_{\mathrm{r}}} - \underbrace{x_{q+1}(\kappa-1)}_{x_q(\kappa-1)+\tau_{\mathrm{d}}+\tau_{\mathrm{r}}} - \tau_{\mathrm{d}}\\ &= \underbrace{x_q(\kappa) - x_q(\kappa-1) - \tau_{\mathrm{d}}}_{\tau_{\mathrm{h},q}(\kappa)} + \alpha_q(\kappa)\\ \tau_{\mathrm{h},q+1}(\kappa) &= \tau_{\mathrm{h},q}(\kappa) + \alpha_q(\kappa)\end{aligned}$$

If we multiply the expression above with λ, we can substitute that $\tau_{d,j}(k) = \lambda \cdot \tau_{h,j}(k)$ for every j and k, and since every dwell time $\tau_{d,j}(\kappa)$ for train κ after station $q-1$ is a combination of the constant dwell time τ_d and some additional positive $\alpha_j(\kappa)$, it can be found that:

$$\begin{aligned} \tau_{\mathrm{h},q+1}(\kappa) &= \tau_{\mathrm{h},q}(\kappa) + \alpha_q(\kappa) \\ \underbrace{\lambda \cdot \tau_{\mathrm{h},q+1}(\kappa)}_{\tau_\mathrm{d}+\alpha_{q+1}(\kappa)} &= \underbrace{\lambda \cdot \tau_{\mathrm{h},q}(\kappa)}_{\tau_\mathrm{d}+\alpha_q(\kappa)} + \lambda \cdot \alpha_q(\kappa) \\ \alpha_{q+1}(\kappa) &= \big(1+\lambda\big) \cdot \alpha_q(\kappa) \end{aligned}$$

The last equation holds for all $j \in \{q, ..., J-1\}$, so:

$$\begin{aligned} \alpha_{q+2}(\kappa) &= \big(1+\lambda\big) \cdot \alpha_{q+1}(\kappa) \\ &= \big(1+\lambda\big)^2 \cdot \alpha_q(\kappa) \\ &\vdots \\ \alpha_{J-1}(\kappa) &= \big(1+\lambda\big)^{J-1} \cdot \alpha_q(\kappa) \end{aligned}$$

Finally, by substituting these new relations back into (4-7), it can be found that:

$$\begin{aligned} \Delta_J(\kappa) - \Delta_q(\kappa) &= \alpha_q(\kappa) + \cdots + \alpha_{J-1}(\kappa) \\ &= \alpha_q(\kappa) + \cdots + \big(1+\lambda\big)^{J-1} \cdot \alpha_q(\kappa) \\ &= \Big(\sum_{a=0}^{J-1} \big(1+\lambda\big)^a\Big) \cdot \alpha_q(\kappa) \end{aligned}$$

and since clearly, the limit $\lim_{J\to\infty}\Big(\sum_{a=0}^{J-1}\big(1+\lambda\big)^a\Big) = \infty$, it is proven that:

$$\lim_{J\to\infty}\left(\Delta_J(\kappa) - \Delta_q(\kappa)\right) = \lim_{J\to\infty}\left(\Big(\sum_{a=0}^{J-1}\big(1+\lambda\big)^a\Big) \cdot \alpha_q(\kappa)\right) = \infty \qquad \blacksquare$$

This proof shows that it is in fact possible for the system to become unstable when there is either no bound on the amount of dwell time, or when train $\kappa-1$ is not delayed accordingly with train κ. Since it is only possible to delay the network in max-plus algebra, previous trains $k \in \{1, ..., \kappa-1\}$ have to be delayed to avoid instability of the urban train line. This is why it is proposed in this thesis to allow these delays in order to avoid the proportional increase of the delay for train κ (see assumption 5 in section sec:maxpluseq). Such a strategy corresponds with the same reasoning behind the avoidance of the so-called accordion effect in traffic.

4-2 Converting the max-plus system to linear inequalities

The multiple cycle max-plus system as could be observed in (3-43) consists of all the constraints to simulate the trajectories of $M-1$ trains in once, by using the iterative method.

However, if we want to find the ideal control law $\breve{u}(k)$ for every set of $M-1$ trains, optimization according to some kind of objective function will be necessary. In order to simulate the max-plus model following some kind of objective function, it is more convenient to transform the system back to a set of linear inequalities. The reason for this is that a set of linear inequalities lents itself perfectly for Linear Programming or Quadratic Programming, as long as the objective function is linear or quadratic respectively.

Such a strategy is inspired and motivated by following the theory on the so-called extended linear complementarity problem (ELCP) in De Schutter and van den Boom (2001), where a max-plus equality system can be rewritten to a set of linear inequalities as long as one of them actually holds as an equality. The reason for rewriting the constraints is that solving an optimization problem with a set of linear inequality constraints is much easier to solve numerically than it would have been with equality constraints. Chapter 4 of (Kersbergen et al., 2016) will be used as the main reference for this procedure. If the ideal schedule $\breve{x}_{\text{opt}}(k)$ is found with optimization, then that schedule can be used as the control law $\breve{u}(k)$ (section 4-4).

Now, let us refine some definitions. If M is the total amount of train cycles that will be considered at once, and $m = \{1, 2, ..., M\}$ is the cycle counter, then we can define $\breve{x}(k) = \begin{bmatrix} x^T(k) & x^T(k+1) & ... & x^T(k+M-1) \end{bmatrix}^T$ such that $\breve{x}_m(k) = x(k+m-1)$. Furthermore, $\breve{x}_m(k)$ can be partitioned into $2J$ scalars $\breve{x}_{m,k}(k)$, such that $\breve{x}_m(k) = \begin{bmatrix} x_1^T(k+m-1) & x_2^T(k+m-1) & ... & x_{2J}^T(k+m-1) \end{bmatrix}^T$, where J is the total amount of train stations on a line. To show the line of thought, the conversion of the max-plus equations to a set of linear inequality constraint will be done with the number of stations chosen to be $J = 2$.

$\hat{A}_0(p)$, $\tilde{A}_0$, A_1 and B to linear inequalities

$\hat{A}_0(p)$) To begin with, we have to recall that:

$$\hat{A}_0(p) = \left(\begin{array}{cccc|c} \multicolumn{4}{c|}{\mathcal{E}} & \mathcal{E} \\ \hline \tau_{\text{d,min},1}(k) & \epsilon & \cdots & \epsilon & \\ \epsilon & \tau_{\text{d,min},2}(k) & \cdots & \epsilon & \mathcal{E} \\ \vdots & \vdots & \ddots & \vdots & \\ \epsilon & \epsilon & \cdots & \tau_{\text{d,min},J}(k) & \end{array} \right) \tag{4-8}$$

and furthermore that $x(k) = \hat{A}_0(p) \otimes x(k)$. Since $J = 2$, we know that our matrix $\hat{A}_0(p)$ is a $2J = 4$ by $2J = 4$ matrix, and thus there should be 4 inequalities. But since we observe that only the botom-left sub-matrix matters, we only need J inequalities. Namely:

$$\begin{aligned} x_{J+1}(k) &\geq \tau_{\text{d,min},1}(k) + x_1(k) \\ x_{J+2}(k) &\geq \tau_{\text{d,min},2}(k) + x_2(k) \end{aligned} \tag{4-9}$$

From (3-37), we know that every $\tau_{\text{d,min},j}(k)$ is a linear combination of $x(k)$ and $x(k-1)$. If we define the matrix product $\Psi = \Phi \cdot \lambda$ and partition the matrices Φ and Ψ as

$$\Phi = \begin{pmatrix} \Phi_{1,1} & \Phi_{1,2} \\ \Phi_{2,1} & \Phi_{2,2} \end{pmatrix} \qquad \Psi = \begin{pmatrix} \Phi_{1,1}\lambda_1 & \Phi_{1,2}\lambda_2 \\ \Phi_{2,1}\lambda_1 & \Phi_{2,2}\lambda_2 \end{pmatrix}, \tag{4-10}$$

we find that every $\tau_{\text{d,min},j}(k)$ can be written as:

$$\tau_{\text{d,min},j}(k) = \alpha_1 + \sum_{i=1}^{J} \Psi_{j,i} \cdot (x_i(k) - x_{J+i}(k-1)) \tag{4-11}$$

$\tilde{A}_0$) For $\tilde{A}_0$, we have that:

$$\tilde{A}_0 = \left(\begin{array}{cccc|ccccc} & & & & \epsilon & \epsilon & \cdots & \epsilon & \epsilon \\ & & & & \tau_{\text{r,min},1} & \epsilon & \cdots & \epsilon & \epsilon \\ & \mathcal{E} & & & \epsilon & \tau_{\text{r,min},2} & \cdots & \epsilon & \epsilon \\ & & & & \vdots & \vdots & \ddots & \vdots & \vdots \\ & & & & \epsilon & \epsilon & \cdots & \tau_{\text{r,min}\,.J-1} & \epsilon \\ \hline \tilde{\tau}_{\text{min}} & \epsilon & \cdots & \epsilon & & & & & \\ \epsilon & \tilde{\tau}_{\text{min}} & \cdots & \epsilon & & & \mathcal{E} & & \\ \vdots & \vdots & \ddots & \vdots & & & & & \\ \epsilon & \epsilon & \cdots & \tilde{\tau}_{\text{min}} & & & & & \end{array}\right) \tag{4-12}$$

and that $x(k) = \tilde{A}_0 \otimes x(k)$. Because of $J = 2$, we have 2 inequalities from the bottom-left sub-matrix and 2 inequalities from the top-right sub-matrix. These are defined as following:

$$\begin{aligned} x_1(k) &\geq \tau_{\text{r,min},0} + x_J(k) \\ x_2(k) &\geq \tau_{\text{r,min},1} + x_{J+1}(k) \\ x_{J+1}(k) &\geq \tilde{\tau}_{\text{min}} + x_1(k) \\ x_{J+2}(k) &\geq \tilde{\tau}_{\text{min}} + x_2(k) \end{aligned} \tag{4-13}$$

A_1) The third matrix is A_1, which could be written as:

$$A_1 = \left(\begin{array}{c|cccc} & \tau_{\text{h,min}} & \epsilon & \cdots & \epsilon \\ & \epsilon & \tau_{\text{h,min}} & \cdots & \epsilon \\ \mathcal{E} & \vdots & \vdots & \ddots & \vdots \\ & \epsilon & \epsilon & \cdots & \tau_{\text{h,min}} \\ \hline \mathcal{E} & & & \mathcal{E} & \end{array}\right) \tag{4-14}$$

This matrix connects previous and current train cycles with each other, as $x(k) = A_1 \otimes x(k-1)$. Again, we find that there are only J constraints, and those are:

$$\begin{aligned} x_1(k) &\geq \tau_{\text{h,min}} + x_{J+1}(k-1) \\ x_2(k) &\geq \tau_{\text{h,min}} + x_{J+2}(k-1) \end{aligned} \tag{4-15}$$

B**)** At last, we can define the matrix B simply such that $x(k) \geq d(k)$. So we just write that:

$$\begin{aligned} x_1(k) &\geq d_1(k) \\ x_2(k) &\geq d_2(k) \\ x_3(k) &\geq d_3(k) \\ x_4(k) &\geq d_4(k) \end{aligned} \tag{4-16}$$

and consequently, $B = E$.

Collecting all the constraints

With all constraints known, we can start writing the problem in the form $F_0 \cdot \breve{x}(k) + F_1 \cdot x(k-1) \leq g_0 + G_d \cdot \breve{d}(k)$, were the matrices F_0, F_1, g_0 and G_d are defined in conventional algebra. First, we collect all the constraints and rewrite them as $x_j(k+m-1) = \breve{x}_{m,j}(k)$:

$$\begin{aligned} \breve{x}_{m,1}(k) &\geq \breve{x}_{m,2}(k) + \tau_{\mathrm{r,min,0}} \\ &\geq \breve{x}_{m-1,3}(k) + \tau_{\mathrm{h,min}} \\ &\geq d_{m,1}(k) \\ \breve{x}_{m,2}(k) &\geq \breve{x}_{m,3}(k) + \tau_{\mathrm{r,min,1}} \\ &\geq \breve{x}_{m-1,4}(k) + \tau_{\mathrm{h,min}} \\ &\geq d_{m,2}(k) \\ \breve{x}_{m,3}(k) &\geq \breve{x}_{m,1}(k) + \tilde{\tau}_{\mathrm{min}} \\ &\geq \breve{x}_{m,1}(k) + \alpha_1 + \Psi_{1,1}\breve{x}_{m,1}(k) + \Psi_{1,2}\breve{x}_{m,2}(k) - \Psi_{1,1}\breve{x}_{m-1,3}(k) - \Psi_{1,2}\breve{x}_{m-1,4}(k) \\ &\geq d_{m,3}(k) \\ \breve{x}_{m,4}(k) &\geq \breve{x}_{m,2}(k) + \tilde{\tau}_{\mathrm{min}} \\ &\geq \breve{x}_{m,2}(k) + \alpha_1 + \Psi_{2,1}\breve{x}_{m,1}(k) + \Psi_{2,2}\breve{x}_{m,2}(k) - \Psi_{2,1}\breve{x}_{m-1,3}(k) - \Psi_{2,2}\breve{x}_{m-1,4}(k) \\ &\geq d_{m,4}(k) \end{aligned} \tag{4-17}$$

Now we can move all the variables in $\breve{x}_{m,j}(k)$ to the left side and all the control variables and constants to the right side. We find that:

$$
\begin{aligned}
\breve{x}_{m,2}(k) - \breve{x}_{m,1}(k) &\leq -\tau_{\mathrm{r,min},0} \\
\breve{x}_{m-1,3}(k) - \breve{x}_{m,1}(k) &\leq -\tau_{\mathrm{h,min}} \\
-\breve{x}_{m,1}(k) &\leq -d_{m,1}(k) \\
\breve{x}_{m,3}(k) - \breve{x}_{m,2}(k) &\leq -\tau_{\mathrm{r,min},1} \\
\breve{x}_{m-1,4}(k) - \breve{x}_{m,2}(k) &\leq -\tau_{\mathrm{h,min}} \\
-\breve{x}_{m,2}(k) &\leq -d_{m,2}(k) \\
\breve{x}_{m,1}(k) - \breve{x}_{m,3}(k) &\leq -\tilde{\tau}_{\mathrm{min}} \\
\breve{x}_{m,1}(k) - \breve{x}_{m,3}(k) + \Psi_{1,1}\breve{x}_{m,1}(k) + \Psi_{1,2}\breve{x}_{m,2}(k) - \Psi_{1,1}\breve{x}_{m-1,3}(k) - \Psi_{1,2}\breve{x}_{m-1,4}(k) &\leq -\alpha_1 \\
-\breve{x}_{m,3}(k) &\leq -d_{m,3}(k) \\
\breve{x}_{m,2}(k) - \breve{x}_{m,4}(k) &\leq -\tilde{\tau}_{\mathrm{min}} \\
\breve{x}_{m,2}(k) - \breve{x}_{m,4}(k) + \Psi_{2,1}\breve{x}_{m,1}(k) + \Psi_{2,2}\breve{x}_{m,2}(k) - \Psi_{2,1}\breve{x}_{m-1,3}(k) - \Psi_{2,2}\breve{x}_{m-1,4}(k) &\leq -\alpha_1 \\
-\breve{x}_{m,4}(k) &\leq -d_{m,4}(k)
\end{aligned}
\tag{4-18}
$$

and with the vector $\breve{x}_m(k)$ being defined earlier, and a similarly defined $\breve{d}_m(k)$, we see that:

$$
\begin{aligned}
&\underbrace{\begin{pmatrix}
-1 & 1 & 0 & 0 \\
-1 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 \\
0 & -1 & 1 & 0 \\
0 & -1 & 0 & 0 \\
0 & -1 & 0 & 0 \\
1 & 0 & -1 & 0 \\
1+\Psi_{1,1} & \Psi_{1,2} & -1 & 0 \\
0 & 0 & -1 & 0 \\
0 & 1 & 0 & -1 \\
\Psi_{2,1} & 1+\Psi_{2,2} & 0 & -1 \\
0 & 0 & 0 & -1
\end{pmatrix}}_{F_{0,m}} \cdot \begin{pmatrix} \breve{x}_{m,1}(k) \\ \breve{x}_{m,2}(k) \\ \breve{x}_{m,3}(k) \\ \breve{x}_{m,4}(k) \end{pmatrix} + \underbrace{\begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & -\Psi_{1,1} & -\Psi_{1,2} \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & -\Psi_{2,1} & -\Psi_{2,2} \\
0 & 0 & 0 & 0
\end{pmatrix}}_{F_{1,m}} \cdot \begin{pmatrix} \breve{x}_{m-1,1}(k) \\ \breve{x}_{m-1,2}(k) \\ \breve{x}_{m-1,3}(k) \\ \breve{x}_{m-1,4}(k) \end{pmatrix} \\
&\leq \underbrace{\begin{pmatrix}
-\tau_{\mathrm{r,min},0} \\
-\tau_{\mathrm{h,min}} \\
0 \\
-\tau_{\mathrm{r,min},1} \\
-\tau_{\mathrm{h,min}} \\
0 \\
-\tilde{\tau}_{\mathrm{min}} \\
-\alpha_1 \\
0 \\
-\tilde{\tau}_{\mathrm{min}} \\
-\alpha_1 \\
0
\end{pmatrix}}_{g_{0,m}} + \underbrace{\begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1
\end{pmatrix}}_{G_{d,m}} \cdot \begin{pmatrix} \breve{d}_{m,1}(k) \\ \breve{d}_{m,2}(k) \\ \breve{d}_{m,3}(k) \\ \breve{d}_{m,4}(k) \end{pmatrix}
\end{aligned}
\tag{4-19}
$$

which can be summarized in short as:

$$F_{0,m} \cdot \breve{x}_m(k) + F_{1,m} \cdot \breve{x}_{m-1}(k) \leq g_{0,m} + G_{d,m} \cdot \breve{d}_m(k) \tag{4-20}$$

We can make the last step by writing out (4-20) for $m = 1, 2, ..., M-1$, to find that:

$$\begin{aligned}
&F_{0,1} \cdot \breve{x}_1(k) + F_{1,1} \cdot \breve{x}_0(k) \leq g_{0,1} + G_{d,1} \cdot \breve{u}_1(k) \\
&\begin{pmatrix} F_{0,1} & 0 & \cdots & 0 \end{pmatrix} \cdot \breve{x}(k) + F_{1,1} \cdot \underbrace{\breve{x}_0(k)}_{x(k-1)} \leq g_{0,1} + \begin{pmatrix} G_{d,1} & 0 & \cdots & 0 \end{pmatrix} \cdot \breve{u}(k) \\
\\
&F_{0,2} \cdot \breve{x}_2(k) + F_{1,2} \cdot \breve{x}_1(k) \leq g_{0,2} + G_{d,2} \cdot \breve{u}_2(k) \\
&\begin{pmatrix} F_{1,2} & F_{0,2} & \cdots & 0 \end{pmatrix} \cdot \breve{x}(k) \leq g_{0,2} + \begin{pmatrix} 0 & G_{d,2} & \cdots & 0 \end{pmatrix} \cdot \breve{u}(k) \\
&\qquad \vdots \\
&F_{0,M-1} \cdot \breve{x}_{M-1}(k) + F_{1,M-1} \cdot \breve{x}_{M-2}(k) \leq g_{0,M-1} + G_{d,M-1} \cdot \breve{u}_{M-1}(k) \\
&\begin{pmatrix} 0 & \cdots & F_{1,M-1} & F_{0,M-1} \end{pmatrix} \cdot \breve{x}(k) \leq g_{0,M-1} + \begin{pmatrix} 0 & \cdots & 0 & G_{d,M-1} \end{pmatrix} \cdot \breve{d}(k)
\end{aligned}$$

eventually resulting into:

$$\begin{aligned}
&\begin{pmatrix} F_{0,1} & 0 & \cdots & 0 \\ F_{1,2} & F_{0,2} & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & F_{1,M-1} & F_{0,M-1} \end{pmatrix} \cdot \breve{x}(k) + \begin{pmatrix} F_{1,1} \\ 0 \\ \vdots \\ 0 \end{pmatrix} \cdot x(k-1) \\
&\qquad \leq \begin{pmatrix} g_{0,1} \\ g_{0,2} \\ \vdots \\ g_{0,M-1} \end{pmatrix} + \begin{pmatrix} G_{d,1} & 0 & \cdots & 0 \\ 0 & G_{d,2} & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & G_{d,M-1} \end{pmatrix} \cdot \breve{d}(k)
\end{aligned} \tag{4-21}$$

and thus that $F_0 \cdot \breve{x}(k) + F_1 \cdot x(k-1) \leq g_0 + G_d \cdot \breve{d}(k)$. It can be observed that the matrices F_0, F_1, g_0 and G_d contain multiple matrices with different indices ($F_{0,1}$, $F_{1,2}$, etc.). This is purely distinguished for optional future work: in the case of different Ψ values over time, the matrices will differ from each other, but when λ and Φ do not change over time, then these matrices are all equal to each other (i.e. $F_{0,1} = F_{0,2} = ... = F_{0,M-1}$).

This procedure remains similarly structured for more stations (i.e. $J > 2$) and more trains M. The next step is define an objective function J_{obj} in section 4-3.

4-3 Objective function

The objective function basically describes the goal of the optimization problem. Operators of urban rail networks probably prefer economically profitable scheduling, for example by minimizing energy consumption or maximizing profit. Governments and municipalities probably prefer the best for their residents by minimizing the total travel time or total waiting time for passengers. Environmentalists do also prefer minimal energy consumption while passengers themselves would prefer maximal capacity, minimal travel time and minimal transfers. That is why the 'optimal' schedule is a very tricky definition; it is very hard to satisfy all the stakeholders at once.

Many studies focus on minimizing energy consumption. Take for instance Li and Lo (2014), whom focus on scheduling while minimizing the total energy consumption while also taking into account regenerative energy. A genetic algorithm was used to solve the (non-convex) optimization problem. In (Yang et al., 2017), not only net energy consumption is minimized, but also the total travel time while also modelling dwell time uncertainty. This results into an optimum between travel time and energy consumption, but also in the amount of vehicles that have to be used. Again, the optimization is non-convex and is solved by a genetic algorithm. In (Wang, Ning, et al., 2015) and (Wang, Tang, et al., 2015), two papers which are being used more often as a source for this study, optimization is being done with respect to minimizing both energy consumption and total travel time. The problem formulation is a little less complex due to not taking into account regenerative energy, but it is still a non-linear programming problem which requires non-linear solvers, mainly due to a min-function in the constraints. Niu et al. (2015) focus on minimizing the total waiting times of passengers at stations, both with and without a predetermined skip-stop pattern for trains. Their algorithm can be used for both real-time scheduling (computation time: 2.7 minutes) and medium-term planning (computation time: 10.7 minutes), by using a non-linear mixed integer programming model.

As can be seen, all of the above-mentioned studies contain non-linear programming problems. It does not necessarily have to be a problem, but in order to compute on-line scheduling problems it is of high importance that the total computation time of the optimization is smaller than the interval between two subsequent optimizations. To achieve this, linear of quadratic programming problems are the most ideal. In the next subsection, two different objective functions will be shown that fulfil this requirement: minimizing total travel time and minimizing total passenger travel time.

4-3-1 Minimizing total travel time

Minimizing total travel time is basically the same as minimizing every arrival or departure time of the total schedule; the absolute limits of the constraints will be pushed all the way until the railway network can not be operated any quicker. This means that we just have to minimize every value such that the total travel time $x_{\text{tot}}(k)$ is the lowest, resulting in the following objective function:

$$\min_{x(k)} \quad x_{\text{tot}}(k) \tag{4-22}$$

where $x_{\text{tot}}(k)$ can be interpret as:

$$x_{\text{tot}}(k) = \begin{pmatrix} 1 & 1 & \cdots & 1 \end{pmatrix} \cdot \breve{x}(k) = \bar{\mathbf{1}} \cdot \breve{x}(k) \tag{4-23}$$

This optimization problem is linear, since both the constraints as well as the objective function are linear. Therefore, it is the easiest and simplest optimization problem. However, it contains some important information. For instance, the modelled railway network can never run fast than the solution $x_{\text{sol}}(k)$ of this optimization problem. Therefore, it can be taken as a lower bound for every other optimization problem.

4-3-2 Minimizing total passenger travel time

To formulate the real-time scheduling problem with respect to minimizing the total travel time of all passengers, a specific objective function has to be defined. The total travel time of passengers is the sum of the total waiting time and total in-vehicle time for every passenger.

Passenger waiting time

The passenger waiting time $t_j^{\text{wait}}(k)$ at station j for train k is the waiting time of passengers that were left behind by train $k-1$ and the passengers that arrived randomly after worth:

$$t_j^{\text{wait}}(k) = w_j(k-1) \cdot (d_j(k) - d_j(k-1)) + \frac{1}{2}\lambda_j \cdot (d_j(k) - d_j(k-1))^2 \tag{4-24}$$

Since it is assumed that no passengers are left behind by train $k-1$ and no passengers arrive during the dwell time, $w_j^{\text{remain}}(k-1) = 0$ and $d_j(k)$ can be replaced with $a_j(k)$:

$$t_j^{\text{wait}}(k) = \frac{1}{2}\lambda_j \cdot (a_j(k) - d_j(k-1))^2 \tag{4-25}$$

If we write this equation out in terms of the vector $x(k)$, we find that $a_j(k) = x_j(k)$ and $d_j(k-1) = x_{J+j}(k-1)$.

$$\begin{aligned} t_j^{\text{wait}}(k) &= \frac{1}{2}\lambda_j \cdot (x_j(k) - x_{J+j}(k-1))^2 \\ &= \frac{1}{2}\lambda_j \cdot (x_j(k) - x_{J+j}(k-1))^T (x_j(k) - x_{J+j}(k-1)) \end{aligned} \tag{4-26}$$

It can be found that for $j = 1, ..., J$:

$$
\begin{aligned}
t_1^{\text{wait}}(k) &= \frac{1}{2}\lambda_1 \cdot (x_1(k) - x_{J+1}(k-1))^T (x_1(k) - x_{J+1}(k-1)) \\
&= \frac{1}{2}\lambda_1 \cdot \left(x^T(k) \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} - x^T(k-1) \cdot \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} \right) \left(\begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}^T \cdot x(k) - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}^T \cdot x(k-1) \right) \\
&= x^T(k) \cdot \begin{pmatrix} \frac{1}{2}\lambda_1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 \end{pmatrix} \cdot x(k) + x^T(k-1) \cdot \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \frac{1}{2}\lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 \end{pmatrix} \cdot x(k-1) \\
&\quad - x^T(k-1) \cdot \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \lambda_1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 \end{pmatrix} \cdot x(k) \\
&\ \ \vdots \\
t_J^{\text{wait}}(k) &= \frac{1}{2}\lambda_J \cdot (x_J(k) - x_{2J}(k-1))^T (x_J(k) - x_{2J}(k-1)) \\
&= x^T(k) \cdot \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & \frac{1}{2}\lambda_J & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 \end{pmatrix} \cdot x(k) + x^T(k-1) \cdot \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & \frac{1}{2}\lambda_J \end{pmatrix} \cdot x(k-1) \\
&\quad - x^T(k-1) \cdot \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_J & 0 & \cdots & 0 \end{pmatrix} \cdot x(k)
\end{aligned}
\tag{4-27}
$$

From these relations, we can construct the term $t^{\text{wait}}(k)$ which is the sum of all the separate $t_j^{\text{wait}}(k)$:

$$
\begin{aligned}
t^{\text{wait}}(k) &= \sum_{j=1}^{J} t_j^{\text{wait}}(k) \\
&= x^T(k) \cdot \begin{pmatrix} \frac{1}{2}\lambda_1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & \frac{1}{2}\lambda_J & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 \end{pmatrix} \cdot x(k) + x^T(k-1) \cdot \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \frac{1}{2}\lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & \frac{1}{2}\lambda_J \end{pmatrix} \cdot x(k-1) \\
&\quad - x^T(k-1) \cdot \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \lambda_1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_J & 0 & \cdots & 0 \end{pmatrix} \cdot x(k) \\
&= x^T(k) \cdot \underbrace{\begin{pmatrix} \frac{1}{2}\boldsymbol{\lambda} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}}_{\boldsymbol{\lambda_0}} \cdot x(k) + x^T(k-1) \cdot \underbrace{\begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{2}\boldsymbol{\lambda} \end{pmatrix}}_{\boldsymbol{\lambda_1}} \cdot x(k-1) - x^T(k-1) \cdot \underbrace{\begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \boldsymbol{\lambda} & \mathbf{0} \end{pmatrix}}_{\boldsymbol{\lambda_2}} \cdot x(k)
\end{aligned}
\tag{4-28}
$$

Passenger in-vehicle time

The in-vehicle time $t_j^{\text{in-vehicle}}(k)$ for all passengers is sum of the dwell time $\tau_{\text{d},j}(k)$ for the passengers who did not get out of the train at the current stop j and the running time $r_j(k)$ for all passengers:

$$
\begin{aligned}
t_j^{\text{in-vehicle}}(k) &= n_{j-1}(k) \cdot r_{j-1}(k) + n_{j-1}(k) \cdot (1-\rho_j) \cdot \tau_{\text{d},j}(k) \\
&= n_{j-1}(k) \cdot (r_{j-1}(k) + (1-\rho_j) \cdot \tau_{\text{d},j}(k)) \\
&= n_{j-1}(k) \cdot (x_j(k) - x_{j+J-1}(k) + \bar{\rho}_j \cdot \tau_{\text{d},j}(k))
\end{aligned}
\tag{4-29}
$$

because $r_{j-1}(k)$ can be written as the time between the departure time $d_{j-1}(k) = x_{j+J-1}(k)$ at $j-1$ and the arrival time $a_j(k) = x_j(k)$ at j. Furthermore, we write $\bar{\rho}_j = 1 - \rho_j$ for convenience. If we define the actual dwell time $\tau_{\text{d},j}(k)$ simply as the difference between the departure and arrival time at station j (i.e., as $\tau_{\text{d},j}(k) = d_j(k) - a_j(k) = x_{j+J}(k) - x_j(k)$), we can substitute that:

$$
t_j^{\text{in-vehicle}}(k) = n_{j-1}(k) \cdot (\rho_j \cdot x_j(k) - x_{j+J-1}(k) + \bar{\rho}_j \cdot x_{j+J}(k)) \tag{4-30}
$$

By definition, we know that $t_1^{\text{in-vehicle}}(k) = 0$, since there are no passengers in the train yet. By writing out (4-30) for $j = 2, 3, ..., J$, we find that:

$$\begin{aligned}
t_2^{\text{in-vehicle}}(k) &= n_1(k) \cdot (\rho_2 \cdot x_2(k) - x_{1+J}(k) + \bar{\rho}_2 \cdot x_{2+J}(k)) \\
t_3^{\text{in-vehicle}}(k) &= n_2(k) \cdot (\rho_3 \cdot x_3(k) - x_{2+J}(k) + \bar{\rho}_3 \cdot x_{3+J}(k)) \\
&\vdots \\
t_J^{\text{in-vehicle}}(k) &= n_{J-1}(k) \cdot (\rho_J \cdot x_J(k) - x_{2J-1}(k) + \bar{\rho}_J \cdot x_{2J}(k))
\end{aligned} \tag{4-31}$$

and thus that:

$$\underbrace{\begin{pmatrix} t_1^{\text{in-vehicle}}(k) \\ t_2^{\text{in-vehicle}}(k) \\ t_3^{\text{in-vehicle}}(k) \\ \vdots \\ t_J^{\text{in-vehicle}}(k) \end{pmatrix}}_{\boldsymbol{t}^{\text{in-vehicle}}(k)} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & n_1(k) & 0 & \cdots & 0 \\ 0 & 0 & n_2(k) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & n_{J-1}(k) \end{pmatrix} \cdot$$
$$\underbrace{\left(\begin{array}{ccccc|ccccc} 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \rho_2 & 0 & \cdots & 0 & -1 & \bar{\rho}_2 & 0 & \cdots & 0 \\ 0 & 0 & \rho_3 & \cdots & 0 & 0 & -1 & \bar{\rho}_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \rho_J & 0 & 0 & \cdots & -1 & \bar{\rho}_J \end{array}\right)}_{\mathrm{P}} \cdot \underbrace{\begin{pmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_{2J-1}(k) \\ x_{2J}(k) \end{pmatrix}}_{x(k)} \tag{4-32}$$

To find the total in-vehicle time $t^{\text{in-vehicle}}(k)$, we can take the sum of all the individual in-vehicle times per station, as following:

$$t^{\text{in-vehicle}}(k) = \sum_{j=1}^{J} t_j^{\text{in-vehicle}}(k) = \bar{\mathbf{1}} \cdot \boldsymbol{t}^{\text{in-vehicle}}(k) \tag{4-33}$$

where again $\bar{\mathbf{1}} \in \mathbb{R}^{1\times J}$ is a unity vector such that the sum-operator is eliminated. We find that this unity vector also eliminates the diagonal-operator in (4-32), such that:

$$t^{\text{in-vehicle}}(k) = \begin{pmatrix} 0 \\ n_1(k) \\ n_2(k) \\ \vdots \\ n_{J-1}(k) \end{pmatrix}^T \cdot \mathrm{P} \cdot x(k) \tag{4-34}$$

We would like to replace the first vector in (4-32) by an expression with $x(k)$ in it. In order to do so, the theory in section 3-4 can be used. Since it is assumed that there is no capacity constraint for the trains, and no passengers board during the dwell time, the following can be concluded:

$$n_j(k) = n_{j-1}(k) + \underbrace{n_j^{\text{board}}}_{w_j^{\text{wait}}} - \underbrace{n_j^{\text{leave}}}_{\rho_j \cdot n_{j-1}(k)} = w_j^{\text{wait}} + \underbrace{(1-\rho_j)}_{\bar{\rho}_j} \cdot n_{j-1}(k) \tag{4-35}$$

and therefore, we can write out that:

$$\begin{aligned} n_1(k) &= w_1^{\text{wait}} + \bar{\rho}_1 \cdot n_0(k) \\ &= w_1^{\text{wait}} \\ n_2(k) &= w_2^{\text{wait}} + \bar{\rho}_2 \cdot n_1(k) \\ &= w_2^{\text{wait}} + \bar{\rho}_2 \cdot w_1^{\text{wait}} \\ n_3(k) &= w_3^{\text{wait}} + \bar{\rho}_3 \cdot n_2(k) \\ &= w_3^{\text{wait}} + \bar{\rho}_3 \cdot w_2^{\text{wait}} + \bar{\rho}_3\bar{\rho}_2 \cdot w_1^{\text{wait}} \\ &\vdots \end{aligned} \tag{4-36}$$

which can be written in matrix-form as:

$$\begin{pmatrix} n_1(k) \\ n_2(k) \\ n_3(k) \\ \vdots \\ n_J(k) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \bar{\rho}_2 & 1 & 0 & \cdots & 0 \\ \bar{\rho}_3\bar{\rho}_2 & \bar{\rho}_3 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{\rho}_J \cdots \bar{\rho}_2 & \bar{\rho}_J \cdots \bar{\rho}_3 & \cdots & \cdots & 1 \end{pmatrix} \cdot \underbrace{\begin{pmatrix} w_1^{\text{wait}}(k) \\ w_2^{\text{wait}}(k) \\ w_3^{\text{wait}}(k) \\ \vdots \\ w_J^{\text{wait}}(k) \end{pmatrix}}_{\boldsymbol{w}^{\text{wait}}(k)} \tag{4-37}$$

Since we only need the first $J-1$ terms of $n_j(k)$, the expression in (4-38) is rewritten as:

$$\begin{pmatrix} 0 \\ n_1(k) \\ n_2(k) \\ \vdots \\ n_{J-1}(k) \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ \bar{\rho}_2 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \bar{\rho}_{J-1} \cdots \bar{\rho}_2 & \bar{\rho}_{J-1} \cdots \bar{\rho}_3 & \cdots & \cdots & 1 & 0 \end{pmatrix}}_{T_\rho} \cdot \boldsymbol{w}^{\text{wait}}(k) \tag{4-38}$$

and finally, by substituting (4-38) into (4-39), it can be found that:

$$t^{\text{in-vehicle}}(k) = \boldsymbol{w}^{\text{wait}}(k)^T \cdot T_\rho^T \cdot \mathrm{P} \cdot x(k) \tag{4-39}$$

Finally, by substituting the results from (3-36), the total in-vehicle travel time for train k can be defined in terms of $x(k)$:

$$\begin{aligned} t^{\text{in-vehicle}}(k) &= \left(\begin{bmatrix} \boldsymbol{\lambda} & \mathbf{0} \end{bmatrix} x(k) - \begin{bmatrix} \mathbf{0} & \boldsymbol{\lambda} \end{bmatrix} x(k-1) \right)^T \cdot T_\rho^T \cdot \mathrm{P} \cdot x(k) \\ &= x^T(k) \cdot \begin{bmatrix} \boldsymbol{\lambda} & \mathbf{0} \end{bmatrix}^T \cdot T_\rho^T \cdot \mathrm{P} \cdot x(k) - x^T(k-1) \cdot \begin{bmatrix} \mathbf{0} & \boldsymbol{\lambda} \end{bmatrix}^T \cdot T_\rho^T \cdot \mathrm{P} \cdot x(k) \end{aligned} \tag{4-40}$$

which can be written in the following form:

$$t^{\text{in-vehicle}}(k) = x^T(k) \cdot H \cdot x(k) + x^T(k-1) \cdot F \cdot x(k) \tag{4-41}$$

$$H = \begin{bmatrix} \boldsymbol{\lambda} & \mathbf{0} \end{bmatrix}^T \cdot T_\rho^T \cdot \mathrm{P} \tag{4-42}$$

$$F = - \begin{bmatrix} \mathbf{0} & \boldsymbol{\lambda} \end{bmatrix}^T \cdot T_\rho^T \cdot \mathrm{P} \tag{4-43}$$

$$\tag{4-44}$$

From (4-41), (4-42) and (4-43), it can be seen that the total travel time for passengers in train k is a quadratic optimization function, with some dependencies on past trains $k-1$. We can also observe that the matrix H is playing a major role in the ease of this optimization problem. If this matrix is symmetric and positive definite, then the problem is convex and easily solvable.

Total travel time

The total travel time of all passengers for train k is now defined as the weighted sum of the passenger waiting time for train k and the total in-vehicle time in train k:

$$\begin{aligned} t_{\text{total}}(k) &= \sum_{j=1}^{J-1} (\gamma_{\text{wait}} t_j^{\text{wait}}(k) + t_j^{\text{in-vehicle}}(k)) \\ &= \gamma_{\text{wait}} t^{\text{wait}}(k) + t^{\text{in-vehicle}}(k) \\ &= x^T(k) \cdot \underbrace{(H + \gamma_{\text{wait}} \boldsymbol{\lambda}_0)}_{H_*} \cdot x(k) + x^T(k-1) \cdot \underbrace{(F - \gamma_{\text{wait}} \boldsymbol{\lambda}_2)}_{F_*} \cdot x(k) + x^T(k-1) \cdot \underbrace{(\gamma_{\text{wait}} \boldsymbol{\lambda}_1)}_{C} \cdot x(k-1) \end{aligned} \tag{4-45}$$

where γ_{wait} is a weighting factor that can be taken larger than 1, because passengers usually feel that time passes slower when they are waiting for a train. It is now important to also write the objective function in terms of $\breve{x}(k)$, such that we have a quadratic optimization problem with linear constraints (4-21). To do so, we start by defining the total travel time for M trains as $\breve{\boldsymbol{t}}_{\text{total}}(k) = \begin{pmatrix} t_{\text{total}}(k) & t_{\text{total}}(k+1) & \cdots & t_{\text{total}}(k+M-1) \end{pmatrix}^T$. Then we see that:

$$\begin{aligned}
\begin{pmatrix} t_{\text{total}}(k) \\ t_{\text{total}}(k+1) \\ \vdots \\ t_{\text{total}}(k+M-1) \end{pmatrix} &= \operatorname{diag}\begin{pmatrix} x(k) \\ x(k+1) \\ \vdots \\ x(k+M-1) \end{pmatrix} \cdot \begin{pmatrix} H_* & 0 & \cdots & 0 \\ 0 & H_* & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & H_* \end{pmatrix} \cdot \begin{pmatrix} x(k) \\ x(k+1) \\ \vdots \\ x(k+M-1) \end{pmatrix} \\
&\quad + \operatorname{diag}\begin{pmatrix} x(k-1) \\ x(k) \\ \vdots \\ x(k+M-2) \end{pmatrix} \cdot \begin{pmatrix} F_* & 0 & \cdots & 0 \\ 0 & F_* & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & F_* \end{pmatrix} \cdot \begin{pmatrix} x(k) \\ x(k+1) \\ \vdots \\ x(k+M-1) \end{pmatrix} \\
&\quad + \operatorname{diag}\begin{pmatrix} x(k-1) \\ x(k) \\ \vdots \\ x(k+M-2) \end{pmatrix} \cdot \begin{pmatrix} C & 0 & \cdots & 0 \\ 0 & C & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & C \end{pmatrix} \cdot \begin{pmatrix} x(k-1) \\ x(k) \\ \vdots \\ x(k+M-2) \end{pmatrix}
\end{aligned} \tag{4-46}$$

This expression can be reshuffled, because $\breve{x}(k-1)$ can be written as a combination of $x(k-1)$ and the first $M-1$ rows of $\breve{x}(k)$. Furthermore, the total travel time for all passengers of the M trains that are considered, is $\breve{t}_{\text{total}}(k) = \bar{\mathbf{1}} \cdot \breve{\boldsymbol{t}}_{\text{total}}(k) = t_{\text{total}}(k) + t_{\text{total}}(k+1) + \cdots + t_{\text{total}}(k+M-1)$. The reduced relation is:

$$\begin{aligned}
\breve{t}_{\text{total}}(k) &= \begin{pmatrix} x(k) \\ x(k+1) \\ \vdots \\ x(k+M-1) \end{pmatrix}^T \cdot \begin{pmatrix} H_*+C & F_* & \cdots & 0 \\ 0 & H_*+C & \ddots & \vdots \\ \vdots & \vdots & \ddots & F_* \\ 0 & 0 & \cdots & H_* \end{pmatrix} \cdot \begin{pmatrix} x(k) \\ x(k+1) \\ \vdots \\ x(k+M-1) \end{pmatrix} \\
&\quad + x(k-1)^T \cdot \begin{pmatrix} F_* \\ 0 \\ \vdots \\ 0 \end{pmatrix}^T \cdot \begin{pmatrix} x(k) \\ x(k+1) \\ \vdots \\ x(k+M-1) \end{pmatrix} + x(k-1)^T \cdot C \cdot x(k-1)
\end{aligned} \tag{4-47}$$

which can be written more compactly as:

$$\breve{t}_{\text{total}}(k) = \breve{x}(k)^T \cdot \underbrace{\begin{pmatrix} H_* + C & F_* & \cdots & 0 \\ 0 & H_* + C & \ddots & \vdots \\ \vdots & \vdots & \ddots & F_* \\ 0 & 0 & \cdots & H_* \end{pmatrix}}_{\breve{H}} \cdot \breve{x}(k)$$
$$+ x(k-1)^T \cdot \underbrace{\begin{pmatrix} F_* \\ 0 \\ \vdots \\ 0 \end{pmatrix}^T}_{\breve{F}} \cdot \breve{x}(k) + x(k-1)^T \cdot C \cdot x(k-1) \tag{4-48}$$

The optimization problem now becomes:

$$\min_{\breve{x}(k)} \quad \breve{x}(k)^T \cdot \breve{H} \cdot \breve{x}(k) + x(k-1)^T \cdot \breve{F} \cdot \breve{x}(k) + x(k-1)^T \cdot C \cdot x(k-1) \tag{4-49}$$

Again, it can be observed that the matrix $\breve{H}$ is crucial: when this matrix is symmetric and positive definite (i.e., when all its (conventional) eigenvalues are larger than 0), then we see that this objective function is convex.

4-4 Control

By transforming the max-plus system back to linear inequality-constraints and adding an objective function, an optimal solution $\breve{x}_{\text{sol}}(k)$ can be found by using LP or QP. This schedule can be used as a control variable when implemented as:

$$\breve{u}(k) = \breve{x}_{\text{sol}}(k) \tag{4-50}$$
$$\breve{x}(k) \geq \breve{u}(k) \tag{4-51}$$

where $\breve{u}(k)$ is the control variable that can steer $\breve{x}(k)$ into the desired schedule within the max-plus framework. By adding (4-51) as an extra control variable to the original MP-LPV system that was defined in (3-43), it can be seen that this would result into the following expression:

$$\breve{x}(k) = \breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus \breve{A}_1 \otimes x(k-1) \oplus \breve{B} \otimes \breve{d}(k) \oplus \breve{u}(k) \tag{4-52}$$

where for convenience, $\breve{A}(\breve{p})$ and $\breve{A}_0$ were merged back into one matrix $\breve{A}_0(\breve{p})$, i.e. $\breve{A}_0(\breve{p}) = \breve{A}(\breve{p}) \oplus \breve{A}_0$.

Because of the $\oplus$-operator, $\breve{u}(k)$ can only delay $\breve{x}(k)$. This is not necessarily a problem: as long as the (convex) optimization problems find a minimal cost, it is guaranteed that no other (quicker) scheduling would give a better result in terms of the objective function, and thus that the solution $\breve{x}_{\text{sol}}$ yields the most ideal schedule. Consider the following Lemma:

Lemma 4 (Relaxation of the MP-LPV system)

The optimization problem $\min_{\breve{x}(k)} J(\breve{x}(k))$ subject to either $\breve{x}(k) \geq \breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus b$ **[1]** or $\breve{x}(k) = \breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus b \oplus \breve{u}(k)$ **[2]** gives the same cost value $J(\breve{x}(k))$, as long as $\breve{u}(k) = \breve{x}_{\text{sol},[1]}(k)$ is the solution to the first of the two problems.

A **Proof** will follow underneath.

Proof *Consider the MP-LPV system underneath:*

$$\breve{x}(k) = \breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus b$$

Since we have an equality sign, we can add a cost function to the problem without violating any mathematical rule:

$$\begin{aligned} \min_{\breve{x}(k)} \quad & J(\breve{x}(k)) \\ \text{s.t.} \quad & \breve{x}(k) = \breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus b \end{aligned}$$

As long as there exists a solution $\breve{x}(k)$ that fulfils this constraint, it is obvious that minimizing this objective function yields only one solution $\breve{x}_{sol,1}(k)$, because of the equality-sign. The cost that follows according to this solution is $J(\breve{x}_{sol,1}(k))$. This solution is what we can call the 'free-run schedule', which is the MP-LPV without any control. Now consider the following optimization problem:

$$\begin{aligned} \min_{\breve{x}(k)} \quad & J(\breve{x}(k)) \\ \text{s.t.} \quad & \breve{x}(k) \geq \breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus b \end{aligned}$$

This expression can also be rewritten to linear inequalities as could be seen in the previous sections. As long as $J(\breve{x}(k))$ is a convex function of $\breve{x}(k)$, then this optimization problem yields an optimal solution $\breve{x}_{sol,2}(k)$ which gives us that $J(\breve{x}_{sol,2}(k))$. Now we can add the steering control signal $\breve{u}(k)$ to the first problem, defining it as the solution of the second optimization problem:

$$\begin{aligned} \min_{\breve{x}(k)} \quad & J(\breve{x}(k)) \\ \text{s.t.} \quad & \breve{x}(k) = \breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus b \oplus \breve{u}(k) \\ & \breve{u}(k) = \breve{x}_{\text{sol},2}(k) \end{aligned}$$

Now we assume:

$$\begin{aligned}\breve{x}(k) &= \underbrace{\breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus b}_{\breve{\chi}(k)} \oplus \underbrace{\breve{u}(k)}_{\breve{x}_{\text{sol},2}(k)} \\ &= \max(\breve{\chi}(k), \breve{x}_{\text{sol},2}(k))\end{aligned}$$

and fill in at the right side of the equation that $\breve{x}(k) = \breve{x}_{sol,1}(k)$ (i.e., the free run schedule). There are three scenarios that possible: $\breve{x}_{sol,1}(k)$ is larger, smaller or equal to $\breve{x}_{sol,2}(k)$. Suppose that $\breve{x}_{sol,1}(k) \geq \breve{x}_{sol,2}(k)$, then $\breve{\chi}(k) > \breve{x}_{sol,2}(k)$ and the new solution $\breve{x}_{sol}(k) = \breve{\chi}(k)$ would be larger than $\breve{x}_{sol,2}(k)$, which implies that the schedule $\breve{x}_{sol}(k)$ would become worse than $\breve{x}_{sol,2}(k)$. However, the solution $\breve{x}_{sol,2}(k)$ was supposed to be the solution to a convex optimization problem, indicating that $J(\breve{x}_{sol,2}(k))$ is minimal. So a slower schedule $\breve{x}_{sol}(k)$ indicates a higher cost value, i.e. $J(\breve{x}_{sol}(k)) > J(\breve{x}_{sol,2}(k))$ which is not desired. The steering control variable would not do its job well.

Now consider $\breve{x}_{sol,2}(k) > \breve{x}_{sol,1}(k)$ and thus that $\breve{\chi}(k) < \breve{x}_{sol,2}(k)$. It would be clear that $\breve{x}_{sol}(k) = \max(\breve{\chi}(k), \breve{x}_{sol,2}(k))) = \breve{x}_{sol,2}(k)$ and the cost would remain $J(\breve{x}_{sol,2}(k))$.

Finally, the third condition is when $\breve{x}_{sol,1}(k) = \breve{x}_{sol,2}(k)$, indicating that $\breve{\chi}(k) = \breve{x}_{sol,2}(k)$, implying $J(\breve{x}_{sol}(k)) = J(\breve{x}_{sol,1}(k)) = J(\breve{x}_{sol,2}(k))$ and thus that adding $\breve{u}(k) = \breve{x}_{sol,2}(k)$ did not change the problem formulation. This means that $\breve{x}_{sol,2}(k) \geq \breve{x}_{sol,1}(k)$ guarantees that the two optimization problems will give the same cost when we add $\breve{u}(k) = \breve{x}_{sol,2}(k)$ and that we can rewrite the MP-LPV to linear inequalities to find the scheduling solution. ■

The **Proof** shows that the result of the optimization problem would be no different when adding $\breve{u}(k) = \breve{x}_{\text{sol},2}(k)$. Since the cost of the optimization problem with both a steering control signal and equality-sign constraints was the same as the cost with the inequality-sign constraints, converting them to linear inequalities is allowed and this relaxation will not influence the result. However, this can only be guaranteed as long as it holds that the objective function $J(\breve{x}(k))$ is convex.

It also shows us that in case of no disturbance, the two schedules $\breve{x}_{\text{sol},1}(k)$ and $\breve{x}_{\text{sol},2}(k)$ would be the same and that their cost would then also be the same. In case of a disturbance and thus a delayed $\breve{x}_{\text{sol},1}(k)$ and $\breve{x}_{\text{sol},2}(k)$, we can see that by adding the delayed $\breve{x}_{\text{sol},2}(k) > \breve{x}_{\text{sol},1}(k)$, the cost remains the same. This indicates that we can rewrite the second optimization problem to linear inequalities and still maintain the same cost.

Chapter 5

Case study

In the previous chapters, the MP-LPV system has been described and it was proposed how such a model could still be used and controlled. In this chapter, the theory up until now will be used in a case study to show if the formulation also holds in practice.

The case study consists of a fictional urban rail line, consisting of $J = 10$ stations and $K = 10$ trains. The idea of the optimization problem is that it takes M trains into account in every iteration. So if $M = 3$, there would be $K - M + 1 = 8$ updates. But since in this case study, we will only compute one M-steps-ahead prediction, we choose $M = K = 10$. Furthermore, the following constants will be used:

Constants		
Description	Constant	Value
Passenger rate	λ_1	0.9 [pass/s]
per station	λ_2	0.8 [pass/s]
	λ_3	1.3 [pass/s]
	λ_4	1.2 [pass/s]
	λ_5	1.4 [pass/s]
Minimal running time	$\tau_{\mathrm{r,min},1}$	60 [s]
	$\tau_{\mathrm{r,min},2}$	80 [s]
	$\tau_{\mathrm{r,min},3}$	40 [s]
	$\tau_{\mathrm{r,min},4}$	60 [s]
	$\tau_{\mathrm{r,min},5}$	30 [s]
Ratio of	ρ_1	0.2 [-]
alighting passengers	ρ_2	0.3 [-]
	ρ_3	0.5 [-]
	ρ_4	0.8 [-]
	ρ_5	0.7 [-]
Dwell time characteristic	$\alpha_{1,d}$	4.002 [s]
Dwell time characteristic	$\alpha_{2,d}$	0.047 [s/pass]
Dwell time characteristic	$\alpha_{3,d}$	0.05 [s/pass]
Ratio objective function	γ	0.5 [-]

Constants		
Description	Constant	Value
Passenger rate	λ_6	0.8 [pass/s]
per station	λ_7	0.8 [pass/s]
	λ_8	0.9 [pass/s]
	λ_9	1.2 [pass/s]
	λ_{10}	0 [pass/s]
Minimal running time	$\tau_{\mathrm{r,min},6}$	40 [s]
	$\tau_{\mathrm{r,min},7}$	50 [s]
	$\tau_{\mathrm{r,min},8}$	60 [s]
	$\tau_{\mathrm{r,min},9}$	80 [s]
	$\tau_{\mathrm{r,min},10}$	0 [s]
Ratio of	ρ_6	0.9 [-]
alighting passengers	ρ_7	0.6 [-]
	ρ_8	0.5 [-]
	ρ_9	0.3 [-]
	ρ_{10}	1 [-]
Minimal dwell time	$\tilde{\tau}_{\mathrm{d}}$	5 [s]
Minimal headway time	$\tau_{\mathrm{h,min}}$	60 [s]
Frequency of trains	$\tau_{\mathrm{h},1}$	90 [s]

Furthermore, the initialization vector, i.e. the 'zeroth' train $x(k-1)$ is chosen to be:

$$a^T(k-1) = \begin{pmatrix} 0 & 70 & 155 & 200 & 265 & 300 & 345 & 400 & 465 & 550 \end{pmatrix}$$
$$d^T(k-1) = \begin{pmatrix} 10 & 75 & 160 & 205 & 270 & 305 & 350 & 405 & 470 & 555 \end{pmatrix}$$
$$x(k-1) = \begin{pmatrix} a^T(k-1) & d^T(k-1) \end{pmatrix}^T$$

The simulations are done with MATLAB R2016b, with an Intel Core i5-3340M processor with 2.70 GHz CPU and 8 GB of RAM, on a 64-bits Windows 10 operating system. Most of the variables are loaded into MATLAB through an Excel-sheet. The LP optimization is solved with the MATLAB-commmand 'linprog' while the QP optimization is solved with 'quadprog'. An important mentioning has to be done about 'quadprog': it appears that the $\breve{H}$-matrix contains negative eigenvalues with these (and also for other choices of the) aforementioned constants, therefore not being positive definite and consequently, the optimization problem is non-convex. Furthermore, $\breve{H}$ is not symmetric. The latter can be solved easily by taking $\breve{H}^* = \frac{\breve{H}^T + \breve{H}}{2}$, something that is automatically done within the 'quadprog' command.

But the non-convex problem is a little more complex. Since it is not convex, it can not be guaranteed that the solution that is found is also the global optimum. The optimization might get stuck in a local optimum. However, trying to solve the same optimization problem with other solvers like CPLEX gives no result at all; it can not compute since $\breve{H}$ is not positive definite in the first place. This means that the build-in 'quadprog' command in MATLAB somehow manages to still compute for some cases. Even though the QP optimization perfectly computes in Case A and Case B, it can be seen in Case 3 that this is not always the case. Therefore, we are forced to mention that the results obtained with the 'quadprog' command should not be entirely trusted; it can not be guaranteed that it actually computes the global minimum. As a result of this, we will introduce a non-linear optimization method for Case C as well, by using the 'fmincon' build-in function in MATLAB and see if this gives trustworthy results.

Case A: undisturbed line

As a first case, $\breve{d}(k)$ will be set to $\mathcal{E}$ which means that there is no disturbance. This mean that we have the following MP-LPV:

$$\breve{x}(k) = \breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus \breve{A}_1 \otimes x(k-1) \oplus \breve{B} \otimes \breve{d}(k) = \breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus \breve{A}_1 \otimes x(k-1) \tag{5-1}$$

The simulation will be done in 4 different ways; both as a single-at-once [1] and multiple-at-once [2] cycle iterative MP-LPV, as a Linear Programming problem while minimizing $\breve{x}(k)$ [3] and as a Quadratic Programming problem while minimizing total passenger travel time [4]. The results can be seen in Figure 5-1.

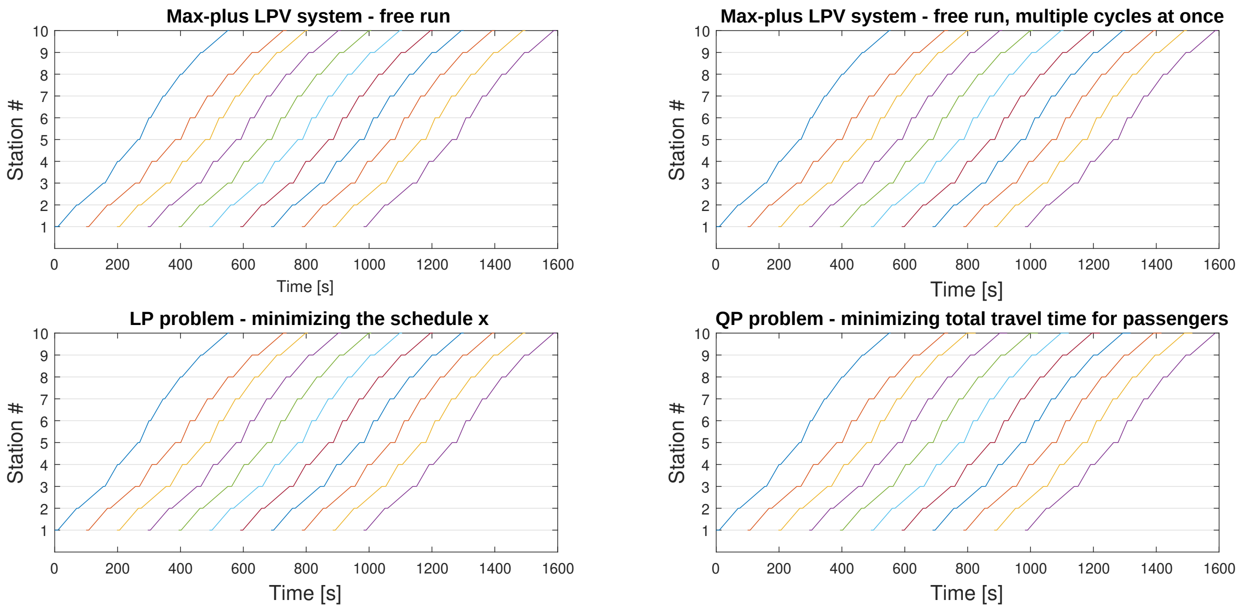


Figure 5-1: Comparison of performance between the free run iterative MP-LPV simulations (above) and the relaxed, linear inequality versions solved within an optimization framework (under). There is no disturbance so they all behave the same. The 'first' train is just an initialization and is in fact x(k-1).

Since there is no disturbance, all the methods yield the same result: the 'quickest' schedule $\breve{x}(k)$ is the fastest free run. With free run, the pure max-plus system is meant, without any objective function; every train k leaves whenever it is possible. Since the LP problem minimizes $\breve{x}(k)$, and since we showed before that even though it is a relaxed formulation it should give the same answer, we can find that its optimum is equal to a free run schedule. To show that that the schedules are indeed equal, we can take a look at the value of the cost for minimizing $\breve{x}(k)$ ($J_{\mathrm{LP}}(\breve{x}(k))$) and for minimizing total passenger travel time ($J_{\mathrm{QP}}(\breve{x}(k))$):

	Results		
Method	$J_{\mathrm{LP}}(\breve{x}(k))$	$J_{\mathrm{QP}}(\breve{x}(k))$	Computation time [s]
1	$1.6904 \cdot 10^5$	$1.2704 \cdot 10^7$	0.0899
2	$1.6904 \cdot 10^5$	$1.2704 \cdot 10^7$	1.0581
3	$1.6904 \cdot 10^5$	$1.2704 \cdot 10^7$	0.3830
4	$1.7079 \cdot 10^5$	$1.2704 \cdot 10^7$	0.4048

As can be seen from Table 5, the schedules are indeed the same. The only side-note that has to be given is the value for the QP-solution (number 4) in the LP cost function; it differs slightly from the other cost values, but the reason is simple. Since the QP problem minimizes the total passenger travel time, it does not take into account the departure time of the last train at the last station. The reason being that because $\lambda_{10} = 0$ and $\rho_{10} = 1$, there are no passengers in the train any more. Since there is no next train, it is not forced to leave the station. It can be seen from the value for the QP-solution in the QP cost function that it does not matter for the optimization if $x_{10}(10)$ is large or small: they all give the same value. But by taking away the last entry of $\breve{x}(k)$, this problem can be avoided and we would see that the costs are all the same in the case of simulation without disturbance. In Figure 5-1, this last departure time of last train has manually been reduced such that the figure shows up well.

Case B: disturbed line

In the second case, we would like to see the effect of anticipating on the (expected) disturbance: the entry of $\breve{d}(k)$ that is responsible for an external disturbance on train 3, station 2 will be set to 500 seconds. This mean that the departure time of the third train is now fixed at 500 seconds. The resulting MP-LPV is:

$$\breve{x}(k) = \breve{A}_0(\breve{p}) \otimes \breve{x}(k) \oplus \breve{A}_1 \otimes x(k-1) \oplus \breve{B} \otimes \breve{d}(k)$$

$$\breve{d}_i(k) = \begin{cases} 500 & \text{, if } i = 2 \cdot J \cdot k + (j + J) = 2 \cdot 10 \cdot 3 + (2 + 10) = 72 \\ \epsilon & \text{, else} \end{cases}$$

Now, there will be some notable difference with the undisturbed case. This can be seen from Figure 5-2.

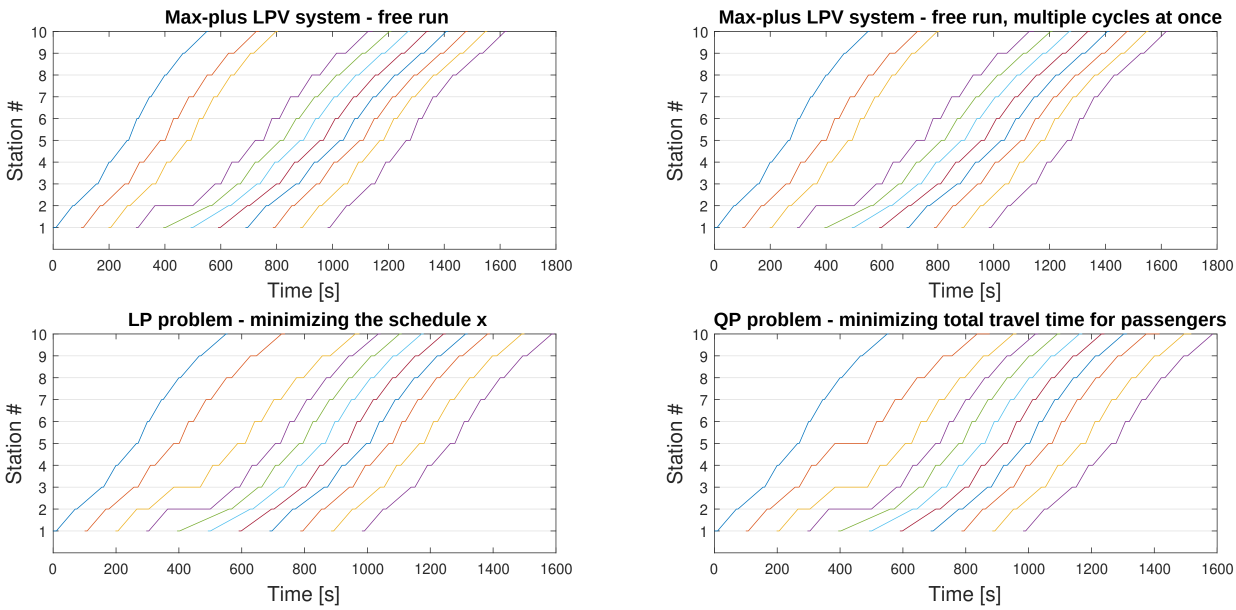


Figure 5-2: Comparison of performance between the free run iterative MP-LPV simulations (above) and the relaxed, linear inequality versions solved within an optimization framework (under). There is a disturbance on the departure time of train 3 at station 2. The 'first' train is just an initialization and is in fact x(k-1).

Clearly, the disturbance causes the LP and QP solutions to be different from the free run-variants. The optimizations work with an M-steps-ahead principle, which lets them anticipate on the disturbance with respect to their objective functions. This can clearly be seen by inspecting the behaviour of train 2, the train before train 3: it anticipates on the disturbance by going slower as well, which makes sure that the train 3 does not run away from its predecessors. The free run (both one cycle as multiple cycles at once) MP-LPV systems behave similar to each other but are not controlled, which explains the results in Table 5. Also, observe that the free run schedules are still finite, but from the behaviour of train 3 it seems obvious that - as the station counter J goes to infinity -, the headway time between train 2 and 3 goes to infinity (as has been shown in the **Proof** in Section 4-1). This does not happen for the optimized schedule: everything gets delayed accordingly to the disturbance such that a lower cost value can be found (both for minimizing $\breve{x}(k)$ and for minimizing total passenger travel time).

	Results		
Method	$J_{\mathrm{LP}}(\breve{x}(k))$	$J_{\mathrm{QP}}(\breve{x}(k))$	Computation time [s]
1	$1.8100 \cdot 10^5$	$1.5751 \cdot 10^7$	0.0466
2	$1.8100 \cdot 10^5$	$1.5751 \cdot 10^7$	1.0758
3	$1.7767 \cdot 10^5$	$1.3826 \cdot 10^7$	0.2374
4	$1.8364 \cdot 10^5$	$1.3698 \cdot 10^7$	0.2545

From the results, it becomes clear that the best value for the LP cost is achieved by the LP optimization itself, something that also can be said for the best value for the QP cost (which is found by the QP optimization). Clearly, the optimization strategies find a minimum cost that satisfies their respective objective functions. However, for the QP formulation, it can not be proven that this is a global optimum. The reason why will be explained in Case C. Still, the result is valid with respect to the constraints, which can be tested by taking $\breve{u} = \breve{x}_{\mathrm{sol,QP}}(k)$. If this control action is implemented in the original MP-LPV formulation, we can observe in Figure 5-3 that it is indeed a solution since the two schedules are equal.

Another interesting conclusion that can be made for Case B is about the computation speed: for both optimization problems, for these settings, it takes roughly a quarter of a second to compute the optimum.

Case C: disturbed with high passenger load

As has been mentioned in both the introduction of this chapter, as well as in the last paragraph of Case B: the build-in 'quadprog' command within MATLAB computes, but since we do not have a positive definite matrix $\breve{H}$ it can not be guaranteed that the QP solution for Case B is the global optimum. It will be shown that 'quadprog' does actually not compute any more if we raise the aforementioned passenger loads λ_j. We believe that the reason for 'quadprog' to still compute in Cases A and B is that the constraints create a safe bound on

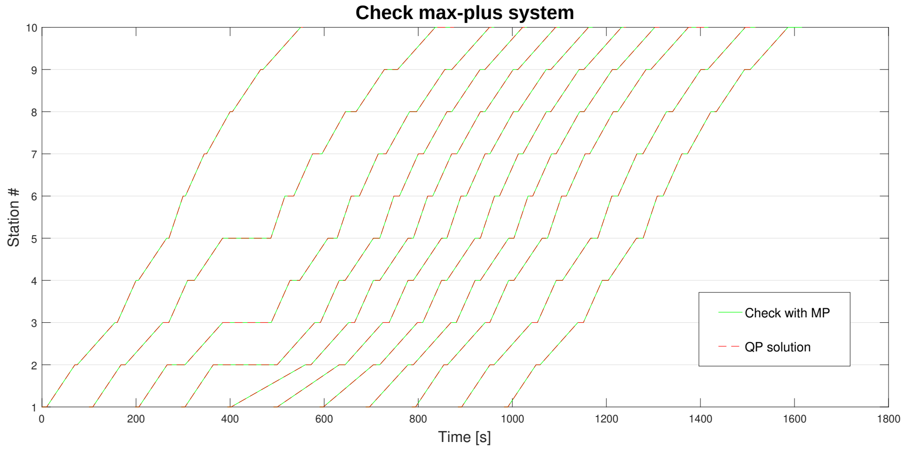


Figure 5-3: Solution of the QP optimization where afterwords $\breve{u} = \breve{x}_{\text{sol,QP}}(k)$ is taken as a control signal in the original MP-LPV, as a check. The 'first' train is just an initialization and is in fact x(k-1).

the objective function in which 'quadprog' can search as a convex problem. If we increase the passengers load sufficiently, these bounds shift and there is a point where it concludes that convex optimization is not possible any more.

Therefore, a non-linear optimization strategy [5] will be used for this case. The build-in function 'fmincon' can be used for this. If we increase the default number of iterations from 3000 to at least one million, it can find a local minimum which is very similar to the results of 'quadprog'. By increasing all but the last one of the λ_j by 1 (passenger per second), the following characteristics are found:

Constants		
Description	Constant	Value
Passenger rate	λ_1	1.9 [pass/s]
per station	λ_2	1.8 [pass/s]
	λ_3	2.3 [pass/s]
	λ_4	2.2 [pass/s]
	λ_5	2.4 [pass/s]

Constants		
Description	Constant	Value
Passenger rate	λ_6	1.8 [pass/s]
per station	λ_7	1.8 [pass/s]
	λ_8	1.9 [pass/s]
	λ_9	2.2 [pass/s]
	λ_{10}	0 [pass/s]

The rest of the characteristics will remain the same. The non-linear optimization basically uses the exact same objective function as the quadratic programming objective function as defined in (4-49), which is:

$$\min_{\breve{x}(k)} \quad \breve{x}(k)^T \cdot \breve{H} \cdot \breve{x}(k) + x(k-1)^T \cdot \breve{F} \cdot \breve{x}(k) + x(k-1)^T \cdot C \cdot x(k-1)$$

As a starting initial point, we use a vector $x_0 = \bar{1} \in \mathbb{R}^{2 \cdot J \cdot M}$ i.e. a vector filled with only ones. As a lower bound on the solution, we use the solution of the free run MP-LPV systems for the undisturbed case in Case A, since we know that the max-plus algebra will only allow trains to be equally fast or slower in case of a disturbance. There is no upper bound and there are no equality constraints. The amount of maximum function evaluations is set to 10^6.

Since the QP optimization does not work for these constraints, its solution is replaced with the non-linear optimization solution in Figure 5-4:

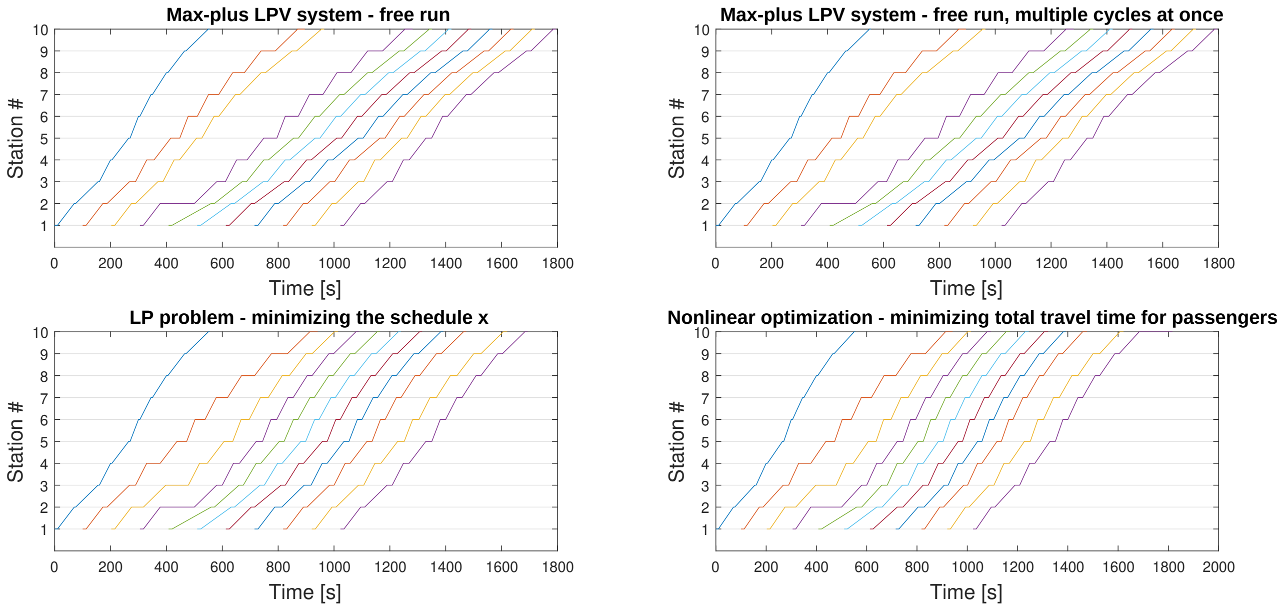


Figure 5-4: Comparison of performance between the free run iterative MP-LPV simulations (above) and the relaxed, linear inequality versions solved within an optimization framework (under). There is a disturbance on the departure time of train 3 at station 2. The 'first' train is just an initialization and is in fact x(k-1).

The resulting cost values for both minimizing $\breve{x}(k)$ $(J_{\mathrm{LP}}(\breve{x}(k)))$ and minimizing the total passenger travel time $(J_{\mathrm{QP}}(\breve{x}(k)))$ as well as the computation times can be found underneath:

	Results		
Method	$J_{\mathrm{LP}}(\breve{x}(k))$	$J_{\mathrm{QP}}(\breve{x}(k))$	Computation time [s]
1	$1.9581 \cdot 10^5$	$3.9206 \cdot 10^7$	0.0456
2	$1.9581 \cdot 10^5$	$3.9206 \cdot 10^7$	1.0418
3	$1.8729 \cdot 10^5$	$3.0357 \cdot 10^7$	0.2486
4	-	-	-
5	$1.8778 \cdot 10^5$	$3.0087 \cdot 10^7$	17.3326

Clearly, the non-linear optimization takes more computation time than the LP optimization, but it is still a tolerable amount of time considering that an update is only required approximately every $\tau_{\mathrm{h},1} = 90$ seconds (i.e. the time it approximately takes for the next train $x(k)$ to finish and become the new $x(k-1)$). Furthermore, we see that the non-linear solution is a little bit better than the LP solution, but not considerably much better. It also still holds that the LP solution is still the 'quickest' $\breve{x}(k)$ considering its objective function, which is to be expected. However, it does not differ that much. Since both $J_{\text{LP}}(\breve{x}(k))$ and $J_{\text{QP}}(\breve{x}(k))$ do not differ that much from each other, it can be concluded that the solutions are very much similar and thus that the LP solution is much more favourable in terms of computation time.

If computation time is considered an issue, it is also possible to work with a 'relaxed' QP optimization [6]. In this case, we can compute the conventional eigenvalues of the matrix $\breve{H}$. Since some of them are negative and thus the matrix is indefinite, it can artificially be made convex by adding a slightly larger value than the largest (negative) eigenvalue, as following:

$$\begin{aligned}
\min_{\breve{x}(k)} \quad & \breve{x}(k)^T \cdot \breve{H} \cdot \breve{x}(k) \boldsymbol{+c \cdot \breve{x}(k)^T \breve{x}(k)} + x(k-1)^T \cdot \breve{F} \cdot \breve{x}(k) + x(k-1)^T \cdot C \cdot x(k-1) \\
& = \breve{x}(k)^T \cdot \Big(\underbrace{\breve{H} + cI}_{\breve{H}^*}\Big) \cdot \breve{x}(k) + x(k-1)^T \cdot \breve{F} \cdot \breve{x}(k) + x(k-1)^T \cdot C \cdot x(k-1) \\
& \qquad \text{with} \\
& \quad c > -\min \operatorname{eig}\Big(\breve{H}^T + \breve{H}\Big)
\end{aligned}$$

For the current setting, we find that $\min \operatorname{eig}\Big(\breve{H}^T + \breve{H}\Big) = -3.5668$, so by adding $c = 3.6$ we will find a new $\breve{H}^*$ that is positive definite since all its eigenvalues are positive. We can now use the 'quadprog' command to find Figure 5-5:

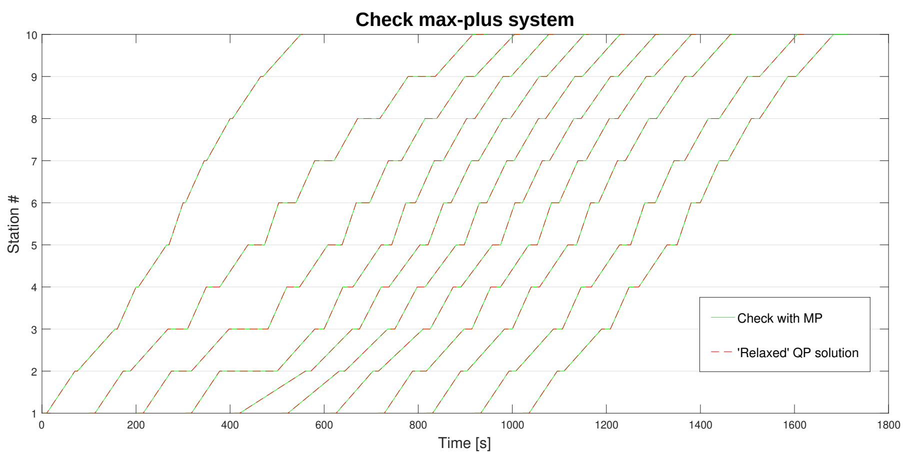


Figure 5-5: Solution of the 'relaxed' QP optimization where afterwords $\breve{u} = \breve{x}_{\text{sol,QP}}(k)$ is taken as a control signal in the original MP-LPV, as a check. The 'first' train is just an initialization and is in fact x(k-1).

Furthermore, the resulting new cost values for option [6] become:

	Results		
Method	$J_{\mathrm{LP}}(\breve{x}(k))$	$J_{\mathrm{QP}}(\breve{x}(k))$	Computation time [s]
3	$1.8729 \cdot 10^5$	$3.0357 \cdot 10^7$	0.2546
5	$1.8778 \cdot 10^5$	$3.0087 \cdot 10^7$	17.3326
6	$1.8731 \cdot 10^5$	$3.0351 \cdot 10^7$	0.2459

In able to compare, options [3] and [5] have also been added to the table. Since the objective function has changed due to the addition of $c \cdot \breve{x}(k)^T \breve{x}(k)$, the cost would normally be higher for [6]. For comparison, this has also been subtracted again while computing $J_{\mathrm{QP}}(\breve{x}(k))$. We can see that by making the objective function truly convex, the solution is computable again and almost entirely similar to the schedule that was obtained with LP. This indicates that introducing non-linear programming did not give us a better solution than LP and (relaxed) QP did. We can also observe that the difference between LP cost and the relaxed QP cost became smaller, indicating that minimizing $\breve{x}(k)$ gives similar results to minimizing the total travel time of passengers.

Case D

In order to be able to fully understand the flaws of 'quadprog', a situation in between of Case B and Case C will be shown: the case where the QP optimization still computes, but where we can see that the solution is not unique because the matrix $\breve{H}$ is not positive definite. Therefore, the QP optimization is not convex and we can not guarantee that the solution is unique to the cost value. The characteristics from the first table of this chapter will be used again, instead of the modified version in Case C.

In order to compare these systems, there will be 3 sub-cases: one case where the minimal eigenvalue of the matrix $\breve{H}$ is smaller than 0, another case where it is larger than 0 and finally one more case where it is 'sufficiently' larger than 0 to show that the solution from the non-linear optimization becomes the same as the solution from the QP optimization. The meaning of sufficiently will be explained in the latter case.

Case D.1

The first sub-case will be a similar to the Case B, except that we also compute the non-linear optimization problem. Furthermore, we raise the disturbance on the departure time of the third train in the second station to 700 seconds, i.e. $\breve{d}_{72} = 700$ instead of 500. This in order to show the difference more clearly in the figures; similar results can be obtained with the previous settings but they are less visible.

It can be observed from Figure 5-6 that the scheduling results for the non-linear optimization problem and the (non-convex) quadratic programming problem are different. Although the cost varies very little (the quadratic optimization problem gives a cost value that is 0.12% larger than the non-linear optimization problem). The difference in scheduling can be explained by the fact that the quadratic optimization problem is not convex, and that even though it computes it is not a guaranteed global minimum, so other schedules and minimums are possible. The small difference in cost value is harder to directly explain: it might be a numerical error but also (again) the fact that 'quadprog' finds a different local minimum than 'fmincon'.

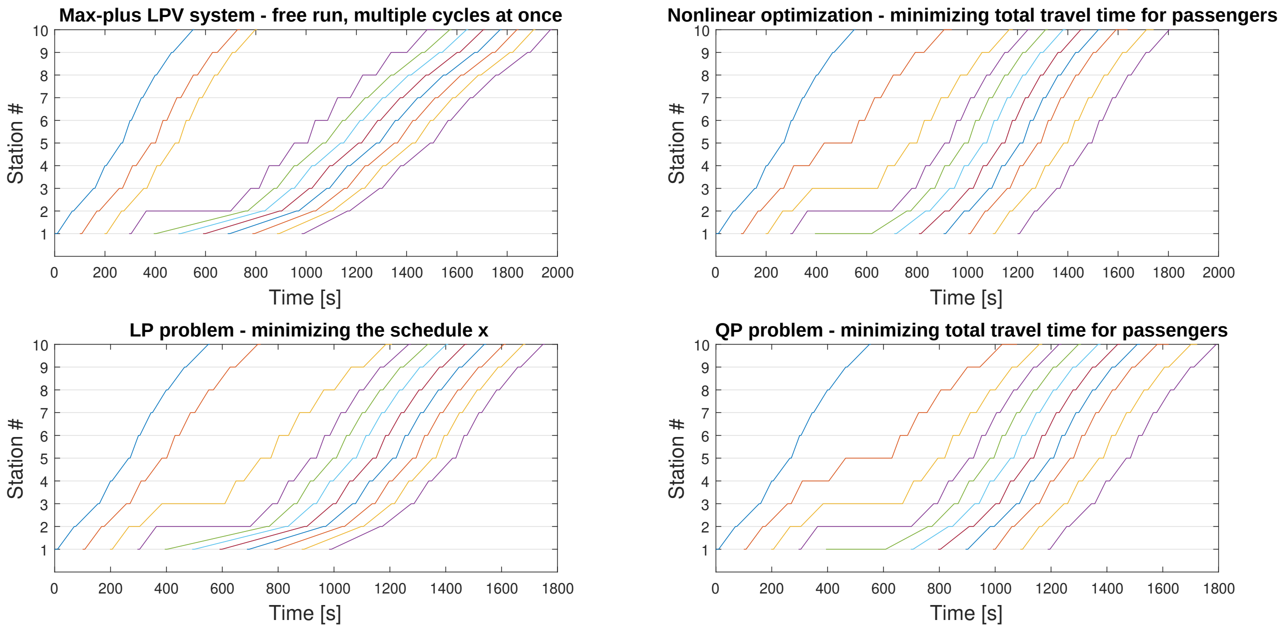


Figure 5-6: Comparison of performance between the free run iterative MP-LPV simulation and non-linear optimization (above) and linear programming and quadratic programming versions (under). There is a disturbance on the departure time of train 3 at station 2. The 'first' train is just an initialization and is in fact x(k-1).

From the table, it can be seen that the cost for the minimizing $\breve{x}(k)$ is still achieved by the LP optimization, but clearly the non-linear optimization finds a slightly better cost for minimizing total passenger travel time than the QP optimization. As mentioned before, this might be a numerical error, but at least indicates that 'quadprog' does not find a global minimum. Also, notice that reducing back the passengers rates made the non-linear optimization way quicker in terms of computation time in comparison with the non-linear optimization in Case C.

	Results		
Method	$J_{\text{LP}}(\breve{x}(k))$	$J_{\text{QP}}(\breve{x}(k))$	Computation time [s]
2	$2.2006 \cdot 10^5$	$2.5863 \cdot 10^7$	1.0335
3	$2.0929 \cdot 10^5$	$1.8861 \cdot 10^7$	0.2402
4	$2.2274 \cdot 10^5$	$1.6889 \cdot 10^7$	0.2543
5	$2.1577 \cdot 10^5$	$1.6867 \cdot 10^7$	3.9137

Case D.2

The second sub-case will take into account the minimum eigenvalue of $\breve{H} + \breve{H}$ again. This time, it can be found that it is $\lambda_{\min} = -2.0235$, which means that we can add $c = 2.1$ and make the QP objective function convex again. The following interesting effects are visible:

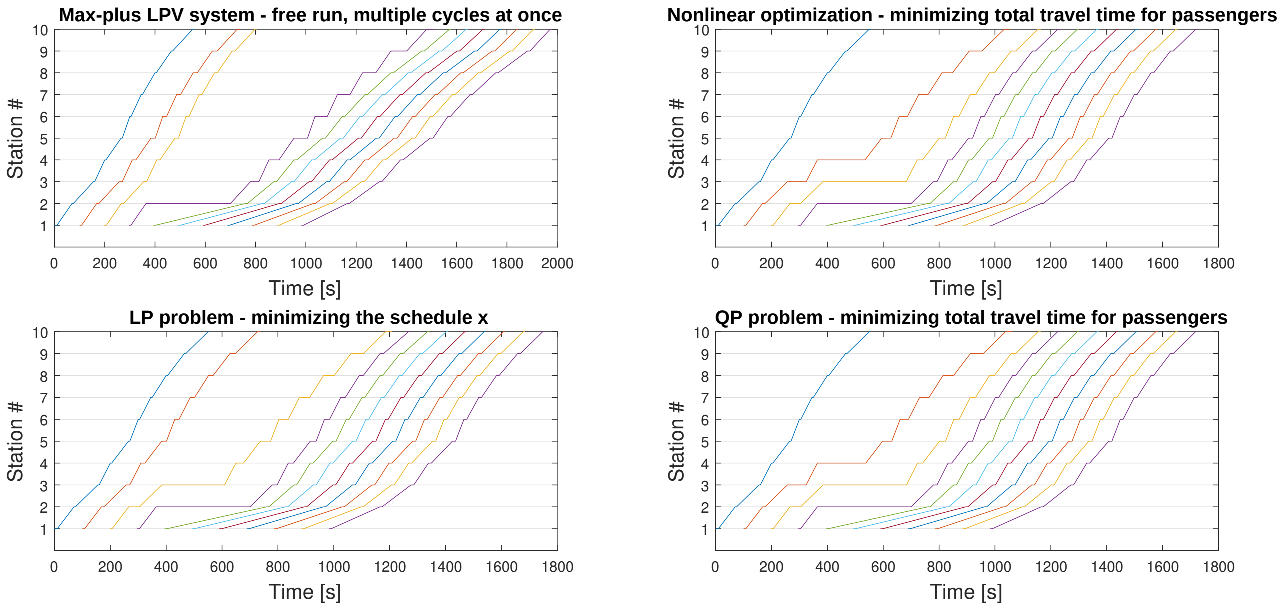


Figure 5-7: Comparison of performance between the free run iterative MP-LPV simulation and non-linear optimization (above) and linear programming and quadratic programming versions (under). There is a disturbance on the departure time of train 3 at station 2. The 'first' train is just an initialization and is in fact x(k-1).

In Figure 5-7, it can be seen that the two schedules on the right side are almost equal now, indicating that the QP optimization [6] indeed finds a global optimum and that the non-linear optimization [5] finds similar results. Now observe the table underneath:

	Results		
Method	$J_{\mathrm{LP}}(\breve{x}(k))$	$J_{\mathrm{QP}}(\breve{x}(k))$	Computation time [s]
2	$2.2006 \cdot 10^5$	$2.5863 \cdot 10^7$	1.0106
3	$2.0929 \cdot 10^5$	$1.8861 \cdot 10^7$	0.2404
5	$2.1039 \cdot 10^5$	$1.8166 \cdot 10^7$	11.7185
6	$2.1041 \cdot 10^5$	$1.8163 \cdot 10^7$	0.2534

The differences between the QP and non-linear optimizations are very small again, but this time, it can be seen from Figure 5-7 that they give very similar schedules. So clearly, the fact that the matrix $\breve{H}^*$ is convex now, seemingly results into a global minimum. The difference in cost value can be attributed to numerical differences in the two methods, since it is only 0.2% for the minimizing total passenger travel time-cost and 0.1% for the minimizing $\breve{x}(k)$-cost.

To show the actual little difference between the two methods, we can compute the average squared error between the two. We find that:

$$\frac{1}{n}e^T e = \frac{1}{2 \cdot J \cdot M}(\breve{x}_{QP} - \breve{x}_{NL})^T(\breve{x}_{QP} - \breve{x}_{NL}) = \frac{201.2}{200} = 1.0061$$

The average difference between arrival and departure times for the two schedules is approximately one second. Case D.3 shows that this can be reduced by increasing the value c and thus the eigenvalues of the matrix $\breve{H}^*$.

Case D.3

This time, the value c is set to 10. A figure will not be shown, because it would be too similar to Figure 5-7. Instead of a figure, we can observe the table with result underneath:

	Results		
Method	$J_{\text{LP}}(\breve{x}(k))$	$J_{\text{QP}}(\breve{x}(k))$	Computation time [s]
2	$2.2006 \cdot 10^5$	$2.5863 \cdot 10^7$	1.0328
3	$2.0929 \cdot 10^5$	$1.8861 \cdot 10^7$	0.2384
5	$2.1035 \cdot 10^5$	$1.8174 \cdot 10^7$	7.4794
6	$2.1036 \cdot 10^5$	$1.8172 \cdot 10^7$	0.2594

The differences become a little bit smaller again, which can be better seen from the average squared error:

$$\frac{1}{n}e^T e = \frac{1}{2 \cdot J \cdot M}(\breve{x}_{QP} - \breve{x}_{NL})^T(\breve{x}_{QP} - \breve{x}_{NL}) = \frac{47.2}{200} = 0.24$$

The numerical differences between Case D.2 and Case D.3 can be explained as following: by increasing the eigenvalues of $\breve{H}^*$, the slopes of the convex function become steeper and therefore it is easier for both programs to find the global optimum. Since many of the eigenvalues in Case D.2 were close to 0, the matrix $\breve{H}^*$ was close to being positive semi-definite, which could indicate a poly-dimensional surface of solutions. The steeper the slopes, the better the algorithms can work in order to approximate the global optimum.

Therefore, by substantially increasing the eigenvalue of the matrix $\breve{H}^*$, a global optimum can be guaranteed and the non-linear programming problem shows that this is indeed a similar solution.

Chapter 6

Evaluation within the max-plus algebra framework

In the previous chapter, it was shown how the max-plus algebra framework could be used to control an urban railway line with a variable dwell time, by using a linear, quadratic and non-linear optimization problems that minimize both $\breve{x}(k)$ and the total travelling time for passengers. The resulting matrices $\hat{A}_0(p(k))$ in the matrix $\breve{A}(\breve{p})$ change from train to train and can therefore not easily be evaluated like most matrices in max-plus algebra (Heidergott et al., 2014). Still, the matrix $\breve{A}(\breve{p})$ contains important information about the urban railway line, such as the dwell times, minimal travel times, and some relations between $M-1$ trains with respect to each other.

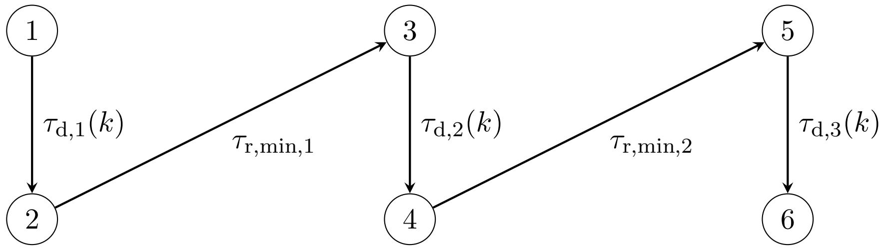
Therefore, it is interesting to use this information somehow, and evaluate such a linear parameter varying matrix nevertheless.

6-1 Analysis of the matrix $\hat{A}_0(p)$

For this section the matrix $A_0(p)$ (3-42) will be used, but with the simplification that we do not consider the minimal dwell times $\tilde{\tau}_\mathrm{d}$. If we consider $j = 3$, we find that $A_0(p) \in \mathbb{R}^{6\times 6}$ and that it looks as following:

$$A_0(p) = \left(\begin{array}{ccc|ccc} & & & \epsilon & \epsilon & \epsilon \\ & \mathcal{E} & & \tau_{\mathrm{r,min},1} & \epsilon & \epsilon \\ & & & \epsilon & \tau_{\mathrm{r,min},2} & \epsilon \\ \hline \tau_{\mathrm{d},1}(k) & \epsilon & \epsilon & & & \\ \epsilon & \tau_{\mathrm{d},2}(k) & \epsilon & & \mathcal{E} & \\ \epsilon & \epsilon & \tau_{\mathrm{d},3}(k) & & & \end{array}\right) \tag{6-1}$$

Or, equivalently in graph theory:



With a close look to this graph, it can be be observed that every node is connected to the following node, except for node 6 which does not relate back to node 1. Whenever an arc (i, j) exists but the retuning arc (j, i) does not exist, we have a so-called *directed graph*. When there exists a path between node i and node j (in this order), then j is reachable from node i. A graph $\mathcal{G}(A)$ is *strongly connected* when every node j is reachable from node i. A strongly connected graph $\mathcal{G}(A)$ implies that the matrix A is *irreducible*. Since node 1 is not reachable, $\mathcal{G}(A_0(p))$ is not strongly connected and therefore the matrix $A_0(p)$ is called *reducible*.

The irreducibility of a matrix A is of great importance for the max-plus eigenvalue of that matrix. When a matrix is irreducible, it contains at least one circuit which contains an average circuit weight. The average circuit weight is directly coupled to the definition of a max-plus eigenvalue, as could be seen from (2-20), and it is unique.

It should be noted that the matrix $A_0(p)$ on itself is not of great importance if the eigenvalue of the MP-LPV system has to be found: it does not contain all the information of the system. The rest of the information can be found in A_1, so we will need to combine this information somehow.

6-1-1 Kleene star of the matrix $A_0(p)$

It is best to introduce some more theory first. An important property of a matrix is the *nilpotency*. A matrix $A \in \mathbb{R}_{\max}^{n\times n}$ is nilpotent if the following is true:

$$\begin{aligned} &A^{\otimes\, n-1} \neq \mathcal{E} \\ &A^{\otimes\, n+i} = \mathcal{E} \quad \text{for} \quad i \geq 0 \end{aligned} \tag{6-2}$$

If this is true, then it is also possible to compute the Kleene star of that matrix A (2-17). The Kleene star would allow us to write the problem as (2-15) (see Example 1.2), which on its turn can be used to extract information from using Karp's algorithm (subsection 6-1-2).

If we consider again the example of the matrix $A_0(p)$ as in (6-1), with $n = 6$, it is easily verifiable to see that:

$$\begin{aligned} A_0(p)^{\otimes\, n-1} &= A_0(p)^{\otimes\, 5} \\ &= A_0(p) \otimes A_0(p) \otimes A_0(p) \otimes A_0(p) \otimes A_0(p) \\ &= \left(\begin{array}{ccc|c} & \mathcal{E} & & \mathcal{E} \\ \hline \epsilon & \epsilon & \epsilon & \\ \epsilon & \epsilon & \epsilon & \mathcal{E} \\ \tau_{\mathrm{d},1}(k) + \tau_{\mathrm{d},2}(k) + \tau_{\mathrm{d},3}(k) + \tau_{\mathrm{r,min},1} + \tau_{\mathrm{r,min},2} & \epsilon & \epsilon & \end{array}\right) \end{aligned} \tag{6-3}$$

Clearly, this matrix has one finite element left. Since $A_0(p)^{\otimes n} = A_0(p)^{\otimes n-1} \otimes A_0(p) = \mathcal{E}$, it can be concluded that $A_0(p)$ is nilpotent. Therefore, it is also possible to compute the Kleene star $A_0^*(p)$:

$$
\begin{aligned}
A_0^*(p) &= E \oplus A_0(p) \oplus A_0^2(p) \oplus \ldots \oplus A_0^{n-1}(p) \\
&= E \oplus A_0(p) \oplus A_0^2(p) \oplus \ldots \oplus A_0^5(p) \\
&= E \oplus \left(\begin{array}{ccc|ccc}
 & & & \epsilon & \epsilon & \epsilon \\
 & \mathcal{E} & & \tau_{\mathrm{r,min},1} & \epsilon & \epsilon \\
 & & & \epsilon & \tau_{\mathrm{r,min},2} & \epsilon \\
\hline
\tau_{\mathrm{d},1}(k) & \epsilon & \epsilon & & & \\
\epsilon & \tau_{\mathrm{d},2}(k) & \epsilon & & \mathcal{E} & \\
\epsilon & \epsilon & \tau_{\mathrm{d},3}(k) & & &
\end{array}\right) \\
&\quad \oplus \ldots \oplus \left(\begin{array}{ccc|c}
 & \mathcal{E} & & \mathcal{E} \\
\hline
\epsilon & \epsilon & \epsilon & \\
\epsilon & \epsilon & \epsilon & \mathcal{E} \\
\tau_{\mathrm{d},1}(k) + \tau_{\mathrm{d},2}(k) + \tau_{\mathrm{d},3}(k) + \tau_{\mathrm{r,min},1} + \tau_{\mathrm{r,min},2} & \epsilon & \epsilon &
\end{array}\right) \\
&= \left(\begin{array}{c|c}
A^*_{0,[a,a]}(p) & A^*_{0,[a,d]}(p) \\
\hline
A^*_{0,[d,a]}(p) & A^*_{0,[d,d]}(p)
\end{array}\right)
\end{aligned}
\tag{6-4}
$$

where:

$$
\begin{aligned}
A^*_{0,[a,a]}(p) &= \begin{pmatrix}
e & \epsilon & \epsilon \\
\tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{d},1}(k) & e & \epsilon \\
\tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},1}(k) \otimes \tau_{\mathrm{d},2}(k) & \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},2}(k) & e
\end{pmatrix} \\
A^*_{0,[a,d]}(p) &= \begin{pmatrix}
\epsilon & \epsilon & \epsilon \\
\tau_{\mathrm{r,min},1} & \epsilon & \epsilon \\
\tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},2}(k) & \tau_{\mathrm{r,min},2} & \epsilon
\end{pmatrix} \\
A^*_{0,[d,a]}(p) &= \begin{pmatrix}
\tau_{\mathrm{d},1}(k) & \epsilon & \epsilon \\
\tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{d},1}(k) \otimes \tau_{\mathrm{d},2}(k) & \tau_{\mathrm{d},2}(k) & \epsilon \\
\boldsymbol{\tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},1}(k) \otimes \tau_{\mathrm{d},2}(k) \otimes \tau_{\mathrm{d},3}(k)} & \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},2}(k) \otimes \tau_{\mathrm{d},3}(k) & \tau_{\mathrm{d},3}(k)
\end{pmatrix} \\
A^*_{0,[d,d]}(p) &= \begin{pmatrix}
e & \epsilon & \epsilon \\
\tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{d},2}(k) & e & \epsilon \\
\tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},2}(k) \otimes \tau_{\mathrm{d},3}(k) & \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},3}(k) & e
\end{pmatrix}
\end{aligned}
$$

The link to graph theory can be made very clearly now. The partitioning of $A_0^*(p)$ is done on purpose like this: the subscripts a and d indicate if we talk about an arrival of departure event respectively. The second subscript basically indicates 'coming from' and the first subscript indicates 'going to'. For example, the bold expression in $A^*_{0,[d,a]}(p)$ is the weight of the arc from node/station 1 to 3, since it is element $\{3, 1\}$. It is evident that this element is the sum of all the individual dwell and minimal travel times, since node 1 in $A^*_{0,[d,a]}(p)$ represents the arrival time at station 1 and node 3 represents the departure time at station 3 and thus, the weight of the arc should be exactly all these times summed up. We can also observe that going from a specific arrival or departure time back to itself has weight $e = 0$, because it it already there. Similarly, going from a departure time to an arrival time has weight $\epsilon = -\infty$ because it is not possible to go back to this event any more. See Figure 6-1 for a more visual representation.

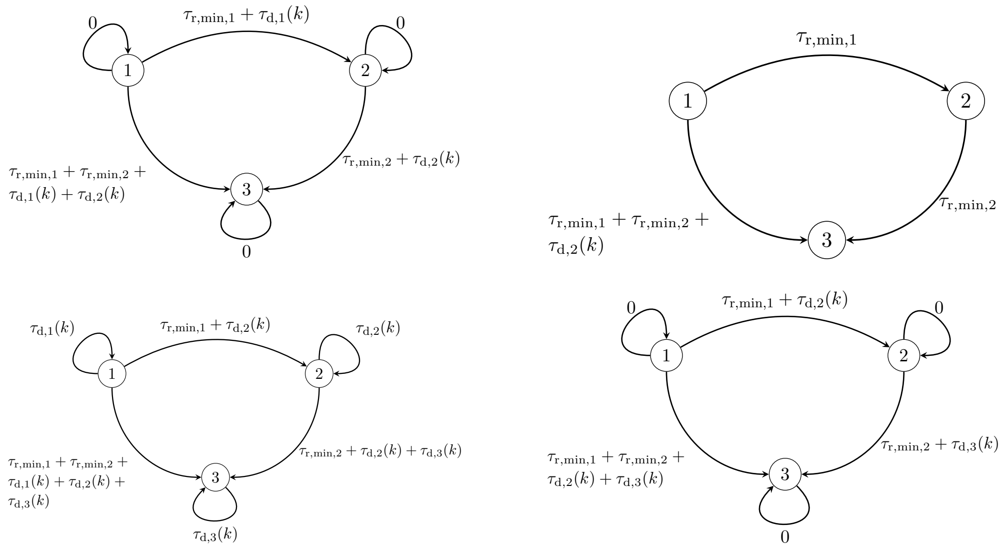


Figure 6-1: The four sub-graphs of $A_0^*(p)$, from left to right: $A^*_{0,[a,a]}(p)$ and $A^*_{0,[a,d]}(p)$ above and $A^*_{0,[d,a]}(p)$ and $A^*_{0,[d,d]}(p)$ below

Another interesting property of the Kleene star can be used as well. According to (2-15), the unique solution x_{sol} can be found by writing:

$$x_{\text{sol}} = A^* \otimes b \tag{6-5}$$

which means that:

$$x_{\text{sol}}(k) = A_0^*(p) \otimes b = \underbrace{A_0^*(p) \otimes A_1}_{\mathcal{A}(p)} \otimes x(k-1) \tag{6-6}$$

where of course again, the only problem is the implicit relation of $\mathcal{A}(p)$ on $x(k)$. But nevertheless, once the entries of $A_0^*(p)$ are known or well-estimated, this theorem can be applied and the schedule of train k can be found in just one computation.

6-1-2 Karp's algorithm

As has been mentioned before, irreducibility of a matrix A guarantees that there is one, unique eigenvalue. According to Heidergott et al. (2014), there are however also other methods to discover an eigenvalue of a matrix A, even if it reducible. One of those algorithms is Karp's algorithm, which can help to find (multiple) eigenvalues $\lambda(A)$ of a matrix A. The methodology will be explained below, after which an example will be shown:

Karp's algorithm

1. *Choose an arbitrary* $j \in \{1, 2, ..., n-1\}$ *and set the initial state as a unit vector* $x(0) = e_j$

2. *Compute* $x(k)$ *for* $k = 0, 1, ..., n$

3. *Compute the eigenvalue as:* $\lambda = \max_{i=1,...,n} \min_{k=0,...,n-1} \frac{x_i(n)-x_i(k)}{n-k}$

Example 5.1 *Given a matrix* $A = \left(\begin{array}{ccc|ccc} & \mathcal{E} & & \epsilon & \epsilon & \epsilon \\ & & & 60 & \epsilon & \epsilon \\ & & & \epsilon & 80 & \epsilon \\ \hline 38 & \epsilon & \epsilon & & & \\ \epsilon & 58 & \epsilon & & \mathcal{E} & \\ \epsilon & \epsilon & 36 & & & \end{array}\right)$ *and* $x(0) = e_1 = \begin{pmatrix} 0 \\ \epsilon \\ \epsilon \\ \epsilon \\ \epsilon \\ \epsilon \end{pmatrix}$,

we find that: $x(1) = \begin{pmatrix} \epsilon \\ \epsilon \\ \epsilon \\ 36 \\ \epsilon \\ \epsilon \end{pmatrix}$, $x(2) = \begin{pmatrix} \epsilon \\ 96 \\ \epsilon \\ \epsilon \\ \epsilon \\ \epsilon \end{pmatrix}$, $x(3) = \begin{pmatrix} \epsilon \\ \epsilon \\ \epsilon \\ \epsilon \\ 154 \\ \epsilon \end{pmatrix}$, $x(4) = \begin{pmatrix} \epsilon \\ \epsilon \\ 234 \\ \epsilon \\ \epsilon \\ \epsilon \end{pmatrix}$, $x(5) = \begin{pmatrix} \epsilon \\ \epsilon \\ \epsilon \\ \epsilon \\ \epsilon \\ 272 \end{pmatrix}$

and for $k \geq 6$, $x(k) = \begin{pmatrix} \epsilon \\ \epsilon \\ \epsilon \\ \epsilon \\ \epsilon \\ \epsilon \end{pmatrix}$. *We now find, for* $n = 6$ *and* $i = 1$*:*

$$\begin{aligned} \lambda_1 &= \min_{k=0,...,5} \frac{x_1(n) - x_1(k)}{6 - k} \\ &= \min(\frac{x_1(6) - x_1(0)}{6 - 0}, \frac{x_1(6) - x_1(1)}{6 - 1}, ..., \frac{x_1(6) - x_1(5)}{6 - 5}) \\ &= \min(\epsilon, 0, ..., 0) \\ &= \epsilon \end{aligned} \tag{6-7}$$

If we apply the algorithm for $i = 2, 3, ..., 6$ *as well, we find similar results as* $\lambda_i = \epsilon$*, which means that the eigenvalue of this reducible matrix is* $\lambda = max(\epsilon, \epsilon, ..., \epsilon) = \epsilon$*. If we would remove the first row and last column of* A*, the situation would be quite different since this would be an irreducible matrix. In that case, the eigenvalue according to Karp's algorithm would be finite.*

It is interesting to apply Karp's algorithm on $\mathcal{A}(p)$ [1] and see if we can finally extract some information on the eigenvalue of this (reducible) matrix. Since A_1 only consists of constant

[1] In other work, this matrix is often just called the A matrix of the max-plus system, but since this thesis adds the dependency on $p(k)$ it has been chosen to rename it to avoid confusion.

values, $\mathcal{A}(p)$ can be determined to be:

$$\mathcal{A}(p) = \left(\begin{array}{c|ccc} \mathcal{E} & \tau_{\mathrm{h},1} & \epsilon & \epsilon \\ & \tau_{\mathrm{h},1} \otimes \tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{d},1}(k) & \tau_{\mathrm{h},2} & \epsilon \\ & \tau_{\mathrm{h},1} \otimes \tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},1}(k) \otimes \tau_{\mathrm{d},2}(k) & \tau_{\mathrm{h},2} \otimes \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},2}(k) & \tau_{\mathrm{h},3} \\ \hline & \boldsymbol{\tau_{\mathrm{h},1} \otimes \tau_{\mathrm{d},1}(k)} & \epsilon & \epsilon \\ \mathcal{E} & \tau_{\mathrm{h},1} \otimes \tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{d},1}(k) \otimes \tau_{\mathrm{d},2}(k) & \boldsymbol{\tau_{\mathrm{h},2} \otimes \tau_{\mathrm{d},2}(k)} & \epsilon \\ & \tau_{\mathrm{h},1} \otimes \tau_{\mathrm{r,min},1} \otimes \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},1}(k) \otimes \tau_{\mathrm{d},2}(k) \otimes \tau_{\mathrm{d},3}(k) & \tau_{\mathrm{h},2} \otimes \tau_{\mathrm{r,min},2} \otimes \tau_{\mathrm{d},2}(k) \otimes \tau_{\mathrm{d},3}(k) & \boldsymbol{\tau_{\mathrm{h},3} \otimes \tau_{\mathrm{d},3}(k)} \end{array}\right) \tag{6-8}$$

where the bold expressions are especially important, because applying Karp's algorithm (with $j = 4, 5, 6$ because $j = 1, 2, 3$ results in vectors of $x(k)$ filled with ϵ only) results into the following:

$$\lambda(\mathcal{A}(p)) = \max(\tau_{\mathrm{h},1} + \tau_{\mathrm{d},1}(k), \tau_{\mathrm{h},2} + \tau_{\mathrm{d},2}(k), \tau_{\mathrm{h},3} + \tau_{\mathrm{d},3}(k)) \tag{6-9}$$

The eigenvalue of this matrix $\mathcal{A}(p)$ is the largest combination of dwell and minimal headway time. In general, this means that:

$$\lambda(\mathcal{A}(p)) = \bigoplus_{j=1}^{J} (\tau_{\mathrm{h},j} \otimes \tau_{\mathrm{d},j}(k)) \tag{6-10}$$

It should again be noted that $\tilde{A}_0$ has not been entirely included in this analysis. It has been assumed that the minimal dwell times $\tilde{\tau}_{\mathrm{d}}$ will always be exceeded, which can be done as long as there are enough passengers per second λ_j on the station of the urban rail line. Nevertheless, the addition of $\tilde{\tau}_{\mathrm{d}}$ would not change the general procedure; there would just be an extra max-operator around the dwell times, because of the $\oplus$ in $A_0(p) = \hat{A}_0(p) \oplus \tilde{A}_0$.

6-1-3 Growth rate of the max-plus system

Stability of the urban railway network is an important feature we would like to prove. The question is whether one of the dwell times $\tau_{\mathrm{d},j}(k))$ or some disturbance $d(k)$ can cause the system to become unstable. For a good estimation of the reachable region of $A_0^*(p)$, we can try to bound it with a lower and upper bound. First, let us define a new relation (Lemma 25 from (Farlow, 2009) or Lemma 3.10 from Heidergott et al. (2014)). For any (not necessarily square) regular matrix $A \in \mathbb{R}_{\max}^{m \times n}$ and vectors $u, v \in \mathbb{R}^n$, we can define:

$$\|(A \otimes u) - (A \otimes v)\|_\infty \leq \|u - v\|_\infty \tag{6-11}$$

One could be tempted to find the bounds or the system by using (6-11), but two options are not possible: using both $A_0(p(k))$ and $A_0(p(k+1))$ on the places of A is not possible since A should be constant and $A_0(p(k))$ and $A_0(p(k+1))$ very likely are not. A second option is to somehow define a bound on both $A_{0,\min}$ and $A_{0,\max}$, two matrices that can be defines by taking $\tau_{\mathrm{d}}(k) = \tilde{\tau}_{\min}$ and $\tau_{\mathrm{d}}(k) = \tau_{\mathrm{d},\max}$ with these limits representing the limits given in

(3-29). Since A_0^* is regular (i.e., at least one finite element in every row) (6-4), we found find that:

$$\begin{aligned} \|(A_{0,\min}^* \otimes b(k)) - (A_{0,\min}^* \otimes b(k-1))\|_\infty \leq \|b(k) - b(k-1)\|_\infty \\ \|(A_{0,\max}^* \otimes b(k)) - (A_{0,\max}^* \otimes b(k-1))\|_\infty \leq \|b(k) - b(k-1)\|_\infty \end{aligned} \tag{6-12}$$

where $b(\cdot) = A_1 \otimes x(\cdot)$ (6-6). However, this is equally invalid since $b(k)$ and $b(k-1)$ will be n-by-1 vectors in $\mathbb{R}^n_{\max}$ (due to A_1), which contradicts the fact that they should be finite vectors in $\mathbb{R}^n$.

However, a bound can be put on the eigenvalue λ. If the minimum allowable dwell time is $\tilde{\tau}_{\min}$ and the maximum allowable dwell time is $\tau_{\mathrm{d,max}}$, then we find that:

$$\lambda_{\min} = \bigoplus_{j=1}^{J} (\tau_{\mathrm{h},j} \otimes \tilde{\tau}_{\min}) \tag{6-13}$$

$$\lambda_{\max} = \bigoplus_{j=1}^{J} (\tau_{\mathrm{h},j} \otimes \tau_{\mathrm{d,max}}) \tag{6-14}$$

and thus that:

$$\lambda_{\min} \leq \lambda(\mathcal{A}(k)) \leq \lambda_{\max} \tag{6-15}$$

which means that the eigenvalue of the system is upper- and lower bounded when both a maximum and minimum dwell time are specified by the operator.

The problem with such a statement is the bound itself: how should an operator (or engineer) set a bound? In order to answer such a question, it is important to realize that the max-plus system described in this thesis is solely lower bounded but not upper bounded by the assumptions which were done a little before (3-40). Therefore, in theory, when train $k-1$ would break down, train k can not continue its way and the delay would go to ∞. Also, when too many passengers would be entering the stations for too long, dwell times would become very large and therefore, it might feel like the system stops working as well. Since these situations are not common, it has not been modelled and therefore, a general advice would be to not use the max-plus model that has been proposed in such a situation.

The following section will therefore focus more on the suggestion of when it is useful to use this model.

6-2 Bounds on the max-plus system

As mentioned before, it is clear that the max-plus system as described in this paper in not upper bounded, because of the linear parameter varying nature of the $\mathcal{A}(p)$ (or $A_0(p)$) matrix.

To show this, we will try to put an upper bound on the maximum value of the difference between train $k+1$ and k, as following:

$$\|x(k+1) - x(k)\|_\infty \leq U \tag{6-16}$$

where U is a scalar upper bound that has to be found, in order to maintain a stability margin. Applying the reverse triangle inequality theorem results into:

$$\|x(k+1) - x(k)\|_\infty \geq |\|x(k+1)\|_\infty - \|x(k)\|_\infty| \tag{6-17}$$

Since by definition, train $k+1$ will always be later than train k, we know that $\|x(k+1)\|_\infty - \|x(k)\|_\infty$ is always positive so we can neglect the absolute value-operator. So we find that:

$$\begin{aligned}
\|x(k+1)\|_\infty - \|x(k)\|_\infty \leq \|x(k+1) - x(k)\|_\infty &\leq U \\
\|x(k+1)\|_\infty - \|x(k)\|_\infty &\leq U \\
\|\mathcal{A}(p(k+1)) \otimes x(k)\|_\infty - \|x(k)\|_\infty &\leq U \\
\left\| \begin{matrix} \max(\mathcal{A}_{1,J+1}(p(k+1)) + x_{J+1}(k), ..., \mathcal{A}_{1,2J}(p(k+1)) + x_{2J}(k)) \\ \max(\mathcal{A}_{2,J+1}(p(k+1)) + x_{J+1}(k), ..., \mathcal{A}_{2,2J}(p(k+1)) + x_{2J}(k)) \\ \vdots \\ \max(\mathcal{A}_{J,J+1}(p(k+1)) + x_{J+1}(k), ..., \mathcal{A}_{J,2J}(p(k+1)) + x_{2J}(k)) \end{matrix} \right\|_\infty - \left\| \begin{matrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_{2J}(k) \end{matrix} \right\|_\infty &\leq U
\end{aligned} \tag{6-18}$$

Since the ∞-norm takes the maximum of all the elements in the supplied vector, and since we can presume that $x_{2J}(k)$ and $x_{2J}(k+1)$ are the largest components of these vectors, we can only look at the last row and write:

$$\begin{aligned}
\max(\mathcal{A}_{J,J+1}(p(k+1)) + x_{J+1}(k), ..., \mathcal{A}_{J,2J}(p(k+1)) + x_{2J}(k)) - x_{2J}(k) &\leq U \\
\max(\mathcal{A}_{J,J+1}(p(k+1)) + \underbrace{x_{J+1}(k) - x_{2J}(k)}_{\leq 0}, ..., \mathcal{A}_{J,2J}(p(k+1))) &\leq U
\end{aligned} \tag{6-19}$$

From (6-8), we know that the largest of the elements of $\mathcal{A}(p(k+1))$ is $\mathcal{A}_{J,J+1}(p(k+1))$, so if the bound U is defined for this element, it will be valid for all the others as well, also taking in mind that the elements of $x(k)$ will not make a difference in this. By taking a closer look at (6-8), we see that the expression for $\mathcal{A}_{J,J+1}(p(k+1))$ in general is:

$$\begin{aligned}
\tau_{\mathrm{h},1} + \sum_{j=1}^{J} \tau_{\mathrm{d},j}(k+1) + \sum_{j=1}^{J-1} \tau_{\mathrm{r,min},j} &\leq U \\
\sum_{j=1}^{J} \tau_{\mathrm{d},j}(k+1) &\leq \underbrace{U - \tau_{\mathrm{h},1} - \sum_{j=1}^{J-1} \tau_{\mathrm{r,min},j}}_{U^*} \\
\begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \boldsymbol{\tau}_{\mathrm{d},j}(k+1) &\leq U^*
\end{aligned} \tag{6-20}$$

If we take again that $\tau_{\mathrm{d},j}(k+1) = x_{j+J}(k+1) - x_j(k+1)$, i.e. as the difference between arrival and departure time, then we find that:

$$\begin{aligned}
\begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \boldsymbol{\tau}_{\mathrm{d},j}(k+1) &\leq U^* \\
\begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \begin{pmatrix} x_{J+1}(k+1) - x_1(k+1) \\ x_{J+2}(k+1) - x_2(k+1) \\ \vdots \\ x_{2J}(k+1) - x_J(k+1) \end{pmatrix} &\leq U^* \\
\begin{bmatrix} -1 & -1 & \cdots & -1 & \vert & 1 & 1 & \cdots & 1 \end{bmatrix} x(k+1) &\leq U^*
\end{aligned} \tag{6-21}$$

Clearly, this is were the user-defined bound kicks in: since the schedule $x(k+1)$ for train $k+1$ can delay up until ∞, a bound U^* can be defined to avoid that. See the example below. However, as another conclusion, as long as the schedule of train $k+1$ does not contain any elements $x_{1,\ldots,2J} = \infty$, then the max-plus system is bounded and will never become ∞ itself.

Example 6.1 *Imagine that for an $J = 5$-stop urban rail line, the maximum allowable dwell time is 200 seconds at every station. Furthermore, the headway time between two trains at the first station is defined to be $\tau_{h,1} = 240$ seconds, while the minimal travel time between all the stops is 60 seconds. Then the upper bound U can be found to be:*

$$\begin{aligned}
\begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \boldsymbol{\tau}_{\mathrm{d},j}(k+1) &\leq U^* \\
J \cdot \tau_{\mathrm{d,max}} &\leq U^* \\
U^* &\geq J \cdot \tau_{\mathrm{d,max}} = 5 \cdot 200 = 1000 \\
U = U^* + \tau_{\mathrm{h},1} + \sum_{j=1}^{J-1} \tau_{\mathrm{r,min},j} &\geq 1000 + 240 + 4 \cdot 60 = 1480
\end{aligned} \tag{6-22}$$

The upper bound U has to be at least 1480 or larger to guarantee that $x(k+1) - x(k)$ remains bounded. If $\|x(k+1) - x(k)\|_\infty > 1480$, the system might become unstable.

As a final note, one can use a user defined bound as an additional linear constraint for the optimization process. Observe (6-21), which concludes that:

$$\begin{bmatrix} -1 & -1 & \cdots & -1 & \vert & 1 & 1 & \cdots & 1 \end{bmatrix} x(k+1) \leq U^* \tag{6-23}$$

Of course, if the upper bound U^* is set on $x(k+1)$, it is also an upper bound on $x(k+2)$, $x(k+3)$, ..., $x(k+M-1)$. Therefore, we find that:

$$\begin{aligned}
\begin{bmatrix} -1 & -1 & \cdots & -1 & \vert & 1 & 1 & \cdots & 1 \end{bmatrix} x(k) &\leq U^* \\
\begin{bmatrix} -1 & -1 & \cdots & -1 & \vert & 1 & 1 & \cdots & 1 \end{bmatrix} x(k+1) &\leq U^* \\
&\vdots \\
\begin{bmatrix} -1 & -1 & \cdots & -1 & \vert & 1 & 1 & \cdots & 1 \end{bmatrix} x(k+M-1) &\leq U^*
\end{aligned}$$

and thus that:

$$\breve{S} \cdot \breve{x}(k) \leq \breve{U}^* \tag{6-24}$$

where $\breve{S} \in \mathbb{R}^{1 \times 2 \cdot J \cdot M}$ is a vector filled with only -1 and 1, $\breve{x}(k)$ is the same as in all the previous definitions and $\breve{U}^* \in \mathbb{R}^{M \times 1}$ is a vector filled with the user defined upper bound $\breve{U}^*$.

Chapter 7

Conclusions and future work

In chapter 1 and section 2-3, the goals of this thesis were stated. The main question was whether it is possible to use max-plus algebra in some kind of linear parameter varying context. Secondly, this thesis aims to contribute to the field of max-plus (linear parameter varying) systems by modelling an urban railway line in max-plus algebra. Last but not least, an attempt has been done to analyse the potential stability criteria for MP-LPV systems. The main conclusion will be discussed in this chapter, after which some future work will be suggested.

7-1 Conclusions

7-1-1 Max-plus Linear Parameter Varying systems

This thesis introduces a new kind of max-plus systems, namely the Max-Plus Linear Parameter Varying (MP-LPV) systems. Such systems can be expressed in max-plus algebra, but one or multiple systems matrices contain dependencies on current and/or previous states. These state-dependent matrices make the system non-linear in max-plus algebra, but it has been shown that the dependency on states can be modelled as a linear relation p. The linear relation p in this thesis was considered to be a linear relation in conventional algebra rather than in max-plus algebra, due to scaling and subtraction of the states (3-41). Furthermore, only the $A = A(q)$ matrix was state-dependent (3-42); the rest of the state matrices could be taken as constant matrices.

By applying the iterative procedure of sections 3-6 and 3-7, an MP-LPV could be implemented in MATLAB both as one-cycle-at-once and multiple-cycles-at-once systems. An important assumption here is that the dwell time of train k is dependent on the arrival time of train k and the departure time of train $k-1$, instead of on the departure times of both trains k and $k-1$. As long as there are not too much passengers per second λ, this is not a problem. The assumption will not affect the results that much. But when λ increases, there exists a possibility that - in the case of dependency on the departure times - the flow of incoming passengers is too large, and thus that it is not possible for a train k to depart. In this case, the iterative process might not converge and thus not compute the schedule $x(k)$ (or $\breve{x}(k)$). Therefore, this will be mentioned as future research.

The iterative procedure can be seen as some kind of 'free run'-principle. The trains run whenever they are allowed to, mostly because of the very definition in max-plus algebra that the left side of equation (3-42) is *equal* to the right side: they are forced to leave at the maximum value that is found on the right side. This can be 'solved' by modelling the system as a 'relaxed' problem.

7-1-2 Optimization of an MP-LPV system

In chapter 4, the MP-LPV system that was found in chapter 3 was rewritten to a set of linear inequalities. This directly implies that the MP-LPV could be modelled as a linear constrained problem which is a lot easier to solve for optimization techniques. Furthermore, by modelling the MP-LPV as a relaxed problem, trains are allowed to be delayed which can be used as a control variable to work around disturbances.

Therefore, a linear, a quadratic and a non-linear programming problem were defined to compute the steering control variable $\breve{u}(k)$ based on minimizing the schedule $\breve{x}(k)$ for M trains ahead, and on minimizing total passenger travel time. The first optimization problem was a linear programming problem, while the latter one could be solved as a quadratic programming problem and non-linear programming problem. It could be seen that, in the case of no disturbance $\breve{d}(k)$, these optimization problems gave the same results as the 'free run'-principle. But once there is a disturbance somewhere in the planning horizon M, the LP, QP and NLP frameworks work around the disturbance to avoid that the headway time between two trains grows out of bound (section 4-1).

Furthermore, it could be observed that minimizing total passenger travel time resulted in an indefinite matrix $\breve{H}$ in (4-49). Since it is not positive definite, the QP problem is not convex and - even though the MATLAB-command 'quadprog' still computes - a global optimum can not be guaranteed. It was shown that the non-linear programming problem finds similar results as the QP problem when the latter one is 'made convex', i.e. by making $\breve{H}$ positive definite by adding the largest negative eigenvalue to the diagonal of the matrix. This indicates a global optimum, but this is not proven.

Also, the non-linear programming problem can be used to compute the cases that the QP problem can not compute, which happens when the amount of arriving passengers per second at the stations is made too large. However, it is relatively slow compared to LP and QP, which does not make it the most preferred optimization strategy.

7-1-3 Max-plus analysis on MP-LPV systems

Last but not least, an analysis has been done on the stability of MP-LPV systems. In conventional algebra, stability can be guaranteed by having bounds on the state-dependent system matrices. But in max-plus algebra, these bounds are hard to define, as could be seen in chapter 5. A user-defined bound can be implemented as a linear inequality constraint in a programming problem (6-24), but it is not a system-defined bound that guarantees stability.

As a consequence of a large disturbance or a very high amount of passengers, an MP-LPV can grow out of bound which results into a vector $\breve{x}(k)$ with elements j that are growing to infinity, i.e. $\breve{x}_j(k) \to \infty$. When this happens, we find that the system is not upper-bounded and goes to ∞ as well. More research on this topic is still necessary.

7-2 Future work

From both the conclusion and the fact that this is an exploratory study, it can be concluded that there is still a lot to be improved and discovered on this new class of max-plus algebraic systems, the so-called Max-Plus Linear Parameter Varying systems. These points of improvement or further investigation will be summarized in this section.

1. Throughout this paper, some mathematical proofs have been provided to show the line of thought in this work. By proofing certain conditions, it has been motivated that there was a necessity for controlling disturbed urban railway lines because they might evolve to grow out of bounds. Furthermore, it has been shown that the MP-LPV 'free run' schedule is supposed to be the quickest scheduling solution possible, since there is no quicker $\tilde{x}(k)$ when $\gamma > 0$. However, it has not been proved that it is also the quickest solution when not all but only some entries of $\hat{x}(k)$ are sped up. It was assumed that all entries of $\gamma > 0$. But it seems logical that this would also be the case, since the equality sign in the MP-LPV 'free run' is supposed to guarantee the quickest arrival and departure times as possible. But a mathematical proof for this is still missing.

2. Furthermore, the mathematical analysis of the stability of MP-LPV systems requires more research. It has only been shown that a finite upper bound guarantees stability, but this is not proven to be a system property.

3. More research is necessary when the assumption that the dwell time can be computed as the difference between the arrival time of train k and departure time of train $k-1$, is withdrawn. This assumption can be done when the amount of passengers that arrive between the arrival time of train k and departure time of train $k-1$ is relatively very large compared to the amount of arriving passengers during the dwell time itself. But, when this is not the case, then it can be shown that there is a turnover-point when too many passengers arrive per second. When this happens, trains get delayed relatively more seconds than they can handle. A solution to this is a forced maximum dwell time, but this is hard to model in max-plus algebra. Wang, Ning, et al. (2015) have incorporated such a maximum dwell time in their conventional algebra model of an urban train network.

4. Another reason to incorporate a maximum dwell time would be that trains do not have unlimited capacity, as has been assumed in this paper. But introducing a maximum capacity introduces a min-operation in the definition of the dwell time, which directly influences the parameter p in a way that has not been investigated for this thesis. The

effects of such a constraint on the MP-LPV system is therefore future work.

5. In this work, the MP-LPV system is considered to only contain a matrix $A(p)$ that is dependent on the parameter p. For future work, it might be interesting to also find an application where the matrix $B = B(p)$, or even where the matrices $C(p)$ and $D(p)$ exist and are state dependent.

6. The computation time of a 10-stops case with a computation horizon of $M = 10$ trains has been shown to be low, i.e. around a quarter of a second for the LP and QP problems. No fair comparison with other papers could be done, since these papers often (also) consider energy as an (extra) objective. But due to the non-linear optimizations in most of these papers, the prediction horizon is often shorter than the computation time which means that these models do not compute quick enough. It seems that writing out the max-plus system in this paper did result in a very quick optimization strategy, but more research is necessary if it is actually better than results in conventional algebra.

7. It might be interesting to take a better look at the optimization strategy for the control of an urban railway line. In this paper, only one M-steps ahead computation of the schedule $\breve{x}(k)$ is done. But in reality, this has to be repeated every now and then in order to update $\breve{x}(k)$. One way to do this is by computing $\breve{x}(k + l)$ every time train $x(k + l - 1)$ has finished, with $l = N - M$ and N as the total amount of trains on a day. But as can be seen from Figure 5-1, whenever a train is finished, the next trains have already started their journeys. So a framework has to be implemented that distinguishes between arrival and departure times that have already taken place, and arrival and departure times that still can be controlled. Such a framework has not yet been implemented for this paper, but this is more a practical implementation for urban railway networks rather than an implementation for MP-LPV systems in general.

8. Last but not least, the disturbances in this paper might not be very realistic. It is hard to know beforehand when a train will have a problem, but more importantly when exactly a station is more crowded than it would normally be. For example, because a football-game just ended and all visitors suddenly enter the train-station. Lately, urban railway network-operators have access to many data, like the check-in/check-out data from public transport cards or for example by cameras on the platforms. The disturbance as modelled in this paper is therefore not considered unrealistic, but rather very ambitious. It would be interesting to test the model presented in this paper with actual real data supplied by urban railway operators. This in order to see if the MP-LPV model handles real-time situations as well.

References

Baccelli, F., Cohen, G., Olsder, G. J., & Quadrat, J.-P. (1992). *Synchronization and linearity.*

De Schutter, B., & van den Boom, T. (2001). Model predictive control for max-plus-linear discrete event systems. *Automatica*, *37*(7), 1049 - 1056. Retrieved from http://www.sciencedirect.com/science/article/pii/S0005109801000541 doi: https://doi.org/10.1016/S0005-1098(01)00054-1

De Schutter, B., & van den Boom, T. (2008, May). Max-plus algebra and max-plus linear discrete event systems: An introduction. In *Proceedings of the 9th international workshop on discrete event systems (wodes'08)* (pp. 36–42). Göteborg, Sweden.

Farlow, K. G. (2009). *Max-plus algebra* (Unpublished doctoral dissertation). Virginia Tech.

Gáspár, P., Szabó, Z., Bokor, J., & Németh, B. (2017). Modeling of lpv systems. In *Robust control design for active driver assistance systems: A linear-parameter-varying approach* (pp. 11–70). Cham: Springer International Publishing. Retrieved from https://doi.org/10.1007/978-3-319-46126-7_2 doi: 10.1007/978-3-319-46126-7_2

Heidergott, B., Olsder, G. J., & Van der Woude, J. (2014). *Max plus at work: modeling and analysis of synchronized systems: a course on max-plus algebra and its applications.* Princeton University Press.

Kersbergen, B., Rudan, J., van den Boom, T., & De Schutter, B. (2016). Towards railway traffic management using switching max-plus-linear systems. *Discrete Event Dynamic Systems*, *26*(2), 183–223.

Li, X., & Lo, H. K. (2014). An energy-efficient scheduling and speed control approach for metro rail operations. *Transportation Research Part B: Methodological*, *64*, 73 - 89. Retrieved from http://www.sciencedirect.com/science/article/pii/S0191261514000484 doi: https://doi.org/10.1016/j.trb.2014.03.006

Mohammadpour, J., & Scherer, C. (2012). *Control of linear parameter varying systems with applications.* Springer New York. Retrieved from https://books.google.nl/books?id=Gvr6yurJI9IC

Niu, H., Zhou, X., & Gao, R. (2015). Train scheduling for minimizing passenger waiting time with time-dependent demand and skip-stop patterns: Nonlinear integer programming models with linear constraints. *Transportation Research Part B: Methodological*, *76*, 117 - 135. Retrieved from http://www.sciencedirect.com/science/article/pii/S0191261515000478 doi: https://doi.org/10.1016/j.trb.2015.03.004

Sun, L., Jin, J. G., Lee, D.-H., Axhausen, K. W., & Erath, A. (2014). Demand-driven timetable design for metro services. *Transportation Research Part C: Emerging Technologies*, *46*, 284 - 299. Retrieved from http://www.sciencedirect.com/science/article/pii/S0968090X1400182X doi: https://doi.org/10.1016/j.trc.2014.06.003

Tóth, R., C Willems, J., Heuberger, P., & Van den Hof, P. (2009, 01). A behavioral approach to lpv systems. *2009 European Control Conference, ECC 2009*.

van den Boom, T., & De Schutter, B. (2006). Modelling and control of discrete event systems using switching max-plus-linear systems. *Control Engineering Practice*, *14*(10), 1199 - 1211. Retrieved from http://www.sciencedirect.com/science/article/pii/S0967066106000232

(The Seventh Workshop On Discrete Event Systems (WODES2004)) doi: https://doi.org/10.1016/j.conengprac.2006.02.006

van den Boom, T., & De Schutter, B. (2011). A stabilizing model predictive controller for uncertain max-plus-linear systems and uncertain switching max-plus-linear systems. *IFAC Proceedings Volumes*, *44*(1), 8663 - 8668. Retrieved from http://www.sciencedirect.com/science/article/pii/S1474667016450019 (18th IFAC World Congress) doi: https://doi.org/10.3182/20110828-6-IT-1002.00872

Wang, Y., Ning, B., van den Boom, T., & Schutter, B. D. (2015). *Optimal trajectory planning and train scheduling for urban rail transit systems*. Springer.

Wang, Y., Tang, T., Ning, B., van den Boom, T. J., & Schutter, B. D. (2015). Passenger-demands-oriented train scheduling for an urban rail transit network. *Transportation Research Part C: Emerging Technologies*, *60*, 1 - 23. Retrieved from http://www.sciencedirect.com/science/article/pii/S0968090X15002624 doi: https://doi.org/10.1016/j.trc.2015.07.012

Yang, X., Chen, A., Ning, B., & Tang, T. (2017). Bi-objective programming approach for solving the metro timetable optimization problem with dwell time uncertainty. *Transportation Research Part E: Logistics and Transportation Review*, *97*, 22 - 37. Retrieved from http://www.sciencedirect.com/science/article/pii/S1366554515301058 doi: https://doi.org/10.1016/j.tre.2016.10.012

Appendix A

The back of the thesis

A-1 Modelling of a railway network in max-plus algebra

In section 3-2, the constraints of the urban railway line were defined. In (Kersbergen et al., 2016), where this section is based on, the constraints were modelled slightly different. In the first place because a train network consists of multiple lines and contains scheduled arrival and departure times as well, but more importantly because this model works with a cycle counter k instead of a train counter k. Underneath are the constraints as modelled in (Kersbergen et al., 2016), followed by how these constraints are modelled into an MPL.

A-1-1 Running time constraints

A running time constraint simply deals with the time between arrival and departure of a train run i. This time is then called the *running time*, denoted as $\tau_{\mathrm{r},i}(k)$. Mathematically, we can write:

$$a_i(k) \geq d_i(k) + \tau_{\mathrm{r},i}(k) \tag{A-1}$$

In other words; a train can not arrive at the next stop before this minimal traversing time.

A-1-2 Dwell time constraints

The dwell time constraint (also called continuity constraint) is an inequality that tries to connect different train runs to each other. When the train run from station one to station two is called p_i and the train run from station two to station three is called i, then the underlying mathematical relation is:

$$d_i(k) \geq a_{p_i}(k - \mu_{i,p_i}) + \tau_{\mathrm{d},i,p_i}(k) \tag{A-2}$$

where $\tau_{\mathrm{d},i,p_i}(k)$ is the so-called *dwell time*. If train run p_i continues as i in the same cycle k, then $\mu_{i,p_i} = 0$. If train run p_i previously was in cycle $k - \alpha$, but continues as train run i in cycle k, then $\mu_{i,p_i} = \alpha$. It can be noted that for a railway network, μ_{i,p_i} is typically only 0, -1 or 1, i.e. only the current, previous and future cycles are considered.

A-1-3 Timetable constraints

A timetable constraint is simply a guarantee that trains cannot depart before the actual departure time according to a predefined timetable. Sometimes, trains are also not allowed to enter a station before the timetable says they can. Mathematically, we can denote:

$$d_i(k) \geq r_{\mathrm{d},i}(k) \tag{A-3}$$
$$a_i(k) \geq r_{\mathrm{a},i}(k) \tag{A-4}$$

where $r_{\mathrm{a},i}(k)$ and $r_{\mathrm{d},i}(k)$ are the arrival and departure time according to a predefined timetable, respectively.

A-1-4 Headway constraints

To keep a sufficient (time-)distance between two trains, headway constraints are necessary. Mathematically, we can write:

$$d_i(k) \geq d_l(k - \mu_{i,l}) + \tau_{\mathrm{h,d},i,l}(k) \tag{A-5}$$
$$a_i(k) \geq a_l(k - \mu_{i,l}) + \tau_{\mathrm{h,a},i,l}(k) \tag{A-6}$$

Here, $\mu_{i,l}$ is defined in a similar way as it was for the continuity constraints, and $\tau_{\mathrm{h,d},i,l}$ and $\tau_{\mathrm{h,a},i,l}$ are the departure and arrival *headway times* respectively, the times between two train runs.

A-1-5 Coupling constraints

In train networks, it can occur that two trains have to be coupled to each other. If train i and o_i have to be coupled, we can denote:

$$d_i(k) \geq d_{o_i}(k) \tag{A-7}$$
$$d_{o_i}(k) \geq d_i(k) \tag{A-8}$$
$$a_i(k) \geq a_{o_i}(k) \tag{A-9}$$
$$a_{o_i}(k) \geq a_i(k) \tag{A-10}$$

which basically reads as $d_i(k) = d_{o_i}(k)$ and $a_i(k) = a_{o_i}(k)$.

A-1-6 Connection constraints

Another important constraint for railway networks is the connection constraint, which guarantees that passengers are able to transfer from one train to another. In practice, this means that one train cannot leave if the other train did not arrive yet. Besides that, it is necessary to have some extra *connection time* $\tau_{\mathrm{c},i,e}(k)$ for the passengers to change trains. Mathematically, we denote:

$$d_i(k) \geq a_e(k - \mu_{i,e}) + \tau_{\mathrm{c},i,e}(k) \tag{A-11}$$

where e and i are different train runs and $\mu_{i,e}$ is defined similarly as for the continuity and headway constraints.

A-1-7 Max-plus linear modelling

Now that (A-1) up until (A-11) represent all the constraints of the train network, the max-plus linear system can be formulated. This can be done by writing all inequalities that represent the arrival and departure times of train run i into two different expressions. As all equations are "equal or greater than", the actual departure and arrival times are the maximums of all these expressions, as following:

$$\begin{aligned} d_i(k) = \max\Bigg(&(a_{p_i}(k - \mu_{i,p_i}) + \tau_{\mathrm{d},i,p_i}(k),\ \max_{l \in \mathcal{H}_i}\big(d_l(k - \mu_{i,l}) + \tau_{\mathrm{h,d},i,l}(k)\big), \\ &\max_{m \in \mathcal{S}_i}\big(a_m(k - \mu_{i,m}) + \tau_{\mathrm{s},i,m}(k)\big),\ \max_{e \in \mathcal{C}_i}\big(a_e(k - \mu_{i,e}) + \tau_{\mathrm{c},i,e}(k)\big),\ d_{o_i}(k),\ r_{\mathrm{d},i}(k)\Bigg) \end{aligned} \tag{A-12}$$

$$a_i(k) = \max\Bigg(\max_{l \in \mathcal{H}_i}\big(a_l(k - \mu_{i,l}) + \tau_{\mathrm{h,a},i,l}(k)\big),\ d_i(k) + \tau_{\mathrm{r},i}(k),\ a_{o_i}(k),\ r_{\mathrm{a},i}(k)\Bigg) \tag{A-13}$$

From the very definition of max-plus algebra, we can replace the max-expressions with $\oplus$ and the summation-expressions with $\otimes$, which will lead to the following expressions:

$$\begin{aligned} d_i(k) = &(a_{p_i}(k - \mu_{i,p_i}) \otimes \tau_{\mathrm{d},i,p_i}(k) \oplus \bigoplus_{l \in \mathcal{H}_i}\big(d_l(k - \mu_{i,l}) \otimes \tau_{\mathrm{h,d},i,l}(k)\big) \oplus \\ &\bigoplus_{m \in \mathcal{S}_i}\big(a_m(k - \mu_{i,m}) \otimes \tau_{\mathrm{s},i,m}(k)\big) \oplus \bigoplus_{e \in \mathcal{C}_i}\big(a_e(k - \mu_{i,e}) \otimes \tau_{\mathrm{c},i,e}(k)\big) \oplus d_{o_i}(k) \oplus r_{\mathrm{d},i}(k) \end{aligned} \tag{A-14}$$

$$a_i(k) = \max\Bigg(\bigoplus_{l \in \mathcal{H}_i}\big(a_l(k - \mu_{i,l}) \otimes \tau_{\mathrm{h,a},i,l}(k)\big) \oplus \big(d_i(k) \otimes \tau_{\mathrm{r},i}(k)\big) \oplus a_{o_i}(k) \oplus r_{\mathrm{a},i}(k)\Bigg) \tag{A-15}$$

By defining a network with n trains, a state vector $x(k)$ and timetable vector $r(k)$ can be defined as

$$x(k) = \begin{pmatrix} d_1(k) \\ d_2(k) \\ \vdots \\ d_n(k) \\ \hline a_1(k) \\ a_2(k) \\ \vdots \\ a_n(k) \end{pmatrix} \in \mathbb{R}_\epsilon^{2n} \qquad r(k) = \begin{pmatrix} r_{\mathrm{d},1}(k) \\ r_{\mathrm{d},2}(k) \\ \vdots \\ r_{\mathrm{d},n}(k) \\ \hline r_{\mathrm{a},1}(k) \\ r_{\mathrm{a},2}(k) \\ \vdots \\ r_{\mathrm{a},n}(k) \end{pmatrix} \in \mathbb{R}_\epsilon^{2n} \tag{A-16}$$

then we can finally find the max-plus linear model to be

$$x(k) = r(k) \oplus \bigoplus_{\mu=0}^{\mu_{\max}} A_\mu(k) \otimes x(k-\mu) \tag{A-17}$$

with $A_\mu(k) \in \mathbb{R}_\epsilon^{2n\times 2n}$ for $\mu = 0, 1, \cdots, \mu_{\max}$, where $\mu_{\max} = \max_{i,j} \mu_{i,j}$. The matrix $A_\mu(k)$ now contains all the constraints and relates the arrival and departure times of the current and previous cycles to those of the current cycle.