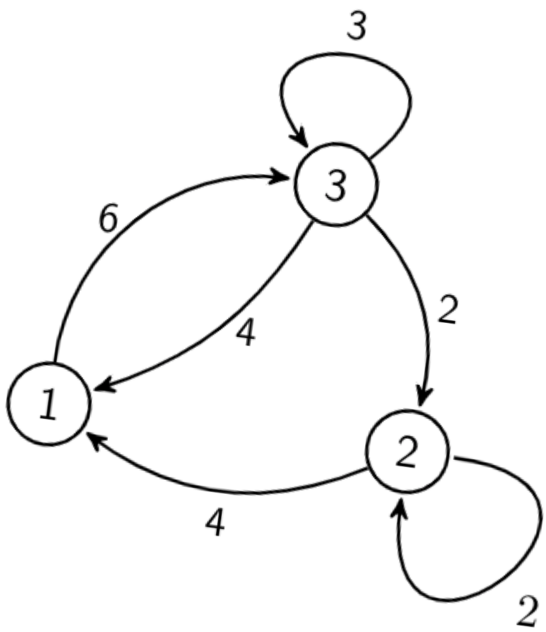
Max-plus linear parameter varying systems

A framework & solvability

M. Abdelmoumni

Master of Science Thesis

$$A = \begin{pmatrix} \varepsilon & 4 & 4 \\ \varepsilon & 2 & 2 \\ 6 & \varepsilon & 3 \end{pmatrix}$$



$$A \oplus B = \begin{pmatrix} \max(e,-1) & \max(2,3) \\ \max(\varepsilon,1) & \max(3,\varepsilon) \end{pmatrix} = \begin{pmatrix} e & 3 \\ 1 & 3 \end{pmatrix}$$

$$A \otimes B = \begin{pmatrix} \max(e-1,2+1) & \max(e+3,2+\varepsilon) \\ \max(\varepsilon-1,3+1) & \max(\varepsilon+3,3+\varepsilon) \end{pmatrix} = \begin{pmatrix} 3 & 3 \\ 4 & \varepsilon \end{pmatrix}$$

$$x(k) = A_0(p(k)) \otimes x(k) \oplus A_1 \otimes x(k-1) \oplus u(k)$$

TU Delft Delft University of Technology

Delft Center for Systems and Control

Max-plus linear parameter varying systems

A framework & solvability

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

M. Abdelmoumni

January 11, 2021

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Delft University of Technology
Department of
Delft Center for Systems and Control (DCSC)

The undersigned hereby certify that they have read and recommend to the Faculty of Mechanical, Maritime and Materials Engineering (3mE) for acceptance a thesis entitled

Max-plus linear parameter varying systems

by

M. Abdelmoumni

in partial fulfillment of the requirements for the degree of

Master of Science Systems and Control

Dated: January 11, 2021

Supervisor(s): ______________________
dr. ir. T.J.J. van den Boom

____
dr. ir. J.W. van der Woude

____
Ir. A. Gupta

Reader(s): ______________________
dr. P. Mohajerin Esfahani

Abstract

In this research, we introduce and discuss an extension of max-plus linear systems, called max-plus linear parameter varying systems. In this extension, uncertainty is modelled as a varying parameter, making it possible to leverage non-linearities and non-max-plus terms within the model. Issues arise when the varying parameter is dependent on previous and current states, so-called implicit max-plus linear parameter varying systems. Implicit max-plus linear parameter varying systems are not guaranteed to have a solution. In this work, we introduce some analytical and graph-theoretical methods to analyze the solvability of implicit max-plus linear parameter varying systems. Using these methods, we show that it is not possible to conclude structural solvability for general implicit max-plus linear parameter varying systems. Although, we show that, under mild assumptions, there exist conditions for which the implicit system is solvable. These conditions are explained and applied on an implicit max-plus linear parameter varying model with a state-dependent varying parameter. We illustrate our results with an urban railway network with passenger-dependent dwell time.

Table of Contents

Acknowledgements **xi**

1 Introduction **1**

1.1 Background 1

1.2 Problem statement 2

1.3 Contribution 2

1.4 Outline 3

2 Max-plus algebra **5**

2.1 Max-plus algebra basics 5

2.2 Vectors and matrices 7

2.3 Spectral theory 10

2.4 Max-plus linear systems 15

2.5 Max-plus linear systems a first model: The urban railway network 17

2.6 Convex theory 19

2.7 MMPS and PWA 20

2.8 Summary 21

3 Max-plus linear parameter varying systems **23**

3.1 Linear parameter varying systems 23

3.2 General MP-LPV framework 24

3.2.1 MP-LPV with varying parameter in conventional algebra 26

3.2.2 MP-LPV with varying parameter in max-plus algebra 29

3.3 Relation MP-LPV with MMPS functions 31

3.3.1 Proposition 1: MMPS to MPS 31

3.3.2 Proposition 2: MP-LPV to MPS 32

3.3.3 Proposition 3: MPS to MP-LPV 33

3.4 Summary 35

4 Solvability: Graph theoretical approach **37**
4.1 Structural solvability in representation graphs . . . 37
4.2 Structural solvability of MP-LPV systems . . . 40
4.2.1 Structural solvability: Explicit case . . . 41
4.2.2 Structural solvability: Implicit case . . . 43
4.3 Summary . . . 47

5 Solvability: Analytical approach **49**
5.1 Preliminaries analytical approach . . . 49
5.2 Case 1: Varying parameter in matrix A_0 . . . 51
5.2.1 Intermezzo solving the equation: $y = \max(\alpha y + \beta, l_{\max})$ . . . 52
5.3 Case 2: Varying parameter in both matrix A_0, and A_1 . . . 54
5.3.1 Intermezzo solving the equation: $y = \max(\alpha_1 y + \beta_1, \alpha_2 y + \beta_2)$ . . . 55
5.4 Summary . . . 56

6 A first MP-LPV system: An urban railway model with passenger dependent dwell time **59**
6.1 Remodelling urban railway network . . . 59
6.2 Variable dwell time . . . 60
6.3 The max-plus linear parameter varying model . . . 62
6.4 Rewriting the model equation . . . 62
6.5 Summary . . . 65

7 Case study **67**
7.1 Model constants . . . 67
7.2 Structural solvability of the urban railway network . . . 69
7.3 Analytical solvability of the urban railway network . . . 70
7.4 Numerical issue . . . 71
7.5 Summary . . . 72

8 Conclusion and future work **73**
8.1 Conclusions . . . 73
8.2 Future work . . . 74
8.2.1 Stability . . . 74
8.2.2 Controllability . . . 74
8.2.3 Relation of p-time event graphs with MP-LPV . . . 74
8.2.4 Possible application . . . 75

A Simulation of the urban railway network **77**
A.1 Model constants . . . 77
A.2 Numerical problem: part 1 . . . 78
A.3 Numerical problem: part 2 . . . 79

B Future work **81**
B.1 Relation MP-LPV and PTEG 81
B.2 New system: Production system with cooling down component 83

Bibliography **85**

Glossary **89**
List of Acronyms 89

List of Figures

2.1 Communication graph of matrix A as proposed in Example 2.3.1 11

2.2 Communication graph of matrix A as proposed in Example 2.3.3 14

2.3 A simplified illustration of the urban railway model. 17

3.1 When using the LPV framework we can extract the scheduling variable p out of the system. When we extract this variable from the dynamics of the system we are left with a remaining relation between inputs and outputs that is linear. . . . 24

3.2 Model classes with the proposition that states the relation 31

4.1 Representation graph of system of equations proposed in Example 4.1.1 38

4.2 Representation graph of system of equations proposed in Example 4.1.2 39

4.3 Representation graphs of different explicit cases. In Figure (e), the problematic self-loops have been drawn in red. 42

4.4 Representation graph: implicit I without varying function 44

4.5 Representation graph: implicit II with varying function but no self-loops 45

4.6 Representation graph: implicit III, with varying function and self-loops 46

4.7 Representation graph: implicit IV, with varying function and self-loops 46

5.1 The four possibilities are plotted with some appropriate values of $\alpha, \beta \in \mathbb{R}$. It can be seen that the number of solutions is dependent on the value of the variables α, β as proposed in Eq. (5.22). 53

5.2 Plots of the two examples. We are trying to find possible intersection points with line $y = x$. We can see that line y_2, with $\alpha < 0$ has an intersection with $y = x$ at $x = 8$. For line y_1, where $\alpha > 0$, we see that there is no intersection with y. . . 54

5.3 The three cases proposed in Eq. (5.34) & (5.35) are plotted with some appropriate values of $\alpha, \beta \in \mathbb{R}$. It can be seen that the number of solutions is dependent on the value of the variables α, β as proposed in Eq. (5.34) & (5.35). 56

5.4 The intersections of y_1 and y_2 with $y = x$ as proposed in the example. 57

6.1 The model of two stations of the urban railway network including its variables. Note image is partly from [1]. 60

7.1 Free run simulation of the urban railway network for different values of α_{in} . . . 68
7.2 Representation graph of the urban railway network. 69
7.3 Figure with plots of both equations. Notice how the the blue lines rotate around the line $y = x$. 72

B.1 PTEG with interdependent residence durations (between p_2 and p_4) 81
B.2 Production system with three machines 83
B.3 Production system with three machines, but now with two possible modes. . . . 84

List of Tables

2.1 Properties of max-plus algebra and their analogies. 7

4.1 Summary of all findings on the explicit cases 42
4.2 Summary of all findings on the implicit cases 47

7.1 Comparison of analytically computed departure times $x_{i,ana}$ and departure times from simulation $x_{i,sim}$for different values of α_{in} 70

A.1 Values in simulation of passenger rates per station and minimal running times per station 77
A.2 Other constants 78
A.3 Ratios of alighting passengers 78

Acknowledgements

It finally comes to an end. The period I would describe as most difficult and most loved at the same time. Many things have happened and I have learned so many things and grown as a person in this period. For this, I would like to thank my three supervisors: Abhimanyu, Ton and Jacob.

Abhimanyu was my daily supervisor and I would like to thank him for his patience and willingness to answer my questions (even the stupid ones). I have learned many things from him and without him this thesis would not be possible. I would like to thank Ton for his always inspiring enthusiasm on this topic. While diligently explaining difficult subjects, he was always able to motivate and guide me in the right directions. I would like to thank Jacob for joining us in this research and giving us his view on the topic. His knowledge and concise feedback have helped me a lot in writing my thesis.

Also, I would like to thank all my friends with whom I have studied and discussed many things. Thank you for this wonderful period in my life. A special thanks to my dear friend Balaji. Thank you for reading the thesis and your insights throughout my research. With your help, it was almost as if I had an extra supervisor.

Lastly, I would like to thank my family for supporting me unconditionally. Without their help, I would never be where I am now. Thank you, mother, for working hard to support me and thank you for never giving up on me. To finalize, I would like to dedicate this work to my uncle who is fighting for his life in the intensive care due to complications of COVID-19. It is my deepest wish that he will wake up soon.

Delft, University of Technology M. Abdelmoumni
January 11, 2021

Chapter 1

Introduction

1.1 Background

Studying a complex system yields the necessity of describing it in the language of mathematics. Describing systems as mathematical models has a lot of benefits, one of them is that we can process mathematical models in computers and perform numerical simulations to study real-world systems. Most mathematical models in research have been described in conventional algebra, but there are also other algebras of interest with their own benefits.

For this research, we are particularly interested in max-plus algebra, where the conventional operators are replaced by the maximum and addition operators [2]. This algebra can be used to describe complex discrete event systems (DESs) such as railway networks, manufacturing systems, logistic systems, and many more. Techniques to model, analyze and control DESs have been a subject of interest of research for quite some time. In conventional algebra, modelling DESs almost always leads to a nonlinear system description. In some cases translating this description into max-plus algebra results into a description that is linear in max-plus algebra. This subclass of linear DESs has two important characteristics that can be linked to the basic operators in max-plus algebra. The maximum operator coheres with *synchronization*, which means that as soon as an operation is finished a subsequent operation will start. Addition corresponds to the idea that the starting time plus the time the operation takes together is equal to the finishing time of that operation. In literature, it is often said that the subclass of discrete event systems with synchronization, but without concurrency or choice is called *max-plus linear (MPL) systems* [2], [3], [4]. It is possible, for example, to model railroad networks, production systems, and urban traffic systems as MPL systems [5], [6], [7].

Additionally, it is possible to extend the description of an MPL system by adding the possibility to switch between different modes, called a switching max-plus linear (SMPL) system [5], [8]. Examples of this kind of systems are when a huge plant with multiples of the production processes is modelled. When these processes are the same except for their order or the processing time, it is possible to model the plant as a switching system. The mode switching gives the possibility to describe the changing behaviour of the system. In more recent research another MPL system extension has been discussed. MPL systems with a linear parameter varying (LPV) structure called: max-plus linear parameter varying (MP-LPV) systems [9], [10], and [11]. LPV systems are a special case of non-linear systems, where the system matrices are dependent on a varying parameter [12], [13], [14]. The system matrices are linear in the parameter, but the varying parameter does not necessarily have to be linear. In the max-plus framework, the idea of having matrices that are dependent on the state is a fairly untouched field of research. However, in [11] an urban railway network with a varying dwell time, i.e. time a train spends on a platform such that passenger can get in and out of the train, is modelled. The dwell time itself is amongst other things dependent on the time a train arrives and departs from a platform. Hence, we have a model with a varying parameter that is a function of the state itself. This model can be described as an MP-LPV system. In this work, the model has been described such that the varying parameter is only a function of previous states. In this thesis, we will focus more on the cases where the varying parameter is also a function of the current states.

1.2 Problem statement

This work will be focused on two main parts. Since there is a paucity of literature on MP-LPV one part will be about creating a general framework that allows us to define any MP-LPV system:

Defining a general framework for MP-LPV systems.

To achieve this the varying parameter has to be looked at. How can we define the varying parameter? What forms are there in literature? How does this fit in the general model? These questions need to be answered to define the general framework. The other part will discuss the analysis of solvability of MP-LPV. It is not a given that every MP-LPV system is structurally solvable. How can we analyze solvability of MP-LPVs that are not structurally solvable. The main topic will be:

Analysis of solvability of MP-LPV systems by looking at the structure of the graphs and equations.

To achieve this, two side-steps need to be taken. First, we need to look at the structure of the systems by drawing the graphs and applying known theorems on them. Secondly, we need to rewrite MP-LPV systems into max-min-plus-scaling (MMPS) functions under assumptions that will be introduced in the thesis, to look at conditions that ensure solvability.

1.3 Contribution

This thesis contributes to the research on MP-LPV systems by four main points:

- A general framework for MP-LPV systems described in Section 3.2.
- Relations and proofs between MMPS and MP-LPV systems described in Section 3.3.
- The solvability analysis that has been done on MP-LPV systems in Chapter 4 & 5.

- The application of an implicit MP-LPV system with only affine terms, where the varying parameter is described in conventional algebra (the model of the urban railway network).

1.4 Outline

In Chapter 2 an introduction to max-plus algebra is given. Basic operators, elements, and vectors and matrices are defined. Furthermore, an overview of the spectral theory on matrices concerning max-plus algebra is given. This includes the definition of eigenvalues and eigenvectors in max-plus algebra. Lastly, MPL systems are introduced with an introductory example of an MPL system, namely an urban railway network. MP-LPV systems are introduced and defined in Chapter 3. Also, propositions that relate MP-LPV systems with MMPS are given in this chapter. In Chapter 4 a graph-theoretical approach to finding structural solvability of MP-LPV systems is discussed. Then in Chapter 4 solvability is approached from an analytical view. Conditions for the existence of a solution are found and proposed. In Chapter 6 the running example is extended such that it becomes an implicit MP-LPV system with affine relations. This model is simulated and analysed using the results of Chapter 4 and 5 in Chapter 7. Finally, in Chapter 8 all conclusions and recommendations for future work are given.

Chapter 2

Max-plus algebra

Max-plus algebra has been used thoroughly for the modelling of discrete event systems (DESs). In our case, we are particularly interested in the sub-class of max-plus linear (MPL) systems and its extensions. In this chapter, a summary of max-plus basics is given. We will start with important properties and operations. Furthermore, max-plus algebra will be related to graph theory. Then, a small introduction to MPL systems is given. After introducing MPL systems, the urban railway network is introduced in an MPL framework. We conclude the chapter with a brief introduction to convexity regarding max-plus algebra, piecewise-affine (PWA) systems, and max-min-plus-scaling (MMPS) systems.

2.1 Max-plus algebra basics

In this section, we introduce the basic concepts and definitions of max-plus algebra. The choice was made to follow and use the notation of [6]. Max-plus algebra is denoted by:

$$\mathcal{R}_{\varepsilon} = (\mathbb{R}_{\varepsilon}, \oplus, \otimes, \varepsilon, e). \tag{2.1}$$

In which $\varepsilon := -\infty$, $e := 0$, $\mathbb{R}_{\varepsilon}$ is the set $\mathbb{R} \cup \{\varepsilon\}$ and $\mathbb{R}$ represents the set of real numbers. The two neutral numbers ε and e are essential in the max-plus algebra, and are comparable to 0 and 1 in conventional algebra. Now to define the operators, following [6]:

Definition 2.1.1. We look at elements $a, b \in \mathbb{R}_{\varepsilon}$ and define the so-called "oplus" operator as:

$$a \oplus b := \max(a, b), \tag{2.2}$$

■

Definition 2.1.2. Then the so-called "otimes" operator will be defined as:

$$a \otimes b := a + b. \tag{2.3}$$

■

In Eq. (2.2) and (2.3) it has been shown how the max-plus operators translate into conventional algebra. We can also show the resemblance of the neutral numbers by the following operations

$$\varepsilon \oplus a = a \tag{2.4}$$

$$e \otimes a = a \tag{2.5}$$

In line with conventional algebra, where the multiplication operator has priority over the plus operator, the same order in which operations should happen is used. In short, this means that the "otimes" operator has priority.

Below a number of properties of max-plus algebra and links with conventional algebra are denoted.

- **Associativity**
  The order in which operations are performed does not matter if you keep the sequence of operands the same. This means that for all $a, b, c \in \mathbb{R}_\varepsilon$:

  $$a \oplus (b \oplus c) = (a \oplus b) \oplus c.$$

  The same holds for the "otimes" operator

- **Commutativity**
  If we would change the order of operations the result would not change. For all $a, b \in \mathbb{R}_\varepsilon$:

  $$a \oplus b = b \oplus a \text{ and } a \otimes b = b \otimes a$$

  Note that the difference with associativity lies within the fact that now only the order of the elements has been changed, for associativity we also need to reorder the grouping of the elements.

- **Distributivity of $\otimes$ over $\oplus$**
  If the $\otimes$ operation is performed on a number of $\oplus$ operations, each element within the second group will be multiplied in the max-plus algebra. This means for all $a, b, c \in \mathbb{R}_\varepsilon$:

  $$a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c).$$

- **Existence of a zero element ε**
  Just as has been shown in Eq. (2.4): for all $a \in \mathbb{R}_\varepsilon$:

  $$a \oplus \varepsilon = \varepsilon \oplus a = a$$

- **Existence of a unit element e**
  Also shown previously in Eq. (2.5): for all $a \in \mathbb{R}_\varepsilon$:

  $$a \otimes e = e \otimes a = a$$

- **The zero absorbs in combination with the $\otimes$ operator**
  For all $a \in \mathbb{R}_\varepsilon$:

  $$a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon$$

- **Idempotency of the $\oplus$ operator**
  Performing the "oplus" operation multiple times will not change the result from the initial application:
  For all $a \in \mathbb{R}_\varepsilon$:
  $$a \oplus a \oplus a = \max(a, a, a) = a$$

The properties and their analogies in conventional algebra are summarized in the following table:

Property	Max-plus Algebra	Conventional Algebra
Associativity	$a \oplus (b \oplus c) = (a \oplus b) \oplus c$	$a + (b + c) = (a + b) + c$
Associativity II	$a \otimes (b \otimes c) = (a \otimes b) \otimes c$	$a \times (b \times c) = (a \times b) \times c$
Commutativity	$a \oplus b = b \oplus a$	$a + b = b + a$
Commutativity II	$a \otimes b = b \otimes a$	$a \times b = b \times a$
Distributivity	$a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$	$a \times (b + c) = (a \times b) + (b \times a)$
Zero element	$a \oplus \varepsilon = a$	$a + 0 = a$
Unit element	$a \otimes e = a$	$a \times 1 = a$
Absorption	$a \otimes \varepsilon = \varepsilon$	$a \times 0 = 0$

Table 2.1: Properties of max-plus algebra and their analogies.

Another analogy with conventional algebra is the way powers are computed.

Definition 2.1.3. Denote the set of natural numbers including zero as $\mathbb{N}$ and define $a \in \mathbb{R}_\varepsilon$ and $n \in \mathbb{N}$. In max-plus algebra powers are then defined as follows

$$a^{\otimes n} := \underbrace{a \otimes a \otimes \cdots \otimes a}_{\text{n times, n>0}} \tag{2.6}$$

■

As you have probably noticed, except for the usage of the "otimes" operator instead of the multiplication operator, power is defined in the same way as in conventional algebra. Even though the notation and definition resemble, there clearly is a difference in the result of taking a power as $a^{\otimes n}$. Of which the latter would be translated in conventional algebra into $n \times a$. While the power in conventional algebra is equal to $a \times a \times a \times \cdots$. To finalize, we define the zeroth power as $a^{\otimes 0} := e$.

2.2 Vectors and matrices

In this section, the aforementioned concepts will be extended to matrices over $\mathbb{R}_\varepsilon$. We denote the set of $n \times m$ matrices in max-plus algebra by $\mathbb{R}_\varepsilon^{n \times m}$. In case n is a natural number, $n \in \mathbb{N}$

with $n \neq 0$, define $\underline{n}$ as $\underline{n} := \{1, 2, \ldots, n\}$. A matrix $A \in \mathbb{R}_{\varepsilon}^{n \times m}$ will be denoted as

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{pmatrix}.$$

Throughout the report element a_{ij} will also be written as

$$[A]_{ij}, \quad i \in \underline{n}, j \in \underline{m}. \tag{2.7}$$

In Section 2.1, definitions for summation and multiplication in max-plus algebra were given for scalars, in this section these definitions are extended to operations over matrices and vectors.

Definition 2.2.1. For two matrices $A, B \in \mathbb{R}_{\varepsilon}^{n \times m}$ the summation, written as $A \oplus B$, is defined as:

$$[A \oplus B]_{ij} = a_{ij} \oplus b_{ij} = \max(a_{ij}, b_{ij}) \text{ for } i \in \underline{n}, j \in \underline{m} \tag{2.8}$$

■

Then starting with the scalar multiple of a matrix, matrix multiplication is introduced in max-plus algebra.

Definition 2.2.2. The multiplication of a scalar with a matrix will be written as $\alpha \otimes A$, for $\alpha \in \mathbb{R}_{\varepsilon}$ and $A \in \mathbb{R}_{\varepsilon}^{n \times m}$ and defined by:

$$[\alpha \otimes A]_{ij} = \alpha \otimes a_{ij}, \quad \forall i \in \underline{n} \text{ and } j \in \underline{m}. \tag{2.9}$$

■

Definition 2.2.3. For $A \in \mathbb{R}_{\varepsilon}^{n \times l}$ and $B \in \mathbb{R}_{\varepsilon}^{l \times m}$ matrix multiplication is defined as:

$$[A \otimes B]_{ik} = \bigoplus_{j=1}^{l} a_{ij} \otimes b_{jk} = \max_{j \in \underline{l}} \{a_{ij} + b_{jk}\} \tag{2.10}$$

for $i \in \underline{n}$ and $k \in \underline{m}$. ■

Again, there is a strong resemblance with conventional algebra.

Definition 2.2.4. Lastly, the matrix power function is defined by:

$$A^{\otimes k} := \underbrace{A \otimes A \otimes \cdots \otimes A}_{\text{k times}} \tag{2.11}$$

for $k \in \mathbb{N}$ with $k > 0$, and $A^{\otimes 0} := E$. Where E will be defined in Eq. (2.12) as the max-plus identity matrix. ■

Notice that (2.11) is nothing more than an extension of (2.6) to matrices. In the following example the previous definitions will be used to do some calculations in max-plus algebra.

Example 2.2.1. Given $\alpha = 2$, $A = \begin{pmatrix} e & 2 \\ \varepsilon & 3 \end{pmatrix}$ and $B = \begin{pmatrix} -1 & 3 \\ 1 & \varepsilon \end{pmatrix}$, the following computations can be done:

$$A \oplus B = \begin{pmatrix} \max(e,-1) & \max(2,3) \\ \max(\varepsilon,1) & \max(3,\varepsilon) \end{pmatrix} = \begin{pmatrix} e & 3 \\ 1 & 3 \end{pmatrix},$$
$$\alpha \otimes A = \begin{pmatrix} 2 \otimes e & 2 \otimes 2 \\ 2 \otimes \varepsilon & 2 \otimes 3 \end{pmatrix} = \begin{pmatrix} 2 & 4 \\ \varepsilon & 6 \end{pmatrix},$$
$$A \otimes B = \begin{pmatrix} \max(e-1, 2+1) & \max(e+3, 2+\varepsilon) \\ \max(\varepsilon - 1, 3+1) & \max(\varepsilon + 3, 3 + \varepsilon \end{pmatrix} = \begin{pmatrix} 3 & 3 \\ 4 & \varepsilon \end{pmatrix},$$
$$B \otimes A = \begin{pmatrix} \max(e-1, 3+\varepsilon) & \max(-1+2, 3+3) \\ \max(1+e, \varepsilon+\varepsilon) & \max(1+2, \varepsilon+3 \end{pmatrix} = \begin{pmatrix} -1 & 6 \\ 1 & 3 \end{pmatrix},$$
$$A^{\otimes 2} = A \otimes A = \begin{pmatrix} e & 5 \\ \varepsilon & 6 \end{pmatrix}.$$

◁

Notice that commutativity does not always hold for matrix multiplication, hence $A \otimes B = B \otimes A$ does not hold in general, but it may hold for special cases. Now we introduce two special matrices; the max-plus identity matrix, denoted by E, and the max-plus zero matrix, denoted by $\mathcal{E}$. Then the $n \times m$ identity matrix, $E(n,m)$, will be defined by

$$[E(n,m)]_{ij} := \begin{cases} e & \text{for i = j}, \\ \varepsilon & \text{for i} \neq \text{j}. \end{cases} \tag{2.12}$$

The $n \times m$ zero matrix, $\mathcal{E}(n,m)$, is a matrix of which all elements are equal to ε. In the case that $n = m$ we have the following representation of the two matrices:

$$E = \begin{pmatrix} e & \varepsilon & \cdots & \varepsilon \\ \varepsilon & e & \cdots & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ \varepsilon & \varepsilon & \cdots & e \end{pmatrix}, \quad \mathcal{E} = \begin{pmatrix} \varepsilon & \varepsilon & \cdots & \varepsilon \\ \varepsilon & \varepsilon & \cdots & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ \varepsilon & \varepsilon & \cdots & \varepsilon \end{pmatrix}. \tag{2.13}$$

Some of the properties that were mentioned in the scalar case, also hold when using these special matrices. For matrix $A \in \mathbb{R}_{\varepsilon}^{n \times m}$ the following neutral matrices exist:

- existence of a zero matrix: $A \oplus \mathcal{E}(n,m) = \mathcal{E}(n,m) \oplus A = A$,

  also for $k \geq 1$ we can write $A \otimes \mathcal{E}(m,k) = \mathcal{E}(n,k)$,
- existence of an identity matrix: $A \otimes E(m,m) = E(n,n) \otimes A = A$ (notice how commutativity holds true here).

From [6] we summarize the following:

- In max-plus algebra matrix addition is associative, commutative, and has the zero matrix, $\mathcal{E}$, as defined before.

- Matrix multiplication is associative, distributive (w.r.t. matrix addition), has matrix $\mathcal{E}$ which has the ability that it absorbs when multiplying, and it has a unit matrix.

Vectors will be defined such that the elements are in $\mathbb{R}_\varepsilon^{n\times 1}$. Define this space as $\mathbb{R}_\varepsilon^{n\times 1} := \mathbb{R}_\varepsilon^{n}$. There are two important vectors that we will define: the base vector and the unit vector. The base vector is a column of the identity matrix $\mathcal{E}$, hence it is a vector that consists of one e and the other elements will be equal to ε. The order depends on which column is chosen. The unit vector will be a vector with only e-elements, and is denoted with u.

2.3 Spectral theory

In this section, an overview of the spectral theory on matrices concerning max-plus algebra will be given. It aims to show the link of weighted graphs with eigenvector and eigenvalue theory of a square matrix over the max-plus semiring. Observations show us that square matrices can be represented as weighted graphs. Furthermore, there is a graph-theoretical visualisation of products and powers of matrices. The main result of this section is that when assuming certain conditions that allow the existence of a unique eigenvalue, this eigenvalue will equal the maximal average weight of circuits in the graph that represents the matrix.

When introducing the relationship between matrices and graphs it is important to start with certain definitions. First of all, a word that has already been used in the introduction; directed graph $\mathcal{G}$. This is represented by a pair $(\mathcal{N}, \mathcal{D})$, where $\mathcal{N}$ represents a finite set of nodes or vertices and $\mathcal{D}$ represents a set of ordered pairs (because the direction in which the arc is viewed is part of the distinction between arcs) called arcs, hence $\mathcal{D} \subset \mathcal{N} \times \mathcal{N}$. Assuming $i, j \in \mathcal{D}$, but $j, i \notin \mathcal{D}$, we can say that the directed graph has an arc that goes from i to j, but not from j to i. This also means that we have an outgoing arc from i and one incoming arc at j. When a directed graph is associated with any $n \times n$ matrix A over $\mathbb{R}_\varepsilon$ it will be called a communication graph, denoted by $\mathcal{G}(A)$. Furthermore, the set of nodes is denoted by $\mathcal{N}(A) = \underline{n}$. Then a pair (j, i) is an arc of this graph with weight a_{ij} only if $a_{ij} \neq \varepsilon$, where $\mathcal{D}(A)$ will denote the set of arcs of the communication graph.

When there are at least two nodes i, j, a sequence of arcs $p = \big((i_k, j_k) \in \mathcal{D} : k \in \underline{m}\big)$ in such a way that $i = i_1$ and $j_k = i_{k+1}$, for $k < m$, and $j_m = j$ is called a path from node i to node j. On this path the nodes $i = i_1, i_2, \ldots, i_m$, and $j_m = j$ can be found, also this path has length m. A circuit is a path with begin node equal to final node, i.e. if $i = j$. We speak of an elementary circuit when we have a circuit where each node has only one incoming and one outgoing arc. The weight of a circuit (i, j) is given by corresponding matrix elements a_{ji}. The length of a circuit is given by the number of arcs within the circuit. And lastly, the average circuit weight is given by the weight of a circuit divided by the length of a circuit. Denote the set of all paths of length $m \geq 1$ from i to j by $P(i, j; m)$. The weight of an arc (i, j) in $\mathcal{G}(A)$ is given by the value of a_{ji}. Then, the weight of a path in $\mathcal{G}(A)$ is defined by the sum of the weights of the arcs that are part of that path. Formally, we can write that for $p = \big((i_1, i_2), (i_2, i_3), \ldots, (i_m, i_{m+1})\big) \in P(i_1, i_{m+1})$ the weight of p can be defined as:

$$|p|_w = \bigotimes_{k=1}^{m} a_{i_{k+1}, i_k}. \tag{2.14}$$

We define the *average weight* of a path p as $\frac{|p|_w}{|p|_l}$, where $|p|_w$ and $|p|_l$ represents the weight and length respectively of path p.

Example 2.3.1. To summarize aforementioned definitions an example is given in the form of a matrix

$$A = \begin{pmatrix} \varepsilon & 4 & 4 \\ \varepsilon & 2 & 2 \\ 6 & \varepsilon & 3 \end{pmatrix}.$$

This can be translated into a communication graph as shown in **Figure 2.1**.

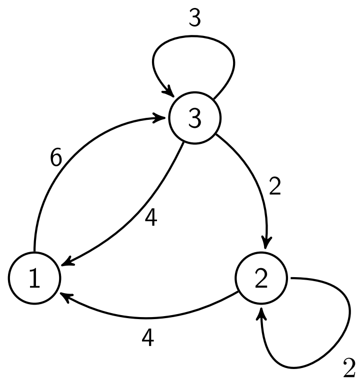


Figure 2.1: Communication graph of matrix A as proposed in Example 2.3.1

This graph $\mathcal{G}(A)$ has **node set** $\mathcal{N}(A) = \{1, 2, 3\}$, **arc set** $\mathcal{D}(A) = \{(1, 3), (3, 2), (2, 1), (3, 1), (3, 3), (2, 2)\}$. Furthermore, we can identify the **elementary circuits**, the circuits where each node has only one incoming and outgoing arc, namely: $\rho = \big((1, 3), (3, 2), (2, 1)\big)$, $\theta = \big((1, 3), (3, 1)\big)$, $\eta = \big((3, 3)\big)$ and, $\zeta = \big((2, 2)\big)$. The weight of these circuits can be found by adding together the weights of corresponding arcs and by counting the number of these arcs we can find the length. For the weight of ρ we find $|\rho|_w = a_{31} + a_{23} + a_{12} = 12$. For the length we find $|\rho|_l = 3$. Then, we can say that the average weight is to be found by dividing these two: $\frac{|\rho|_w}{|\rho|_l} = 4$. In the same way we can find these specifications for the other elementary circuits:

- $|\theta|_w = 10$, $|\theta|_l = 2$ and $\frac{\theta_w}{\theta_l} = 5$
- $|\eta|_w = 3$, $|\eta|_l = 1$ and $\frac{\eta_w}{\eta_l} = 3$
- $|\zeta|_w = 2$, $|\zeta|_l = 1$ and $\frac{\zeta_w}{\zeta_l} = 1$

This average circuit weight will become important when arriving to the theory on eigenvalues and eigenvectors. ◁

A graph is called *strongly connected* if for any two nodes $i, j \in \mathcal{N}$, node j is reachable from i, i.e. there exists a path between node j and node i. If a communication graph is strongly connected then its corresponding matrix $A \in \mathbb{R}_\varepsilon^{n \times n}$ will be called *irreducible*. If a matrix is irreducible it means that within the corresponding communication graph all nodes communicate. In the case that a graph $\mathcal{G}$ is not strongly connected, we can say that not all nodes are communicating. The corresponding matrix will then be *reducible*. Also, partitioning the set of nodes into subsets that do communicate with each other is possible. This is done when certain parts of a communication graph are strongly connected. When $\mathcal{N}$ is partitioned into subsets such that there exists a subgraph $\mathcal{G}_r$, of which the nodes, represented by the subset $\mathcal{N}_r$, are all communicating with each other. Furthermore, its arcs have a begin and end node in $\mathcal{N}_r$, represented by $\mathcal{D}_r$. If $\mathcal{D}_r$ is non-empty, we can state that the subgraph $\mathcal{G}_r = (\mathcal{N}_r, \mathcal{D}_r)$ is maximal strongly connected.

We define the *cyclicity* of a graph, denoted by $\sigma_\mathcal{G}$, by separating the definition into two parts. In the case that $\mathcal{G}$ is strongly connected and in the case it is not. When $\mathcal{G}$ is strongly connected, we can say that the graph's cyclicity is equal to the greatest common divisor of the lengths of all elementary circuits. Cyclicity is defined to be equal to one if the graph is just one node (without self-loop). In the case that $\mathcal{G}$ is not strongly connected, we find its cyclicity by looking at the least common multiple of all maximal strongly connected subgraph (m.s.c.s.). For an excellent example see [6, Example 2.1.3].

We have finally arrived at one of the critical parts of spectral theory. How all of the previous is related to eigenvalues and eigenvectors. Define these two notions following [6].

Definition 2.3.1. Given a square matrix $A \in \mathbb{R}_\varepsilon^{n \times n}$ and defining $\mu \in \mathbb{R}_\varepsilon$ as a scalar and $v \in \mathbb{R}_\varepsilon^N$ as a vector with one or more finite valued elements. Then we can say that if

$$A \otimes v = \mu \otimes v \tag{2.15}$$

holds, then μ represents an eigenvalue and v an eigenvector of the matrix A. ■

As mentioned in Example 2.3.1 the average weights of circuits are of great importance. In [6] they come back to this and relate this average weight of a circuit with a finite eigenvalue of matrix A. This will be summarized in the following lemma:

Lemma 2.3.1. *Assuming $A \in \mathbb{R}_\varepsilon^{n \times n}$ with corresponding finite eigenvalue μ. Then, the communication graph of matrix A has a circuit γ where*

$$\mu = \frac{\gamma_w}{\gamma_l} \tag{2.16}$$

The authors then propose to take the maximal average circuit weight as the primary candidate for an eigenvalue. In this proposition, it is important to start distinguishing critical circuits. When having a circuit γ in $\mathcal{G}(A)$ it will be called a *critical circuit* if its corresponding average weight is maximal. Furthermore, we will start to distinguish the *critical graph* of matrix A, denoted by $\mathcal{G}^c(A) = (\mathcal{N}^c(A), \mathcal{D}^c(A))$. Which in words means that it is the graph that consists of the nodes and arcs that belong to the critical circuit. Lastly, we will call nodes in $\mathcal{N}^c(A)$ *critical nodes*. To summarize these notions we revisit Example 2.3.1.

Example 2.3.2. Looking back at the average weights found in Example 2.3.1, we observe that circuit θ has the maximal average circuit weight equal to 5. Therefore, the circuit θ is part of the critical graph $\mathcal{G}^c(A)$ ◁

Continuing with important notions we come a closer to a generalization of how to find the eigenspace in spectral theory. For this, a new matrix will be introduced.

Definition 2.3.2. The *normalized matrix* is a matrix of which the maximal average weight is zero. It is defined as

$$[A_\lambda]_{ij} = a_{ij} - \lambda. \tag{2.17}$$

Where λ is supposed to represent the maximal average weight (and chosen to be an eigenvalue of the corresponding matrix)

$$\lambda = \max_{\gamma \in \mathcal{G}^c(A)} \frac{|\gamma_w|}{|\gamma_l|}. \tag{2.18}$$

■

Definition 2.3.3. Now we need to define another matrix A_λ^* that is necessary for finding the eigenspace. Define this matrix as

$$A_\lambda^* := E \oplus A_\lambda^+ = \bigoplus_{k \geq 0} A_\lambda^{\otimes k}, \tag{2.19}$$

in which A_λ^+ is defined as

$$A_\lambda^+ := \bigoplus_{k \geq 1}^{\infty} A_\lambda^{\otimes k} \tag{2.20}$$

■

It should be noticed that these definitions are closely related and that it is possible to write

$$A_\lambda^+ = A_\lambda \otimes (E \oplus A_\lambda^+) = A_\lambda \otimes A_\lambda^*. \tag{2.21}$$

Then for $\eta \in \mathcal{N}^c(A)$ we can say:

$$[A_\lambda^+]_{.\eta} = [A_\lambda^*]_{.\eta} \text{ with } [A_\lambda^+]_{\eta\eta} = e \;\; \forall \eta \in \mathcal{N}^c(A) \tag{2.22}$$

in which the notation $[A^*]_{.\eta}$ denotes the ηth column of matrix A^*. Combining Eq. (2.21) with the previous gives:

$$A_\lambda \otimes [A_\lambda^*]_{.\eta} = [A_\lambda^*]_{.\eta},$$

this can again be written as:

$$A \otimes [A_\lambda^*]_{.\eta} = \lambda \otimes [A_\lambda^*]_{.\eta}.$$

From this we can infer that λ is an eigenvalue of A with corresponding eigenvector the ηth column of A_λ^*. This is summarized in the following lemma.

Lemma 2.3.2. *Assume matrix $A \in \mathbb{R}_\varepsilon^{n \times n}$ with associated communication graph $\mathcal{G}(A)$ and finite maximal average circuit weight λ. The scalar associated with the maximal average circuit weight λ will also be an eigenvalue of A, furthermore column $[A_\lambda^*]_{.\eta}$ will be an eigenvector associated with eigenvalue λ, this for any node $\eta \in \mathcal{G}^c(A)$.*

In the following theorem the eigenspace of an irreducible square matrix is characterized:

Theorem 2.3.1. *Assuming that A is irreducible and square, and A^*_λ is defined by Definition 2.3.3. The following will hold.*

First of all, if node i is within $\mathcal{G}^c(A)$, then we can say that the ith column of A^*_λ is an eigenvector of A.

Secondly, the eigenspace of matrix A will be given as:

$$V(A) = \left\{ v \in \mathbb{R}^n_\varepsilon : v = \bigoplus_{i \in \mathcal{N}^c(A)} a_i \otimes [A^*_\lambda]_{.i} \quad \text{for } a_i \in \mathbb{R}_\varepsilon \right\}. \tag{2.23}$$

And lastly, when nodes i, j belong to the critical graph, there exists $a \in \mathbb{R}$ where:

$$a \otimes [A^*_\lambda]_{.i} = [A^*_\lambda]_{.j}, \tag{2.24}$$

which only holds when both nodes belong to the same m.s.c.s. For more information and proofs consult [6].

Example 2.3.3. In this example, everything that has been discussed before will be put into practice. Given matrix

$$A = \begin{pmatrix} 2 & 1 \\ 3 & e \end{pmatrix}.$$

This can be translated into a communication graph as shown in **Figure 2.2**. So, looking at

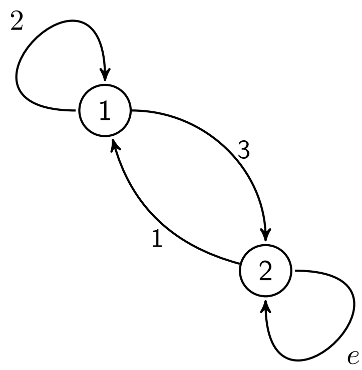


Figure 2.2: Communication graph of matrix A as proposed in Example 2.3.3

the graph $\mathcal{G}(A)$ we see that it is *irreducible*. This can be concluded, because if we choose any of the two nodes, the other is always reachable. The set of nodes consists of: $\mathcal{N}(A) = \{1, 2\}$, and the set of arcs consists of: $\mathcal{D}(A) = \{(1,1), (1,2), (2,1), (2,2)\}$. In this graph there are three elementary circuits:

- $\rho = (1, 1)$, weight $\rho_w = 2$, and length $\rho_l = 1$

- $\theta = (2,2)$, weight $\theta_w = 0$, and length $\theta_l = 1$
- $\eta = \big((1,2),(2,1)\big)$, weight $\eta_w = 4$, and length $\eta_l = 2$

To find the maximal average circuit weight, and thus the eigenvalue, we look at the average weights:

- $\dfrac{\rho_w}{\rho_l} = \dfrac{2}{1} = 2$
- $\dfrac{\theta_w}{\theta_l} = \dfrac{0}{1} = 0$
- $\dfrac{\eta_w}{\eta_l} = \dfrac{4}{2} = 2$

Hence, we find two maximal weights, and $\lambda = 2$, according to Lemma 2.3.2. Furthermore, the critical graph has one maximal strongly connected subgraph. Now that we have collected this information we can find the normalized matrix:

$$A_\lambda = \begin{pmatrix} e & -1 \\ 1 & -2 \end{pmatrix}.$$

With this matrix, and Definition 2.3.3 we can construct:

$$A_\lambda^* = \begin{pmatrix} e & -1 \\ 1 & e \end{pmatrix}.$$

By Theorem 2.3.1 we can say that there is one eigenvector belonging to the eigenvalues, as the columns of A_λ^* are clearly max-plus multiples of each other (also it follows from the fact that we have one m.s.c.s.). In short, the eigenspace of matrix A is given by: $V(A) = \overline{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}$.[1] ◁

2.4 Max-plus linear systems

As mentioned before, in research on DESs max-plus algebra is often used to describe these systems. These DESs are systems that have a finite number of resources and have several shared users that are all working towards the same goal [3]. These models are of special interest because there is a class of discrete event systems that is nonlinear in conventional algebra, but when described in max-plus algebra they become linear. This class that is linear in max-plus algebra is called MPL systems. For describing this class of systems there must be only synchronization and no concurrency or choice [4]. Following the book of [3] MPL systems will be defined in the following way:

$$x(k) = A \otimes x(k-1) \oplus B \otimes u(k) \tag{2.25}$$
$$y(k) = C \otimes x(k), \tag{2.26}$$

[1] The bar operator represents the canonical projection of $\mathbb{R}_\varepsilon^n$ onto the projective space. This space has not been introduced in the thesis as it is not of much importance for understanding the main matter, and except for this example it will not come back. For further understanding, consult e.g. [6, Section 1.4]

in which $A \in \mathbb{R}_{\varepsilon}^{n \times n}$, $B \in \mathbb{R}_{\varepsilon}^{n \times m}$, and $C \in \mathbb{R}_{\varepsilon}^{p \times n}$ represent the system matrices, where n is the number of states, m the number of inputs, and p is the number of of outputs. Furthermore, $x(k)$ and $u(k)$ represent the state and input respectively and the index k is the event counter. The state and output often contain instants at which the described events occur for the kth time. Important is that often the state is equal to the output in the systems described by this class, which means that the output equation $y(k)$ is left out. This will leave us with only the state Eq. (2.25).

In some cases events depend on the previous cycle and cycles before that one. In other words the event is dependent on events in cycle $k - \mu$, for $\mu \in \{2, \ldots, \mu_{max}\}$, such that the state equation can be written as:

$$x(k) = \bigoplus_{\mu=1}^{\mu_{max}} A_\mu \otimes x(k - \mu) \oplus B \otimes u(k) \tag{2.27}$$

Since the model described in (2.27) only contains past entries of the state it is referred to as an *explicit model.* Often when describing a system it leads to another form, the *implicit form.* This form is defined as [5]:

$$x(k) = \bigoplus_{\mu=0}^{\mu_{max}} A_\mu \otimes x(k - \mu) \oplus B \otimes u(k) \tag{2.28}$$

where μ_{max} represents the number of past cycles that are regarded in the system. As can be seen in Eq. (2.28), $x(k)$ now depends on the state vector of both the previous cycle(s) and on itself, hence the current cycle. This directly displays the disadvantage of having a system described in the implicit form: if it is desired to determine the state vector $x(k)$, the model needs to be iterated multiple times [7]. This is not desired as iterating could make the computational effort greater. Luckily, there is a way to rewrite (2.28) into the explicit form. As we have seen in Section 2.3 it is possible to find eigenvalues and eigenvectors of matrices and in [6] it is proposed that it is also possible to find unique solutions of MPL systems. Let us consider the following simplified MPL system:

$$x = A \otimes x \oplus b \tag{2.29}$$

where A has to be in $\mathbb{R}_{\varepsilon}^{n \times n}$.

Theorem 2.4.1. *It follows that under the condition that the communication graph of A has maximal circuit weight less than or equal to zero, or in other words only non-positive weights. Then the unique solution of this system will be:*

$$x = A^* \otimes b \tag{2.30}$$

where A^* is actually just the same as has been defined in Eq. (2.3.3):

$$A^* := \bigoplus_{k \geq 0} A^{\otimes k}, \tag{2.31}$$

With this we have all ingredients to show that the implicit form described in (2.28) can be rewritten into the following explicit form [5]:

$$x(k) = A_0^* \otimes \Big(\bigoplus_{\mu=1}^{\mu_{max}} A_\mu \otimes x(k-\mu) \oplus B \otimes u(k) \Big) \tag{2.32}$$

Assume that the conditions to use **Theorem 2.4.1** hold and that $\mu_{max} = 1$. Then the implicit form defined in Eq. (2.28) becomes:

$$\begin{aligned} x(k) &= A_0 \otimes x(k) \oplus \underbrace{A_1 \otimes x(k-1) \oplus B \otimes u(k)}_{=b} && (2.33)\\ &= A_0 \otimes x(k) \oplus b && (2.34)\\ &= A_0^* \otimes b && (2.35)\\ &= A_0^* \otimes \Big(A_1 \otimes x(k-1) \oplus B \otimes u(k) \Big) && (2.36) \end{aligned}$$

which is in accordance with Eq. (2.32), and also in the explicit form.

2.5 Max-plus linear systems a first model: The urban railway network

To explain the concept of MPL systems, we introduce an application: an urban railway network. Consider a simple urban railway network with J stations, as indicated in **Figure 2.3**. The starting station will be station 1 and the final station will be station J.

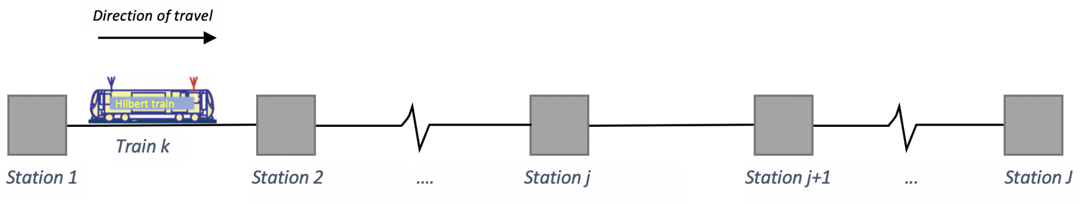


Figure 2.3: A simplified illustration of the urban railway model.

The railway model that we will create will be made for the sake of introducing and analyzing max-plus linear parameter varying (MP-LPV) models. Let us first define some variables:

1. $k \in \{1, 2, \ldots\}$: the train counter. This means it is a product counter instead of a cycle counter.
2. $j \in \{1, 2, \ldots, J\}$: the station counter.
3. $a_j(k)$: the arrival time of train k at station j.
4. $d_j(k)$: the departure time of train k at station j.
5. $\tau_{r,min,j-1}$: the running minimum time between station $j-1$ and j.

6. $\tau_{d,j}$: the time that a train dwells at station j.

7. $\tau_{h,j}$: the headway time at station j, ensures minimal distance between two trains.

To formulate the model the following assumptions are made:

Assumption A1. *The state is said to be non-decreasing in the event counter k:*

$$a_j(k) \geq a_j(k-1), \tag{2.37}$$
$$d_j(k) \geq d_j(k-1) \tag{2.38}$$

Assumption A2. *There will be one train track between all stations $j \in \{1, 2, \ldots, J\}$ where overtaking is not possible.*

Assumption A3. *Except for the first and final station, all other stations, hence $j \in \{2, 3, \ldots, J-1\}$, can only accommodate one train at the time.*

Assumption A4. *The trains have a non-zero velocity limit, hence the running time between two stations is finite.*

Assumption A5. *The dwell time at the different stations is predefined and independent of the passengers arriving at a station. Trains leave directly after this dwell time.*

Assumption A6. *The trains leave as soon as possible.*

Assumption A7. *The capacity of the trains is infinite, hence we never arrive at the situation where the train is full.*

Assumption A1 is evident since **Assumption A2** and **Assumption A3** assure that the arrival and departure times of a train should always be earlier than the trains that leave after this train. **Assumption A2** and **Assumption A3** are the general case for urban railway systems, e.g. due to physical limitations [1]. **Assumption A4** is also explained by physical limitations, trains in general have a maximum velocity they can reach on the tracks. **Assumption A5** is made to keep the model simple for now, later in the thesis this assumption will be changed into an Origin-Destination passenger dependent dwell time [1], i.e. a varying dwell time that will be our varying parameter. **Assumption A6** and **Assumption A7** allow us to write this model solely in max-plus equations with an equality sign.

These assumptions lead to the following equations for the arrival and departure time:

$$a_j(k) \geq d_{j-1}(k) + \tau_{r,min,j-1}, \tag{2.39}$$
$$a_j(k) \geq d_j(k-1) + \tau_{h,min}, \tag{2.40}$$
$$d_j(k) \geq a_j(k) + \tau_{d,j}. \tag{2.41}$$

Observe that these equations can be rewritten (due to **Assumption A6**) into:

$$a_j(k) = \max(d_{j-1}(k) + \tau_{r,min,j-1}, \quad d_j(k-1) + \tau_{h,min}), \tag{2.42}$$
$$d_j(k) = a_j(k) + \tau_{d,j}. \tag{2.43}$$

This can be translated into max-plus notation, as follows:

$$a_j(k) = d_{j-1}(k) \otimes \tau_{r,min,j-1} \oplus d_j(k-1) \otimes \tau_{h,min}, \tag{2.44}$$
$$d_j(k) = a_j(k) \otimes \tau_{d,j}. \tag{2.45}$$

These equations lead to the following max-plus linear model:

$$x(k) = A_0 \otimes x(k) \oplus A_1 \otimes x(k-1) \tag{2.46}$$

$$= \begin{pmatrix} \varepsilon & \varepsilon & \dots & \varepsilon & \varepsilon \\ \tau_{r,1} \otimes \tau_{d,2} & \varepsilon & \ddots & \vdots & \vdots \\ \varepsilon & \ddots & \ddots & \varepsilon & \vdots \\ \vdots & \ddots & \ddots & \varepsilon & \varepsilon \\ \varepsilon & \dots & \varepsilon & \tau_{r,J-1} \otimes \tau_{d,J} & \varepsilon \end{pmatrix} \otimes x(k) \dots \tag{2.47}$$

$$\dots \oplus \begin{pmatrix} \tau_h \otimes \tau_{d,1} & \varepsilon & \dots & \varepsilon \\ \varepsilon & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \varepsilon \\ \varepsilon & \dots & \varepsilon & \tau_h \otimes \tau_{d,J} \end{pmatrix} \otimes x(k-1)$$

with system matrices A_0 and $A_1 \in \mathbb{R}_\varepsilon^{n\times n}$, and state vector $x(k) = [d_1(k)\ \ d_2(k) \dots d_J(k)]^T \in \mathbb{R}_\varepsilon^n$.

Throughout the thesis we will use this model as a running example. We will change the assumptions such that it will eventually lead to an MP-LPV system. This will give us the possibility to apply new theory directly on an application.

2.6 Convex theory

In this section, we provide definitions regarding convexity and max-plus algebra. This is a concise summary of the literature, and for a better understanding of the material please consult the concerned literature, e.g. [15]. This section is necessary, since using the results in this section will allow us to show the equivalence between the different forms in which we can write models of interest. In our case, we will later show that it is possible to write the urban railway model as an MP-LPV and as a MMPS function (will be defined in the next section). To prove this relation, a number of definitions regarding convexity are used. We start by introducing the basic notion of convex sets as in [15]:

Definition 2.6.1. A convex set is defined as a subset C of $\mathbb{R}^n$ for which it holds that

$$\alpha x + (1-\alpha)y \in C, \quad \forall x, y \in C, \quad \forall \alpha \in [0,1]. \tag{2.48}$$

■

In this thesis some special convex sets will be considered, which we will introduce in the following definitions.

Definition 2.6.2. We define a halfspace when we have a set that is specified by one linear inequality. We can write the set $\mathcal{X}$ as follows:

$$\mathcal{X} = \{x \mid ax \leq b, \quad j = 1, \ldots, r\}, \tag{2.49}$$

with $a \in \mathbb{R}^n$ a non-zero vector and $b \in \mathbb{R}$ a scalar. ■

The definition of halfspaces is needed to define polyhedral sets.

Definition 2.6.3. A polyhedral set is defined, when it is not an empty set, as the intersection of finitely many halfspaces, where the set $\mathcal{X}$ has the following form:

$$\mathcal{X} = \{x \mid a'_j x \leq b_j, \quad j = 1, \ldots, r\}, \tag{2.50}$$

with $a_1, \ldots, a_r \in \mathbb{R}^n$ some vectors, and $b_1, \ldots, b_r \in \mathbb{R}$ some scalars. ■

Note that a polyhedral set is convex and closed, see [15, Proposition 1.1.1(a)]. Also, a bounded polyhedral set is called a *polytope*.

With this we have introduced all notions regarding convexity that are used in this thesis. For further understanding or more extensive explanation consult e.g. [15].

2.7 MMPS and PWA

MMPS systems are described by expressions where scaling, minimization, maximization, and addition are allowed. Up until now we have discussed systems where only the maximization and addition is used, hence MMPS systems are an extension of those. For the variables x_1 and x_2 three examples of MMPS expressions are listed below [16]:

Example 2.7.1.

$$\begin{aligned} f_1 &= x_1 - x_2 + 3 \\ f_2 &= \max(\min(3x_1, -x_2), x_1 + x_2) \\ f_3 &= x_1 + 2\max(x_1 + 3x_2, x_1 - \min(x_1, x_2, \max(x_2 + 3, x_1 - x_2))) \end{aligned}$$

◁

Let us define MMPS functions as follows [17]:

Definition 2.7.1. An MMPS function is defined by the following recursive grammar [17]:

$$f(x) := x_i \mid \alpha \mid \max(f_k(x); f_l(x)) \mid \min(f_k(x); f_l(x)) \mid f_k(x) + f_l(x) \mid \beta f_k(x) \tag{2.51}$$

with $i \in \{1, \ldots, n\}$, $\alpha, \beta \in \mathbb{R}$, and where $f_k : \mathbb{R}^n \to \mathbb{R}^m$, $f_l : \mathbb{R}^n \to \mathbb{R}^m$ are both again MMPS functions. In this context the | symbol means *or*. ■

MMPS are closely related to PWA functions. Equivalence of continuous PWA and MMPS has been show in [17], [18]. A continuous PWA function is defined as follows [18]:

Definition 2.7.2. Define a vector-valued function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ as a continuous PWA function if:

- A finite family of closed polyhedral regions $\mathcal{C}_1, \ldots, \mathcal{C}_N$ that covers $\mathbb{R}^n$ exists.
- Also, it should be so that for each $i \in \{1, \ldots, N\}, j \in \{1, \ldots, m\}$, it is possible to express every component f_j of f as $f_j(x) = \alpha_{i,j}^T x + \beta_{i,j}$ for any $x \in \mathcal{C}_i$, and $\alpha_{i,j} \in \mathbb{R}^n, \beta_{i.j} \in \mathbb{R}$.

■

Since we have defined the polyhedral regions $\mathcal{C}_i$ as closed the result is that every component of the function f is continuous, for the reason that there is continuity on the boundary between any two regions [18].

2.8 Summary

In this chapter, we have seen a small summary of max-plus algebra and its usage in modelling dynamical systems. In Section 2.1, we have discussed the basic concepts and definitions of max-plus algebra. We have seen the similarities and differences with conventional algebra. Afterwards, these concepts and definitions are extended to vectors and matrices in Section 2.2. Furthermore, in Section 2.3, spectral theory concerning max-plus algebra is discussed. This section should be read as a very basic and compact overview of the theory on this subject. Then an introduction to MPL systems has been given in Section 2.4. After which we introduce an MPL model, namely the urban railway network. Then, we conclude with two sections related to max-plus algebra, in these sections we give a brief introduction to convex theory, PWA, and MMPS systems.

In the next chapter linear parameter varying (LPV) systems are discussed and the extension of MPL systems by adding a varying parameter is introduced.

Chapter 3

Max-plus linear parameter varying systems

In the previous chapter, we introduced max-plus basics and max-plus linear (MPL) systems. We briefly discussed max-min-plus-scaling (MMPS) systems and convex theory which will be used in Section 3.3. In this chapter, we will introduce a relatively new framework: max-plus linear parameter varying (MP-LPV) systems [11], [19], and [20]. This framework is an extension of MPL with a varying parameter. Before laying the foundations of MP-LPV we will give a brief introduction to linear parameter varying (LPV) systems. Then, we will introduce a general MP-LPV framework. Finally, the relation of MP-LPV systems with MMPS functions is shown through three propositions and their proofs.

3.1 Linear parameter varying systems

The framework of LPV systems was introduced by Shamma [14] during the analysis of the control of nonlinear systems, using several linear controllers. These linear controllers were all able to control the system at a different operating point. This has been defined in shorter terms as *gain-scheduling*. Unfortunately, this gain-scheduling method has several shortcomings [21]. Some of the disadvantages are that to achieve certain performance specifications a large number of linear controllers is needed, furthermore the stability or performance cannot be guaranteed if there are rapid changes in the scheduling variable, and lastly this method is ad hoc in general. These drawbacks could be overcome by designing one global controller for an LPV system. LPV systems are systems of which the system matrices are known functions of time-varying parameters [22]. Shamma describes the LPV framework as a middle ground between linear and nonlinear systems. If we use the LPV framework we make it possible to "extract" the scheduling variable out of the systems, see **Figure 3.1**. In the figure we first see a nonlinear system, after disconnecting the scheduling variable from the dynamics of the system the relation between input and output becomes linear. Let us now define LPV systems in conventional algebra.

Figure 3.1: When using the LPV framework we can extract the scheduling variable p out of the system. When we extract this variable from the dynamics of the system we are left with a remaining relation between inputs and outputs that is linear.

Consider the following state-space form of a discrete dynamic system:

$$\begin{aligned} x(k+1) &= A(p)x(k) + B(p)u(k) \\ y(k) &= C(p)x(k) + D(p)u(k) \end{aligned} \tag{3.1}$$

with $p \in \mathcal{P}$ where $\mathcal{P} \subseteq \mathbb{R}^s$ denotes the bounded parameter set. Moreover, the state is represented by $x(k) \in \mathbb{R}^n$, the output by $y(k) \in \mathbb{R}^l$, the input $u(k) \in \mathbb{R}^m$, and the parameter varying system matrices by $A(p)$, $B(p)$, $C(p)$ and $D(p)$. Depending on how the varying parameter is defined in (3.1) we can describe different systems, such as linear time varying (LTV) or nonlinear systems [23]. In our case, we are particularly interested in the nonlinear system description, when $p = p(x(k))$[1]. If we denote this parameter vector as $p(k) \in \mathbb{R}^s$, it is possible to consider the class of discrete LPV systems as

$$x(k+1) = A(p(k))x(k) + B(p(k))u(k) \tag{3.2}$$
$$y(k) = C(p(k))x(k) + D(p(k))u(k) \tag{3.3}$$

in which the parameter $p(k)$ is now a function of the system's signals. LPV systems are linear in the parameter $p(k)$, but that does not mean that $p(k)$ itself is linear. This varying parameter can be the difference between two or more time-steps of the state-vector x. An example of this is: $p(k) = \lambda(x(k) - x(k-1) - x(k-2))$. Notice that, since the system matrices have expressions of the state itself in them, this system is nonlinear. Also, an important assumption is made in this framework, namely that the time-varying parameter is not known in advance, but measurable in real-time.

In the literature concerning MPL systems, dealing with a varying parameter set has been addressed by modelling systems as a switching max-plus linear (SMPL) system [8]. This description has system matrices that vary per event but the matrices are still constant. We are interested in cases where there is a parametric dependency on the states of the system. In this thesis, this class of systems will be called MP-LPV systems.

3.2 General MP-LPV framework

From the previous discussion, it can be understood that the extension of MPL systems with a varying parameter set with dependency on the state can be defined as a combination of system representations, in this case: the state space form, and parameter trajectories. The

[1]For simplicity we will write $p(k)$

parameter p is not necessarily known beforehand. We will define p such that it lies in a bounded set.

Definition 3.2.1. (Parameter set)
The variable p is the linear varying parameter and belongs to a closed, bounded set $\mathcal{P} \subseteq \mathbb{R}_{\varepsilon}^{s}$, where s is the size of the varying parameter $p \in \mathcal{P}$. ■

Note that this definition gives us the opportunity to convert an MP-LPV system to an MPL or SMPL system.

Definition 3.2.2. (General MP-LPV system) A dynamic system of the form:

$$x(k) = A(p) \otimes x(k-1) \oplus B(p) \otimes u(k) \tag{3.4}$$

$$y(k) = C(p) \otimes x(k), \tag{3.5}$$

is called a general MP-LPV system, where $A \in \mathbb{R}_{\varepsilon}^{n \times n}$, $B \in \mathbb{R}_{\varepsilon}^{n \times m}$, and $C \in \mathbb{R}_{\varepsilon}^{l \times n}$ represent the system matrices, where n is the number of states, m the number of inputs, and l is the number of outputs. Furthermore, $x(k)$, $y(k)$ and $u(k)$ represent the state, output and input respectively and the index k is the event counter. In general, the state $x(k)$ of MP-LPV systems contains time instants where internal events occur for the k-th time. The input $u(k)$ and output $y(k)$ contain the time instant at which input events and respectively output events occur. ■

Note that in this thesis the focus will be on MP-LPV systems with a nonlinear representation. This means that we have a varying parameter that is a function of the system's signals, such as states, inputs and/or outputs. Denoted as follows:

$$p(k) = p(k, x(k), y(k), u(k)), \tag{3.6}$$

Moreover, the parameter dependence in MP-LPV systems can have many forms. In the max-plus framework, it is important to note that we can write the varying parameter in both conventional algebra and max-plus algebra. When we write the varying parameter in conventional algebra, we are not only "hiding" nonlinearity of the system, but we are also using operators outside of the max-plus framework. In this thesis, we will consider an affine and polytopic parametric dependence both. Lastly, we will extend our description of the MP-LPV framework by considering not only events of one previous cycle, but also on the current cycle or cycles $(k - \mu)$ for $\mu = 1, \ldots, \mu_{max}$. Again there is a difference between explicit and implicit models. The *explicit MP-LPV state equation* can be written as:

$$x(k) = \bigoplus_{\mu=1}^{\mu_{max}} A_{\mu}(p(k)) \otimes x(k-\mu) \oplus B(p(k)) \otimes u(k) \tag{3.7}$$

Additionally, there will be models that take into account both the previous cycles and the current cycle. These models can be described by the *implicit MP-LPV state equation*:

$$x(k) = \bigoplus_{\mu=0}^{\mu_{max}} A_{\mu}(p(k)) \otimes x(k-\mu) \oplus B(p(k)) \otimes u(k) \tag{3.8}$$

The description given in Eq. (3.8) encapsulates both the explicit and implicit description of the system[2]. In Section 2.4 we have seen why it is important to make a distinction between these two descriptions. Now, it is at least as important, because for MP-LPV systems with dependency on the states, it is not possible anymore to rewrite implicit equations into explicit equations by using the Kleene star matrix A^*. Because when computing the Kleene star one will find a matrix with a the state-dependent varying parameter, hence one would not be able to get rid of the implicitness in the equations. Also, solving this implicit problem brings some difficulty. Since it would mean that we should solve the equation for every event k iteratively. In [11] an algorithm has been proposed, but this algorithm is incomplete. Issues arise for certain values, which would imply that the equations are not always solvable. In the next chapter, we will focus on the solvability of MP-LPV systems. In order to discuss solvability we need to extend the general case, we start with two main differences: defining the varying parameter in conventional algebra or max-plus algebra.

3.2.1 MP-LPV with varying parameter in conventional algebra

Defining the varying parameter in conventional algebra would mean that we are not only hiding non-linearities in max-plus algebra, but also the fact that we are using conventional algebra. As has been mentioned previously, it is possible to have an affine parameter dependence [12]. Looking at Eq. (3.8) we can then write the affine parameter dependence of the system matrices $A_\mu(p)$ and $B(p)$ as:

$$A_\mu(p) = A_{\mu,0} + \sum_{l=1}^{L} p_{\mu,l} A_{\mu,l}, \tag{3.9}$$

$$B(p) = B_0 + \sum_{l=1}^{L} p_l B_l \tag{3.10}$$

It is possible to further restrict this representation by saying that the parameter set $\mathcal{P}$ is a hypercube, i.e. a higher dimensional analogue of a three-dimensional cube [24]. In other words the varying parameter is bounded such that $\mathcal{P} = \{\underline{p}_l \leq p_l \leq \overline{p}_l\}$. Under these assumptions the system has a so-called polytopic representation:

$$A_\mu(p) = \sum_{l=1}^{2L} \delta_l^\mu \hat{A}_l^\mu, \quad \sum_{l=1}^{2L} \delta_l^\mu = 1, \quad \delta_l \geq 0. \tag{3.11}$$

where $\hat{A}_l^\mu$ is found by evaluating (3.9) at the vertices of the set $\mathcal{P}$ [12]. The affine and polytopic description, although not general descriptions, encompass a lot of practical examples [25]. In our framework the examples are less evident one of them will be discussed in the next section.

Case study: MP-LPV with varying parameter in conventional algebra

When modelling an urban railway network in max-plus algebra a model with affine parametric dependency arises [11]. This single-cycle max-plus system is described as follows:

$$x(k) = A_0(p(k)) \otimes x(k) \oplus A_1 \otimes x(k-1) \oplus u(k), \tag{3.12}$$

[2] Note that the output of the system is left out in the equation. This can be done because often output events are stacked in the state vector.

in which the A_0 matrix contains the dwell times, and the minimal traversal times. The A_1 matrix contains the headway times. What is important is that the lack of linearity is hidden in the $A_0(p(k))$ matrix, as the dwell times are also dependent on the train-counter. These elements come forth from the constraints that have to be taken into account when modelling a train network:

1. Dwell time constraints: the time the passengers spend at the platform for boarding or alighting the train.
2. No passengers arrive during the dwell time of a train. This is a very important assumption as it reduces the implicit complexity of the model that is discussed.
3. Trains may be delayed to improve the overall performance of the system.
4. Lastly, trains can only leave a station after a specified dwell time τ_d, such that it is possible for passengers to alight or board the train.

This last assumption is the keystone in the modelling of the railway network. From this assumption we can write the following equations:

$$a_j(k) \geq d_{j-1}(k) + \tau_{r,min,j-1} \tag{3.13}$$
$$a_j(k) \geq d_j(k-1) + \tau_{h,min} \tag{3.14}$$
$$d_j(k) \geq a_j(k) + \tau_{d,j}(k) \tag{3.15}$$

where $a_j(k)$ and $d_j(k)$ represent the the arrival and departure times at station j, $\tau_{r,min,j-1}$ is the minimum running time between station $j-1$ and station j, $\tau_{h,min}$ is the minimum headway time, and $\tau_{d,j}(k)$ is the dwell time of train k at station j. This can be translated into max-plus notation, as follows:

$$a_j(k) = d_{j-1}(k) \otimes \tau_{r,min,j-1} \oplus d_j(k-1) \otimes \tau_{h,min} \tag{3.16}$$
$$d_j(k) = a_j(k) \otimes \tau_{d,j}(k) \tag{3.17}$$

Furthermore, the state is denoted as the arrival and departure times of train k and the set of equations are written as:

$$x(k) = \left(\begin{array}{c} a_1(k) \\ a_2(k) \\ \vdots \\ a_J(k) \\ \hline d_1(k) \\ d_2(k) \\ \vdots \\ d_J(k) \end{array}\right) = \left(\begin{array}{c} d_1(k-1) \otimes \tau_{h,min} \\ d_1(k) \otimes \tau_{r,min,1} \oplus d_2(k-1) \otimes \tau_{h,min} \\ \vdots \\ d_{J-1}(k) \otimes \tau_{r,min,J-1} \oplus d_J(k-1) \otimes \tau_{h,min} \\ \hline a_1(k) \otimes \tau_{d,1}(k) \\ a_2(k) \otimes \tau_{d,2}(k) \\ \vdots \\ a_J(k) \otimes \tau_{d,J}(k) \end{array}\right) \tag{3.18}$$

It is important to notice that the varying parameter comes forward through the variable dwell time:

$$p(k) = \boldsymbol{\tau}_{d,j}(k) = \boldsymbol{\alpha}_{1,d} + \begin{bmatrix} \boldsymbol{\Psi} & \mathbf{0} \end{bmatrix} x(k) - \begin{bmatrix} \mathbf{0} & \boldsymbol{\Psi} \end{bmatrix} x(k-1) \tag{3.19}$$

where $\boldsymbol{\alpha}_{1,d} \in \mathbb{R}^{J\times 1}$ is simply a vector filled with a coefficient that can be estimated from historical data, $\boldsymbol{\Psi} \in \mathbb{R}^{J\times J}$ contains other coefficients estimated from historical data α, constants such as λ_j (passengers arrival rate at station j) and ρ_j (fixed proportion of passengers leaving at station j), and zero vector $\mathbf{0} \in \mathbb{R}^{J\times J}$. This variable dwell time is defined in conventional algebra and shows a scaled relation between $x(k)$ and $x(k-1)$. This dependency relates to the affine parameter dependency we discussed in the previous subsection. We can show this by looking at two stations for this model in the following example

Example 3.2.1. Consider the urban railway network, but now only with two stations. We can describe the model as follows

$$x(k) = A_0(p(k)) \otimes x(k) \oplus A_1 \otimes x(k-1), \tag{3.20}$$

with the state vector, and system matrices being equal to:

$$x(k) = \begin{pmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{pmatrix} = \begin{pmatrix} a_1(k) \\ a_2(k) \\ d_1(k) \\ d_2(k) \end{pmatrix}, \tag{3.21}$$

$$A_0(p(k)) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \tau_{r,1} & \varepsilon \\ p_1(k) & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & p_2(k) & \varepsilon & \varepsilon \end{pmatrix}, \tag{3.22}$$

$$A_1 = \begin{pmatrix} \varepsilon & \varepsilon & \tau_h & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \tau_h \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}. \tag{3.23}$$

where $p_1(k)$, and $p_2(k)$ are row vectors from Eq. (3.19). It is easy to see the affine parametric dependence, described in Eq. (3.9). in the system matrices. We can for example describe:

$$A_0(p(k)) = A_{0,0} + p_{0,1}A_{0,1} + p_{0,2}A_{0,2} \tag{3.24}$$

$$= \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \boldsymbol{\tau_{r,1}} & \varepsilon \\ \boldsymbol{e} & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \boldsymbol{e} & \varepsilon & \varepsilon \end{pmatrix} + p_1(k) \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \boldsymbol{e} & \varepsilon \\ \mathbf{1} & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \boldsymbol{e} & \varepsilon & \varepsilon \end{pmatrix} + p_2(k) \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \boldsymbol{e} & \varepsilon \\ \boldsymbol{e} & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \mathbf{1} & \varepsilon & \varepsilon \end{pmatrix} \tag{3.25}$$

◁

What is interesting about this case is that relaxations had to be done to guarantee solvability. After this relaxation, the system became explicit, hence only dependent on known values. In Chapter 6 we will propose a simplified version of the urban railway network with passenger dependent dwell time and discuss the solvability when the relaxation is not made and the model is still implicit. But first, let us discuss how to define models with a varying parameter in max-plus algebra.

3.2.2 MP-LPV with varying parameter in max-plus algebra

Using the general framework proposed in Eq. (3.4) and (3.5), we can now define the varying parameter in max-plus algebra [20]. In a similar matter, we define the system matrices $A(p)$, $B(p)$ and $C(p)$ in a parameter affine form:

$$A(p) = p_0 \otimes A_0 \oplus \ldots \oplus p_L \otimes A_L = \bigoplus_{l=0}^{L} p_l \otimes A_l, \tag{3.26}$$

$$B(p) = p_0 \otimes B_0 \oplus \ldots \oplus p_L \otimes B_L = \bigoplus_{l=0}^{L} p_l \otimes B_l, \tag{3.27}$$

$$C(p) = p_0 \otimes C_0 \oplus \ldots \oplus p_L \otimes C_L = \bigoplus_{l=0}^{L} p_l \otimes C_l, \tag{3.28}$$

with $l = 1, \ldots, L$, and A_l, B_l and C_l are matrices whose elements are either e or ε. Again, p will represent the parameter vector with elements:

$$p = [p_0, p_1, \ldots, p_l], \tag{3.29}$$

$$p_0 = e, \tag{3.30}$$

$$p_i > 0 \ \text{ for } \ i = \{1, \ldots, l\} \tag{3.31}$$

In the following subsection, we will give an example of a max-plus affine parametric description.

Case study: MP-LPV with varying parameter in max-plus algebra

A case where affine parameter dependence comes forward is proposed in [9] through a two-inputs, one-output production system with three machines. The first two machines process different parts and the third machine makes one piece out of it, see [9, Figure 1]. Define the variables as:

- $u_i(k)$: time instants at which the k-th part is fed to machine i for $i = \{1, 2\}$,
- $x_i(k)$: time instant at which the k-th parts enter machine i for $i = \{1, 2, 3\}$,
- $y(k)$: time instant at which machine 3 outputs the k-th part,
- $d_i(k)$: processing times of part k in machine i for $i = \{1, 2, 3\}$,

with $u_i(k)$ the input of the system, $x_i(k)$ the states, $y(k)$ the output, and $d_i(k)$ the varying parameter. The constraints on the system are defined as:

1. The machines can process one part at the time.
2. The machines start processing directly when the parts enter the machine.

Now we can describe these as:

$$x_1(k+1) = d_1(k) \otimes x_1(k) \oplus u_1(k+1), \tag{3.32}$$

$$x_2(k+1) = d_2(k) \otimes x_2(k) \oplus u_2(k+1), \tag{3.33}$$

$$x_3(k+1) = d_1(k+1) \otimes x_1(k+1) \oplus d_2(k+1) \otimes x_2(k+1) \oplus y(k), \tag{3.34}$$

$$y(k) = d_3(k) \otimes x_3(k) \tag{3.35}$$

which can be written in the state-space form as:

$$x(k+1) = A(k) \otimes x(k) \oplus B(k) \otimes u(k+1) \tag{3.36}$$

$$y(k) = C(k) \otimes x(k), \tag{3.37}$$

with system matrices being:

$$A(k) = \begin{pmatrix} d_1(k) & \varepsilon & \varepsilon \\ \varepsilon & d_2(k) & \varepsilon \\ d_1(k) \otimes d_1(k+1) & d_2(k) \otimes d_2(k+1) & d_3(k) \end{pmatrix}, \tag{3.38}$$

$$B(k) = \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \\ d_1(k+1) & d_2(k+1) \end{pmatrix}, \tag{3.39}$$

$$C(k) = \begin{pmatrix} \varepsilon & \varepsilon & d_3(k) \end{pmatrix}. \tag{3.40}$$

We can clearly see that the system matrices contain a varying parameter that is at least dependent on the event counter. Also, it is easy to see the parameter affine form described in (3.26), (3.27) and (3.28) attains in these system matrices. Look for example at matrix $B(k)$:

$$B(k) = e \otimes \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \\ \varepsilon & \varepsilon \end{pmatrix} \oplus d_1(k+1) \otimes \begin{pmatrix} \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ e & \varepsilon \end{pmatrix} \oplus d_2(k+1) \otimes \begin{pmatrix} \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & e \end{pmatrix} \tag{3.41}$$

with $L = 3$, and $p = [e, d_1(k+1), d_2(k+1)]$.

Remark 1. *It is not difficult to write this production system in the most general form* (3.8)*. The only change we have to make is add the output to the state vector, hence define:* $y(k) = x_4(k)$*. The equation would then become:*

$$x(k) = A_0(k) \otimes x(k) \oplus A_1(k) \otimes x(k-1) \oplus B \otimes u(k) \tag{3.42}$$

with the system matrices:

$$A_0(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ d_1(k) & d_2(k) & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & d_3(k) & \varepsilon \end{pmatrix}, \tag{3.43}$$

$$A_1(k) = \begin{pmatrix} d_1(k-1) & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & d_2(k-1) & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & d_3(k-1) & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \tag{3.44}$$

$$B = \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix} \tag{3.45}$$

Remark 2. *Secondly, notice that with a simple change in assumptions we can create a varying parameter that is state dependent. Instead of saying that the process time is counter dependent, you can say that the process time is defined as follows:*

$$d_i(k) = x_i(k) - x_{i+2}(k-1) + \beta_i, \quad \textit{for } i = 1, \tag{3.46}$$
$$d_j(k) = x_j(k) - x_{j+1}(k-1) + \beta_j, \quad \textit{for } j = 2, \tag{3.47}$$

with $\beta_i, \beta_j \in \mathbb{R}_\varepsilon$ being some constant time unit the part needs to travel from machine i, j to the final machine. This means that the process time is the difference between the time a part enters one of the two first machines and the time it enters the final machine. In this way we have created an implicit, state-dependent varying parameter that resembles the varying parameter in the urban railway example [11].

3.3 Relation MP-LPV with MMPS functions

In this section, we discuss the relation between MMPS, max-plus-scaling (MPS), and MP-LPV systems. It is possible, although under assumptions, to write MP-LPV into MMPS functions and the other way around. The relations between the models are depicted in **Figure 3.2**.

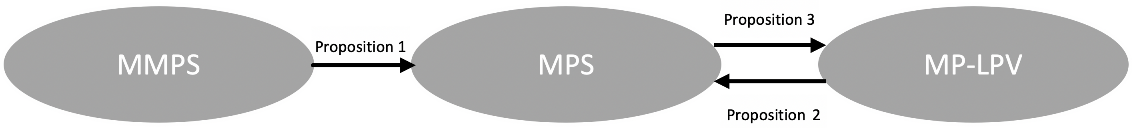


Figure 3.2: Model classes with the proposition that states the relation

3.3.1 Proposition 1: MMPS to MPS

Any MMPS system:

$$x(k) = f(x(k), x(k-1), \ldots, x(k-P)), \tag{3.48}$$

with P being any integer greater than 0 and f a convex function, can be written as an MPS system:

$$x_i(k) = \max_j(l_{ij}(k)), \tag{3.49}$$

where $l_{ij}(k)$ is an affine expression:

$$l_{ij}(k) = \sum_{p=0}^{P} \alpha_{ijp} x(k-p) + \beta_{ij} \tag{3.50}$$

Proof. An MMPS function is defined by the following recursive grammar [17][3]:

$$f(x) := x_i \mid \alpha \mid \max(f_k(x); f_l(x)) \mid \min(f_k(x); f_l(x)) \mid f_k(x) + f_l(x) \mid \beta f_k(x) \tag{3.51}$$

[3]For readability Definition 2.7.1 is repeated.

with $i \in \{1, \ldots, n\}$, $\alpha, \beta \in \mathbb{R}$, and where $f_k : \mathbb{R}^n \to \mathbb{R}^m$, $f_l : \mathbb{R}^n \to \mathbb{R}^m$ are both again MMPS functions.

From [17, Lemma 2.4]: Any MMPS function is also a continuous piecewise-affine (PWA) function.

According to [17, Theorem 3.1] a scalar-valued MMPS function f can be rewritten into the min–max canonical form:

$$f = \min_{i=1,\ldots,K} \max_{j=1,\ldots,n_i} (\alpha_{(i,j)}^T x + \beta_{(i,j)}) \tag{3.52}$$

for some integers $K, n_1, \ldots, n_K$, vector $\alpha_{(i,j)}$ and real numbers $\beta_{(i,j)}$.

From [18, Proposition 2.6], and [15, Proposition 2.5.3] we understand that a convex MMPS function only includes the max expression if the function is a polyhedral and its domain $dom(f)$ is a polyhedral set. This means we can write the previous canonical form as:

$$f = \max_{j=1,\ldots,n_l} (\alpha_j^T x + \beta_j) \tag{3.53}$$

which holds componentwise for vector-valued MMPS functions. □

3.3.2 Proposition 2: MP-LPV to MPS

Any MP-LPV system:

$$x(k) = \bigoplus_{p=0}^{P} A_p(\rho(k)) \otimes x(k-p) \tag{3.54}$$

where there is an affine dependence of A_p on the varying parameter $\rho(k)$ as follows:

$$A_p(k) = A_0^p + \sum_{l=1}^{L} \rho_l^p A_l^p \tag{3.55}$$

with

$$[A_l^p]_{ij} = \begin{cases} \alpha_{ij}^p & \text{if } j = l, \\ 0 & \text{elsewhere}, \end{cases} \tag{3.56}$$

$$[A_0^p]_{ij} = \beta_{ij}^p, \tag{3.57}$$

$$\rho_l^p = x_l(k-p) - x_l(k-p-1) \tag{3.58}$$

A system of this form can be written as an MPS system:

$$x_i(k) = \max_{j,r}(l_{ij}^r(k)), \tag{3.59}$$

where $l_{ij}(k)$ is an affine expression:

$$l_{ij}^r(k) = \sum_{r=0}^{R} \gamma_{ij}^r x(k-r) + \zeta_{ij}^r \tag{3.60}$$

Proof. Assume an MP-LPV system defined as above. Without loss of generality take $P = 0, n = 2$, and $L = 2$. This will give us a system of the form:

$$x(k) = A_0(\rho(k)) \otimes x(k), \tag{3.61}$$

with the matrix A_0 and varying parameter $\rho(k)$ defined as above. This will translate into:

$$\begin{aligned} A_0(\rho(k)) &= A_0^0 + \rho_1^0 A_1^0 + \rho_2^0 A_2^0 & (3.62)\\ &= \begin{pmatrix} \beta_{11}^0 & \beta_{12}^0 \\ \beta_{21}^0 & \beta_{22}^0 \end{pmatrix} + \begin{pmatrix} \alpha_{11}^0 & 0 \\ \alpha_{21}^0 & 0 \end{pmatrix} \rho_1^0 + \begin{pmatrix} 0 & \alpha_{12}^0 \\ 0 & \alpha_{22}^0 \end{pmatrix} \rho_2^0 & (3.63)\\ &= \begin{pmatrix} \alpha_{11}^0(x_1(k) - x_1(k-1)) + \beta_{11}^0 & \alpha_{12}^0(x_2(k) - x_2(k-1)) + \beta_{12}^0 \\ \alpha_{21}^0(x_1(k) - x_1(k-1)) + \beta_{21}^0 & \alpha_{22}^0(x_2(k) - x_2(k-1)) + \beta_{22}^0 \end{pmatrix} & (3.64) \end{aligned}$$

Plugging this matrix into Eq. (3.61), and writing this in conventional algebra will give us:

$$\begin{aligned} x_1(k) &= \max\left((1 + \alpha_{11}^0)x_1(k) - \alpha_{11}^0 x_1(k-1) + \beta_{11}^0, \quad (1 + \alpha_{12}^0)x_2(k) - \alpha_{12}^0 x_2(k-1) + \beta_{12}^0\right), \\ x_2(k) &= \max\left((1 + \alpha_{12}^0)x_1(k) - \alpha_{12}^0 x_2(k-1) + \beta_{12}^0, \quad (1 + \alpha_{22}^0)x_2(k) - \alpha_{22}^0 x_2(k-1) + \beta_{22}^0\right) \end{aligned}$$

This can be summarized by the following equation:

$$x_i(k) = \max_{j,p} \left((1 + \alpha_{ij}^p)x_j(k) - \alpha_{ij}^p x_j(k-1) + \beta_{ij}^p \right) \tag{3.65}$$

This equation is the same as the MPS system defined by Eq. (3.59) and (3.60), hence an MMPS function with only affine expressions. □

3.3.3 Proposition 3: MPS to MP-LPV

Given the following MPS system:

$$x_i(k) = \max_j(l_{ij}(k)), \tag{3.66}$$

where $l_{ij}(k)$ is an affine expression:

$$l_{ij}(k) = \sum_{p=0}^{P} a_{ijp} x(k-p) + \beta_{ijp} \tag{3.67}$$

Now define the following affine expressions:

$$l_{ij}^p(k) = \sum_{m=p}^{P} a_{ijmp} x_j(k-m) + \beta_{ij}, \quad \text{with} \quad p = 0, \ldots, P \tag{3.68}$$

where $a_{ijpp} \neq 0$.
Define the following matrix

$$A_p(k) = \begin{pmatrix} l_{11}^p(k) & l_{12}^p(k) & \ldots & l_{1n}^p(k) \\ l_{21}^p(k) & l_{22}^p(k) & \ldots & l_{2n}^p(k) \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1}^p(k) & l_{n2}^p(k) & \ldots & l_{nn}^p(k) \end{pmatrix} - \bar{1}x^T(k-p) \tag{3.69}$$

where $A_p(k) \in \mathbb{R}^{n\times n}$, $x(k-p) \in \mathbb{R}^n$ and $p = 0, \ldots, P$. Then we can write:

$$x(k) = \bigoplus_{p=0}^{P} A_p(k) \otimes x(k-p) \tag{3.70}$$

and $A_p(k)$ has no functional dependence on $x(k-m)$ for $m < p$.

Proof. The system will then look as such:

$$x_i(k) = \max_{j,p}(l_{ijp}(k)), \tag{3.71}$$

Take $P = 1$ and $n = 2$.

$$x_1(k) = \max(l_{110}(k), l_{120}(k), l_{111}(k), l_{121}(k)) \tag{3.72}$$

with

$$l_{110}(k) = a_{1100}x_1(k) + a_{1110}x_1(k-1) + b_{110} \tag{3.73}$$
$$l_{120}(k) = a_{1200}x_2(k) + a_{1210}x_2(k-1) + b_{120} \tag{3.74}$$
$$l_{111}(k) = a_{1111}x_1(k-1) + b_{111} \tag{3.75}$$
$$l_{121}(k) = a_{1211}x_2(k-1) + b_{121} \tag{3.76}$$

We can rewrite Eq. (3.72) as

$$x_1(k) = \max\Big(\max\left(l_{110}(k), l_{120}(k)\right), \max\left(l_{111}(k), l_{121}(k)\right)\Big) \tag{3.77}$$

Now, we can repeat this process for x_2 and write

$$x_2(k) = \max\Big(\max\left(l_{210}(k), l_{220}(k)\right), \max\left(l_{211}(k), l_{221}(k)\right)\Big) \tag{3.78}$$

This can be rewritten into:

$$\begin{aligned}
\begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} &= \begin{pmatrix} l_{110}(k) \oplus l_{120}(k) \\ l_{110}(k) \oplus l_{110}(k) \end{pmatrix} \oplus \begin{pmatrix} l_{111}(k) \oplus l_{121}(k) \\ l_{211}(k) \oplus l_{221}(k) \end{pmatrix} \\
&= \begin{pmatrix} (a_{1100}-1)x_1(k) + a_{1110}x_1(k-1) + b_{110} & (a_{1200}-1)x_2(k) + a_{1210}x_2(k-1) + b_{120} \\ a_{2100}-1)x_1(k) + a_{2110}x_1(k-1) + b_{210} & (a_{2200}-1)x_2(k) + a_{2210}x_2(k-1) + b_{220} \end{pmatrix} \\
&\quad \otimes x(k) \oplus \begin{pmatrix} (a_{1111}-1)x_1(k-1) + b_{111} & (a_{1211}-1)x_2(k-1) + b_{121} \\ (a_{2111}-1)x_1(k-1) + b_{211} & (a_{2211}-1)x_2(k-1) + b_{221} \end{pmatrix} \otimes x(k-1) \\
&= A_0(k) \otimes x(k) \oplus A_1(k) \otimes x(k-1)
\end{aligned}$$

For $P = 2$ and $i = j = 2$, we can repeat the above process:

$$x_i(k) = \max(l_{i10}(k), l_{i20}(k), l_{i11}(k), l_{i21}(k), l_{i12}(k), l_{i22}(k)) \tag{3.79}$$

with

$$
\begin{aligned}
l_{i10}(k) &= a_{i100}x_1(k) + a_{i110}x_1(k-1) + a_{i120}x_1(k-2) + b_{i10} && (3.80)\\
l_{i20}(k) &= a_{i200}x_2(k) + a_{i210}x_2(k-1) + a_{i220}x_2(k-2) + b_{i20} && (3.81)\\
l_{i11}(k) &= a_{i111}x_1(k-1) + a_{i121}x_1(k-2) + b_{i11} && (3.82)\\
l_{i21}(k) &= a_{i211}x_2(k-1) + a_{i221}x_2(k-2) + b_{i21} && (3.83)\\
l_{i12}(k) &= a_{i122}x_1(k-2) + b_{i12} && (3.84)\\
l_{i22}(k) &= a_{i222}x_2(k-2) + b_{i22} && (3.85)
\end{aligned}
$$

Again we can write this as:

$$
\begin{aligned}
\begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} &= \begin{pmatrix} l_{110}(k) \oplus l_{120}(k) \\ l_{110}(k) \oplus l_{110}(k) \end{pmatrix} \oplus \begin{pmatrix} l_{111}(k) \oplus l_{121}(k) \\ l_{211}(k) \oplus l_{221}(k) \end{pmatrix} \oplus \begin{pmatrix} l_{112}(k) \oplus l_{122}(k) \\ l_{212}(k) \oplus l_{222}(k) \end{pmatrix}\\
&= A_0(k) \otimes x(k) \oplus A_1(k) \otimes x(k-1) \oplus A_2(k) \otimes x(k-2)
\end{aligned}
$$

Repeat this process for $P = P$ and $i = j = 2$:

$$
x_i(k) = \max(l_{i10}(k), l_{i20}(k), l_{i11}(k), l_{i21}(k), \ldots, l_{i1(P-1)}(k), l_{i2(P-1)}(k), l_{i1P}(k), l_{i2P}(k)) \quad (3.86)
$$

with

$$
\begin{aligned}
l_{i10}(k) &= a_{i100}x_1(k) + a_{i110}x_1(k-1) + \ldots + a_{i1P0}x_1(k-P) + b_{i10} && (3.87)\\
l_{i20}(k) &= a_{i200}x_2(k) + a_{i210}x_2(k-1) + \ldots + a_{i2P0}x_2(k-P) + b_{i20} && (3.88)\\
l_{i11}(k) &= a_{i111}x_1(k-1) + a_{i121}x_1(k-2) + \ldots + a_{i1P1}x_1(k-P) + b_{i11} && (3.89)\\
l_{i21}(k) &= a_{i211}x_2(k-1) + a_{i221}x_2(k-2) + \ldots + a_{i2P1}x_2(k-P) + b_{i21} && (3.90)\\
&\vdots\\
l_{i1P}(k) &= a_{i12P}x_1(k-P) + b_{i1P} && (3.91)\\
l_{i2P}(k) &= a_{i22P}x_2(k-P) + b_{i2P} && (3.92)
\end{aligned}
$$

Again we can write this as:

$$
\begin{aligned}
\begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} &= \begin{pmatrix} l_{110}(k) \oplus l_{120}(k) \\ l_{110}(k) \oplus l_{110}(k) \end{pmatrix} \oplus \begin{pmatrix} l_{111}(k) \oplus l_{121}(k) \\ l_{211}(k) \oplus l_{221}(k) \end{pmatrix} \oplus \ldots \oplus \begin{pmatrix} l_{11P}(k) \oplus l_{12P}(k) \\ l_{21P}(k) \oplus l_{22P}(k) \end{pmatrix}\\
&= A_0(k) \otimes x(k) \oplus A_1(k) \otimes x(k-1) \oplus \ldots \oplus A_P(k) \otimes x(k-P)\\
&= \bigoplus_{p=0}^{P} A_p(k) \otimes x(k-p)
\end{aligned}
$$

□

3.4 Summary

In this chapter, we have seen the introduction of a general framework of MP-LPV systems and the relations of MP-LPV systems with MMPS functions. We started with a brief summary of

LPV systems and applied this notion in the MPL framework. Combining these two we have proposed a general framework for MP-LPV systems. We have also given two (possibly the only two) MP-LPV examples from literature and explained how they fit in our framework. We concluded the chapter with propositions that show the relations of MP-LPV systems with MMPS functions. Three propositions are given and proved.

In the next chapter, we will look at a graph-theoretical method to look at solvability of systems. This method is applied to explicit and implicit systems to look at the similarities and differences. This analysis will be used in Chapter 7 to see if our case study indeed has the properties of a structurally solvable system.

Chapter 4

Solvability: Graph theoretical approach

In this chapter, solvability of max-plus linear parameter varying (MP-LPV) systems will be discussed. In the first part, graph theoretic approaches to analyse the solvability of these systems will be discussed. Graph representations have shown to be efficient in testing structural solvability, detecting inconsistencies in models, and decomposition of a system into subsystems [26]. In Section 2.3, an introduction to graph theory concerning max-plus algebra was given. In this chapter, the focus will be less narrow, as graphical representations of systems have been used in engineering for all kind of purposes, such as flow-charts in computer science, block diagrams, signal-flow graphs of control systems, transportation networks, etc. These representations were used as a tool to structurally analyse systems. Not all of them are mathematically well supported or useful for our research. In Section 4.1, we discuss the mathematical formulation of the structure of graphs that correlate with nonlinear systems that have a unique solution. Then, this theory is applied to MP-LPV systems, where a difference is made in the analysis of explicit and implicit systems.

4.1 Structural solvability in representation graphs

Following [27], consider a system of nonlinear equations of the standard form defined as:

$$\begin{cases} y_i = f_i(x, u) \;\; i = \{1, \cdots, M\}, \\ x_r = g_r(x, u) \;\; r = \{1, \cdots, R\} \end{cases} \tag{4.1}$$

where $u_j (j = \{1, \cdots, N\})$ and $x_r (r = \{1, \cdots, R\})$ are unknowns and $y_i (i = \{1, \cdots, M\})$ are parameters. Furthermore, it is assumed that f_i and g_r are sufficiently smooth real-valued functions. Noticeable is the fact that this standard form allows both explicit and implicit form to be present in the set of equations.

The system (4.1) is said to be *structurally solvable* if its Jacobian matrix $J(x, u)$ is nonsingular. Or in other words if:

$$\det J(x,u) \neq 0, \quad \text{where } J(x,u) = \begin{pmatrix} \dfrac{\partial f_i}{\partial u_l} & \dfrac{\partial f_i}{\partial x_j} \\ \dfrac{\partial g_r}{\partial u_l} & \dfrac{\partial g_r}{\partial x_j} - I_r \end{pmatrix}. \tag{4.2}$$

However, this condition depends on both the functional forms of f_i and g_r and on the particular value of the possible solution of (4.1). This could become difficult because before you start a computation you usually do not know what the solution will be. Next to that, there are numerical difficulties with distinguishing between zero and a number extremely close to zero.

Belonging to this system of equations (4.1) is a *representation graph* $\mathcal{G}(\mathcal{N}, \mathcal{D})$, defined as a graph that shows functional dependencies between unknowns and parameters. Consider the set of nodes $\mathcal{N} = X \cup Y \cup U$, where $X = \{x_1, \cdots, x_R\}$, $Y = \{y_1, \cdots, y_M\}$, and $U = \{u_1, \cdots, u_N\}$. In the graph, the functional dependence expressed as $y_i = f_i(x,u)$ is a set of arcs going from u_j and x_l into y_i, idem for x_r. Moreover, it is assumed that there are no arcs that go into $u_j \in U$ or come out of $y_i \in Y$.

Example 4.1.1. Consider the following system of equations [27]:

$$y_1 = f_1(u_1) \tag{4.3}$$
$$y_2 = f_2(x_1, x_3) \tag{4.4}$$
$$y_3 = f_3(u_2, x_3, x_4) \tag{4.5}$$
$$x_1 = g_1(u_1, x_2) \tag{4.6}$$
$$x_2 = g_2(u_1, u_2, x_3) \tag{4.7}$$
$$x_3 = g_3(x_1) \tag{4.8}$$
$$x_4 = g_4(u_2, u_3, x_3) \tag{4.9}$$

This system of equations corresponds with the representation graph shown in **Figure 4.1** ◁

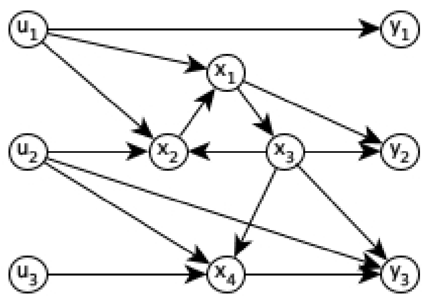


Figure 4.1: Representation graph of system of equations proposed in Example 4.1.1

A *Menger-type linking* [27] from U to Y is visualized by a set of *pair-wise vertex-disjoint directed paths* from a vertex in U to a vertex in Y. The size of a linking is the number of directed paths from U to Y contained in the linking. In case $|U| = |Y|$ (with $M = N$), a linking of size[1] $|U|$ is called a complete linking, where $|.|$ is the cardinality of a set, i.e. the number of elements in a set. The graphical condition of the structural solvability is then the given by the following theorem:

Theorem 4.1.1. *Linkage theorem [27]. When a system of equations is written in the standard form (see Eq.* (4.1)*), it is said to be structurally solvable iff a Menger-type complete linking exists from U to Y on the graph.*

Looking back at Example 4.1.1 we can apply this theorem and see whether the system of equations is structurally solvable. We see that there exists a Menger-type complete linking in the representation graph for

$$u_1 \to y_1, u_2 \to x_2 \to x_1 \to y_2, u_3 \to x_4 \to y_3.$$

Hence, we may conclude that this system is structurally solvable. Let us now regard an example that is not structurally solvable.

Example 4.1.2. Considering the following system of equations [27]:

$$y_1 = f_1(x_1, x_3), \tag{4.10}$$
$$y_2 = f_2(x_2, x_3), \tag{4.11}$$
$$y_3 = f_3(x_2), \tag{4.12}$$
$$x_1 = g_1(u_1, u_2, x_2), \tag{4.13}$$
$$x_2 = g_2(u_2, u_3, x_3), \tag{4.14}$$
$$x_3 = g_3(x_1) \tag{4.15}$$

This system of equations corresponds with the representation graph shown in **Figure 4.2**. In the graph it can be seen that there is a path from $U = \{u_1, u_2, u_3\}$ to any $Y = \{y_1, y_2, y_3\}$.

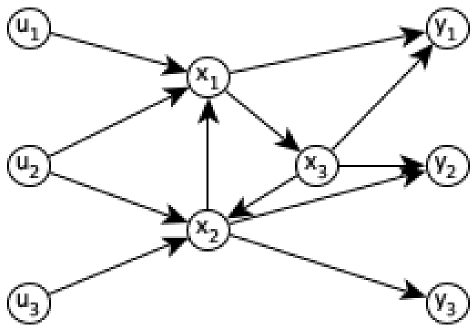


Figure 4.2: Representation graph of system of equations proposed in Example 4.1.2

But it is not difficult to see that the size of a maximum linking from U to Y is 2, which makes it not complete (should be 3). As the Menger-type complete linking does not exist, it is concluded that this system of equations is not structurally solvable.

[1]In this context size represents the number of Menger-type linkings.

Intuitively speaking this would mean that we have three degrees of freedom in our input, but this is reduced to two values x_1 and x_2. In other words, in the representation graph the set $\{x_1, x_2\}$ is a minimum separator of (U, Y). The number of minimum separators in the representation graph can be interpreted as the *effective* degrees of freedom of the system. A minimum separator in a system can expose where certain inconsistencies, if they are present, come from. ◁

In the next section, we will apply this notion to our framework of MP-LPV systems. A difference will be made between analyzing explicit and implicit cases. We start with the explicit cases, as we will notice that these cases are less complex to represent and apply Menger's theorem on.

4.2 Structural solvability of MP-LPV systems

To apply this notion of structural solvability while using representation graphs for MP-LPV systems discussed in Chapter 3, we need to start separating between different cases that arise. Before we do that let us simplify the most general case given in Eq. (3.8) with the following assumptions:

Assumption A8. *Let us assume that the size of the input $|U|$ is always sufficient, such that we can analyse the system as if it is autonomous.*

Assumption A9. *Let us only look at systems that have matrix A_μ in (3.8) in the following form:*

$$A_\mu = \begin{pmatrix} a_{1,1} & \varepsilon & \cdots & \cdots & \varepsilon \\ a_{2,1} & a_{2,2} & \ddots & & \varepsilon \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \varepsilon \\ a_{n,1} & a_{n,2} & \cdots & \cdots & a_{n,n} \end{pmatrix} \tag{4.16}$$

where $a_{i,j}$ with $i, j \in \{1, 2, \ldots, n\}$ are assumed to be scalars.

Assumption A10. *We will only go as far as looking at one previous cycle, in Eq. (3.8) this would mean that $\mu = \{0, 1\}$. We can make this assumption due to the chosen structure in **Assumption A9**. Graph theoretically this structure will lead to a repeating form.*

For our research the importance lies heavily on the influence of the varying parameter on the solvabilty. Without loss of generality of the analysis it is possible to make these simplifications:

Assumption A11. *In the implicit case the varying parameter $p(k)$ will only be in A_0 matrix. This is possible since adding a varying parameter will only add edges (as $p(k)$ is a function of the state), and adding edges can never break structural solvability, but it will make the representation graphs bigger and less clear.*

Assumption A12. *The varying parameter $p(k)$ will be an affine function in the state x.*

Analyzing the structural solvability of a system with matrices in this form forces us to make different cases. We will differ between explicit and implicit cases, see Eq. (3.7) and (3.8).

4.2.1 Structural solvability: Explicit case

As can be seen in Eq. (3.7) the A_0 matrix is non-existent in the explicit case. It is important to note that the parameter could lead to an implicit relation. Taking into account **Assumption A10**, the following explicit cases can be made:

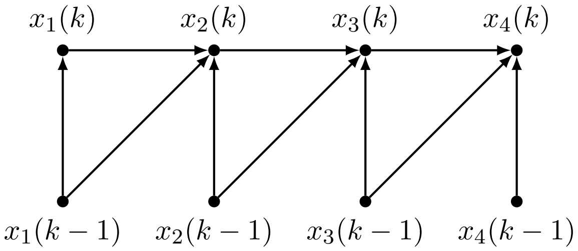
- Explicit I:
$$x(k) = A_1(p(k)) \otimes x(k-1), \tag{4.17}$$
with $p_i(k) = g(x_i(k-1),\ \ x_{i-1}(k-1))$.
- Explicit II:
$$x(k) = A_1(p(k)) \otimes x(k-1), \tag{4.18}$$
with $p_i(k) = g(x_{i-1}(k),\ \ x_{i-1}(k-1))$.
- Explicit III:
$$x(k) = A_1(p(k)) \otimes x(k-1), \tag{4.19}$$
with $p_i(k) = g(x_i(k-1),\ \ x_i(k-2))$.
- Explicit IV
$$x(k) = A_1(p(k)) \otimes x(k-1), \tag{4.20}$$
with $p_i(k) = g(x_i(k-1),\ \ x_{i-1}(k))$.
- Explicit V
$$x(k) = A_1(p(k)) \otimes x(k-1), \tag{4.21}$$
with $p_i(k) = g(x_i(k),\ \ x_i(k-1))$.

Important is to understand is that mostly there is no trouble in analyzing the structural solvability of these cases. Trouble arises when implicitness is added to the equation, e.g. Explicit V. Also in the next section, it can be seen that some implicit cases can be related to the explicit case. This could mean that it is either possible to rewrite it into an explicit case or that the case is an extended version of the explicit case. In **Table 4.1** we summarize all findings.

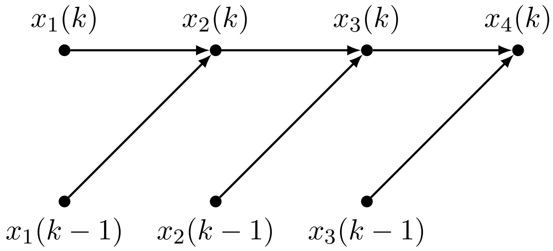
In **Figure 4.3**, all representation graphs for the corresponding explicit forms discussed above can be seen.

Cases	Varying parameter $p(k)$	Implicitness in p(k)	Structural solvability with Menger's th.
Explicit I	Yes	No	Yes
Explicit II	Yes	Yes	Yes
Explicit III	Yes	No	Yes
Explicit IV	Yes	No	Yes
Explicit V	Yes	Yes	No

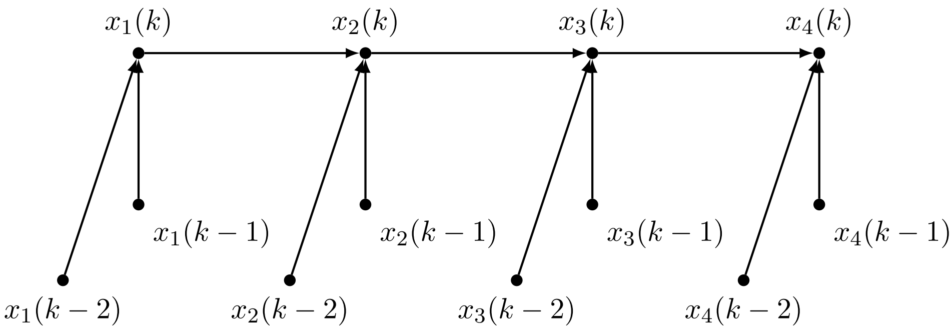
Table 4.1: Summary of all findings on the explicit cases



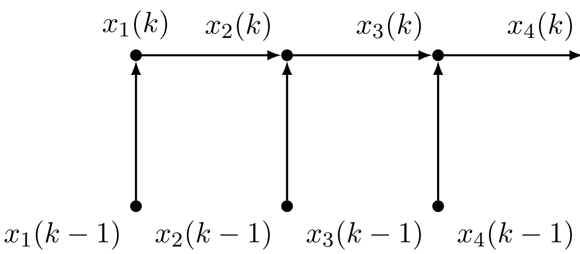
(a) Explicit I with varying function



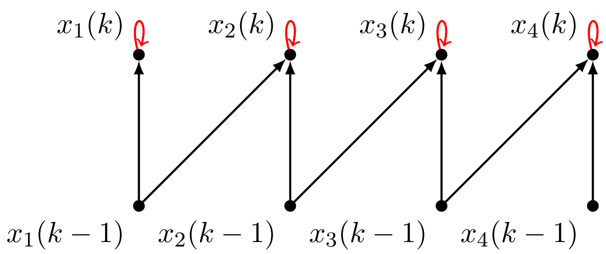
(b) Explicit II with varying function



(c) Explicit III with varying function



(d) Explicit IV, with varying function and no self-loops



(e) Explicit V, with varying function and self-loops

Figure 4.3: Representation graphs of different explicit cases. In Figure (e), the problematic self-loops have been drawn in red.

4.2.2 Structural solvability: Implicit case

In the implicit cases, the procedure is the same. Only now the matrix A_0 is existent, hence there is an implicit relation without the varying parameter intervening. Assume for simplicity that the varying parameter is only, if present, present in the A_0 matrix. Let us look at the following case, where there is no varying parameter:

$$x(k) = A_0 \otimes x(k) \oplus A_1 \otimes x(k-1). \tag{4.22}$$

with $A_0, A_1 \in \mathbb{R}_\varepsilon^{n\times n}$, and $x(k) \in \mathbb{R}_\varepsilon^n$. To have consistency we will assume that the structure of the A_0 and A_1 matrices as follows:

$$A_0 = \begin{pmatrix} \varepsilon & \varepsilon & \cdots & \cdots & \varepsilon \\ a_{2,1} & \varepsilon & \ddots & & \varepsilon \\ a_{3,1} & a_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \varepsilon \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n-1} & \varepsilon \end{pmatrix}, \quad A_1 = \begin{pmatrix} a_{1,1} & \varepsilon & \cdots & \cdots & \varepsilon \\ a_{2,1} & a_{2,2} & \ddots & & \varepsilon \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \varepsilon \\ a_{n,1} & a_{n,2} & \cdots & \cdots & a_{n,n} \end{pmatrix} \tag{4.23}$$

Without knowing too much about the actual system we can conclude the following functional dependencies:

$$x_i(k) = f\big(x_1(k),\ x_2(k), \cdots,\ x_{n-1}(k),\ x_1(k-1),\ x_2(k-1), \cdots,\ x_n(k-1)\big) \tag{4.24}$$

To make the representation graphs not too complex we will relate it to the running example proposed in Section 2.5.

Example 4.2.1. To make it easy to read, let us repeat the equations:

$$a_j(k) = d_{j-1}(k) \otimes \tau_{r,\min,j-1} \oplus d_j(k-1) \otimes \tau_{h,\min} \tag{4.25}$$
$$d_j(k) = a_j(k) \otimes \tau_{d,j}, \tag{4.26}$$

for $j \in \{1,2,3,4\}$. By eliminating the arrival times this can be written as:

$$d_j(k) = d_{j-1}(k) \otimes \tau_{r,\min,j-1} \otimes \tau_{d,j} \oplus d_j(k-1) \otimes \tau_{h,\min} \otimes \tau_{d,j} \tag{4.27}$$

Then for a railway network with $j = \{1,2,3,4\}$ we can write

$$d_1(k) = d_0(k) \otimes e \otimes \tau_{d,1} \oplus d_1(k-1) \otimes \tau_{\mathrm{h,min}} \otimes \tau_{d,1} \tag{4.28}$$
$$d_2(k) = d_1(k) \otimes \tau_{\mathrm{r,min,1}} \otimes \tau_{d,2} \oplus d_2(k-1) \otimes \tau_{\mathrm{h,min}} \otimes \tau_{d,2} \tag{4.29}$$
$$d_3(k) = d_2(k) \otimes \tau_{\mathrm{r,min,2}} \otimes \tau_{d,3} \oplus d_3(k-1) \otimes \tau_{\mathrm{h,min}} \otimes \tau_{d,3} \tag{4.30}$$
$$d_4(k) = d_3(k) \otimes \tau_{\mathrm{r,min,3}} \otimes \tau_{d,4} \oplus d_4(k-1) \otimes \tau_{\mathrm{h,min}} \otimes \tau_{d,4}. \tag{4.31}$$

Say that $x(k) = d(k)$ such that we can write:

$$x_1(k) = x_0(k) \otimes \tau_{d,1} \oplus x_1(k-1) \otimes \tau_{\mathrm{h,min}} \otimes \tau_{d,1} \tag{4.32}$$
$$x_2(k) = x_1(k) \otimes \tau_{\mathrm{r,min,1}} \otimes \tau_{d,2} \oplus x_2(k-1) \otimes \tau_{\mathrm{h,min}} \otimes \tau_{d,2} \tag{4.33}$$
$$x_3(k) = x_2(k) \otimes \tau_{\mathrm{r,min,2}} \otimes \tau_{d,3} \oplus x_3(k-1) \otimes \tau_{\mathrm{h,min}} \otimes \tau_{d,3} \tag{4.34}$$
$$x_4(k) = x_3(k) \otimes \tau_{\mathrm{r,min,3}} \otimes \tau_{d,4} \oplus x_4(k-1) \otimes \tau_{\mathrm{h,min}} \otimes \tau_{d,4}. \tag{4.35}$$

The system can then be written as follows:[2]

$$x(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \tau_{r,1} \otimes \tau_{d,2} & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \tau_{r,2} \otimes \tau_{d,3} & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \tau_{r,3} \otimes \tau_{d,4} & \varepsilon \end{pmatrix} \otimes x(k) \dots \tag{4.36}$$
$$\dots \oplus \begin{pmatrix} \tau_h \otimes \tau_{d,1} & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \tau_h \otimes \tau_{d,2} & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \tau_h \otimes \tau_{d,3} & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \tau_h \otimes \tau_{d,4} \end{pmatrix} \otimes x(k-1)$$

This looks quite complex, but for the representation graph only the functional dependencies are of importance, which is why we can write these dependencies for the system in Eq. (4.36) as described in the following section. ◁

Implicit I: no varying parameter

$$x_1(k) = f_1\Big(x_1(k-1)\Big) \tag{4.37}$$
$$x_2(k) = f_2\Big(x_2(k-1),\ \ x_1(k)\Big) \tag{4.38}$$
$$x_3(k) = f_3\Big(x_3(k-1),\ \ x_2(k)\Big) \tag{4.39}$$
$$x_4(k) = f_4\Big(x_4(k-1),\ \ x_3(k)\Big) \tag{4.40}$$

Looking at the representation graph in **Figure 4.4** we can see a Menger complete linking, which tells us that the system is structurally solvable by Menger's linking theorem [27]. It

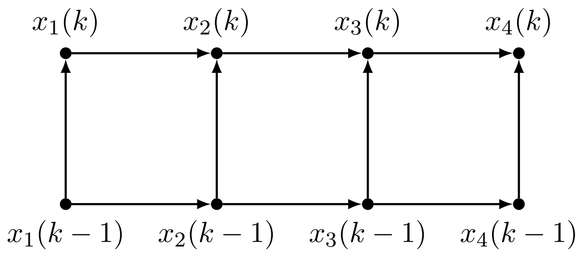


Figure 4.4: Representation graph: implicit I without varying function

should be mentioned that for keeping the representations graph simple we will assume that all coming cases have the structure of the running example:

$$A_0 = \begin{pmatrix} \varepsilon & \varepsilon & \cdots & \cdots & \varepsilon \\ a_{2,1} & \varepsilon & \ddots & & \varepsilon \\ \varepsilon & a_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \varepsilon \\ \varepsilon & \varepsilon & \cdots & a_{i,i-1} & \varepsilon \end{pmatrix}, \quad A_1 = \begin{pmatrix} a_{1,1} & \varepsilon & \cdots & \cdots & \varepsilon \\ \varepsilon & a_{2,2} & \ddots & & \varepsilon \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \varepsilon \\ \varepsilon & \varepsilon & \cdots & \cdots & a_{i,i} \end{pmatrix}. \tag{4.41}$$

Where the varying function will be present in the sub-diagonal of matrix A_0.

[2]For the sake of clarity *min* in the subscript will be omitted for both $\tau_{\mathrm{r,min},j-1}$ and $\tau_{\mathrm{h,min}}$.

Implicit II: explicit varying parameter

$$x(k) = A_0(p(k)) \otimes x(k) \oplus A_1 \otimes x(k-1), \tag{4.42}$$

with $p_i(k) = g(x_{i-1}(k),\ \ x_i(k-1))$. This might not make a lot of sense, but these relations summarize the urban railway model described in [11]. There are two differences compared to the previous case.

- There is a varying parameter present, this will be translated in a passenger dependent dwell time, hence $p(k) = \tau_d(k)$.
- The state vector will now be a vector with both the arrival times and the departure times.

Then for a system with $i = \{1, 2, 3, 4\}$ the following relations can be written:

$$x_1(k) = f_1\Big(x_1(k-1)\Big) \tag{4.43}$$

$$x_2(k) = f_2\Big(x_1(k),\ \ x_1(k-1),\ \ x_2(k-1)\Big) \tag{4.44}$$

$$x_3(k) = f_3\Big(x_2(k),\ \ x_2(k-1),\ \ x_3(k-1)\Big) \tag{4.45}$$

$$x_4(k) = f_4\Big(x_3(k),\ \ x_3(k-1),\ \ x_4(k-1)\Big) \tag{4.46}$$

Looking at the representation graph in **Figure 4.5** we can see a Menger complete linking, which tells us that the system is structurally solvable by Menger's linking theorem [27].

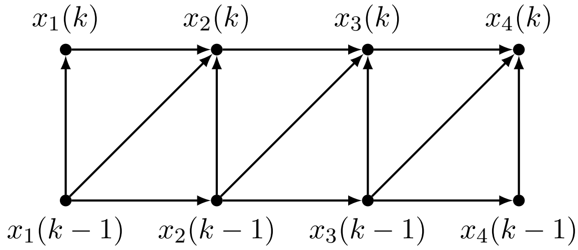


Figure 4.5: Representation graph: implicit II with varying function but no self-loops

Implicit III: implicit varying parameter

$$x(k) = A_0(p(k)) \otimes x(k) \oplus A_1 \otimes x(k-1), \tag{4.47}$$

with $p_i(k) = g(x_i(k),\ \ x_i(k-1))$. Then for a system with $i = \{1, 2, 3, 4\}$ the following relations can be written:

$$x_1(k) = f_1\Big(x_1(k-1)\Big) \tag{4.48}$$

$$x_2(k) = f_2\Big(x_1(k),\ \ x_2(k),\ \ x_1(k-1),\ \ x_2(k-1)\Big) \tag{4.49}$$

$$x_3(k) = f_3\Big(x_2(k),\ \ x_3(k),\ \ x_2(k-1),\ \ x_3(k-1)\Big) \tag{4.50}$$

$$x_4(k) = f_4\Big(x_3(k),\ \ x_4(k),\ \ x_3(k-1),\ \ x_4(k-1)\Big) \tag{4.51}$$

Looking at the representation graph in **Figure 4.6** we can see that Menger's theorem is not sufficient to draw any conclusions on structural solvability [27]. What this actually means is that it is not possible to move disjointly from the input to the output without getting stuck at one of the vertices.

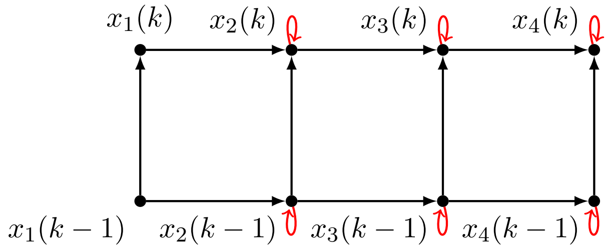


Figure 4.6: Representation graph: implicit III, with varying function and self-loops

Implicit IV: implicit varying parameter

$$x(k) = A_0(p(k)) \otimes x(k) \oplus A_1 \otimes x(k-1), \tag{4.52}$$

with $p_i(k) = g(x_i(k),\ \ x_{i-1}(k))$. Then for a system with $i = 1, 2, 3, 4$ the following relations can be written[3]:

$$x_1(k) = f_1\Big(x_1(k-1)\Big) \tag{4.53}$$

$$x_2(k) = f_2\Big(x_1(k),\ \ x_2(k),\ \ x_2(k-1)\Big) \tag{4.54}$$

$$x_3(k) = f_3\Big(x_2(k),\ \ x_3(k),\ \ x_3(k-1)\Big) \tag{4.55}$$

$$x_4(k) = f_4\Big(x_3(k),\ \ x_4(k),\ \ x_4(k-1)\Big) \tag{4.56}$$

Looking at the representation graph in **Figure 4.7** we can see the same problem pops up as for the previous case. It is not possible to conclude about structural solvability.

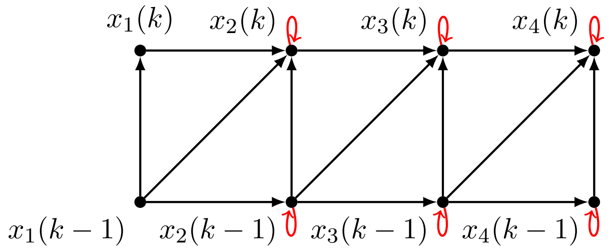


Figure 4.7: Representation graph: implicit IV, with varying function and self-loops

[3]This relation might not make sense for our urban railway example, but imagine a production system with heating and cooling down of a product in between two machines.

In the following table our findings have been summarized:

Cases	**Varying parameter** $p(k)$	**Implicitness in p(k)**	**Structural solvability with Menger's th.**
Implicit I	No	No	Yes
Implicit II	Yes	Yes	Yes
Implicit II	Yes	Yes	No
Implicit IV	Yes	No	No

Table 4.2: Summary of all findings on the implicit cases

4.3 Summary

In this chapter, the notion of structural solvability by applying Menger's theorem has been discussed. The main thing to take away is that we cannot conclude structural solvability when there is an implicit relation in the model. We have seen that having an A_0 matrix does not necessarily mean that the system has implicit relations. This completely depends on the structure of the matrix.

In the next chapter, we will look at the analytical conditions for an MP-LPV system to be solvable. We will write the system as a max-min-plus-scaling (MMPS) function and simplify such a function to such an extent that we can find conditions for which the function is solvable.

Chapter 5

Solvability: Analytical approach

Our interest in solvability must be clear by now. Up until now, we have seen that it is possible to show structural solvability for explicit systems. In the previous chapter, we concluded that it is not possible to conclude structural solvability for implicit equations, especially the ones with a varying parameter. In this chapter, we will analyze the solvability of these implicit equations analytically. Firstly, we will define the structure of the kind of system we will analyze. Then a classification of these systems is given. After this, we will conclude with the conditions under which the systems become solvable.

5.1 Preliminaries analytical approach

To analyze the solvability extending some of the assumptions proposed in Section 4.2 is necessary. We have discussed the structure of the A_μ matrices in **Assumption A9**. This form is actually named the max-plus algebraic Frobenius normal form [28]. The Frobenius normal form is defined in the following theorem:

Theorem 5.1.1 (Max-plus-algebraic Frobenius normal form)**.** *Consider $A \in \mathbb{R}_\varepsilon^{n\times n}$ then there is a max-plus permutation matrix $S \in \mathbb{R}_\varepsilon^{n\times n}$ in such a way that the following computation will yield a max-plus-algebraic block upper triangular matrix:*

$$\hat{A} = S \otimes A \otimes S^T,$$

with

$$\hat{A} = \begin{pmatrix} \hat{A}_{11} & \mathcal{E} & \dots & \mathcal{E} \\ \hat{A}_{21} & \hat{A}_{22} & \dots & \mathcal{E} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{A}_{l1} & \hat{A}_{l2} & \dots & \hat{A}_{ll} \end{pmatrix}.$$

In this case $l \geq 1$ and the matrices $\{\hat{A}_{11}, \hat{A}_{12}, \dots, \hat{A}_{ll}\}$ are both square and irreducible. The ordering of matrices $\hat{A}_{11}, \hat{A}_{12}, \dots, \hat{A}_{ll}$ does not need to be unique, but they are uniquely determined due to the simultaneous permutation of the rows and columns [28].

It is possible to apply this theorem on the model equations proposed in Eq. (2.28), assuming that there exists a max-plus permutation matrix S [28] such that we can write:

$$x'(k) = S \otimes x(k) \tag{5.1}$$

$$A'_\mu = S \otimes A_\mu \otimes S^{\otimes -1} \tag{5.2}$$

$$B' = S \otimes B \tag{5.3}$$

where $S \in \mathbb{R}_\varepsilon^{n \times n}$, $x(k) \in \mathbb{R}_\varepsilon^n$, and A'_μ will get the following structure:

$$A'_\mu = \begin{pmatrix} a_{11}^\mu & \varepsilon & \cdots & \cdots & \varepsilon \\ a_{21}^\mu & a_{22}^\mu & \ddots & & \varepsilon \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \varepsilon \\ a_{n1}^\mu & a_{n2}^\mu & \cdots & \cdots & a_{nn}^\mu \end{pmatrix} \tag{5.4}$$

with $a_{ij}^\mu \in \mathbb{R}_\varepsilon$, note that as has been mentioned in **Assumption A9** the elements of the matrix are assumed to be scalars. The form seen in Eq. (5.4) is thus called the max-plus algebraic Frobenius normal form, which is a lower triangular matrix. In the case that A is irreducible there will only be one block in the whole matrix, and hence A is already in the Frobenius form [28]. We can also apply this transformation on the model proposed in Eq. (3.8).

Assumption A13. *To simplify assume for now that we analyze systems where this transformation is possible, hence we can write:*

$$x'(k) = A'_0(p(k)) \otimes x'(k) \oplus A'_1(p(k)) \otimes x'(k-1) \oplus B'(p(k)) \otimes u'(k) \tag{5.5}$$

*with $A'_0(p(k))$, $A'_1(p(k))$, and $B'(p(k))$ all in the Frobenius normal form described in **Theorem 5.1.1**. From now on, omit the usage of the accent, i.e. $A'_0(p(k)) := A_0(p(k))$.*

By doing this it is possible to solve the recurrence relation in one direction. As you can say that

$$[A'(x'(k), p(k))]_{ij} = f(x'_m(k), p(k)) \text{ for } i \geq j \text{ and with } m \leq i \tag{5.6}$$

To simplify our analysis even further consider the input to be zero for now and specify the inter-state affine relation of the varying parameter as:

$$p(k) = \alpha(x(k) - x(k-1)) \tag{5.7}$$

This will leave us with a max-plus linear parameter varying (MP-LPV) system of the following form:

$$x(k) = \begin{pmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ \vdots \\ x_n(k) \end{pmatrix} = \begin{pmatrix} a_{11}^0 & \varepsilon & \cdots & \cdots & \varepsilon \\ a_{21}^0 & a_{22}^0 & \ddots & & \varepsilon \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \varepsilon \\ a_{n1}^0 & a_{n2}^0 & \cdots & \cdots & a_{nn}^0 \end{pmatrix} \otimes x(k) \oplus \begin{pmatrix} a_{11}^1 & \varepsilon & \cdots & \cdots & \varepsilon \\ a_{21}^1 & a_{22}^1 & \ddots & & \varepsilon \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \varepsilon \\ a_{n1}^1 & a_{n2}^1 & \cdots & \cdots & a_{nn}^1 \end{pmatrix} \otimes x(k-1), \tag{5.8}$$

where $x(k) \in \mathbb{R}_\varepsilon^n$, $A_0(p(k))$, and $A_1(p(k)) \in \mathbb{R}_\varepsilon^{n\times n}$. With regards to the varying parameter we will differentiate between two cases:

- Case 1: Varying parameter present only in A_0, hence the elements of the system matrices are defined as:

$$a_{ij}^{\mu} = \alpha_j(x_i(k) - x_i(k-1)) \quad \text{for} \quad \mu = 0, \quad \forall\ i,j \in \{1,2,\dots,n\} \tag{5.9}$$
$$a_{ij}^{\mu} = \gamma \quad \text{for} \quad \mu = 1, \quad \forall\ i,j \in \{1,2,\dots,n\} \tag{5.10}$$

  with $\alpha_j \in \mathbb{R}$, and $\gamma \in \mathbb{R}_\varepsilon$.

- Case 2: Varying parameter present in both A_0 and A_1, hence the elements are defined as:

$$a_{ij}^{\mu} = \alpha_j^{\mu}(x_i(k) - x_i(k-1)) \quad \text{for} \quad \mu = 0,1, \quad \forall\ i,j \in \{1,2,\dots,n\} \tag{5.11}$$

  with $\alpha_j^{\mu} \in \mathbb{R}$.

In the next subsection we will look more closely at these cases. We will try to rewrite the state-space model into conventional algebra and analyze the system within the conventional toolbox.

5.2 Case 1: Varying parameter in matrix A_0

Let us try to write the MP-LPV system in Eq. (5.8) in a more general form. We can do this by looking at a large number of iterations:

$$\begin{aligned}
x_1(k) &= \max\Big(a_{11} + x_1(k), x_1(k-1) + \gamma\Big) && (5.12)\\
&= \max\Big(\alpha_1(x_1(k) - x_1(k-1)) + x_1(k), x_1(k-1) + \gamma\Big) && (5.13)\\
x_2(k) &= \max\Big(a_{21} + x_1(k), a_{22} + x_2(k), x_1(k-1) + \gamma, x_2(k-1) + \gamma\Big) && (5.14)\\
&= \max\Big(\alpha_1(x_2(k) - x_2(k-1)) + x_1(k), \alpha_2(x_2(k) - x_2(k-1)) + x_2(k), \dots && (5.15)\\
&\qquad \dots x_1(k-1) + \gamma, x_2(k-1) + \gamma\Big)\\
&\ \ \vdots\\
x_i(k) &= \max\Big(a_{i1} + x_1(k), a_{i2} + x_2(k), \dots, a_{(i-1)i} + x_{i-1}(k), a_{ii} + x_i(k), \dots && (5.16)\\
&\qquad \dots x_1(k-1) + \gamma, x_2(k-1) + \gamma, \dots, x_i(k-1) + \gamma\Big).
\end{aligned}$$

Remark 3. *Due to the non-decreasing nature of systems in this framework, we will see that the diagonal elements of the A_0 matrix in some cases will be equal to ε. If this is not the case one should be aware for inconsistencies in the equations.*

If we want to write Eq. (5.16) in a more general way, it takes a couple of steps:

- First we assume that the known terms that come from matrix A_1 in Eq.(5.8), can be rewritten to one term: $l_{\max}$. This can be explained by the fact that only the maximum value of the known terms is critical for this iteration. So, we can write:

$$x_i(k) = \max\Big(a_{i1} + x_1(k), a_{i2} + x_2(k), \dots, a_{(i-1)i} + x_{i-1}(k), a_{ii} + x_i(k), l_{\max}\Big), \quad (5.17)$$

  with $l_{\max} \in \mathbb{R}_\varepsilon$.

- Rewrite Eq. (5.17):

$$x_i(k) = \max\Big(\alpha_1(x_i(k) - x_i(k-1)) + x_1(k), \alpha_2(x_i(k) - x_i(k-1)) + x_2(k), \dots \quad (5.18)$$
$$\alpha_{i-1}(x_i(k) - x_i(k-1)) + x_{i-1}(k), \alpha_i(x_i(k) - x_i(k-1)) + x_i(k), l_{\max}\Big),$$

- Now to simplify this equation and make it more general we say that $x_i(k) = y$ and $\beta_j = x_j(k) - \alpha_j x_i(k-1)$, so that we can write:

$$y = \max\Big(\alpha_1 y + \beta_1, \alpha_2 y + \beta_2, \dots, \alpha_{i-1} y + \beta_{i-1}, \alpha_i y + \beta_i, l_{\max}\Big) \quad (5.19)$$
$$= \max\Big(f_1(y), f_2(y), \dots, f_{i-1}(y), y + f_i(y), l_{\max}\Big). \quad (5.20)$$

Now we have created a form for which we can easily find analytical solutions. In the next section we will look at the conditions for solving these equations.

5.2.1 Intermezzo solving the equation: $y = \max(\alpha y + \beta, l_{\max})$

Suppose we have a simplified function of the form proposed in Eq. (5.19):

$$y = \max(\alpha y + \beta, l_1, l_2, \dots, l_n) \quad (5.21)$$
$$= \max(\alpha y + \beta, l_{\max}) \quad (5.22)$$

In **Figure 5.1** you see the possibilities in this case. It is possible to have 0, 1 or 2 solutions. When there is a solution you have to differentiate between two cases that can be made for solving this equation:

$$\Big\{\alpha \cdot l_{\max} + \beta \leq l_{\max} \rightarrow \text{solution for (5.22) is} \quad y = l_{\max} \quad (5.23)$$

$$\begin{cases} \alpha \cdot l_{\max} + \beta > l_{\max}, \quad \text{and} \\ \alpha < 1 \rightarrow \text{solution for (5.22) is} \quad y = \dfrac{\beta}{1-\alpha} \geq l_{\max} \end{cases} \quad (5.24)$$

This means that when $\alpha \cdot l_{\max} + \beta > l_{\max}$, and $\alpha > 1$ there will be no (unique) solution. This result is portrayed in the following example:

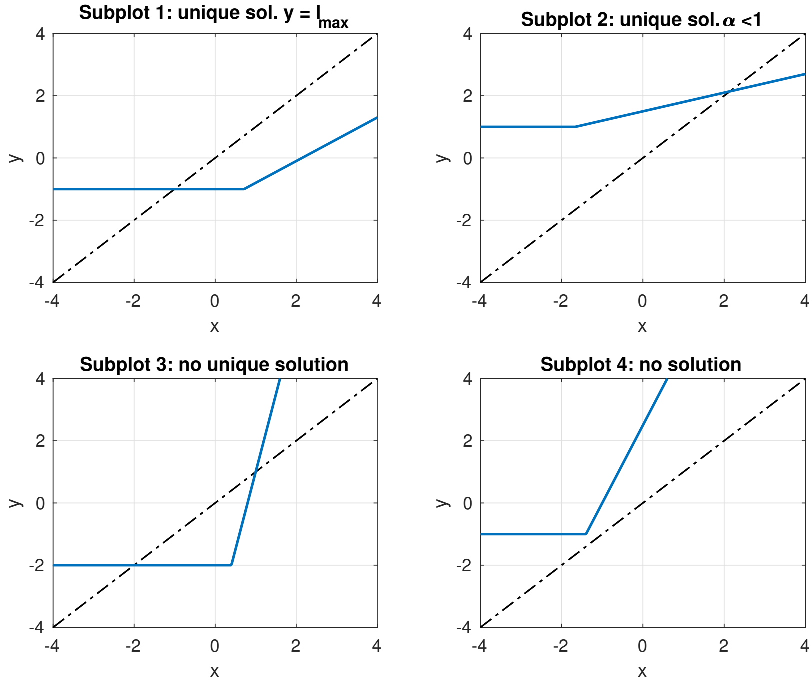


Figure 5.1: The four possibilities are plotted with some appropriate values of $\alpha, \beta \in \mathbb{R}$. It can be seen that the number of solutions is dependent on the value of the variables α, β as proposed in Eq. (5.22).

Example 5.2.1. In this example the the results from Eq. (5.23) & (5.24) will be used to solve the following two functions $y_1 = \max(3x + 2, 1)$ and $y_2 = \max(0.5x + 4, 1)$ for $y = x$. Starting with $y_1 = x$, when $l_{\max} = x$ is plugged in, it will give

$$\max(3 \cdot 1, 1) = 3 \neq 1,$$

hence $l_{\max}$ is not the solution. Then, $x = \dfrac{\beta}{1-\alpha}$ could be the solution.

$$x = \frac{\beta}{1-\alpha} = \frac{2}{-2} = -1 < l_{\max}$$

So, this solution also does not hold, which we could have known since $\alpha > 1$. This means that the equation is not solvable. This can also be seen in **Figure 5.2**. . The equation does not intersect x and it never will due to slope being greater than 1. Looking at $y_2 = \max(0.5 \cdot x + 4, 1)$ where α is indeed less than 1, we can try to solve this equation in the same way. When $l_{\max} = x$ is plugged in it will give:

$$\max(0.5 \cdot 1, 1) = 2 \neq 1.$$

Then the other solution will give:

$$x = \frac{\beta}{1-\alpha} = \frac{4}{0.5} = 8 > l_{\max}.$$

This means that $x = 8$ is the solution, which can also be seen in **Figure 5.2**.

◁

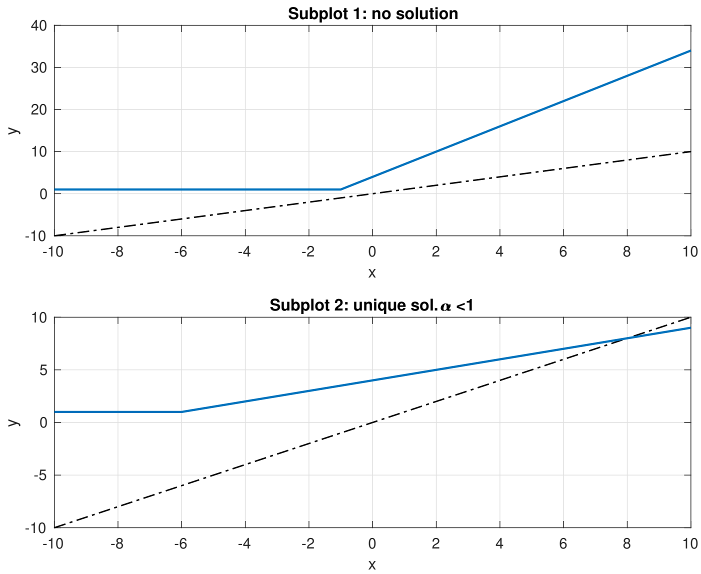


Figure 5.2: Plots of the two examples. We are trying to find possible intersection points with line $y = x$. We can see that line y_2, with $\alpha < 0$ has an intersection with $y = x$ at $x = 8$. For line y_1, where $\alpha > 0$, we see that there is no intersection with y.

5.3 Case 2: Varying parameter in both matrix A_0, and A_1

We can now repeat the process seen in the previous section. First, we will have to rewrite the model equations. Since the varying function is now in both system matrices, the constant term will not be present.

$$x_1(k) = \max\Big(a_{11}^0 + x_1(k), \quad a_{11}^1 + x_1(k-1)\Big) \tag{5.25}$$

$$= \max\Big(\alpha_1^0(x_1(k) - x_1(k-1)) + x_1(k), x_1(k-1) + \gamma\Big) \tag{5.26}$$

$$x_2(k) = \max\Big(a_{21}^0 + x_1(k), \quad a_{22}^0 + x_2(k), \quad a_{21}^1 + x_1(k-1), \quad a_{22}^1 + x_2(k-1)\Big) \tag{5.27}$$

$$= \max\Big(\alpha_1^0(x_2(k) - x_2(k-1)) + x_1(k), \quad \alpha_2^0(x_2(k) - x_2(k-1)) + x_2(k), \ldots \tag{5.28}$$

$$\ldots \alpha_1^1(x_2(k) - x_2(k-1)) + x_1(k-1), \quad \alpha_2^1(x_2(k) - x_2(k-1)) + x_2(k-1)\Big)$$

$$\vdots$$

$$x_i(k) = \max\Big(a_{i1}^0 + x_1(k), a_{i2}^0 + x_2(k), \ldots, a_{ii}^0 + x_i(k), \ldots \tag{5.29}$$

$$\ldots a_{i1}^1 + x_1(k-1), \quad a_{i2}^1 + x_2(k-1), \ldots, a_{ii}^1 + x_i(k-1)\Big).$$

Repeating the previous process, if we want to write Eq. (5.29) in a more general way, this takes the following steps:

- Rewrite Eq. (5.29):

$$x_i(k) = \max\Big(\alpha_1^0(x_i(k) - x_i(k-1)) + x_1(k), \quad \alpha_2^0(x_i(k) - x_i(k-1)) + x_2(k), \dots \\ \alpha_i^0(x_i(k) - x_i(k-1)) + x_i(k), \quad \alpha_1^1(x_i(k) - x_i(k-1)) + x_1(k-1), \dots \\ \alpha_2^1(x_i(k) - x_i(k-1)) + x_2(k-1), \dots \alpha_i^1(x_i(k) - x_i(k-1)) + x_i(k-1)\Big)$$

- Now to simplify this equation and make it more general we say that $x_i(k) = y$ and $\beta_j^\mu = x_j(k-\mu) - \alpha_j^\mu x_i(k-1)$ with $\mu = 0, 1$, so that we can write:

$$y = \max\Big(\alpha_1^0 y + \beta_1^0, \alpha_2^0 y + \beta_2^0, \dots, \alpha_i^0 y + \beta_i^0, \alpha_1^1 y + \beta_1^1, \alpha_2^1 y + \beta_2^1, \dots, \alpha_i^1 y + \beta_i^1\Big) \quad (5.30)$$

$$= \max\Big(f_1^0(y), f_2^0(y), \dots, f_i^0(y), f_1^1(y), f_2^1(y), \dots, f_i^1(y)\Big). \quad (5.31)$$

Now we have created a form for which we can easily find analytical solutions. In the next section we will look at the conditions for solving these equations. Later in Chapter 6 we will notice that our urban railway network is a special case of the form found in Eq. (5.30).

5.3.1 Intermezzo solving the equation: $y = \max(\alpha_1 y + \beta_1, \alpha_2 y + \beta_2)$

In accordance with Section 5.2, we will now simplify the problem such that we can find mathematical conditions that imply solvability. Suppose you have a simplified equation of the form found in (5.30):

$$y = \max(f_1(y), f_2(y)) \quad (5.32)$$

$$= \max(\alpha_1 y + \beta_1, \alpha_2 y + \beta_2) \quad (5.33)$$

with $y \in \mathbb{R}_\varepsilon^n$, $\alpha_1, \alpha_2 \in \mathbb{R}^+$, β_1, and $\beta_2 \in \mathbb{R}$. In **Figure 5.3** we can see the possibilities that are of interest. Looking at **Figure 5.3** we can conclude the following:

$$\begin{cases} \alpha_1 > 1, \alpha_2 < 1 & \to \text{intersection } f_1 = f_2 \text{ above } y = x, \quad \text{no sol.} \\ & \to \text{intersection } f_1 = f_2 \text{ on } y = x, \quad \text{one sol.: } (\frac{\beta_i}{1-\alpha_i})_{i=1,2}, \\ & \to \text{intersection } f_1 = f_2 \text{ below } y = x, \quad \text{two sol.: } (\frac{\beta_i}{1-\alpha_i})_{i=1,2} \end{cases} \quad (5.34)$$

$$\begin{cases} \alpha_1, \alpha_2 > 1 & \to \text{Always one solution: } \max_{i=1,2}(\frac{\beta_i}{1-\alpha_i}) \\ \alpha_1, \alpha_2 < 1 & \to \text{Always one solution: } \max_{i=1,2}(\frac{\beta_i}{1-\alpha_i}) \end{cases} \quad (5.35)$$

Let us look at an example to put these conditions to practice.

Example 5.3.1. Look at the following function:

$$y_1 = \max(0.2x + 0.5, \quad 0.7x + 1)$$

According to our previous finding in (5.35) the solution should be:

$$\max_{i=1,2}(\frac{\beta_i}{1-\alpha_i}) = \max(\frac{0.5}{0.8}, \quad \frac{1}{0.3}) = 3.33,$$

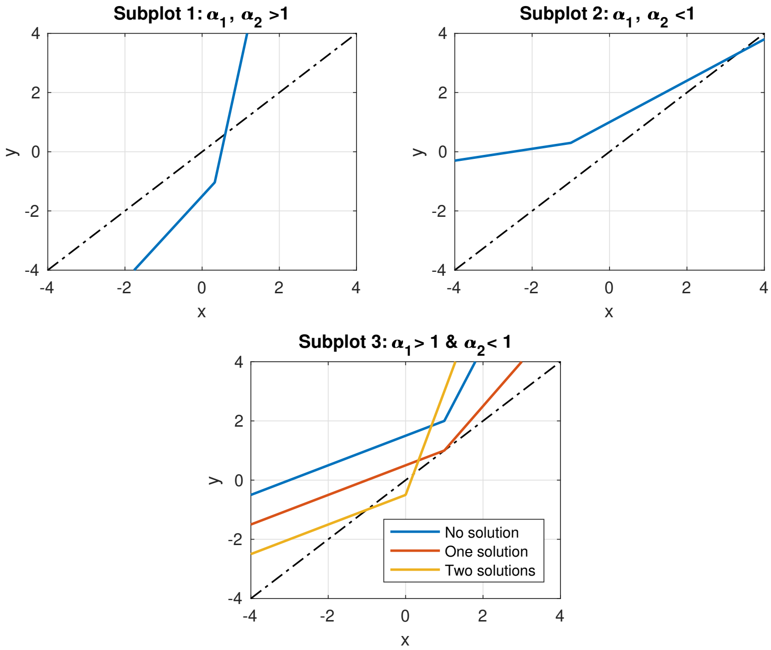


Figure 5.3: The three cases proposed in Eq. (5.34) & (5.35) are plotted with some appropriate values of $\alpha, \beta \in \mathbb{R}$. It can be seen that the number of solutions is dependent on the value of the variables α, β as proposed in Eq. (5.34) & (5.35).

which can also be seen in **Figure 5.4**. This illustrates the case where the slopes are both in the same direction. Let us now look at a case where $\alpha_1 < 1$, but $\alpha_2 > 1$:

$$y_2 = \max(0.5x + 0.5, \quad 1.5x - 0.5).$$

Use the solution proposed in (5.34):

$$(\frac{\beta_i}{1-\alpha_i})_{i=1,2} = \frac{0.5}{1-.5} = \frac{-0.5}{1-1.5} = 1.$$

This result can be checked by looking at **Figure 5.4**. ◁

We will use this analytical approach to analyze our urban railway model. In the next chapter we will see that it is not possible to claim structural solvability, but by using the conditions found in this section we can indeed predict when the railway model is solvable.

5.4 Summary

In this chapter, we have analyzed MP-LPV systems by rewriting them as MMPS functions. We have defined the necessary structure and classified the possibilities into two cases. For both cases, we were able to find clear conditions when the system is solvable. With this result, we will be able to find solvability conditions for the urban railway case study in Chapter 7.

In the next chapter, we will extend the urban railway network introduced in Section 2.5 to such an extent that we have an implicit MP-LPV with a varying parameter that is affine and state-dependent. We create this model such that we can apply all the results in the final chapter.

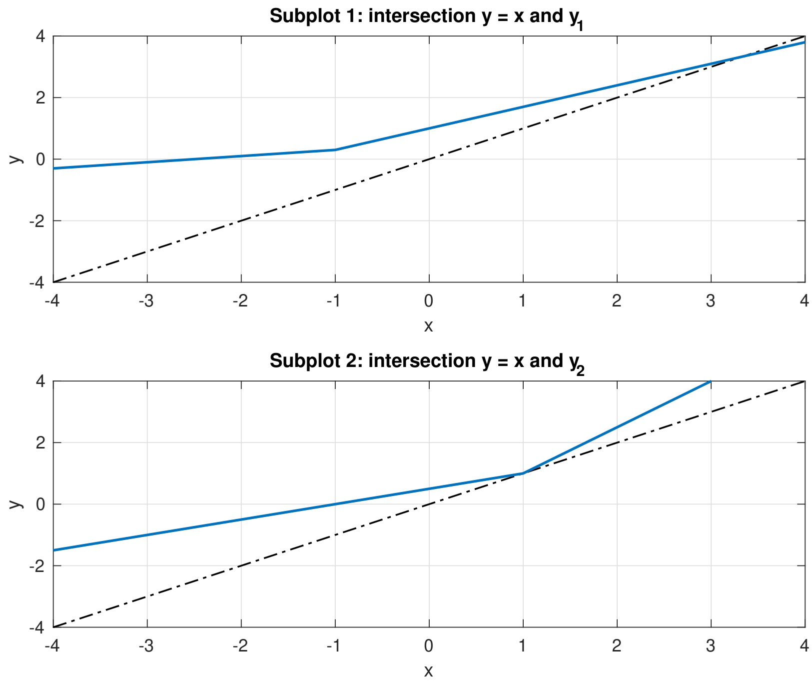


Figure 5.4: The intersections of y_1 and y_2 with $y = x$ as proposed in the example.

Chapter 6

A first MP-LPV system: An urban railway model with passenger dependent dwell time

In this chapter, we will extend the running example in such a way that it will become an implicit max-plus linear parameter varying (MP-LPV) systems with affine relations. A big difference with [8] is that the dwell time will be variable, and dependent on the system states themselves. Note that notation wise [1] is followed. First, we will introduce some extra assumptions, then we will explain how the variable dwell time is modelled. After this, the equations are rewritten and ordered in such a way that we can relate them to the results of the previous chapters.

6.1 Remodelling urban railway network

In Section 2.5, we have seen a first draft of the urban railway model. In this section, we will extend it such that it is possible to describe an urban railway network as an MP-LPV system. In **Figure 6.1** the model and three new variables can be seen. These variables transform the MPL system into an MP-LPV system. As can be seen, we now take into account the time people need to enter the stations (λ_j), to alight the train ($\tau_{out,j}(k)$), and to get on the train ($\tau_{in,j}(k)$). The following assumptions are made:

Assumption A14. *Passengers enter the station at a constant rate:* $\lambda_j(\frac{people}{time\ unit})$

Assumption A15. *The urban railway network is in a country where etiquette is important, hence passengers always alight the train first before the waiting passengers are allowed into the train.*

From these assumptions equations (6.1)-(6.3) follow:

$$a_j(k) \geq d_{j-1}(k) + \tau_{r,min,j-1} \tag{6.1}$$
$$a_j(k) \geq d_j(k-1) + \tau_{h,min} \tag{6.2}$$
$$d_j(k) \geq a_j(k) + \tau_{d,j}(k) \tag{6.3}$$

where $a_j(k)$ and $d_j(k)$ represent the arrival and departure times at station j, $\tau_{r,min,j-1}$ is the minimum running time between station $j-1$ and station j, $\tau_{h,min}$ is the minimum headway time, and $\tau_{d,j}(k)$ is the dwell time of train k at station j. As can be seen in Eq. (6.3), the dwell times become dependent on the train counter, which partly reveals the varying parameter. In previous chapters, it has already been mentioned that the variable dwell time will be dependent on the state variables. In Section 6.2, we will give more detail on the variable dwell time, and it will become clear where this dependence comes from.

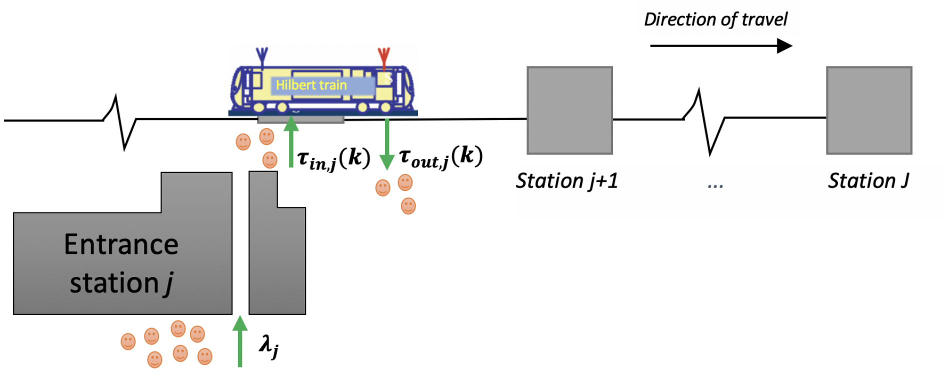


Figure 6.1: The model of two stations of the urban railway network including its variables. Note image is partly from [1].

6.2 Variable dwell time

Since the train leaves as soon as all conditions are met, we can define the dwell time:

$$\tau_{d,j}(k) = \tau_{out,j}(k) + \tau_{in,j}(k) \tag{6.4}$$

where $\tau_{out,j}(k)$ represents the time to alight the train, and $\tau_{in,j}(k)$ to get on the train. These two times are defined as follows:

$$\tau_{out,j}(k) = \alpha_{out} m_{out,j}(k) \tag{6.5}$$
$$\tau_{in,j}(k) = \alpha_{in} m_{in,j}(k) \tag{6.6}$$

where α_{out} is the average amount of time needed for a person to get out of the train, α_{in} is the average amount of time needed for a person to get on the train, $m_{out,j}(k)$ is the number of people getting out of train k at station j, and $m_{in,j}(k)$ is the number of people that get on train k at station j. The last two variables are further specified by defining them as follows:

$$m_{out,j}(k) = \sum_{i=1}^{j-1} \eta_{i,j} m_{in,i}(k) \tag{6.7}$$
$$m_{in,j}(k) = \lambda_j (d_j(k) - d_j(k-1)) \tag{6.8}$$

where $\eta_{i,j}$ is the proportion of people that gets in at station i and gets out at station j. Note that $\sum_{j=i+1}^{J} \eta_{i,j} = 1$, which actually means that everyone that gets the onto the train at station i will eventually have to get out at the final station J or earlier. Furthermore, it is assumed that the train will be empty at station 1, i.e. $m_{out,1} = 0$. All of this results in the following expression for the variable dwell time:

$$\tau_{d,j}(k) = \alpha_{out} \sum_{i=1}^{j-1} \eta_{i,j} m_{in,i}(k) + \alpha_{in}\lambda_j(d_j(k) - d_j(k-1)), \quad \text{for} \quad j \geq 2. \tag{6.9}$$

Then, we try to look more closely at these equations. This is done to see whether it is possible to rewrite the equations in a more fitting way. It is easy to see that the number of stations j will determine the length of the expression of $\tau_{d,j}(k)$. If j is large, then the expression of $\tau_{d,j}$ will also be large. Taking $j = \{1, \ldots, J\}$, with $J \in \mathbb{N}$, it follows that:

$$\begin{aligned}
\tau_{d,1}(k) &= 0 + \alpha_{in}\lambda_1(d_1(k) - d_1(k-1)) \\
&= \underbrace{\begin{bmatrix} \alpha_{in}\lambda_1 & 0 & 0 & \ldots & 0 \end{bmatrix}}_{=\Theta_1} \underbrace{\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_J \end{bmatrix}}_{=\mathbf{d}} (k) - \begin{bmatrix} \alpha_{in}\lambda_1 & 0 & 0 & \ldots & 0 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_J \end{bmatrix} (k-1) \\
\tau_{d,2}(k) &= \alpha_{in}\lambda_2(d_2(k) - d_2(k-1)) + \alpha_{out}\eta_{1,2}\lambda_1(d_1(k) - d_1(k-1)) \\
&= \underbrace{\begin{bmatrix} \alpha_{out}\eta_{1,2}\lambda_1 & \alpha_{in}\lambda_2 & 0 & \ldots & 0 \end{bmatrix}}_{=\Theta_2} \boldsymbol{d}(k) - \begin{bmatrix} \alpha_{out}\eta_{1,2}\lambda_1 & \alpha_{in}\lambda_2 & 0 & \ldots & 0 \end{bmatrix} \boldsymbol{d}(k-1) \\
\tau_{d,3}(k) &= \alpha_{in}\lambda_3(d_3(k) - d_3(k-1)) + \alpha_{out}\eta_{2,3}\lambda_2(d_2(k) - d_2(k-1)) + \alpha_{out}\eta_{1,3}\lambda_1(d_1(k) - d_1(k-1)) \\
&= \underbrace{\alpha_{out} \begin{bmatrix} \eta_{1,3}\lambda_1 & \eta_{2,3}\lambda_2 & \frac{\alpha_{in}}{\alpha_{out}}\lambda_3 & 0 & \ldots & 0 \end{bmatrix}}_{=\Theta_3} \boldsymbol{d}(k) \\
&\qquad - \alpha_{out} \begin{bmatrix} \eta_{1,3}\lambda_1 & \eta_{2,3}\lambda_2 & \frac{\alpha_{in}}{\alpha_{out}}\lambda_3 & 0 & \ldots & 0 \end{bmatrix} \boldsymbol{d}(k-1) \\
&\vdots \\
\tau_{d,J}(k) &= \underbrace{\alpha_{out} \begin{bmatrix} \eta_{1,J}\lambda_1 & \eta_{2,J}\lambda_2 & \ldots & \eta_{J-2,J}\lambda_{J-2} & \eta_{J-1,J}\lambda_{J-1} & \frac{\alpha_{in}}{\alpha_{out}}\lambda_J \end{bmatrix}}_{=\Theta_J} \boldsymbol{d}(k) \\
&\qquad - \underbrace{\alpha_{out} \begin{bmatrix} \eta_{1,J}\lambda_1 & \eta_{2,J}\lambda_2 & \ldots & \eta_{J-2,J}\lambda_{J-2} & \eta_{J-1,J}\lambda_{J-1} & \frac{\alpha_{in}}{\alpha_{out}}\lambda_J \end{bmatrix}}_{=\Theta_J} \boldsymbol{d}(k-1) \\
&= \Theta_J(\boldsymbol{d}(k) - \boldsymbol{d}(k-1)).
\end{aligned}$$

Now, we can substitute all the above equations by defining one vector:

$$\boldsymbol{\tau}_d(k) = \begin{bmatrix} \tau_{d,1}(k) \\ \tau_{d,2}(k) \\ \vdots \\ \tau_{d,J}(k) \end{bmatrix} = \begin{bmatrix} \Theta_1 \\ \Theta_2 \\ \vdots \\ \Theta_J \end{bmatrix} (\boldsymbol{d}(k) - \boldsymbol{d}(k-1)) \tag{6.10}$$

6.3 The max-plus linear parameter varying model

To create a max-plus version of our model equations (6.1)-(6.3) have to be translated into max-plus notation. This is where **Assumption A6** becomes important. The assumption here is made that the frequencies of departures is as high as possible, or in other words: trains will leave as soon as soon as the conditions are met. This has as a result that one of the inequalities of (6.1) and (6.2) will become an equality and that the inequality in (6.3) will have to be an equality. This results in the following:

$$a_j(k) = d_{j-1}(k) \otimes \tau_{r,min,j-1} \oplus d_j(k-1) \otimes \tau_{h,min} \tag{6.11}$$

$$d_j(k) = a_j(k) \otimes \tau_{d,j}(k) \tag{6.12}$$

It is possible to write this set of equations as follows:

$$x(k) = \left(\begin{array}{c} a_1(k) \\ a_2(k) \\ \vdots \\ a_J(k) \\ \hline d_1(k) \\ d_2(k) \\ \vdots \\ d_J(k) \end{array}\right) = \underbrace{\left(\begin{array}{ccccc|ccccc} & & & & & \varepsilon & \varepsilon & \cdots & \varepsilon & \varepsilon \\ & & & & & \tau_{r,min,1} & \varepsilon & \cdots & \varepsilon & \varepsilon \\ & & \mathcal{E} & & & \varepsilon & \tau_{r,min,1} & \ddots & \vdots & \vdots \\ & & & & & \vdots & \vdots & \ddots & \varepsilon & \varepsilon \\ & & & & & \varepsilon & \varepsilon & \cdots & \tau_{r,min,J-1} & \varepsilon \\ \hline \tau_{d,1}(k) & \varepsilon & \varepsilon & \cdots & \varepsilon & & & & & \\ \varepsilon & \tau_{d,2}(k) & \varepsilon & \cdots & \varepsilon & & & & & \\ \varepsilon & \varepsilon & \ddots & \ddots & \vdots & & & \mathcal{E} & & \\ \vdots & \vdots & \ddots & \ddots & \varepsilon & & & & & \\ \varepsilon & \varepsilon & \cdots & \varepsilon & \tau_{d,J}(k) & & & & & \end{array}\right)}_{=A_0(p(k))} \otimes x(k)$$

$$\oplus \underbrace{\left(\begin{array}{c|ccccc} & \tau_{h,min} & \varepsilon & \varepsilon & \cdots & \varepsilon \\ & \varepsilon & \tau_{h,min} & \varepsilon & \cdots & \varepsilon \\ \mathcal{E} & \varepsilon & \varepsilon & \ddots & \ddots & \vdots \\ & \vdots & \vdots & \ddots & \ddots & \varepsilon \\ & \varepsilon & \varepsilon & \cdots & \varepsilon & \tau_{h,min} \\ \hline \mathcal{E} & & & \mathcal{E} & & \end{array}\right)}_{=A_1} \otimes x(k-1) \tag{6.13}$$

Notice the Frobenius normal forms in the matrices. In the next section, we will use these matrices and write them in max-plus algebra, such that we can do the solvability analysis on them.

6.4 Rewriting the model equation

As has been discussed earlier we want to rewrite the model such that we can look at its solvability. Just by looking at the state-space, we expect the translation into max-plus algebra to resemble a function:

$$x_j(k) = \max\Bigg(x_j(k) + \tau_{d,j}\Big(x_1(k), \dots, x_j(k), x_1(k-1), \dots, x_j(k-1)\Big), x_{j-1}(k) + \\ \tau_{d,j-1}\Big(x_1(k), \dots, x_{j-1}(k), x_1(k-1), \dots, x_{j-1}(k-1)\Big), \dots, x_1(k) + \tau_{d,1}\Big(x_1(k), x_1(k-1)\Big)\Bigg) \tag{6.14}$$

If we have this form we can relate Eq. (6.14) to Eq. (5.30). If we can relate these two, we might be able to find conditions for solvability. Writing the equations down for $j \in \{1, 2, 3\}$ will give us:

$$\begin{aligned}
d_1(k) &= a_1(k) + \tau_{d,1}(k) = \tau_{d,1}(k) \\
d_2(k) &= a_2(k) + \tau_{d,2}(k) \\
&= \max\Big(d_1(k) + \tau_{r,1}, d_2(k-1) + \tau_{h,min}\Big) + \tau_{d,2} \\
&= \max\Big(\tau_{d,1}(k) + \tau_{r,1}, d_2(k-1) + \tau_{h,min}\Big) + \tau_{d,2} \\
d_3(k) &= a_3(k) + \tau_{d,3}(k) \\
&= \max\Big(d_2(k) + \tau_{r,2}, d_3(k-1) + \tau_{h,min}\Big) + \tau_{d,3} \\
&= \max\Bigg(\max\Big(\tau_{d,1}(k) + \tau_{r,1} + \tau_{d,2}(k), d_2(k-1) + \tau_{h,min} + \tau_{d,2}(k)\Big) + \dots \\
&\qquad \tau_{r,2} + \tau_{d,3}(k), d_3(k-1) + \tau_{h,min}\Bigg) + \tau_{d,3}(k) \\
&= \max\Big(\tau_{d,1}(k) + \tau_{d,2}(k) + \tau_{d,3}(k) + \tau_{r,1} + \tau_{r,2}, d_2(k-1) + \tau_h + \dots \\
&\qquad \tau_{d,2}(k) + \tau_{d,3}(k) + \tau_{r,2}, d_3(k-1) + \tau_h + \tau_{d,3}(k)\Big)
\end{aligned}$$

Using that $d(k) = x(k)$, and thus eliminating the arrival times, we get the following for $j = \{1, 2, \dots, J\}$ with $J \in \mathbb{N}$.

$$x_j(k) = \max\Big(\tau_{d,1}(k) + \dots + \tau_{d,j-1}(k) + \tau_{r,1} + \dots + \tau_{r,j-1}, x_2(k-1) + \tau_h + \\ \tau_{d,2}(k) + \dots + \tau_{d,j-1}(k) + \tau_{r,2}(k) + \dots + \tau_{r,j-1}(k), \dots, x_j(k-1) + \tau_h\Big) + \tau_{d,j}(k) \tag{6.15}$$

After rewriting it in this form it is still difficult to draw any conclusions on solvability. So, the idea was to plug in the expressions for $\tau_{d,j}(k)$ so that it is possible to relate it to equation (5.30). For that, we first had to find an appropriate way of writing $\tau_{d,j}(k)$.

$$
\begin{aligned}
\tau_{d,1}(k) &= \alpha_{in}\lambda_1 d_1(k) - \alpha_{in}\lambda_1 d_1(k-1) + 0 \\
&= \alpha_{in}\lambda_1 d_1(k) + \underbrace{\begin{bmatrix} 0 & -\alpha_{in}\lambda_1 & 0 & 0 & 0 & \dots & 0 \end{bmatrix}}_{=\Phi_1} \underbrace{\begin{bmatrix} d_1(k) \\ d_1(k-1) \\ d_2(k) \\ \vdots \\ d_J(k) \\ d_J(k-1) \end{bmatrix}}_{=\tilde{\mathbf{d}}} \\
\tau_{d,2}(k) &= \alpha_{in}\lambda_2 d_2(k) - \alpha_{in}\lambda_2 d_2(k-1) + \alpha_{out}\eta_{1,2}\lambda_1(d_1(k) - d_1(k-1)) \\
&= \alpha_{in}\lambda_2 d_2(k) + \underbrace{\begin{bmatrix} \alpha_{out}\eta_{1,2}\lambda_1 & -\alpha_{out}\eta_{1,2}\lambda_1 & 0 & -\alpha_{in}\lambda_2 & 0 & \dots & 0 \end{bmatrix}}_{=\Phi_2} \tilde{\mathbf{d}} \\
\tau_{d,3}(k) &= \alpha_{in}\lambda_3 d_3(k) - \alpha_{in}\lambda_3 d_3(k-1) + \alpha_{out}\eta_{2,3}\lambda_2(d_2(k) - d_2(k-1)) + \dots \\
&\qquad \alpha_{out}\eta_{1,3}\lambda_1(d_1(k) - d_1(k-1)) \\
&= \alpha_{in}\lambda_3 d_3(k) + \dots \\
&\quad \underbrace{\begin{bmatrix} \alpha_{out}\eta_{1,3}\lambda_1 & -\alpha_{out}\eta_{1,3}\lambda_1 & \alpha_{out}\eta_{2,3}\lambda_2 & -\alpha_{out}\eta_{2,3}\lambda_2 & 0 & -\alpha_{in}\lambda_3 & 0 & \dots \end{bmatrix}}_{=\Phi_3} \tilde{\mathbf{d}} \\
&\vdots \\
\tau_{d,J}(k) &= \alpha_{in}\lambda_J d_J(k) + \dots \\
&\quad \underbrace{\begin{bmatrix} \alpha_{out}\eta_{1,J}\lambda_1 & -\alpha_{out}\eta_{1,J}\lambda_1 & \alpha_{out}\eta_{2,J}\lambda_2 & -\alpha_{out}\eta_{2,J}\lambda_2 & \dots & 0 & -\alpha_{in}\lambda_J \end{bmatrix}}_{=\Phi_J} \tilde{\mathbf{d}} \\
&= \alpha_{in}\lambda_J d_J(k) + \Phi_J \tilde{\mathbf{d}}.
\end{aligned}
$$

Hence, we have a stacked variable dwell time that is written as:

$$
\boldsymbol{\tau}_d(k) = \begin{bmatrix} \tau_{d,1}(k) \\ \tau_{d,2}(k) \\ \vdots \\ \tau_{d,J}(k) \end{bmatrix} = \alpha_{in}\text{diag}(\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_J)\boldsymbol{d}(k) + \begin{bmatrix} \Phi_1 \\ \Phi_2 \\ \vdots \\ \Phi_J \end{bmatrix} \tilde{\mathbf{d}} \tag{6.16}
$$

This makes it possible to write the the departure times as follows:

$$
\begin{aligned}
d_1(k) &= \alpha_{in}\lambda_1 d_1(k) + \Phi_1\tilde{\mathbf{d}} \\
d_2(k) &= \max\Big(\alpha_{in}\lambda_2 d_2(k) + \Phi_2\tilde{\mathbf{d}} + d_1(k) + \tau_{r,1}, \quad \alpha_{in}\lambda_2 d_2(k) + \Phi_2\tilde{\mathbf{d}} + d_2(k-1) + \tau_{h,min}\Big) \\
d_3(k) &= \max\Big(\alpha_{in}\lambda_3 d_3(k) + \Phi_3\tilde{\mathbf{d}} + d_2(k) + \tau_{r,2}, \quad \alpha_{in}\lambda_3 d_3(k) + \Phi_3\tilde{\mathbf{d}} + d_3(k-1) + \tau_{h,min}\Big) \\
&\vdots \\
d_J(k) &= \max\Big(\alpha_{in}\lambda_J d_J(k) + \Phi_J\tilde{\mathbf{d}} + d_{J-1}(k) + \tau_{r,J-1}, \quad \alpha_{in}\lambda_J d_J(k) + \Phi_J\tilde{\mathbf{d}} + d_J(k-1) + \tau_{h,min}\Big)
\end{aligned}
$$

Using that $d(k) = x(k)$ we get the following for $j = \{1, 2 \ldots, J\}$ with $J \in \mathbb{N}$

$$x_j(k) = \max\left(\underbrace{\alpha_{in}\lambda_j}_{=\alpha_1} x_j(k) + \underbrace{\Phi_j\tilde{\mathbf{x}} + x_{j-1}(k) + \tau_{r,j-1}}_{=\beta_1},\quad \underbrace{\alpha_{in}\lambda_j}_{=\alpha_2} x_j(k) + \underbrace{\Phi_j\tilde{\mathbf{x}} + x_j(k-1) + \tau_{h,min}}_{=\beta_2}\right) \tag{6.17}$$

Now, we can relate it to Eq. (5.30). It seems to be a simple, but special case of what we have discussed. As it is clear that the slopes are equal, i.e. $\alpha_1 = \alpha_2$. This implies that there will always be a solution, except for $\alpha_1 = 1$, as we had concluded in Section 5.3.1.

6.5 Summary

In this chapter, an implicit MP-LPV model with an affine parametric dependence described in conventional algebra has been introduced. The dwell time is defined as a varying parameter which an affine function that is dependent on the scaled relation of the departure time of the current and previous train. In this chapter, we have written the model into one equation that can be related to the results found in the previous chapter.

In the next chapter, we will simulate this model and apply the results found in Chapter 4 and 5 to the model.

Chapter 7

Case study

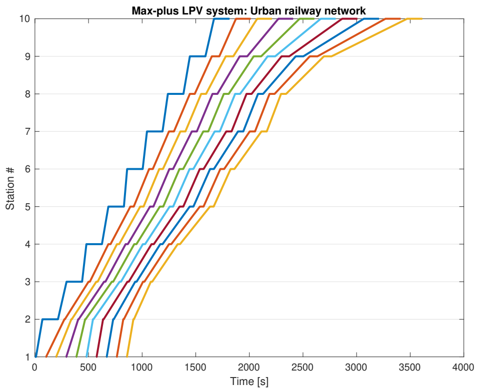
In the previous chapters, we have seen methods to analyze solvability and we have worked through an example which led to a simplified model of the urban railway network. It is important to analyze this model since it has all aspects that are of interest. The model is an implicit max-plus linear parameter varying (MP-LPV) model with a state-dependent varying parameter. In this chapter, we will show simulations of this model, show that the model is not structurally solvable, and lastly we will compare analytical results with the results obtained from the simulations.

7.1 Model constants

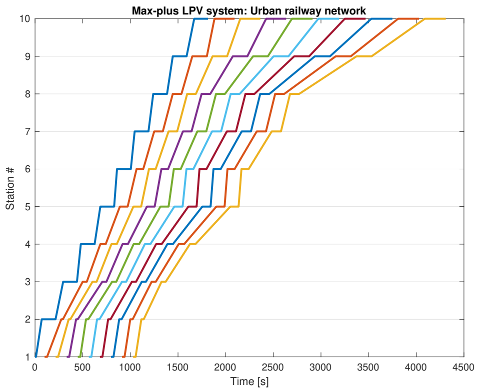
To analyze the solvability of implicit MP-LPV systems, a fictional urban railway line is studied. There are 10 stations in this railway line, i.e. $J = 10$. Important is to note that some values are just chosen randomly, to analyze and understand when and why solvability could break if it was solvable, to begin with. Mostly, we base our constants on [1], [11], see **Appendix A.1**.

The simulation is done without any disturbances and for the constants found in **Appendix A.1**. In **Figure 7.1** different values of α_{in} are displayed to demonstrate the influence of α_{in} on the departure times. The model behaves as we expected, the slower the passengers become the longer the dwell times of the trains. Note that the first train is just the initialization in the simulation, which is why it does not diverge as nicely as the other trains.

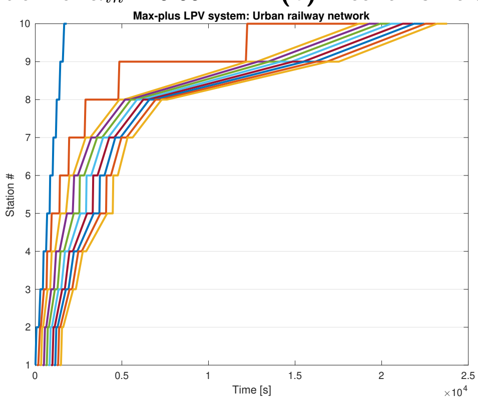
Remark 4. *For the attentive reader, there is a numerical reason for portraying only values* $\alpha_{in} < 1$. *This has to do with the divergence of fixed-point iterations with a slope that is greater than one. In Section 7.4, this problem is explained and an algorithm that solves this problem is proposed.*



(a) Free run simulation for $\alpha_{in} = 0.05$



(b) Free run simulation for $\alpha_{in} = 0.25$



(c) Free run simulation for $\alpha_{in} = 0.5$

Figure 7.1: Free run simulation of the urban railway network for different values of α_{in}

7.2 Structural solvability of the urban railway network

In the previous section, we have seen that the railway network shows expected behaviour for different values of α_{in}. In this section, we will analyze the structure of this model by using the theory proposed in Chapter 4. Our urban railway network is an extended version of **Implicit III** found in Section 4.2.2. Looking at Eq. (6.17) we can write the following functional dependencies for our case study:

$$x_1(k) = f(x_1(k-1),\ x_1(k)) \tag{7.1}$$
$$x_2(k) = f(x_1(k),\ x_2(k),\ x_1(k-1)),\ x_2(k-1)) \tag{7.2}$$
$$x_3(k) = f(x_1(k),\ x_2(k),\ x_3(k),\ x_1(k-1)),\ x_2(k-1),\ x_3(k-1),) \tag{7.3}$$
$$\vdots$$
$$x_{10}(k) = f(x_1(k),\ldots,x_9(k),\ x_{10}(k),\ x_1(k-1),\ldots,\ x_9(k-1)),\ x_{10}(k-1)). \tag{7.4}$$

These equations will give us the following representation graph:

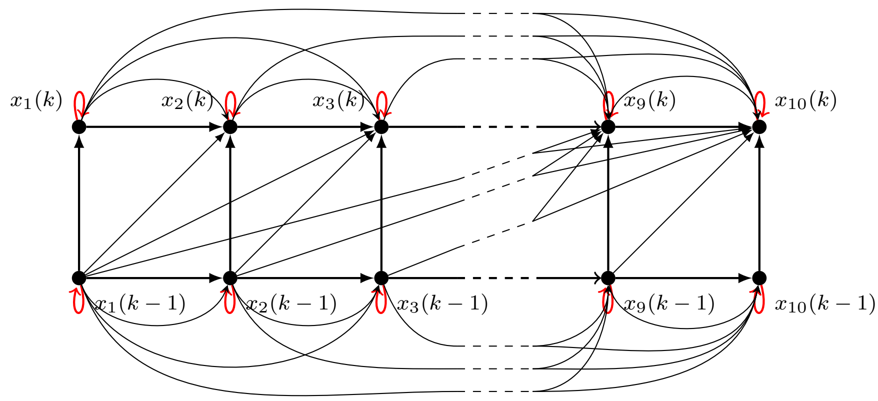


Figure 7.2: Representation graph of the urban railway network.

As you might already suspect, we cannot conclude structural solvability due to the red-drawn self-loops. Theoretically, if we leave out the self-loops, the functional dependencies lead to a system that is structurally solvable. Note that this is exactly the relaxation that has been done in [11]. Now that we cannot conclude structural solvability, we will try to find conditions for solvability analytically in the next section.

7.3 Analytical solvability of the urban railway network

In Eq. (6.17), we have arrived to a form that we can use in combination with the conditions in Section 5.3.1. For 10 stations write down the equations as follows:

$$
\begin{aligned}
x_1(k) &= \alpha_{in}\lambda_1 x_1(k) + \underbrace{\Phi_1\tilde{\mathbf{x}}}_{=\beta_1},\\
x_2(k) &= \max\Big(\alpha_{in}\lambda_2 x_2(k) + \underbrace{x_1(k) + \tau_{r,1} + \Phi_2\tilde{\mathbf{x}}}_{=\beta_2},\ \alpha_{in}\lambda_2 x_2(k) + \underbrace{x_2(k-1) + \tau_{h,min} + \Phi_2\tilde{\mathbf{x}}}_{=\gamma_2}\Big),\\
&\vdots\\
x_{10}(k) &= \max\Big(\alpha_{in}\lambda_{10} x_{10}(k) + \underbrace{x_{10}(k) + \tau_{r,9} + \Phi_{10}\tilde{\mathbf{x}}}_{=\beta_{10}},\ \alpha_{in}\lambda_{10} x_{10}(k) + \underbrace{x_{10}(k-1) + \tau_{h,min} + \Phi_{10}\tilde{\mathbf{x}}}_{=\gamma_{10}}\Big)
\end{aligned}
$$

Then the result found in Section 5.3.1 gives us the following solutions:

$$
\text{If } \beta_i > \gamma_i: \quad x_i(k) = \frac{\beta_i}{1-\alpha_{in}\lambda_i}, \quad \text{for } i = \{1, 2, \ldots, 10\}, \tag{7.5}
$$

$$
\text{If } \beta_i < \gamma_i: \quad x_i(k) = \frac{\gamma_i}{1-\alpha_{in}\lambda_i}, \quad \text{for } i = \{2, 3, \ldots, 10\}. \tag{7.6}
$$

Let us now compare this analytical result with the result we have found in the simulation. In the following table the comparison for three different values of α_{in} is done:

$\alpha_{in} = 0.05$

$\alpha_{in}\lambda_i$	$x_{i,\text{ana}}(10)$	$x_{i,\text{sim}}(10)$
0.045	858.17	858.17
0.04	923.72	923.72
0.065	1093.8	1093.8
0.06	1353.2	1353.2
0.07	1667.0	1667.0
0.04	1860.7	1860.7
0.04	2162.2	2162.2
0.045	2342.7	2342.7
0.06	2766.7	2766.7
0	3613.4	3613.4

$\alpha_{in} = 0.25$

$\alpha_{in}\lambda_i$	$x_{i,\text{ana}}(10)$	$x_{i,\text{sim}}(10)$
0.225	1055.2	1055.2
0.2	1140.6	1140.6
0.325	1370.9	1370.9
0.3	1682.5	1682.5
0.35	2135.6	2135.6
0.2	2241.3	2241.3
0.2	2580.3	2580.3
0.225	2771.6	2771.6
0.3	3531.2	3531.2
0	4309.7	4309.7

$\alpha_{in} = 0.5$

$\alpha_{in}\lambda_i$	$x_{i,\text{ana}}(10)$	$x_{i,\text{sim}}(10)$
0.45	1482.7	1482.7
0.4	1612.0	1612.0
0.65	2341.2	2341.2
0.6	2940.7	2940.7
0.7	4473.3	4473.3
0.4	4764.7	4764.7
0.4	5619.8	5619.8
0.45	7597.7	7597.7
0.6	17524.0	17524.0
0	23781.0	23781.0

Table 7.1: Comparison of analytically computed departure times $x_{i,ana}$ and departure times from simulation $x_{i,sim}$for different values of α_{in}

Clearly, the behaviour is as expected. The final departure time becomes higher as the people become slower (or higher value of α_{in}). Also, we see that our analytical results are identical to the results retrieved from the simulation.

7.4 Numerical issue

In the simulation, there is a numerical issue. For certain values of the slope of the function the algorithm diverges. In this section, we will discuss why it diverges and propose an algorithm to solve this.

In **Figure 7.1**, it can be seen that the larger the value of α_{in} becomes, the larger the dwell time becomes. It is reasonable to expect that at some point the dwell time becomes infinity, i.e. the train will never leave. Looking at Eq. (7.5) and (7.6) one would expect the breaking point to be at the slope being equal to one. This is where the first issue arises, the simulation seems to break before this point. Additionally, there is an issue for values of the slope greater than 1. Since we are trying to solve an equation that looks as follows:

$$x = \max\left(ax + b_1, ax + b_2\right) \tag{7.7}$$

with $x \in \mathbb{R}^n$ and $a, b_1, b_2 \in \mathbb{R}$. We can guarantee that there is a solution for all values of a, except for $a = 1$. Depending on the values we can classify the solutions as follows:

$$\begin{cases} \text{If} \quad |a| > 1 \quad \text{and} \quad \max\limits_{i=1,2}(b_i) < 0, \;\; \text{or} \\ \text{if} \quad |a| < 1 \quad \text{and} \quad \max\limits_{i=1,2}(b_i) > 0 \rightarrow \quad \text{positive solution} \end{cases} \tag{7.8}$$

$$\begin{cases} \text{If} \quad |a| > 1 \quad \text{and} \quad \max\limits_{i=1,2}(b_i) > 0, \;\; \text{or} \\ \text{if} \quad |a| < 1 \quad \text{and} \quad \max\limits_{i=1,2}(b_i) < 0 \rightarrow \quad \text{negative solution} \end{cases} \tag{7.9}$$

This means that for these values the problem should always be solvable, as we already found out in the thesis. What these solutions would mean is a different discussion for now.

The problems are solved in two steps:

- The first step in our solution is to not run the simulation for all stations in one loop, but split the loops such that every station is computed separately. See **Appendix A.2**.
- After the first step, it becomes clear that the break-point of the simulation is at $a = 1$ (in Eq. (7.7))). But then there is still the problem that it diverges and does not converge to a negative solution as suggested in (7.9). This is because not every sequence generated by fixed-point iteration will converge. In our case, the derivative of $f(x) = ax + b$ (see Eq. (7.7)) is equal to $f'(x) = a$. So, if $|a| \geq 1$ the fixed point will always diverge [29]. This can be solved by simply checking whenever $|a| \geq 1$, and then solve the mirrored equation. How does this work out? In our iteration, we are actually trying to solve an equation that looks like:

  $$x = ax + b \tag{7.10}$$

  And we have seen that the iteration diverges when $|a| \geq 1$. So for the case that $|a| \geq 1$, we rewrite Eq. (7.10) and solve the following form:

  $$z = \frac{1}{a}z - \frac{b}{a}, \tag{7.11}$$

  with $z \in \mathbb{R}^n$. In the following example this has been illustrated:

Example 7.4.1. Say that you have two equations:

$$x_1 = 3x_1 - 6, \tag{7.12}$$

$$x_2 = \frac{1}{3}x_2 + 2. \tag{7.13}$$

We see the same relation between Eq. (7.12) & (7.13) as between Eq. (7.10) & (7.11). Solving these two separately should then lead to the same solution. Which actually is the case, since the solutions of Eq. (7.12) & (7.13) are:

$$x_1 = 3,$$
$$x_2 = 3.$$

In **Figure 7.3** the plots of the two equations can be seen.

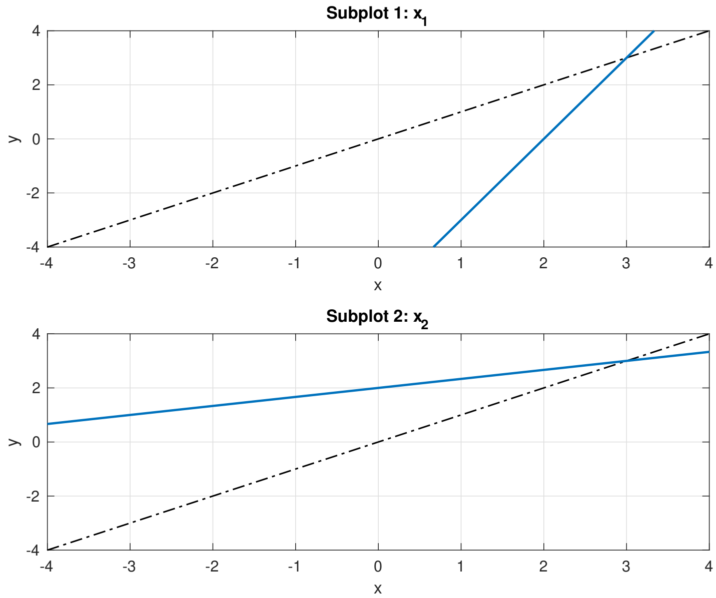


Figure 7.3: Figure with plots of both equations. Notice how the the blue lines rotate around the line $y = x$.

◁

The result portrayed in this simple example has been applied on the urban railway model. See the code in **Appendix A.3**.

7.5 Summary

In this chapter, the results of simulating the model proposed in Chapter 6 have been discussed. Solvability of the model is analyzed by using the results found in Chapter 4 and 5. We are not able to conclude structural solvability of the system, but we can give conditions for which the system is solvable and what kind of solution to expect. Finally, we discuss a numerical problem and propose a solution to this problem.

In the next chapter, we will conclude the thesis and propose directions for future work.

Chapter 8

Conclusion and future work

This work focused on two main points: creating a general framework for max-plus linear parameter varying (MP-LPV) systems, and analyzing solvability of MP-LPV systems. The conclusions of this thesis can be found in Section 8.1, after which recommendations for future research will be given in Section 8.2.

8.1 Conclusions

In this thesis, a newly introduced class of systems is discussed, namely MP-LPV systems. This subclass of discrete event systems (DESs) is linear in max-plus algebra and includes a varying parameter that is not necessarily linear. This extension of max-plus linear (MPL) systems can be defined as a combination of system representations. In our case, these would be the state-space representation and parameter trajectories. We classify the parametric dependence of MP-LPV systems as parameter affine in max-plus algebra or as parameter affine or polytopic in conventional algebra. When we describe the varying parameter in max-plus algebra (MPA), the nonlinearities are hidden in the varying parameter. In conventional algebra, not only nonlinearities are hidden, but also the usage of conventional algebra. In other words, it is possible to write an extended MPL system containing, although hidden, conventional elements in it.

It is important to take away that the interesting part for us is when the varying parameter is dependent on the current and/or previous states. When an implicit state-dependent varying parameter is present in a system it could disrupt solvability of the system. By using graph-theoretical methods, we have shown that the implicit relation makes it impossible to conclude structural solvability. In short, this means that we cannot assume that the system will be solvable for every possible trajectory. When we analyze the model by writing it as an max-min-plus-scaling (MMPS) function, we can find conditions for which the system is solvable. We see that when we limit ourselves to affine functions in the system that we can divide the possibilities into two cases. For both of them, it is possible to write them in such a way that you can point out when a system will be solvable or not. These results have been used and applied on an MP-LPV model, i.e. the urban railway network as proposed in Chapter 6. Due to the implicit varying parameter in the model it is not possible to conclude structurally solvability. Fortunately, the urban railway network is a special form of one of the cases discussed in Chapter 5. This makes it possible to rewrite it into a form closely related to the form found in Section 5.3.1. Because of that, we can point out when solvability of the system will break, and what could happen when certain constants of the system would change.

In this way, our insights of MP-LPV systems have broadened. We have defined a framework that leaves open possibilities to extend it along the lines of how LPV models are defined. Furthermore, we now know that there is a clear relation between MMPS functions and MP-LPV systems with only affine elements. With three propositions the relations have been shown and proven. For these systems, it is possible to analytically find conditions on their solvability, which we have seen and applied for the urban railway network.

8.2 Future work

Based on the ideas and results obtained during this research, a number of recommendations for future work are formed. These recommendations are listed in the following subsections.

8.2.1 Stability

In this thesis, stability of MP-LPV systems has not been discussed. Stability for MPL systems has been discussed in [4]. Here, they found stability for systems under a number of assumptions if the due date signal is in a bounded set. If these assumptions were used and one could proof that the varying parameter $p(k)$ is in a bounding box, i.e. $\{\underline{p}(k) \leq p(k) \leq \overline{p}(k)\}$. Then it is possible to look at stability like they did in [12]. Here they speak of stability if all possible solutions for a certain initial condition x_0 have the property of asymptotic stability. This is comparable to the notion of strong asymptotic stability of differential inclusions. Another way to stability is by introducing a parameter dependent Lyapunov function [12]. Although, we are unsure about the possibility to utilize this in the max-plus framework.

8.2.2 Controllability

We have already seen that to stabilize (switching) MPL systems it is necessary to have a weakly controllable system [4], therefore the notion of controllability is of importance for future work. In [26] structural controllability is discussed by using representation graphs, comparable to how we showed this in Chapter 4. This could be used as a starting point to analyze structural controllability, and subsequently one could work up to milder notions of controllability.

8.2.3 Relation of p-time event graphs with MP-LPV

It is possible to write a p-time event graph (PTEG) into an MP-LPV. We have already written the example proposed in [30, Example 2] into an MP-LPV, see **Appendix B.1**. This is done by ignoring the upper bounds, then simple mathematics suffice. But if we want to clarify, we have to look deeper into this and explain when and how it is possible to write one framework into the other. If we can show and prove these relations we should be able to make use of the tools within the PTEG framework. For instance, in [30] they discuss existence of trajectories for PTEGs with inter-affine residence durations. It is possible that we could make use of these notions to discuss existence of trajectories within our framework.

8.2.4 Possible application

It is possible to model a production system with an internal cooling process as an MP-LPV with an implicit varying parameter. For instance, a production line with a product that needs to cool down in a machine before you can put it in its package. The time the product needs to cool down is described by an affine relation between states, i.e. departure times. See **Appendix B.2** for the proposed model.

Appendix A

Simulation of the urban railway network

A.1 Model constants

In **Table A.2, A.3,** and **A.1** all constants can be found. These constants are used in the simulation and are inspired by [1] and [11].

Passenger rate	**[pass/s]**	**Min. running time**	**[s]**
λ_1	0.9	$\tau_{r,\min,1}$	60
λ_2	0.8	$\tau_{r,\min,2}$	80
λ_3	1.3	$\tau_{r,\min,3}$	40
λ_4	1.2	$\tau_{r,\min,4}$	60
λ_5	1.4	$\tau_{r,\min,5}$	30
λ_6	0.8	$\tau_{r,\min,6}$	40
λ_7	0.8	$\tau_{r,\min,7}$	50
λ_8	0.9	$\tau_{r,\min,8}$	60
λ_9	1.2	$\tau_{r,\min,9}$	80
λ_{10}	0	$\tau_{r,\min,10}$	0

Table A.1: Values in simulation of passenger rates per station and minimal running times per station

Description	Symbol	Value
Minimal headway time [s]	$\tau_{h,\min}$	60
Minimal dwell time [s]	$\tau_{d,\min}$	5
Average amount of time to get out of the train [s]	α_{out}	0.42
Average amount of time to get in the train [s], simulation 1	α_{in}	0.05
Average amount of time to get in the train [s], simulation 2	α_{in}	0.25
Average amount of time to get in the train [s], simulation 3	α_{in}	0.5

Table A.2: Other constants

station	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0	0.05	0.2	0.05	0.4	0.1	0.03	0.07	0.05	0.05
2	0	0	0.03	0.09	0.25	0.08	0.2	0.1	0.04	0.21
3	0	0	0	0.3	0.07	0.15	0.13	0.12	0.15	0.08
4	0	0	0	0	0.07	0.16	0.23	0.14	0.23	0.17
5	0	0	0	0	0	0.24	0.23	0.07	0.12	0.34
6	0	0	0	0	0	0	0.32	0.24	0.27	0.17
7	0	0	0	0	0	0	0	0.4	0.26	0.34
8	0	0	0	0	0	0	0	0	0.5	0.5
9	0	0	0	0	0	0	0	0	0	1
10	0	0	0	0	0	0	0	0	0	0

Table A.3: Ratios of alighting passengers

A.2 Numerical problem: part 1

The first step in our solution is to not run the simulation for all stations in one loop, but compute it separately for each station. See the code below:

```

alph_i   = 1.06;              % avg amount of time to get in the train
alph_o   = 2.42;              % avg amount of time to get out of the train
d1       = zeros(1, M+1);     % init departure time vector for station 1
d1(1)    = 10;
d1(2)    = 10 + min_headway;

d1_i     = zeros(1, M+1,I);   % init departure time array for the fixed point iteration
taud     = zeros(1, M+1);     % init the vector where all dwell times are stored
taud_i   = zeros(1,M+1,I);    % init array for dwell times in fixed point iteration

for k = 2:M+1
    d1_i(1,k,1) = d1(1,k);
    for i = 2:I
       taud_i(1,k,i) = alph_i*lambda(1)*d1_i(1,k,i-1) - alph_i*lambda(1)*d1(1,k-1);
       d1_i(1,k,i) = tauh1 + taud_i(1,k,i) +d1(1,k-1);
        if abs(d1_i(1,k,i)-d1_i(1,k,i-1))<10^(-6)
            d1(1,k)=d1_i(1,k,i);
            taud(1,k)=taud_i(1,k,i);
            i_index(k) = i;
            break
        end
    end
end
```

```
% here the above process is repeated for the second station.
d2      = zeros(1, M+1);
d2(1)   = d1(1) + 60 + 5;
d2(2)   = d2(1) + min_headway;

d2_i    = zeros(1, M+1,I);
taud    = zeros(1, M+1);
taud_i  = zeros(1,M+1,I);

for k = 2:M+1
    d2_i(1,k,1) =  d2(1,k);
    for i = 2:I
       taud_i(1,k,i) = alph_i*lambda(2)*d2_i(1,k,i-1) - alph_i*lambda(1)*d2(1,k-1) +
           alph_o*lambda(1)*d1(1,k) -alph_o*lambda(1)*d1(1,k-1);
       d2_i(1,k,i) = oplus(tau_r_min(1) +taud_i(1,k,i) + d1(1,k), tauh2 + taud_i(1,k,i
           ) +d2(1,k-1));
        if abs(d2_i(1,k,i)-d2_i(1,k,i-1))<10^(2)
            d2(1,k)=d2_i(1,k,i);
            taud(1,k)=taud_i(1,k,i);
            i_index2(k) = i;
            break
        end
    end
end
```

A.3 Numerical problem: part 2

The second part of the solution is presented for station 2 in the code below:

```
d2      = zeros(1, M+1);
d2(1)   = d1(1) + 60 + 5;
d2(2)   = d2(1) + min_headway;

d2_i    = zeros(1, M+1,I);
taud    = zeros(1, M+1);
taud_i  = zeros(1,M+1,I);

for k = 2:M+1
    d2_i(1,k,1) =   d2(1,k);
    for i = 2:I
       taud_i(1,k,i) = 1/(alph_i*lambda(2))*d2_i(1,k,i-1) - 1/(alph_i*lambda(1)*d2(1,k
           -1)) + 1/(alph_o*lambda(1)*d1(1,k)) -1/(alph_o*lambda(1)*d1(1,k-1));
       d2_i(1,k,i) = oplus(1/(tau_r_min(1)) + taud_i(1,k,i) + 1/(d1(1,k)), 1/(tauh2) +
            taud_i(1,k,i) +1/(d2(1,k-1)));
        if abs(d2_i(1,k,i)-d2_i(1,k,i-1))<10^(-6)
            d2(1,k)=-d2_i(1,k,i);
            taud(1,k)=taud_i(1,k,i);
            i_index2(k) = i;
            break
        end
    end
end
```

The mirrored equation is solved as a fixed point iteration and the solution is multiplied with minus 1 for every loop. This code works, and is also able to find negative solutions. It could be used to work around the variety of the coefficient by using if-else statements.

Appendix B

Future work

B.1 Relation MP-LPV and PTEG

This section provides an intent on how to write the general model of a p-time event graph (PTEG) into an MP-LPV model. The description of the model comes from the literature on PTEGs [30, Example 2].

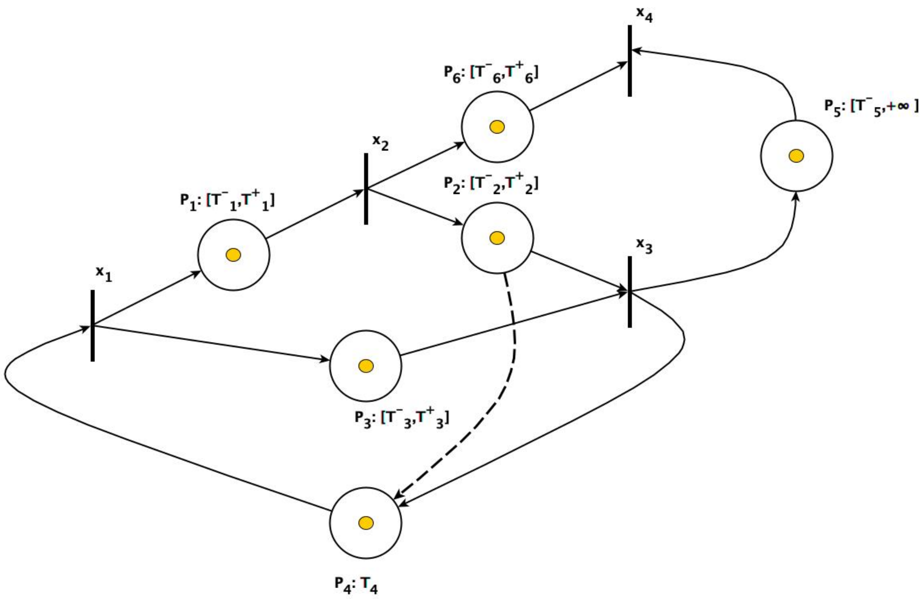


Figure B.1: PTEG with interdependent residence durations (between p_2 and p_4)

The PTEG in **Figure B.1** is explained as follows:

- All places have just one token in them, which makes the inequalities relatively simple.

- The state is $x(k) = \begin{pmatrix} x_1(k) & x_2(k) & x_3(k) & x_4(k) \end{pmatrix}^T$
- The temporal intervals are $[T_1^-, T_1^+] = [3, 10]$, $[T_2^-, T_2^+] = [3, 20]$, $[T_3^-, T_3^+] = [1, 2]$, $[T_5^-, +\infty] = [11.5, +\infty]$, $[T_6^-, T_6^+] = [1, 5]$
- Place p_4 is the one where the affine dependence on previous place p_2 is. The duration time at p_4 is $T_4 = \alpha(x_3(k) - x_2(k-1)) + \beta$, where $\alpha = 5$ and $\beta = 3$.

Having all this information we can directly write the algebraic model as:

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & -\alpha & 1 & 0 & -1 & 0 & \alpha & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}}_{=G} \times \begin{pmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{pmatrix} \leq \begin{pmatrix} -3 \\ -3 \\ -1 \\ -\beta \\ -11.5 \\ -1 \\ 10 \\ 20 \\ 2 \\ +\infty \\ +\infty \\ 5 \end{pmatrix} \tag{B.1}$$

This matrix G is equally partitioned as follows:

$$G = \begin{pmatrix} G^- \\ G^+ \end{pmatrix} = \begin{pmatrix} G_1^- & G_0^- \\ G_1^+ & G_0^+ \end{pmatrix} \tag{B.2}$$

To write it as an MP-LPV the upper bounds have to be ignored, i.e. we only use $G^- \times x(k) \leq T^-$. Those equations can be rewritten into:

$$x_1(k) \geq \alpha(x_3(k) - x_2(k-1) + \beta + x_3(k-1) \tag{B.3}$$
$$x_2(k) \geq x_1(k-1) + T_1^- \tag{B.4}$$
$$x_3(k) \geq x_2(k-1) + T_2^- \tag{B.5}$$
$$x_3(k) \geq x_1(k-1) + T_3^- \tag{B.6}$$
$$x_4(k) \geq x_3(k-1) + T_5^- \tag{B.7}$$
$$x_4(k) \geq x_2(k-1) + T_6^- \tag{B.8}$$

Regard this system as a product that goes through a number of different processes. In p_4 the cooling down of the product happens. Assume for now that the product leaves a place as soon as it is possible. This will change Eq. (B.3) to (B.8) into:

$$x_1(k) = \alpha(x_3(k) - x_2(k-1) \otimes \beta \otimes x_3(k-1) = p(k) \otimes x_3(k-1) \tag{B.9}$$
$$x_2(k) = T_1^- \otimes x_1(k-1) \tag{B.10}$$
$$x_3(k) = T_2^- \otimes x_2(k-1) \oplus T_3^- \otimes x_1(k-1) \tag{B.11}$$
$$x_4(k) = T_5^- \otimes x_3(k-1) \oplus T_6^- \otimes x_2(k-1) \tag{B.12}$$
$$\tag{B.13}$$

This again can be written in matrix form as follows:

$$x(k) = \begin{pmatrix} \varepsilon & \varepsilon & p(k) & \varepsilon \\ T_1^- & \varepsilon & \varepsilon & \varepsilon \\ T_3^- & T_2^- & \varepsilon & \varepsilon \\ \varepsilon & T_6^- & T_5^- & \varepsilon \end{pmatrix} \otimes \begin{pmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \\ x_4(k-1) \end{pmatrix} \tag{B.14}$$

with $p(k) = \alpha(x_3(k) - x_2(k-1) + \beta$.

B.2 New system: Production system with cooling down component

In this section, we will try to propose a new model. Note that this model is an early prototype and needs more research especially on the model constants. Furthermore, the model is kept as simple as possible, if this model is to be used it should be extended.

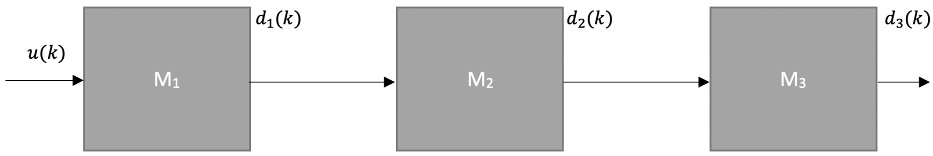


Figure B.2: Production system with three machines

Consider the production system in **Figure B.2**. In this system there are three machines M_1, M_2, M_3. The product is fed to M_1 where processing is done, after which it goes to M_2 where it needs to cool down, before it goes to M_3 to put the product in a package. There are two main assumptions, firstly we assume that the machines start working as soon as the product is able (under certain circumstances) to enter the machine. Secondly, we assume that only the second processing time is varying. Define $u(k)$ as time instant at which M_1 is fed with the product. Furthermore, in this model we speak of arrival times of the product and times for which product k leaves machine i defined as $a_i(k)$ and $d_i(k), i = 1, 2, 3$ respectively. One of the differences with other production systems is that we do not neglect transportation times between machines. Also, assume that the transportation times between machine can be neglected. Lastly, we define the processing times in the machines as $\rho_i(k), i = 1, 2, 3$. We can now write the following equations:

$$d_1(k) = \max(a_1(k) + \rho_1, d_2(k-1)) \tag{B.15}$$

$$d_2(k) = \max(a_2(k) + \rho_2(k), d_3(k-1)) \tag{B.16}$$

$$d_3(k) = a_3(k) + \rho_3 \tag{B.17}$$

Note how the second processing time is the only one that is dependent on the product counter k. Let us define this processing time as:

$$\rho_2(k) = \alpha(d_2(k) - a_2(k)) + \beta \tag{B.18}$$

We could also extend this to have a switching MP-LPV. Simply by adding a machine and permitting two modes into the system, e.g. because there are two products that require different cooling processes. In **Figure B.3** an idea of how this would look is given.

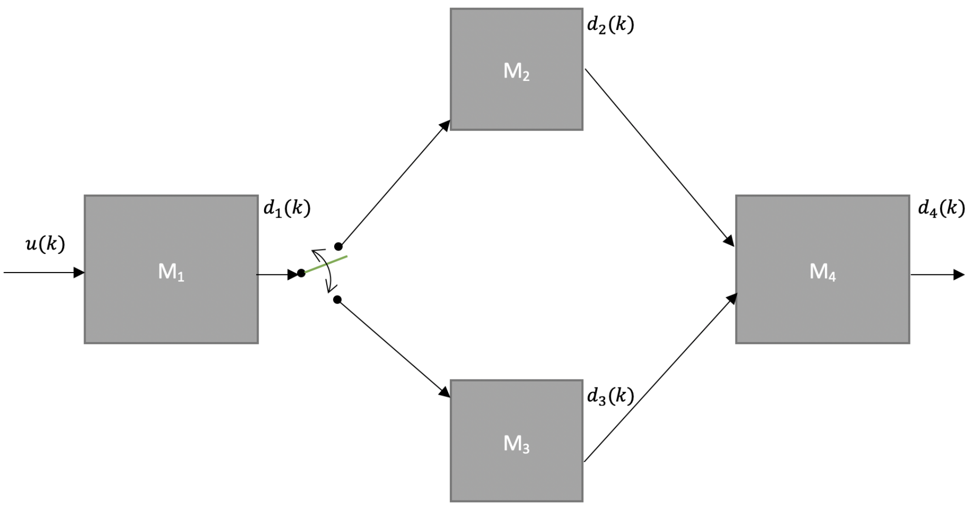


Figure B.3: Production system with three machines, but now with two possible modes.

Bibliography

[1] Y. Wang, B. Ning, T. J. J. van den Boom, and B. De Schutter, *Optimal Trajectory Planning and Train Scheduling for Urban Rail Transit Systems.* Springer, 2016.

[2] B. De Schutter and T. J. J. van den Boom, "Max-plus algebra and max-plus linear discrete event systems: An introduction," in *9th International Workshop on Discrete Event Systems*, 2008.

[3] F. Baccelli, G. Cohen, G. Olsder, and J. Quadrat, *Synchronization and Linearity: An Algebra for Discrete Event Systems.* John Wiley & Sons, 1992.

[4] T. J. J. van den Boom and B. De Schutter, "A stabilizing model predictive controller for uncertain max-plus-linear systems and uncertain switching max-plus-linear systems," in *Proceedings of the 18th IFAC World Congress*, 2011.

[5] T. J. J. van den Boom, M. van den Muijsenberg, and B. De Schutter, "Model predictive scheduling of semi-cyclic discrete-event systems using switching max-plus linear models and dynamic graphs," *Discrete Event Dynamic Systems: Theory and Applications*, 2019.

[6] B. F. Heidergott, G. Olsder, and J. van der Woude, *Max Plus at Work-Modeling and Analysis of Synchronized Systems.* Series in Applied mathematics, Princeton University Press, 2006.

[7] B. Kersbergen, T. J. J. van den Boom, and B. De Schutter, "On implicit versus explicit max-plus modeling for the rescheduling of trains," in *Proceedings of the 5th International Seminar on Railway Operations Modelling and Analysis (RailCopenhagen)*, (Copenhagen, Denmark), 2013.

[8] T. J. J. van den Boom and B. De Schutter, "Modelling and control of discrete event systems using switching max-plus-linear systems," *Control Engineering Practice*, 2006.

[9] H. Goto, K. Takeyasu, S. Masuda, and T. Amemiya, "A gain scheduled model predictive control for linear-parameter-varying max-plus-linear systems," 2003.

[10] S. Masuda, H. Goto, T. Amemiya, and K. Takeyasu, "An inverse system for linear parameter-varying max-plus-linear systems," 2003.

[11] B. Zwerink Arbones, "An exploratory study in max-plus linear parameter varying systems with application to an urban railway line," Master's thesis, Delft University of Technology, 2019.

[12] F. Bruzelius, "Linear parameter-varying systems an approach to gain scheduling," *Doktorsavhandlingar vid Chalmers Tekniska Hogskola*, 2004.

[13] J. S. Shamma, J. Mohammadpour, and C. W. Scherer, *Control of linear parameter varying systems with applications.* Springer-Verlag New York, 1 ed., 2012.

[14] J. S. Shamma, I. of Technology. Laboratory for Information, and M. Decision Systems, "Analysis and design of gain scheduled control systems," 1988.

[15] D. P. Bertsekas, "Convex optimization theory," 2009.

[16] B. De Schutter and T. J. J. van den Boom, "Model predictive control for max-min-plus-scaling systems," in *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, 2001.

[17] B. De Schutter and T. J. J. van den Boom, "MPC for continuous piecewise-affine systems," *Systems & Control Letters*, 2004.

[18] I. Necoara, B. De Schutter, T. J. J. van den Boom, and J. Hellendoorn, "Model predictive control for perturbed continuous piecewise affine systems with bounded disturbances," in *Proceedings of the 43rd IEEE Conference on Decision and Control*, 2004.

[19] S. Masuda, "A model predictive control for max-plus linear systems with interval parameters," in *SICE-ICASE International Joint Conference*, 2006.

[20] S. Masuda, H. Goto, T. Amemiya, and K. Takeyasu, "An internal model control for max-plus linear systems with linear parameter varying structure," in *IFAC 7th International Symposium on Advanced Control of Chemical Processes*, 2004.

[21] J. S. Shamma and M. Athans, "Gain scheduling: potential hazards and possible remedies," *IEEE Control Systems Magazine*, 1992.

[22] V. Verdult, *Non linear system identification: a state-space approach.* PhD thesis, Twente University, Netherlands, 2002.

[23] J. Bokor and G. Balas, "Linear parameter varying systems: A geometric theory and applications," *4th International Symposium on Applied Computational Intelligence and Informatics*, 2005.

[24] A. Delgado, D. Gagliardi, M. Gargano, M. Lewinter, and W. Widulski, "Subcubes of hypercubes," *Congressus Numerantium*, 2008.

[25] P. Apkarian, P. Gahinet, and G. Becker, "Self-scheduled h control of linear parameter-varying systems: a design example," *Autom.*, 1995.

[26] K. Murota, "Menger-decomposition of a graph and its application to the structural analysis of a large-scale system of equations," *Discrete Applied Mathematics*, vol. 17, 1987.

[27] K. Murota, *Systems Analysis by Graphs and Matroids-Structural Solvability and Controllability*. World Publishing Company, 1987.

[28] B. De Schutter, "On the ultimate behavior of the sequence of consecutive powers of a matrix in the max-plus algebra," *Linear Algebra and Its Applications*, 2000.

[29] P. B. Laval, "Fixed point iterations." http://ksuweb.kennesaw.edu/~plaval//previous%20semesters/spring2010/m4400/fixedpt.pdf, 2000.

[30] P. Declerck, "Cycle time of a p-time event graph with affine-interdependent residence durations," *Discrete Event Dynamic Systems*, vol. 24, 2014.

Glossary

List of Acronyms

MPA	max-plus algebra
LPV	linear parameter varying
m.s.c.s.	maximal strongly connected subgraph
DES	discrete event system
MPL	max-plus linear
SMPL	switching max-plus linear
MP-LPV	max-plus linear parameter varying
PTEG	p-time event graph
LTV	linear time varying
MMPS	max-min-plus-scaling
MPS	max-plus-scaling
PWA	piecewise-affine