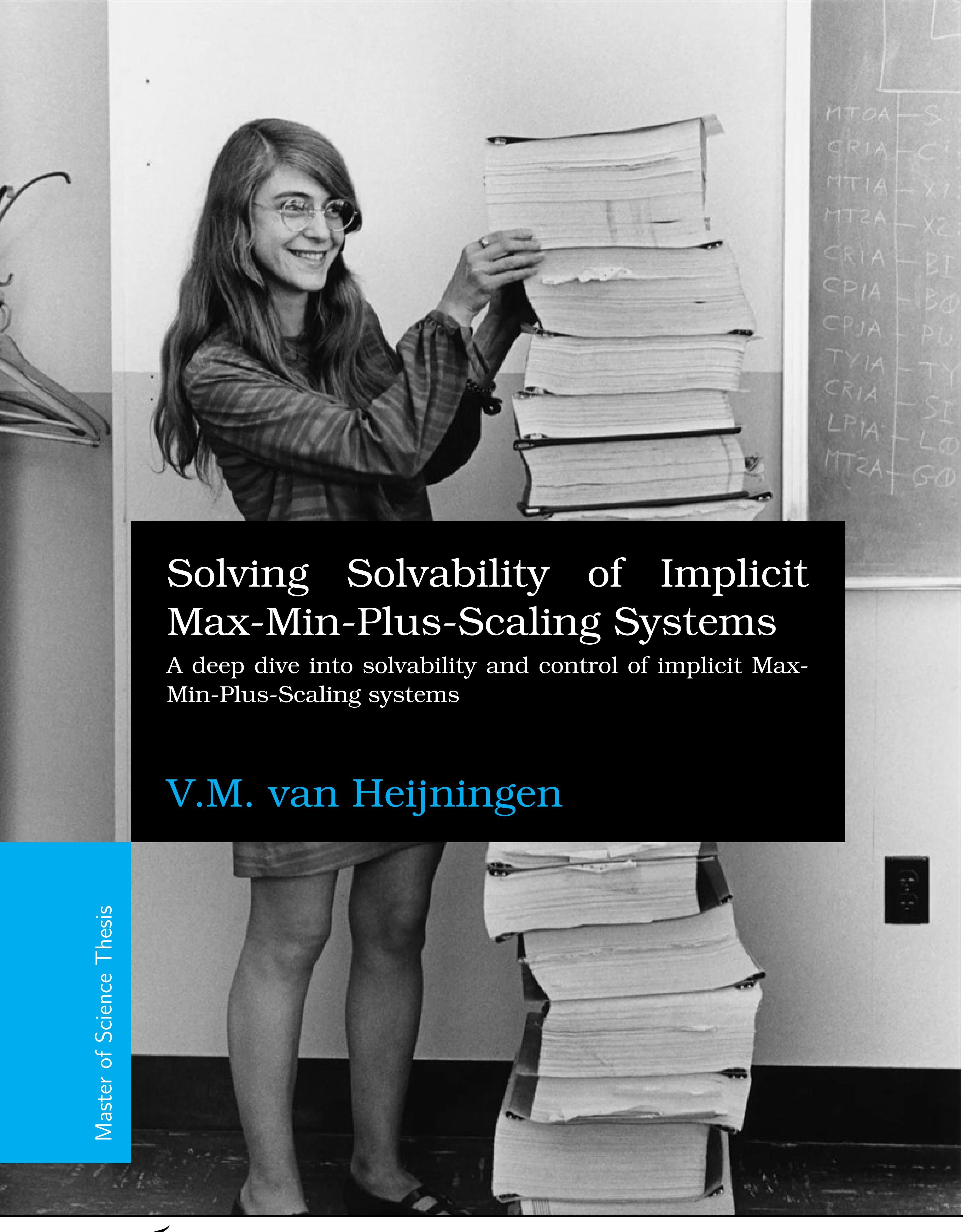


Solving Solvability of Implicit Max-Min-Plus-Scaling Systems

A deep dive into solvability and control of implicit Max-Min-Plus-Scaling systems

V.M. van Heijningen

Master of Science Thesis

TU Delft Delft University of Technology

Delft Center for Systems and Control

Solving Solvability of Implicit Max-Min-Plus-Scaling Systems

A deep dive into solvability and control of implicit Max-Min-Plus-Scaling systems

Master of Science Thesis

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

V.M. van Heijningen

June 20, 2025

Faculty of Mechanical Engineering (ME) · 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 Engineering (ME) for acceptance a thesis entitled

Solving Solvability of Implicit Max-Min-Plus-Scaling Systems

by

V.M. van Heijningen

in partial fulfillment of the requirements for the degree of

Master of Science Systems and Control

Dated: June 20, 2025

Supervisor(s): ______________________

Dr. Ir. A.J.J. van den Boom

____

Ir. S. Markkassery

Reader(s): ______________________

Dr. Ir. R.D. McAllister

Abstract

This thesis dives deep into the concepts of solvability and control of implicit Max-Min-Plus-Scaling (MMPS) systems. An advanced mathematical framework used to model discrete-event systems combining max-plus, min-plus, and conventional algebraic operations. These systems have a broad spectrum of applications in fields such as scheduling, transportation, and performance evaluation of networks. An initial overview of MMPS systems, and necessary background is provided through the mathematical preliminaries, including max-plus and min-plus algebra, spectral theory, and their graph-theoretical interpretations. This thesis recognizes the distinction between explicit and implicit MMPS systems, where the latter involves current state dependencies, leading to challenges in analysis and solvability. The focus of the thesis will solely lie in researching implicit MMPS systems, and is split into two main parts. The first part providing novel theoretical concepts regarding control and solvability of implicit MMPS systems.The main contribution of the first part lies in extending the existing solvability theory. This thesis shows that previously proposed solvability conditions are merely sufficient, but not necessary. A graph-theoretic interpretation of solvability is introduced by analyzing the structure matrix S, and conditions are developed to identify circuit subsystems, which pinpoint implicit dependencies within the system. The thesis further proposes a classification of solvability into uniquely solvable-, parametrically solvable-, parametrically unsolvable-, and strictly unsolvable modes and derives a necessary and sufficient condition for solvability using rank tests on linear algebraic subsystems. Furthermore, the control of implicit MMPS systems is explored by proposing open-loop and closed-loop control strategies. The effects of these control strategies on system properties such as time-invariance and solvability are analytically derived. In the second part, the theoretical results are supported by application to an urban railway system (URS), which is augmented in order to accommodate complex passenger flows, and controlled using the developed implicit MMPS control framework. Results of the simulation demonstrate the system's stability and effectiveness of the control strategies under various disturbances. Overall, this thesis provides significant theoretical advancements in implicit MMPS system analysis, and offers practical methodologies and illustrative examples regarding modeling and controlling complex discrete-event systems.

Table of Contents

Acknowledgements

These sentences are the last I am writing in this 176 (!!) pages long document I am proud to call my Master Thesis. I specifically chose to let the acknowledgments be the part that would bring this period of my life to a close, because I have been looking forward to writing them since September 2nd, 2024, the day I embarked on this exactly 9 month long project. A thesis project is something you do by yourself, as plagiarism is outright wrong. However, you do not do it alone. I am someone who tries to let their loved ones know how much they mean to me, so I hope these sentimental words do not come as a surprise. But allow me to use this opportunity to let this be like a love letter to my loved ones, and an ode to all the people whose presence has made it possible for me to complete my thesis, and look back on this special time so fondly. I will shamelessly take 2 pages for this recital of praise so buckle up.

It would honestly be a disservice to them, to not start by thanking my supervisors Ton and Sreeshma. Ton, the devotion and passion you have for the field of study you devoted your academic career to is something I only hope to be lucky enough to find. You drew the first lines on a blank canvas, and I feel honored to have been able to contribute to the field of study that is your life's work. To Sreeshma, somehow you were always excited (or at least seemed excited) when I showed up to your office unannounced to enthusiastically share any tiny result I had. I deeply value your opinion, as you have time and time again shown how great of a researcher you are. Thank you for always hearing me out when I came to you with some ridiculous theory I tried to convince you was really a thing. You made research fun, accessible, and always made me feel like my ideas mattered. Ton and Sreeshma, your efforts have awakened my love for research, and I hope to make you proud when I one day become a Doctor as well.

To my dear roommates, Mirjam, Marieke, Mirthe, Benthe and Sharon, who made our house my home. Thank you for tolerating my late night shenanigans, for watching my thrift store fashion shows, for making me laugh so hard I would cry. The small things mattered, the "how are you?" when I was about to burst into tears when Matlab chose to disobey once again, made it all so much lighter. I feel so lucky to have found a group of girls like you who feel like a little family of just big sisters.

To my phenomenal study friends, Carine, Niels, Rutger, Bas, Essan, Jessie, Sophie, Maartje, and especially Mees, my fellow MMPS warrior. Trough rain or shine, come wind or weather, no matter what, I could always count on you to be there for a coffee, snack, "spar wandeling", or to simply populate the couch at the Leeghwater office. Sometimes we were all in the trenches of academic clownery, but we were there together. The countless days we spent at Landscape were only tolerable because of your presence, and I feel honored to watch you all become fantastic engineers. You can count on me to cheer you on for every victory you ever have.

To Givani, and all my other friends from "Ongecontroleerd" who have truly pulled me through the early control days. Systems and Control might be tough, but we were tougher.

To all of the friends, and family who have absolutely no clue what my thesis is about, but whose support, and love was sometimes the only thing keeping me going, and made sure my life was not only matrices and Matlab. My OHP, my amazing high school friends, my dear friends from JC Appa, my personal future lawyer Nienke, my grandparents, uncles, aunts, and cousins, I hope I have made you proud.

To my sisters Amber and Vira, who have kept me humble by subtly reminding me all I do all day is just sitting in front of a computer. But who at the end of the day, I know are my absolute biggest cheerleaders. We are three sides of the same coin, and I am beyond excited to do life together with you, wherever it takes us.

To my parents, the people who will be beaming with pride as they will watch me present this work, and not even because it is a pivotal moment in my academic career, but because they know how much it took to get here. It is thanks to you I am the person I am today. I do not mean it lightly when I say you are the absolute best parents I could have ever asked for, and therefore I dedicate this thesis to you.

With love, Vicky

Delft, University of Technology
June 20, 2025

“I hope you will go out and let stories happen to you, and that you will work them, water them with your blood and tears and your laughter till they bloom, till you yourself burst into bloom”

— *Clarissa Pinkola Estés*

Chapter 1

Introduction

This first chapter provides a concise overview of what this thesis consists of. Some relevant background and context to the topic is provided in Section 1-1. Subsequently, the academic incentive to perform the research conducted in this report, based on the literature research conducted prior to this thesis [19] will be elaborated on in Section 1-2. Within this section, the research questions answered in this report are given. Thereafter, section 1-3 outlines the approach taken as to answer the research questions. Lastly, the outline of this document is described in 1-4.

1-1 Background

There is a plethora of modeling techniques to choose from in the pursuit of modeling discrete-event systems. Some examples of these are; (extended) state machines, max-plus algebra, hybrid automata, temporal logic, generalized semi-Markov processes, Petri nets, and so on [5]. Max-plus algebra arose as a tool in system theory for classes of discrete event systems around the early 1980's. It is likely inspired by the observation that synchronization, a very nonlinear, non-smooth phenomenon when described in conventional algebra, can be linearly modeled using algebraic structures such as max-plus algebra [12]. Most Discrete Event System (DES) are generally not linear, when written in conventional algebra. When DES can be modeled within the max-plus or min-plus algebra framework, the DES becomes 'linear', i.e. linear in the max-plus or min-plus algebraic sense, which can reduce the computational complexity of controlling, and analyzing the system. Max-Min-Plus-Scaling (MMPS) systems combine max-plus algebra, min-plus algebra, and conventional algebra, and are a powerful modeling framework for discrete-event dynamic systems that exhibit synchronization, competition, and accumulation, common features in applications such as railway networks, manufacturing systems, and communication protocols. By integrating max-plus and min-plus algebra with conventional scaling operations, MMPS systems can describe complex, non-linear behaviors in a mathematically structured yet computationally manageable way. Their strength lies in their ability to represent both temporal states (timing of events such as the arrival of a train) and quantity states (quantities such as the number of passengers embarking a train) within this mathematical structure. MMPS systems are equivalent to continuous piecewise-affine systems, making them a natural tool for control and analysis of systems with hybrid dynamics.

1-2 Problem Description

Despite their broad applicability, important theoretical challenges remain unresolved. In particular, the analysis of implicit MMPS systems, where current states are defined in terms of themselves, poses significant difficulties. The solvability of implicit systems can be violated by these implicit dependencies, but this phenomenon is not fully understood or researched. Existing conditions are largely sufficient but not necessary, leaving gaps in the theoretical foundation. Moreover, as MMPS systems are increasingly applied in control contexts, there is a growing need to understand how input signals, both open-loop and closed-loop, influence solvability and stability. These gaps highlight the importance of developing more comprehensive methods for analyzing, classifying, and controlling implicit MMPS systems, especially in the presence of real-world constraints and disturbances. Previous work such as [18] made attempts to model, and control a complex implicit MMPS system in the form of an Urban Railway System (Urban Railway System (URS)). However, this model does not allow for complex passenger flows throughout the system, which does occur in real-life. Furthermore, very little elaborate examples of analysis, and control of implicit MMPS systems exist, leaving much room for improvement.

1-2-1 Research Questions

This thesis seeks to fill the identified research gap by investigating the following research questions;

1. Is it possible to find a necessary solvability condition for implicit MMPS systems?
   (a) Can a graph-theoretic interpretation be used to understand, and generalize beyond the current algebraic criteria?
   (b) What degrees of solvability exist for implicit MMPS systems?
   (c) Is it possible to identify a method to classify all implicit MMPS systems with regards to their degree of solvability?
2. How can the existing control strategies for explicit MMPS systems be extended to control strategies for implicit MMPS systems?
   (a) Is it possible to find an open-loop control strategy for implicit MMPS systems?
   (b) Is it possible to find a closed-loop control strategy for implicit MMPS systems?
   (c) Can conditions for system properties such as time-invariance and solvability be derived for controlled implicit MMPS systems?

3. Can the theoretical results regarding solvability and control of implicit MMPS systems be validated and/or tested by applying them to a complex real-world system such as an Urban Railway System?
   (a) Is it possible to augment the Urban Railway System proposed in [18] such that it accommodates complex passenger flows?
   (b) What insights can be gained from analyzing the dynamic behaviour of this Augmented Urban Railway System?
   (c) Can this Augmented Urban Railway System subsequently be simulated according to a uniform timetable, disturbed, and controlled using the proposed control strategies for implicit MMPS systems?

1-3 Approach

After introducing the subject in 1-1, identifying the research gap, and formulating the research questions in 1-2 let us introduce an appropriate approach as to answer these research questions. Before attempting to answer said questions, thorough investigation of existing theory and literature is essential. An overview of the mathematical foundation is presented, covering all known relevant literature. A thorough analysis of the existing solvability condition is done with the aim to relate the structure matrix S to a graph-theoretic interpretation of solvability, thereby answering research question 1.(a). Thereafter, the sufficiency of the existing solvability condition is proven. The graph-theoretic interpretation of the existing solvability condition is of great importance when attempting to develop a method to classify implicit systems in terms of their degree of solvability, which will be subsequently investigated, ultimately answering research questions 1.(b) and 1.(c). The extension of the solvability theory is concluded by proposing a necessary condition for solvability, answering research question 1.

Afterwards, the existing explicit control strategies will be extended in order to incorporate implicit dynamics, addressing research question 2. This is subsequently done for both open-loop control strategies, and closed-loop control strategies, by which research questions 2.(a) and 2.(b) are answered. The results regarding solvability theory are applied to the derived control strategies, and existing time-invariance conditions are rederived for the open-loop-, and closed-loop controlled systems in the pursuit of answering research question 2.(c).

A comprehensive case study as to validate these results is conducted in the form of augmenting the Urban Railway System (URS) as presented in [18]. By close examination of the assumptions done in this research, the mathematical model will be redesigned as to allow for more complex flows of passengers, addressing research question 3.(a). Thereafter, extensive analysis of this Augmented Urban Railway System is performed. Topics like solvability, time-invariance, stability, initialization, etcetera will be elaborated on with regards to this newly derived system, by which research question 3.(b) is addressed. Said analysis subsequently serves as a validation of the results regarding the proposed solvability theory. Furthermore, the simulation of the Augmented Urban Railway System will be disturbed, and controlled using the proposed implicit MMPS control strategies, addressing research question 3.(c).

1-4 Document Outline

The thesis is structured such that the reader experiences a coherent flow of reasoning, guiding them through the concepts, methods, and results in a natural order. Below, the chapters are presented in the order they appear in the thesis, along with a summary of their respective contents. Note that Chapters 4, 5, 6 and 7 provide the academic contributions, whereas Chapters 1 and 3 provide an overview of the literature study, and introduce the concepts that are of importance for the research.

Chapter 2 - Mathematical preliminaries: Introduces the existing algebraic foundations of max-plus, and min-plus algebra, and the MMPS modeling framework, including some relevant spectral theory concepts, and piecewise-affine system equivalence.

Chapter 3 – Analysis of MMPS systems: Discusses existing theory regarding key system properties such as time-invariance, monotonicity, and homogeneity, distinguishing between explicit and implicit MMPS systems. Algorithms for solvability and eigenvalue analysis are given. Furthermore existing theory regarding bounded-buffer stability, (maximal) invariant sets, and conditions under which MMPS systems remain stable over time is explained.

Chapter 4 – Solving Solvability: Evaluates existing solvability theory, identifies knowledge gaps, and presents new solvability theory using graph-theoretic tools and matrix structure analysis. The chapter defines necessary and sufficient conditions and proposes a methodology to classify implicit MMPS systems according to their degree of solvability.

Chapter 5 – Control of Implicit MMPS Systems: Proposes open-loop and closed-loop control strategies for implicit MMPS systems. Conditions for system properties such as solvability and time-invariance are analytically derived.

Chapter 6 – Augmenting and Analyzing the Urban Railway System: Applies the MMPS framework to a real-world transportation system. The model is extended and validated through simulation, demonstrating the theory's practical relevance.

Chapter 7 – Disturbance and Control of the Augmented Urban Railway System: Evaluates the effects of disturbances on the AURS and implements a proposed control strategy to reject the applied disturbance. The results of the applied control strategy are thoroughly discussed.

Chapter 8 - Conclusions and Contributions: Reflects on all research questions and shortly summarizes the answers to each. Provides a concise overview of all the research carried out in this thesis.

Chapter 2

Mathematical Preliminaries

The aim of this Chapter is to provide an elaborate, complete, yet concise mathematical basis for understanding the concept of MMPS systems. Firstly, section 2-1 provides the core concepts of what makes max-plus algebra, what properties operators have, and the properties of max-plus algebra will be discussed in, all of which will be done for scalars. Subsequently, this knowledge will be applied to min-plus algebra. Thereafter, the theory will be extended towards matrices and vectors. Then, section 2-2 provides insights into spectral theory for max-plus algebra, focusing on graph theory, and the graph-theoretic interpretation of max-plus matrices. Thereafter, max-plus algebraic eigenvalues and eigenvectors will be introduced, and an algorithm for determining them will be presented. Lastly, in section 2-3 a model for MMPS systems will be introduced, as well as the class of Continuous Piecewice Affine systems, which is a class equivalent to MMPS systems.

2-1 Max-Plus Algebra

The aim of this section is to introduce the concept of max-plus algebra, and lay the mathematical foundation on which all topics discussed after this is built. After reading this section, the reader should be able to understand what the basic operations of max-plus algebra are, and how to apply them to matrix operations as well as scalar operations. Furthermore, the concept of min-plus algebra is defined.

2-1-1 Definitions and Core Concepts

The cornerstones of max-plus algebra are the operations of maximization and addition. These two binary operations are represented by the mathematical symbols $\oplus$ "oplus" for maximization, and $\otimes$ "otimes" for addition. From an algebraic point of view, it can be shown that max-plus algebra is an example of an algebraic structure called a semiring. The following general definition of a semiring is used.

Definition 2-1.1. *[9] (semiring) A semiring is a nonenmpty set R, endowed with two binary operations $\oplus_R$ and $\otimes_R$ such that;*

- *The operation $\oplus_R$ is associative, commutative, and has a zero element ε_R*
- *The operation $\otimes_R$ is associative, and it is distributive with respect to $\oplus_R$ and its identity element e_R satisfies $\varepsilon_R \otimes_R e_R = e_R \otimes_R \varepsilon_R = \varepsilon_R$, so essentially, ε_R is absorbing for $\otimes_R$*

Any semiring can be denoted by the notation $\mathcal{R} = (R, \oplus_R, \otimes_R, \varepsilon_R, e_R)$

Associative means that it does not matter how the elements were grouped, i.e. which part we calculate first. For example, $(2 \times 3) \times 6 = 2 \times (3 \times 6)$, where the order of evaluation is not of significance for the result. *Commutative* means that the order of elements in the operation does not matter, for example $3 + 2 = 5$, just like $2 + 3 = 5$, so even though the order of 2 and 3 changed, the answer stayed the same.

Furthermore, the semiring is *idempotent* [13] if the first operation is idempotent, which means that a mathematical quantity, when applied to itself under a given binary operation, equals itself. For example, this holds for $\max(a, a) = a$. Also, the semiring is commutative if the group is commutative.

Operations

Here, the operations of maximization and addition, $\oplus$ and $\otimes$ will be introduced. For $\oplus$ and $\otimes$ we define

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

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

for $a, b \in \mathbb{R}_\varepsilon$. Within max-plus algebra, $\varepsilon = -\infty$ and $e = 0$ are defined as the neutral elements of $\oplus$ and $\otimes$, they function as the "zero" and "one", respectively [13]. This is analogous to how in conventional algebra, 0 and 1 are used as the neutral elements. The algebraic structure that represents max-plus algebra is given by;

$$\bar{\mathbb{R}}_\varepsilon = (\mathbb{R}_\varepsilon, \oplus, \otimes, \varepsilon, e) \tag{2-3}$$

Where $\mathbb{R}_\varepsilon = \mathbb{R} \cup -\infty$. Sometimes in literature, the set $\mathbb{R}_\varepsilon$ is referred to as $\mathbb{R}_{\max}$ [13], however, in this report, $\mathbb{R}_\varepsilon$ is used. There is no inverse operation of the $\oplus$ operation. An example of an inverse operation in conventional algebra would be multiplication, whose inverse operation is division. For the conventional operation of addition, the inverse operation is subtraction. This symmetry of conventional algebra is at the expense of the property of idempotency. In max-plus algebra, symmetry is lost, but idempotency is gained. Interestingly enough, the symmetry of the conventional multiplication operation, makes the conventional algebra defined by binary operations $+$ and $\times$, not a semiring.
Let r now be defined as $r \in \mathbb{R}$. In max-plus algebra, the rth max-plus algebraic power of $x \in \mathbb{R}$ can be defined as $x^{\otimes^r}$[6]. In conventional algebra, this corresponds to $x^{\otimes^r} = rx$. This leads to the following theorem;
Theorem 2-1.1. *[6] If $x \in \mathbb{R}$, then $x^{\otimes^0} = 0$, and the inverse of x with respect to $\otimes$ is $x^{\otimes^{-1}} = -x$*

Using this theorem it can be explained that there exists no inverse element for ε with respect to $\otimes$, as any ε is absorbing for $\otimes$. Lastly, max-plus algebraic powers have priority over max-plus algebraic multiplication, which then have priority over max-plus algebraic addition. The max-plus algebraic order of evaluation corresponds to the evaluation order within conventional algebra. A few other mathematical properties regarding max-plus and min-plus algebra are given below;

- $-\min(a, b) = \max(-a, -b)$
- $-\max(a, b) = \min(-a, -b)$
- $\min(a, \min(b, c)) = \min(\min(a, b), c)$
- $\max(c, \min(a, b)) = \min(\max(c, a), \max(c, b))$
- $\min(c, \max(a, b)) = \max(\min(c, a), \min(c, b))$

Min-plus Algebra

In this section, the concept of min-plus algebra will be introduced, which is equally as important for the construction of MMPS systems as max-plus algebra. As defined in the previous Section, the algebraic structure for max-plus algebra is given by $\bar{\mathbb{R}}_\varepsilon = (\mathbb{R}_\varepsilon, \oplus, \otimes, \varepsilon, e)$. The structure describing min-plus algebra is given by $\bar{\mathbb{R}}_\top = (\mathbb{R}_\top, \oplus', \otimes', \top, e)$. Here, $\top = \infty$, and $\mathbb{R}_\top$ is the set of real numbers, and $\top$, so $\mathbb{R}_\top = \mathbb{R} \cup \{\infty\}$. Min-plus algebra is isomorphic to max-plus algebra, meaning that, because of the structural similarities between the two, all concepts in max-plus algebra can be transformed to concepts in min-plus algebra [17]. But first, the operators of min-max algebra have to be properly defined;

$$\begin{aligned} a \oplus' b &= \min(a, b) \\ a \otimes' b &= a \otimes b = a + b \end{aligned} \tag{2-4}$$

So, having introduced the min-plus algebraic structure, its importance in MMPS systems can be highlighted. In MMPS systems, the operations of maximization, minimization, addition, and scaling are used. Therefore, min-plus algebra is used for the minimization and addition, max-plus algebra for the maximization and addition, and conventional algebra for the scaling operation. All three defined types of algebras will occur. For max-min-plus (scaling) algebra, the set $\mathbb{R}_c = \mathbb{R} \cup \{-\infty\} \cup \{\infty\}$ can be defined. The notation $\mathcal{R}$ can be used to refer to either $\mathbb{R}$,$\mathbb{R}_\varepsilon$, $\mathbb{R}_\top$ or $\mathbb{R}_c$ [16]. In order to transform concepts from max-plus algebra to min-plus algebra, generally speaking, the maximization operations have to be changed to the minimization operation, and the set over which the algebra is defined has to be changed from $\mathbb{R}_\varepsilon$ to $\mathbb{R}_\top$.

2-1-2 Matrix Calculations

Now that the max-plus algebraic basis is established, this knowledge can be extended to, and applied to matrix calculations in $\mathbb{R}_\varepsilon$ [9]. Let us introduce n,m and p in the set of positive integers, $n, m, p \in \mathbb{Z}^+$, and matrices A and B in $A, B \in \mathbb{R}_\varepsilon^{n \times m}$. Elements of A and B will be referred to as a_{ij} or b_{ij}, with $i \in n$ and $j \in m$ as the rows and columns, respectively. Furthermore, matrix $C \in \mathbb{R}_\varepsilon^{n \times p}$ is introduced, with elements c_{ij} with $i \in n$ and $j \in p$.

Matrix A can be written as;

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{pmatrix} \tag{2-5}$$

And matrices B and C can be written as

$$B = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ b_{21} & b_{22} & \cdots & b_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nm} \end{pmatrix} \tag{2-6}$$

$$C = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{np} \end{pmatrix} \tag{2-7}$$

The $\oplus$ operator can be applied to matrices in the following way;

$$(A \oplus B)_{ij} = a_{ij} \oplus b_{ij} = \max\left(a_{ij}, b_{ij}\right), \forall i, j \tag{2-8}$$

Applying the $\otimes$ operator to matrices yields the following relation;

$$(A \otimes C)_{ij} = \bigoplus_{k=1}^{n} a_{ik} \otimes c_{kj} = \max_{k=1,\dots,n}\left(a_{ik} + c_{kj}\right), \forall i, j \tag{2-9}$$

The elements of $\mathbb{R}_\varepsilon^{n\times 1}$ are called the vectors of the matrices. The jth column of matrix A can be referred to as $\mathbf{a}_j$, or $[\mathbf{a}]_j$. Notice that $\alpha \otimes \mathbf{x}$, $\mathbf{x}$ being a vector in $\mathbb{R}_\varepsilon^{n\times 1}$ and for any scalar $\alpha \in \mathbb{R}_\varepsilon$, is actually a vector with all entries of value α. In Section 2-1-1, the neutral elements of max-plus algebra were defined as $\varepsilon = -\infty$ and $e = 0$. Within max-plus algebraic matrix computations, an identity matrix, and a zero matrix need to be defined as well. Let us define $E(n, n)$ and $\mathcal{E}(n, m)$ as follows [9];

$$E = \begin{pmatrix} e & \varepsilon & \cdots & \varepsilon \\ \varepsilon & e & \cdots & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ \varepsilon & \varepsilon & \cdots & e \end{pmatrix} \tag{2-10}$$

$$\mathcal{E} = \begin{pmatrix} \varepsilon & \varepsilon & \cdots & \varepsilon \\ \varepsilon & \varepsilon & \cdots & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ \varepsilon & \varepsilon & \cdots & \varepsilon \end{pmatrix} \tag{2-11}$$

The max-plus identity matrix $E(n, n)$ has e on its diagonal entries, and ε at all other entries, and is always a square matrix. The zero matrix $\mathcal{E}(n, m)$ has ε at all entries. Using any arbitrary matrix A, it is easily verified that the following statements hold;

$$A \oplus \mathcal{E}(n, n) = A = \mathcal{E}(n, n) \oplus A \tag{2-12}$$

$$A \otimes E(m, m) = A = E(n, n) \otimes A \tag{2-13}$$

Furthermore, for $k \geq 1$ the following tho statements hold as well;

$$A \otimes \mathcal{E}(m, k) = \mathcal{E}(n, k) \qquad \mathcal{E}(k, n) \otimes A = \mathcal{E}(k, m) \tag{2-14}$$

Besides the identity, and zero matrices, the unit vector, and zero vector needs to be defined as well. The unit vector can be denoted by u, or $[\mathbf{u}]_j$ mathematically, and each entry of this vector has value e [9]. For any $j \in n$, the jth column of of the identity matrix $E(n, n)$ is called the jth base vector of $\mathbb{R}_\varepsilon^n$ and is denoted by e_j.

For $\mathbb{R}_\varepsilon^{n\times m}$, the matrix addition operation $\oplus$ is associative, commutative, and has zero element $\mathcal{E}(n,m)$ [9]. The matrix product operation $\otimes$ is associative, distributive with respect to $\oplus$, has unit element $E(n,m)$, and the zero matrix $\mathcal{E}(n,m)$ is absorbing for $\oplus$. The transpose of a max-plus algebraic matrix A can be determined analogous to how the transpose of a matrix in conventional algebra is determined, and is denoted by A^T. A^T is determined as follows;

$$[A^T]_{ij} = [A]_{ji} \tag{2-15}$$

Lastly, the method to compute the power of a max-plus algebraic matrix needs to be provided. Higher powers of matrices can only be computed for square matrices, so matrices in $\mathbb{R}_\varepsilon^{n\times n}$. The k^{th} power of matrix A can be written down as $A^{\otimes^k}$, and the definition of the computation of matrix powers is given as;

Definition 2-1.2. *[9] For any matrix $A \in \mathbb{R}_\varepsilon^{n\times n}$, denote the kth power of A by $A^{\otimes^k}$, defined by*

$$A^{\otimes^k} = \underbrace{A \otimes A \otimes \cdots \otimes A}_{k\ times} \tag{2-16}$$

for $k \in \mathbb{N}$ with $k \neq 0$

Lastly, we set $A^{\otimes^0} = E(n,n)$.

2-2 Spectral Theory

In the previous section, the necessary mathematical base principles of max-plus algebra for both scalars, and matrices were introduced. This section deepens that mathematical knowledge, as it provides a graph-theoretic interpretation of max-plus matrices, and an introduction to eigenvalues and eigenvectors in max-plus algebra.

2-2-1 Graph Theory

There exists a rich relationship between graphs and matrices [13], which is of great use within max-plus algebra. The extent of this relationship, and its implications on max-plus algebra will be elaborated on within this Section. The basic observation is that any square matrix can be translated into a weighted graph, and that the products and powers of matrices over the max-plus semiring have entries with a graph-theoretical interpretation [9]. In this whole section [13] is used as a source. A *directed graph* $\mathcal{G}$ is defined as a pair $(\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is defined as a set of nodes, and $\mathcal{E}$ is defined as a set ordered pars of nodes, called arcs. The pair can consist the same node twice (an arc to itself), and it is also possible to have multiple arcs between a pair of nodes, in which case we speak of a multigraph. An *undirected graph* is a graph in which there is no specified order for arcs. The focus in this Section is almost exclusively on directed graphs with at most one arc between any two nodes, so it can be assumed that the word "graph" refers to a directed graph. Now, some basic concepts and definitions will be introduced, that will lay the base in pursuit of understanding graph theory, which is useful in the concepts introduced later on.

Predecessor, successor If in a graph $(i, j) \in \mathcal{E}$, then i is called the predecessor of j, and j is called the successor of i. The set of predecessors is denoted by the notation $\pi(j)$, and the set of successors is denoted by $\sigma(i)$.

Path, circuit, loop, lengths A path p is a sequence of nodes $(i_1, i_2, \ldots, i_p)\,, p > 1$ such that $i_j \in \pi\,(i_{j+1})\,, j = 1, \ldots, p-1$. This means that for each node on the path, the current node is in the set of predecessors of the next node on the path. The path starts at node i_1 and ends at node i_p. An *elementary* path is a path in which no node appears more than once. If the initial and final node coincide, the path is actually a *circuit*. A circuit $(i_1, i_2, \ldots, i_p = i_1)$ is an *elementary circuit* if the path $(i_1, i_2, \ldots, i_{p-1})$ is elementary. A *loop* is actually a circuit that consists of one node, that is both the initial and final node. Basically, it is a path that consists of an arc from a node to itself. If $i \in \pi(i)$, this loop exists. The *length* of a path or a circuit is the sum of the lengths of the arcs it is composed of. The length of an arc is assumed to be 1, unless specified otherwise. The length of a path p is denoted by the notation p_l. The set of all paths and circuits in a graph is denoted by R. If R does not contain any circuits, the graph is said to be acyclic.

Descendant, ascendant The set of descendants $\sigma^+(i)$ of node i consists of all nodes j such that a path exists from i to j. Similarly, the set of ascendants $\pi^+(i)$ of node i is the set of all nodes j such that a path exists from j to i. The mapping $i \mapsto \pi^*(i) = \{i\} \cup \pi^+(i)$ is the transitive closure of π; the mapping $i \mapsto \sigma^*(i) = \{i\} \cup \sigma^+(i)$ is the transitive closure of σ.

Chain, connected graph A graph is called *connected* if for all pairs of nodes i and j either a path from i to j, or a path from j to i exists. An undirected path is called a chain. So a graph is called connected if for all pairs of i and j there exists a chain joining i and j.

Strongly connected A graph is called *strongly connected* if for any two different nodes i and j there exists a path from i to j. Mathematically this can be described as $i \in \sigma^*(j)$ for all $i, j \in \mathcal{V}$ with $i \neq j$. An isolated node, with or without a loop, is also a strongly connected graph, by definition.

Weights directed graph is called *weighted* if a weight $w(i, j) \in \mathbb{R}$ is associated with any arc $(i, j) \in \mathcal{E}$ [9]

Reduced graph The reduced graph of $\mathcal{G}$ is the graph with nodes $\overline{\mathcal{V}} \overset{\text{def}}{=} \{1, \ldots, q\}$ and with arcs $\overline{\mathcal{E}}$ where $(i, j) \in \overline{\mathcal{E}}$ if $(k, l) \in \mathcal{E}$ for some node k of $\mathcal{V}_i$ and some node l of $\mathcal{V}_j$

Lemma 2-2.1. *Any reduced graph is acyclic*

Connection to Matrices

Considering all previously introduced knowledge, the strong connection between graph theory, matrices, and its applications in max-plus algebra will be elaborated on. The relation between graphs and matrices over the max-plus semiring $\mathbb{R}_\varepsilon$ is described below. Firstly, any square $(n \times n)$ matrix A in $\mathbb{R}_\varepsilon$ can be associated with a graph, which is denoted as $\mathcal{G}(A)$[9]. The graph associated with matrix A is called the precedence graph of A, of which the definition is given by;

Definition 2-2.2. *[9](Precedence graph) Consider matrix $A \in \mathbb{R}_{\varepsilon}^{n\times n}$. The precedence graph of A, denoted by $\mathcal{G}(A)$, is a weighted directed graph with vertices $1, 2, ..., n$ and an arc (i, j) in $\mathcal{G}(A)$, the weight of (i, j) is given by a_{ji} for each $a_{ji} \neq \varepsilon$.*

The set of nodes of this graph is given by $\mathcal{E}(A)$, and contains n nodes. Any entry of matrix A corresponds to a possible arc. If the value of $a_{ji} \neq \varepsilon$, an arc between node i and j exists, which mathematically translates to $(i, j) \in \mathcal{V}(A) \Leftrightarrow a_{ji} \neq \varepsilon$. Here, the set of arcs is denoted by $\mathcal{E}(A)$. Next, let us introduce the Kleene star operator of matrix A [9]; The Kleene star operator $A^\star$ is obtained by the following expression;

$$A^\star = E \otimes A^+ = \bigoplus_{k\geq 0} A^{\otimes k} \tag{2-17}$$

Where

$$A^+ = \bigoplus_{k=1}^{\infty} A^{\otimes k} = A \oplus A^{\otimes 2} \oplus A^{\otimes 3} + \cdots + \oplus A^{\otimes n} \in \mathbb{R}_{\varepsilon}^{n\times n} \tag{2-18}$$

Here, entry $[A^\star]_{ij}$ is the maximal weight of any path of arbitrary length in $\mathcal{G}(A)$ between node j and node i. Lastly, it can be noted that $[A^{\otimes n}]_{ij}$ refers to the maximal weight of a path from node j to node i in $\mathcal{G}(A)$ of length n.

2-2-2 Eigenvalues and Eigenvectors

Given a matrix A in with entries in $\mathbb{R}_\varepsilon$, statements can be made about the existence of eigenvalues and eigenvectors, and how these can be computed. Firstly, the definition of a max-plus algebraic eigenvalue is given as follows ;

Definition 2-2.3. *[13] (Max-plus algebraic eigenvalue) Let $A \in \mathbb{R}_{\varepsilon}^{n\times n}$. If there exists $\lambda \in \mathbb{R}_\varepsilon$ and $v \in \mathbb{R}_{\varepsilon}^{n}$ with $v \neq \varepsilon_{n\times 1}$ such that $A \otimes v = \lambda \otimes v$ then, λ is a max-plus-algebraic eigenvalue of A and that v is a corresponding max-plus algebraic eigenvector of A*

This definition is similar to the definition of conventional eigenvalues and eigenvectors. But where in conventional algebra any matrix $A \in \mathbb{R}^{n\times n}$ has n eigenvalues and n eigenvectors, this does not hold true for matrices in $A \in \mathbb{R}_{\varepsilon}^{n\times n}$. The total number of max-plus algebraic eigenvalues and eigenvectors is generally less than n [6]. In fact, it can even be proven that if a matrix is irreducible, it has only one eigenvalue [3]. The first possible method for computing an eigenvalue of a max-plus algebraic matrix, is given by Karp's theorem. This theorem is accompanied by an algorithm, Karp's algorithm, which computes the eigenvalue of an irreducible max-plus algebraic matrix.

Theorem 2-2.1. *[9] (Max-plus eigenvalue of an irreducible matrix) Let $A \in \mathbb{R}_{\max}^{n\times n}$ be irreducible with eigenvalue λ, then;*

$$\lambda = \max_{i=1,\ldots,n} \min_{k=0,\ldots,n-1} \frac{[A^{\otimes n}]_{ij} - [A^{\otimes k}]_{ij}}{n-k} \tag{2-19}$$

where $j \in \underline{n}$ can be chosen arbitrarily, and division has to be understood in conventional algebra.

Subsequently, the algorithm for determining the eigenvalue described in Karp's Theorem is given by;
Theorem 2-2.2. *The max-plus algebraic eigenvalue of an irreducible matrix*

Algorithm 1 [11] Karp's algorithm

1: Choose arbitrary $j \in \underline{n}$, and set $x(0) = e_j$
2: Compute $x(k)$ for $k = 0, ..., n$
3: Compute the eigenvalue eigenvalue λ as

$$\lambda = \max_{i=1,\dots,n} \min_{k=0,\dots,n-1} \frac{x_i(n) - x_i(k)}{n-k}$$

Karp's algorithm is actually applicable to reducible matrices, as it yields an eigenvalue depending on the choice of j. In [9] it is shown that, in this case, the associated eigenvector may contain elements equal to ε. Furthermore, Karp's algorithm does not propose any method to determine the eigenvector corresponding to the proposed eigenvalue. The second proposed algorithm computes an eigenvalue for any square max-plus algebraic matrix, contains a method to find the corresponding eigenvector, and is called the Power algorithm;

Algorithm 2 Power algorithm for max-plus matrices[16]

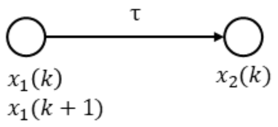
1: Take an arbitrary initial vector $x(0) = x_0 \neq \varepsilon\mathbf{1}$,where $\varepsilon\mathbf{1}$ is a vector with all entries ε, such that, x_0 has at least one finite element.
2: Iterate $x(k) = A \otimes x(k-1)$ until there are integers p and q such that $x(p) - x(q) = c\mathbf{1}$, where $p > q >= 0$, and c is a real number
3: Compute eigenvalue λ as $\lambda = \frac{c}{p-q}$
4: Compute the eigenvector v as $v = \bigoplus_{j=1}^{p-q} \left(\lambda^{\otimes(p-q-j)} \otimes x(q+j-1)\right)$

2-3 MMPS Systems

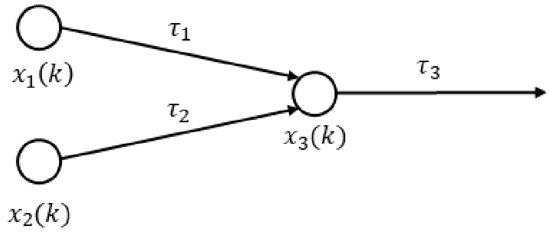
While the previous section laid a foundation in understanding the basic concepts of max-plus algebra and some important computations and analyses, this section focuses on introducing the max-min-plus-scaling systems. The maximization, minimization, addition and scaling operations form the baseline of the MMPS framework. Each of these four operations serve their purpose in describing discrete-event systems into an MMPS system [18].

1. **Maximization** Consider a situation with either sequential processing, or synchronization. Sequential processing means that, an operation of the next cycle can only start when the operation of the current cycle is completed. Let $u_1(k+1)$ be the earliest possible start time of operation x_1 in cycle $k+1$. Using the *max* operation, the starting time $x_1(k+1)$ is given by $x_1(k+1) = \max(x_1(k) + \tau, u_1(k+1))$. In the case of synchronization, consider operations $1, 2$ and 3 where operation 3 can only start when operations 1 and 2 have been completed. Starting time $x_3(k)$ can therefore be given by $x_3(k) = \max(x_1(k) + \tau_1, x_2(k) + \tau_2)$.

So the latest finish time of the previous operations is used as the starting time for the next operation. These maximization operations are shown in figures 2-1a and 2-1b.



(a) Sequential processing for maximization



(b) Synchronization for maximization

Figure 2-1: Maximization operation

2. **Minimization** The minimization operation has a 'first come, first serve' principle. Again, consider operations $1, 2$ and 3. Operation 3 will start as soon as either operation 1 or 2 has been completed. So using the *min* operation, starting time $x_3(k)$ can be given by $x_3(k) = \min(x_1(k) + \tau_1, x_2(k) + \tau_2)$. The minimization operation is shown in Figure 2-2

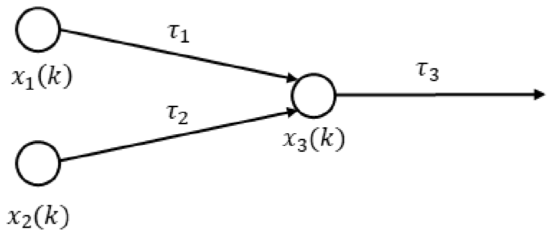


Figure 2-2: Minimization operation

3. **Addition** Let $x_1(k)$ and $x_2(k)$, be the start and finish time of event cycle k, and let τ be the processing time. The relation between $x_1(k)$ and $x_2(k)$ can be given by the *plus* operation; $x_2(k) = x_1(k) + \tau$. The operation of addition is shown in Figure 2-3

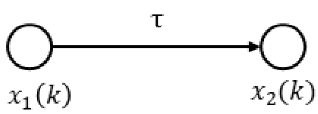
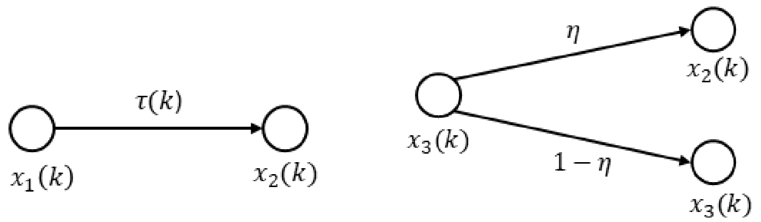


Figure 2-3: Plus operation

4. **Scaling** Let us illustrate two relevant scenario's using the scaling operation. Firstly, when the processing time is state dependent, the relation between starting time $x_1(k)$ and starting time $x_2(k)$ includes a scaling element. The following relation can be derived; $x_2(k) = x_1(k) + \alpha + \beta^T x(k)$, where $\tau(k) = \alpha + \beta^T x(k)$. Secondly, if a quantity state $x_1(k)$ splits in two new quantity states $x_2(k)$ and $x_3(k)$ with ratio η and $1-\eta$, the quantity states can be described using a scaling operation. The following expression can be derived; $x_2(k) = \eta x_1(k)$ and $x_3(k) = (1-\eta)x_1(k)$. The concept of a quantity state will be elaborated on later in this Section. Figures 2-4a and 2-4b show the scaling operations.



(a) State dependent processing time in scaling operation

(b) Splitting of quantity states in scaling operation

Figure 2-4: Scaling operation

2-3-1 State-Space model

Using the four introduced operations, a few ways of describing general MMPS state-space expressions can be introduced. First, we introduce the definition of the max-min-plus-scaling (MMPS) expression ;

Definition 2-3.1. *[5] A max-min-plus-scaling expression $f : \mathcal{R}^m \rightarrow \mathcal{R}$ of the variables $x_1, x_2, ..., x_n$ is defined by the grammar*

$$f := x_i|\alpha|\max(f_k, f_l)\,|\min(f_k, f_l)|\,f_k + f_l \mid \beta f_k \tag{2-20}$$

with $i \in \{1, 2, ..., n\}$, $\alpha \in \mathbb{R}$, and f_k and f_l are again max-min-plus-scaling expressions over the set $\mathcal{R}$. The symbol '$|$' means "or". For vector-valued MMPS functions, the given statements hold component-wise.

Keep in mind that in this expression, elements of max-plus algebra (maximization and addition), min-plus algebra (minimization and addition) and conventional algebra (scaling) appear. Therefore, this system has lost the property of linearity within the max-plus algebraic framework. Using the MMPS expression as given in (2-20), the general definition of an MMPS system can be given

Definition 2-3.2. *[18] (Max-min-plus-scaling system) The vector given by*

$$p(k) = \left[x^T(k), x^T(k-1), ..., x^T(k-M)u^T(k), w^T(k)\right]^T \tag{2-21}$$

consists of the state $x \in \mathcal{R}^n$, the control input $u \in \mathcal{R}^p$, external signal $w \in \mathcal{R}^z$, where $p \in \mathcal{P} \subseteq \mathcal{R}^{n_p}$. A max-min-plus-scaling (MMPS) system description can be given by the following state-space model where f is a vector-valued MMPS function of the vector p;

$$x(k) = f(p(k)) \tag{2-22}$$

As this is a general model, a few specific cases can be discussed as well.
For example, if the state $x(k)$ does not depend on itself, i.e. the system is explicit, the vector $p(k)$ looks as follows;

$$p(k) = \left[x^T(k-1), ..., x^T(k-M)u^T(k), w^T(k)\right]^T \tag{2-23}$$

In the case of an autonomous system, the system is without external input. An autonomous system can be both implicit, or explicit. The implicit form is given by;

$$p(k) = [x^T(k), x^T(k-1), ..., x^T(k-M)]^T \tag{2-24}$$

Any system described using the MMPS framework will have states that represent the times at which operations for event cycle k start, and end, called the temporal states. However, some states may represent a quantity rather than a time, for example, the number of passengers in a train [18]. Therefore, let us distinguish between two types of states within MMPS systems, namely, temporal states and quantity states. The state of an MMPS system is therefore denoted as [18]

$$x(k) = \begin{bmatrix} x_{\mathrm{t}}(k) \\ x_{\mathrm{q}}(k) \end{bmatrix} \tag{2-25}$$

Here, $[x_t(k)]$ refers to the temporal states, and $[x_q(k)]$ refers to the quantity states. $[x_t(k)]_i$ Represents the time instant at which event i occurs for the kth time. $[x_q(k)]_j$ Denotes the value of the jth quantity at time step k. Subsequently, the vector $p(k)$ can be properly divided as well, given that [18];

$$p(k) = \begin{bmatrix} p_t(k) \\ p_q(k) \end{bmatrix} \tag{2-26}$$

With

$$\begin{aligned} p_t(k) &= \begin{bmatrix} x_t^T(k), & x_t^T(k-1), & u_t(k) \end{bmatrix}^T \\ p_q(k) &= \begin{bmatrix} x_q^T(k), & x_q^T(k-1), & u_q(k) \end{bmatrix}^T \end{aligned} \tag{2-27}$$

where $p_t \in \mathcal{P}_t$ an $p_q \in \mathcal{P}_q$. Using these definitions, the MMPS system can be rewritten as;

$$\begin{aligned} x_t(k) &= f_t(p_t(k), p_q(k)) \\ x_q(k) &= f_q(p_t(k), p_q(k)) \end{aligned} \tag{2-28}$$

2-3-2 Continuous Piecewise-Affine Systems

It can be shown that, the introduced class of MMPS system is mathematically equivalent to the class of Continuously Piecewise Affine systems. In [7], the equivalence between continuous piecewise-affine (PWA) systems and MMPS systems is proven. Firstly, the definition of continuous PWA function is given by;

Definition 2-3.3. *[7] (Continuous piecewise-affine function) A scalar-valued function $f : \mathbb{R}^n \to \mathbb{R}$ is considered a continuous PWA function if and only if the following conditions hold;*

1. *The domain space $\mathbb{R}^n$ is divided into a finite number of polyhedral regions $R_{(1)}, ..., R_{(n)}$*
2. *For each $i \in \{1, ..., n\}$, f can be expressed as $f(x) = \alpha_{(i)}^{\mathrm{T}} x + \beta_{(i)}$ for any $x \in R_{(i)}$ with $\alpha_{(i)} \in \mathbb{R}^n$ and $\beta_{(i)} \in \mathbb{R}$*
3. *f is continuous on any boundary between two regions*

Furthermore, a vector-valued function is continuously PWA if each of its components in continuous PWA. From PWA functions, PWA systems can be constructed, given the following expression [7];

$$x(k) = \mathscr{P}_x(x(k-1), u(k)), \quad y(k) = \mathscr{P}_y(x(k), u(k)) \tag{2-29}$$

Here, $\mathscr{P}_x$ and $\mathscr{P}_y$ are vector valued PWA functions. In the case that $\mathscr{P}_x$ and $\mathscr{P}_y$ are continuous, it can be said that the system is continuously PWA. Next, the equivalence between MMPS and PWA functions can be proven;

Definition 2-3.4. *(Continuous PWA functions as MMPS functions) If f is a continuous PWA function of the form given in definition 2-3.3, then there exist index sets $I_1, ..., I_l \subseteq \{1, ..., N\}$ such that;*

$$f = \max_{j=1,\ldots,\ell} \min_{i \in I_j} \left(\alpha_{(i)}^{\mathrm{T}} x + \beta_{(i)} \right) \tag{2-30}$$

Which is an MMPS function according to (2-20). Therefore, any MMPS function is also a continuous PWA function.

From this definition, it can be concluded that for a given continuous PWA system, there exists an MMPS system (and vice versa) such that the input-output behaviour of both models coincides.

Chapter 3

Analysis of MMPS Systems

This chapter aims to provide mathematical tools to analyze the dynamic behaviour of MMPS systems up to the boundaries of existing research. Section 3-1 elaborates on various system properties of boht implicit, and explicit systems, such as time-invariance, monotonicity, non-expansiveness, et cetera. Secondly, section 3-2 discusses the property of solvability, and addresses the existing theory regarding how to determine whether an implicit MMPS system is solvable. The following section, section 3-3 illustrates the concepts of eigenvalues and eigenvectors in the context of MMPS systems. Thereafter, steady-state behaviour of MMPS systems is shortly discussed in section 3-4. The section after that, section 6-8-1 presents existing theory regarding stability of MMPS systems. Concepts such as linearizing the MMPS system into conventional algebra are explained, which give rise to the theory presented in the last two sections. The region for which the linearization is valid, Ω_θ is elaborated on in section 3-6. Lastly, section 3-7 discusses the concepts of maximal invariant sets, and provides an algorithm on how to determine the maximal invariant set of an MMPS system.

3-1 Properties of MMPS Systems

In the pursuit of thorough analysis, let us distinguish between two types of MMPS systems, implicit MMPS systems and explicit MMPS systems. It is desirable to rewrite an implicit system into an explicit system when studying the system dynamics [15]. But besides explicitness or implicitness, there is a plethora of properties that can apply to an MMPS system. Knowing whether a property applies to a given system allows for inference about dynamic behaviour of the system. In this section, the properties of explicitness, implicitness, autonomy, time-invariance, monotonicity, non-expansiveness, and homogeneity will be elaborated on. The focus of this chapter lies on the analysis of implicit MMPS systems

3-1-1 Explicit MMPS Systems

Firstly, the concepts of time-invariance, homogeneity, monotonicity, and non-expansiveness will be presented for explicit MMPS systems. For now, systems that are *autonomous* and *explicit* are considered. In a system that is explicit, the current state does not depend on itself, only on previous states, so the current state can be explicitly computed, making an explicit system always solvable [15]. In an autonomous system, the state does not depend on any external (control)inputs, or disturbances, but rather on previous states. For explicit MMPS systems, the concepts of homogeneity, monotonicity, and non-expansiveness will be defined. A global, vector-based definition is given here, to define a homogeneous, monotone, and non-expansive explicit MMPS system.

Definition 3-1.1. *[16] (Homogeneous, monotone, and non-expansive system) Consider an explicit, autonomous system $x(k) = f(x(k-1))$. The system is considered homogeneous if it holds that;*

$$f(x(k-1) + \alpha\mathbf{1}) = f(x(k-1)) + \alpha\mathbf{1} \tag{3-1}$$

Furthermore, the system is considered monotone if it holds that;

$$\textit{if } x \leq y \textit{ then } f(x) \leq f(y) \tag{3-2}$$

Lastly, the system is non-expansive in the l-norm if it holds that;

$$\|f(x) - f(y)\|_l \leq \|x - y\|_l \tag{3-3}$$

In other words, homogeneity of a system means that adding jump $\alpha\mathbf{1}$, so, adding a vector with all entries of value α to to the state of the system, yields a jump of the same $\alpha\mathbf{1}$ in the output of the system. A system that is homogeneous, monotonic, and non-expansive, is called a *topical MMPS system*. Furthermore, the definition of time-invariance for a topical MMPS system can be introduced.

Definition 3-1.2. *[16] (Time invariance) A system $x(k) = f(x(k-1))$, where x is a time signal, is called time-invariant if for any $\tau \in \mathbb{R}$ it holds that*

$$x(k) + \tau\mathbf{1} = f(x(k-1) + \tau\mathbf{1}) \tag{3-4}$$

From this definition, it can be deduced that an MMPS system is only time invariant if and only if the system is homogeneous. The reverse is true as well, homogeneity can be deduced from knowing an MMPS system is time-invariant.

3-1-2 Implicit MMPS Systems

In the previous section, some properties of explicit MMPS systems were elaborated on. The set of implicit systems can be seen as a superset of explicit systems, because all statements that hold for implicit MMPS systems, also hold for explicit MMPS systems, but not necessarily the other way around. Mathematically, the definitions of explicit, and implicit MMPS systems seem very similar, the only difference being the dependence of the state on itself, for implicit systems. This slight difference does however impact the ability to analyze the systems' properties significantly, which will be thoroughly explained throughout this section. The vector-valued expression of an implicit MMPS system is defined in subsection 2-3-1. An alternative representation, in the form of a state-space matrix equation, will be presented in the next section, which will form the basis for further system analysis.

Canonical Form

The canonical form of implicit MMPS systems can be given by a matrix-based state-space equation, with matrices A, B, C and D, see the formal definition below;

Definition 3-1.3. *[15] (Implicit disjunctive ABCD canonical form) Consider the following system;*

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k))) \tag{3-5}$$

Here, $A \in \mathbb{R}_{\varepsilon}^{n \times m}, B \in \mathbb{R}_{\top}^{m \times p}, C, D \in \mathbb{R}^{p \times n}$, *and* $x \in \mathcal{R}^n, k \in \mathbb{Z}^+$ *This is an implicit MMPS system in the disjunctive ABCD canonical form.*

As the definition suggests, this is the disjunctive ABCD canonical form. Its conjunctive counterpart also exists, and has the operations of maximization and minimization in reversed order.

Definition 3-1.4. *[15] (Implicit conjunctive ABCD canonical form) Consider the following system;*

$$x(k) = A_2 \otimes' (B_2 \otimes (C_2 \cdot x(k-1) + D_2 \cdot x(k))) \tag{3-6}$$

Here, $A_2 \in \mathbb{R}_{\top}^{n \times m}, B_2 \in \mathbb{R}_{\varepsilon}^{m \times p}$ *and* $C_2, D_2 \in \mathbb{R}^{p \times n}$. *This system is an implicit MMPS system in the conjunctive ABCD canonical form.*

So, the canonical form is not unique. With the different order of the operations, which cause the existence of two forms, the same system is represented, just in slightly different formats. Therefore, for simplicity, the disjunctive convention is used throughout this thesis. Furthermore, observe that when matrix $D = 0$, an explicit MMPS system is obtained. The structure of the matrices in the canonical form can be defined. It is possible for the system to both include temporal, and quantitative states. The canonical form of the system containing both temporal, and quantity states is given by the following definition;

Definition 3-1.5. *[15] (Autonomous implicit MMPS system) An autonomous implicit MMPS system, containing both temporal states and quantity states can be written as;*

$$\begin{bmatrix} x_\mathrm{t}(k) \\ x_\mathrm{q}(k) \end{bmatrix} = \underbrace{\begin{bmatrix} A_\mathrm{t} & \varepsilon \\ \varepsilon & A_\mathrm{q} \end{bmatrix}}_{A} \otimes \left(\underbrace{\begin{bmatrix} B_\mathrm{t} & \top \\ \top & B_\mathrm{q} \end{bmatrix}}_{B} \otimes' \left(\underbrace{\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_\mathrm{t}(k-1) \\ x_\mathrm{q}(k-1) \end{bmatrix} + \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \cdot \begin{bmatrix} x_\mathrm{t}(k) \\ x_\mathrm{q}(k) \end{bmatrix} \right) \right) \tag{3-7}$$

Here, $x_\mathrm{t} \in \mathcal{R}^{n_\mathrm{t}}, x_\mathrm{q} \in \mathcal{R}^{n_\mathrm{q}}, A_\mathrm{t} \in \mathbb{R}_{\varepsilon}^{n_\mathrm{t} \times m_\mathrm{t}}, A_\mathrm{q} \in \mathbb{R}_{\varepsilon}^{n_\mathrm{q} \times m_\mathrm{q}}, B_\mathrm{t} \in \mathbb{R}_{\top}^{m_\mathrm{t} \times p_\mathrm{t}}, B_\mathrm{q} \in \mathbb{R}_{\top}^{m_\mathrm{q} \times p_\mathrm{q}}, C_{11}, D_{11} \in \mathbb{R}^{p_\mathrm{t} \times n_\mathrm{t}} C_{12}, D_{12} \in \mathbb{R}^{p_\mathrm{t} \times n_\mathrm{q}}, C_{21}, D_{21} \in \mathbb{R}^{p_\mathrm{q} \times n_\mathrm{t}}$, *and* $C_{22}, D_{22} \in \mathbb{R}^{p_q \times n_q}$. *Also, analogous to explicit MMPS systems, the subscript t is associated with temporal states, and the subscript q is associated with quantity states. The matrices denoted by* ε *and* $\top$ *represent matrices of the appropriate sizes in the system matrices where all entries are equal to* ε *and* $\top$ *respectively.*

An alternative, but equivalent notation will be proposed to introduce some overview. Let us represent the implicit MMPS system in an extended state notation;

Definition 3-1.6. *[15] (Extended state MMPS system) An MMPS system can be represented in the following extended state form;*

$$\begin{aligned} x(k) &= A \otimes y(k) \\ y(k) &= B \otimes' z(k) \\ z(k) &= C \cdot x(k-1) + D \cdot x(k) \end{aligned} \tag{3-8}$$

Note that, in the case of the matrix $D = 0$, the extended state MMPS system represents an explicit MMPS system rather than an implicit one. Showcasing how explicit MMPS systems are a subset of implicit MMPS systems.

Time-Invariance of Implicit MMPS Systems

The concept of time-invariance of MMPS systems was briefly introduced for explicit MMPS systems in 3-1.2. It can be proven that an MMPS system that is time-invariant, is also partly additive homogeneous [15]. Considering the MMPS system described by the vector-valued function $x(k) = f(p(k))$ as per 2-22, the property of partial additive homogeneity can be given by the following definition;

Definition 3-1.7. *[18] (Partly additive homogeneous system) Consider an MMPS system with time signal p_t and quantity signal p_q such that the system is given by 2-28. The MMPS system is partly additive homogeneous if;*

$$\begin{aligned} f_t(p_t + \lambda p_q) &= f_t(p_t, p_q) + \lambda \\ f_q(p_t + \lambda, p_q) &= f_q(p_t, p_q) + \lambda \end{aligned} \tag{3-9}$$

for any $\lambda \in \mathbb{R}$.

The intuition behind partial additive homogeneity is found in the concept of time-invariance [18]. Time-invariance of a system with only time signals $x_t(k)$ means that shifting the signals p_t in time, i.e. $p_t(k) \to p_t(k) + \tau$ means shifting the states $x_t(k)$ in time as well, i.e. $x_t(k) \to x_t(k) + \tau$. Therefore, the system f_t will be time-invariant if it is additive homogeneous. A system containing quantity signals as well, is time-invariant if it is *partially* additive homogeneous, i.e. both $(x_t(k), x_q(k), p_t(k), p_q(k))$ and $(x_t(k) + \tau, x_q(k), p_t(k) + \tau, p_q(k))$ are valid trajectories of the system. For implicit MMPS systems, a deduction of the property time-invariance can also be done using the ABCD form as per 3-7. In [16], it is proven that an implicit MMPS system is time invariant if the following properties hold;

$$\begin{aligned} \sum_{i \in \overline{n_\mathrm{t}}} \begin{bmatrix} C_{11} & D_{11} \end{bmatrix}_{\ell i} &= 1, \forall \ell \in \overline{p_\mathrm{t}} \\ \sum_{i \in \overline{n_\mathrm{t}}} \begin{bmatrix} C_{21} & D_{21} \end{bmatrix}_{t i} &= 0, \forall t \in \overline{p_\mathrm{q}} \end{aligned} \tag{3-10}$$

The proof of this will be to show that, if the temporal states $x_t(k)$ are shifted by an amount $h\mathbf{1}$, the extended temporal states are shifted by that same amount.

This shift can be implemented into the extended state MMPS system, which yields the following;

$$\begin{aligned} x_{\mathrm{t}}(k) + h\mathbf{1} &= A_{\mathrm{t}} \otimes (y_{\mathrm{t}}(k) + h\mathbf{1}) \\ y_{\mathrm{t}}(k) + h\mathbf{1} &= B_{\mathrm{t}} \otimes' (z_{\mathrm{t}}(k) + h\mathbf{1}) \\ z_{\mathrm{t}}(k) + h\mathbf{1} &= C_{11} \cdot (x_{\mathrm{t}}(k-1) + h\mathbf{1}) + C_{12} \cdot x_{\mathbf{q}}(k-1) + D_{11} \cdot (x_{\mathrm{t}}(k) + h\mathbf{1}) + D_{12} \cdot x_{\mathbf{q}}(k) \end{aligned} \tag{3-11}$$

Since it is known that the following holds;

$$\begin{aligned} x_t(k) + h\mathbf{1} &= A_{\mathrm{t}} \otimes y_{\mathrm{t}}(k) + h\mathbf{1} \\ y_t(k) + h\mathbf{1} &= B_{\mathrm{t}} \otimes' z_{\mathrm{t}}(k) + h\mathbf{1} \end{aligned} \tag{3-12}$$

Only time-invariance for the extended temporal state $z_t(k)$ will need to be proven now. If it holds that;

$$z_{\mathrm{t}}(k) + h\mathbf{1} = C_{11} \cdot x_{\mathrm{t}}(k-1) + C_{12} \cdot x_{\mathrm{q}}(k-1) + D_{11} \cdot x_{\mathrm{t}}(k) + D_{12} \cdot x_{\mathrm{q}}(k) + h\mathbf{1} \tag{3-13}$$

the system is considered time-invariant. This is valid for $C_{11} \cdot h\mathbf{1} + D_{11} \cdot h\mathbf{1} = h\mathbf{1}$. A similar condition can be derived for quantity states, where the following should hold;

$$\begin{aligned} x_q(k) &= A_q \otimes y_q(k) \\ y_q(k) &= B_q \otimes' z_q(k) \\ z_q(k) &= C_{21} \cdot (x_t(k-1) + h\mathbf{1}) + C_{22} \cdot x_q(k-1) + D_{21} \cdot (x_t(k) + h\mathbf{1}) + D_{22} \cdot x_q(k) \end{aligned} \tag{3-14}$$

Time invariance for $z_q(k)$ is proven when it holds that; $C_{21} \cdot h\mathbf{1} + D_{21} \cdot h\mathbf{1} = 0$. Conclusively, time-invariance of an implicit MMPS system true for the following conditions;

$$\sum_{i \in \overline{n_{\mathrm{t}}}} [C_{11} D_{11}]_{\ell i} = 1, \forall \ell \in \overline{p_{\mathrm{t}}} \quad \sum_{i \in \overline{n_{\mathrm{t}}}} [C_{21} D_{21}]_{ti} = 0, \forall t \in \overline{p_{\mathrm{q}}} \tag{3-15}$$

These conditions are equal to the outcome of the proof in 3-10.

The properties of non-expansiveness and monotonicity have not been proven for general implicit MMPS systems, and no conditions for the properties non-expansiveness and monotonicity have been derived in known literature.

3-2 From Implicit to Explicit: Solvability

In this section, the solvability of implicit systems is examined. If it is possible to rewrite an implicit system into an explicit one, it can be guaranteed that there always exists a solution $x(k), k > 0$ for any state $x(k-1)$. Generally, it is possible to obtain an explicit system from an implicit one, but it will become nested, which means, a system within a system, withing a system, andsoforth. The resulting system is obtained by substituting $x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k)))$ into $D \cdot x(k)$, which is still implicit, so again, $A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k)))$ will be substituted the same way. This substitution is recursive, and the resulting system is of the following form [15];

$$\begin{aligned} x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D\ (A \otimes (B \otimes' \cdots (C \cdot x(k-1) + D \\ \cdots (A \otimes (B \otimes' (C \cdot x(k-1)))) \cdots)))) \end{aligned} \tag{3-16}$$

This newly obtained nested explicit MMPS system in ABC form is not necessarily easier to work with compared to its original implicit MMPS system, as the complexity will become very high. A vector-valued implicit MMPS system can be given by the following expression;

$$x(k) = f(x(k-1), x(k)) \tag{3-17}$$

Whereas a vector-valued explicit MMPS system can be given by the following expression;

$$x(k) = g(x(k-1)) \tag{3-18}$$

By successive substitution an explicit function is obtained [15]

$$\begin{aligned} x_i(k) = f_i\left(x(k-1), f_1(x(k-1)), f_2\left(x(k-1), f_1(x(k-1))\right)\right., \\ \left. \ldots, f_{i-1}\left(x(k-1), f_1(x(k-1)), \ldots, f_{i-2}(x(k-1), \ldots,)\right)\right) \end{aligned} \tag{3-19}$$

Determining whether an explicit mapping of an implicit MMPS system exists, i.e. whether it is solvable, can be done using the structure matrices;
Considering the implicit MMPS system as in (3-7), three structure matrices, S_A, S_B and S_D can be defined as follows[15];

$$[S_A]_{i,j} = \begin{cases} 1 \text{ if } [A]_{i,j} \neq \varepsilon \\ 0 \text{ if } [A]_{i,j} = \varepsilon \end{cases} \tag{3-20}$$

$$[S_B]_{i,j} = \begin{cases} 1 \text{ if } [B]_{i,j} \neq \top \\ 0 \text{ if } [B]_{i,j} = \top \end{cases} \tag{3-21}$$

$$[S_D]_{i,j} = \begin{cases} 1 \text{ if } [D]_{i,j} \neq 0 \\ 0 \text{ if } [D]_{i,j} = 0 \end{cases} \tag{3-22}$$

It can be proven that, using these structure matrices, there may exist a permutation matrix $T \in \mathbb{R}^{n \times n}$ such that;

$$F = T \cdot S_A \cdot S_B \cdot S_D \cdot T^{-1} \tag{3-23}$$

where F is a strictly lower-triangular matrix. If this permutation matrix exists such that F is strictly lower-triangular, there exists a unique solution $x(k), k > 0$ for any state $x(k-1)$ for this implicit MMPS system, and the implicit MMPS system is solvable.

3-3 Eigenvalues and Eigenvectors

The topic of eigenvalues and eigenvectors was briefly touched upon in section 2-2-2. In this Section, a deduction of eigenvalues and eigenvectors is given. Algorithms for both (topical) explicit, and implicit MMPS systems will be provided. Let us refer to eigenvalues and eigenvectors of MMPS systems as the *additive* eigenvalues and *additive* eigenvectors. The definition for additive eigenvalues and additive eigenvectors are given by;

Definition 3-3.1. *[14] (Additive eigenvalue, additive eigenvector) A time-invariant MMPS system* $x(k) = f(x(k), x(k-1))$, $\quad x \in \mathcal{R}^n$ *and* $f \quad : \quad \mathcal{R}^n \to \mathcal{R}^n$ *where both quantity states, and temporal states are present, has an additive eigenvector if there exists a real number* $\lambda \in \mathcal{R}$ *and a vector* $v \in \mathbb{R}^n$ *such that;*

$$f(v) = v + \lambda \begin{bmatrix} \mathbf{1}_{n_t}^\intercal & \mathbf{0}_{n_q}^\intercal \end{bmatrix}^\intercal \tag{3-24}$$

Here, n_t *and* n_q *are the number of temporal states, and the number of quantity states, respectively. The scalar* λ *is called the additive eigenvalue, of which* v *is the corresponding additive eigenvector. Furthermore, if* v *is an additive eigenvector,* $v + h[\mathbf{1}_{n_t}^T \quad \mathbf{0}_{n_q}^T]^T$, *with* $h \in \mathbb{R}$, *is also an additive eigenvector.*

From this point , the additive eigenvalue can also be referred to as the growth rate, and the additive eigenvector as the fixed-point.

3-3-1 Eigenvalues of Implicit MMPS Systems

This Section proposes an algorithm to calculate the growth rates and fixed-points for implicit MMPS systems. It is assumed that the implicit MMPS systems are solvable, and time-invariant. So an implicit MMPS system as in (3-7) with time-invariance is considered. Firstly, a few useful properties will be provided as mathematical background. The value of the additive eigenvalue is the rate at which the system grows [16], i.e. a growth rate. An eigenvalue whose existence only depends on on the structure of the matrices A, B, C, and D is called a structural eigenvalue. Any finite numerical changes in the system matrices to not influence the existence of the additive eigenvalue. The following definition about the existence of structural eigenvalues can be proven;

Definition 3-3.2. *[16] (Existence of structural eigenvalue) A topical MMPS system, characterized by matrices A,B and* C, *has a structural additive eigenvalue and additive eigenvector if and only if matrices* A *and* B *are elementary and regular;*

The definition of an elementary MMPS systems is given by;

Definition 3-3.3. *[16] (Elementary MMPS system) An MMPS system is called elementary, if for each* $i \in 1, ..., n$ *and for each* $j \in 1, ..., m$, *at least either one of the two entries* a_{ij}, b_{ji} *is finite, and* $c_{ij} \neq 0$ *if* $a_{ij} = \varepsilon$

Furthermore, the following definition of a regular matrix is used;

Definition 3-3.4. *[15] (Regular matrix) A matrix* $A \in \mathbb{R}^{n \times m}$ *is considered regular if each row of* A *has at least one finite element*

The exact proof of the existence of a structural eigenvalue and eigenvector can be found in [16]. Furthermore, let an MMPS system such as 3-7 have an additive eigenvalue λ, and additive eigenvector $(x_{te}, x_{qe}, y_{te}, y_{qe}, z_{te}, z_{qe})$. Then, the systems satisfies the following [15];

$$\begin{aligned} z_{te} &= C_{11} \cdot (x_{te} - \lambda \mathbf{1}) + C_{12} \cdot x_{qe} + D_{11} \cdot x_{te} + D_{12} \cdot x_{qe} \\ z_{qe} &= C_{21} \cdot (x_{te} - \lambda \mathbf{1}) + C_{22} \cdot x_{qe} + D_{21} \cdot x_{te} + D_{22} \cdot x_{qe} \\ y_{te} &= B_t \otimes' z_{te}, \quad y_{qe} = B_q \otimes' z_{qe} \\ x_{te} &= A_t \otimes y_{te}, \quad x_{qe} = A_q \otimes y_{qe} \end{aligned} \tag{3-25}$$

Before providing the Linear Programming Problem (LPP) algorithm by which the growth rates and fixed-points can be computed, let us introduce the normalized form of the system. The aim of all of finding the normalized form of a system is to find a an expression for the matrices $\tilde{A}_t$, $\tilde{A}_q$, $\tilde{B}_t$ and $\tilde{B}_q$ such that they obtain a specific structure that eventually makes the system have its fixed-point at $(\tilde{x}_{\mathrm{te}}, \tilde{x}_{\mathrm{qe}}, \tilde{y}_{\mathrm{te}}, \tilde{y}_{\mathrm{qe}}, \tilde{z}_{\mathrm{te}}, \tilde{z}_{\mathrm{qe}}) = (\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0})$ and its growth rate is $\tilde{\lambda} = 0$. The derivation of the normalized form can be found in [15]. From this derivation, the following state-space equation emerges;

$$\begin{aligned} \begin{bmatrix} \tilde{x}_{\mathrm{t}}(k) \\ \tilde{x}_{\mathrm{q}}(k) \end{bmatrix} = \underbrace{\begin{bmatrix} \tilde{A}_{\mathrm{t}} & \varepsilon \\ \varepsilon & \tilde{A}_{\mathrm{q}} \end{bmatrix}}_{\tilde{A}} \otimes \Bigg(\underbrace{\begin{bmatrix} \tilde{B}_t & \top \\ \top & \tilde{B}_q \end{bmatrix}}_{\tilde{B}} \otimes' \Bigg(\underbrace{\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}}_{C} \cdot \begin{bmatrix} \tilde{x}_t(k-1) \\ \tilde{x}_q(k-1) \end{bmatrix} \\ + \underbrace{\begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix}}_{D} \cdot \begin{bmatrix} \tilde{x}_t(k) \\ \tilde{x}_q(k) \end{bmatrix} \Bigg) \Bigg) \end{aligned} \tag{3-26}$$

The construction of the system matrices of this system has been done in such a way that the growth rate $\tilde{\lambda}$ is equal to 0, and the fixed-point is equal to $(\tilde{x}_{\mathrm{te}}, \tilde{x}_{\mathrm{qe}}, \tilde{y}_{\mathrm{te}}, \tilde{y}_{\mathrm{qe}}, \tilde{z}_{\mathrm{te}}, \tilde{z}_{\mathrm{qe}}) = (\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0})$. A connection can be drawn between the original MMPS state equations and the normalized MMPS state equations. which can be represented by the following expressions;

$$\begin{aligned} x_{\mathrm{t}}(k) &= \tilde{x}_{\mathrm{t}}(k) + (k\lambda)\mathbf{1} + x_{\mathrm{te}}, \quad x_{\mathrm{q}}(k) = \tilde{x}_{\mathrm{q}}(k) + x_{\mathrm{qe}} \\ y_{\mathrm{t}}(k) &= \tilde{y}_{\mathrm{t}}(k) + (k\lambda)\mathbf{1} + y_{\mathrm{te}}, \quad y_{\mathrm{q}}(k) = \tilde{y}_{\mathrm{q}}(k) + y_{\mathrm{qe}} \\ z_{\mathrm{t}}(k) &= \tilde{z}_{\mathrm{t}}(k) + (k\lambda)\mathbf{1} + z_{\mathrm{te}}, \quad z_{\mathrm{q}}(k) = \tilde{z}_{\mathrm{q}}(k) + z_{\mathrm{qe}} \end{aligned} \tag{3-27}$$

Here, it can be seen that the normalized quantity states are basically the original quantity states, offset by the fixed-point, placing the normalized quantity state fixed-point at 0. It can be deduced that for the normalized system matrices, the following holds[14]; Each row in $\tilde{A}$ has at least one zero element, and each non-zero element is less than zero. Each column in $\tilde{B}$ has at least one zero element, and each non-zero element is larger than zero, this can be represented by the following expressions;

$$\max_{s \in \bar{m}} \left[\tilde{A}\right]_{rs} = 0 \quad \forall r \in \bar{n} \quad \min_{t \in \bar{p}} \left[\tilde{B}\right]_{st} = 0 \quad \forall s \in \bar{m} \tag{3-28}$$

3-3-2 LPP Algorithm

Having introduced the implicit normalized MMPS system, the algorithm for calculating the growth rates and fixed-points will be introduced. It is possible for multiple growth rates to exist, and the set of these is denoted as $\lambda_\theta, \quad \theta \in \{1,\ldots,S\}$. Let $\tilde{A}_\theta, \tilde{B}_\theta$ be the corresponding normalized system matrices as in (3-26). Now, the concept of footprint matrices can be introduced. Footprint matrices are used to define the location of zeros in the normalized system matrices, and they are of use in the construction of the LPP that is used to possibly find the growth rate and corresponding fixed-point [16]. The pair of footprint matrices $(G_{A_\theta}, G_{B_\theta})$ can now be defined as;

$$G_{A_\theta} = \begin{bmatrix} G_{A_{\mathrm{t}\theta}} & 0 \\ 0 & G_{A_{\mathrm{q}\theta}} \end{bmatrix} \quad G_{B_\theta} = \begin{bmatrix} G_{B_{\mathrm{t}\theta}} & 0 \\ 0 & G_{B_{\mathrm{q}\theta}} \end{bmatrix} \tag{3-29}$$

Matrix G_{A_θ} locates the entries that are exactly 0, as those are the maximal values in the matrix $\tilde{A}_\theta$. Any value that is not equal to 0, will be 0 in the footprint matrix. Therefore, the off-diagonal submatrices of the footprint matrix of $\tilde{A}_\theta$ will be 0 by default. Mathematically, this yields the following expressions [14];

$$[G_{A_{\mathrm{t}\theta}}]_{ij} = \begin{cases} 1 \text{ if } \left[\tilde{A}_{\mathrm{t}\theta}\right]_{ij} = 0 \\ 0 \text{ if } \left[\tilde{A}_{\mathrm{t}\theta}\right]_{ij} < 0 \end{cases} \quad \left[G_{A_{\mathrm{q}\theta}\theta}\right]_{rs} = \begin{cases} 1 \text{ if } \left[\tilde{A}_{\mathrm{q}\theta}\right]_{rs} = 0 \\ 0 \text{ if } \left[\tilde{A}_{\mathrm{q}\theta}\right]_{rs} < 0 \end{cases} \tag{3-30}$$

Similarly, for matrix G_{B_θ}, it is the aim to try and locate the values that are exactly 0. However, since every non-zero entry in $\tilde{B}_\theta$ is larger than 0, determining the footprint matrix is a little different than for G_{A_θ}, since now, any value that is larger than 0 in the system matrix, will be 0 in the footprint matrix. This is mathematically described by;

$$[G_{B_{\mathrm{t}\theta}}]_{jl} = \begin{cases} 1 \text{ if } \left[\tilde{B}_{\mathrm{t}\theta}\right]_{jl} = 0 \\ 0 \text{ if } \left[\tilde{B}_{\mathrm{t}\theta}\right]_{jl} > 0 \end{cases} \quad \left[G_{B_{\mathrm{q}\theta}}\right]_{st} = \begin{cases} 1 \text{ if } \left[\tilde{B}_{\mathrm{q}\theta}\right]_{st} = 0 \\ 0 \text{ if } \left[\tilde{B}_{\mathrm{q}\theta}\right]_{st} > 0 \end{cases} \tag{3-31}$$

Let the growth rate be λ, and the fixed-point of the system be given by;

$$v = (x_{\mathrm{te}}, x_{\mathrm{qe}}, y_{\mathrm{te}}, y_{\mathrm{qe}}, w_{\mathrm{te}}, w_{\mathrm{qe}}) \tag{3-32}$$

Define vector $s = \begin{bmatrix} \mathbf{1}_{n_{\mathrm{t}}}^T & \mathbf{0}_{n_{\mathrm{q}}}^T \end{bmatrix}^T$. Using the footprint matrices obtained from the implicit normalized form as per 3-26, and vector s, all possible linear programming problems (LPP) can be generated; $\forall i \in \bar{n}$,$\forall j \in \bar{m}$, $\forall \ell \in \bar{p}$ where finally, the LPP is given as follows [15];

$$\begin{aligned} \min_{x_{\mathrm{e}}, y_{\mathrm{e}}, w_{\mathrm{e}}} \; & \lambda \\ \text{s.t.} \quad & -[s\lambda]_i - [x]_i + [y]_j \leq -[A]_{ij} \quad \text{if } [G_{\mathrm{A}_\theta}]_{ij} = 0 \\ & [s\lambda]_i + [x]_i - [y]_j = [A]_{ij} \quad \text{if } [G_{\mathrm{A}_\theta}]_{ij} = 1 \\ & [y]_j - [d]_\ell \lambda - [w]_\ell \leq [B]_{j\ell} \quad \text{if } [G_{\mathrm{B}_\theta}]_{jl} = 0 \\ & -[y]_j + [d]_\ell \lambda + [w]_l = [B]_{jl} \quad \text{if } [G_{\mathrm{B}_\theta}]_{jl} = 1 \\ & d = D \cdot s, \quad w = (C + D) \cdot x \end{aligned} \tag{3-33}$$

The footprint matrices $(G_{A_\theta}, G_{B_\theta})$, defined in (3-29) are used to determine the number of linear programming problems need to be solved. The total number of possible footprint matrix pairs is $m_{\mathrm{t}}^{n_{\mathrm{t}}} p_{\mathrm{t}}^{m_{\mathrm{t}}} m_{\mathrm{q}}^{n_{\mathrm{q}}} p_{\mathrm{q}}^{m_{\mathrm{q}}}$. Again, the size of each LLP increases quadratically as the system matrices A, B, C and D increases. The computational complexity is drastically reduced by dropping the constraints corresponding to any element of ε and $\top$ in matrices A and B. The exact number of possible footprint matrices is given by[15];

$$\Pi_{i=1}^{n} a_i \cdot \Pi_{j=1}^{m} b_j \tag{3-34}$$

Where $a_i, i \in \bar{n}$ and $b_j, j \in \bar{m}$ are the number of finite entries in A and B respectively. Consider solution $(\lambda^\star, v^\star = \begin{bmatrix} x^{\star\intercal} & y^{\star\intercal} & w^{\star\intercal} \end{bmatrix})$ to the LPP 3-33, by substituting value $\lambda^\star$ into the equations, a set of equality and inequality constraints can be obtained [15];

$$H_{eq} \cdot v = h_{eq}, \quad H_{ineq} \cdot v \leq h_{ineq} \tag{3-35}$$

Matrix H_{eq} is square, and its rank will at most be $n + m + p - 1$, as the fixed-point is shift-invariant in the direction of vector $s = \begin{bmatrix} \mathbf{1}_{n_{\mathrm{t}}}^T & \mathbf{0}_{n_{\mathrm{q}}}^T \end{bmatrix}^T$. The rank deficiency of matrix H_{eq} is equal to the number of direction vectors where the fixed-points are time-invariant [15]. Let $v^\star$ be a solution to 3-33, and let $g_1, g_2, \ldots, g_f$ be a set of direction vectors where

$$g_1 = \begin{bmatrix} s \\ B \otimes' ((C + D) \cdot s) \\ (C + D) \cdot s \end{bmatrix} \tag{3-36}$$

Which is a vector with similar characteristics to vector s, but extended to incorporate 1s for temporal states in y_e and w_e. Please note that this notation of g_1 is slightly different than what was proposed in [15]. The change accommodates for the dimensions of fixed-point v correctly, as opposed to what was proposed in [15]. The set of fixed-points $\mathcal{V}_{\lambda^\star}$ of the system for a growth rate $\lambda^\star$ can be described by;

$$\mathcal{V}_{\lambda^\star} = \{v | H_{ineq} \cdot v \leq h_{ineq}\} \tag{3-37}$$

This set can be parametrically described by the following polyhedron [15];

$$v = v^\star + \sigma_1 g_1 + \sigma_2 g_2 + \cdots + \sigma_f g_f \tag{3-38}$$

where σ_i are scaling factors. Scaling factor σ_1 is unconstrained, the other scaling factors may be constrained, and their range can be found from the inequality constraints. In summary, the set of fixed-points corresponding to growth rate $\lambda^\star$ will be a polyhedron which is unconstrained in the direction of g_1.

3-3-3 Algorithms for Explicit MMPS Systems

Whereas the previous section elaborated on a very general method to determine the growth rates and fixed-points for implicit MMPS systems, there exist algorithms which compute the growht rates and fixed-points of explicit MMPS systems, specifically. Two different methods of calculating the additive eigenvalue and additive eigenvector for topical MMPS systems will be elaborated on. Firstly, the power algorithm, and secondly, the explicit LPP algorithm will be introduced.

Power Algorithm

This algorithm is applicable to systems satisfying the requirements for a topical MMPS system, and the aforementioned requirements needed for the existence of a structural eigenvalue 3-3.2. It is able to find an additive eigenvalue and corresponding additive eigenvector.

Algorithm 3 [16] Power algorithm for topical MMPS systems

1: Take an arbitrary initial vector $x(0) = x_0 \neq \varepsilon\mathbf{1}$, such that, x_0 has at least one finite element
2: Iterate $x(k) = f(x(k-1))$ until there are integers p and q such that $x(p) - x(q) = c\mathbf{1}$, where $p > q >= 0$, and c is a real number
3: Compute eigenvalue λ as $\lambda = \frac{c}{p-q}$
4: Compute the eigenvector v as $v = \frac{1}{p-q}\sum_{j=q}^{p-1} x(j)$

The other proposed algorithm, the LPP algorithm for topical MMPS systems will be elaborated on in the next Section.

Explicit LPP Algorithm

The LPP algorithm described in this section is very similar to the LPP algorithm given by 3-33. It also requires the explicit system to be normalized, in order to obtain all the possible footprint matrix combinations G_{A_θ} and G_{B_θ}. Normalizing the system is done through a similar procedure [16], and all properties that hold for the implicit normalized form, hold for the explicit normalized form as well. The explicit normalized form is simply the implicit normalized form, without matrix D. Let the variables λ and $(x_{\mathrm{t}}, x_{\mathrm{q}}, y_{\mathrm{t}}, y_{\mathrm{q}}, z_{\mathrm{t}}, z_{\mathrm{q}})$ be the to be determined growth rate, and fixed-point, respectively. For each pair of $(G_{A_\theta}, G_{B_\theta})$ we formulate a Linear Programming Problem (LPP) [14];

$$\begin{aligned}
&\min_{\lambda, x_{\mathrm{t}}, x_{\mathrm{q}}, y_{\mathrm{t}}, y_{\mathrm{q}}, z_{\mathrm{t}}, z_{\mathrm{q}}} \quad \lambda \quad \text{s.t.} \\
&-\lambda - [x_{\mathrm{t}}]_i + [y_{\mathrm{t}}]_j \leq -[A_{\mathrm{t}}]_{ij} && \text{if } [G_{A_{\mathrm{t}\theta}}]_{ij} = 0 \\
&\lambda + [x_{\mathrm{t}}]_i - [y_{\mathrm{t}}]_j = [A_{\mathrm{t}}]_{ij} && \text{if } [G_{A_{\mathrm{t}\theta}}]_{ij} = 1 \\
&-[x_{\mathrm{q}}]_r + [y_{\mathrm{q}}]_s \leq -[A_{\mathrm{q}}]_{rs} && \text{if } \left[G_{A_{\mathrm{q}\theta}}\right]_{rs} = 0 \\
&[x_{\mathrm{q}}]_r - [y_{\mathrm{q}}]_s = [A_{\mathrm{q}}]_{rs} && \text{if } \left[G_{A_{\mathrm{q}\theta}}\right]_{rs} = 1 \\
&[y_{\mathrm{t}}]_j - [z_{\mathrm{t}}]_l \leq [B_{\mathrm{t}}]_{jl} && \text{if } [G_{B_{\mathrm{t}\theta}}]_{jl} = 0 \\
&-[y_{\mathrm{t}}]_j + [z_{\mathrm{t}}]_l = [B_{\mathrm{t}}]_{jl} && \text{if } [G_{B_{\mathrm{t}\theta}}]_{jl} = 1 \\
&[y_{\mathrm{q}}]_s - [z_{\mathrm{q}}]_t \leq [B_{\mathrm{q}}]_{st} && \text{if } \left[G_{B_{\mathrm{q}\theta}}\right]_{st} = 0 \\
&-[y_{\mathrm{q}}]_s + [z_{\mathrm{q}}]_t = [B_q]_{st} && \text{if } \left[G_{B_{\mathrm{q}\theta}}\right]_{st} = 1 \\
&z_{\mathrm{t}} = C_{11} \cdot x_{\mathrm{t}} + C_{12} \cdot x_{\mathrm{q}} \\
&z_{\mathrm{q}} = C_{21} \cdot x_{\mathrm{t}} + C_{22} \cdot x_{\mathrm{q}}
\end{aligned} \tag{3-39}$$

Similarly to the LPP for implicit MMPS systems, the size of this LPP will grow quadratically, as the size of the system matrices increases [16]. However, the computational complexity is reduced by any element in A_t and A_q being ε, as the corresponding constraint is always satisfied, and thus, always holds and can be disregarded. Same goes for elements in B_t and B_q being $\top$.

3-4 Steady-State Behaviour

This section focuses on the steady-state behaviour of time-invariant MMPS systems, and the difference in steady-state behaviour for temporal, and quantity states. The theory discussed in this Section holds both for implicit, and explicit systems. Consider the time-invariant MMPS system [18];

$$\begin{aligned} x_{\mathrm{t}}(k) &= f_{\mathrm{t}}\left(p_{\mathrm{t}}(k), p_{\mathrm{q}}(k)\right) \\ x_{\mathrm{q}}(k) &= f_{\mathrm{q}}\left(p_{\mathrm{t}}(k), p_{\mathrm{q}}(k)\right) \end{aligned} \tag{3-40}$$

Due to the difference in nature between the two types of states, their steady-state behaviour will be different. Temporal states will be nondecreasing, and therefore will not reach an equilibrium. Therefore, steady-state behaviour will be considered, and stationary regimes will be studied, which refer to the growth rate of x_t becoming constant. For (x_t, p_t) a steady-state is reached if, for a certain k_{ss}, the growth of x_t and p_t has become constant;

$$p_{\mathrm{t}}(k) = p_{\mathrm{t}}(k-1) + \tau_{\mathrm{t},ss} \quad \forall k \geq k_{ss} \tag{3-41}$$

For quantity states, an equilibrium can be reached, and steady-state behaviour refers to p_q becoming constant;

$$p_{\mathrm{q}}(k) = p_{\mathrm{q}}(k-1) \quad \forall k \geq k_{ss} \tag{3-42}$$

From these results, the steady-state conditions are obtained;

$$\begin{bmatrix} p_{\mathrm{t}}(k) \\ p_{\mathrm{q}}(k) \end{bmatrix} = \begin{bmatrix} p_{\mathrm{ss,t}} + k\tau_{\mathrm{ss,t}} \\ p_{\mathrm{ss,q}} \end{bmatrix} \tag{3-43}$$

for $k \geq k_{ss}$.

3-5 Bounded-Buffer Stability of MMPS Systems

This section elaborates on the concepts of stability of explicit, and implicit MMPS systems. Generally it can be stated that, a DES is stable when all the temporal states of the system have the same growth rate, and the quantity states have a growth rate of zero. This can be translated into the concept of max-plus bounded-buffer stability [15],i.e. the buffer of the system stays bounded, and the quantity states do not grow with every event cycle "k". From this point on, max-plus bounded-buffer stability will be referred to as bounded-buffer stability. The buffer of an MMPS system is defined as the difference between the time states in each even "k", such that there is no overflow in any state of the system [8]. Bounded-buffer stability can be formally defined as follows;

Definition 3-5.1. *[15] (bounded-buffer stability) An autonomous DES is max-plus bounded-buffer stable if for every initial time state, $x_0 \in \mathbb{R}$, a bound $M(x_0) \in \mathbb{R}$ exists such that the states are bounded in Hilbert's projective norm; $\|x(k)\|_{\mathbb{P}} \leq M(x_0) \quad \forall k \in \mathbb{Z}^+$*

Hilbert's projective norm can be defined as follows;

Definition 3-5.2. *[9] (Hilbert's projective norm) The Hilbert projective norm of a vector $x \in \mathbb{R}^n$ in max-plus algebra is defined as;*

$$||x||_{\mathbb{P}} = \max_{i \in \bar{n}} x_i - max_{i \in \bar{n}} x_j \tag{3-44}$$

Let us first introduce the linear mappings of explicit- and implicit MMPS systems in conventional algebra, wherefrom the conditions that need to be satisfied for the system to be bounded-buffer stable will be determined.

3-5-1 Linear Mappings in Conventional Algebra

As was mentioned in the previous section, it is possible to compute a linear mapping of both implicit, and explicit MMPS systems in conventional algebra. This mapping is valid within the to be specified polyhedron Ω_θ. Firstly, the definition of the conventional linear mapping of explicit MMPS systems is given.

Definition 3-5.3. *[14](Conventional algebra notation of explicit MMPS systems) Any normalized explicit MMPS system can be rewritten as a linear system in conventional algebra for all $\tilde{x}_\theta(k) \in \Omega_\theta, k \in \mathbb{Z}^+$ in the following way [14];*

$$\begin{aligned} \tilde{x}_\theta(k) &= M_\theta \cdot \tilde{x}_\theta(k-1) \\ M_\theta &= G_{A_\theta} \cdot G_{B_\theta} \cdot C \end{aligned} \tag{3-45}$$

For implicit MMPS systems, the linear mapping is slightly more complex, as it is not guaranteed to exist for implicit MMPS systems, as they might be unsolvable [15];

Definition 3-5.4. *[15] (Conventional algebra notation of implicit MMPS system) Any normalized implicit MMPS system can be reformulated as a linear system in conventional algebraic notation for all $\tilde{x}_\theta(k) \in \Omega_\theta, k \in \mathbb{Z}^+$ using the following expressions;*

$$\begin{aligned} \tilde{x}_\theta(k) &= M_\theta \cdot \tilde{x}_\theta(k-1) \\ M_\theta &= (I - M_1)^{-1} \cdot M_2 \\ M_1 &= G_{\mathrm{A}_\theta} \cdot G_{\mathrm{B}_\theta} \cdot D \\ M_2 &= G_{\mathrm{A}_\theta} \cdot G_{\mathrm{B}_\theta} \cdot C \end{aligned} \tag{3-46}$$

if the inverse $(I - M_1)^{-1}$ exists

Hereby, the linearized implicit MMPS system is automatically rewritten into an explicit system. It can be proven that, for any implicit MMPS system for which a strictly lower-triangular matrix F exists, the inverse of $(I - M_1)$ exists [15]. For the linear mapping of both implicit, and explicit MMPS systems, it can be proven that the following holds [14], [15];

$$M_\theta \cdot s = s, \quad s = \begin{bmatrix} \mathbf{1}_{n_\mathrm{t}}^\intercal & \mathbf{0}_{n_\mathrm{q}}^\intercal \end{bmatrix}^\intercal \tag{3-47}$$

Furthermore, it can be proven that M_θ has at least one eigenvalue 1 with eigenvector $v_1 = \begin{bmatrix} \mathbf{1}_{n_\mathrm{t}}^\intercal & \mathbf{0}_{n_\mathrm{q}}^\intercal \end{bmatrix}^\intercal$.

3-5-2 Bounded-Buffer Stability of MMPS Systems

Ultimately, the conditions which are to be satisfied for an MMPS system to be bounded-buffer stable are given by [10]; For $\theta \in \{1, \ldots, S\}$, the MMPS system is;

- Max-plus bounded-buffer stable if the system matrix M_θ only has eigenvalues that are less than, or equal to 1. All eigenvalues of value 1 have to have corresponding Jordan blocks of 1×1
- Unstable if either, at least one eigenvalue is greater than 1, or the corresponding Jordan block of an eigenvalue 1 does not have dimension 1×1

3-6 Polyhedron Ω_θ

The aim of this section is to provide the equations describing polyhedron Ω_θ, for both implicit, and explicit MMPS systems. This polyhedron is the region for which the linearization of the MMPS system is valid. The equations describing the normalized form of explicit, and implicit MMPS systems, respectively, are given below [14], [15];

$$\tilde{x}_\theta(k) = \tilde{A}_\theta \otimes \left(\tilde{B}_\theta \otimes' (C \cdot \tilde{x}_\theta(k-1))\right) \tag{3-48}$$

$$\tilde{x}_\theta(k) = \tilde{A}_\theta \otimes \left(\tilde{B}_\theta \otimes' (C \cdot \tilde{x}_\theta(k-1) + D \cdot \tilde{x}_\theta(k))\right) \tag{3-49}$$

Here, $\theta \in \{1 \dots S\}$. Furthermore, let the region Ω_θ be described by the following equations for explicit, and implicit systems, respectively, such that it contain all the vectors $x \in \mathbb{R}^n$ [14], [15];

$$\tilde{A}_\theta \otimes \left(\tilde{B}_\theta \otimes' (C \cdot x)\right) = G_{\mathrm{A}_\theta} \cdot G_{\mathrm{B}_\theta} \cdot C \cdot x \tag{3-50}$$

$$\tilde{A}_\theta \otimes \left(\tilde{B}_\theta \otimes' (C \cdot x + D \cdot w)\right) = G_{\mathrm{A}_\theta} \cdot G_{\mathrm{B}_\theta} \cdot (C \cdot x + D \cdot w) \tag{3-51}$$

These regions Ω_θ associated to equalities 3-50 and 3-51 are polyhedra given by sets of inequalities. Let us first introduce some mathematical properties, the understanding of which is necessary for the sets of inequalities describing the polyhedra. The operations $\boxtimes$ and $vec(\cdot)$ are called the Kronecker product and the vector constructed in row-major order, respectively, and need to be properly introduced [14].

Definition 3-6.1. *[14] (Kronecker product) The Kronecker product of a vector $\mathbf{1}_n$ and a matrix A can represent two different actions depending on the notation and order. $A \boxtimes \mathbf{1}_n$ vertically stacks n copies of each row of matrix A, while $\mathbf{1}_n \boxtimes A$ stacks n copies of the entire matrix A vertically.*

Definition 3-6.2. *[14] (Row-major order of a matrix) The row-major order of matrix $A \in \mathcal{R}^{n\times m}$ is the order of mapping a matrix to a column vector, $vec(A)$. Here, the columns of A are stacked in one column*

$$\mathrm{vec}(A) = \begin{bmatrix} A_1^\intercal & A_2^\intercal \dots & A_n^\intercal \end{bmatrix}^\intercal \tag{3-52}$$

Here, $A_i, i \in \bar{n}$ denotes the row i of matrix A.

Definition 3-6.3. *[15] (Standard basis vector $\mathbf{e}_j$) The standard basis vector $\mathbf{e}_j$ is a row vector, with the j-th component of this row vector equal to 1 and all other components equal to 0, the size of the vector is determined by the context in which it is used.*

Using the aforementioned mathematical properties, the sets of inequalities describing Ω_θ where the linear mappings in conventional algebra are valid can be defined. Firstly introducing the set of inequalities describing the region Ω_θ for explicit systems;

Definition 3-6.4. *[14] (Ω_θ for explicit systems) The polyhedral region Ω_θ associated with 3-50 is given by the following set of inequalities;*

$$\begin{aligned} &H \cdot \tilde{x} \leq h, \quad H = \begin{bmatrix} U \\ -L \end{bmatrix}, \quad h = \begin{bmatrix} \tilde{b} \\ -\tilde{a} \end{bmatrix} \\ &U = ((G_{B_u} \boxtimes \mathbf{1}_p) - (\mathbf{1}_m \boxtimes I_p)) \cdot C, \quad \tilde{b} = \mathrm{vec}(\tilde{B}_u) \\ &L = ((G_{A_u} \boxtimes \mathbf{1}_m) - (\mathbf{1}_n \boxtimes I_m)) \cdot G_{B_\theta} \cdot C, \quad \tilde{a} = \mathrm{vec}(\tilde{A}_u) \end{aligned} \tag{3-53}$$

Here, $x \in \mathbb{R}^n$, $\boxtimes$ is the Kronecker product, and $vec(\cdot)$ is the vector constructed row-major product.

A comparable definition for the polyhedron Ω_θ of implicit MMPS systems can be drawn up, which will only slightly differ from the explicit case given above.
Definition 3-6.5. *[15] (Ω_θ for implicit systems) The region Ω_θ associated with implicit linearized systems such as in (3-46) is a polyhedron that can be described given the following set of inequalities;*

$$\begin{aligned} &H \cdot \tilde{x} \leq h, \quad H = \begin{bmatrix} U \\ -L \end{bmatrix}, \quad h = \begin{bmatrix} \tilde{b} \\ -\tilde{a} \end{bmatrix} \\ &\tilde{b} = \text{vec}\left(\tilde{B}_\theta\right), \quad \tilde{a} = \text{vec}\left(\tilde{A}_\theta\right) \\ &U = ((G_{\text{B}_\theta} \boxtimes \mathbf{1}_p) - (\mathbf{1}_m \boxtimes I_p)) \cdot (C + D \cdot M_\theta) \\ &L = ((G_{\text{A}_\theta} \boxtimes \mathbf{1}_m) - (\mathbf{1}_n \boxtimes I_m)) \cdot G_{\text{B}_\theta} \cdot (C + D \cdot M_\theta) \end{aligned} \tag{3-54}$$

Here, $x \in \mathbb{R}^n$, $\boxtimes$ is the Kronecker product, and vec$(\cdot)$ is the vector constructed row-major product.

As mentioned before, the linearization of both the implicit, and explicit linearized systems are only valid when $\tilde{x}_\theta(k)$ lies in Ω_θ. However, it is not generally guaranteed that for any $\tilde{x}_\theta(k)$ that lies within Ω_θ, $\tilde{x}_\theta(k+1)$ lies in Ω_θ as well. There may exist a subset of Ω_θ, called an invariant set, such that, any state initialized in the invariant set, will not leave the set, and thus, the system is stable within that set, as the states will always be bounded in some way. In the next section, the algorithm used to derive the largest possible invariant set, i.e. the maximal invariant set, will be given.

3-7 Maximal Invariant Sets

As thoroughly discussed in the previous sections, the mapping between the normalized systems 3-48, 3-49, and the linearized systems 3-45, 3-46 are valid for $\tilde{x}_\theta \in \Omega_\theta$, defined by $\Omega_\theta := H \cdot \tilde{x} \leq h$ [14], [15]. In this section, an approximation of the largest invariant subset of Ω_θ will be given, such that, states of the stable equivalent linear systems 3-48 and 3-49 initialized in this region will stay there [15]. In order to find this set, the definition of a precursor set to a set $\mathcal{X}$ should be given;
Definition 3-7.1. *[2] (Precursor set) For autonomous systems such as 3-46, and 3-45 the precursor set to a set $\mathcal{X}$ is denoted as;*

$$\text{Pre}(\mathcal{X}) = \{x \in \mathbb{R}^n : M_\theta \cdot x \in \mathcal{X}\} \tag{3-55}$$

Pre$(\mathcal{X})$ *is the set of states that evolve into the target set $\mathcal{X}$ in one event-step.*

Furthermore, the definition of a positive invariant set, such as Ω_θ is given by;
Definition 3-7.2. *[2] (Positive invariant set $\mathcal{O}$) A set $\mathcal{O} \subseteq \Omega_\theta$ is considered to be a positive invariant set for the linearized system 3-46 if;*

$$\tilde{x}_\theta(0) \in \mathcal{O} \Longrightarrow \tilde{x}_\theta(k) \in \mathcal{O}, \forall k > 0 \tag{3-56}$$

Next, the definition of when an invariant set is maximal, is given by;
Definition 3-7.3. *[2] (Maximal invariant set $\mathcal{O}_\infty$) The set $\mathcal{O}_\infty \subseteq \Omega_\theta$ is considered the maximal invariant set of the autonomous system as in 3-46, if $\mathcal{O}_\infty$ is invariant, and $\mathcal{O}_\infty$ contains all the invariant sets contained in Ω_θ*

Lastly, the requirement for a set $\mathcal{O}$ to be positive invariant, is given in the following theorem;
Theorem 3-7.1. *[2] (Positive invariance of $\mathcal{O}$) A set $\mathcal{O}$ is considered a positive invariant set for the autonomous system as in 3-46, if and only if;*

$$\mathcal{O} \subseteq \text{Pre}(\mathcal{O}) \tag{3-57}$$

This condition is equivalent to;

$$\text{Pre}(\mathcal{O}) \cap \mathcal{O} = \mathcal{O} \tag{3-58}$$

The precursor set to the set Ω_θ for the autonomous linear systems such as in 3-45 and 3-46 is as follows;

$$\text{Pre}(\Omega_\theta) = \{\tilde{x} \in \mathbb{R}^n : M_\theta \cdot \tilde{x} \in \Omega_\theta\} \tag{3-59}$$

From 3-50 and 3-51 it is known that $\Omega_\theta := H \cdot \tilde{x} \leq h$, therefore the precursor set $\text{Pre}(\Omega_\theta)$ is given by;

$$\text{Pre}(\Omega_\theta) := H \cdot M_\theta \cdot \tilde{x} \leq h \tag{3-60}$$

According to Theorem 3-7.1, in case $\Omega_\theta \subseteq \text{Pre}(\Omega_\theta)$, the set Ω_θ is the maximal invariant set [15]. However, if this is not the case, the following algorithm can be used to iteratively approximate the maximal invariant set $\Omega_{\theta,\infty}$ for the autonomous systems 3-48 and 3-49;

Algorithm 4 [2] Maximal positive invariant set

Input: M_θ, Ω_θ
Output: Ω_∞
$\mathcal{O}_0 \leftarrow \Omega_\theta, k \leftarrow -1$

repeat
$k \leftarrow k + 1$ $\mathcal{O}_{k+1} \leftarrow \text{Pre}\left(\mathcal{O}_k\right) \cap \mathcal{O}_k$

until $\mathcal{O}_{k+1} = \mathcal{O}_k$
$\mathcal{O}_\infty \leftarrow \Omega_k$

Please note that, this obtained maximal invariant set may not be the maximal set for the original MMPS system as per 3-5 [15].

Chapter 4

Solving Solvability

The contributions of this chapter consist of analyzing the existing solvability condition in a graph-theoretic context, and extending the solvability condition to a necessary condition. Furthermore, conditions for when a linear mapping of an implicit MMPS system in conventional algebra exists, are expanded. Let us first present the structure of this chapter. It commences with a thorough analysis of the existing solvability condition, and a graph-theoretic interpretation of this condition is elaborated on in Section 4-1. Observations regarding structure matrix S allow for the emergence of several relevant theorems which form the basis for extending the solvability conditions, which is subsequently discussed in Section 4-2. Four degrees of solvability are described in Section 4-2 as well. The necessary solvability condition is proposed in Section 4-3. Lastly, the condition for which the existence of the inverse of $(I - M_1)$ can be guaranteed, provided the condition is satisfied, is presented in Section 4-4. The concept of solvability in the context of MMPS systems was briefly introduced in Section 3-2. However, this condition is not a necessary one. Solvability of MMPS systems is an important system property, as a solvable system is known to have a solution for any given initial condition. This solution does not necessarily have to be unique, however, it does have to exist.
This gives rise to introduce the definition used in this thesis of when an MMPS system is considered solvable.

Definition 4-0.1. *(Solvability of an MMPS system) An MMPS system is solvable if and only if, there always exists a solution $x(k), k > 0$ for the implicit system [15];*

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k))) \tag{4-1}$$

for any state $x(k-1)$

As opposed to the definition for solvability given in [15], the definition given in 4-0.1 does not require the solution to be unique, but merely to exist. This allows for parametric solutions as well. The distinction between the system having a unique solution, and having a solution, albeit not a unique one, will be elaborated on further in this chapter. The existing theorem for proving solvability, which is identical to the theory provided in 3-2, is taken from [15], and can formally be defined as;

Theorem 4-0.1. *[15] (Solvability of Max-Min-Plus-Scaling systems) If a permutation matrix $T \in \mathbb{R}^{n\times n}$ exists, such that $F \in \mathbb{R}^{n\times n}$ is a strictly lower-triangular matrix in the following context;*

$$F = T \cdot S_A \cdot S_B \cdot S_D \cdot T^{-1} \tag{4-2}$$

Then there always exists a unique solution $x(k), k > 0$ for any state $x(k-1)$ for this implicit system.

Here, the structure matrices are defined as; $S_A \in \mathbb{R}^{n\times m}$, $S_B \in \mathbb{R}^{m\times p}$ and $S_D \in \mathbb{R}^{p\times n}$.

4-1 Graph-Theoretic Interpretation of Solvability

Within the theorem proposed in 4-2, for the purpose of simplicity, the matrix S can be defined as follows;

$$S = S_A \cdot S_B \cdot S_D \tag{4-3}$$

The matrix S as defined in 4-3 can be represented as a graph [9]. Section 2-2 provides a brief overview of relevant concepts in spectral theory useful in this Section. Consider matrix F as per 4-2, due to matrix T being merely a permutation matrix, only permuting the rows and columns of the matrix S, the graph corresponding to matrix S must be the same as the graph corresponding to matrix F.
Matrices F and S are both defined within conventional algebra, and the notion of graphs as referred to in [9] defines graphs as precedence graphs to max-plus algebraic matrices. Therefore, the max-plus equivalent matrix of matrix S will be defined as matrix $S_\otimes$.

$$[S_\otimes]_{i,j} = \begin{cases} [S]_{i,j} \text{ if } [S]_{i,j} \neq 0 \\ \varepsilon \text{ if } [S]_{i,j} = 0 \end{cases} \tag{4-4}$$

The only difference between matrices S and $S_\otimes$ is that all entries that are 0 in matrix S, are ε in matrix $S_\otimes$. Any square matrix A in $\mathbb{R}_\varepsilon$ can be associated with a graph called the *communication graph* of matrix A [9]. In this section, all graphs are *weighted* and *directed.* However, the graphs corresponding to any matrix $S_\otimes$ as defined in 4-4 will not be referred to as *communication graphs* but rather as *interconnection graphs.* Matrix $S_\otimes$ and its entries carry information about how states in an MMPS system implicitly depend on each other, i.e how they are interconnected with each other. The value of the non$-\varepsilon$ entries in matrix $S_\otimes$ are of importance in the further solvability analysis, and will be elaborated on further in this chapter. Within graph theory a distinction is made between graphs containing a *circuit* and *acyclic* graphs. The formal definition of a circuit can be found in Subsection 2-2-1 The following theorem regarding graphs and circuits can be introduced, which is an extension of the existing theory;
Theorem 4-1.1. *(Solvability of MMPS systems using graphs) If the graph of matrix $S_\otimes$ does not contain any circuits, the MMPS system corresponding to this graph is solvable.*

Proof. Given matrix $S_\otimes \in \mathbb{R}_\varepsilon^{n\times n}$ and matrix $F \in \mathbb{R}^{n\times n}$ from the following equation;

$$F = T \cdot S_A \cdot S_B \cdot S_D \cdot T^{-1} \tag{4-5}$$

If the graph corresponding to matrix $S_\otimes$ does not contain any circuit, a permutation matrix T exists that permutes the rows an columns such that S can be rewritten into a strictly lower-triangular F. Because $S_\otimes$, and therefore S is acyclic, there only exist paths in the graph corresponding to $S_\otimes$. Therefore, there exists a topological ordering of the vertices in those paths, as the graph is directed. For every edge (i, j) in the graph, by reordering the edges such that $i < j$ for every edge, entry (i, j) in matrix F will appear below the diagonal by this permutation, yielding a strictly lower-triangular matrix. □

This theorem and proof can be illustrated by the example given below;
Example 4-1.2. *(Graph corresponding to matrix $S_\otimes$ containing a circuit) Suppose the state-space description of an MMPS system yields the following matrix S;*

$$S = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 2 & 1 & 1 \end{pmatrix} \tag{4-6}$$

The equivalent max-plus matrix $S_\otimes$ is given by;

$$S_\otimes = \begin{pmatrix} \varepsilon & 1 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 1 & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 2 & 1 & 1 \end{pmatrix} \tag{4-7}$$

The corresponding interconnection graph is given in Figure 4-1

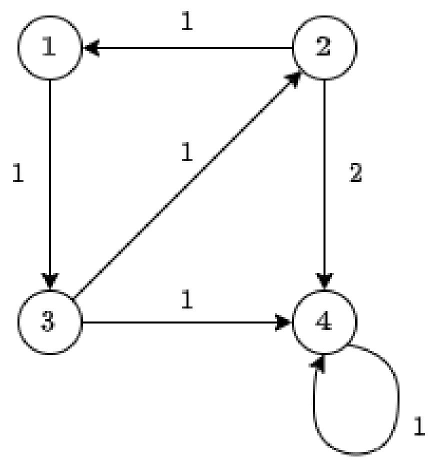


Figure 4-1: Communication graph corresponding to $S_\otimes$

It is immediately visible that there are ***two*** *circuits present in this graph, one from node 1 to node 3 to node 2, and a loop on node 4. Due to the presence of these circuits there does not exist a permutation matrix T which permutes matrix S into a strictly lower-triangular matrix.*

Whereas the previous example showed a system containing a circuit, an example of an acyclic system is given here;

Example 4-1.3. *(Graph corresponding to acyclic matrix $S_\otimes$) Suppose the state-space description of an MMPS system yields the following matrix S;*

$$S = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix} \tag{4-8}$$

The equivalent max-plus matrix $S_\otimes$ is given by;

$$S_\otimes = \begin{pmatrix} \varepsilon & 1 & 1 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 1 & \varepsilon & 1 \\ \varepsilon & 1 & \varepsilon & \varepsilon \end{pmatrix} \tag{4-9}$$

The corresponding interconnection graph is given in Figure 4-2

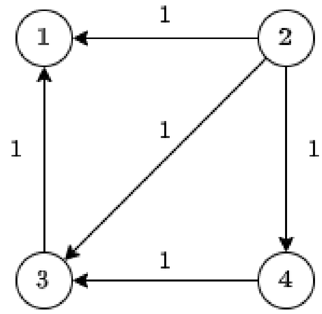


Figure 4-2: Communication graph corresponding to $S_\otimes$

In this graph, no circuits are present. Therefore, there must exist a permutation matrix T which permutes matrix S into a strictly lower-triangular matrix. This permutation matrix T, and subsequent matrix F are given by;

$$T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \quad F = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix} \tag{4-10}$$

Matrices T and F as per 4-10 clearly shows the lower-triangular permutation of F, therefore proving this system is solvable

In Example 4-1.2, it is immediately visible that there are circuits present by looking at the interconnection graph. However, it could be that an interconnection graph is very complex, and the existence of circuits cannot be immediately concluded by simply looking at the graph. Therefore, a new theorem can be introduced that mathematically proves the existence of circuits in an interconnection graph corresponding to matrix $S_\otimes$.

Theorem 4-1.4. *(Existence of a circuit in an interconnection graph) Define matrix $S_\otimes^+ \in \mathbb{R}^{n\times n}$ as follows;*

$$S_\otimes^+ = \bigotimes_{k=1}^{n} S_\otimes^{\otimes k} \tag{4-11}$$

Where;

$$S_\otimes^+ = S_\otimes \oplus S_\otimes^2 \oplus \cdots \oplus S_\otimes^n \tag{4-12}$$

If all entries $[S_\otimes^+]_{i,i}$, so all diagonal entries of $S_\otimes^+$, are equal to ε, no path of any length $l \in \{1, 2, ..., n\}$ exists from any node $i \in 1, 2, ..., n$ to itself, and therefore no circuits appear in the interconnection graph of matrix $S_\otimes$

Proof. Consider the definition of A^+ as in [9], and presented in 2-18. Because of the way matrix $S_\otimes^+$ is calculated, entry $[S_\otimes]_{i,j}$ represents the maximum weight of a path of any length $l \in \{1, 2, ..., n\}$ from node i to node j. If an entry $[S_\otimes]_{i,j}$ is equal to ε no such path exists of any length. It is not necessary to check if a path from a node to itself from length larger than n exists. In case a circuit of any length exists, it must encompass less than n, or exactly n nodes, as it is impossible for a circuit of length $n + 1$ to exist, without already containing a circuit of less than, or exactly n. It is of importance to know whether **any** circuit exist, and which states are part of **any** circuit, its length does not matter. In case all diagonal entries of $S_\otimes^+$ would be ε, no path from any node to itself exists, therefore excluding the possibility that a circuit appears in the communication graph corresponding to $S_\otimes$. □

The following example demonstrates this proposed theorem;

Example 4-1.5. *(Proving the existence of a circuit) Consider the matrix $S_\otimes$ from Example 4-1.2. Applying Theorem 4-1.4, matrix $S_\otimes^+$ corresponding to this matrix $S_\otimes$ is given by;*

$$S_\otimes^+ = \begin{bmatrix} 3 & 4 & 2 & \varepsilon \\ 2 & 3 & 4 & \varepsilon \\ 4 & 2 & 3 & \varepsilon \\ 5 & 5 & 5 & 4 \end{bmatrix} \tag{4-13}$$

It is immediately visible that all entries on the diagonal of $[S_\otimes^+]_{ii}$ are not ε, indicating that the path for all states, at least one path of finite length from a node to itself exists. On the other hand, calculating matrix $S_\otimes^+$ for matrix $S_\otimes$ corresponding to the example given in 4-1.3 yields the following;

$$S_\otimes^+ = \begin{bmatrix} \varepsilon & 3 & 1 & 2 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 2 & \varepsilon & 1 \\ \varepsilon & 1 & \varepsilon & \varepsilon \end{bmatrix} \tag{4-14}$$

This matrix $S_\otimes$ does have ε on its diagonal for all $[S_\otimes^+]_{ii}$, confirming the finding that this matrix is acyclic and therefore, strictly lower-triangular matrix F exists, thus making this system solvable.

Hereby, the solvability condition stemming from the graph-theoretic interpretation of the matrix S is concluded. This proof yields a condition equally strong as the condition already introduced in 3-2. However, it is possible to use this graph-theoretic solvability condition as a starting point to extend the solvability condition and yield a stronger one. The next section is dedicated to extending this condition, and proving validity of the extended condition.

4-2 Extending the Solvability Condition

In this section, the sufficiency of the existing solvability condition is proven. Furthermore, the condition is extended, and four degrees of solvability are proposed. The contents of this section are all extensions of the existing theory, with the exception of some recalled theory that was discussed in previous sections, in which case, a clear reference to the source is provided.

4-2-1 Proving Sufficiency of the Existing Condition

The previous section proved that the absence of circuits in the interconnection graph corresponding to matrix $S_\otimes$ proves the solvability of the corresponding MMPS system. However, systems that do not adhere to this condition may very well still be solvable. Example 4-2.1 shows a very simple MMPS system with three states. it can be shown that, while the matrix $S_\otimes$ corresponding to this system has an interconnection graph containing a circuit, the explicit mapping $x(k) = g(x(k-1))$ of the implicit system $x(k) = f(x(k-1), x(k))$ still exists. This will again prove what was already known, that the existing solvability condition is merely a sufficient condition, not a necessary one.

Example 4-2.1. *(Sufficiency of the existing solvability condition) Given the following vector-valued implicit MMPS system, a solvability analysis can be conducted;*

$$\begin{aligned} x_1(k) &= 2x_2(k) + x_1(k-1) + 4 \\ x_2(k) &= x_3(k) + 2x_3(k-1) \\ x_3(k) &= x_1(k) + 3x_2(k-1) - 1 \end{aligned} \tag{4-15}$$

The ABCD-form of this system can easily be computed, and is given by the following equation;

$$\begin{aligned} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} = \begin{bmatrix} 4 & \varepsilon & \varepsilon \\ \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & -1 \end{bmatrix} \otimes \begin{bmatrix} 0 & \top & \top \\ \top & 0 & \top \\ \top & \top & 0 \end{bmatrix} \otimes' \left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \\ 0 & 3 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \end{bmatrix} \right. \\ \left. + \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \right) \end{aligned} \tag{4-16}$$

From this ABCD-form, the structure matrices S_A,S_B and S_D can be computed, which are given by;

$$S_A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad S_B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad S_D = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \tag{4-17}$$

Using these obtained matrices S_A, S_B and S_D, matrix S and subsequently, matrices $S_\otimes$ and $[S_\otimes^+]$ can be computed.

$$S = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \quad S_\otimes = \begin{bmatrix} \varepsilon & 1 & \varepsilon \\ \varepsilon & \varepsilon & 1 \\ 1 & \varepsilon & \varepsilon \end{bmatrix} \quad S_\otimes^+ = \begin{bmatrix} 3 & 1 & 2 \\ 2 & 3 & 1 \\ 1 & 2 & 3 \end{bmatrix} \tag{4-18}$$

The interconnection graph corresponding to matrix $S_\otimes$, visible in Figure 4-3 clearly shows the existence of a circuit in this system.

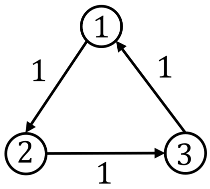


Figure 4-3: Interconnection graph of matrix $S_\otimes$

So the interconnection graph corresponding to matrix $S_\otimes$ does contain a circuit. But simply by analyzing the vector-valued set of equations as given in 4-15 within the context of conventional algebra, by successive substitution, an explicit mapping of this system can be found. The result of this successive substitution is given by the following set of equations;

$$\begin{aligned} x_1(k) &= -6x_2(k-1) - 4x_3(k-1) - 2 \\ x_2(k) &= -3x_2(k-1) - 2x_3(k-1) - 3 \\ x_3(k) &= -3x_2(k-1) - 4x_3(k-1) - 3 \end{aligned} \tag{4-19}$$

Example 4-2.1 shows that MMPS systems can still be solvable, even though the interconnection graph is not acyclic. In this specific example, successive substitution was used to actually show what this explicit mapping is, and that it therefore exists. However, the greater purpose of proving solvability in MMPS systems is not necessarily to rewrite an implicit system into an explicit one, but to determine **whether it is possible** to rewrite an implicit system into an explicit one. As Section 3-2 describes, subsequent substitution of implicit states may result in a nested system, yielding very large state-space equations, which is undesirable. Even though the resulting nested expression may be explicit, merely knowing whether an explicit mapping exists is enough. At each k, the configuration of the system can be characterized by a system of linear equations. Considering there are max and min terms present in the state equations, at each k, only one of the affine terms in these min or max expressions can be minimal or maximal. The set of affine equations formed by the expressions that define the state evolution at k is referred to as the *mode* of the system. Only one specific combination of entries of matrices A and B will be relevant at each k, combined with the scaled states $z(k) = C \cdot x(k-1) + D \cdot x(k)$.

Proposition 4-2.1. *(Mode of a Max-Min-Plus-Scaling system) Suppose we have an MMPS system of the following form;*

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k))) \tag{4-20}$$

The mode of the system at k can be characterized by the following equation;

$$x_i(k) = [A]_{i,j} + [B]_{j,k} + z_k(k) \tag{4-21}$$

Where $x_i(k) \in [x_1(k), x_2(k), \ldots, x_n(k)]$, $z_k(k) \in [z_1(k), z_2(k), \ldots, z_p(k)]$, $[A]_{i,j}$ is entry $\{i,j\}$ of matrix A, similarly, $[B]_{j,k}$ is entry $\{j,k\}$ of matrix B.

Proof. Each affine term that is present in the expression of state $x_i(k)$, is an addition of some term $z_k(k)$, and some entry $[A]_{i,j}$, and some entry $[B]_{j,k}$.

Any affine term in $x_i(k)$ is an expression of some $z_k(k)$, which appears in the expression of $x_i(k)$ by being fed through matrices A and B. From the definitions of max-plus, and min-plus algebraic matrix operations as per 2-9, it becomes apparent that the corresponding entries of A and B are added to the affine term $z_k(k)$, yielding the proposed expression. □

When a system is in a stable configuration, in the region of a fixed-point, the successive modes are the same. This concept is introduced in [4] as the dominant equations of a system. Each possible mode corresponds with one entry per row of matrices A and B being "active", i.e. contributing to the state evolution. Therefore, the number of possible modes can be given by the same equation that is used to compute the total number of possible footprint matrices[15];

$$\prod_{i=1}^{n} a_i \cdot \prod_{j=1}^{m} b_j \tag{4-22}$$

The concept of different modes of an MMPS systemis illustrated in the following example;
Example 4-2.2. *(Modes of an MMPS system) Consider the following MMPS system equations;*

$$\begin{aligned} x_1(k) &= \max(x_2(k) + 4, 3x_1(k-1) + 2x_3(k-1)) \\ x_2(k) &= 5x_1(k) - 2x_2(k-1) + 1 \\ x_3(k) &= \min(4x_1(k-1) - 2, x_3(k-1) + x_2(k-1)) \end{aligned} \tag{4-23}$$

This MMPS system can be viewed as a set of general systems of linear equations, considering possible combinations of affine terms. When simulating an MMPS system, at each event step k, one of multiple possible modes modes may be active. In this system, the following modes exist;

$$\begin{aligned} x_1(k) &= x_2(k) + 4 \\ x_2(k) &= 5x_1(k) - 2x_2(k-1) + 1 \\ x_3(k) &= 4x_1(k-1) - 2 \end{aligned} \tag{4-24}$$

$$\begin{aligned} x_1(k) &= x_2(k) + 4 \\ x_2(k) &= 5x_1(k) - 2x_2(k-1) + 1 \\ x_3(k) &= x_3(k-1) + x_2(k-1)) \end{aligned} \tag{4-25}$$

$$\begin{aligned} x_1(k) &= 3x_1(k-1) + 2x_3(k-1)) \\ x_2(k) &= 5x_1(k) - 2x_2(k-1) + 1 \\ x_3(k) &= 4x_1(k-1) - 2 \end{aligned} \tag{4-26}$$

$$\begin{aligned} x_1(k) &= 3x_1(k-1) + 2x_3(k-1)) \\ x_2(k) &= 5x_1(k) - 2x_2(k-1) + 1 \\ x_3(k) &= x_3(k-1) + x_2(k-1)) \end{aligned} \tag{4-27}$$

As shown above, there are four modes that could arise in simulation. When analyzing this system, the matrices S, $S_\otimes$ and $S_\otimes^+$ can be computed.

$$S = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad S_\otimes = \begin{bmatrix} \varepsilon & 1 & \varepsilon \\ 1 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{bmatrix} \quad S_\otimes^+ = \begin{bmatrix} 2 & 3 & \varepsilon \\ 3 & 2 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{bmatrix} \tag{4-28}$$

Matrix $S_\otimes^+$ reveals that a circuit containing nodes $x_1(k)$ and $x_2(k)$ exists.

However, when examining the four possible modes, only two out of four modes exhibit contain a circuit. This means that only two out of the four modes require further inspection as to analyze their solvability. In order to proceed the deduction of the extended solvability theorem, let us reintroduce the extended state MMPS system. The extended state MMPS system can be defined by;

Definition 4-2.1. *[15] (Extended state MMPS system) An MMPS system can be represented in the following extended state form;*

$$\begin{aligned} x(k) &= A \otimes y(k) \\ y(k) &= B \otimes' z(k) \\ z(k) &= C \cdot x(k-1) + D \cdot x(k) \end{aligned} \tag{4-29}$$

This definition is identical to the one given in 3-8. Let us now introduce the necessary, but sometimes not obvious facts required to structure the extended solvability theorem and its proof are introduced below;

- Solvability of an MMPS system can only be violated by the states that are included in any circuit in the interconnection graph of $S_\otimes$
- The number of non-ε entries in row i of matrix A defines how many entries of vector $y(k)$ are included in the expression of state $x_i(k)$.
- The number of non-$\top$ entries in row i of matrix B defines how many entries of vector $z(k)$ are included in the expression of $y_i(k)$
- The number of non-0 entries in row i of matrix D defines how many states of state vector $x(k)$ are included in the expression of $z_i(k)$
- If $S_{i,j}$ of matrix S is not equal to 0, the expression of state $x_i(k)$ contains state $x_j(k)$ at least once.
- The entry $S_{i,j}$ of matrix $S \in \mathbb{R}^{n \times n}$ is a non-negative integer. This is due to matrix S being constructed from conventional multiplication of matrices S_A, S_B and S_D, whose entries are all of value 0 or 1.

These statements, some of which may seem obvious, aid tremendously in the eventual extended solvability proof. It can be concluded that an MMPS system could be unsolvable if a circuit is present. However, as shown in Example 4-2.2, not all modes necessarily contain a circuit, as not all entries corresponding to a circuit may be active at the same time, essentially breaking the circuit. It is known that if no circuit is present, the system is solvable. Therefore, it is sufficient to verify whether the modes that do contain a circuit are solvable. Even more so, since only the states included in any circuit can violate the solvability of the whole system, it is also sufficient to only analyze the solvability of the states included in any circuit. The subset of states that are included in any circuit will be referred to as the circuit subsystem;

Definition 4-2.2. *(Circuit subsystem) Consider an MMPS system containing at least one circuit. The set of states included in any circuit in the system is called the circuit subsystem and is given by $x_c(k)$; The state-space equation of the circuit subsystem just excludes all entries corresponding to states not in any circuit, and is given by;*

$$x_c(k) = A_c \otimes (B_c \otimes' (C_c \cdot x_c(k-1) + D_c \cdot x_c(k))) \tag{4-30}$$

Where A_c, B_c, C_c and D_c are matrices consisting of a selection of the entries of matrices A, B, C and D corresponding to the states included in a circuit.

Let us demonstrate how matrices A_c, B_c,C_c and D_c are constructed, assuming it is known which states are included in any circuit;

Example 4-2.3. *(Constructing matrices A_c, B_c, C_c and D_c) Consider the MMPS system as per Example 4-2.2, where states $x_1(k)$ and $x_2(k)$ are included in a circuit. The ABCD matrices of this system are given by;*

$$A = \begin{bmatrix} 4 & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top \\ \top & \top & 0 & \top & \top \\ \top & \top & \top & -2 & 0 \end{bmatrix}$$
$$C = \begin{bmatrix} 0 & 0 & 0 \\ 3 & 0 & 2 \\ 0 & -2 & 0 \\ 4 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}, D = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 5 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (4\text{-}31)$$

Since states $x_1(k)$ and $x_2(k)$ are the only states included in any cycle, the entries of A, B, C and D corresponding to these states will be used to form A_c,B_c,C_c and D_c. These entries are coloured light blue in the matrices A,B,C and D above. The resulting matrices A_c, B_c, C_c and D_c can be given by;

$$A_c = \begin{bmatrix} 4 & 0 & \varepsilon \\ \varepsilon & \varepsilon & 0 \end{bmatrix}, B_c = \begin{bmatrix} 0 & \top & \top \\ \top & 0 & \top \\ \top & \top & 0 \end{bmatrix}$$
$$C_c = \begin{bmatrix} 0 & 0 \\ 3 & 0 \\ 0 & -2 \end{bmatrix}, D_c = \begin{bmatrix} 0 & 1 \\ 5 & 0 \\ 0 & 0 \end{bmatrix} \quad (4\text{-}32)$$

Using a similar approach, matrices S_c, $S_{\otimes,c}$ and $S^+_{\otimes,c}$, which are matrices consisting of the entries of S, $S_\otimes$ and $S^+_\otimes$ corresponding to the circuit subsystem, can be computed as well. It is worth mentioning that the diagonal entries $[S^+_{\otimes,c}]_{i,i}$ of $S^+_{\otimes,c}$ consists of only non-ε values, as only states part of any circuit are included in $S_{\otimes,c}$. Since only the modes containing a circuit will need further analysis, the number of modes that is to be analyzed will be less than, or equal to the number of existing modes. A mode containing a circuit will be referred to as a *circuit mode*.

4-2-2 Decoding matrix S

This subsection provides a thorough analysis of structure matrix S, and proposes two theorems regarding the findings of this analysis. Let us first define the concept of a minimal realization of an MMPS system as follows;

Definition 4-2.3. *(Minimal realization of an MMPS system) An MMPS system is a minimal realization if the following conditions hold;*

1. *Every mode of the system can logically be reached, i.e. no mode of the system is redundant. For example, if a state is described by the following equation;*

$$x_1(k) = \max(x_2(k) + 6, x_2(k) + 5) \tag{4-33}$$

The first term, $x_2(k) + 6$ will always be maximal, making the second affine term redundant, and the modes containing this term cannot logically be reached. A system containing such a mode is not a minimal realization

2. *Every term $z_i(k)$ must appear in the expression of any state $x_j(k)$. This is equivalent to no row, and no column of matrices A and B being entirely ε or $\top$, respectively.*

This definition is used in the further analysis of structure matrix S. The maximum number of circuit modes can be computed by using the information embedded in the entries of matrix S. Matrix S carries important information regarding the implicit dependency of states on each other, as a non-zero value on entry $S_{i,j}$ signifies an implicit dependency of state $x_i(k)$ on state $x_j(k)$. The value of this entry is of importance as well. The following theorem regarding the value of entry $S_{i,j}$ will aid in deriving a formula to compute the maximum number of circuit modes in an implicit MMPS system;

Theorem 4-2.4. *(Interpretation of entry $S_{i,j}$) The value of entry $S_{i,j}$ represents the number of affine terms in the expression of state $x_i(k)$ in which state $x_j(k)$ appears.*

Proof. This line of reasoning in this proof originates in the ABCD matrices, and specifically in the extended state-space description 4-29. The facts stated below, in that order, provide a conclusive deduction that proves Theorem 4-2.4.

- In case all modes of the systems can logically be reached, i.e. no affine term is redundant, and all terms of $z(k)$ appear in the final state-space equations, the system is a minimal realization.
- If a state $x_j(k)$ is included in the linear term $z_i(k)$, entry $[D]_{i,j} \neq 0$
- If linear term $z_i(k)$ is included in the expression of $y_p(k)$, entry $[B]_{p,i} \neq \top$
- If expression $y_p(k)$ is included in the expression of state $x_q(k)$, entry $[A]_{q,p} \neq \varepsilon$
- For the system to be a minimal realization, every linear term $z_i(k)$ must be included in at least one expression of an entry in $y(k)$, so there exists an entry $[B]_{p,i} \neq \top$. Equivalently, no column of B should be all $\top$ if the system is a minimal realization.
- For the system to be a minimal realization, any term $y_p(k)$ must be included in at least one expression of an entry in $x(k)$, so there exists an entry $[A]_{q,p} \neq \varepsilon$. Equivalently, no column of A should be all ε if the system is a minimal realization.

From the facts stated above, it can be concluded that, if state $x_j(k)$ is included in $z_i(k)$, $z_i(k)$ is included in $y_p(k)$, and $y_p(k)$ is included in $x_q(k)$, entries $[D]_{i,j}$,$[B]_{p,i}$ $[A]_{q,p}$ will be non-$0/\top/\varepsilon$, respectively. Therefore, at least one affine term in the expression of $x_q(k)$ implicitly depends on state $x_j(k)$. Assume that the same state $x_j(k)$ is included in $z_s(k)$, $z_s(k)$ is included in $y_t(k)$, and $y_t(k)$ is included in $x_q(k)$, entries $[D]_{s,j}$,$[B]_{t,s}$ $[A]_{q,t}$ will also be non-$0/\top/\varepsilon$, respectively. Basically, two different sequences of entries are defined that represent the implicit dependency of state $x_q(k)$ on state $x_j(k)$ in the ABD matrices. Sequences like this also exist for explicit state mappings with entries of C.

Nonetheless, every entry in matrices B, C, D and A that is non-$0/\top/\varepsilon$ is part of such a sequence, either an implicit one or an explicit one. So besides the entries that are part of a sequence, all entries are $0/\top/\varepsilon$ in a minimal realization. It is known that the aforementioned entries which are non-$0/\top/\varepsilon$, are 1 in the structure matrices, as the structure of the matrices is conserved. The distinction between the matrix operations $\otimes, \otimes'$ and conventional multiplication will give important insights into how the implicit mapping of the MMPS system relates to the conventional multiplication of the corresponding structure matrices; Given

$$\begin{aligned}
[B \otimes' D]_{i,j} &= \bigotimes_{k=1}^{p}{}' \, b_{i,k} \otimes' d_{k,j} \\
&= \min_{k \in p}\{b_{i,k} + d_{k,j}\} \\
[S_B \cdot S_D]_{i,j} &= \sum_{k=1}^{p} S_{B,i,k} \cdot S_{D,k,j}
\end{aligned} \tag{4-34}$$

and

$$\begin{aligned}
[A \otimes (B \otimes' D)]_{i,j} & \bigotimes_{k=1}^{m} a_{i,k} \otimes (B \otimes' D)_{k,j} = \\
&= \max_{k \in m}\{a_{i,k} + (B \otimes' D)_{k,j}\} \\
[S_A \cdot S_{BD}]_{i,j} &= \sum_{k=1}^{m} S_{A,i,k} \cdot S_{BD,k,j}
\end{aligned} \tag{4-35}$$

Given that $A \in \mathbb{R}_{\varepsilon}^{n \times m}, B \in \mathbb{R}_{\top}^{m \times p}, D \in \mathbb{R}^{p \times n}$ and $S_A \in \mathbb{R}^{n \times m}$,$S_B \in \mathbb{R}^{m \times p}$, $S_D \in \mathbb{R}^{p \times n}$, $S_{BD} = S_B \cdot S_D$. So essentially, the max-plus addition is replaced by conventional multiplication, and the maximization/minimization is replaced by conventional addition. This is not necessarily new information, but is an important notion when it comes to relating the value of the entries in S to the state expressions. Because knowing the only non-0 entry sequences that exist are the ones that implicitly relate states to each other, the existence of one sequence $[S_D]_{i,j} = 1, [S_B]_{p,i} = 1, [S_A]_{q,p} = 1$ results in the value of entry $[S]_{q,j} = 1$. In the case that another entry sequence $[S_D]_{s,j} = 1, [S_B]_{t,s} = 1, [S_A]_{q,t} = 1$ exists, the value of $[S]_{q,j} = 2$, andsoforth. Let us illustrate this by representing the implicit dependencies and non-$\varepsilon/\top/0$ entries in A,B and D in a graph like manner as to clarify this concept of sequences, and how they relate to the entries in S;

Example 4-2.5. *(Graph representation of sequences)*
Consider an implicit MMPS system given by the following state-space equations;

$$\begin{aligned}
x_1(k) &= \min(2x_1(k) + 1, x_1(k) + x_2(k)) + 2 \\
x_2(k) &= 3x_2(k) - 1
\end{aligned} \tag{4-36}$$

Matrices A, B, C and D can be given as follows;

$$\begin{aligned}
A &= \begin{bmatrix} 0 & \varepsilon \\ \varepsilon & -1 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 & \top \\ \top & \top & 0 \end{bmatrix} \\
C &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, D = \begin{bmatrix} 2 & 0 \\ 1 & 1 \\ 0 & 3 \end{bmatrix}
\end{aligned} \tag{4-37}$$

The corresponding matrices S_A,S_B, S_D and S are given by;

$$S_A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, S_B = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \ , S_D = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}, S = \begin{bmatrix} 2 & 1 \\ 0 & 1 \end{bmatrix} \tag{4-38}$$

So, according the to be proven theorem, the state $x_1(k)$ contains two affine terms containing $x_1(k)$, and one affine term containing $x_2(k)$, which is true. The point of this example is to illustrate how this entry $[S]_{1,1} = 2$ has come about. In Figure 4-4, a graph is visible, showcasing how state $x_1(k)$ and $x_2(k)$ implicitly depend on themselves, and each other.

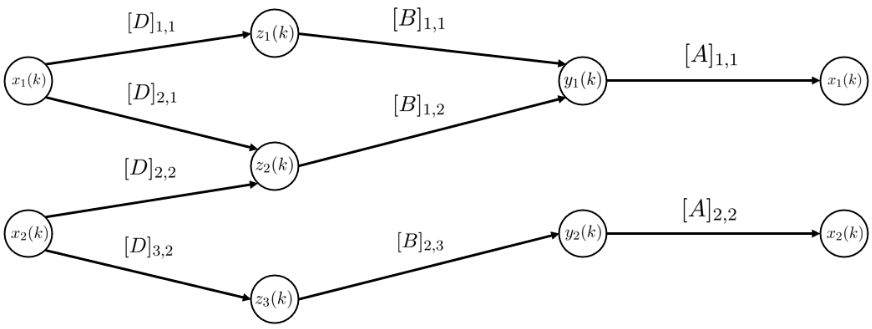


Figure 4-4: Visual representation of interconnection sequences in matrices A,B and D

It becomes visible that there are two distinct paths or ways state $x_1(k)$ is connected to state $x_1(k)$. This is a visual representation of how there are two sequences of entries in matrices A,B and D causing an implicit dependency within a state expression. The weights $[D]_{i,j}$, $[B]_{i,j}$ and $[D]_{i,j}$ represent the entries included in these sequences.

It is however possible that the linear term $z_i(k)$ appears in multiple expressions in $y(k)$. If both of these expressions then appear in the expression for state $x_q(k)$, the sequences of non-$0/\top/\varepsilon$ partly coincide. The same could theoretically be true for two linear terms $z_i(k)$ and $z_s(k)$, both containing state $x_j(k)$, which can both be included in the expression $y_p(k)$, which then appears in the state expression of $x_q(k)$. So it does not matter if in those entry sequences, parts overlap, however the combinations of the three entries composing a sequence must be unique to yield a value of $[S]_{q,j} > 1$. Ultimately, this results in the value of entry $[S]_{q,j}$ representing the number of times state $x_q(k)$ implicitly depends on state $x_j(k)$, which concludes this proof. □

So, the matrix S contains information about all implicit dependencies of states. Considering the knowledge proven in Theorem 4-2.4, the following theorem regarding the number of circuit modes can be proven;

Theorem 4-2.6. *(Maximum number of circuit modes in an MMPS system) Consider an implicit MMPS system containing a circuit, and therefore containing circuit modes. The maximum number of circuit modes is given by multiplying all non-zero entries with each other. This is mathematically described by;*

$$\Pi_{i=1}^{n_c} s_{c_i} \tag{4-39}$$

Where $s_{c_i}, i \in n_c$ is the number of non-0 entries in matrix S_c

Proof. As per Theorem 4-2.4, any entry $[S]_{ij}$ represents the number of affine terms in the expression of state $x_i(k)$ in which state $x_j(k)$ appears. The same holds for matrix S_c, as this matrix consists of entries taken from matrix S, and only considers states actually within a circuit. Matrix S_c excludes all explicit dependencies, and all implicit dependencies not related to any circuit. If the value of $[S_c]_{i,j}$ is 1, only one mode of state $x_i(k)$ will yield a circuit, in which it is in a circuit with $x_j(k)$. In case the value of $[S_c]_{i,j}$ is larger than 1, 3 for example, there are 3 modes of state $x_i(k)$ which yield a circuit with an implicit affine term including $x_j(k)$, so essentially, three distinct possible violations of the solvability of the system, that should be investigated separately. If another entry of S, $[S_c]_{p,q}$ for example, has value 2, yielding 2 possible violations of solvability. Considering both $[S_c]_{i,j} = 3$ and $[S_c]_{p,q} = 2$, and all other non-zero entries being 1, in total, there are a maximum of 6 modes in which a circuit occurs. Therefore, the maximum number of combinations of affine terms, i.e. modes, that yield a circuit is given by multiplying all non-0 entries in S_c. This leads to the following equation;

$$\Pi_{i=1}^{n_c} s_{c_i} \tag{4-40}$$

It is possible that the actual amount of circuit modes to be examined is actually lower than the value computed by 4-40. This is because the computed value does not account for affine terms containing more than one implicit state. Example 4-2.7 showcases this nicely. □

Example 4-2.7. *Consider the implicit Max-Min-Plus-Scaling system given by the following equations;*

$$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \begin{bmatrix} 0 & 0 & \varepsilon \\ \varepsilon & \varepsilon & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & \top & \top & \top \\ \top & 0 & \top & \top \\ \top & \top & 3 & 0 \end{bmatrix} \otimes' \left(\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \end{bmatrix} + \begin{bmatrix} 2 & -7 \\ 4 & 3 \\ 1 & 3 \\ 6 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \right) \tag{4-41}$$

$$S_A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad S_B = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad S_D = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 0 \end{bmatrix}, \quad S = \begin{bmatrix} 2 & 2 \\ 2 & 1 \end{bmatrix} \tag{4-42}$$

This system has multiple circuits, which can immediately be concluded, because matrix S does not have any entries of value ε.

According to the theory proposed in Theorem 4-2.6, the maximum number of circuit modes is equal to $\Pi_{i=1}^{2} s_{c_i} = 2 \cdot 2 \cdot 2 \cdot 1 = 8$*. However, the system only consists of two states, which each consist of two affine terms. And upon closer inspection, the actual number of circuit modes is equal to* 4*, not* 8 *as Theorem 4-2.6 suggests. Since the size of matrix* $S_D = S_{D_c} \in \mathbb{R}^{4\times 2}$*, there are only* 4 *distinct affine terms present, and* 2 *states in the circuit subsystem. Since the state evolution of each state is defined by one affine term, it is not possible to find* 8 *combinations of affine terms for* 2 *states, given that there are only* 4 *affine terms to make these combinations with. This example illustrates that Theorem 4-2.6 merely provides the theoretical maximum number of circuit modes, not necessarily the actual value.*

4-2-3 Degrees of Solvability

This subsection proposes a classification of implicit MMPS systems into four degrees of solvability. Each type is accompanied by an elaborate example. All theory proposed in this section is an extension of existing knowledge, unless specified otherwise. So, all results obtained in this Chapter so far leads to the following Theorem regarding solvability of MMPS states that contain at least one circuit;

Theorem 4-2.8. *(Solvability of Max Min Plus Scaling systems containing a circuit) An implicit Max Min Plus Scaling system containing a circuit is solvable if the circuit subsystem of all modes containing a circuit are solvable.*

Proof. In Theorem 4-1.1, it was proven that the absence of any circuit in the interconnection graph of matrix S is a sufficient condition to conclude an implicit MMPS system is solvable. Therefrom, it can be concluded that the only states that will possibly violate the solvability of an implicit MMPS system, are the states included in any circuit. Example 4-2.1 established the sufficiency of the condition derived in 4-1.4, proving implicit MMPS systems containing a circuit can still be solvable. By introducing the concepts of modes, circuit modes, and the circuit subsystem as per 4-2.1 and 4-2.2, the scope of the possible solvability violation was clarified. If only circuits can violate solvability, but not all modes contain a circuit, as was illustrated in example 4-2.2, only modes containing a circuit should be further analyzed. Within such a circuit mode, the states not contained in any circuit within that circuit mode will never make the system unsolvable, as their implicit dependency will never yield an entry in matrix F above the diagonal. If all possibly unsolvable parts of the system are proven to be solvable, the entire system is solvable. Therefore, if solvability is proven for the circuit subsystem in each circuit mode, the entire MMPS system is solvable. □

Having proven that solvable circuit subsystems in all circuit modes begets a solvable implicit MMPS system, an algorithm to actually substantiate this finding can be derived. Let us first propose an alternative notation for modes of a circuit subsystem, which will form the base for the classification methodology.

Proposition 4-2.2. *(Alternative notation of a circuit mode) Each mode of the circuit subsystem can be rewritten into a linear system of the following form;*

$$x_c(k) = Q \cdot x_c(k) + b_f \tag{4-43}$$

Where $Q \in \mathbb{R}^{n\times n}$ *is a square matrix representing the implicit dynamics of the circuit mode of the circuit subsystem.*

Vector $b_f \in \mathbb{R}^{n\times 1}$ contains all explicit states, implicit states not included in the circuit subsystem, and scalars that are part of the original expression of the states included in the circuit subsystem

Proof. Recall the circuit subsystem described by 4-30. This ABCD form of the circuit subsystem can be written out as a set of MMPS equations. By extending this set of MMPS equations by incorporating all terms that fell out of the equation as they were not terms of the circuit subsystem states, the full equations describing the circuit subsystem states of a circuit mode of the implicit MMPS system could be obtained. This set of MMPS equations, a linear set of equations can be determined to describe each possible mode. Let us illustrate this with the following example;

Example 4-2.9. *(Rewriting a circuit mode in a linear system of equations) Suppose an MMPS system is described by the following MMPS equations;*

$$\begin{aligned} x_1(k) &= x_2(k) + x_3(k-1) + 1 \\ x_2(k) &= -2x_1(k) + 3x_2(k) + x_3(k-1) + 2 \\ x_3(k) &= \min(x_1(k-1) + x_2(k-1) + x_2(k) - 3, x_1(k-1) + x_3(k-1)) \end{aligned} \tag{4-44}$$

By analyzing this system, it turns out that at least one circuit is present, and that the circuit subsystem is given by $x_c(k) = \begin{bmatrix} x_1(k)^T & x_2(k)^T \end{bmatrix}^T$, and by writing out the corresponding circuit subsystem state-space matrix equation, the only circuit mode can be defined by;

$$\begin{aligned} x_1(k) &= x_2(k) + x_3(k-1) + 1 \\ x_2(k) &= -2x_1(k) + 3x_2(k) + x_3(k-1) + 2 \end{aligned} \tag{4-45}$$

This system of linear equations in conventional algebra can be rewritten as;

$$x_c(k) = \underbrace{\begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix}}_{Q} \cdot x_c(k) + \begin{bmatrix} 1 \\ 2 \end{bmatrix} \tag{4-46}$$

By incorporating all terms that fell out of the original system equations due to the circuit subsystem matrix equation, the following system of linear equations can be obtained;

$$x_c(k) = \underbrace{\begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix}}_{Q} \cdot x_c(k) + \underbrace{\begin{bmatrix} x_3(k-1) + 1 \\ x_3(k-1) + 2 \end{bmatrix}}_{b_f} \tag{4-47}$$

Hereby, the full circuit mode of the implicit MMPS system is obtained, and rewritten into the proposed form of $x_c(k) = Q \cdot x_c(k) + b_f$

This example showcased how the circuit subsystem states in a circuit mode could be rewritten into a system of linear equations in conventional algebra. The symbols Q and b_f assigned to these parts of such a system of linear equations is the convention used in this thesis. □

As was mentioned in 4-0.1, a solvable implicit MMPS systems is required to have at least one solution. A distinction can be made between systems having a unique solution, and systems having multiple solutions, i.e. a parametric solution.

Firstly, let us introduce a theorem that provides a method aiming to determine whether a mode of a circuit subsystem has a unique solution.

Theorem 4-2.10. *(Unique solution of a mode of a circuit subsystem) Consider a circuit mode of a circuit subsystem. This mode can be rewritten into the following conventional algebraic linear system as per Theorem 4-2.2;*

$$x_c(k) = Q \cdot x_c(k) + b_f \tag{4-48}$$

If equation 4-49 has a unique solution, or equivalently, equation 4-50 holds, and matrix $(I-Q)$ has full rank, this circuit mode of this circuit subsystem has a unique solution, and is solvable.

$$(I - Q) \cdot x_c(k) = b_f \tag{4-49}$$

$$rank(I - Q) = n_c \tag{4-50}$$

Proof. In each mode, one entry per row of matrix A_c and B_c is active, and the active entries can be described by the pair of footprint matrices G_{A_c}, G_{B_c} corresponding to this particular mode. Matrix Q can be obtained using the following expression;

$$Q = G_{A_c} \cdot G_{B_c} \cdot D_c \tag{4-51}$$

Matrices G_{A_c} and G_{B_c} essentially select the entries of matrix D_c active in the corresponding circuit mode. The circuit mode is rewritten as a system of linear equations in 4-48 within conventional algebra. Using Gaussian elimination, it can easily be concluded that, if matrix $(I - Q)$ is full rank, the system of linear equations is uniquely solvable. This leads to the conclusion that, if matrix $(I - Q)$ is full rank, or equivalently, there is only one solution to $(I - Q) \cdot x_c(k) = b_f$, this circuit subsystem circuit mode is uniquely solvable. □

This theorem can be accompanied by an elaborate example, aiming to illustrate a situation in which a uniquely solvable circuit mode occurs;

Example 4-2.11. *(Uniquely solvable circuit mode of an implicit MMPS system) Consider an MMPS system given by the following ABCD form;*

$$\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} = \begin{bmatrix} 3 & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top \\ \top & \top & 2 & 0 & \top \\ \top & \top & \top & \top & 0 \end{bmatrix} \otimes' \left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 3 \\ 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \right) \tag{4-52}$$

With matrices S_A,S_B,S_D and S as follows;

$$S_A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad S_B = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad S_D = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}, \quad S = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix} \tag{4-53}$$

This system contains at least one cycle, as becomes apparent when computing $S_\otimes^+$, which is given by;

$$S_\otimes^+ = \begin{bmatrix} 2 & 2 & 3 \\ \varepsilon & \varepsilon & \varepsilon \\ 3 & 3 & 2 \end{bmatrix} \tag{4-54}$$

Since entries $[S_\otimes^+]_{11}$ and $[S_\otimes^+]_{33}$ are non-ε, a path from node 3 to node 3, and a path from node 1 to node 1 exists. The circuit subsystem can therefore be given by;

$$\begin{aligned} \begin{bmatrix} x_1(k) \\ x_3(k) \end{bmatrix} = \begin{bmatrix} 3 & 0 & \varepsilon \\ \varepsilon & \varepsilon & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & \top & \top \\ \top & 0 & \top \\ \top & \top & 0 \end{bmatrix} & \\ \otimes' \left(\begin{bmatrix} 0 & 0 \\ 0 & 3 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k-1) \\ x_3(k-1) \end{bmatrix} + \begin{bmatrix} 0 & 2 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_3(k) \end{bmatrix} \right) & \end{aligned} \tag{4-55}$$

Matrix S_c corresponding to this circuit subsystem is given by;

$$S_c = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{4-56}$$

Therefore, the maximum number of circuit modes is given by $1 \cdot 1 = 1$. When inspecting the system, the full circuit mode that is to be analyzed can be given by the following system of linear equations;

$$\begin{aligned} x_1(k) &= 2x_3(k) + 3 \\ x_3(k) &= x_1(k) + x_2(k) \end{aligned} \tag{4-57}$$

When rewriting this system into the form proposed in 4-2.2, the following expression is found;

$$x_c(k) = \underbrace{\begin{bmatrix} 0 & 2 \\ 1 & 0 \end{bmatrix}}_{Q} \cdot x_c(k) + \underbrace{\begin{bmatrix} 3 \\ x_2(k) \end{bmatrix}}_{b_f} \tag{4-58}$$

All that is left is to simply determine if $(I - Q)$ is full rank, as to be able to conclude a unique solution exists for this circuit subsystem circuit mode;

$$rank(I - Q) = rank(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 2 \\ 1 & 0 \end{bmatrix}) = rank(\begin{bmatrix} 1 & -2 \\ -1 & 1 \end{bmatrix}) = 2 = n_c \tag{4-59}$$

So, $(I - Q)$ is full rank, so the only circuit mode in the circuit subsystem given in 4-55 is uniquely solvable. As this was the only instance in this system where solvability could be violated, and it has been proven that it will not. Therefore, the entire system as given in 4-52 is uniquely solvable, as all possible modes are uniquely solvable.

Theorem 4-2.10 proposed a method to determine whether a circuit subsystem circuit mode has a unique solution. In case this unique solution exists for all circuit subsystem circuit modes, the entire MMPS system is solvable, yielding a unique solution for all $x(k), k > 0$ for any $x(k-1)$. In case one or more circuit modes of the circuit subsystem do not have a unique solution, the system could still be solvable.

However, in this case, there will be at least one mode of the system that is not *uniquely* solvable. Regarding circuit subsystem circuit modes that do not have a unique solution, a discrepancy can be made between systems that are *unsolvable* and systems that are *parametrically solvable*. The following two theorems propose a method which conclusively proves which case transpires.

Theorem 4-2.12. *(Parametric solution of a mode of a circuit subsystem) Consider a circuit mode of a circuit subsystem described by the following equation;*

$$x_c(k) = Q \cdot x_c(k) + b_f \tag{4-60}$$

If the following equation holds, this circuit mode of the circuit subsystem is said to be solvable, with a parametric solution;

$$rank(I - Q) = rank(I - Q|b_f) < n_c \tag{4-61}$$

Here, vector b_f is similar to vector b as defined in 4-48. The main difference is that all explicit, and implicit terms which were initially included in the affine term, but fell out due to the reduction of states to only the circuit states, are included again.

Proof. Any system that has a parametric solution will not have full rank $(I-Q)$, the reduced version of this matrix will be of the following structure;

$$(I - Q) = \begin{bmatrix} \star & 0 & \cdots & 0 \\ 0 & \star & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \tag{4-62}$$

This system is by definition underdefined, and could have a parametric solution. Extended matrix $(I - Q|b_f)$ may be able to be reduced into the following form;

$$(I - Q|b_f) = \left[\begin{array}{cccc|c} \star & 0 & \cdots & 0 & \star \\ 0 & \star & \cdots & 0 & \star \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \end{array}\right] \tag{4-63}$$

If this form exists, there is no inconsistency in the system, and the solution to this circuit subsystem circuit mode is parametric. In case $\text{rank}(I - Q) = \text{rank}(I - Q|b_f) = n_c$, $(I - Q)$ if full rank, and a unique solution exists, as described in Theorem 4-2.10. □

This theorem can be illustrated by the following example;

Example 4-2.13. *(Parametrically solvable circuit mode of an implicit MMPS system) Consider an MMPS system given by the following ABCD form;*

$$\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} = \begin{bmatrix} 2 & 1 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top \\ \top & \top & 2 & \top & \top \\ \top & \top & \top & -3 & 0 \end{bmatrix} \otimes' \left(\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 2 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ -2 & 3 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \right) \tag{4-64}$$

Matrices S_A, S_B, S_D and S can be given as follows;

$$S_A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad S_B = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}, \quad S_D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad S = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{4-65}$$

Using these matrices, matrix $S_{\otimes}^{+}$ can be computed;

$$S_{\otimes}^{+} = \begin{bmatrix} 3 & 3 & \varepsilon \\ 3 & 3 & \varepsilon \\ 3 & 3 & \varepsilon \end{bmatrix} \tag{4-66}$$

So, at least 1 circuit is present in this system, the circuit subsystem consists of states $x_1(k)$ and $x_2(k)$, and is given by the following equation;

$$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \begin{bmatrix} 2 & 1 & \varepsilon \\ \varepsilon & \varepsilon & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & \top & \top \\ \top & 0 & \top \\ \top & \top & 2 \end{bmatrix} \otimes' \left(\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ -2 & 3 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \right) \tag{4-67}$$

Matrix S_c corresponding to this circuit subsystem is given by;

$$S_c = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \tag{4-68}$$

Therefore, the maximum number of circuit modes is given by $1 \cdot 1 \cdot 1 = 1$. The to be analyzed circuit mode can be given by the following set of MMPS equations;

$$\begin{aligned} x_1(k) &= x_2(k) + x_3(k-1) + 1 \\ x_2(k) &= -2x_1(k) + 3x_2(k) + 2x_3(k-1) + 2 \end{aligned} \tag{4-69}$$

Which in turn, can be rewritten into the system of linear equations in conventional algebra as per 4-2.2, given by;

$$x_c(k) = \underbrace{\begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix}}_{Q} \cdot x_c(k) + \underbrace{\begin{bmatrix} x_3(k-1)+1 \\ 2x_3(k-1)+2 \end{bmatrix}}_{b_f} \tag{4-70}$$

Now the following test can be perormed to determine the rank of $(I-Q)$, which is done below;

$$rank(I-Q) = rank(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix}) = rank(\begin{bmatrix} 1 & -1 \\ 2 & -2 \end{bmatrix}) = 1 \neq n_c \tag{4-71}$$

So, $(I-Q)$ is not full rank, therefore, this circuit mode of the circuit subsystem does not have a unique solution. However, the solution to this system may be parametric. By applying the theory presented in Theorem 4-2.12

Now evaluating whether $rank(I-Q) = rank(I-Q|b_f) < n_c$ allows us to draw conclusions regarding the parametric solvability of this circuit mode;

$$\begin{aligned} rank(I-Q|b_f) &= rank(\left[\begin{array}{cc|c} 1 & -1 & x_3(k-1)+1 \\ 2 & -2 & 2x_3(k-1)+2 \end{array}\right]) \\ &= rank(\left[\begin{array}{cc|c} 1 & -1 & x_3(k-1)+1 \\ 0 & 0 & 0 \end{array}\right]) = 1 = rank(I-Q) \end{aligned} \tag{4-72}$$

So, this convincingly proves that $rank(I-Q) = rank(I-Q|b_f) < n_c$ holds, and a parametric solution to this mode exists. Since the only circuit mode in the circuit subsystem has been proven to be solvable, the entire system can be considered solvable, with one mode having infinitely many solutions.

In case Theorems 4-2.10 and 4-2.12 do not hold, the implicit MMPS system may be unsolvable. The method to determine whether a circuit mode of an implicit MMPS system is unsolvable is described by the following theorem;

Theorem 4-2.14. *(Unsolvable mode of a circuit subsystem) Consider a circuit mode of a circuit subsystem described by the following equation;*

$$x_c(k) = Q \cdot x_c(k) + b_f \tag{4-73}$$

In the case that $rank(I-Q) < rank(I-Q|b_f)$, this circuit mode of the circuit subsystem is said to be unsolvable.

Proof. In case that $\text{rank}(I-Q) < \text{rank}(I-Q|b_f)$, the reduced form of this inequality will have the following structure;

$$\text{rank}\left(\begin{bmatrix} \star & 0 & \cdots & 0 \\ 0 & \star & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}\right) < \text{rank}\left(\left[\begin{array}{cccc|c} \star & 0 & \cdots & 0 & \star \\ 0 & \star & \cdots & 0 & \star \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & \star \end{array}\right]\right) \tag{4-74}$$

It becomes apparent that in case $\text{rank}(I-Q) < \text{rank}(I-Q|b_f)$, at least one row of $(I-Q|b_f)$ contains an inconsistency in the system of linear equations. For example, interpretation such an inconsistency can be described by;

$$x_1(k) = x_1(k) + 6 \tag{4-75}$$

There is no value of $x_1(k)$ for which this equation can be solved. Such an inconsistency within the system of linear equation describing the circuit mode will lead to $\text{rank}(I-Q) < \text{rank}(I-Q|b_f)$. Such behaviour will cause this circuit mode to be unsolvable. □

To illustrate the theorem regarding unsolvable systems, the following example is provided;
Example 4-2.15. *(Unsolvable mode of a circuit subsystem) Consider an MMPS system given by the following ABCD form;*

$$\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} = \begin{bmatrix} 6 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 3 & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & \top & \top & \top & \top \\ \top & 1 & 0 & \top & \top \\ \top & \top & \top & 0 & \top \\ \top & \top & \top & \top & 0 \end{bmatrix}$$
$$\otimes' \left(\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \\ 0 & -4 & 3 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \right) \tag{4-76}$$

Matrices S_A, S_B, S_D and S can be given as follows;

$$S_A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad S_B = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad S_D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}, \quad S = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 2 \\ 0 & 1 & 1 \end{bmatrix} \tag{4-77}$$

Using these matrices, matrix $S_\otimes^+$ can be computed;

$$S_\otimes^+ = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 4 & 5 \\ \varepsilon & 4 & 4 \end{bmatrix} \tag{4-78}$$

Due to entry $[S_\otimes^+]_{11}$ and $[S_\otimes^+]_{22}$ being non-ε, this system contains at least one circuit, and its circuit subsystem consists of state $x_2(k)$ and $x_3(k)$. The circuit subsystem can be given by;

$$\begin{bmatrix} x_2(k) \\ x_3(k) \end{bmatrix} = \begin{bmatrix} 0 & \varepsilon & \varepsilon \\ \varepsilon & 0 & 3 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & \top & \top \\ \top & \top & 0 & \top \\ \top & \top & \top & 0 \end{bmatrix}$$
$$\otimes' \left(\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -1 & 1 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_2(k-1) \\ x_3(k-1) \end{bmatrix} + \begin{bmatrix} -1 & 1 \\ 0 & 2 \\ 0 & 0 \\ -4 & 3 \end{bmatrix} \cdot \begin{bmatrix} x_2(k) \\ x_3(k) \end{bmatrix} \right) \tag{4-79}$$

This circuit subsystem will turn out to be unsolvable, but to prevent an unnecessarily long calculation, the test to prove a unique solution as proposed in Theorem 4-2.10 will not be carried out. Instead, the proposed algorithm in 4-2.12 and 4-2.14 will be applied. Any system that is solvable will never prove to be unsolvable by applying this algorithm. The circuit subsystem as per 4-79 has the following corresponding matrix S_c;

$$S_c = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \tag{4-80}$$

The maximum number of circuit modes in this circuit subsystem is given $1 \cdot 2 \cdot 1 \cdot 1 = 2$. Upon further inspection of this circuit subsystem, there indeed exist two different circuit modes. The following linear system of equations can be obtained;

$$\begin{cases} x_2(k) & = -x_2(k) + x_3(k) + 1 \\ x_3(k) & = -4x_2(k) + 3x_3(k) + 3 \end{cases} \tag{4-81}$$

$$\begin{cases} x_2(k) & = 2x_3(k) + x_1(k-1) \\ x_3(k) & = -4x_2(k) + 3x_3(k) + 3 \end{cases} \tag{4-82}$$

Which in turn can be rewritten into;

$$x_c(k) = \underbrace{\begin{bmatrix} -1 & 1 \\ -4 & 3 \end{bmatrix}}_{Q} \cdot x_c(k) + \underbrace{\begin{bmatrix} 1 \\ 3 \end{bmatrix}}_{b_f} \tag{4-83}$$

$$x_c(k) = \underbrace{\begin{bmatrix} 0 & 2 \\ -4 & 3 \end{bmatrix}}_{Q} \cdot x_c(k) + \underbrace{\begin{bmatrix} x_1(k-1) \\ 3 \end{bmatrix}}_{b_f} \tag{4-84}$$

In both cases, it should be evaluated whether $rank(I - Q) = rank(I - Q|b_f)$;

$$rank(I - Q) = rank(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} -1 & 1 \\ -4 & 3 \end{bmatrix}) = rank(\begin{bmatrix} 2 & -1 \\ 4 & -2 \end{bmatrix}) = 1 \neq n_c \tag{4-85}$$

$$rank(I - Q) = rank(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 2 \\ -4 & 3 \end{bmatrix}) = rank(\begin{bmatrix} 1 & -2 \\ 4 & -2 \end{bmatrix}) = 2 = n_c \tag{4-86}$$

The first system has $rank(I - Q) = 1$, which is not full rank, making it either parametrically solvable, or unsolvable. So, the second system has $rank(I - Q) = 2$, which is full rank, making this circuit mode uniquely solvable. The first system has to be analyzed further to determine its degree of solvability;

$$rank(I - Q|b_f) = rank(\left[\begin{array}{cc|c} 2 & -1 & 1 \\ 4 & -2 & 3 \end{array}\right]) = rank(\left[\begin{array}{cc|c} 2 & -1 & 1 \\ 0 & 0 & 1 \end{array}\right]) = 2 \neq rank(I - Q) \tag{4-87}$$

So, this linear system of equations is inconsistent, and therefore, unsolvable. Therefore, the corresponding circuit mode is unsolvable, and the entire system is unsolvable, because one mode is unsolvable. When reflecting back on the definition of solvability as per 4-0.1, the system given in this Example does not always have a solution $x(k), k > 0$ for any state $x(k-1)$.

In this section, three degrees of solvability were introduced. A system can either be uniquely solvable, parametrically solvable, or unsolvable. However, a remark regarding unsolvable systems needs to be made to accommodate for inconsistent systems, which under certain conditions, may still be parametrically solvable. The following example illustrates this phenomenon;

Example 4-2.16. *(Parametrically unsolvable system) Consider a circuit mode of a circuit subsystem given by the following linear system of equations, matrix notation;*

$$x_c(k) = \underbrace{\begin{bmatrix} -4 & -3 \\ -10 & -5 \end{bmatrix}}_{Q} \cdot x_c(k) + \underbrace{\begin{bmatrix} 3 + x_1(k-1) \\ 2x_1(k-1) + x_2(k-1) + 5 \end{bmatrix}}_{b_f} \tag{4-88}$$

By evaluating $rank(I - Q|b_f)$*, the following can be obtained;*

$$\begin{aligned} rank(I - Q|b_f) &= rank(\left[\begin{array}{cc|c} 5 & 3 & 3 + x_1(k-1) \\ 10 & 6 & 2x_1(k-1) + x_2(k-1) + 5 \end{array}\right]) \\ &= rank(\left[\begin{array}{cc|c} 5 & 3 & 3 + x_1(k-1) \\ 0 & 0 & x_2(k-1) - 1 \end{array}\right]) = 2 \neq rank(I - Q) \end{aligned} \tag{4-89}$$

Therefore, this system of linear equations corresponding to the given circuit mode, is theoretically unsolvable.

The solution to the system of linear equations as per is definitely of the structure as seen in 4-2.14;

$$\text{rank}\left(\begin{bmatrix} \star & 0 & \cdots & 0 \\ 0 & \star & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}\right) < \text{rank}\left(\left[\begin{array}{cccc|c} \star & 0 & \cdots & 0 & \star \\ 0 & \star & \cdots & 0 & \star \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & \star \end{array}\right]\right) \tag{4-90}$$

However, the term $x_1(k-1) - 1$, which is causing the inconsistency in the system, is actually parametric. So, in the specific case that $x_1(k-1)-1 = 0$, the system would be parametrically solvable. However, definitely falls under the category "even a broken clock is right twice a day", and it is still definitely true that $x(k), k > 0$ does not always have a solution for any $x(k-1)$, still yielding an unsolvable system. Nevertheless, the theorem for unsolvable modes of a circuit subsystem does need to be extended to accommodate for this case, by adding the possibility of a system being *parametrically unsolvable*. Systems where the term(s) causing the inconsistency are merely a scalar, will from henceforth be referred to as *strictly unsolvable systems*.

Theorem 4-2.17. *(Parametrically unsolvable mode of a circuit subsystem) Consider a circuit mode of a circuit subsystem described by the following equation;*

$$x_c(k) = Q \cdot x_c(k) + b_f \tag{4-91}$$

In the case that $rank(I - Q) < rank(I - Q|b_f)$*, and all terms causing this system of linear equations to be inconsistent are parametric, the system may be parametrically unsolvable. In case there exists an initial condition* $x(k-1)$ *for which all terms causing the inconsistency are* 0*, yielding a parametrically solvable system, the corresponding circuit mode is parametrically unsolvable.*

Proof. Similarly to strictly unsolvable circuit modes, the inequality $\text{rank}(I - Q) < \text{rank}(I - Q|b_f)$ can be reduced to having the following structure;

$$\text{rank}\left(\begin{bmatrix} \star & 0 & \cdots & 0 \\ 0 & \star & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}\right) < \text{rank}\left(\left[\begin{array}{cccc|c} \star & 0 & \cdots & 0 & \star \\ 0 & \star & \cdots & 0 & \star \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & u \end{array}\right]\right) \tag{4-92}$$

In case u is a (set of) parametric expression(s), it is theoretically possible for this reduced system to obtain the following structure, if $u = 0$, yielding a parametrically solvable system.

$$(I - Q|b_f) = \left[\begin{array}{cccc|c} \star & 0 & \cdots & 0 & \star \\ 0 & \star & \cdots & 0 & \star \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \end{array}\right] \tag{4-93}$$

If it turns out that the equality $u = 0$ holds for **all** initial conditions $x(k-1)$, the system just becomes parametrically solvable. This will very rarely be the case, as there likely exists an initial condition for which $u \neq 0$. □

4-3 Deriving a Necessary Solvability Condition

As proven in Section 4-2-1, the solvability conditions as defined in [15] are merely sufficient conditions, not necessary conditions. Within the field of mathematical proofs, a necessary condition is a much stronger condition than a sufficient proof. In order to prove the necessity of the derived solvability theorems, a proof by contradiction can be carried out. A system will be assumed to be solvable, but the solvability theorems do not hold. Firstly, an all-encompassing solvability theorem can be defined, combining all degrees of solvability in order to accommodate the given definition of solvability, as per 4-0.1. This is relevant because a system may violate Theorem 4-2.10, because the solution to some circuit mode is not unique, but parametric. Therefore the entire system still complies with the definition of a solvable system. The theorem given below proposes a condition which will prove to be a necessary condition for solvability of an implicit MMPS system containing a circuit;

Theorem 4-3.1. *(Solvability of implicit MMPS systems containing a circuit) An implicit MMPS system as in 4-1that is assumed to be a minimal realization, i.e. there are no redundant or logically unreachable modes, and contains at least one circuit, is solvable if and only if for all circuit modes of the circuit subsystem, it holds that* $\text{rank}(I - Q) = \text{rank}(I - Q|b_f) \leq n_c$ *for all* $x(k-1)$

Proof. Let us assume a circuit mode of a circuit subsystem is solvable, and described by the following equation;

$$x_c(k) = Q \cdot x_c(k) + b_f \tag{4-94}$$

Let us also assume that $\text{rank}(I - Q) \neq \text{rank}(I - Q|b_f) \leq n_c$ for all $x(k-1)$, but the system this circuit mode corresponds to is still solvable.

Since $\text{rank}(I-Q) = \text{rank}(I-Q|b_f) \leq n_c$ for all $x(k-1)$ does not hold, the rank of $(I-Q|b_f)$ of the system of linear equations as per 4-94 can be rewritten into the following structure;

$$\text{rank}\left(\left[\begin{array}{cccc|c} \star & 0 & \cdots & 0 & \star \\ 0 & \star & \cdots & 0 & \star \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & \star \end{array}\right]\right) \tag{4-95}$$

In order for the system of linear equations to be solvable, they must not be inconsistent. The system rewritten into the structure above, is inconsistent for some $x(k-1)$. This causes a logical contradiction in the reasoning, as a circuit mode of a circuit subsystem that is solvable, but $\text{rank}(I-Q) = \text{rank}(I-Q|b_f) \leq n_c$ for all $x(k-1)$ does not hold, is inconsistent for at least some $x(k-1)$, and therefore it cannot be solvable. Since it is assumed that this unsolvable mode can logically be reached, the system can end up in this unsolvable mode. Therefore, it can be concluded that an implicit MMPS system as in 4-1 that contains at least one circuit, is solvable if and only if for all circuit modes of the circuit subsystem, it holds that $\text{rank}(I-Q) = \text{rank}(I-Q|b_f) \leq n_c$ for all $x(k-1)$. □

4-4 Systems with Circuits and Conventional Algebraic Mapping

The previous section provided an extension of the solvability criterion, providing a sufficient, and a necessary solvability condition for implicit MMPS systems. The initial solvability criterion assumed the existence of lower-triangular matrix F, see Equation 3-23. When aiming to reformulate an implicit MMPS system as a linear system in conventional algebra, as seen in 3-5.4, the assumption is made that such a transformation is possible, assuming the existence of strictly lower-triangular matrix F. In implicit MMPS systems where a circuit occurs, no such strictly lower-triangular matrix F exists even though the system may be solvable. Therefore, it cannot be immediately be stated that it is possible to recast the implicit MMPS system into a linear conventional form. The aim of this section is to derive the conditions under which implicit MMPS systems containing a circuit can be recast into a conventional linear form which is valid in region Ω_θ. Firstly, theory regarding the linear mapping of implicit MMPS systems in conventional algebra is recalled from Section 3-5. Thereafter, the conditions for invertibility of matrix $(I-M_1)$ are extended, which is the main contribution of this chapter.

4-4-1 Review of Existing Theory

In [15], the following definition is introduced and proven;

Definition 4-4.1. *[15] (Conventional algebra notation of implicit MMPS system) Any normalized implicit MMPS system can be reformulated as a linear system in conventional algebraic notation for all* $\tilde{x}_\theta(k) \in \Omega_\theta, k \in \mathbb{Z}^+$ *using the following expressions;*

$$\begin{aligned} \tilde{x}_\theta(k) &= M_\theta \cdot \tilde{x}_\theta(k-1) \\ M_\theta &= (I-M_1)^{-1} \cdot M_2 \\ M_1 &= G_{\mathrm{A}_\theta} \cdot G_{\mathrm{B}_\theta} \cdot D \\ M_2 &= G_{\mathrm{A}_\theta} \cdot G_{\mathrm{B}_\theta} \cdot C \end{aligned} \tag{4-96}$$

if the inverse $(I-M_1)^{-1}$ *exists*

The specific part of the proof of this definition that is of importance in this section, is regarding the existence of the inverse of $(I - M_1)^{-1}$. In [15], the existence of the inverse of $(I - M_1)^{-1}$ is guaranteed by the following line of reasoning. Knowing $M_1 = G_{\mathrm{A}_\theta} \cdot G_{\mathrm{B}_\theta} \cdot D$, the structure of M_1 is essentially the same as $S_A \cdot S_B \cdot S_D$, but with some entries of S_A and S_B removed, as to obtain the footprint matrices. The product of any footprint matrix combination and matrix D, $G_A \cdot G_B \cdot D$, therefore preserves the strictly lower-triangular structure, which in turn leads to $(I - M_1)$ having full rank, causing the inverse of $(I - M_1)$ to always exist.

4-4-2 Conditions for Invertibility of $(I - M_1)$

From Section 4-4-1, it becomes apparent that due to the existence of strictly lower-triangular matrix F, the existence of the inverse of $(I-M_1)$ can be guaranteed. For MMPS systems with a circuit, this property does not hold true. Firstly, let us properly introduce the requirements for a matrix to be invertible;

Definition 4-4.2. *[1] (Existence of the inverse of a matrix) A matrix $A \in \mathbb{R}^{n\times n}$ is invertible if its determinant in not equal to* 0*, so Det*$(A) \neq 0$.

Considering this definition, it is clear why a strictly lower-triangular M_1 results in $(I - M_1)$ being invertible, as the matrix $(I - M_1)$ is of full rank, and $\mathrm{Det}(I - M_1) = 1$. From this point onwards, the matrix M_1 of an MMPS system containing a circuit, will be referred to as M_{1c}, and conditions will be derived under which the inverse of $(I - M_{1c})$ exists. The expression that calculates matrix M_{1c} is given by;

$$M_{1c} = G_{A_\theta} \cdot G_{B_\theta} \cdot D \tag{4-97}$$

Even though $(I - M_{1c})$ cannot be represented as a lower-triangular matrix with diagonal entries of 1, it can be represented as a block lower-triangular matrix, as per the theorem given below;

Theorem 4-4.1. *(Block lower-triangular form of $(I-M_{1c})$ For every solvable implicit MMPS system containing at least one circuit, a block lower-triangular representation $(I - M_{1c})_B$ exists. Such a block lower-triangular matrix is of the following form;*

$$(I - M_{1c})_B = \begin{bmatrix} L_1 & 0 & \cdots & 0 \\ * & L_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ * & \cdots & * & L_n \end{bmatrix} \tag{4-98}$$

Where $(I - M_{1c})_B \in \mathbb{R}^{n\times n}$, $L_i \in \mathbb{R}^{l\times l}, l \leq n$

Proof. The matrix $(I - M_{1c})$ is computed by subtracting matrix M_{1c} from identity matrix I. Identity matrix I is a block diagonal matrix with all blocks being of size $\mathbb{R}^{1\times 1}$. Matrix M_{1c} is constructed of $G_{A_\theta} \cdot G_{B_\theta} \cdot D$, which, depending on whether the dominant mode corresponding to this eigenvalue contains a circuit, cannot be transformed into a strictly lower-triangular matrix. However, a permutation exists such that the only nonzero (upper) diagonal matrices, are corresponding to the cyclic states. This is easily proven by inspecting the proof given for Theorem 4-1.1. The proof states that a strictly lower-triangular matrix F exists because of the possibility to topologically order all edges in the graph such that no entry in matrix F would appear on, or above the diagonal, due to the absence of a circuit.

It is known that for matrix S of a system containing a circuit, which is related to matrix M_1 as per [15], no topological ordering exists such that matrix F is strictly lower-triangular. As all states not included in a circuit do not cause any entries on, or above the diagonal of matrix F to be non-zero, a topological ordering of the edges must exist such that the only entries on, or above the diagonal are corresponding to the states included in any circuit. Hereby, the block-lower-triangular form of matrix M_{1c} can be obtained. By subtracting the permuted block lower-triangular matrix M_{1c} from I, a block lower-triangular matrix is obtained. □

Knowing such a block lower-triangular form $(I - M_{1c})_B$ exists for any $(I - M_{1c})$ forms the basis in understanding the conditions under which $(I - M_{1c})_B$, and therefore $(I - M_{1c})$ is invertible. From Definition 4-4.2, a matrix is proven to be invertible if its determinant is non-zero. This definition combined with the block lower-triangular form $(I - M_{1c})_B$ allows for the introduction of the following Theorem;

Theorem 4-4.2. *(Invertibility of matrix $(I - M_{1c})_B$) Let $(I - M_{1c})_B$ be a block lower-triangular matrix with diagonal blocks $\{L_i\}$. Matrix $(I - M_{1c})_B$ is invertible if the following statement holds;*

$$\forall i, \quad \begin{cases} \textit{if } \dim(L_i) > 1, & \det(L_i) \neq 0 \\ \textit{if } \dim(L_i) = 1, & L_i \neq 0 \end{cases} \tag{4-99}$$

Proof. From the definition of invertibility of a matrix as per Definition 4-4.2, the inverse of $(I - M_{1c})_B$ exists if its determinant is not equal to 0. The determinant of $(I - M_{1c})_B$ is computed by the following equation;

$$\text{Det}((I - M_{1c})_B) = \text{Det}(L_1) \cdot \text{Det}(L_2) \cdot \cdots \cdot \text{Det}(L_n) \tag{4-100}$$

Which is only non-zero if the determinant of block L_i is non-zero for all i. □

For acyclic matrices M_1, All diagonal blocks L_i are of size 1×1, so $L_i \in \mathbb{R}^{1\times 1} \quad \forall i \in \{1, ..., n\}$, and have value 1.

Chapter 5

Control of Implicit MMPS Systems

The contributions of this chapter lie in the proposition of open-loop and closed-loop control strategies for implicit MMPS systems. In [4], multiple control strategies for MMPS systems were proposed. However, the work was mostly limited to control of explicit MMPS systems, and multiple unjust assumptions were made. The aim of this chapter is to extend the existing explicit control strategies to incorporate implicit MMPS systems, and derive general descriptions for such a framework. The results presented in this chapter are additions to the existing knowledge. If material is taken from an existing source, this is specifically mentioned. Section 5-1 proposes a general input expression that includes implicit dynamics. Subsequently, Section 5-2 introduces open-loop control strategies for implicit MMPS systems. Furthermore, this section contains the derivation of time-invariance- and solvability conditions for the open-loop controlled implicit MMPS system. The last section, Section 5-3 offers closed-loop control strategies for implicit MMPS systems, and describes a closed-loop extended-state-space description. Ultimately, conditions for time-invariance and solvability of the proposed closed-loop control strategies are provided as well. The ABCD canonical form as given in 3-1.5, from [15]. In [4], an extension of the ABCD canonical form was presented, where input signals were implemented into the system definition;

Definition 5-0.1. *[4] (Implicit ABCDE form) The ABCD canonical from given in 3-1.5 can be extended with an additional input matrix (E) multiplied with input vector $u(k)$ consisting of both temporal and quantity input signals;*

$$\begin{bmatrix} x_{\mathrm{t}}(k) \\ x_{\mathrm{q}}(k) \end{bmatrix} = \underbrace{\begin{bmatrix} A_{\mathrm{t}} & \varepsilon \\ \varepsilon & A_{\mathrm{q}} \end{bmatrix}}_{A} \otimes \left(\underbrace{\begin{bmatrix} B_{\mathrm{t}} & \top \\ \top & B_{\mathrm{q}} \end{bmatrix}}_{B} \otimes' \left(\underbrace{\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_{\mathrm{t}}(k-1) \\ x_{\mathrm{q}}(k-1) \end{bmatrix} \right.\right.$$
$$\left.\left. + \underbrace{\begin{bmatrix} D_{11} D_{12} \\ D_{21} D_{22} \end{bmatrix}}_{D} \cdot \begin{bmatrix} x_t(k) \\ x_q(k) \end{bmatrix} + \underbrace{\begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix}}_{E} \cdot \begin{bmatrix} u_t(k) \\ u_q(k) \end{bmatrix} \right)\right) \tag{5-1}$$

The block diagram of this ABCDE canonical form is given in Figure 5-1;

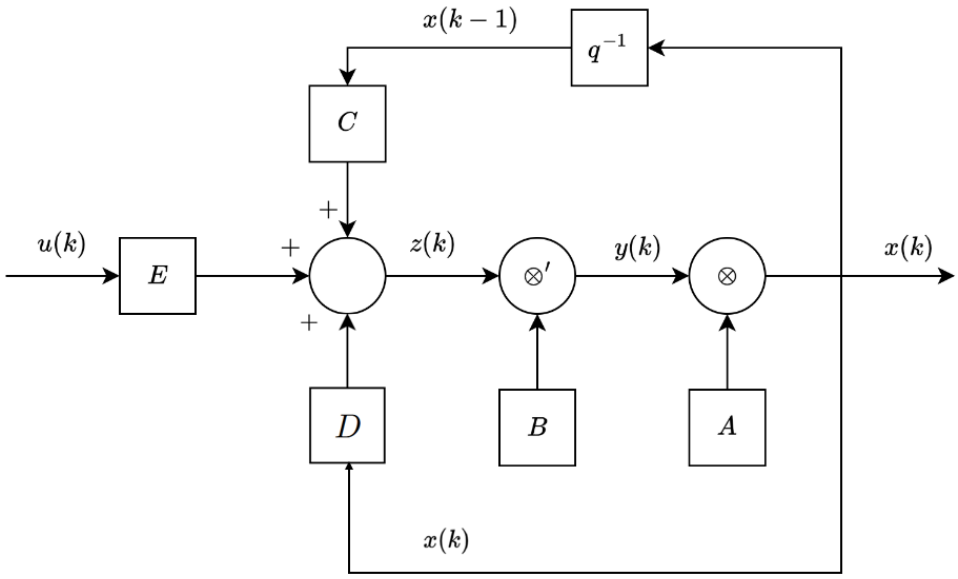


Figure 5-1: Block diagram of an implicit MMPS system with an applied input signal

After defining this ABCDE canonical form it was assumed in [4] that $D = 0$, and therefore, all presented results only apply to explicit MMPS systems. When aiming to apply the control strategies for explicit systems to implicit MMPS systems, logical additions to the theory proposed in [4] can be done. Firstly, a general input function is derived, allowing for implicit input signals.

5-1 Adding Input to Implicit MMPS Systems

Adding an input signal of any kind will change the implicit MMPS description, as the ABCDE form in 5-1 shows. Considering what this input signal consists of is of great importance. Depending on the expression of the input signal, either open-loop control, or closed-loop control occurs. In this chapter, both open-loop and closed-loop input signals will be analyzed, and their influence on the MMPS system description, or characteristics like time-invariance. A function for the input signal encompassing all types of input signals can be described by;

$$u(k) = f(x(k), u(k), r(k)) \tag{5-2}$$

Where $x(k)$ is the state of the system, $u(k)$ is the input signal itself, as the input may implicitly, or explicitly depend on itself. Lastly, $r(k)$ is the reference signal. A matrix-based expression can be derived for the most general input function, essentially a translation of the ABCDE form used to describe the dynamics of the state, but for the input;

Proposition 5-1.1. *(Implicit input function) Any input signal for an MMPS system can be described by the following expression;*

$$u(k) = F \otimes (H \otimes' (K_0 \cdot x(k-1) + K_1 \cdot x(k) + L_0 \cdot u(k-1) + L_1 \cdot u(k) + R_0 \cdot r(k-1) + R_1 \cdot r(k)) \quad (5\text{-}3)$$

Where $F \in \mathbb{R}_{\varepsilon}^{n \times m}$, $H \in \mathbb{R}_{\top}^{m \times p}$, $K_0, K_1 \in \mathbb{R}^{p \times n_x}$, $L_0, L_1 \in \mathbb{R}^{p \times n_u}$, $R_0, R_1 \in \mathbb{R}^{p \times r}$ and $k \in \mathbb{Z}^+$. Depending on the type of signal input, and therefore, the type of control applied, some of these matrices may be not relevant.

Proof. This is an extension of the input function proposed in [4]. The implicit terms in this expression can logically be added. □

This input function itself is also an MMPS function, and therefore, an extended state form for this function can be derived, similar to how the extended state form was defined for the ABCD form of an implicit MMPS system as in 3-8 from [16].

Proposition 5-1.2. *(Extended state MMPS input system) The expression for an implicit input signal can be represented by the following extended state form;*

$$\begin{aligned} u(k) &= F \otimes q(k) \\ q(k) &= H \otimes' w(k) \\ w(k) &= K_0 \cdot x(k-1) + K_1 \cdot x(k) + L_0 \cdot u(k-1) + L_1 \cdot u(k) + R_0 \cdot r(k-1) + R_1 \cdot r(k) \end{aligned} \quad (5\text{-}4)$$

Proof. By successively substituting $w(k)$ into $q(k)$ into $u(k)$, this result follows directly, as by successive substitution the implicit input function 5-3 is obtained. □

Since this input function is an MMPS function by itself, it can be represented by the following block diagram;

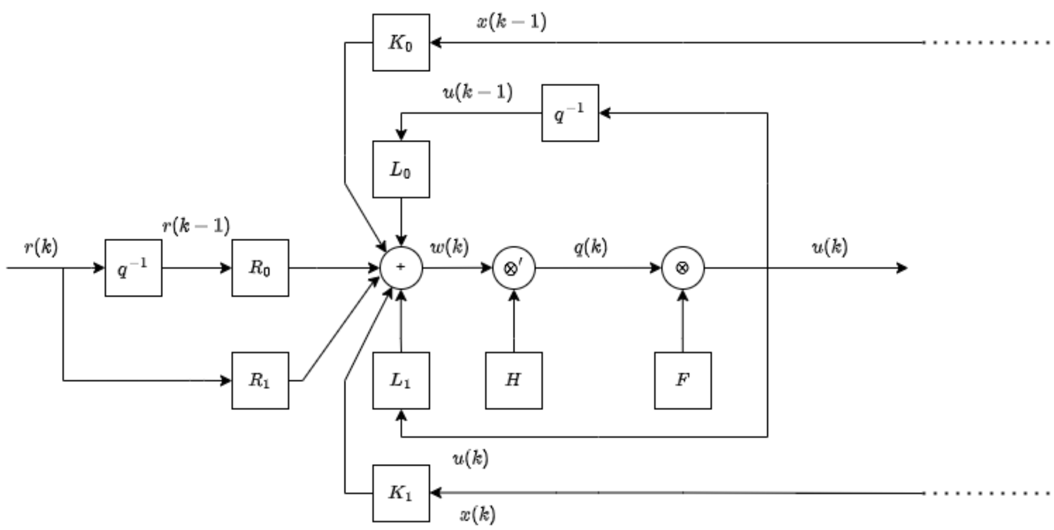


Figure 5-2: General implicit input function block diagram

This proposed block diagram can be connected to the implicit MMPS system which is given in Figure 5-1.

The applied input signal $u(k)$ can either be an external signal, in a scenario with open-loop control, or can be computed using a block-diagram structure of the form proposed in 5-2 in the case of closed-loop control. In the closed-loop control scenario, the current, and previous states $x(k)$ and $x(k-1)$ will be fed into the matrices K_0 and K_1 as shown in Figure 5-2. Open-loop control occurs when the input signal does not depend on the state of the system, i.e. there is no state-feedback control. Closed-loop control occurs when the input signal is an expression containing the state $x(k)$ of the system, either explicitly, or implicitly. In the next two sections, these two types of control, open-loop and closed-loop control, will be thoroughly examined in the context of applying them to implicit MMPS systems.

5-2 Open-Loop Control

Open-loop control can be visualized using the same block diagram given in 5-1, as the input is an external signal;

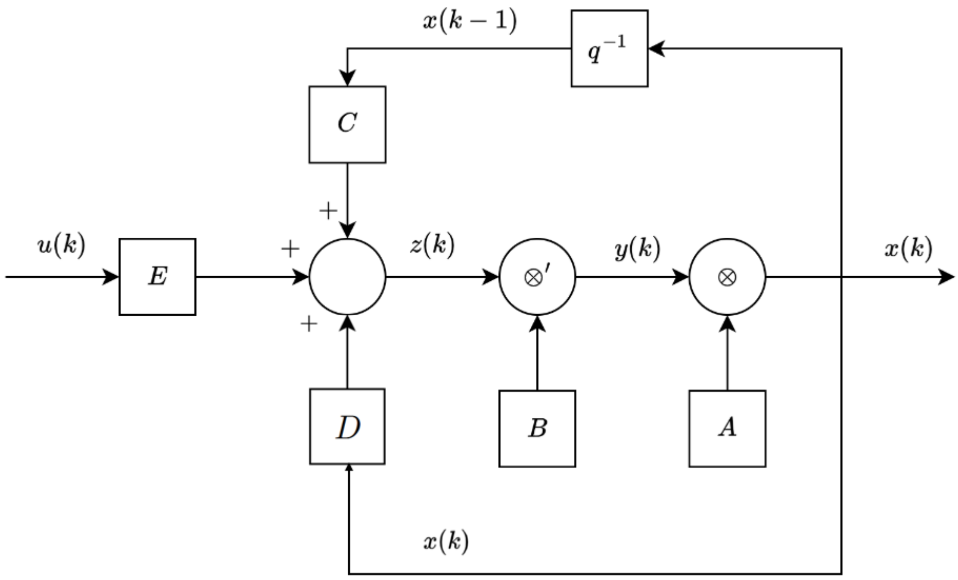


Figure 5-3: Open-loop control block diagram

This figure clearly shows there is no feedback of the state of the system into the control input signal. Therefore, in open-loop control, matrices K_0 and K_1 are $\mathbf{0}$, and the input signal can be defined by the following function;

$$u(k) = f(u(k), r(k)) \tag{5-5}$$

Theoretically, the broadest open-loop input signal matrix description can be given by;

$$u(k) = F \otimes (H \otimes' (L_0 \cdot u(k-1) + L_1 \cdot u(k) + R_0 \cdot r(k-1) + R_1 \cdot r(k)) \tag{5-6}$$

This equation is an implicit MMPS function in itself. However, when analyzing how the input signal $u(k)$ influences the implicit MMPS system, only matrix E is relevant. Assuming the values of $u(k)$ are finite, when examining the behaviour of the implicit MMPS system the input signal is applied to, it is only relevant to know the value of $u(k)$, no state-space expression as in 5-9 is necessary in open-loop control. Therefore, the ABCDE form of the open-loop controlled system with input signal $u(k)$ is given by the following equation;

$$\begin{bmatrix} x_{\mathrm{t}}(k) \\ x_{\mathrm{q}}(k) \end{bmatrix} = \underbrace{\begin{bmatrix} A_{\mathrm{t}} & \varepsilon \\ \varepsilon & A_{\mathrm{q}} \end{bmatrix}}_{A} \otimes \left(\underbrace{\begin{bmatrix} B_{\mathrm{t}} & \top \\ \top & B_{\mathrm{q}} \end{bmatrix}}_{B} \otimes' \left(\underbrace{\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_{\mathrm{t}}(k-1) \\ x_{\mathrm{q}}(k-1) \end{bmatrix} \right.\right.$$
$$\left.\left. + \underbrace{\begin{bmatrix} D_{11} D_{12} \\ D_{21} D_{22} \end{bmatrix}}_{D} \cdot \begin{bmatrix} x_t(k) \\ x_q(k) \end{bmatrix} + \underbrace{\begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix}}_{E} \cdot \begin{bmatrix} u_t(k) \\ u_q(k) \end{bmatrix} \right)\right) \tag{5-7}$$

A distinction is made between the temporal input signals $u_t(k)$ and the quantity input signals $u_q(k)$. Depending on where input signal is applied, the ABCDE form may require a slight modification. In the next subsection, two possible input strategies are elaborated on, and their respective ABCDE forms are presented.

5-2-1 Input Strategies

As mentioned in the previous section, depending on where the input signals are added in the system equations, the proposed ABCDE form might slightly change. Input strategy 1 elaborates on the scenario where the input signal is added in the scaling stage.

Input strategy 2 elaborates on the scenario's where the input signal is applied in the maximization or minimization stage.

Input Strategy 1

The first input strategy is in effect when the input signals are added in the scaling stage, for example;

$$x_1(k) = \max(x_2(k) + u_1(k), x(k-1) + 2) \tag{5-8}$$

Within this strategy, the input signals are added to an existing affine terms in $z(k)$. Therefore, when controlling an MMPS system where only Open-Loop (OL) control strategy 1 is applied, the size of the original matrices A, B, C and D do not need to change in order to accommodate for affine terms in $z(k)$ introduced by $u(k)$. The ABCDE form of this system is therefore given by equation 5-9, which is identical to the ABCDE form given in 5-1.

$$\begin{bmatrix} x_{\mathrm{t}}(k) \\ x_{\mathrm{q}}(k) \end{bmatrix} = \underbrace{\begin{bmatrix} A_{\mathrm{t}} & \varepsilon \\ \varepsilon & A_{\mathrm{q}} \end{bmatrix}}_{A} \otimes \left(\underbrace{\begin{bmatrix} B_{\mathrm{t}} & \top \\ \top & B_{\mathrm{q}} \end{bmatrix}}_{B} \otimes' \left(\underbrace{\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_{\mathrm{t}}(k-1) \\ x_{\mathrm{q}}(k-1) \end{bmatrix} \right.\right.$$
$$\left.\left. + \underbrace{\begin{bmatrix} D_{11} D_{12} \\ D_{21} D_{22} \end{bmatrix}}_{D} \cdot \begin{bmatrix} x_t(k) \\ x_q(k) \end{bmatrix} + \underbrace{\begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix}}_{E} \cdot \begin{bmatrix} u_t(k) \\ u_q(k) \end{bmatrix} \right)\right) \tag{5-9}$$

Input Strategy 2

The second input strategy provides a framework for implementing input signals implemented after the scaling step. This means an input signal could be a separate affine term in a maximization, or minimization operation. Therefore, adding such a input signal requires the dimension of matrices A, B, C and D to possibly grow. The input signal can either occur in a minimization operation, maximization operation, or in both. A separate ABCDE forms will be derived for each of these situations. When the input signal $u(k)$ is applied in the minimization step, the competition operation is in effect. When aiming to push the value of a state to a lower value, this can be an effective input strategy [4]. Because the minimum of some affine terms, and the value of an affine term including the input signal is taken. An example of a state in such a system can be described by the following expression [4];

$$x_2(k) = \min(x_1(k) + d_1, u(k)) \tag{5-10}$$

Proposition 5-2.1. *(OL input strategy* 2 *- minimization) The general ABCDE form of an OL controlled MMPS system with an input signal in the minimization operation can be described as follows;*

$$x(k) = A \otimes \left(\begin{bmatrix} B & B_u \end{bmatrix}' \otimes \left(\begin{bmatrix} C \\ 0 \end{bmatrix} \cdot x(k-1) + \begin{bmatrix} D \\ 0 \end{bmatrix} \cdot x(k) + \begin{bmatrix} 0 \\ E \end{bmatrix} \cdot u(k) \right) \right) \tag{5-11}$$

The added matrix B_u ensures the input signal is included correctly in the system equations. Augmenting matrix C, D and E with zeroes ensures the other states are unaffected.

Proof. This is an extension of the open-loop controlled strategy proposed in [4] by logically adding the implicit dynamics through matrix D. □

Similarly to adding an input signal in the minimization operation, an input can be added in the maximization operation, hereby the synchronization situation is in effect. When aiming to introduce a lower bound to the system, this can be an effective input strategy, since the maximum of some affine terms, and an affine term consisting of the input signal is taken. An example of a state in such a system can be described by the following expression [4];

$$x_2(k) = \max(x_1(k) + d_1, u(k)) \tag{5-12}$$

The ABCDE form of an implicit MMPS system with an input signal applied in the maximization step is given in the following definition;

Proposition 5-2.2. *(OL input strategy 2 - maximization) The general ABCDE form of a system with an input signal in a maximization operation can be described as follows;*

$$x(k) = \begin{bmatrix} A & A_u \end{bmatrix} \otimes \left(\begin{bmatrix} B & \top \\ \top & I_{\otimes'} \end{bmatrix} \otimes' \left(\begin{bmatrix} C \\ 0 \end{bmatrix} \cdot x(k-1) + \begin{bmatrix} D \\ 0 \end{bmatrix} \cdot x(k) + \begin{bmatrix} 0 \\ E \end{bmatrix} \cdot u(k) \right) \right) \tag{5-13}$$

here, matrix A_u implements the input signal in the maximization. Matrix $I_{\otimes'}$ is a min-plus identity matrix which directly feeds through the input. Matrices C and D are augmented with zeroes to accommodate for the change in size of $z(k)$ without affecting the existing system.

Proof. This is an extension of the open-loop controlled strategy proposed in [4] by logically adding the implicit dynamics through matrix D. □

Lastly, when implementation of input signals in the minimization operation, and the maximization operation occur in the same system, a combination of the proposed canonical forms proposed in 5-2.1 and 5-2.2 can be defined as follows;

Proposition 5-2.3. *(OL input strategy 2 - minimization and maximization) An implicit MMPS system to which input signals are applied, that occur in both minimization operations and maximization operations, can be described by the following canonical form;*

$$x(k) = \begin{bmatrix} A & A_u \end{bmatrix} \otimes \left(\begin{bmatrix} B & B_u \\ \top & I_{\otimes'} \end{bmatrix} \otimes' \left(\begin{bmatrix} C \\ 0 \end{bmatrix} \cdot x(k-1) + \begin{bmatrix} D \\ 0 \end{bmatrix} \cdot x(k) + \begin{bmatrix} 0 \\ E \end{bmatrix} \cdot u(k) \right) \right) \tag{5-14}$$

Here, matrix A_u ensures proper processing of the input signal in the maximization operation. Similarly, B_u ensures proper processing of the input signal in the minimization operation. Matrix $I_{\otimes'}$ ensures direct feedthrough of the input signals to the maximization operation. Matrices C, D and E are augmented with zeroes accordingly. In scenario's where this canonical form is needed, both elements of synchronization and competition occur.

Proof. This is an extension of the open-loop controlled strategy proposed in [4] by logically adding the implicit dynamics through matrix D, and combining propositions 5-2.1 and 5-2.2. □

This concludes the input strategies of the OL input signals into implicit MMPS systems. Next, the influence these input signals have on the time-invariance of OL controlled MMPS systems is discussed.

5-2-2 Time-Invariance of Open-Loop Controlled Systems

The conditions for time-invariance of the ABCDE canonical form corresponding to open-loop control can be derived. Recall the time-invariance conditions for the ABCD canonical form as in 3-10, and the definition for partial additive homogeneity 3-1.7. Notice that for determining the time-invariance conditions for the ABCD canonical form, the vector $p(k)$ as defined in 2-3.2 only contains the current and previous state of the system. However, for the ABCDE form corresponding to open-loop control, the vector $p(k)$ also contains input signal $u(k)$. Therefore, when deriving the conditions for partial-additive homogeneity, or equivalently, time-invariance, all temporal signals must be shifted, so the temporal input signal as well. The conditions for time-invariance of the ABCDE form 5-9 are very similar to those of the ABCD form, and defined by;

Theorem 5-2.1. *(Time-invariance of an open-loop controlled implicit MMPS system) An open-loop controlled implicit MMPS system described by the ABCDE form as per 5-1, is time-invariant when the following properties hold;*

$$\begin{aligned} \sum_{i \in \overline{n_t}} \begin{bmatrix} C_{11} & D_{11} & E_{11} \end{bmatrix}_{\ell i} &= 1, \forall \ell \in \overline{p_t} \\ \sum_{i \in \overline{n_q}} \begin{bmatrix} C_{21} & D_{21} & E_{21} \end{bmatrix}_{ti} &= 0, \forall t \in \overline{p_q} \end{aligned} \tag{5-15}$$

Proof. In 3-1-2, the proof for time-invariance, or equivalently, the MMPS system being additively homogeneous with respect to the temporal states was given, taken from [15].

For the ABCDE form, a similar approach in proving this property is taken. Let us shift the temporal states $x_t(k)$ by a finite amount $h\mathbf{1}$, and the temporal input signals $u_t(k)$ by the same finite amount $h\mathbf{1}$. The extended state system equations 3-1.6 are used as a framework for the following expressions;

$$\begin{aligned}
x_t(k) + h\mathbf{1} &= A_t \otimes (y_t(k) + h\mathbf{1}) \\
&= A_t \otimes y_t(k) + h\mathbf{1} \\
y_t(k) + h\mathbf{1} &= B_t \otimes' (z_t(k) + h\mathbf{1}) \\
&= B_t \otimes' z_t(k) + h\mathbf{1} \\
z_t(k) + h\mathbf{1} &= C_{11} \cdot (x_t(k-1) + h\mathbf{1}) + C_{12} x_q(k-1) + D_{11} \cdot (x_t(k) + h\mathbf{1}) + D_{12} \cdot x_q(k) \\
&\quad + E_{11} \cdot (u_t(k) + h\mathbf{1}) + E_{12} \cdot u_q(k)
\end{aligned} \tag{5-16}$$

From these equations, states $x_t(k)$ and $y_t(k)$ are naturally partially additive homogeneous as their extended state system equations are essentially Max-Min-Plus (MMP) expressions, therefore, they are naturally time-invariant [20]. State $z_t(k)$ is additively homogeneous when it holds that;

$$z_t(k) + h\mathbf{1} = C_{11} \cdot x_t(k-1) + C_{12} \cdot x_q(k-1) + D_{11} \cdot x_t(k) + D_{12} \cdot x_q(k) + E_{11} \cdot u_t(k) + E_{12} \cdot u_q(k) + h\mathbf{1} \tag{5-17}$$

This is in turn only true when it holds that;

$$\sum_{i \in \overline{n_t}} \begin{bmatrix} C_{11} & D_{11} & E_{11} \end{bmatrix}_{\ell i} = 1, \forall \ell \in \overline{p_t} \tag{5-18}$$

Furthermore, when the temporal states and temporal inputs are shifted by $h\mathbf{1}$, the quantity states should not be shifted at all. Therefore, the following equations can be derived;

$$\begin{aligned}
x_q(k) &= A_q \otimes y_q(k) \\
y_q(k) &= B_q \otimes' z_q(k) \\
z_q(k) &= C_{21} \cdot (x_t(k-1) + h\mathbf{1}) + C_{22} \cdot x_q(k-1) + D_{21} \cdot (x_t(k) + h\mathbf{1}) + D_{22} \cdot x_q(k) \\
&\quad + E_{21} \cdot (u_t(k) + h\mathbf{1}) + E_{22} \cdot u_q(k)
\end{aligned} \tag{5-19}$$

From these equations, only $z_q(k)$ has additional time-invariance conditions, as the expressions for $x_q(k)$ and $y_q(k)$ are MMP functions, which are always time-invariant [20]. For $z_q(k)$ it must hold that;

$$z_q(k) = C_{21} \cdot x_t(k-1) + C_{22} \cdot x_q(k-1) + D_{21} \cdot x_t(k) + D_{22} \cdot x_q(k) + E_{21} \cdot u_t(k) + E_{22} \cdot u_q(k) \tag{5-20}$$

Which leads to the following condition;

$$\sum_{i \in \overline{n_t}} \begin{bmatrix} C_{21} & D_{21} & E_{21} \end{bmatrix}_{ti} = 0, \forall t \in \overline{p_q} \tag{5-21}$$

The obtained results as in 5-18 and 5-21 are proof the conditions proposed in 5-15. □

Even though this theorem holds for all OL control strategies, a small distinction has to be made between input signals embedded in the system equations, and input signals applied to an already time-invariant system. In case input signals are applied to a time-invariant implicit MMPS system, the following conditions, as in 3-10 are known to have to hold;

$$\begin{aligned} \sum_{i\in\overline{n_\mathrm{t}}} \begin{bmatrix} C_{11} & D_{11} \end{bmatrix}_{\ell i} &= 1, \forall \ell \in \overline{p_\mathrm{t}} \\ \sum_{i\in\overline{n_\mathrm{q}}} \begin{bmatrix} C_{21} & D_{21} \end{bmatrix}_{ti} &= 0, \forall t \in \overline{p_\mathrm{q}} \end{aligned} \tag{5-22}$$

In order for the conditions proposed in 5-2.1 to hold as well, another condition naturally follows;

$$\sum_{i\in\overline{n_\mathrm{t}}} \begin{bmatrix} E_{11} \end{bmatrix}_{\ell i} = 0, \forall \ell \in \overline{p_\mathrm{t}} \tag{5-23}$$

Which must hold for in a scenario where 3-10 holds, and 5-15 has to hold as well.

5-2-3 Solvability of Open-loop Controlled Systems

In chapter 4, an elaborate study to the conditions for solvabiltiy of implicit MMPS systems was carried out. Using the results of this study, the solvability of OL controlled systems can be determined as well. Chapter 4 considered the solvability of implicit MMPS systems, represented in the ABCD form 3-1.5, whereas in OL controlled systems, matrix E is present as well. Even thought this matrix E is added to the system, and applies an external input signal to the system, its presence will never violate solvability of the OL controlled system. The objective of determining whether an implicit MMPS systems is solvable, is essentially figuring out whether an explicit mapping exist;

$$x(k) = f(x(k), x(k-1)) \Rightarrow x(k) = g(x(k-1)) \tag{5-24}$$

In OL controlled implicit MMPS systems, described by $x(k) = f_{OL}(p(k))$, $u(k)$ is an external input signal that does not depend on $x(k)$, and is always explicitly known, as it is an external input signal. The objective of determining solvability is still to figure out whether an explicit mapping exists, but the explicit mapping can be a function of $u(k)$, as $u(k)$ is assumed to be always explicitly known;

$$x(k) = f_{OL}(x(k), x(k-1), u(k)) \Rightarrow x(k) = g(x(k-1), u(k)) \tag{5-25}$$

As long as all modes of the OL controlled system can logically be reached, i.e. no mode is made redundant by the applied input, all solvable implicit MMPS systems that are open-loop controlled, will remain solvable.

5-3 Closed-Loop Control

In a Closed-Loop (CL) controlled system, the input signal $u(k)$ depends on the past state $x(k-1)$ and/or present state $x(k)$ of the system. Here, the most general input signal function applies;

$$u(k) = f(x(k), u(k), r(k)) \tag{5-26}$$

Input $u(k)$ and reference $r(k)$ can also be in this function, as the input may depend on itself, or the external reference signal. Using this general input function, a general structure of the closed-loop ABCDE, or technically, the ABCDR form of the system can be described;

$$\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \underbrace{\begin{bmatrix} A & \varepsilon \\ \varepsilon & F \end{bmatrix}}_{A_{es}} \otimes \left(\underbrace{\begin{bmatrix} B & \top \\ \top & H \end{bmatrix}}_{B_{es}} \otimes' \left(\underbrace{\begin{bmatrix} C & 0 \\ K_0 & L_0 \end{bmatrix}}_{C_{es}} \cdot \begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix} \right.\right.$$
$$\left.\left. + \underbrace{\begin{bmatrix} D & E \\ K_1 & L_1 \end{bmatrix}}_{D_{es}} \cdot \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & R_0 \end{bmatrix}}_{R_{0,es}} \cdot \begin{bmatrix} 0 \\ r(k-1) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & R_1 \end{bmatrix}}_{R_{1,es}} \cdot \begin{bmatrix} 0 \\ r(k) \end{bmatrix} \right)\right) \tag{5-27}$$

Let us introduce the block diagram of this closed-loop control structure in Figure 5-4.

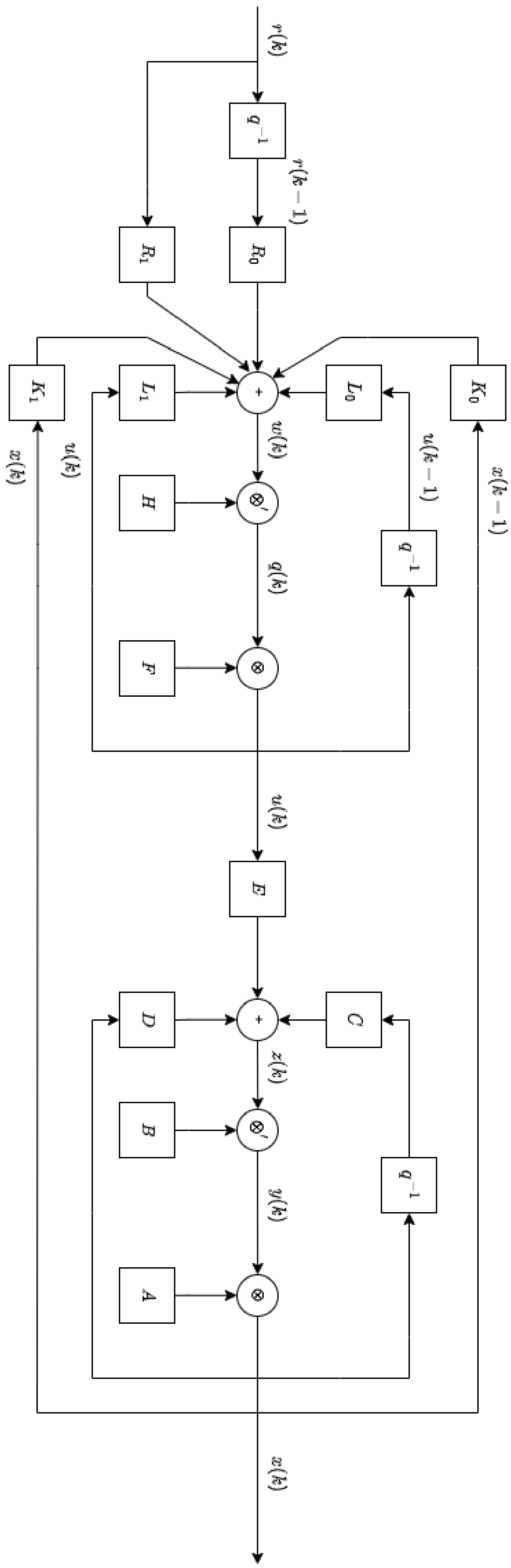


Figure 5-4: Block diagram of closed-loop input function

This form of the closed-loop controlled system can be simplified into the closed-loop extended ABCD form, which can be described by matrices A_{es}, B_{es}, C_{es}, D_{es}, $R_{0,es}$ and $R_{1,es}$;

$$\begin{aligned} A_{es} &= \begin{bmatrix} A & \varepsilon \\ \varepsilon & F \end{bmatrix}, \quad B_{es} = \begin{bmatrix} B & \top \\ \top & H \end{bmatrix}, \quad R_{0,es} = \begin{bmatrix} 0 & 0 \\ 0 & R_0 \end{bmatrix} \\ C_{es} &= \begin{bmatrix} C & 0 \\ K_0 & L_0 \end{bmatrix}, \quad D_{es} = \begin{bmatrix} D & E \\ K_1 & L_1 \end{bmatrix}, \quad R_{1,es} = \begin{bmatrix} 0 & 0 \\ 0 & R_1 \end{bmatrix} \end{aligned} \tag{5-28}$$

The extended state $x_{es}(k)$ can be described as $x_{es} = \begin{bmatrix} x(k)^\intercal & u(k)^\intercal \end{bmatrix}^\intercal$ Let us then introduce the simplified extended ABCDR form of a CL controlled implicit MMPS system as follows;

$$x_{es}(k) = A_{es} \otimes (B_{es} \otimes' (C_{es} \cdot x_{es}(k-1) + D_{es} \cdot x_{es}(k) + R_{0,es} \cdot r_{es}(k-1) + R_{1,es} \cdot r_{es}(k)) \tag{5-29}$$

An interesting observation is that the input signal is no longer an external signal, but rather integrated as an extension of the state $x(k)$. The only external input to this system is the reference signal $r(k)$ or $r(k-1)$. In case there is no reference signal applied to the system, the autonomous closed loop system can be given by;

$$\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \underbrace{\begin{bmatrix} A & \varepsilon \\ \varepsilon & F \end{bmatrix}}_{A_{es}} \otimes \left(\underbrace{\begin{bmatrix} B & \top \\ \top & H \end{bmatrix}}_{B_{es}} \otimes' \left(\underbrace{\begin{bmatrix} C & 0 \\ K_0 & L_0 \end{bmatrix}}_{C_{es}} \cdot \begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix} + \underbrace{\begin{bmatrix} D & E \\ K_1 & L_1 \end{bmatrix}}_{D_{es}} \cdot \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} \right) \right) \tag{5-30}$$

Or equivalently;

$$x_{es}(k) = A_{es} \otimes (B_{es} \otimes' (C_{es} \cdot x_{es}(k-1) + D_{es} \cdot x_{es}(k)) \tag{5-31}$$

Still, it is worth deriving some different structures of the extended ABCDR form for different input strategies, depending on where input $u(k)$ is applied. Similar to the OL input strategies, the CL input strategies explained are the scenario where the input signal is applied within the scaling stage, and the scenario where the input signal is applied in the maximization/minimization stage.

5-3-1 Input Strategies

Similar adjustments to the ABCDR form can be made as was done in Section 5-2 by analyzing the differences in applying the input signal in different stages. These stages are again, during the scaling stage, and the maximization/minimization stage.

Input Strategy 1

When the input signals are applied only in the scaling phase, the ABCDR form does not have to change. The dimensions of the matrices A_{es}, B_{es}, C_{es}, D_{es}, $R_{0,es}$ and $R_{1,es}$ remain the same, and the ABCDR form is given by;

$$\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \underbrace{\begin{bmatrix} A & \varepsilon \\ \varepsilon & F \end{bmatrix}}_{A_{es}} \otimes \left(\underbrace{\begin{bmatrix} B & \top \\ \top & H \end{bmatrix}}_{B_{es}} \otimes' \left(\underbrace{\begin{bmatrix} C & 0 \\ K_0 & L_0 \end{bmatrix}}_{C_{es}} \cdot \begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix} \right.\right.$$
$$\left.\left. + \underbrace{\begin{bmatrix} D & E \\ K_1 & L_1 \end{bmatrix}}_{D_{es}} \cdot \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & R_0 \end{bmatrix}}_{R_{0,es}} \cdot \begin{bmatrix} 0 \\ r(k-1) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & R_1 \end{bmatrix}}_{R_{1,es}} \cdot \begin{bmatrix} 0 \\ r(k) \end{bmatrix} \right)\right) \quad (5\text{-}32)$$

Input Strategy 2

The necessary structure adjustments for CL control of an implicit MMPS system where the input signals are added in the minimization stage, maximization stage, or both, are given by the following definitions, respectively;

Proposition 5-3.1. *(CL input strategy 2 - minimization)*

$$\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \underbrace{\begin{bmatrix} A & \varepsilon \\ \varepsilon & F \end{bmatrix}}_{A_{es}} \otimes \Big(\underbrace{\begin{bmatrix} B & B_u & \top \\ \top & \top & H \end{bmatrix}}_{B_{es}} \otimes' \Big(\underbrace{\begin{bmatrix} C & 0 \\ 0 & 0 \\ K_0 & L_0 \end{bmatrix}}_{C_{es}} \begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix} +$$
$$\underbrace{\begin{bmatrix} D & 0 \\ 0 & E \\ K_1 & L_1 \end{bmatrix}}_{D_{es}} \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & R_0 \end{bmatrix}}_{R_{0,es}} \cdot \begin{bmatrix} 0 \\ r(k-1) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & R_1 \end{bmatrix}}_{R_{1,es}} \cdot \begin{bmatrix} 0 \\ r(k) \end{bmatrix} \Big)\Big) \quad (5\text{-}33)$$

Proof. This is an extension of the open-loop controlled strategy proposed in [4] by logically adding the implicit dynamics matrix D, and rearranging the system matrices such that the ABCDR matrix could be obtained. □

The function of matrix B_u is to ensure the input signal is correctly applied in the system equations, identical to the purpose of matrix B_u in 5-2.1 and 5-2.3.

Proposition 5-3.2. *(CL input strategy 2 - maximization) When applying CL input signals in the maximization stage, some adjustments to the ABCDR form have to be made to accommodate for correct processing of these input signals;*

$$\begin{aligned}\begin{bmatrix}x(k)\\u(k)\end{bmatrix} = \underbrace{\begin{bmatrix}A & A_u & \varepsilon\\ \varepsilon & \varepsilon & F\end{bmatrix}}_{A_{es}} \otimes \Bigg(\underbrace{\begin{bmatrix}B & \top & \top\\ \top & I_{\otimes'} & \top\\ \top & \top & H\end{bmatrix}}_{B_{es}} \otimes' \Bigg(\underbrace{\begin{bmatrix}C & 0\\ 0 & 0\\ K_0 & L_0\end{bmatrix}}_{C_{es}} \begin{bmatrix}x(k-1)\\u(k-1)\end{bmatrix} + \\ \underbrace{\begin{bmatrix}D & 0\\ 0 & E\\ K_1 & L_1\end{bmatrix}}_{D_{es}} \begin{bmatrix}x(k)\\u(k)\end{bmatrix} + \underbrace{\begin{bmatrix}0 & 0\\ 0 & 0\\ 0 & R_0\end{bmatrix}}_{R_{0,es}} \cdot \begin{bmatrix}0\\r(k-1)\end{bmatrix} + \underbrace{\begin{bmatrix}0 & 0\\ 0 & 0\\ 0 & R_1\end{bmatrix}}_{R_{1,es}} \cdot \begin{bmatrix}0\\r(k)\end{bmatrix}\Bigg)\Bigg)\end{aligned} \tag{5-34}$$

Identical to matrix A_u in 5-2.2 and 5-2.3, matrix A_u implements the input signal in the maximization. Matrix $I_{\otimes'}$ is a min-plus identity matrix which directly feeds through the input. Matrices C_{es} and D_{es} are augmented with zeroes in the appropriate places to accommodate for the change in size of $z(k)$ without affecting the existing system.

Proof. This is an extension of the open-loop controlled strategy proposed in [4] by logically adding the implicit dynamics matrix D, and rearranging the system matrices such that the ABCDR matrix could be obtained. □

Lastly, combining the previous two propositions, it is possible for a system to have input signals applied in both the maximization, and minimization step. Propositions 5-3.1 and 5-3.2 can be combined into the following proposition;

Proposition 5-3.3. *(CL input strategy 2 - minimization and maximization) Combining the structural adjustments to the ABCDR form proposed in 5-3.1 and 5-2.2, allowing for CL input signals to be applied in both the maximization, and minimization stage, the following ABCDR form is obtained;*

$$\begin{aligned}\begin{bmatrix}x(k)\\u(k)\end{bmatrix} = \underbrace{\begin{bmatrix}A & A_u & \varepsilon\\ \varepsilon & \varepsilon & F\end{bmatrix}}_{A_{es}} \otimes \Bigg(\underbrace{\begin{bmatrix}B & B_u & \top\\ \top & I_{\otimes'} & \top\\ \top & \top & H\end{bmatrix}}_{B_{es}} \otimes' \Bigg(\underbrace{\begin{bmatrix}C & 0\\ 0 & 0\\ K_0 & L_0\end{bmatrix}}_{C_{es}} \begin{bmatrix}x(k-1)\\u(k-1)\end{bmatrix} + \\ \underbrace{\begin{bmatrix}D & 0\\ 0 & E\\ K_1 & L_1\end{bmatrix}}_{D_{es}} \begin{bmatrix}x(k)\\u(k)\end{bmatrix} + \underbrace{\begin{bmatrix}0 & 0\\ 0 & 0\\ 0 & R_0\end{bmatrix}}_{R_{0,es}} \cdot \begin{bmatrix}0\\r(k-1)\end{bmatrix} + \underbrace{\begin{bmatrix}0 & 0\\ 0 & 0\\ 0 & R_1\end{bmatrix}}_{R_{1,es}} \cdot \begin{bmatrix}0\\r(k)\end{bmatrix}\Bigg)\Bigg)\end{aligned} \tag{5-35}$$

Proof. This is an extension of the open-loop controlled strategy proposed in [4] by logically adding the implicit dynamics matrix D, and rearranging the system matrices such that the ABCDR matrix could be obtained. □

5-3-2 Time-Invariance of Closed-Loop Controlled Systems

Similar to OL control, time-invariance conditions can be derived for CL controlled systems. The approach for deriving these conditions is similar to how the open-loop conditions were derived. In contrast to the open-loop scenario, time-invariance will have to be proven for extended state $x_{es}(k)$, consisting of both $x(k)$ and $u(k)$. Similarly to the open-loop controlled system, the vector $p(k)$ in vector-valued description of this system as per 2-3.2 now contains input signals $u(k)$, but reference signals $r(k)$ as well. When deriving the conditions for partial additive homogeneity or equivalently, time-invariance, these temporal input and reference signals will have to be shifted as well. The following theorem proposes the time-invariance conditions for closed-loop controlled implicit MMPS systems, which will thereafter be proven.

Theorem 5-3.1. *(Time-invariance of closed-loop controlled implicit MMPS systems) A closed-loop controlled implicit MMPS system as per 5-27 can be considered time-invariant when the following conditions are satisfied;*

$$\begin{aligned} \sum_{i \in \bar{n}_t + \bar{u}_t} W_{t,\ell i} &= 1, \forall \ell \in \bar{p}_t \\ \sum_{i \in \bar{n}_t + \bar{u}_t} W_{q,\ell i} &= 0, \forall \ell \in \bar{p}_t \end{aligned} \tag{5-36}$$

Where;

$$\begin{aligned} W_t &= \begin{bmatrix} C_{11} & 0 & D_{11} & E_{11} & 0 & 0 \\ K_{0,11} & L_{0,11} & K_{1,11} & L_{1,11} & R_{0,11} & R_{1,11} \end{bmatrix} \\ W_q &= \begin{bmatrix} C_{21} & 0 & D_{21} & E_{21} & 0 & 0 \\ K_{0,21} & L_{0,21} & K_{1,21} & L_{1,21} & R_{0,21} & R_{1,21} \end{bmatrix} \end{aligned} \tag{5-37}$$

Proof. Essentially, time-invariance has to be proven for the following CL implicit system;

$$x_{es}(k) = A_{es} \otimes (B_{es} \otimes' (C_{es} \cdot x_{es}(k-1) + D_{es} \cdot x_{es}(k) + R_{0,es} \cdot r_{es}(k-1) + R_{1,es} \cdot r(k)) \tag{5-38}$$

However, state $x_{es}(k)$ is the extended state composed off of stacking $x(k)$ and $u(k)$, and as shown in 5-30, all system matrices consist of multiple submatrices. The proposed time-invariance condition applied conditions to some of these submatrices. Therefore, let us define the system matrices as follows, such that time-invariance conditions can be derived for the whole system by considering the characteristics of the submatrices;

$$x_{es}(k) = \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} x_t(k) \\ x_q(k) \\ u_t(k) \\ u_q(k) \end{bmatrix} \tag{5-39}$$

$$r_{es}(k) = \begin{bmatrix} 0 \\ r(k) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ r_t(k) \\ r_q(k) \end{bmatrix} \tag{5-40}$$

$$C_{es} = \begin{bmatrix} C & 0 \\ K_0 & L_0 \end{bmatrix} = \left[\begin{array}{cc|cc} C_{11} & C_{12} & 0 & 0 \\ C_{21} & C_{22} & 0 & 0 \\ \hline K_{0,11} & K_{0,12} & L_{0,11} & L_{0,12} \\ K_{0,21} & K_{0,22} & L_{0,21} & L_{0,22} \end{array}\right] \tag{5-41}$$

$$D_{es} = \begin{bmatrix} D & E \\ K_1 & L_1 \end{bmatrix} = \left[\begin{array}{cc|cc} D_{11} & D_{12} & E_{11} & E_{12} \\ D_{21} & D_{22} & E_{21} & E_{22} \\ \hline K_{1,11} & K_{1,12} & L_{1,11} & L_{1,12} \\ K_{1,21} & K_{1,22} & L_{1,21} & L_{1,22} \end{array}\right] \tag{5-42}$$

$$R_{0,es} = \begin{bmatrix} 0 & 0 \\ 0 & R_0 \end{bmatrix} = \left[\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & R_{0,11} & R_{0,12} \\ 0 & 0 & R_{0,21} & R_{0,22} \end{array}\right] \tag{5-43}$$

$$R_{1,es} = \begin{bmatrix} 0 & 0 \\ 0 & R_1 \end{bmatrix} = \left[\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & R_{1,11} & R_{1,12} \\ 0 & 0 & R_{1,21} & R_{1,22} \end{array}\right] \tag{5-44}$$

By shifting the temporal states $x_t(k)$, $u_t(k)$ and temporal reference signals $r_{x,t}(k)$ and $r_{u,t}(k)$ with $h\mathbf{1}_x$ and $h\mathbf{1}_u$ respectively, the following equations should hold for the time-invariance;

$$\begin{aligned} x_{es,t}(k) + h\mathbf{1} &= A_{es,t} \otimes (y_{es,t}(k) + h\mathbf{1}) \\ &= A_{es,t} \otimes y_{es,t}(k) + h\mathbf{1} \\ y_{es,t}(k) + h\mathbf{1} &= B_{es,t} \otimes' (z_{es,t}(k) + h\mathbf{1}) \\ &= B_{es,t} \otimes' z_{es,t}(k) + h\mathbf{1} \end{aligned} \tag{5-45}$$

$$\begin{aligned} z_{es,t}(k) + h\mathbf{1} = &\begin{bmatrix} C_{11} & 0 \\ K_{0,11} & L_{0,11} \end{bmatrix} \cdot \begin{bmatrix} x_t(k-1) + h\mathbf{1}_x \\ u_t(k-1) + h\mathbf{1}_u \end{bmatrix} + \begin{bmatrix} C_{12} & 0 \\ K_{0,12} & L_{0,12} \end{bmatrix} \cdot \begin{bmatrix} x_q(k-1) \\ u_q(k-1) \end{bmatrix} \\ &+ \begin{bmatrix} D_{11} & E_{11} \\ K_{1,11} & L_{1,11} \end{bmatrix} \cdot \begin{bmatrix} x_t(k) + h\mathbf{1}_x \\ u_t(k) + h\mathbf{1}_u \end{bmatrix} + \begin{bmatrix} D_{12} & E_{12} \\ K_{1,12} & L_{1,12} \end{bmatrix} \cdot \begin{bmatrix} x_q(k) \\ u_q(k) \end{bmatrix} \\ &+ \begin{bmatrix} 0 & 0 \\ 0 & R_{0,11} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ r_t(k-1) + h\mathbf{1}_u \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & R_{0,12} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ r_q(k-1) \end{bmatrix} \\ &+ \begin{bmatrix} 0 & 0 \\ 0 & R_{1,11} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ r_t(k) + h\mathbf{1}_u \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & R_{1,12} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ r_{u,q}(k) \end{bmatrix} \end{aligned} \tag{5-46}$$

By grouping the temporal (sub) states $x_t(k)$ and $u_t(k)$, and the quantity states $x_q(k)$ and $u_q(k)$, the submatrices can be grouped more efficiently. Since the expressions for $x_{es,t}$ and $y_{es,t}$ are MMP expressions, they are already time-invariant [20]. In order for 5-46 to hold, it must hold that the rows of the following matrix add up to 1, providing a condition for the time-invariance of $z_{es,t}$;

$$\sum_{i \in \bar{n}_t + \bar{u}_t} \begin{bmatrix} C_{11} & 0 & D_{11} & E_{11} & 0 & 0 \\ K_{0,11} & L_{0,11} & K_{1,11} & L_{1,11} & R_{0,11} & R_{1,11} \end{bmatrix}_{\ell i} = 1, \forall \ell \in \bar{p}_t \tag{5-47}$$

Which is equivalent to;

$$\begin{aligned} &\sum_{i \in \bar{n}_t + \bar{u}_t} W_{t,\ell i} = 1, \forall \ell \in \bar{p}_t \\ &W_t = \begin{bmatrix} C_{11} & 0 & D_{11} & E_{11} & 0 & 0 \\ K_{0,11} & L_{0,11} & K_{1,11} & L_{1,11} & R_{0,11} & R_{1,11} \end{bmatrix} \end{aligned} \tag{5-48}$$

Which is identical to the proposed condition. Now all is left is to prove the proposed time-invariance condition for the quantity states, which is done by identifying the conditions under which these equations hold;

$$\begin{aligned} x_{es,q}(k) &= A_{es,q} \otimes y_{es,q}(k) \\ y_{es,q}(k) &= B_{es,q} \otimes' z_{es,q}(k) \end{aligned} \tag{5-49}$$

$$\begin{aligned} z_{es,q}(k) = &\begin{bmatrix} C_{21} & 0 \\ K_{0,21} & L_{0,21} \end{bmatrix} \cdot \begin{bmatrix} x_t(k-1) + h\mathbf{1}_x \\ u_t(k-1) + h\mathbf{1}_u \end{bmatrix} + \begin{bmatrix} C_{22} & 0 \\ K_{0,22} & L_{0,22} \end{bmatrix} \cdot \begin{bmatrix} x_q(k-1) \\ u_q(k-1) \end{bmatrix} \\ &+ \begin{bmatrix} D_{21} & E_{21} \\ K_{1,21} & L_{1,21} \end{bmatrix} \cdot \begin{bmatrix} x_t(k) + h\mathbf{1}_x \\ u_t(k) + h\mathbf{1}_u \end{bmatrix} + \begin{bmatrix} D_{22} & E_{22} \\ K_{1,22} & L_{1,22} \end{bmatrix} \cdot \begin{bmatrix} x_q(k) \\ u_q(k) \end{bmatrix} \\ &+ \begin{bmatrix} 0 & 0 \\ 0 & R_{0,21} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ r_t(k-1) + h\mathbf{1}_u \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & R_{0,22} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ r_q(k-1) \end{bmatrix} \\ &+ \begin{bmatrix} 0 & 0 \\ 0 & R_{1,21} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ r_t(k) + h\mathbf{1}_u \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & R_{1,22} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ r_q(k) \end{bmatrix} \end{aligned} \tag{5-50}$$

As the expressions for $x_{es,q}(k)$ and $y_{es,q}$ are MMP expressions, which are always time invariant [20], only condition for the time-invariance of $z_{es,q}$ will have to be derived. It follows that this equation holds under the following condition;

$$\sum_{i \in \bar{n}_t + \bar{u}_t} \begin{bmatrix} C_{21} & 0 & D_{21} & E_{21} & 0 & 0 \\ K_{0,21} & L_{0,21} & K_{1,21} & L_{1,21} & R_{0,21} & R_{1,21} \end{bmatrix}_{\ell i} = 0, \forall \ell \in \bar{p}_t \tag{5-51}$$

Which can be rewritten as;

$$\begin{aligned} &\sum_{i \in \bar{n}_t + \bar{u}_t} W_{q,\ell i} = 0, \forall \ell \in \bar{p}_t \\ &W_q = \begin{bmatrix} C_{21} & 0 & D_{21} & E_{21} & 0 & 0 \\ K_{0,21} & L_{0,21} & K_{1,21} & L_{1,21} & R_{0,21} & R_{1,21} \end{bmatrix} \end{aligned} \tag{5-52}$$

Which again, is identical to the proposed condition. □

Similarly to OL controlled systems, depending on where the input signal is applied, some additional conditions might apply. However, as long as the conditions in Theorem 5-3.1 is satisfied, the CL controlled implicit MMPS system will be time-invariant.

5-3-3 Solvability of Closed-Loop Controlled Systems

Contrary to OL controlled implicit MMPS systems, CL controlled implicit MMPS systems are not always solvable. Input signal $u(k)$ is a function of state $x(k)$, and can therefore violate solvability as it may not always be explicitly know. Essentially, new states are added to the system, and these states can logically violate solvability of the existing system.
For OL controlled systems of the ABCDR form as given below, solvability could immediately be inferred;

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k) + E \cdot u(k))) \tag{5-53}$$

In CL controlled systems, a system of similar structure can be obtained. The extended state system can be given as;

$$x_{es}(k) = A_{es} \otimes (B_{es} \otimes' (C_{es} \cdot x_{es}(k-1) + D_{es} \cdot x_{es}(k) + R_{0,es} \cdot r_{es}(k-1) + R_{1,es} \cdot r_{es}(k)) \tag{5-54}$$

This system also has an ABCDR form, but the uncontrolled implicit MMPS system is incorporated in the extended state matrices. However, the same solvability conditions must apply to this extended ABCDR form. Therefore, it can already be concluded that the reference matrix R_{es} and the known reference signal $r(k)$, which is the only external input signal, will never violate solvability of the extended ABCDR system. Therefore, solvability will have to be concluded for the following ABCD form;

$$x_{es}(k) = A_{es} \otimes (B_{es} \otimes' (C_{es} \cdot x_{es}(k-1) + D_{es} \cdot x_{es}(k)) \tag{5-55}$$

The solvability of this extended ABCD form system can easily be analyzed using the results presented in Chapter 4.

Chapter 6

Augmenting and Analyzing the Urban Railway System

The main contributions of this chapter are the augmentation, and subsequent analysis of the existing Urban Railway System (URS) described in [18], recalled in Section 6-1. The sections that follow contain the actual research carried out, whereas Section 6-1 is merely a recollection of the work presented in [18]. The system model is extended to include complex passenger flows using a flow matrix, and adding two quantity states, which is described in Section 6-2. Sections 6-3, 6-4 and 6-5 propose the ABCD form of the augmented model, analyze its solvability using the theory proposed in Chapter 4, and describe the initialization of the system. Section 6-6 computes and analyzes the properties of the fixed-points of the AURS, after which it is simulated according to a uniform timetable in Section 6-7. The chapter concludes with an analysis of the stability and maximal invariant set of the AURS in Sections 6-8 and 6-9.

6-1 Current Mathematical Model

This section elaborates illustrates the theoretical case study research in [18], and considers an urban railway line with passengers embarking, and disembarking at every station. Consider an urban railway line as shown in Figure 6-1

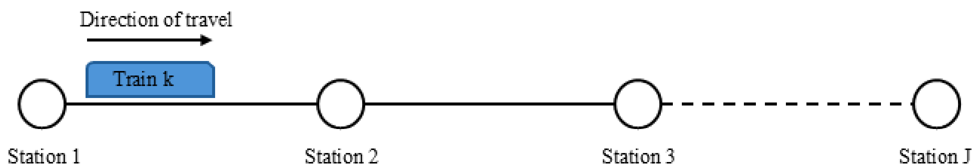


Figure 6-1: Urban railway system

This system can be described by four states, two of which are quantity states, and two of which are temporal states. The arrival, and departure time of train k at station j is denoted by temporal states $a_j(k)$ and $d_j(k)$ respectively. Quantity state $\rho_j(k)$ is the number of passengers in train k when leaving station j. The number of passengers at station j when train k is leaving the station is denoted by quantity state $\sigma_j(k)$. Multiple assumptions will be done in order to properly model this system, which are given below;

- Every train has a maximum capacity of $\rho_{\max}$
- Running times $\tau_{r,j}$ between station $j-1$ and station j are fixed
- The number of passengers entering the platform at station j per time unit is e_j
- The number of passengers boarding the train per unit of time is a fixed value b for the boarding rate
- The number of passengers that can disembark the train per time unit is a fixed value denoted by f
- It is assumed that the boarding rate is larger than the arrival rate, i.e. $b > e_j$, if this would not be the case, trains would never leave the station unless they were full, in which case the system overflows, and is unstable
- The number of passengers leaving train k at station j is always a fixed fraction β_j of the number of passengers in train j at the moment train j arrives
- Passengers that have disembarked the train will leave the station immediately

Firstly, the function to determine the arrival time is modeled. The arrival time of train k at station j is the maximum of the departure time at station $j-1$ plus the running time τ_r, and the departure time of train $k-1$ at station j plus the headway time. The trains $k = 1, ..., K$ depart from station 1 with a headway interval of τ_0. Mathematically, this is described as;

$$a_j(k) = \max\left(d_{j-1}(k) + \tau_{r,j}, d_j(k-1) + \tau_H\right) \tag{6-1}$$

The dwell time is defined as the sum of the time for disembarking, $\tau_{d,j}(k)$, and the time for boarding the train, $\tau_{b,j}(k)$. If it is assumed that there is no additional waiting time for the train, the departure time of train k can be modeled as;

$$d_j(k) = a_j(k) + \tau_{d,j}(k) + \tau_{b,j}(k) \tag{6-2}$$

The number of passengers in train k when leaving station j can be modeled as the number of passengers in train k when it left station $j-1$, subtracting the passengers that disembarked at station j, adding the passengers that boarded at station j. Which mathematically can be described as;

$$\rho_j(k) = \rho_{j-1}(k) - f\tau_{d,j}(k) + b\tau_{b,j}(k) \tag{6-3}$$

The number of passengers that are standing on the platform at station j when train k leaves, can be modeled as the number of passengers still on the platform when train $k-1$ left, plus the number of passengers that entered the station between the departures of train $k-1$ and train k. Subtracting the amount of passengers boarding train k at station j yields the following expression;

$$\sigma_j(k) = \sigma_j(k-1) + e_j\left(d_j(k) - d_j(k-1)\right) - e_j\tau_{b,j}(k) \tag{6-4}$$

Furthermore, the time that it takes to disembark $\tau_{d,j}$ is proportional to the number of passengers disembarking the train;

$$\tau_{d,j}(k) = \frac{\beta_j}{f}\rho_{j-1}(k) \tag{6-5}$$

The boarding time can be modeled as the dwell time, subtracting the arrival time, and subtracting the time it takes to disembark;

$$\begin{aligned} \tau_{b,j}(k) &= d_j(k) - a_j(k) - \tau_{d,j}(k) \\ &= d_j(k) - a_j(k) - \frac{\beta_j}{f}\rho_{j-1}(k) \end{aligned} \tag{6-6}$$

Calculating the equation to calculate the departure time becomes more complex taking into account two scenarios. Because the train departs from station j, either if all passengers on the station have boarded the train, or if the train is full, and no more passengers can board the train k. The first scenario occurs if $\rho_j(k) \leq \rho_{\max}$, and the second scenario occurs when $\rho_j(k) = \rho_{\max}$. In the case that $\rho_j(k) \leq \rho_{\max}$, the number of passengers that board train k, $b\left(d_j(k) - a_j(k) - \frac{\beta_j}{f}\rho_{j-1}(k)\right)$, is equal to the number of passengers that want to board train k, which is $\sigma_j(k-1) + e_j\left(d_j(k) - d_j(k-1)\right)$. Therefrom the following expression follows;

$$b\left(d_j(k) - a_j(k) - \frac{\beta_j}{f}\rho_{j-1}(k)\right) = \sigma_j(k-1) + e_j\left(d_j(k) - d_j(k-1)\right) \tag{6-7}$$

From this expression, $d_j(k)$ can be written explicitly, deriving a new expression for departure time, in the case that $\rho_j(k) \leq \rho_{\max}$;

$$d_j(k) = \mu_1 a_j(k) + \mu_2\rho_{j-1}(k) + \mu_3\sigma_j(k-1) + (1-\mu_1)\,d_j(k-1) \tag{6-8}$$

Where, $\mu_1 = \frac{b}{b-e_j}$, $\mu_2 = \frac{b}{b-e_j}\frac{\beta_j}{f}$ and $\mu_3 = \frac{1}{b-e_j}$, for simplicity.

For the second scenario, where $\rho_j(k) = \rho_{\max}$, the train leaves station k as soon as the train is full. Therefore, the number of passengers that remained seated in train k at station j, and the number of passengers who embarked train k at station j added together is equal to the maximum capacity, $\rho_{\max}$ yielding the following expression;

$$(1-\beta_j)\rho_{j-1}(k) + b(d_j(k) - a_j(k) - \frac{\beta_j}{f}\rho_{j-1}(k)) = \rho_{\max} \tag{6-9}$$

Writing $d_j(k)$ explicitly leads to the following expression;

$$d_j(k) = \gamma_1 + a_j(k) + \gamma_2\rho_{j-1}(k) \tag{6-10}$$

Where $\gamma_1 = \frac{1}{b}\rho_{\max}$ and $\gamma_2 = \frac{\beta_j}{f} - \frac{1-\beta_j}{b}$ By combining the two scenarios, and letting the actual departure time be the minimum of the two, the following expression can be calculated;

$$d_j(k) = \min\left(\mu_1 a_j(k) + \mu_2\rho_{j-1}(k) + \mu_3\sigma_j(k-1) + (1-\mu_1)\,d_j(k-1), \gamma_1 + a_j(k) + \gamma_2\rho_{j-1}(k)\right) \tag{6-11}$$

Now, the final system equations for $j > 1$ and $k > 0$ can be derived;

$$\begin{aligned}
a_j(k) &= \max\left(d_{j-1}(k) + \tau_{r,j}, d_j(k-1) + \tau_H\right) \\
d_j(k) &= \min\left(\mu_1 a_j(k) + \mu_2 \rho_{j-1}(k) + \mu_3 \sigma_j(k-1) + (1-\mu_1)\, d_j(k-1), \gamma_1 + a_j(k) + \gamma_2 \rho_{j-1}(k)\right) \\
\rho_j(k) &= (1-\beta_j)\,\rho_{j-1}(k) + b\left(d_j(k) - a_j(k) - \frac{\beta_j}{f}\rho_{j-1}(k)\right) \\
\sigma_j(k) &= \sigma_j(k-1) + e_j\left(d_j(k) - d_j(k-1)\right) - b\left(d_j(k) - a_j(k) - \frac{\beta_j}{f}\rho_{j-1}(k)\right)
\end{aligned} \tag{6-12}$$

6-2 Augmenting the Mathematical Model

In the mathematical model for the Urban Railway System (URS) as described in 6-12, some assumptions are an abstract approximation of reality. The existing mathematical model can be used as a baseline when aiming to develop model that more closely resembles reality. Currently, the number of passengers entering station j per time unit, e_j, is taken as a constant value, or rather, a constant flow rate. This flow rate is said to be the same for each station. Furthermore, the number of people disembarking train k at station j is taken as a fixed fraction β_j of the number of passengers on train k when it leaves station $j-1$. For the augmented model, the aim is to accommodate for the origin and the destination of traveling passengers, and to allow for different flow rates $e_{j,k}$ for each station, and each train. A vast range of modifications to the system description is required to enable these additions, and model the Augmented Urban Railway System (AURS).

6-2-1 Introducing the Augmentation

1. Rather than having a fixed fraction β_j of passengers, a matrix ζ will be introduced. This matrix described how passengers traveling on this railway line pass through the system. Each entry will represent a fraction of passengers traveling from station i to station j.

2. A state $\beta_j(k)$ will be introduced, representing the number of passengers disembarking train k at station j. Thus, the number of passengers leaving the train k at the station j is not just a fixed fraction of the number of passengers already in the train, but a combination of all passengers who got on the train k at any previous station, whose destination is station j. The state $\beta_j(k)$ is a quantity state.

3. A second state, quantity state $\Delta_j(k)$ will be introduced, representing the number of passengers embarking train k at station j. State $\Delta_j(k)$ is a quantity state.

4. The matrix e will be introduced, which contains the rate at which passengers enter each station, and will replace fixed flow rate e_j.

5. The four existing states are to be re-derived in order to accomodate for the two newly added states.

The next section will provide a structured derivation of the augmented state-space description.

6-2-2 Deriving the Augmented Model

The most straightforward augmentation to the system would be redefining fixed flow rate e_j, which requires the introduction of matrix e;

$$e = \begin{bmatrix} e_{1,1} & e_{1,2} & \cdots & e_{1,K} \\ e_{2,1} & e_{2,2} & \cdots & e_{2,K} \\ \vdots & \ddots & \ddots & \vdots \\ e_{J,1} & e_{J,2} & \cdots & e_{J,K} \end{bmatrix} \tag{6-13}$$

Here, $e_{j,k} \in \mathbb{R}^{J \times K}$, at each station j, passengers are arriving to the stations to travel to any destination. Conclusively, the flow rate of passengers entering station j when train k arrives is described as $e_{j,k}$, an represents the j,k-th entry of matrix e.

Implementing the other functionalities as described in the previous section will be more elaborate, as they concern changing the dynamics of the states. Therefore, the derivation of the new state-space equations will be done in a structured manner. Firstly, the new states $\beta_j(k)$ and $\Delta_j(k)$ are introduced , thereafter the existing states will be modified and re-derived.

Deriving the New States

The two states that are to be added, $\beta_j(k)$ and $\Delta_j(k)$ will be derived in this section. To start with, the state representing the amount of passengers disembarking train k at station j, $\beta_j(k)$ will be derived. Since the aim of augmenting the URS is to implement the possibility of assigning travel destinations to passengers embarking at any station, state $\beta_j(k)$ will have to be the sum of all passengers who embarked train k at previously passed station, whose destination is station j. Since this summation will have to be done for each possible destination j and each origin i, a matrix can be introduced that represents the fraction of passengers embarking at station i, who disembark at station j. Hereby, matrix ζ can be introduced;

$$\zeta = \begin{bmatrix} \zeta_{1,1} & \zeta_{1,2} & \cdots & \zeta_{1,J} \\ \zeta_{2,1} & \zeta_{2,2} & \cdots & \zeta_{2,J} \\ \vdots & \ddots & & \vdots \\ \zeta_{J,1} & \cdots & & \zeta_{J,J} \end{bmatrix} \tag{6-14}$$

Here, $\zeta \in \mathbb{R}^{J \times J}$, entry $[\zeta]_{i,j}$ shows the fraction of passengers getting on train k at station i traveling to station j. In the case of this URS, some simplifications can be made to this matrix due to the nature of the system. Since passengers can only travel to future stations, the matrix ζ will be upper triangular. Furthermore, all diagonal entries will be 0, as passengers cannot travel to the destination they embarked at. Therefore, the matrix ζ can be rewritten for this specific application;

$$\zeta = \begin{bmatrix} 0 & \zeta_{1,2} & \cdots & \zeta_{1,(J-1)} & \zeta_{1,J} \\ 0 & 0 & \cdots & \zeta_{2,(J-1)} & \zeta_{2,J} \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & \ddots & 0 & \zeta_{(J-1),J} \\ 0 & \cdots & \cdots & 0 & 0 \end{bmatrix} \tag{6-15}$$

This matrix however, only shows fractions of passengers, not actual quantities. By multiplying row i of this matrix with the number of passengers embarking at station i, the exact amount of passengers disembarking at which station can be generated. By multiplying matrix ζ by a diagonal matrix with the amount of embarking passengers as the diagonal entries, the following matrix can be obtained, assuming that the amount of passengers embarking at each station is represented by state $\Delta_j(k)$.

$$\text{diag}(\Delta_i(k)) \cdot \zeta = \begin{bmatrix} 0 & \zeta_{1,2} \cdot \Delta_1(k) & \cdots & \zeta_{1,(J-1)} \cdot \Delta_1(k) & \zeta_{1,J} \cdot \Delta_1(k) \\ 0 & 0 & \cdots & \zeta_{2,(J-1)} \cdot \Delta_2(k) & \zeta_{2,J} \cdot \Delta_2(k) \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & \ddots & 0 & \zeta_{(J-1),J} \cdot \Delta_{J-1}(k) \\ 0 & \cdots & \cdots & 0 & 0 \end{bmatrix} \tag{6-16}$$

If the entries of column j of this matrix are summed, the total amount of passengers on train k that will disembark at station j is calculated. This results in the final expression for state $\beta_j(k)$;

$$\beta_j(k) = \sum_{s=1}^{j-1} \Delta_s(k) \cdot \zeta_{s,j} \tag{6-17}$$

At first glance, this state may seem implicit, due to the term $\Delta_s(k)$ within the summation. However, since the summation is done from $s = 1$ up to $s = j - 1$, it only takes into account values of $\Delta_s(k)$ up until the previous station, ergo, passengers who embarked at previous stations.

Lastly, let us derive the expression for quantity state $\Delta_j(k)$. The amount of passengers who embark train k at station j can be computed as the minimum of the amount of passengers who **want** to board, and the amount of passengers who **can** board. Both terms will first be derived separately before combining them into the final expression.

The amount of passengers that **can** embark depends on the available capacity on the train after the disembarking passengers have left the train. One can calculate the available capacity by taking the maximum capacity, $\rho_{\max}$, subtracting the amount of passengers that were on train k when it left station $j - 1$, adding the amount of passengers who disembarked train k at station j, as this space has been freed up. This yields the following expression;

$$\Delta_j(k) = \rho_{\max} + \beta_j(k) - \rho_{j-1}(k) \tag{6-18}$$

Subsequently, the expression to compute the amount of passengers that **want** to disembark at station j can be derived as well, but is slightly more complex. The amount of passengers that were left on station j when train $k-1$ departed from that station are for sure passengers that want to embark train k. In the time between the departure of train $k - 1$ and the time passengers can start boarding train k, passengers arrive on the platform at rate $e_{j,k}$. The time passengers can start boarding train k is equal to the arrival time of train k, plus the time it takes for the disembarking passengers to disembark. The amount of passengers who arrived to the platform in this time frame can be computed using the following expression;

$$e_{j,k}(a_j(k) - d_j(k-1) + \frac{\beta_j(k)}{f}) \tag{6-19}$$

The time it takes for the amount of passengers as calculated in 6-19 to board, can be calculated by dividing the expression in 6-19 by f.

Whilst these passengers are boarding, other passengers can still arrive to the platform, at rate $e_{j,k}$. In the time it takes these passengers to board, new passengers can arrive as well. The concept that, in the time frame that passengers board, new passengers arrive who want to board as well, and during their boarding process, again new passengers arrive, and so forth, can be represented by the following expression;

$$\begin{aligned} e_{j,k}(a_j(k) - d_j(k) + \tfrac{\beta_j(k)}{b}) + \tfrac{e_{j,k}^2}{f}(a_j(k) - d_j(k) + \tfrac{\beta_j(k)}{b}) \\ + \tfrac{e_{j,k}^3}{f^2}(a_j(k) - d_j(k) + \tfrac{\beta_j(k)}{b}) + \tfrac{e_{j,k}^4}{f^3}(a_j(k) - d_j(k) + \tfrac{\beta_j(k)}{b}) + \dots \end{aligned} \quad (6\text{-}20)$$

Within the original system description, the requirement that $e_j << b$, which regards system stability was introduced [18]. Choosing the system parameters in the augmented system such that this still holds, allows for drawing some conclusions from the expression stated in 6-20. Because it is known that $e_{j,k} << b$, the boarding time increments during which new passengers can arrive, decreases. By rewriting the expression stated in 6-20, it can be seen that it can be simplified into a power series;

$$\underbrace{(e_{j,k} + \frac{e_{j,k}^2}{f} + \frac{e_{j,k}^3}{f^2} + \frac{e_{j,k}^4}{f^3} + ...)}_{\text{Power series}}(a_j(k) - d_j(k) + \frac{\beta_j(k)}{b}) \quad (6\text{-}21)$$

The underlined part of the expression in 6-21 is a power series of the form;

$$a + ar + ar^2 + ar^3 + ... = \sum_{k=0}^{\infty} ar^k \quad (6\text{-}22)$$

Here, $r = \frac{e_{j,k}}{f}$, and $a = e_{j,k}$. Since it was previously established that $e_{j,k} << f$, it can be concluded that $|r| = |\frac{e_{j,k}}{f}| < 1$, and therefore, the sequence of partial sums S_n converge to a limit value of $\frac{a}{1-r}$, which in this case is $\frac{e_{j,k}}{1-\frac{e_{j,k}}{f}}$. Therefore, the expression in 6-21 can confidently be simplified into;

$$(\frac{e_{j,k}}{1 - \frac{e_{j,k}}{f}})(a_j(k) - d_j(k) + \frac{\beta_j(k)}{b}) \quad (6\text{-}23)$$

After which the completed expression for the amount of passengers who want to board train k at station j can be derived, where the underbraced expression can be written as γ ;

$$\Delta_j(k) = \sigma_j(k-1) + \underbrace{(\frac{e_{j,k}}{1 - \frac{e_{j,k}}{f}})}_{\gamma}(a_j(k) - d_j(k) + \frac{\beta_j(k)}{b}) \quad (6\text{-}24)$$

Thereupon, the final state-space equation for $\Delta_j(k)$ can be assembled from the formerly defined expressions;

$$\Delta_j(k) = \min(\rho_{\max} + \beta_j(k) - \rho_{j-1}(k), \sigma_j(k-1) + \gamma(a_j(k) - d_j(k) + \frac{\beta_j(k)}{b})) \quad (6\text{-}25)$$

Re-deriving Existing States

The four existing states, $a_j(k)$, $d_j(k)$, $\rho_j(k)$ and $\sigma_j(k)$ will all represent the same times and quantities as they did in the previous model. The equations used to calculate these values will change, for all but the arrival time, which will still be described by the following equation;

$$a_j(k) = \max(d_{j-1}(k) + \tau_{r,j}, d_j(k-1) + \tau_H) \tag{6-26}$$

Subsequently, the three remaining existing states can be derived. In order of appearance in the previous model, commencing with departure time $d_j(k)$. Whereas the departure time was a complex state which required the examination of two scenarios for when train k would leave station j, the approach is much more simple in the augmented model. The departure time of train k at station j can be described as the de arrival time of train k at station j, plus the time it takes for the disembarking passengers to disembark, plus the time it takes for the embarking passengers to embark. Mathematically this relation can be formulated as;

$$d_j(k) = a_j(k) + \frac{\beta_j(k)}{b} + \frac{\Delta_j(k)}{f} \tag{6-27}$$

Hereafter, the state $\rho_j(k)$ will be derived. The updated state-space equation will again be very straightforward, by the help of the newly added states. The number of passengers in train k when it departs station j can, in words, be described by the number of passengers in train k when it departed station $j-1$, subtracting the number of passengers that disembarked the train at station j, adding the number of passengers that embarked the train at station j. This mathematically results in the following description;

$$\rho_j(k) = \rho_{j-1}(k) - \beta_j(k) + \Delta_j(k) \tag{6-28}$$

Lastly, the state $\sigma_j(k)$ can be derived. In the original model, the number of passengers at station j when train k departs was just a summation of terms where no max or min operation occurred. In the augmented model however, the state $\sigma_j(k)$ can be described by taking the maximum of the amount of passengers who **want** to board, minus the amount of passengers who **can** board, and zero. In the case that the amount of passengers who want to board is larger than the amount who can board, this subtraction results in a positive value. When taking the maximum of that positive value and zero, the maximization will yield this positive value. On the other hand, when the amount of passenger who want to board is less than the amount who can board, this value will become negative, resulting in the max operation yielding zero. By implementing this maximization, the possibility of obtaining a negative number of passengers at station j when train k leaves is bypassed. The expression that gives amount of passengers that **want** to and **can** board the train is given by;

$$\begin{aligned} \Delta_j^{want}(k) &= \sigma_j(k-1) + \gamma(a_j(k) - d_j(k-1) + \frac{\beta_j(k)}{f}) \\ \Delta_j^{can}(k) &= \rho_{\max} + \beta_j(k) - \rho_{j-1}(k) \end{aligned} \tag{6-29}$$

Combining these to expressions into a maximization will yield the mathematical expression for $\sigma_j(k)$;

$$\begin{aligned} \sigma_j(k) &= \max(0, \Delta_j^{want}(k) - \Delta_j^{can}(k)) \\ \sigma_j(k) &= \max(0, (\sigma_j(k-1) + \gamma(a_j(k) + d_j(k-1) + \frac{\beta_j(k)}{f})) - (\rho_{\max} + \beta_j(k) - \rho_{j-1}(k))) \end{aligned} \tag{6-30}$$

6-2-3 Augmented State-Space Model

This concludes the (re)derivation of the augmented state-space model, and allows for a proper overview of these new states. Without further ado, the complete set of state space equations is given by;

$$\begin{aligned}
a_j(k) &= \max(d_{j-1}(k) + \tau_r, d_j(k-1) + \tau_H) \\
d_j(k) &= a_j(k) + \frac{\beta_j(k)}{f} + \frac{\Delta_j(k)}{b} \\
\rho_j(k) &= \rho_{j-1}(k) - \beta_j(k) + \Delta_j(k) \\
\sigma_j(k) &= \max(0, (\sigma_j(k-1) + \gamma(a_j(k) + d_j(k-1) + \frac{\beta_j(k)}{f})) - (\rho_{\max} + \beta_j(k) - \rho_{j-1}(k))) \\
\beta_j(k) &= \sum_{s=1}^{j-1} \Delta_s(k) \cdot \zeta_{sj} \\
\Delta_j(k) &= \min(\rho_{\max} + \beta_j(k) - \rho_{j-1}(k), \sigma_j(k-1) + \gamma(a_j(k) - d_j(k-1) + \frac{\beta_j(k)}{f}))
\end{aligned} \tag{6-31}$$

The previous section concluded with the set of state-space equation, given in 6-31.This marks the starting point of the elaborate system analysis performed in this chapter. Where first and foremost, the state-space equations will be transformed into the ABCD form.

6-3 ABCD Form

Using the framework for the ABCD form proposed for the original URS in [15] as inspiration, a framework for constructing the ABCD form for the augmented system can be developed. The state-space equations in 6-31 are the equations used to calculate the dynamics related to train k at station j. The ABCD form should allow for these dynamics to be calculated for each station j, and therefore, the ABCD form can be divided into j sets of state-space systems, compiled together into the larger, all-encompassing system. Matrices A, B, C and D can be divided into submatrices which simplify the task of creating the full system;

$$A = \begin{bmatrix} \bar{A}_1 & \varepsilon & \cdots & \varepsilon \\ \varepsilon & \bar{A}_2 & \cdots & \varepsilon \\ \vdots & \ddots & \ddots & \vdots \\ \varepsilon & \cdots & \varepsilon & \bar{A}_J \end{bmatrix}, \quad B = \begin{bmatrix} \bar{B}_1 & \top & \cdots & \top \\ \top & \bar{B}_2 & \cdots & \top \\ \vdots & \ddots & \ddots & \vdots \\ \top & \cdots & \top & \bar{B}_J \end{bmatrix} \tag{6-32}$$

$$C = \begin{bmatrix} \bar{C}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \bar{C}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \bar{C}_J \end{bmatrix}, \quad D = \begin{bmatrix} \bar{D}_{1,1,1} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \bar{D}_{2,2,1} & \bar{D}_{1,2,2} & \mathbf{0} & \ddots & \vdots \\ \bar{D}_{3,3,1} & \bar{D}_{2,3,2} & \bar{D}_{1,3,3} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \bar{D}_{3,I,1} & \cdots & \bar{D}_{3,I,J-2} & \bar{D}_{2,I,J-1} & \bar{D}_{1,I,J} \end{bmatrix} \tag{6-33}$$

These matrices can be merged into the following ABCD state-space description, which is merely a repetition of the disjunctive state-space description provided in 3-1.3.

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k))) \tag{6-34}$$

In order to fully define these matrices, six different types of submatrices will have to be computed in order to completely define the system. Firstly, an overview of the to be defined submatrices, and their dimensions, is given below;

1. $\bar{A}_j \in \mathbb{R}_{\varepsilon}^{n\times m}$
2. $\bar{B}_j \in \mathbb{R}_{\top}^{m\times p}$
3. $\bar{C}_j \in \mathbb{R}^{p\times n}$
4. $\bar{D}_{1,i,j} \in \mathbb{R}^{p\times n}$
5. $\bar{D}_{2,i,j} \in \mathbb{R}^{p\times n}$
6. $\bar{D}_{3,i,j} \in \mathbb{R}^{p\times n}$

Now defining these submatrices one by one allows for further analysis.

$$\bar{A}_j = \left[\begin{array}{ccc|ccccc} \tau_r & \tau_H & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \hline \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & -\rho_{\max} & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 \end{array}\right] \tag{6-35}$$

$$\bar{B}_j = \left[\begin{array}{ccc|cccccc} 0 & \top & \top & \top & \top & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top & \top & \top & \top & \top \\ \top & \top & 0 & \top & \top & \top & \top & \top & \top \\ \hline \top & \top & \top & 0 & \top & \top & \top & \top & \top \\ \top & \top & \top & \top & 0 & \top & \top & \top & \top \\ \top & \top & \top & \top & \top & 0 & \top & \top & \top \\ \top & \top & \top & \top & \top & \top & 0 & \top & \top \\ \top & \top & \top & \top & \top & \top & \top & \rho_{\max} & 0 \end{array}\right] \tag{6-36}$$

$$\bar{C}_j = \left[\begin{array}{cc|cccc} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\gamma & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\gamma & 0 & 1 & 0 & 0 \end{array}\right] \tag{6-37}$$

$$\bar{D}_{1,i,j} = \left[\begin{array}{cc|cccc} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & \frac{1}{f} & \frac{1}{b} \\ \hline 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \gamma & 0 & 0 & 0 & \frac{\gamma}{f}-1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ \gamma & 0 & 0 & 0 & \frac{\gamma}{f} & 0 \end{array}\right] \tag{6-38}$$

$$\bar{D}_{2,i,j} = \left[\begin{array}{cc|cccc} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \zeta_{i,j} \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \tag{6-39}$$

$$\bar{D}_{3,i,j} = \left[\begin{array}{cc|cccc} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \zeta_{i,j} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \tag{6-40}$$

These submatrices can be compiled together to compute the ABCD form of the AURS. The dimensions of the complete ABCD form for K trains and J stations has the following dimensions; $A \in \mathbb{R}_{\varepsilon}^{nJ\times mJ}$, $B \in \mathbb{R}_{\top}^{mJ\times pJ}$, $C, D \in \mathbb{R}^{pJ\times nJ}$.

6-3-1 Time-Invariance

Since the D matrix of the state-space exists, and is not empty, the AURS is an implicit system. Equation 3-10 provides the conditions under which an implicit MMPS system is time-invariant, and is recalled below.

$$\begin{aligned} \sum_{i\in\overline{n_{\mathrm{t}}}} \begin{bmatrix} C_{11} & D_{11} \end{bmatrix}_{\ell i} &= 1, \forall \ell \in \overline{p_{\mathrm{t}}} \\ \sum_{i\in\overline{n_{\mathrm{t}}}} \begin{bmatrix} C_{21} & D_{21} \end{bmatrix}_{ti} &= 0, \forall t \in \overline{p_{\mathrm{q}}} \end{aligned} \tag{6-41}$$

By examining whether this condition holds, time-invariance can be proven, or disproven. It is important to note that, the property of time-invariance is checked for each individual station, not for the full system matrix.

The dynamics for each train and station are identical, therefore this is not necessary either.

$$\sum_{i\in\overline{n_t}} \begin{bmatrix} C_{11} & D_{1,11} & D_{2,11} & D_{3,11} \end{bmatrix}_{\ell i} = \left[\begin{array}{cc|cc|cc|cc} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{array}\right] = \mathbf{1}$$
$$\sum_{i\in\overline{n_t}} \begin{bmatrix} C_{21} & D_{1,21} & D_{2,21} & D_{3,21} \end{bmatrix}_{ti} = \left[\begin{array}{cc|cc|cc|cc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\gamma & \gamma & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\gamma & \gamma & 0 & 0 & 0 & 0 & 0 \end{array}\right] = \mathbf{0} \tag{6-42}$$

Hereby, the conclusion can be drawn that the AURS is time-invariant.

6-4 Solvability of the AURS

Considering the solvability of the system, the theory presented in Chapter 4 can be applied. Firstly, matrices S_A, S_B and S_D will be derived;

$$S_A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} S_B = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$
$$S_D = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{6-43}$$

Matrix $S = S_A \cdot S_B \cdot S_D$ will therefore have the following structure;

$$S = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 2 & 0 \end{bmatrix} \tag{6-44}$$

The obtained matrix S is not immediately strictly lower-triangular.

In order to determine whether this transformation exists, i.e. the system does not contain a circuit, Theorem 4-1.4 will be applied, which yields matrix $S_{\otimes}^{+}$;

$$S_{\otimes}^{+} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 2 & \varepsilon & \varepsilon & \varepsilon & 3 & 1 \\ 2 & \varepsilon & \varepsilon & \varepsilon & 3 & 1 \\ 1 & \varepsilon & \varepsilon & \varepsilon & 1 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon \end{bmatrix} \tag{6-45}$$

The diagonal entries $[S_{\otimes}^{+}]_{ii}$ are all ε, excluding the possibility a circuit exists in this system. Therefore, the transformation matrix T exists that yields a strictly lower-triangular matrix F, and the Augmented Urban Railways system is solvable as per Theorem 4-1.1.

6-5 Initializing the System

In order to simulate the AURS, it must be initialized. Ultimately, the aim is to simulate the system according to a uniform timetable. This refers to the intervals between the departures and the arrivals to be identical for all trains, and stations. This is desirable as this uniform behaviour, when initialized correctly, yields a stable system, without the need for any control. Since there exists a propagation of dynamics in *two* directions, namely, the trains and the stations, both the dynamics of the initial station(s), and the initial train must be initialized. Within the state-space equations as per 6-31, terms containing dynamics from the previous, and current train, as well as dynamics from the previous and current station are included. Assuming the system is solvable, in order to compute the evolution of a state, all necessary information about previous stations and/or trains must be available. The dynamics of the first station are therefore different than the other stations, as their state evolution equations cannot depend on dynamics from the previous station, as it simply does not exist. The dynamics of the first station are given by the following equations;

$$\begin{aligned} a_1(k) &= a_1(k-1) + \tau_0 & \forall k \in \{1, \dots K\} \\ d_1(k) &= d_1(k-1) + \tau_0 & \forall k \in \{1, \dots K\} \\ \rho_1(k) &= \rho_1(k-1) & \forall k \in \{1, \dots K\} \\ \sigma_1(k) &= \sigma_1(k-1) & \forall k \in \{1, \dots K\} \\ \beta_1(k) &= \beta_1(k-1) & \forall k \in \{1, \dots K\} \\ \Delta_1(k) &= \Delta_1(k-1) & \forall k \in \{1, \dots K\} \end{aligned} \tag{6-46}$$

The state-space equations for the first station as given in 6-46 can be written into the ABCD form;

$$\bar{A}_1 = \left[\begin{array}{cc|cccc} \tau_0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \tau_0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \hline \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 \end{array}\right], \quad \bar{B}_1 = \left[\begin{array}{cc|cccc} 0 & \top & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top & \top \\ \hline \top & \top & 0 & \top & \top & \top \\ \top & \top & \top & 0 & \top & \top \\ \top & \top & \top & \top & 0 & \top \\ \top & \top & \top & \top & \top & 0 \end{array}\right]$$

$$\bar{C}_1 = \left[\begin{array}{cc|cccc} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right], \quad \bar{D}_{1,1,1} = \left[\begin{array}{cc|cccc} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \tag{6-47}$$

Even though the dimensions of $\bar{A}_1, \bar{B}_1, \bar{C}_1$ and $\bar{D}_{1,1}$ are different than those of $\bar{A}_j, \bar{B}_j, \bar{C}_j$ and $\bar{D}_j$ where $j \neq 1$, the matrix structure as described in 6-32 and 6-33 will still concatenate nicely, as all off-diagonal submatrices of matrices A and B are ε or $\top$ anyway. Furthermore, the matrices $\bar{C}_1$ and $\bar{D}_1$ still comply with the time-invariance conditions as per 6-41, and do not violate solvability of the system.

6-6 Growth Rates and Fixed-Points

Having fully described the dynamics and initial station of the augmented model, the growth rates and fixed-points can be derived. The linear programming problem (LPP) that is used to compute the eigenvalues of this implicit system are derived and presented in [15]. This algorithm is already mentioned in 3-33, and repeated below;

$$\begin{aligned} \min_{x_e, y_e, w_e} \quad & \lambda \\ \text{s.t.} \quad & -[s\lambda]_i - [x]_i + [y]_j \leq -[A]_{ij} \quad \text{if } [G_{A_\theta}]_{ij} = 0 \\ & [s\lambda]_i + [x]_i - [y]_j = [A]_{ij} \quad \text{if } [G_{A_\theta}]_{ij} = 1 \\ & [y]_j - [d]_\ell\lambda - [w]_\ell \leq [B]_{j\ell} \quad \text{if } [G_{B_\theta}]_{jl} = 0 \\ & -[y]_j + [d]_\ell\lambda + [w]_l = [B]_{jl} \quad \text{if } [G_{B_\theta}]_{jl} = 1 \\ & d = D \cdot s, \quad w = (C + D) \cdot x \end{aligned} \tag{6-48}$$

The following parameters are chosen for the augmented urban railway network. These can, to some extent, be chosen arbitrarily. However, similarly to the original URS, stability requirement $e_j << b$ was defined. By stating the arrival rate of passengers at the stations must be significantly smaller than the embark rate of passengers, it is at least possible for the train to leave the station without being full and no passengers being left behind at the platform. Comparable stability requirements can be derived for the AURS.

The following stability requirements can be derived for the AURS;

- $e_j << b \quad \forall j \in \{1, 2, \ldots, J\}$ The requirement of the arrival rate being significantly lower than the boarding rate as described above must hold for the AURS as well
- $e_j << f \quad \forall j \in \{1, 2, \ldots, J\}$ Similarly to the arrival rate having to be significantly less than the boarding rate, the disembark rate must be significantly smaller than the arrival rate as well.
- $e_j \cdot \tau_H \leq \rho_{\max}$ The number of passengers arriving in the minimum headway time must be less than the train capacity. In case this number is larger, the amount of passengers wanting to disembark when any train arrives is always larger than the train capacity, causing all trains to always be full, with the value of state $\sigma_j(k)$ always growing.
- $\sum_{j=1}^{J}[\zeta]_{ij} = 1$ Since matrix ζ is a probability matrix, the sum of its rows must always be equal to 1. If the sum is of a row is larger than 1, passengers are created out of thin air, and if the sum is smaller than 1, some passengers never disembark.

The chosen parameters must comply with the stability requirements. A small remark regarding the last stability requirement must be made. Theoretically, if the AURS is an urban railway line beginning at station 1, and ending at station J, no passengers can embark at station the last station, as they have no destination to travel to. Therefore, the sum of the last row of matrix ζ cannot theoretically be 1. However, let us assume this AURS can be physically interpreted as an urban railway line of $N >> J$ stations, where just a segment of J stations is analyzed. Hereby, the first and last stations can function according to the dynamics of an intermediate station, without making unrealistic assumptions. The dimensions of matrix ζ used in simulation can therefore be larger than $\zeta \in \mathbb{R}^{J \times K}$, allowing for assigning a destination of passengers embarking at station J, without actually analyzing the behaviour of the trains past station J. Let us introduce the values of the parameters present in the system matrices in the table below, this is not the actual initial condition of the system.
So, matrix ζ was chosen such that, passengers embarking at any station j, half the passengers disembark at station $j+1$, and the other half disembark at station $j+2$. The state $\beta_j(k)$ can be simplified to the following expression;

$$\beta_j(k) = 0.5 \cdot \Delta_{j-2}(k) + 0.5 \cdot \Delta_{j-1}(k) \tag{6-49}$$

Knowing the state $\beta_j(k)$ can be described by this equation, it becomes apparent that for this choice of matrix ζ, when computing $x_j(k)$, only state values of $x_j(k-1)$, $x_{j-1}(k)$, $x_{j-2}(k)$, and $x_j(k)$ itself are needed. Therefore, the proposed matrix D as per 6-33 can be simplified into the following matrix;

$$D = \begin{bmatrix} \bar{D}_{1,1,1} & \mathbf{0} & \cdots & \cdots & \cdots & \cdots & \mathbf{0} \\ \bar{D}_{2,2,1} & \bar{D}_{1,2,2} & \mathbf{0} & \ddots & \ddots & \ddots & \vdots \\ \bar{D}_{3,3,1} & \bar{D}_{2,3,2} & \bar{D}_{1,3,3} & \mathbf{0} & \ddots & \ddots & \vdots \\ \mathbf{0} & \bar{D}_{3,4,2} & \bar{D}_{2,4,3} & \bar{D}_{1,4,4} & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \bar{D}_{3,5,3} & \bar{D}_{2,5,4} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \cdots & \mathbf{0} & \bar{D}_{3,I,J-2} & \bar{D}_{2,J-1} & \bar{D}_{1,I,J} \end{bmatrix} \tag{6-50}$$

Table 6-1: Parameters of the AURS

Parameter	Value	Unit	Definition
τ_0	60	s	Initial headway time
τ_H	20	s	Minimum headway time
τ_r	180	s	Running time
$\rho_{\max}$	150	passengers	Maximum train capacity
b	2	passengers/s	Boarding rate
f	2	passengers/s	Disembark rate
$\gamma = \frac{e_{j,k}}{1-\frac{e_{j,k}}{b}}$	$\frac{2}{3}$	passengers/s	Sum of power series
$e_{j,k}$	$\begin{bmatrix} 0.5 & 0.5 & \cdots & 0.5 \\ 0.5 & 0.5 & \cdots & 0.5 \\ \vdots & \ddots & \ddots & \vdots \\ 0.5 & 0.5 & \cdots & 0.5 \end{bmatrix}$	passengers/s	Inflow of passengers at each station
ζ	$\begin{bmatrix} 0 & 0.5 & 0.5 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0.5 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & \cdots & 0 & 0 \end{bmatrix}$	-	Matrix with fractions of passenger travel directions

Keep in mind this is a parametric representation, adaptable to any number of stations J Since computation of the number of passengers requires information from the previous two stations the train has passed, not only the first, but also the second station should be initialized. Only from the third station onwards, information about the previous two stations is available. It would theoretically be possible to let these unknown values $\Delta_{-1}(k)$ and $\Delta_0(k)$ be zero, but this would be unfavourable considering the aim to simulate the URS according to a uniform timetable. Therefore, the following equations will describe the dynamics of the second station;

$$\begin{aligned} a_2(k) &= d_1(k) + \tau_r && \forall k = 1, \dots K \\ d_2(k) &= d_2(-1k) + \tau_0 && \forall k = 1, \dots K \\ \rho_2(k) &= \rho_2(k-1) && \forall k = 1, \dots K \\ \sigma_2(k) &= \sigma_2(k-1) && \forall k = 1, \dots K \\ \beta_2(k) &= \beta_2(k-1) && \forall k = 1, \dots K \\ \Delta_2(k) &= \Delta_2(k-1) && \forall k = 1, \dots K \end{aligned} \tag{6-51}$$

The ABCD matrices describing these dynamics are given by;

$$\bar{A}_2 = \left[\begin{array}{cc|cccc} \tau_r & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \tau_0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \hline \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 \end{array}\right], \quad \bar{B}_2 = \left[\begin{array}{cc|cccc} 0 & \top & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top & \top \\ \hline \top & \top & 0 & \top & \top & \top \\ \top & \top & \top & 0 & \top & \top \\ \top & \top & \top & \top & 0 & \top \\ \top & \top & \top & \top & \top & 0 \end{array}\right]$$

$$\bar{C}_2 = \left[\begin{array}{cc|cccc} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right], \quad \bar{D}_{1,2,2} = \left[\begin{array}{cc|cccc} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \tag{6-52}$$

$$, \quad \bar{D}_{2,2,1} = \left[\begin{array}{cc|cccc} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

It can easily be verified that the time-invariance conditions as per 6-41 still apply here. The LPPs given in 6-48 can be solved in MATLAB using a linear solver for $J = 4$ stations. In this case, the Gurobi solver was used. A total of 64 footprint matrix combinations is possible. For each station added to the simulated system, this number would be multiplied by a factor 8, as the dynamics of each additional station would generate 4 extra possible G_A matrices, and 2 extra possible G_B matrices. The running time to evaluate 64 combinations of footprint matrices is 11.7 s. Out of the 64 combinations, 40 yield a growth rate, each of them has value $\lambda^* = 60$. This eigenvalue is expected, as some time dynamics are initialized in the system matrices. The difference between temporal state $a_1(k-1)$ and $a_1(k)$ for example, is always τ_0 which is defined as 60, as visible in the system matrix $\bar{A}_1$. In order for the eigenvalue to exist, all temporal states must grow with the same value for each event step k. The only way for the system to adhere to this requirement, which is also embedded in the LPPs, is for the value λ to be equal to 60. Equivalently, in the original URS as in [18], the initial headway time was set at $\tau_0 = 120$, and the computed eigenvalues were also all equal to 120. Solving the LPP yields 40 eigenvalues 60, all corresponding to a unique combination of footprint matrices G_{A_θ} G_{B_θ}, and corresponding set of eigenvectors, denoted by $V = \{v_1, v_2, \dots, v_{40}\}$, where $v_i = (x_{e,i}^T, y_{e,i}^T, z_{e,i}^T)$. Let us first verify whether these obtained eigenvalues and eigenvectors actually satisfy the conditions for them to be eigenvectors and values. By computing $x_{e,\lambda} = x_e - s\lambda$ and $A_\lambda = \begin{bmatrix} A_{\mathrm{t},\lambda} & \varepsilon \\ \varepsilon & A_{\mathrm{q}} \end{bmatrix}$ where $A_{\mathrm{t},\lambda} = A_{i_\mathrm{t} j_\mathrm{t}} - \lambda$, and verifying whether the set of equations as given in 3-25 hold true for all obtained eigenvectors v_i. All obtained eigenvalue and eigenvector combinations are confirmed to actually be eigenvectors and eigenvalues.

Any of these solutions $\lambda^*, v^* = [\ x^{*\top} \quad y^{*\top} \quad z^{*\top}\ ]$ are solutions to the LLP, but might not be the only solutions. Substituting the value $\lambda^* = \lambda$ in the LPPs in 6-48, a system of equality and inequality constraints such as in 3-35 can be obtained;

$$H_{eq} \cdot v = h_{eq}, \quad H_{ineq} \cdot v \leq h_{ineq} \tag{6-53}$$

Matrices H_{eq}, h_{eq}, H_{ineq} and h_{ineq} will not be given here due to their respective sizes, and quantities; $H_{eq} \in \mathbb{R}^{82\times82}$, $h_{eq} \in \mathbb{R}^{82\times1}$, $H_{ineq} \in \mathbb{R}^{6\times82}$ and $h_{ineq} \in \mathbb{R}^{6\times1}$. All four of these matrices exist for all 40 fixed-points. The set of fixed-points can be described by $\mathcal{V}_{\lambda^*} = \{v | H_{ineq} \cdot v \leq h_{ineq}\}$ as per 3-37 where $v = v^* + \sigma_1 g_1 + \sigma_2 g_2 + \cdots + \sigma_f g_f$. The number of terms in the expression for v is related to the rank deficiency of H_{eq}, as was described in Subsection 3-3-2 . For all 40 instances of eigenvalue 60, the rank deficiency is equal to 11, meaning 11 direction vectors to describe all fixed-points exist. Each of the 40 eigenvectors that were obtained solutions of the LPP is unique, and 11 direction vectors form the base for the fixed-point solution space. Therefore, the rank of the matrix $V_e = [\ v_1 \quad v_2 \quad \ldots \quad v_{40}\ ]$, in which each column is one obtained eigenvector, must be maximum 11, so $\text{rank}(V_e) \leq 11$. The rank of V_e is in this case 4. So, the solutions to the LPP yielded only 4 of the existing 11 direction vectors, leaving 7 of these direction vectors unknown. The remaining 7 direction vectors can theoretically obtained by constructing them one by one from the matrix H_{eq} [15]. The expression for v can be described by;

$$v = v^* + \sigma_1 g_1 + \sigma_2 g_2 + \sigma_3 g_3 + \sigma_4 g_4 + \sigma_6 g_6 + \sigma_7 g_7 + \sigma_8 g_8 + \sigma_9 g_9 + \sigma_{10} g_{10} + \sigma_{11} g_{11} \tag{6-54}$$

Due to the time invariance property, scaling factor σ_1 is unbounded, as mentioned in [15]. The possible bounds on the other scaling factors $\sigma_j \in \{\sigma_2, \sigma_3, \ldots, \sigma_{11}\}$ can be found from the following equation;

$$H_{ineq} \cdot v_p + \sigma_1 \cdot H_{ineq} \cdot g_1 + \sigma_2 \cdot H_{ineq} \cdot g_2 + \cdots + \sigma_{11} \cdot H_{ineq} \cdot g_{11} \leq h_{ineq} \tag{6-55}$$

Since 7 out of the 11 direction vectors remain unknown, the bounds on the scaling factors cannot be determined, as this is an underdefined problem. However, the direction vectors can be determined. The first of the direction vectors, g_1, can already be determined as follows. For each station j, vector $g_{1,j}$ a vector with similar characteristics to $s = [\ \mathbf{1}_{nt}^{\mathsf{T}} \quad \mathbf{0}_{n_q}^{\mathsf{T}}\ ]^{\mathsf{T}}$, as mentioned in Subsection 3-3-2;

$$g_{1,j} = \begin{bmatrix} g_{1,x,j} \\ g_{g,y,j} \\ g_{1,z,j} \end{bmatrix} \tag{6-56}$$

Basically, for $x^{*\mathsf{T}}$, $y^{*\mathsf{T}}$ and $z^{*\mathsf{T}}$ in $v^* = [\ x^{*\top} \quad y^{*\top} \quad z^{*\top}\ ]^{\mathsf{T}}$, a vector with ones on the rows representing temporal states, and zeroes representing quantity states will be computed. Therefore, $g_{1,x,j} = s$, where $s = [\ \mathbf{1}_{nt}^{\mathsf{T}} \quad \mathbf{0}_{n_q}^{\mathsf{T}}\ ]^{\mathsf{T}}$. Vectors $g_{1,y,j}$ and $g_{1,z,j}$ can be found using the following equations;

$$\begin{aligned} g_{1,y,j} &= B \otimes' ((C + D) \cdot s) \\ g_{1,z,j} &= (C + D) \cdot s \end{aligned} \tag{6-57}$$

For the AURS, vectors $g_{1,x,j}$, $g_{1,y,j}$ and $g_{1,z,j}$ can be given by;

$$g_{1,x,j} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad g_{1,y,j} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, g_{1,z,j} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{6-58}$$

Lastly, vectors $g_{1,j}$ will be stacked to compute full direction vector g_1 as shown in equation 6-59.

$$g_1 = \begin{bmatrix} g_{1,1} \\ g_{1,2} \\ \vdots \\ g_{1,j} \end{bmatrix} \tag{6-59}$$

For this AURS with parameters as in Table 6-2, the vectors $g_{1,1}$ and $g_{1,2}$, for the initial stations, can be given by $g_1 = \left[g^{\intercal}_{1,x,j}, g^{\intercal}_{1,x,j}, g^{\intercal}_{1,x}\right]^{\intercal}$, as the sizes of the submatrices $\bar{B}_1$ $\bar{B}_2$, $\bar{C}_1$, $\bar{C}_2$, $\bar{D}_{1,1,1}$, $\bar{D}_{1,2,2}$ and $\bar{D}_{2,2,1}$ are all $\mathbb{R}^{6\times 6}$. Please note that this parametric description of vector g_1 only applies to the AURS, as a more general approach does not include stacking dynamics of multiple stations. A more general approach to computing g_1 is given in 3-3-2.

6-7 Simulation of the AURS

In the previous analysis, fixed-point analysis was performed on the AURS with 4 stations. When aiming to simulate the AURS according to a uniform timetable, the behaviour of a train is the same across all stations, meaning the dwell times of each train at each station must be the same, and the quantity states must not grow as the train passes more stations. This corresponds to a constant number of passengers embarking and disembarking each train at each station, and a constant number of passengers in each train between stations. Uniformity of the temporal states directly relates constant state values for the quantity states, as the dwell time directly relates to the number of passengers embarking and disembarking through the boarding rate b, and disembark rate f. Furthermore, it would be interesting to see the dynamics of the trains across more than 4 stations. In order to obtain a fixed-point that is feasible for a system with $J > 4$ stations, the size of the algorithm used in Section 6-6 will grow. Considering the eightfold increase in footprint matrix combinations, the running time for each extra added station is theoretically 8 times longer. When evaluating this algorithm with a system with 5 stations, the running time for 512 footprint matrix combinations is 85.6 s. The algorithm yields one eigenvalues of value 60, and 136 corresponding eigenvalues. This running time is a little less than $8 \cdot 11.7 = 93.6$ s, this is probably due to the postprocessing of the obtained eigenvalues and assigning them an eigenvector. The number of eigenvalues has not increased with a factor 8, therefore, the postprocessing will have to be done for less than $8 \cdot 40 = 320$ eigenvectors, yielding a lower running time.

However, continuing the trend of increasing the running time by at least a factor 7, the running time for evaluating a system with 12 stations would take approximately 2.14 years, which is a conservative estimation. In conclusion, it is not feasible to run this algorithm for many stations. Furthermore, similarly to the analysis performed on the URS in [18], the eigenvectors obtained by solving the LPPs did not provide a fixed-point that would be fit as an initial condition to attain such uniform behaviour. Therefore, a fixed-point will be proposed which, if proven to be valid, will serve as the initial condition for simulating the system according to a uniform timetable. Initial condition $x(1)$ can be described by the following expression;

$$x(1) = \begin{bmatrix} x_1(1) \\ x_2(1) \\ \vdots \\ x_J(1) \end{bmatrix} \tag{6-60}$$

A parametric description of the initial condition of the j-th station can be manually derived;

$$x_1(1) = \begin{cases} a_1(1) = 0 \\ d_1(1) = \bar{d} \\ \rho_1(1) = \bar{\rho} \\ \sigma_1(1) = 0 \\ \beta_1(1) = \bar{\beta} \\ \Delta_1(1) = \bar{\Delta} \end{cases} \tag{6-61}$$

$$x_j(1) = \begin{cases} a_j(1) = d_{j-1}(1) + \tau_r & \forall j = 2, \dots N \\ d_j(1) = a_j(1) + \tau_d & \forall j = 2, \dots N \\ \rho_j(1) = \rho_{j-1}(1) & \forall j = 2, \dots N \\ \sigma_j(1) = 0 & \forall j = 2, \dots N \\ \beta_j(1) = \beta_{j-1}(1) & \forall j = 2, \dots N \\ \Delta_j(1) = \Delta_{j-1}(1) & \forall j = 2, \dots N \end{cases} \tag{6-62}$$

The values of parameters $\bar{d}$, $\bar{\rho}$, $\bar{\beta}$ and $\bar{\Delta}$ are given in Table 6-2.

Parameter	Value	Unit	Definition
$\bar{d}$	30	s	Dwell time at station 1
$\bar{\rho}$	45	passengers	Number of passengers in train when leaving station 1
$\bar{\beta}$	30	passengers	Number of passengers embarking train at station 1
$\bar{\Delta}$	30	passengers	Number of passengers disembarking train at station 1

Table 6-2: Initial conditions for simulating the AURS

The choice for these exact values is not random, and do not require complex derivation, given some logical assumptions. The interval between two trains arriving at a station is 60 seconds, as this is the growth rate. Within those 60 seconds, $60 \cdot e_{j,k} = 60 \cdot 0.5 = 30$ passengers arrive at each station. The number of embarking passengers $\beta_j(k)$ at any station is therefore equal to 30, making $\bar{\beta} = 30$. In a uniform timetable scenario, the trains are never at capacity, and all of these 30 passengers are able to board the train. Matrix ζ is known, and tells us that half the passengers embarking at station j disembark at station $j+1$, and the other half disembark at station $j+2$. Therefore, from these 30 embarking passengers, 15 disembark at the next station, and 15 at the one after that. Given that the train exhibits the same behaviour at every station, 30 passengers disembark from any given train at each station, yielding a value of $\bar{\Delta} = 30$. Therefore, $\bar{d}$ the dwell time at the first station, and therefore at every station is 30 seconds. In these 30 seconds, all passengers wanting to disembark must have disembarked, and all passengers wanting to embark must have embarked. From this information, the value of $\bar{d}$ can be computed;

$$\begin{aligned} \bar{d} &= \frac{\beta_1(1)}{f} + \frac{\Delta_1(1)}{b} \\ \bar{d} &= \frac{30}{2} + \frac{30}{2} \\ \bar{d} &= 30 \end{aligned} \tag{6-63}$$

Assuming the trains never overflow in a correctly chosen uniform timetable scenario, value $\sigma_j(1) = 0$. Lastly, the number of passengers in the train after leaving station j, $\bar{\rho}$ is equal to 45, as 30 passengers embark at station j, and 15 are traveling to the next station. This constitutes the derivation of the manually constructed initial condition. For $J = 15$ stations, the initial condition vector would be $x(1) \in \mathbb{R}^{90\times 1}$. It can be established this initial condition functions as a fixed-point by substituting $x(1)$ as x_e in 3-25 with $\lambda = 60$, and verifying whether the equations hold, which they do, by which this proposed $x(1)$ also functions as x_e.

So, using this $x(1)$ with $J = 15$ stations, and $K = 10$ trains the AURS can be simulated, which is given in Figures 6-2, 6-3, 6-4. 6-4 and, 6-6. As is visible in these figures, all quantity states $\rho_j(k)$, $\beta_j(k)$, $\Delta_j(k)$ and $\sigma_j(k)$ have constant values of 45, 30, 30 and 0 passengers, respectively. This proves the uniformity of the behaviour of the trains relative to each other, and themselves. Furthermore, the train trajectory plot 6-2 shows a very constant timetable , with every line describing the trajectory of one train. Each train's schedule is represented as a staircase-like function: horizontal segments for when the train is standing still at a station, vertical segments for running between stations. All ten of these functions are translated along the time axis in 60 second increments, so the time between the arrival or departure of any two successive trains is always 60 seconds, which is equal to the growth rate of this system. The difference between the departure of a train and the arrival of the next is always 30 seconds, which is 10 seconds more than the minimum headway time.

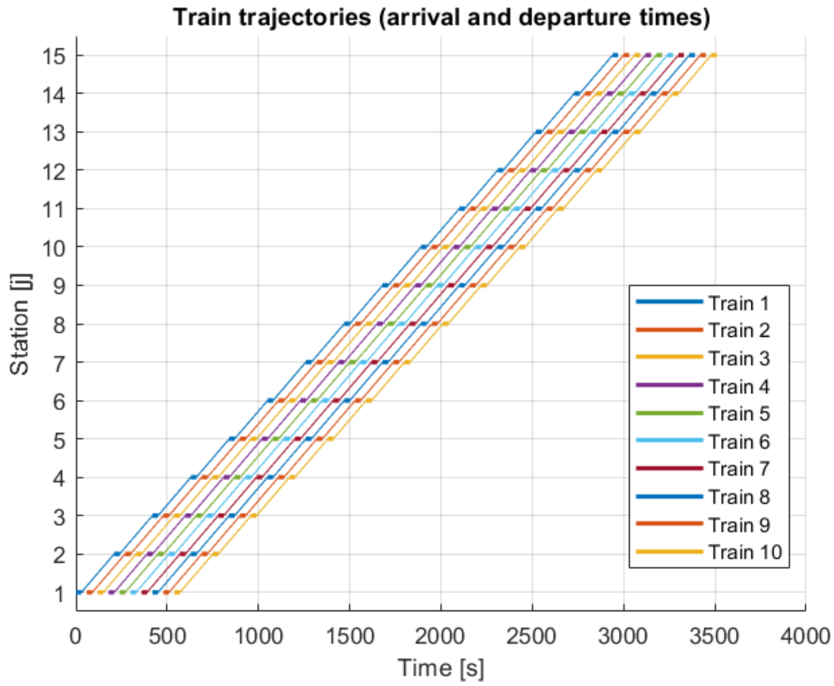


Figure 6-2: Train trajectories of the uniform timetable simulation of the AURS with 10 trains and 15 stations

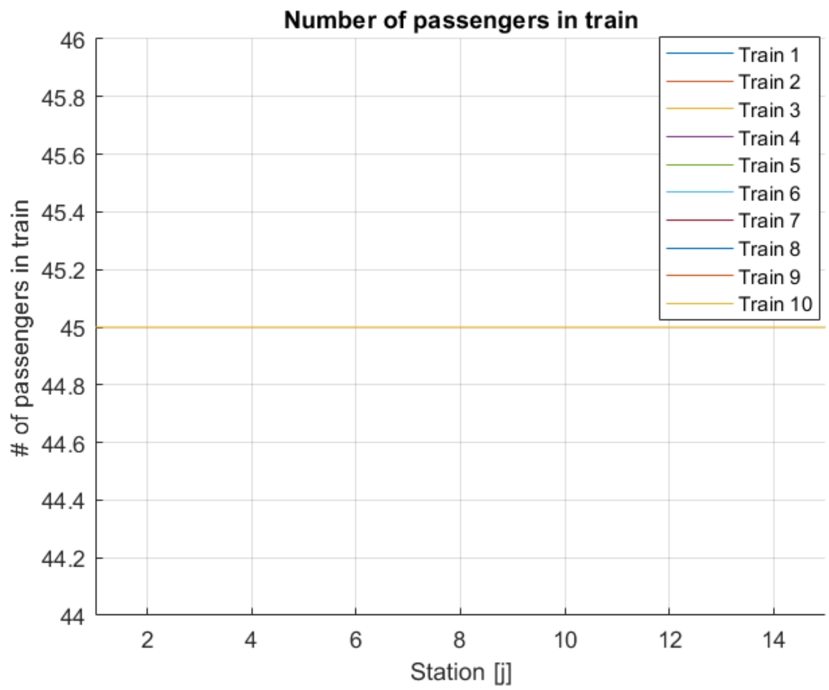


Figure 6-3: Number of passengers in each train at each station of the uniform timetable simulation of the AURS with 10 trains and 15 stations

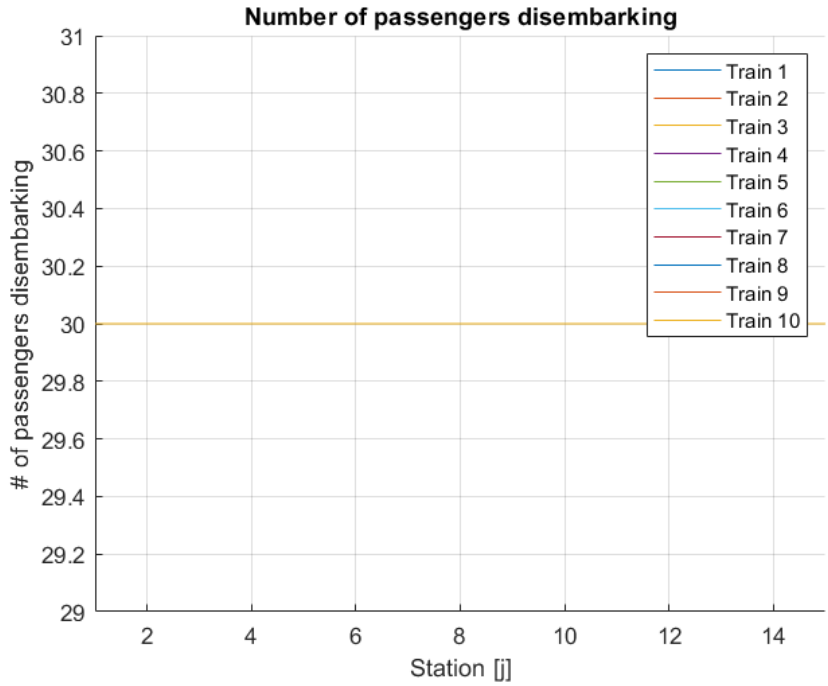


Figure 6-4: Number of passengers disembarking each train at each station of the uniform timetable simulation of the AURS with 10 trains and 15 stations

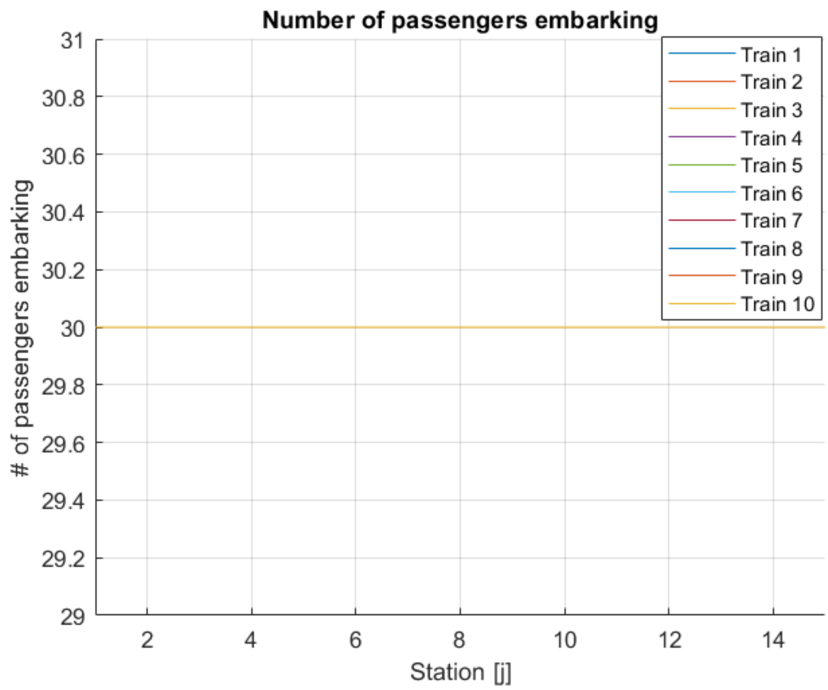


Figure 6-5: Number of passengers embarking each train at each station of the uniform timetable simulation of the AURS with 10 trains and 15 stations

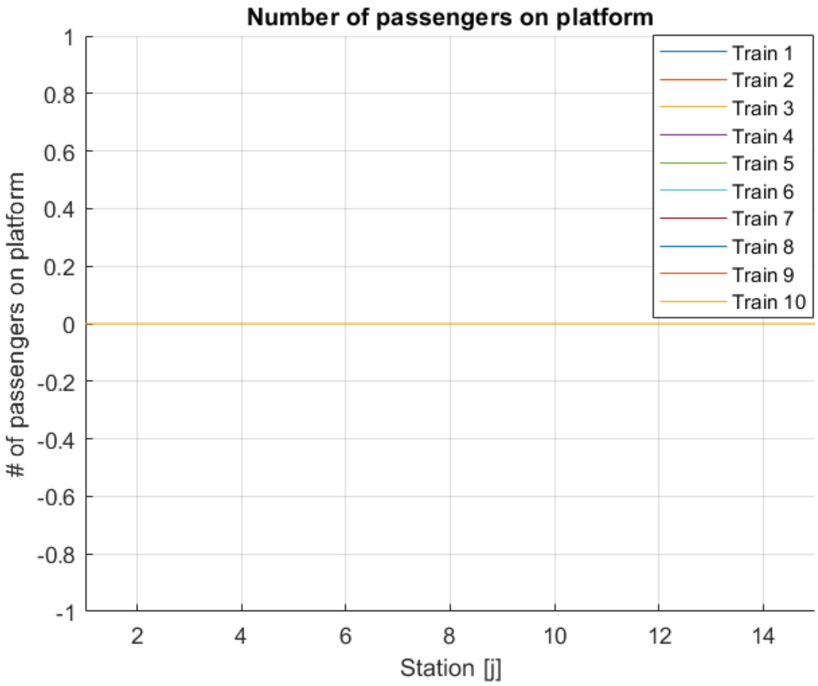


Figure 6-6: Number of passengers at each platform after each train $[k]$ has left station $[j]$ of the uniform timetable simulation of the AURS with 10 trains and 15 stations

The aim of augmenting the URS to this simulated system was to incorporate dynamics that could describe a real-life scenario much more truthfully. However, some necessary assumptions needed to be made in order to be able to simulate the system according to a uniform timetable;

1. The initial conditions assume there are passengers disembarking at the first station, which is not physically possible as these passengers had to board somewhere. This was still a valid design choice, as letting 0 passengers disembark the first will disturb the uniformity, and make the system unstable.
2. Theoretically, it is not possible for passengers to embark at the last station, as these passengers cannot travel any further. However, allowing passengers to embark at the last station allows for preservation of uniformity and stability of the system.

The physical interpretation of the AURS that makes sense including these assumptions, as described in Section 6-6, is that a short segment, of $j \in \{1, 2, \ldots, 15\}$, of a very long Urban Railway line $J >> 15$ is being inspected with regards to its dynamic behaviour. Therefore, the "cornercases" such as the physical limitations of the first and last station regarding the dynamics, can be disregarded.

6-8 Stability of the Augmented Urban Railway System

The concept of stability of explicit, and implicit MMPS systems was thoroughly discussed in Chapter 3. Determining whether the AURS is bounded-buffer stable for certain initial conditions, is very valuable. The buffer of the system is the difference between time states in each event k. The notion of boundedness in the definition of stability for discrete event systems refers to the buffer levels, at an average, taking constant values [8]. The overall purpose is to not let any state of the system overflow, which is the case if the MMPS system is bounded-buffer stable. In the previous section, Section 6-7, the AURS was simulated with fixed-point x_e described by equations 6-60, 6-61, and 6-62. In this section, the bounded-buffer stability of the simulated system is analyzed. Furthermore, the bounded-buffer stability of the fixed-points and growth rate obtained by solving the LPPs as per 6-48 is analyzed and commented on. In order to conclusively determine bounded-buffer stability of any implicit MMPS system with growth rate λ, the system must be normalized, and subsequently, linearized. Normalization is done using the theory provided in Subsection 3-3-1. The general expression for a normalized implicit MMPS system is given by the following expression, which is identical to the expression provided in 3-26;

$$\tilde{x}(k) = \tilde{A} \otimes \left(\tilde{B} \otimes' \left(C \cdot \tilde{x}(k-1) + D \cdot \tilde{x}(k) \right) \right) \tag{6-64}$$

Having obtained this normalized form, the normalized system can be linearized using the theory provided in Section 3-5. The definition of a linearized system as given in Section 3-5 is repeated below;

Definition 6-8.1. *[15](Linearized MMPS system) Any normalized implicit MMPS system can be recast as a linear system in conventional algebraic notation for all $\tilde{x}_\theta(k) \in \Omega_\theta, k \in \mathbb{Z}^+$ using the following expressions;*

$$\begin{aligned} \tilde{x}_\theta(k) &= M_\theta \cdot \tilde{x}_\theta(k-1) \\ M_\theta &= (I - M_1)^{-1} \cdot M_2 \\ M_1 &= G_{\mathrm{A}_\theta} \cdot G_{\mathrm{B}_\theta} \cdot D \\ M_2 &= G_{\mathrm{A}_\theta} \cdot G_{\mathrm{B}_\theta} \cdot C \end{aligned} \tag{6-65}$$

If the inverse of $(I - M)$ exists.

The polyhedron Ω_θ is the region for which this linearization is valid. The next Section will thoroughly examine the polyhedron Ω_θ for various bounded-buffer stable linearized systems. A linearized implicit MMPS system of the form given in 6-65 is;

- Bounded-buffer stable if the system matrix M_θ only has multiplicative eigenvalues smaller than, or equal to 1. All multiplicative eigenvalues of magnitude 1 have to have corresponding Jordan blocks of size 1×1. Equivalently, the multiplicative eigenvectors corresponding to multiplicative eigenvalues of magnitude 1 must be independent.

- Unstable if either, at least one multiplicative eigenvalue of M_θ is greater than 1, or the multiplicative eigenvectors of associated with the multiplicative eigenvalues of magnitude 1 are not independent.

This condition is identical to the one given in Subsection 3-5-2.

6-8-1 Bounded-Buffer Stability of the Simulated AURS

By simply looking at the figures obtained by simulating the AURS using fixed-point x_e given by 6-60, 6-61, and 6-62, the time states seem to constantly adhere to a buffer level of 60 seconds, which coincides with the growth rate of $\lambda_e = 60$ seconds. Furthermore, the quantity states do not grown at all over time for any station or any train, as is visible in Figures 6-3, 6-4, 6-5 and 6-6. This information subtly points to the system possibly being bounded-buffer stable. However, this cannot be definitively concluded by simply looking at the figures. The AURS is linearized around the aforementioned fixed-point $x(1) = x_e$ with $\lambda_\theta = \lambda = 60$. For the AURS as in 6-31, with $J = 15$ stations, the matrix M_θ will be of size $M_\theta \in \mathbb{R}^{90\times90}$, as 6 states per station, for 15 stations leads to a a total of 90 states. The inverse of $(I - M)$ surely exists since for this system, in Section 6-4, it is proven that a strictly lower-triangular matrix F exists. When attempting to prove linearized system $\tilde{x}_\theta = M \cdot \tilde{x}_\theta(k-1)$ is bounded-buffer stable, the bounded-buffer stability condition as defined in 6-8 must hold. When applying this theory to the AURS, the following conclusions can be drawn;

- Matrix $M_\theta \in \mathbb{R}^{90\times90}$ has 11 multiplicative eigenvalues of magnitude 1.
- Of these 11 multiplicative eigenvalues of magnitude 1, all of them have a corresponding Jordan block of size 1×1.
- No multiplicative eigenvalue of matrix M_θ has a magnitude larger than 1.
- Therefore, the simulated AURS is bounded-buffer stable.

The multiplicity of multiplicative eigenvalue 1 can be related to the rank deficiency of matrix H_{eq} [15]. The rank deficiency of this matrix for the AURS with $J = 15$ is equal to 11, which as expected, coincides with the multiplicity of multiplicative eigenvalue 1 in this matrix M_θ.

6-8-2 Bounded-Buffer Stability of the LPP Obtained Fixed-Points

Whereas the previous section proved the bounded-buffer stability of the simulated AURS, this section aims to investigate the bounded-buffer stability of the 40 fixed-points obtained by solving the LPPs in 6-48. The initial condition $x(1) \in \mathbb{R}^{90\times1}$ was proven to be a fixed-point for the AURS with $J = 15$ stations. Due to computational effort and running time constraints as discussed in Section 6-7, it is not possible to solve the LPPs for the AURS with $J = 15$. From analysis of the obtained fixed-points of solving the LPPs for the AURS with $J = 4$, 4 distinct fixed-points direction vectors have been found. Since the initial condition described by 6-60, 6-61 and 6-62 was a fixed-point for $J = 15$, it is worth considering using the same parametric initial condition, but for $J = 4$ and investigating whether this initial condition might be a fixed-point for the AURS with $J = 4$. This would then yield a fifth fixed-point direction vector s_5. This initial condition for $J = 4$ is given as follows;

$$x(1) = [\ x_1(1)^\intercal \quad x_2(1)^\intercal \quad x_3(1)^\intercal \quad x_4(1)^\intercal\ ]^\intercal \tag{6-66}$$

Substituting this $x(1)$ as x_e, into equation 3-25, proposed $x(1)$ turns out to be a fixed-point as well. From this finding, a fifth direction vector g_5 can be determined, leaving "only" 6 direction vectors unknown. So, in total there are 41 known fixed-points corresponding to growth rate $\lambda = 60$. The bounded-buffer stability of the linearized system around each of these fixed-points will be determined through the same reasoning as was carried out in Subsection 6-8-1.

All 41 matrices H_{eq} corresponding to all 41 unique combinations G_A and G_B will be evaluated. It turns out the rank deficiency of H_{eq} is equal to 11 for all 41 G_A, G_B combinations. From the linearization of the AURS around these 41 fixed-points, the following conclusions can be drawn;

- 10 Out of the 41 linearizations are bounded-buffer stable. Each matrix M_θ corresponding to the bounded-buffer stable systems have 11 multiplicative eigenvalues of value 1.
- Out of these 10 bounded-buffer stable systems, 9 are linearizations around the fixed-points obtained from solving the LPP, and 1 is the linearization around fixed-point chosen for the uniform simulation with $J = 4$.
- All 31 unstable system have more than 11 multiplicative eigenvalues 1 in their respective matrices M_θ.
  - 15 Out of these 31 matrices M_θ corresponding to the unstable systems have 12 multiplicative eigenvalues of value 1.
  - 12 Out of these 31 matrices M_θ corresponding to the unstable systems have 13 multiplicative eigenvalues of value 1.
  - 4 Out of these 31 matrices M_θ corresponding to the unstable systems have 14 multiplicative eigenvalues of value 1
- None of the 41 matrices M_θ have a multiplicative eigenvalue larger than 1

So a very large number of fixed-points do not yield a bounded-buffer stable linearized system. The 40 fixed-points found by solving the LPPs 6-48 all had unique corresponding footprint matrix combinations G_{A_θ} and G_{B_θ}. These 40 unique combinations were formed from 12 different matrices G_{A_θ} and 4 different matrices G_{B_θ}. The footprint matrix combination G_{A_θ} and G_{B_θ} corresponding to the fixed-point x_e of the uniformly simulated system consists of matrices G_{A_θ} and G_{B_θ} that did not yield a solution when solving the LPPs. In total, 13 unique matrices G_{A_θ} and 5 unique matrices G_{B_θ} create the 41 footprint matrices corresponding to the 41 fixed-points and growth rate of $\lambda = 60$. The rank deficiency of matrix H_{eq} of the simulated AURS with $J = 15$ is also 11, suggesting the dynamics introduced by adding more stations, did not cause the rank deficiency of H_{eq} to grow.

6-9 Maximal Invariant Set of the Augmented Urban Railway System

In the previous Section, bounded-buffer stability was proven for 10 out of the 41 linearized systems. These systems were all linearized around their respective fixed-points $x_{e,i}, i \in \{1, 2, \ldots, 41\}$. Furthermore, the AURS simulated from a uniform fixed-point with $J = 15$ stations was linearized, and stability was proven. The mapping between the normalized system 3-26 and the linearized system 3-46 is valid for all $\bar{x}_\theta \in \Omega_\theta$. A more elaborate definition of polyhedron Ω_θ is given in Section 3-6. The definition of a maximal invariant set as per 3-7.3 is repeated here;

Definition 6-9.1. *[2] (Maximal invariant set $\mathcal{O}_\infty$) The set $\mathcal{O}_\infty \subseteq \Omega_\theta$ is considered the maximal invariant set of the autonomous system as in 3-46, if $\mathcal{O}_\infty$ is invariant, and $\mathcal{O}_\infty$ contains all the invariant sets contained in Ω_θ*

In this section, we aim to find the maximal invariant set for the uniformly simulated AURS with $J = 15$ stations, and the 10 bounded-buffer stable linearizations of the AURS with $J = 4$ stations. The region, or polyhedron Ω_θ where this linearization is valid is obtained from [15];

$$\Omega_\theta = \{x \in \mathbb{R}^n | H \cdot x \leq h\} \tag{6-67}$$

Where matrix H and vector h are obtained as described in 3-3-2. Using Algorithm 4 from 3-7, we attempt to find $\mathcal{O}_\infty$. The algorithm is repeated below;

Algorithm 5 [15] Maximal positive invariant set

Input: M_θ, Ω_θ
Output: Ω_∞
$\mathcal{O}_0 \leftarrow \Omega_\theta, k \leftarrow -1$

repeat
$k \leftarrow k + 1$ $\mathcal{O}_{k+1} \leftarrow \text{Pre}(\mathcal{O}_k) \cap \mathcal{O}_k$

until $\mathcal{O}_{k+1} = \mathcal{O}_k$
$\mathcal{O}_\infty \leftarrow \Omega_k$

The precursor set to the set Ω_θ is given by [15];

$$\text{Pre}(\Omega_\theta) = \{x \in \mathbb{R}^{90} | H \cdot M \leq h\} \tag{6-68}$$

6-9-1 Maximal Invariant Set of Multiple Fixed-Points

Having concluded that 10 out of the 41 found fixed-points have a corresponding bounded-buffer stable linearization, it is useful to approximate the maximal invariant set for each. For each fixed-point, and their corresponding M_θ and $\Omega_\theta = \{x \in \mathbb{R}^{24} | H \cdot x \leq h\}$, Algorithm 5 is executed a maximum of 300 iterations. This number is arbitrarily chosen. An overview of the characteristics of each fixed-point is presented in Appendix C, where the last row represents the fixed-point as per 6-60, 6-61 and 6-62. From left to right, the following information about the numbered fixed-points is presented; multiplicity of multiplicative eigenvector of value 1 of M_θ, whether the linearized system is bounded-buffer stable, the rank deficiency of matrix H_{eq}, whether a maximum invariant set was found within 300 iterations, and if so, in how many iterations. A few interesting observations can be made about this data in this table. As expected, for all 10 bounded-buffer stable linearized systems, a maximal invariant set was found. Furthermore, this invariant set was reached within 2 to 4 iterations of Algorithm 5. Interestingly, for 14 out of the 31 unstable linearized systems, Algorithm 5 did converge within less than 300 iterations. However, it is obvious that the number of iterations after which Algorithm 5 converged is significantly larger than the number of iterations the algorithm took for the bounded-buffer stable systems to converge. Since Algorithm 5 is programmed to find **any** invariant set, for unstable systems, it is possible that the only way to ensure $\mathcal{O}_{k+1} = \mathcal{O}$, is to force the set $\mathcal{O}_\infty$ to become infinitely small, i.e. empty. Upon verification, it indeed becomes apparent that the maximal invariant sets $\mathcal{O}_\infty$ found for the unstable linearized systems are in fact, empty. Therefore, it can be concluded only bounded-buffer stable linearized systems have nonempty maximal invariant sets.

J	# e.v. 1 M_θ	BB stable	H_{eq} rank def.	$\mathcal{O}_\infty$ found	Iter	Empty
4	11	Yes	11	Yes	4	No
5	11	Yes	11	Yes	5	No
6	11	Yes	11	Yes	8	No
7	11	Yes	11	Yes	10	No
8	11	Yes	11	Yes	3	No
9	11	Yes	11	No	3	Yes
10	11	Yes	11	No	3	Yes
11	11	Yes	11	No	3	Yes
12	11	Yes	11	Yes	2	Yes
13	11	Yes	11	Yes	3	Yes
14	11	Yes	11	Yes	3	Yes
15	11	Yes	11	Yes	2	Yes

Table 6-3: System characteristics of the AURS with $J \in \{4, 5, \ldots, 15\}$ stations

6-9-2 Maximal Invariant Set of the Simulated System

The AURS as in 6-31, linearized around x_e as per 6-60, 6-61 and 6-62 for $J = 15$ stations was proven to be bounded-buffer stable in Subsection 6-8-1. Polyhedron Ω_θ for this system is defined as $\Omega_\theta = \{x \in \mathbb{R}^{90} | H \cdot x \leq h\}$. Within two iterations, a maximal invariant set $\mathcal{O}_\infty$ is found. Upon verificiation, it unfortunately becomes apparent this found maximal invariant set is empty. In the previous subsection, it was concluded that the linearization around the same fixed-point, but with $J = 4$ instead of $J = 15$, did have a nonempty maximal invariant set after 4 iterations. Let us therefore analyze the AURS for $j \in \{4, 5, 6, \ldots 14, 15\}$, as to determine from which number of stations, an empty maximal invariant set is found. The following characteristics of the same parametric initial condition 6-60 for each value of J will be determined;

- Whether the linearization of the AURS around $x(1)$ is bounded-buffer stable
- The multiplicity of multiplicative eigenvalue 1 of M_θ
- The rank deficiency of H_{eq}
- Whether an invariant set $\mathcal{O}_\infty$ can be found
- In how many iterations (Iter) this invariant set $\mathcal{O}_\infty$ was found

Firstly, the number of multiplicative eigenvalues 1 is equal to 11, no matter how many stations were added, and the rank of deficiency of H_{eq} stays 11 even when extra dynamics are added. Also, all linearized systems are bounded-buffer stable. So, nor the bounded-buffer stability, nor the number of multiplicative eigenvalues 1, nor the rank deficiency of H_{eq} is influenced when dynamics of additional stations are added. However, it becomes apparent that, from 8 stations onward, no nonempty maximal invariant set can be found. Intuitively, there would be no reason for there to only be empty maximal invariant sets after 8 stations. However, it is explicable why this phenomenon might occur. Bounded-buffer stability does not guarantee the existence of a nonempty maximal invariant set. As the number of stations grows, the number of inequalities in $H \cdot x \leq h$ grows, and the dimension of H grows, there are simply more constraints the maximal invariant set must satisfy.

Furthermore, since none of the matrices M_θ have all multiplicative eigenvalues **strictly** less than 1. Any eigenvector v of this M_θ satisfies $M_\theta \cdot v = v$. Another interesting takeaway from this table is, bounded-buffer stability is conserved for AURS with larger number of stations. Adding more stations to the system makes analysis more computationally expensive. When analyzing implicit MMPS systems consisting of multiple nodes/stations that have identical dynamics, as is the case for the AURS, it might be interesting to see whether the dynamics and characteristics of a system with a lot of nodes/stations can be determined by simply analyzing the same system with much less nodes/stations, improving the computational efficiency of analysis.

Chapter 7

Disturbance and Control of the Augmented Urban Railway System

This chapter examines how the AURS responds to various disturbances, such as changes in passenger arrival and boarding rates. In Section 7-1, four different types of small, momentary disturbances are applied to the AURS simulated according to a uniform timetable. The propagation of the states of the disturbed system are discussed as well. Thereafter, Section 7-2 proposes an offline, open-loop control strategy using the framework proposed in 5 designed to reject the applied disturbances, and restore uniform behaviour. Lastly, the results of this applied control strategy are discussed. In chapter 6, an augmented mathematical model for describing an urban railway line was derived, modeled, and analyzed. Using an initial condition that was proven to be a fixed-point, the AURS was simulated according to a uniform timetable. The propagation of the states, and the train trajectories in this simulation are shown in Figures 6-2, 6-3, 6-4, 6-5, and 6-6. The aim of augmenting the existing URS in the first place was to model the system in such a way that it better resembles reality, which was achieved by allowing for more complex flows of passengers traveling to multiple destinations. In the pursuit of simulating the system according to the uniform timetable, assumptions regarding the initial and final stations were done to accommodate this uniformity. These assumptions are justified and elaborated on in 6-7. In this chapter, the fragility of the system's uniform behaviour will be shown by applying multiple (small) disturbances to the uniformly simulated system. In a real-life scenario, such disturbances will inevitably occur. Therefore, the disturbed system will be analyzed, and a control strategy will be applied in the pursuit of rejecting said disturbances.

7-1 Disturbing the AURS

In [18], the original URS was disturbed by introducing a sudden momentary decline in passengers arriving to the 5$^{\text{th}}$ station when the 5$^{\text{th}}$ train arrives. The arrival rate decreased from $e_5 = 0.5$ passengers/s to $e_5 = 0.3$ passengers/s. By applying this small disturbance, the uniformity was disturbed, and the quantity states started growing, yielding an unstable system. By applying four different types of disturbance to the uniformly simulated AURS, an analysis of its response to such disturbances can be done. The disturbances applied are;

- A momentary drop of the arrival rate $e_{j,k}$ from 0.5 passengers/s to 0.3 passengers/s for station 5 train 5. All other values of $e_{j,k}$ remain 0.5 for all stations and all trains. This is the same disturbance as applied to the original URS [18].

- A momentary surge of the arrival rate $e_{j,k}$ from 0.5 passengers/s to 0.7 passengers/s for station 4 train 6. All other values of $e_{j,k}$ remain 0.5 for all stations and all trains.

- A momentary drop of the disembark rate f from 2 passengers/s to 1.5 passengers/s for station 4 train 6. All other values of f remain 2 for all stations and all trains.

- A momentary surge of the embark rate b from 2 passengers/s to 2.5 passengers/s for station 5 train 6. All other values of b remain 2 for all stations.

The figures showing the propagation of the train trajectories, and the quantity state $\sigma_j(k)$ are given by the figures below, as these figures nicely demonstrate how the quantity state $\sigma_j(k)$ will grow unbounded over time, after the disturbances have been applied. The divergence in train trajectories demonstrates the non-uniformity of the simulation after the disturbances have been applied.

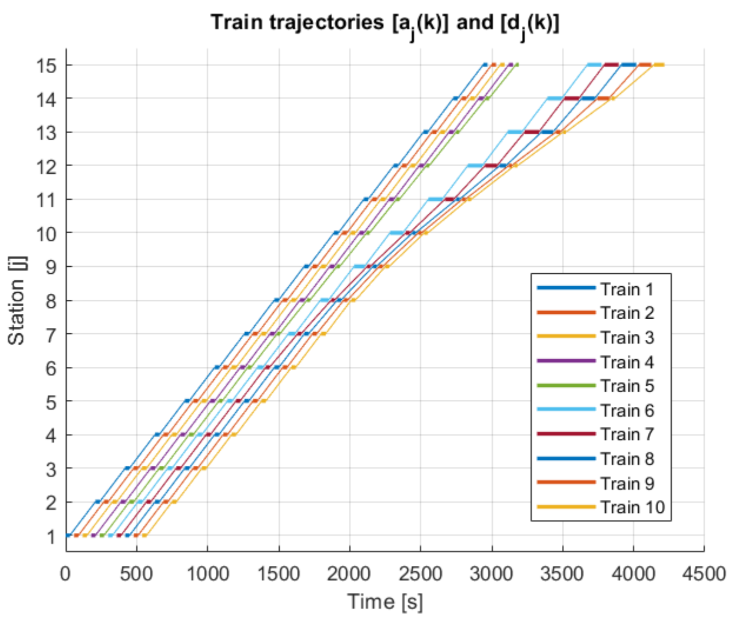


Figure 7-1: Train trajectories of the uniform timetable simulation of the AURS with 10 trains and 15 stations, disturbed with disturbance 1

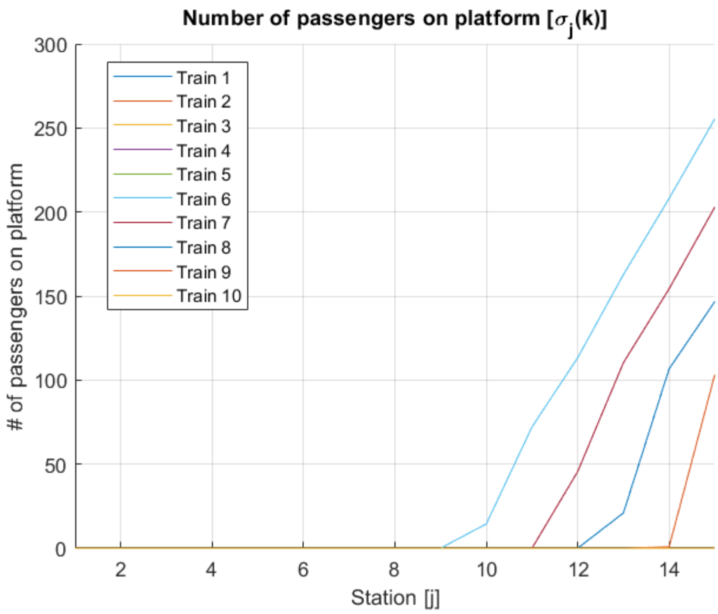


Figure 7-2: Number of passengers left on the platform of each station after each train has departed of the simulation of the AURS with 15 stations and 10 trains, disturbed with disturbance 1

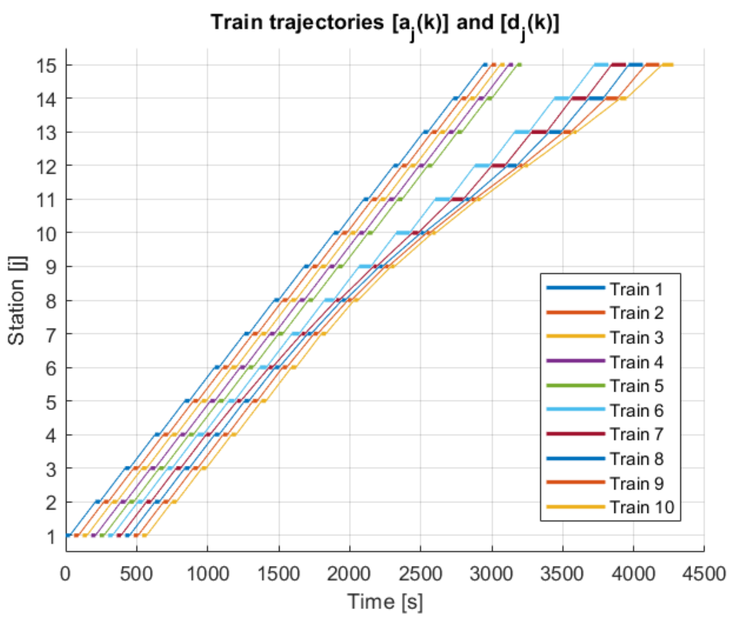


Figure 7-3: Train trajectories of the uniform timetable simulation of the AURS with 10 trains and 15 stations, disturbed with disturbance 2

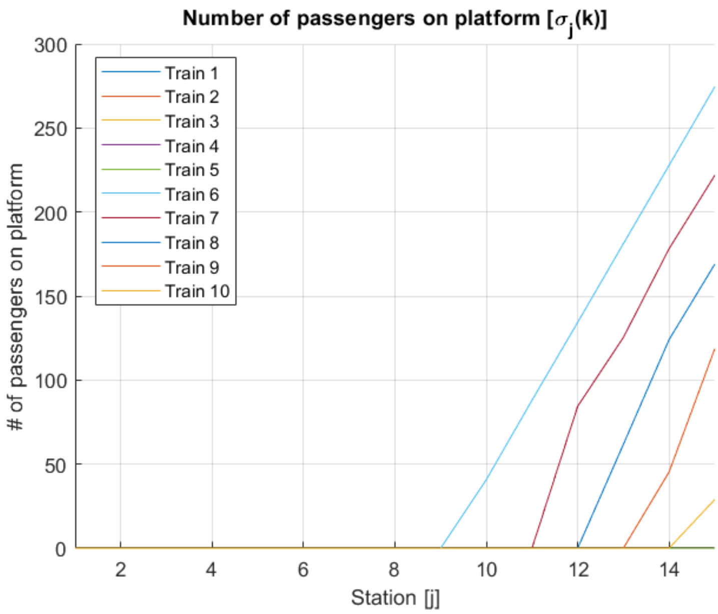


Figure 7-4: Number of passengers left on the platform of each station after each train has departed of the simulation of the AURS with 15 stations and 10 trains, disturbed with disturbance 2

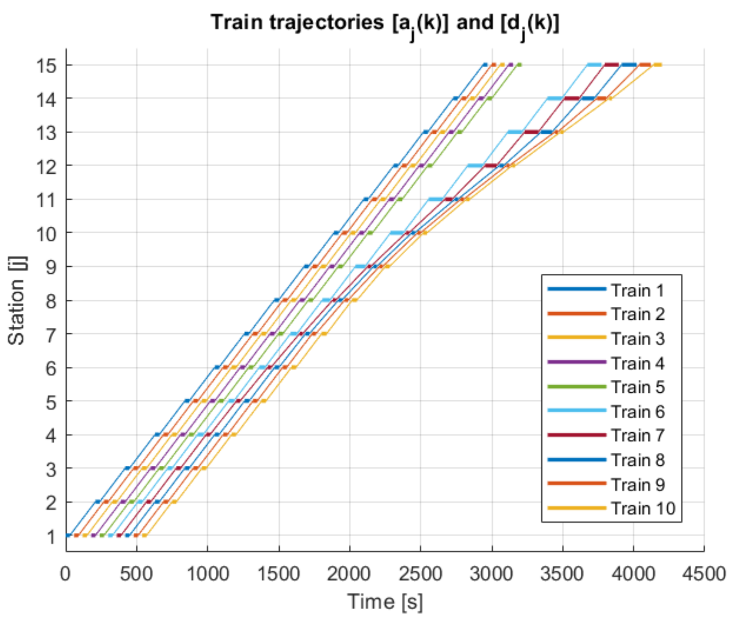


Figure 7-5: Train trajectories of the uniform timetable simulation of the AURS with 10 trains and 15 stations, disturbed with disturbance 3

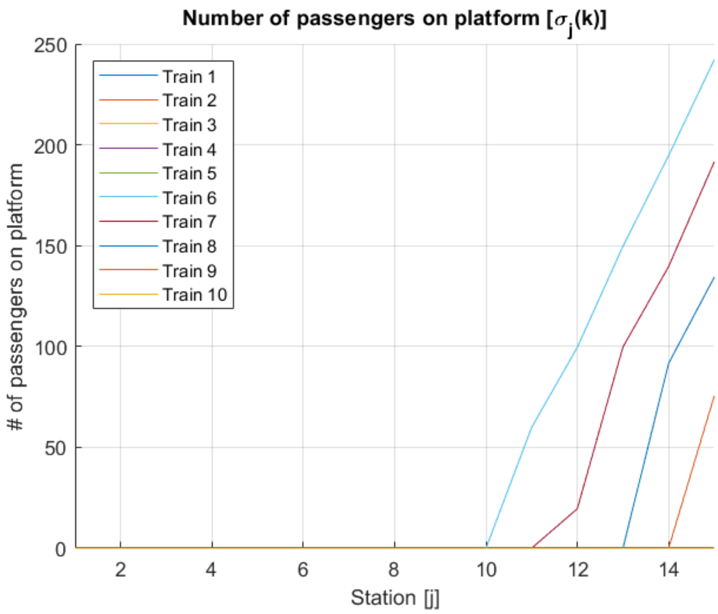


Figure 7-6: Number of passengers left on the platform of each station after each train has departed of the simulation of the AURS with 15 stations and 10 trains, disturbed with disturbance 3

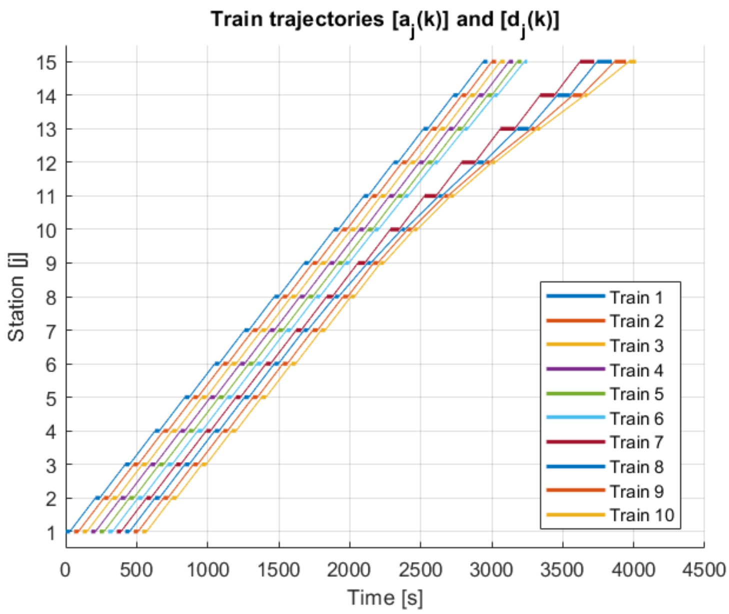


Figure 7-7: Train trajectories of the uniform timetable simulation of the AURS with 10 trains and 15 stations, disturbed with disturbance 4

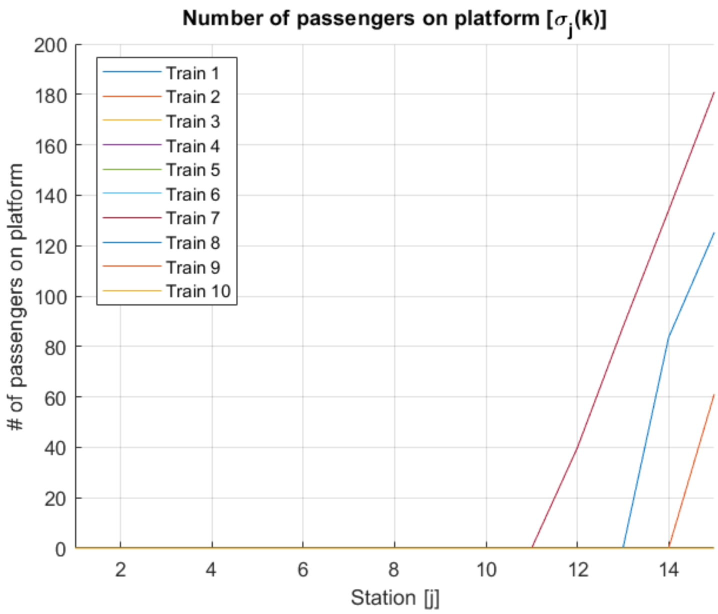


Figure 7-8: Number of passengers left on the platform of each station after each train has departed of the simulation of the AURS with 15 stations and 10 trains, disturbed with disturbance 4

The propagation of the other three quantity states $\beta_j(k)$, $\Delta_j(k)$ and $\rho_j(k)$ for these four disturbed systems are visible in Appendix A. It is immediately visible that for each type of disturbance, whether it be a surge or a drop of the value of a variable, the system becomes unstable. The disturbed original URS as in [18], has been controlled using a model predictive controller, and the applied disturbance has been successfully rejected, stabilizing the system. The next section will propose, apply, and analyze a control strategy for the AURS.

7-2 Controlling the URS

In this section, an Open-Loop (OL) control strategy to stabilize the disturbed AURS will be proposed, applied, and analyzed, respectively. Chapter 5 provided a mathematical framework for applying control strategies to implicit MMPS systems. The AURS is implicit, and therefore, the results from Chapter 5 will apply, and be of use in the process of constructing a disturbance rejection controller.

7-2-1 Control Structure

Similarly to the control strategy applied to the original URS, the control strategy of the AURS relies on shortening or elongating the running time of train k between station $j-1$ and j. The state equation for the arrival time of train k at station j, $a_j(k)$ is given by;

$$a_j(k) = \max(d_{j-1}(k) + \tau_r, d_j(k-1) + \tau_H) \tag{7-1}$$

When applying control input signal $u_j(k)$, this equation can be rewritten as;

$$a_j(k) = \max(d_{j-1}(k) + \tau_r + u_j(k), d_j(k-1) + \tau_H) \tag{7-2}$$

The control input is defined as a time difference with a value between -20 and 20 seconds, $-20 \leq u_j(k) \leq 20$. So the unit is seconds (s), however, this control input signal is a quantity signal. This may sound counterintuitive, as the distinction between temporal signals and quantity signals could lead us to assume otherwise. However, the applied input signal does not show temporal signal behaviour, as it is always bounded between -20 s and 20 s and does not grow steadily as the event counter continues. Therefore, the applied input signal is a quantity signal. Furthermore, what is defined as a temporal signal is not necessarily a signal whose unit is seconds (s), hours (h) or some other time quantity, but rather a signal that is nondecreasing in nature as the event counter k grows.
The system equations of the OL controlled AURS, using the parameters proposed in 6-2 can be given by the following equations;

$$\begin{aligned}
a_j(k) &= \max(d_{j-1}(k) + \tau_r + u_j(k), d_j(k-1) + \tau_H) \\
d_j(k) &= a_j(k) + \frac{\beta_j(k)}{f} + \frac{\Delta_j(k)}{b} \\
\rho_j(k) &= \rho_{j-1}(k) - \beta_j(k) + \Delta_j(k) \\
\sigma_j(k) &= \max(0, (\sigma_j(k-1) + \gamma(a_j(k) + d_j(k-1) + \frac{\beta_j(k)}{f})) - (\rho_{\max} + \beta_j(k) - \rho_{j-1}(k))) \\
\beta_j(k) &= \zeta_{j-2,k}\Delta_{j-2}(k) + \zeta_{j-1,k}\Delta_{j-1}(k) \\
\Delta_j(k) &= \min(\rho_{\max} + \beta_j(k) - \rho_{j-1}(k), \sigma_j(k-1) + \gamma(a_j(k) - d_j(k-1) + \frac{\beta_j(k)}{f}))
\end{aligned} \tag{7-3}$$

From these state-space equations, the ABCDE form can be constructed.

7-2-2 ABCDE Form of the OL Controlled System

The set of state-space equations in 7-3 can be transformed in an ABCDE matrix form as per 5-1. The proposed control strategy is an OL control strategy according to strategy 1, where the input signal is implemented in the scaling stage. The corresponding ABCDE form of this system with the proposed control input can therefore be given by;

$$x_(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k) + E \cdot u(k))) \tag{7-4}$$

In Section 6-3, the ABCD form of this exact system was given. Since the applied control input signal does not affect either the dimensions, or the entries of the ABCD matrices, only matrix E will have to be specified in order to fully define the ABCDE form. Recall that input signal $u_j(k)$ can be described by;

$$u_j(k) = \begin{bmatrix} u_{j,t}(k) \\ u_{j,q}(k) \end{bmatrix} \tag{7-5}$$

$$E = \begin{bmatrix} \bar{E}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \bar{E}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \bar{E}_J \end{bmatrix} \tag{7-6}$$

With $j \in \{1, 2, \ldots, J\}$. All submatrices $\bar{E}_j$ are identical, and can be given by;

$$\bar{E}_j = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^\intercal \tag{7-7}$$

Applying the results of Chapter 5, time-invariance and solvability can be determined for the OL controlled system. Time-invariance will be determined by applying the results from 5-2-2, where the OL controlled AURS is time-invariant if the following holds;

$$\begin{aligned} \sum_{i \in \overline{n_\mathrm{t}}} \begin{bmatrix} C_{11} & D_{11} & E_{11} \end{bmatrix}_{\ell i} &= 1, \forall \ell \in \overline{p_\mathrm{t}} \\ \sum_{i \in \overline{n_\mathrm{q}}} \begin{bmatrix} C_{21} & D_{21} & E_{21} \end{bmatrix}_{ti} &= 0, \forall t \in \overline{p_\mathrm{q}} \end{aligned} \tag{7-8}$$

However, since there are no temporal input signals, and therefore, matrices E_{11} and E_{21} are all $\mathbf{0}$ it must hold that;

$$\begin{aligned} \sum_{i \in \overline{n_\mathrm{t}}} \begin{bmatrix} C_{11} & D_{11} \end{bmatrix}_{\ell i} &= 1, \forall \ell \in \overline{p_\mathrm{t}} \\ \sum_{i \in \overline{n_\mathrm{t}}} \begin{bmatrix} C_{21} & D_{21} \end{bmatrix}_{ti} &= 0, \forall t \in \overline{p_\mathrm{q}} \end{aligned} \tag{7-9}$$

The time-invariance condition is now reduced to the time-invariance condition for implicit MMPS systems as per 3-10. Time-invariance for the uncontrolled AURS was already proven in 6-3-1, therefore, the OL controlled AURS using this input signal is also time-invariant. Since the proposed control method is open-loop control, solvability of the controlled implicit MMPS system will never be violated as per 5-2-3.

7-2-3 Control Methods

Having derived the system properties of the OL controlled AURS, the optimization problem can be defined. Firstly, an objective function will have to be derived. In the control method used for controlling the original URS, the following performance signal was defined [18];

$$\mathrm{p}_j^{\mathrm{wait}}(k) = e_{j,k}(a_j(k) - d_j(k-1) + \sigma_j(k-1)) \tag{7-10}$$

The reference value of this performance signal is given as $\mathrm{p}_{ref}^{wait} = 15$ passengers. The performance signal will be used in the objective function for this optimization problem as well. Let us define the objective function for this optimization as follows;

$$J(k) = ||\mathrm{p}^{wait}(k) - \mathrm{p}_{ref}^{wait}||_1 + ||\rho(k) - \rho_{ref}||_1 + w_u||u(k)||_1 \tag{7-11}$$

This objective function consists of adding three terms. The first term is the absolute difference between the reference signal p_{ref}^{wait} and the performance signal, summed over each station for train k. The second term is the absolute difference between the number of passengers in the train after leaving the station, and its reference value $\rho_{ref} = 45$, summed over each station for train k. Lastly, the control input is penalized slightly with trade-off weight $w_u = 0.001$. Hereby, the formal optimization problem can be defined as follows;

$$\begin{aligned} \min_{u(k)} \quad & J(k) = ||\mathrm{p}^{wait}(k) - \mathrm{p}_{ref}^{wait}||_1 + ||\rho(k) - \rho_{ref}||_1 + w_u||u(k)||_1 \\ \text{s.t.} \quad & x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k) + E \cdot u(k))) \\ & u_j(k) \leq 20 \\ & -20 \leq u_j(k) \\ & x(1) = x_e \\ & j \in \{1, 2, \dots, 15\} \\ & k \in \{2, \dots, 10\} \end{aligned} \tag{7-12}$$

Please note the state-space equation for $x(k)$ is given in this form as to compactly present the optimization problem. These state-space equations can be recast as a set of linear constraints. Furthermore, initial condition x_e is the initial condition described in Section 6-7, used to uniformly simulate the system, ensuring the undisturbed system will operate according to the uniform timetable. Also, as a direct consequence of the way the system is initialized, the input signal $u_j(k)$ is 0 for the all stations of the first train, and the first and second station of all trains. In Sections 6-5 and 6-6, the initial conditions were given as parametric equations, and their respective ABCD forms. These parametric equations and ABCD forms do not include an input signal $u_j(k)$. This problem is an Mixed Integer Linear Programming (MILP) problem, as all constraints are linear, and the system itself can be recast as a continuously piecewise affine system [7]. Therefore, the global optimum of this optimization problem can be found. Furthermore, this optimization is an offline optimization, as a single optimization is performed, and all parameters, constraints and states are known in advance. The result of the optimization is an optimal input sequence $u_j(k)$ for each station and train. A linear solver in Matlab can be used to solve this problem. The analysis described in this chapter has been carried out by solving the optimization problems in Matlab using a Gurobi solver. In the next section, the results of the optimization for the four disturbed systems will be presented and analyzed.

7-3 Results and Observations

In this section, the results of solving the optimization problem for the four disturbed systems given in 7-1 will be presented, and discussed. Each disturbed system will be discussed separately, after which a general conclusion about the proposed control strategy can be drawn. However, there are a few general conclusions that can be drawn that hold for all four controlled systems;

- The value of state $\sigma_j(k)$ never exceeds 0, as no passenger is ever left at any station j when any train k departed as visible in 7-9, B-7, B-12, and B-17. This is to be expected, and desired, as the trains are never full, which is visible in B-3, B-8, B-13, and 7-12.
- All quantity states converge to their steady-state value after the disturbance was attenuated. In the case of the uncontrolled disturbed systems, all quantity states grew over time after the system was disturbed.
- The trains that ran undisturbed, i.e. the disturbance occurred at a later train, all have $u_j(k) = 0$ for all stations. This is because their trajectory is still undisturbed, and runs uniformly, eliminating the need for any control input signal.
- The optimal objective values are significantly lower for the third en fourth disturbed system, as the system deviated much less from the uniform simulation. Their control effort was also much lower, strengthening this argument.

More in depth conclusions regarding the specifics of each controlled system are provided in the following subsections;

7-3-1 Control of Disturbance 1

The AURS disturbed by disturbance 1 had a momentary drop of inflow of passengers at station 5 when train 5 arrived. The inflow of passengers $e_{5,5}$ momentarily dropped from 0.5 passengers/s to 0.3 passengers/s. The figure showing the applied control effort is given in Figure B-2;

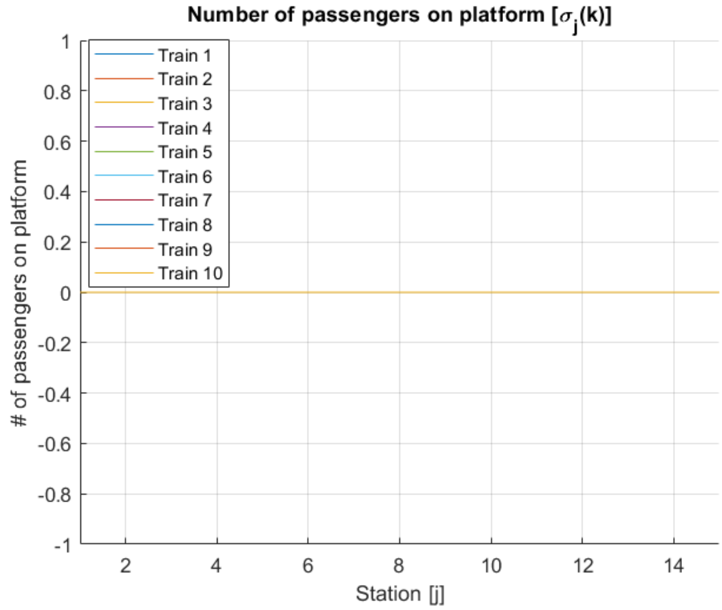


Figure 7-9: Number of passengers left on the platform of each station after each train has departed of the simulation of the controlled AURS with 15 stations and 10 trains, disturbed with disturbance 1

The figures showing the propagation of the quantity states, and the train trajectories having applied the optimal control are visible in Appendix B-1. By applying the proposed control strategy, and solving the optimization for the optimal control input signal $u_j(k)$, the disturbance could be attenuated. The used Gurobi solver in Matlab took 5.3 seconds to solve this optimization problem. Furthermore, the value of the objective function is 11.5726, which is the optimal value for the optimization problem. By closely examining the figures in Appendix B-1, the following conclusions can be drawn;

- The number of passengers in train 5 dropped significantly after the disturbance occurred, which is to be expected if the inflow of passengers decreases.
- The optimal input signal shows some form of oscillatory behaviour within the provided bounds as it aims to stabilize the system. The oscillations die out as the states converge to their steady-state value.
- The number of passengers disembarking, represented by state $\beta_j(k)$, logically also drops when the number of passengers embarking decreases.
- The value of the control input signal reaches the upper bound of 20 seconds for the 5^{th} train on the 5^{th} station, which is exactly when the disturbance occurs.

7-3-2 Control of Disturbance 2

The AURS disturbed by disturbance 2, which consisted of a momentary surge of inflow of passengers at station 4 when train 6 arrived. The inflow of passengers $e_{4,6}$ increased from 0.5 to 0.7 passengers/s. The figure showing the applied control effort is given in Figure 7-10;

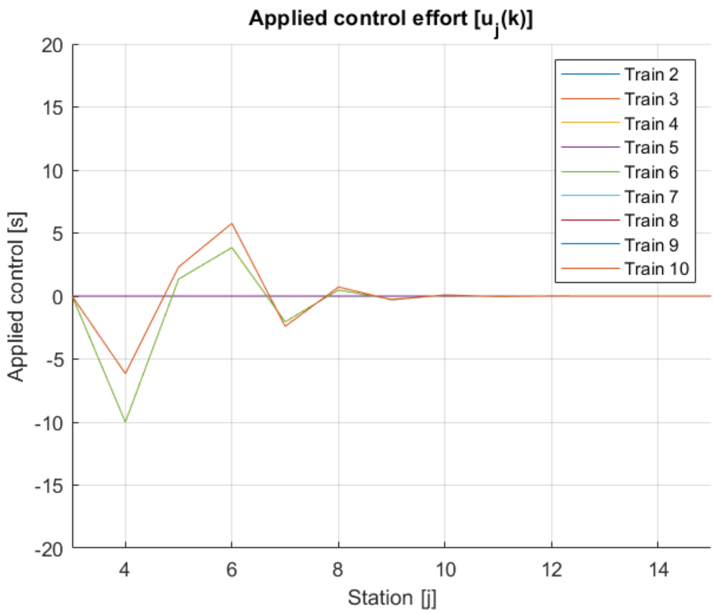


Figure 7-10: Input signal used for attenuating disturbance 2 applied to the AURS with 15 stations and 10 trains

The figures showing the propagation of the quantity states, and the trajectories having applied the optimal control are visible in Appendix B-2. By applying the proposed control strategy, and solving the optimization for the optimal control input signal $u_j(k)$, the disturbance could be attenuated. The used Gurobi solver in Matlab took 6.4 seconds to solve this optimization problem. Furthermore, the value of the objective function is 13.4530, which is the optimal value for the optimization problem. By closely examining the figures in Appendix B-2, the following insights can be obtained;

- The number of passengers in train 6 increased significantly after the disturbance occurred, which is to be expected if the inflow of passengers increased at that moment. This surge is attenuated by the control input signal.
- The value of state $\beta_j(k)$ also oscillates, which is to be expected. This is because the number of passengers embarking is logically tied to the number of passengers disembarking. As more passengers embark, more will eventually disembark.
- The optimal input signal again shows oscillatory behaviour within the provided bounds as it aims to stabilize the system. The oscillations of the control input signal die out as the states converge to their steady-state value.
- The magnitudes of the control input signals are significantly lower than for the controlled system disturbed by disturbance 1.

7-3-3 Control of Disturbance 3

The AURS disturbed by disturbance 3, which consisted of a momentary drop of the disembark rate of passengers at station 4 when train 6 arrived. The disembark rate f momentarily decreased from 2 passengers/s to 1.5 passengers/s. The figure showing the applied control effort is given in Figure 7-11 ;

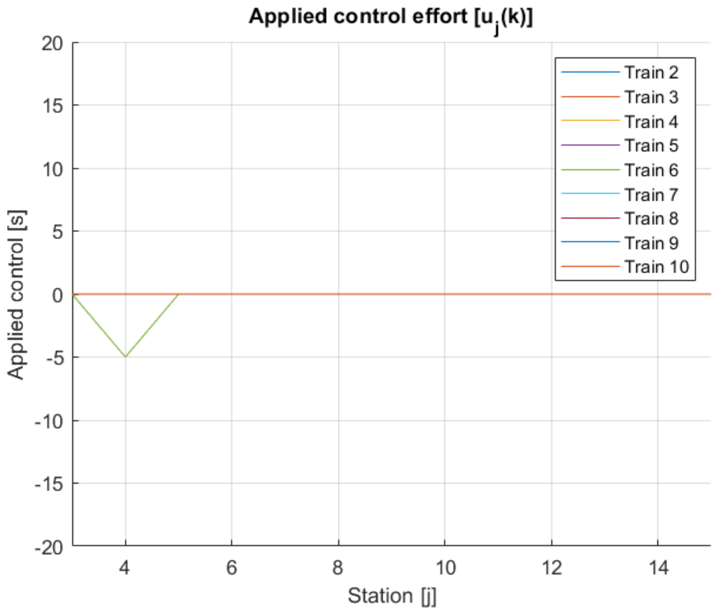


Figure 7-11: Input signal used for attenuating disturbance 3 applied to the AURS with 15 stations and 10 trains

The figures showing the propagation of the quantity states, and the train trajectories having applied the optimal control are visible in Appendix B-3. By applying the proposed control strategy, and solving the optimization for the optimal control input signal $u_j(k)$, the disturbance could be attenuated. The used Gurobi solver in Matlab took 5.5 seconds to solve this optimization problem. Furthermore, the value of the objective function is 2.5050, which is the optimal value for the optimization problem. By closely examining the figures in Appendix B-3, the following insights can be obtained;

- Contrary to the previous two controlled systems, this controlled system does not have a significant drop or surge in passengers in the trains, passengers embarking, or passengers disembarking. This can logically be explained, as in this case, the number of passengers embarking train 6 at station 4 does not increase. The applied disturbance does not change the number of passengers entering the system, just the speed at which they disembark. By making the train run slightly faster, this loss in time is accommodated for, and the number of passengers in the train remains the same.
- As visible in the figure for the number of passengers embarking and disembarking, there is a negligably small decrease in the number of passengers embarking and disembarking. Since any number of passengers can only be an integer, this numerical error can be disregarded.

- The optimal input signal shows a very small control effort at the time of disturbance, but other than that no oscillatory behaviour.

7-3-4 Control of Disturbance 4

The AURS disturbed by disturbance 4, which consists of a momentary surge of the boarding rate of passengers at station 5 when train 6 arrived. The embark rate b momentarily increases from 2 passengers/s to 2.5 passengers/s. The figure showing the applied control effort is given by Figure 7-12 ;

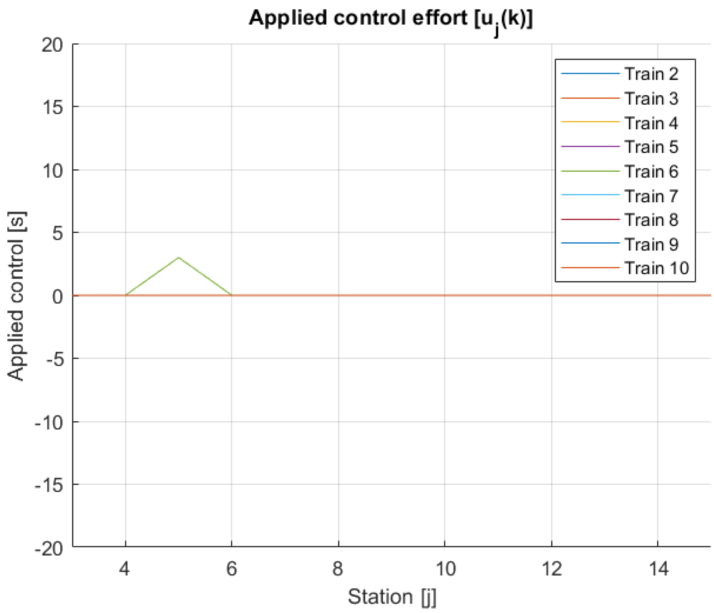


Figure 7-12: Input signal used for attenuating disturbance 4 applied to the AURS with 15 stations and 10 trains

The figures showing the propagation of the quantity states, and the train trajectories having applied the optimal control are visible in Appendix B-4. By applying the proposed control strategy, and solving the optimization for the optimal control input signal $u_j(k)$, the disturbance could be attenuated. The used Gurobi solver in Matlab took 5.8 seconds to solve this optimization problem. Furthermore, the value of the objective function is 1.5030, which is the optimal value for the optimization problem. By closely examining the figures in Appendix B-3, the following insights can be obtained;

- Similar to the third controlled system, this controlled system does not have a significant drop or surge in passengers in the trains, passengers embarking, or passengers disembarking. Again, this can logically be explained, as in this case, the number of passengers embarking train 6 at station 4 does not increase. The applied disturbance does not change the number of passengers entering the system, just the speed at which they embark. By making the train run slightly slower, the time "saved" by the sped up boarding stage is accommodated for, and the number of passengers in the train remains the same.

- The optimal input signal shows a very small control effort at the time of disturbance, but other than that no oscillatory behaviour.

7-3-5 Conclusions on the Applied Control Strategies

As was elaborately described in the previous section, all applied disturbances could be attenuated by the proposed open-loop control strategy. Designing a controller requires taking into account a lot of considerations, and demands choices are being made which all have benefits and drawbacks. The chosen open-loop controller is fast, optimal, will not violate solvablity, and is easy to implement. However, it is not suitable to compensate for real-time disturbances or sudden system changes. A closed-loop controller such as a Model Predictive Controller (MPC) would be a better fit to attenuate real-time disturbances and system changes. Nevertheless, implementing a closed-loop controller is not a fix-all solution, as it brings about its own challenges. For example, it is much more computationally expensive, the design is more complex, and could cause the system to become unsolvable and unstable. The purpose of this chapter was to illustrate that disturbances applied to the AURS could be attenuated by some control strategy, and the choice was made to attempt to do so by applying an open-loop controller. An interesting future research opportunity would be to design, apply, and evaluate different control strategies.

Chapter 8

Conclusions and Contributions

This Chapter concludes the research carried out in this thesis, and reflects on how the obtained results answer the research questions posed in Chapter 1. These conclusions are separated into three parts, with Section 8-1 reflecting on research question 1 and its subquestions. Section 8-2 attempts to form an answer to research question 2 and subquestions, whereas Section 8-3 attempts to answer research question 3 and subquestions. Lastly, a concise overview of the academic contributions made by the work in this thesis work is given in Section 8-4.

8-1 On Solving Solvability

In this section, the first research question and its subquestions are answered by reflecting on the results from the research carried out in Chapter 4. Firstly, let us recall the first research question posed in 1-2. Thereafter, each subquestion, and ultimately, the first main research question is answered.

1. Is it possible to find a necessary solvability condition for implicit MMPS systems?
   (a) Can a graph-theoretic interpretation be used to understand, and generalize beyond the current algebraic criteria?
   (b) What degrees of solvability exist for implicit MMPS systems?
   (c) Is it possible to identify a method to classify all implicit MMPS systems with regards to their degree of solvability?

Having recalled the first research question, the subquestions are subsequently answered;

(a) Can a graph-theoretic interpretation be used to understand, and generalize beyond the current algebraic criteria?
   - In short, yes it can. Section 4-1 describes how structure matrix $S = S_A \cdot S_B \cdot S_D$ can be represented as an interconnection graph.

- The value of all non-0 entries in structure matrix S, i.e. the "weights" on the arcs in the interconnection graph, were proven to have significance as well. Theorem 4-2.4 was proven, and stated that the value of entry $[S]_{ij}$ referred to the number of times state $x_j(k)$ is implicitly included in the expression of state $x_i(k)$.
- It is proven that, the existing solvability condition as described in 3-2, holds if the interconnection graph does not contain any circuits. If the interconnection graph contains a circuit, the system may not be solvable.
- Furthermore, a method is proposed by which the existence of a circuit can be proven. Hereby, an different condition that is equally strong as the initial algebraic solvability condition is derived.
- The states are included in any circuit in the implicit MMPS system, together form the circuit subsystem, which is a starting port for further analysis of solvability for implicit MMPS systems.
- Also considering the concept of modes as introduced in 4-2.1, if a mode does not contain a circuit, it is solvable. Therefore, solvability can only be violated in modes that do contain a circuit. Multiple degrees of solvability emerge from analysis of the concept of circuit modes.

(b) What degrees of solvability exist for implicit MMPS systems?

- Section 4-2 proposes four different types of solvability, a system can either be; uniquely solvable, parametrically solvable, parametrically unsolvable, and strictly unsolvable.
- The initial solvability condition could determine whether a system is uniquely solvable. However, Section 4-2 also showed not all systems that are uniquely solvable satisfy this condition, proving its sufficiency.
- In case an implicit MMPS system is uniquely solvable, all modes have a unique solution, the system satisfies the solvability condition 4-0.1.
- In case at least one mode of the circuit subsystem of the implicit MMPS system has a infinitely many solutions, i.e., has a parametric solution, and none of the modes are (parametrically) unsolvable, the system can be classified as parametrically solvable. Parametrically solvable implicit MMPS systems satisfy the solvability condition 4-0.1.
- If at least one mode of the circuit subsystem of the implicit MMPS system is unsolvable, the entire implicit MMPS system is unsolvable, and the system does not satisfy the solvability condition 4-0.1.
- An unsolvable system is parametrically unsolvable if the unsolvable mode is still parametrically solvable under specific values of state $x(k-1)$. Still, a parametrically unsolvable system does not satisfy the solvability condition 4-0.1
- For implicit MMPS systems that are uniquely solvable, the existence of the inverse of $(I - M_1)$ can be guaranteed, which is necessary for the linear mapping in conventional algebra to exist. This property is proven in section 4-4.

(c) Is it possible to identify a method to classify all implicit MMPS systems with regards to their degree of solvability?

- Yes, Section 4-2 provides a method to determine which degree of solvability an implicit MMPS system has.
- The following steps need to be followed in order to determine the degree of solvability;
  - Compute matrix $S_{\otimes}^{+}$, and determine whether the system contains a circuit
  - If all diagonal entries of $S_{\otimes}^{+}$ are ε, the system is uniquely solvable. This is equivalent to the initial solvability condition
  - Identify the circuit subsystem
  - Identify all modes of the circuit subsystem that contain a circuit
  - Rewrite each circuit mode of the circuit subsystem in conventional algebra in the following form; $x_c(k) = Q \cdot x_c(k) + b_f$
  - If $\text{rank}(I-Q) = \text{rank}(I-Q|b_f) = n_c$, this circuit mode of the circuit subsystem is uniquely solvable
  - If $\text{rank}(I-Q) = \text{rank}(I-Q|b_f) < n_c$, this circuit mode of the circuit subsystem is parametrically solvable
  - If $\text{rank}(I-Q) \neq \text{rank}(I-Q|b_f)$, this circuit mode of the circuit subsystem is unsolvable
  - If there exists a state $x(k)$ for which an unsolvable system, i.e. $\text{rank}(I-Q) \neq \text{rank}(I-Q|b_f)$, becomes parametrically solvable, i.e. $\text{rank}(I-Q) = \text{rank}(I-Q|b_f) < n_c$, this circuit mode of the circuit subsystem is parametrically unsolvable.

Lastly, the first main research question can be answered.

1. Is it possible to find a necessary solvability condition for implicit MMPS systems?
   - Yes, a necessary condition for solvability of implicit MMPS systems exists, encompasses all aforementioned solvability degrees, and is proven in Section 4-3. This condition is only necessary under the assumption the system is a minimal realization.

8-2 On Control of Implicit MMPS systems

In this section, the second research question and its subquestions are answered by reflecting on the results from the research carried out in Chapter 5. Firstly, let us recall the second research question posed in 1-2. Thereafter, each subquestion, and ultimately, the second main research question is answered.

2. How can the existing control strategies for explicit MMPS systems be extended to control strategies for implicit MMPS systems?
   (a) Is it possible to find an open-loop control strategy for implicit MMPS systems?
   (b) Is it possible to find a closed-loop control strategy for implicit MMPS systems?
   (c) What are the conditions for system properties such as time-invariance and solvability for controlled implicit MMPS systems?

Having recalled the research questions, the subquestions can subsequently be answered;

(a) Is it possible to find an open-loop control strategy for implicit MMPS systems?

- Yes, an open-loop control strategy for implicit MMPS systems can easily be derived by extending the existing explicit open-loop control strategies as proposed in [4]. By adding implicit dynamic matrix D appropriately, the open-loop control strategy is applicable to implicit systems as well. This control strategy is elaborated on in Section 5-2.

(b) Is it possible to find a closed-loop control strategy for implicit MMPS systems?

- Yes, similarly to how the open-loop control strategies proposed in [4] were extended to accommodate for implicit dynamics, the closed-loop control strategies could be extended as well. An extended state matrix equation was proposed, making the closed-loop controlled system with state-feedback closely resemble the ABCDE form as in 5-1.
- Furthermore, a more general implicit input signal expression was proposed, as to allow the input signals to be implicit as well.

(c) What are conditions for system properties such as time-invariance and solvability for controlled implicit MMPS systems?

- The conditions for time-invariance of both the open-loop controlled, and the closed-loop controlled implicit MMPS system were derived. These derived conditions included the input system matrices, as their addition ought to not violate time-invariance. These conditions were derived in 5-2-2 and 5-3-2.
- In the case of open-loop control of implicit MMPS systems, there exists no implicit mapping from the input signals onto themselves, and the input signals are also not part of the extended state. Therefore, solvability cannot be violated by open-loop controlling a solvable implicit MMPS system.
- In the case of closed-loop control of implicit MMPS systems however, solvability may be violated if closed-loop control is applied to a solvable implicit MMPS system. In Section 5-3, the closed-loop state space equation shows clearly that

the state is extended by the input signal $u(k)$, which is dependent on the state $x(k)$. Therefore, an implicit mapping from $u(k)$ to $u(k)$ may occur, which could violate the solvablility of the closed-loop controlled system. By applying the theory proposed in Chapter 4, the degree of solvability can be determined.

Lastly, the second main research question can be answered;

2. How can the existing control strategies for explicit MMPS systems be extended to control strategies for implicit MMPS systems?

    - As the answers of the subquestions to this main research question show, this can be done by developing a framework that does allow implicit dynamics into the system, instead of merely allowing explicit dynamics.
    - When open-loop-, or closed-loop controlling an implicit MMPS system, as opposed to an explicit MMPS system, solvability is a factor to take into consideration. The results regarding solvability of implicit MMPS systems as presented in Chapter 4 should be applied to the controlled implicit systems to account for this.

8-3 On Augmenting, Analyzing and Controlling the Urban Railway System

In this Section, the third research question and its subquestions are answered by reflecting on the results from the research carried out in Chapter 6 and Chapter 7. Firstly, let us recall the third research question posed in 1-2. Thereafter, each subquestion, and ultimately, the third, and last main research question is answered.

3. Can the theoretical results regarding solvability and control of implicit MMPS systems be validated and/or tested by applying them to a complex real-world system such as an Urban Railway System?

   (a) Is it possible to augment the Urban Railway System proposed in [18] such that it accommodates complex passenger flows?

   (b) What insights can be gained from analyzing the dynamic behaviour of this Augmented Urban Railway System?

   (c) Can this Augmented Urban Railway System subsequently be simulated according to a uniform timetable, disturbed, and controlled using the proposed control strategies for implicit MMPS systems?

(a) Is it possible to augment the Urban Railway System proposed in [18] such that it accommodates complex passenger flows?

- Yes, by changing the dynamics of the system, complex passenger flows are accommodated for in the system equations,
- Instead of having a fixed fraction of passengers disembark train k at station j, matrix ζ is introduced, where entry $[\zeta]_{ij}$ represents the fraction of passengers who embarked at station i, who disembark at station j. Hereby, it is possible to assign passengers an origin, and a destination, rather than not considering the origin of the passengers in the train, and having a fixed fraction of them disembark.
- Furthermore, the state-space equations must change to accommodate for this flow matrix. This is done by rewriting the existing states, and adding two quantity states, one for the number of passengers embarking train k at station j, and one for the number of passengers disembarking train k at station j.
- The resulting system has 6 states, 2 of which are temporal states, and 4 of which are quantity states.

(b) What insights can be gained from analyzing the dynamic behaviour of this Augmented Urban Railway System?

- By attempting to simulate the system according to a uniform timetable, it became apparent that initialization of such a system is incredibly important. Besides having to initialize the first train, with the chosen matrix ζ, the first two stations have to be initialized as well.
- Furthermore, some of the initialization dynamics, regarding the time interval of trains departing the first, and second station, appear in the state-space matrices,

therefore already constraining the system to have one predetermined growth rate.

- Multiple (stable) fixed-points corresponding to this growth rate could be determined, by solving the LPP given in 3-33 for all possible footprint matrices. An important conclusion regarding these fixed-points is that, even though these fixed-points are fixed-points in a mathematical sense, they do not always correspond to desired behaviour of the system. The fixed-points obtained by solving the LPPs did not correspond to system behaviour corresponding to a uniform timetable.

(c) Can this Augmented Urban Railway System subsequently be simulated according to a uniform timetable, disturbed, and controlled using the proposed control strategies for implicit MMPS systems?

- By manually deriving an initial condition, and verifying that this initial condition is a fixed-point, it turned out to be possible to simulate the system according to a uniform timetable, as was done in Section 6-7.
- Four different types of small, momentary disturbances were applied to the system, all causing the uniformity of the system behaviour to be compromised. The number of passengers left behind on the platforms after the trains left the stations started to grow linearly, indicating an unstable system.
- By applying open-loop control to the disturbed systems, these disturbances could all successfully be attenuated. The control strategy could be described using the open-loop control framework proposed in Chapter 5. This MILP problem could be solved using a Gurobi solver in a matter of seconds.

Lastly, the third main research question could be answered;

3. Can the theoretical results regarding solvability and control of implicit MMPS systems be validated and/or tested by applying them to a complex real-world system such as an Urban Railway System?

   - In short, yes, this is possible. An important remark is that obviously only an open-loop control strategy was applied, so the closed-loop control strategy could not be validated. The solvability could however be determined using the proposed methods.

8-4 Contributions

This thesis contributes to research in the fields of Systems and Control and Discrete Event Systems, specifically, to the field of Max-Min-Plus-Scaling systems. The academic contributions made by this thesis are quantified by the following results;

- Developed a graph-theoretic interpretation of the existing solvability condition,
- Identified four different degrees of solvability within implicit Max-Min-Plus Scaling systems
- Introduced methods to classify all implicit MMPS systems according to their degree of solvability
- Derived a condition for implicit MMPS systems containing a circuit that guarantees the existence of the inverse of $(I - M_1)$, provided the condition is satisfied
- Proved a necessary solvability condition for implicit MMPS systems under the assumption of the system being a minimal realization
- Proposed open-loop control strategies for implicit MMPS systems
- Proposed closed-loop control strategies for implicit MMPS systems
- Derived conditions for solvability and time-invariance for the proposed open-loop, and closed-loop control strategies
- Augmented the Urban Railway System described in [18] to allow for complex passenger flow through the system, more closely modeling reality
- Analyzed, disturbed, and controlled the Augmented Urban Railway System using the theoretical results regarding solvability and control of implicit MMPS systems

Chapter 9

Recommendations for Further Research

As with almost all research, this work raises new questions alongside the answers it provides. This section reflects on the main findings and outlines several directions for future research. All proposed future research opportunities, and described uncovered knowledge gaps aim to deepen the understanding of Max-Min-Plus-Scaling (MMPS) systems, Discrete Event Systems, or control of Implicit (MMPS) systems;

- **Derive conditions for monotonicity and non-expansiveness for implicit MMPS systems**

Having thoroughly researched the property of solvability for implicit MMPS systems, it may be an interesting research direction to try to develop conditions for the properties of monotonicity and non-expansiveness. In case such conditions were to be established, the concept of topical implicit MMPS systems may emerge, allowing properties of topical systems to apply to topical implicit MMPS systems as well. For example, there always being exactly one growth rate.

- **Establish requirements for when a realization of an (implicit) MMPS system is minimal**

The necessary solvability condition only holds under the assumption that the system is a minimal realization, so no redundant mode, or unused term in $z(k)$ exists. However, the concept of minimal realizations in this context has not been properly defined or researched in known literature. Accurately determining conditions that a system must satisfy in order to be a minimal realization is very useful, as it reflects back on whether the necessary solvability condition is applicable. Furthermore, it may aid in reducing the size of the MMPS system, as identifying redundant nodes means they can be removed from the system equations.

- **Accommodate for the dynamics of the first, and last station of the state-space description of the Augmented Urban Railway System (AURS)**

Section 6-7 thoroughly described the simulation of the AURS according to a uniform timetable. Assumptions regarding the first and last station had to be done to preserve this uniformity. Therefore, a logical physical interpretation of the AURS simulated according to the uniform timetable is that the observed stations are part of a larger network, i.e. the physical limitations regarding first and last station can be disregarded. However, when aiming to simulate the system resembling a real-life scenario where the first station is actually the start of an urban railway line, these physical limitations, such as, no passengers disembark at station 1, will have to be accounted for. Further research could investigate how to properly integrate these dynamics into the system equations.

- **Apply closed-loop control strategies, or online control strategies to the AURS**

The control strategy applied to the AURS is an offline, open-loop control strategy. It would be interesting to uncover how other types of control, such as a receding horizon MPC, which is an online control strategy, would attenuate different types of disturbances. The limitation of applying offline control, is that it is not able to deal with real-time disturbances, which a control strategy like a receding horizon MPC could handle better. Furthermore, state-feedback controllers, i.e. closed-loop controllers have not been applied to implicit MMPS system in any known literature, constituting an interesting research opportunity.

- **Improve the AURS by modeling an intersection of two railway lines**

The dynamics describing the AURS are still limited to describing a single urban railway line. An interesting research direction would be to investigate how an intersection of two railway lines, i.e. a URS containing stations where passengers can transfer can be modeled. Adding such complexities in the system require advanced mathematical modeling, and will surely enlarge the size of the model. However, adding such dynamics will broaden the applicability

Appendix A

Disturbances applied to the AURS

A-1 Disturbance 1 - decrease in passenger arrival rate $e_{j,k}$

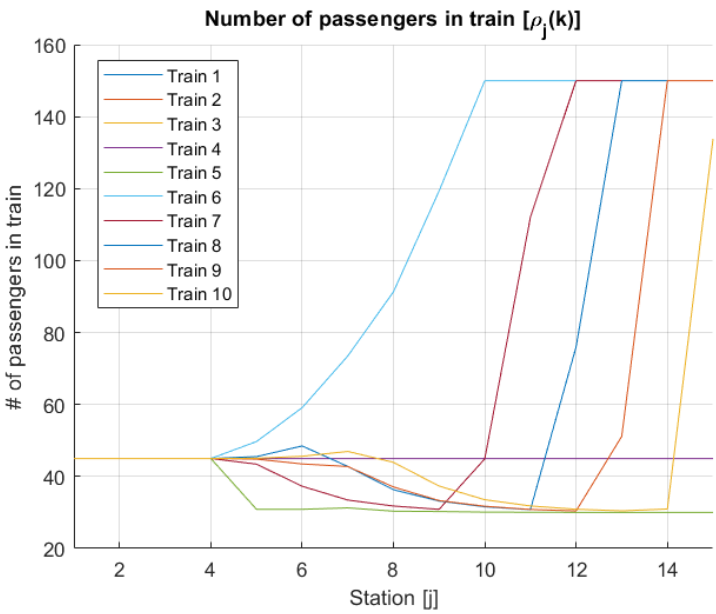


Figure A-1: Number of passengers in each train after leaving each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 1

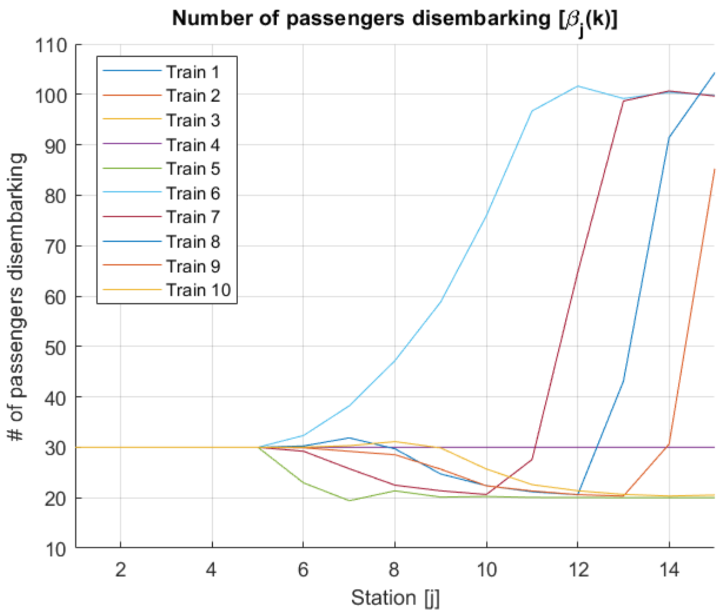


Figure A-2: Number of passengers disembarking each train at each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 1

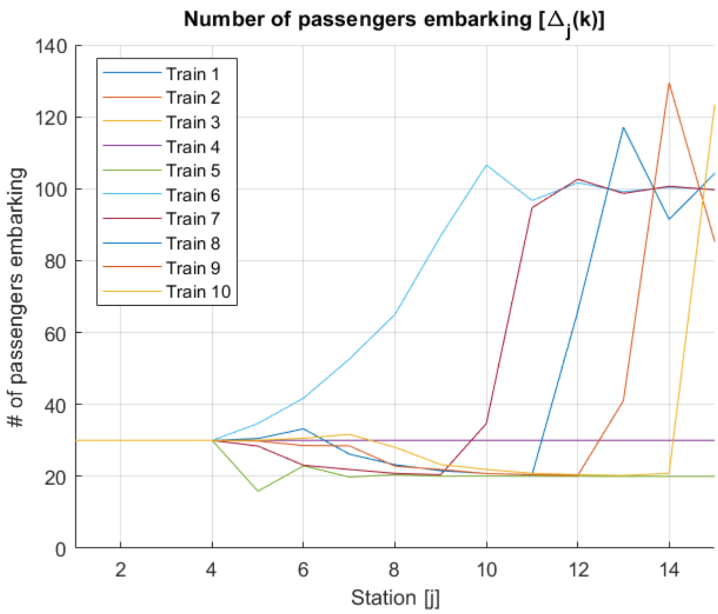


Figure A-3: Number of passengers embarking each train at each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 1

A-2 Disturbance 2 - surge in passenger arrival rate $e_{j,k}$

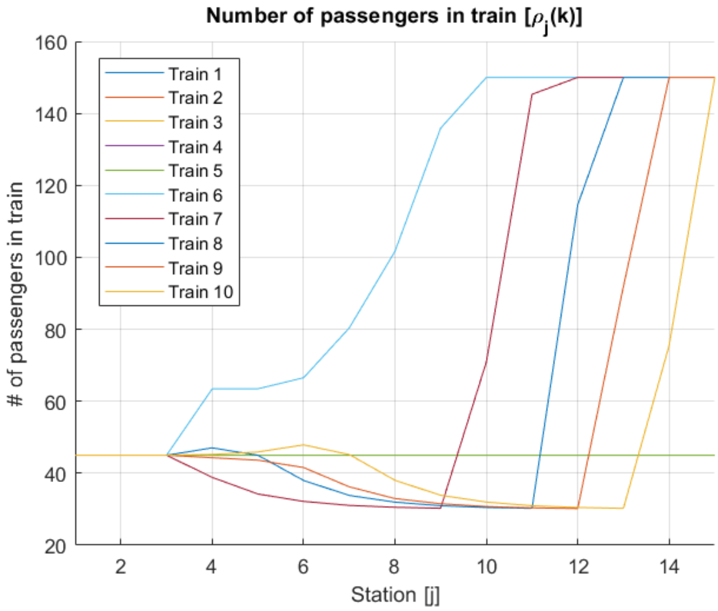


Figure A-4: Number of passengers in each train after leaving each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 2

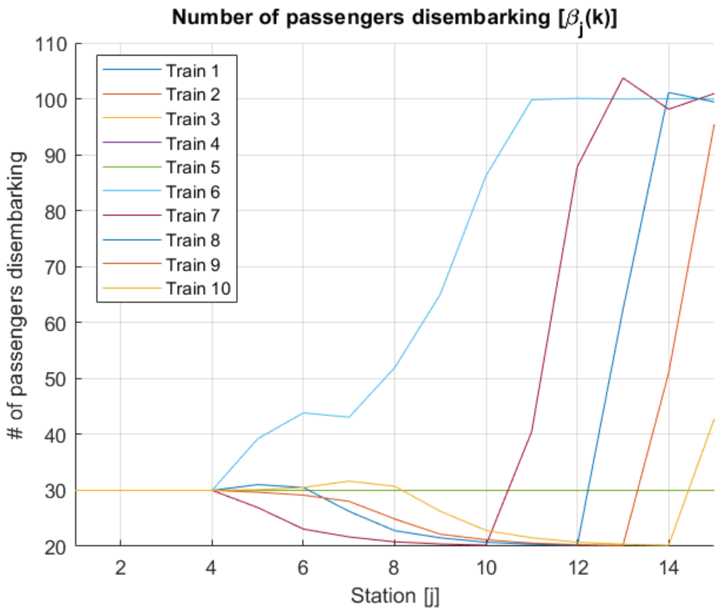


Figure A-5: Number of passengers disembarking each train at each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 2

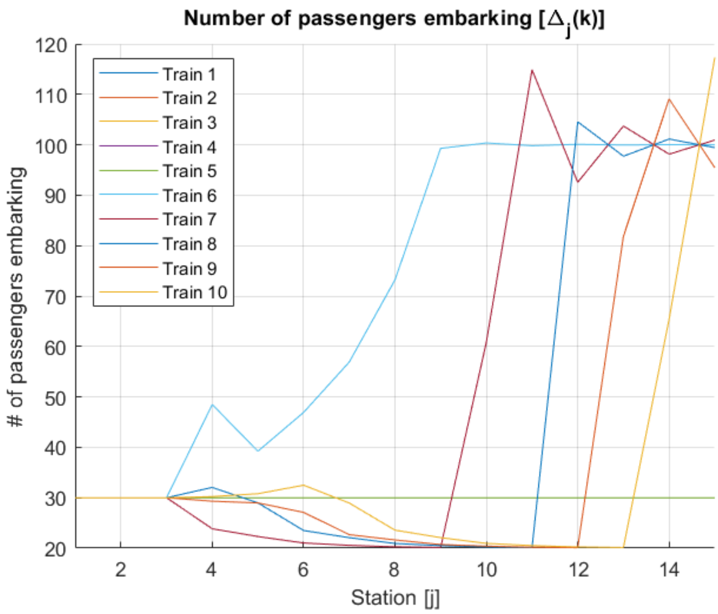


Figure A-6: Number of passengers embarking each train at each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 2

A-3 Disturbance 3 - decrease in disembark rate f

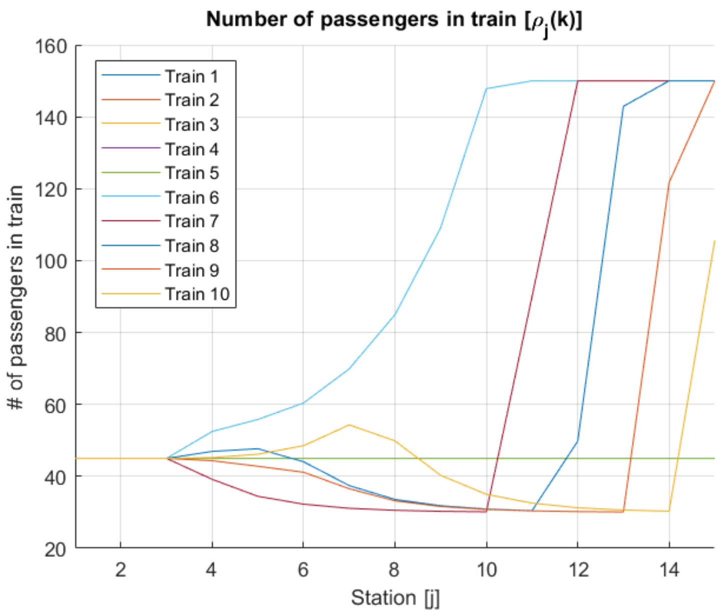


Figure A-7: Number of passengers in each train after leaving each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 3

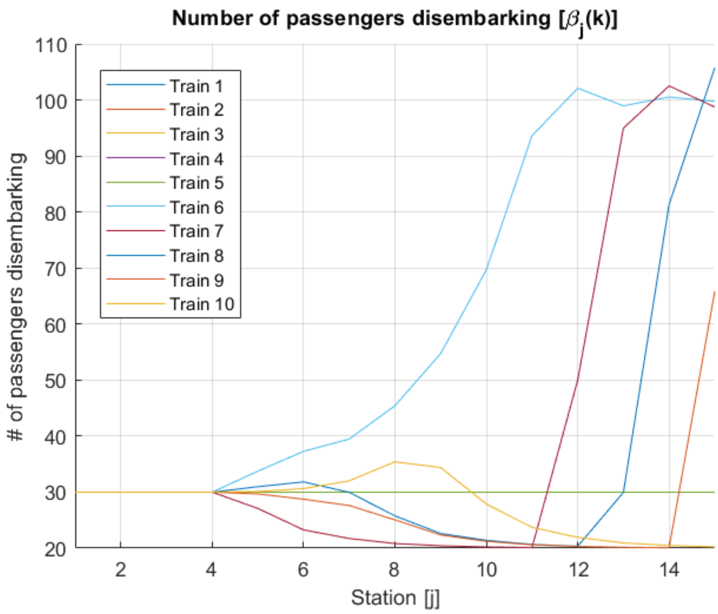


Figure A-8: Number of passengers disembarking each train at each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 3

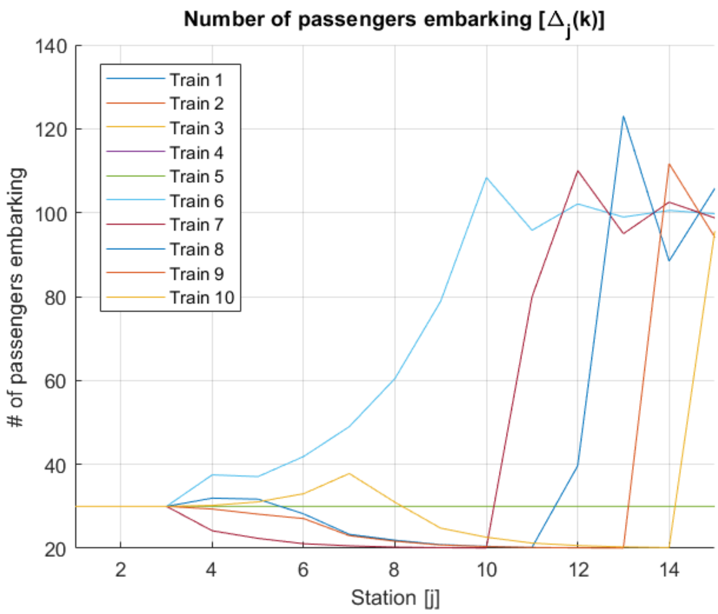


Figure A-9: Number of passengers embarking each train at each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 3

A-4 Disturbance 4 - surge in embark rate b

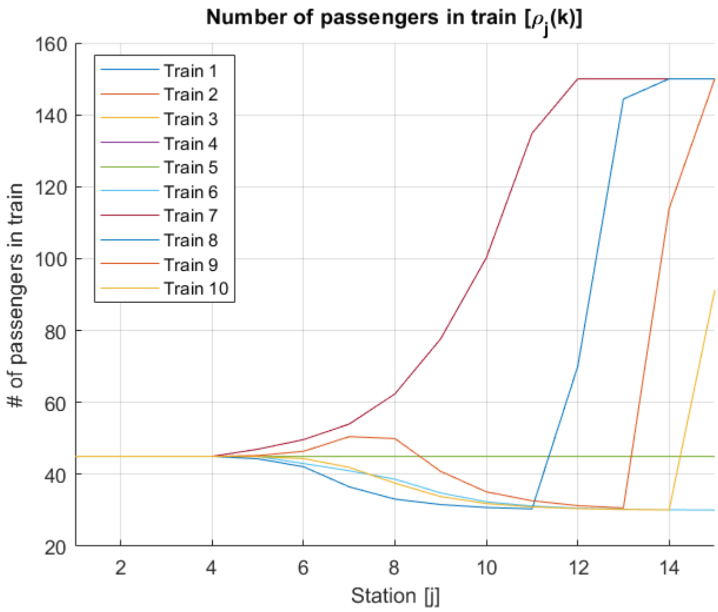


Figure A-10: Number of passengers in each train after leaving each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 4

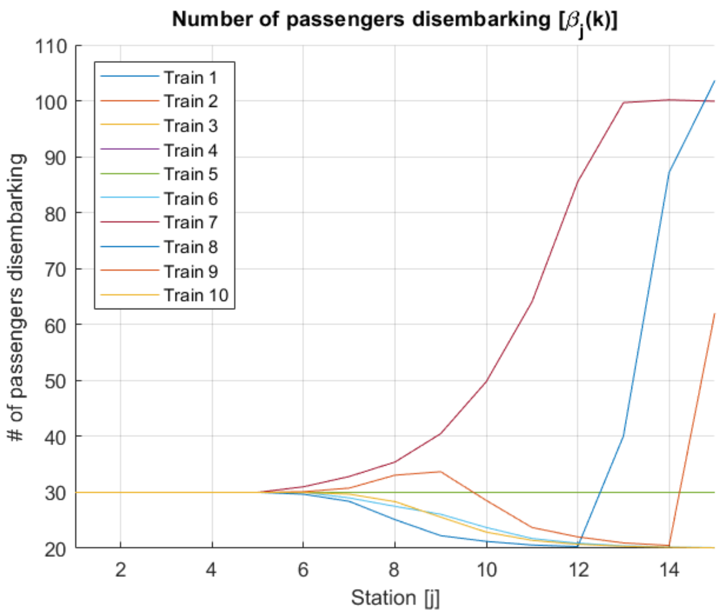


Figure A-11: Number of passengers disembarking each train at each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 4

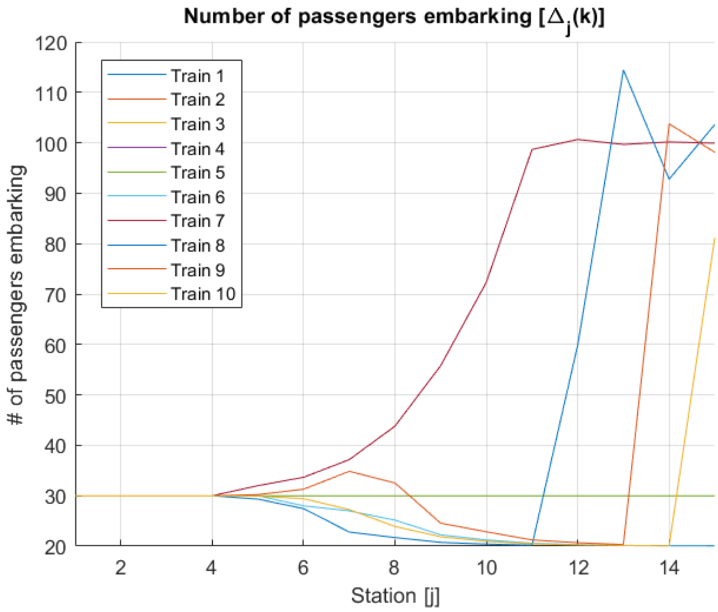


Figure A-12: Number of passengers embarking each train at each station of the simulation of the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 4

Appendix B

Controlled AURS

B-1 Controlled system 1 - decrease in passenger arrival rate $e_{j,k}$

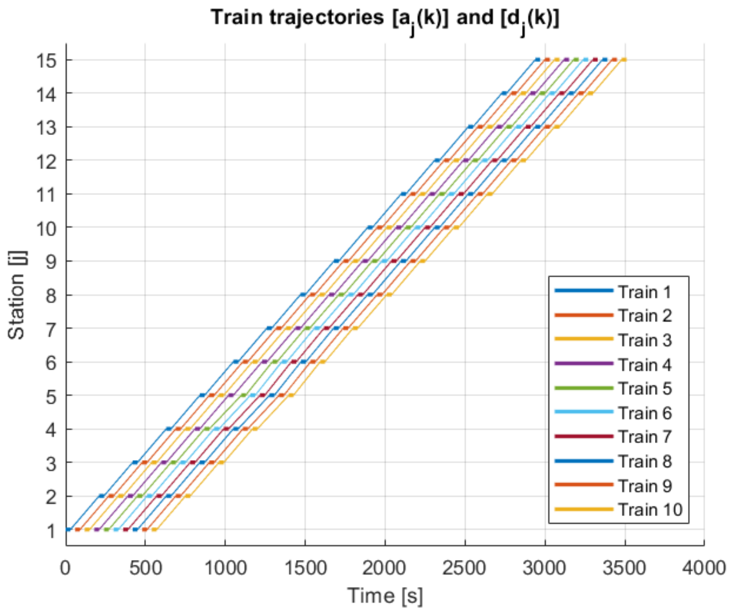


Figure B-1: Train trajectories of the controlled simulation of the Augmented URS with 10 trains and 15 stations, disturbed with disturbance 1

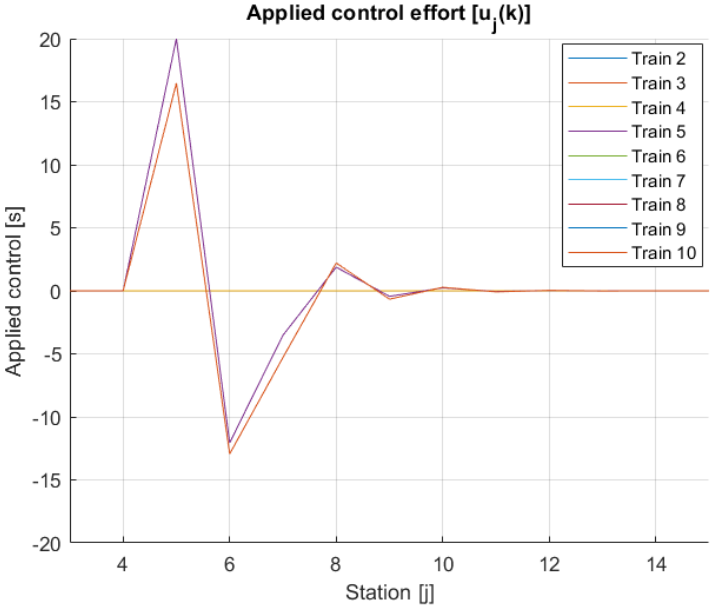


Figure B-2: Input signal used for attenuating disturbance 1 applied to the AURS with 15 stations and 10 trains

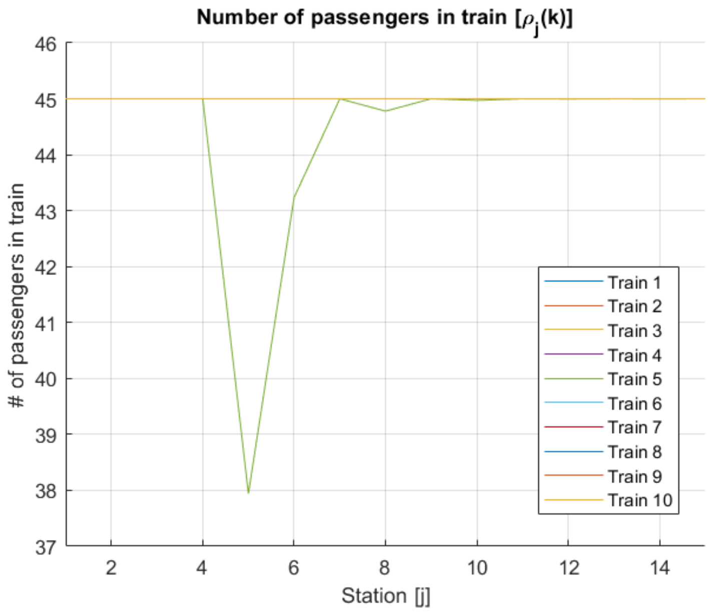


Figure B-3: Number of passengers in each train after leaving each station of the simulation of the controlled Augmented URS with 15 stations and 10 trains, disturbed with disturbance 1

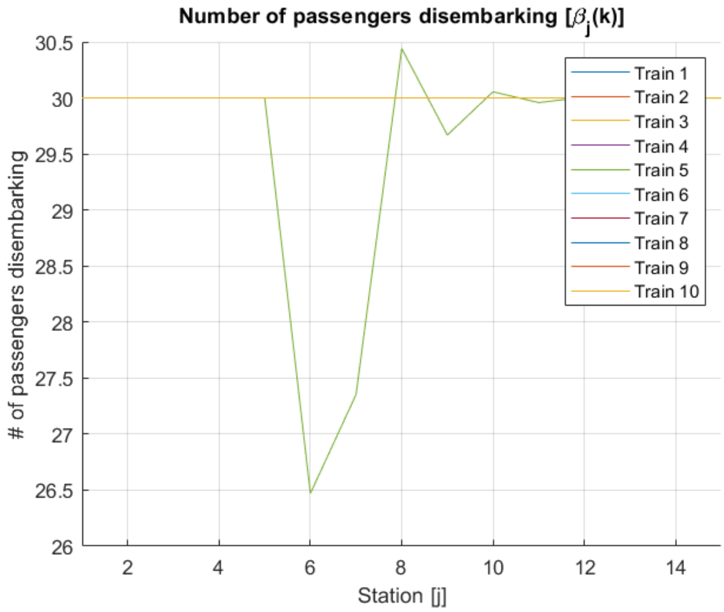


Figure B-4: Number of passengers disembarking each train at each station of the simulation of controlled the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 1

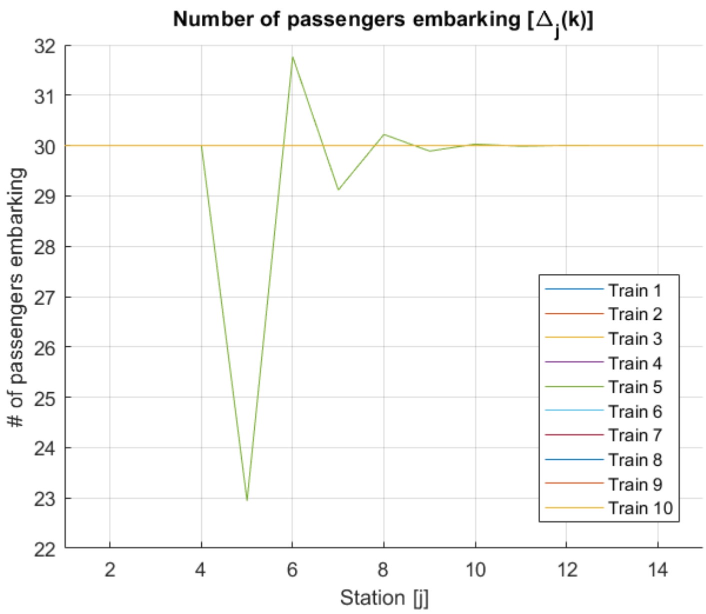


Figure B-5: Number of passengers embarking each train at each station of the simulation of the controlled Augmented URS with 15 stations and 10 trains, disturbed with disturbance 1

B-2 Controlled system 2 - surge in passenger arrival rate $e_{j,k}$

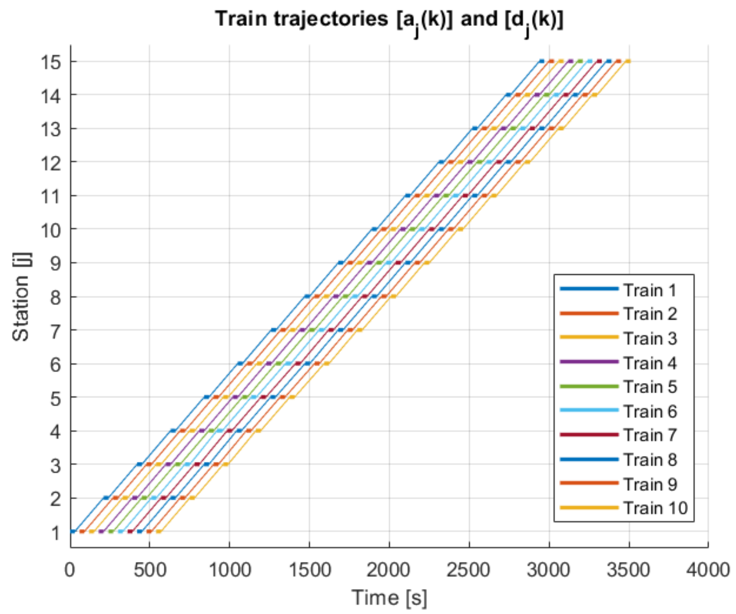


Figure B-6: Train trajectories of the controlled simulation of the Augmented URS with 10 trains and 15 stations, disturbed with disturbance 2

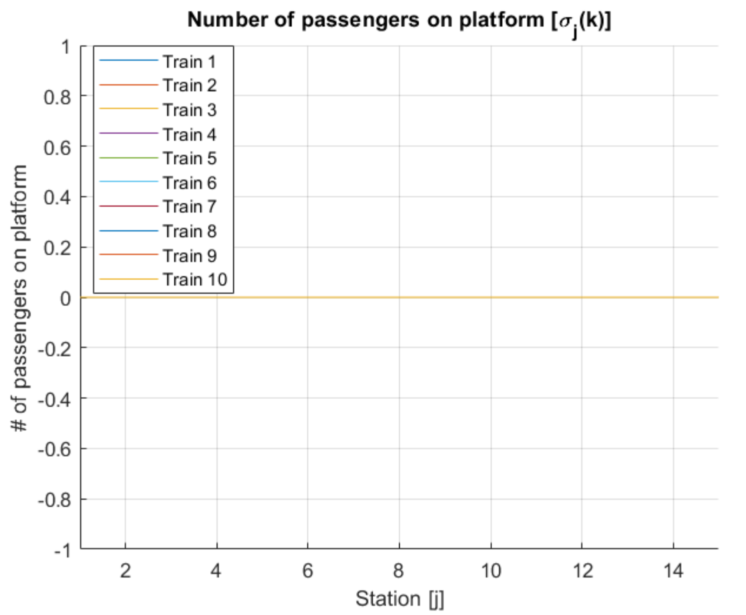


Figure B-7: Number of passengers left on the platform of each station after each train has departed of the simulation of the controlled AURS with 15 stations and 10 trains, disturbed with disturbance 2

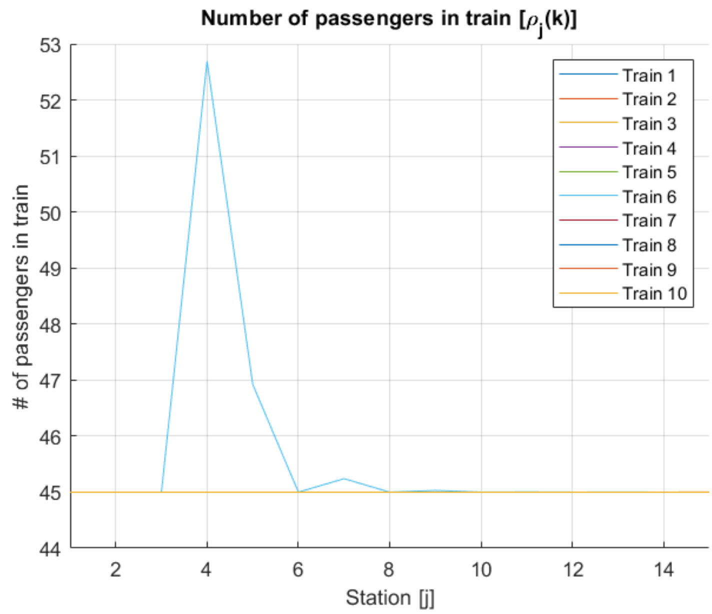


Figure B-8: Number of passengers in each train after leaving each station of the simulation of the controlled Augmented URS with 15 stations and 10 trains, disturbed with disturbance 2

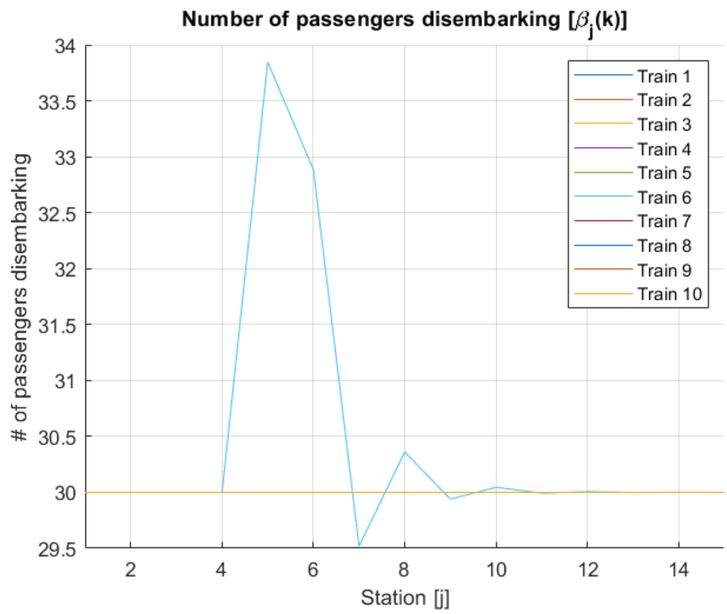


Figure B-9: Number of passengers disembarking each train at each station of the simulation of controlled the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 2

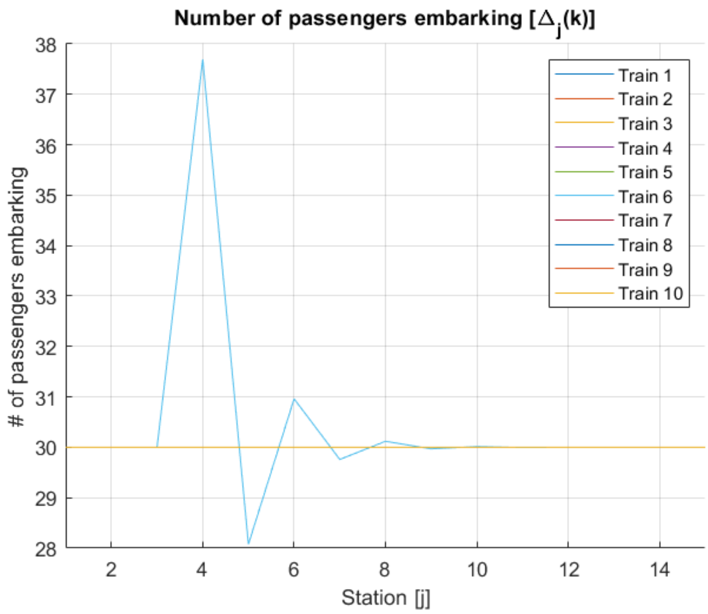


Figure B-10: Number of passengers embarking each train at each station of the simulation of the controlled Augmented URS with 15 stations and 10 trains, disturbed with disturbance 2

B-3 Controlled system 3 - decrease in disembark rate f

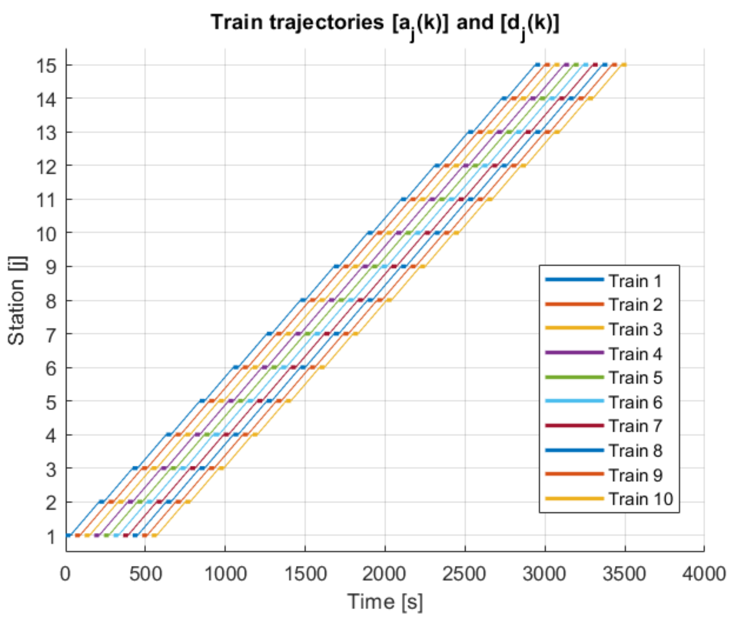


Figure B-11: Train trajectories of the controlled simulation of the Augmented URS with 10 trains and 15 stations, disturbed with disturbance 3

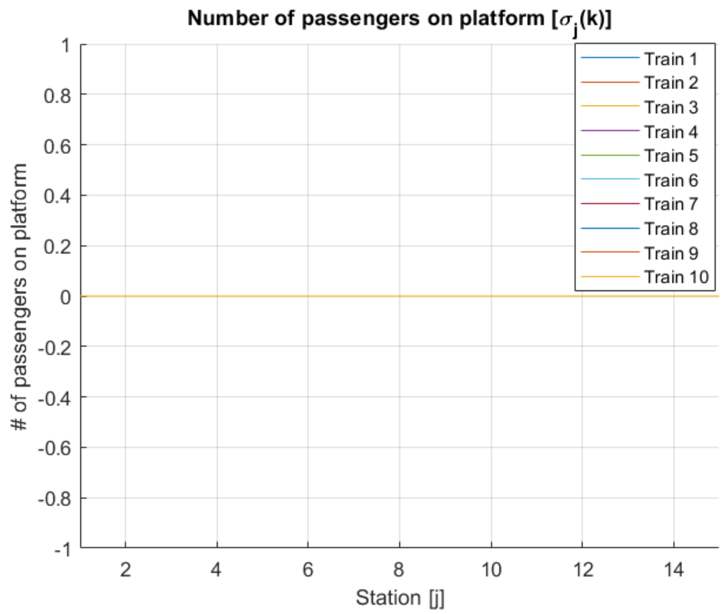


Figure B-12: Number of passengers left on the platform of each station after each train has departed of the simulation of the controlled AURS with 15 stations and 10 trains, disturbed with disturbance 3

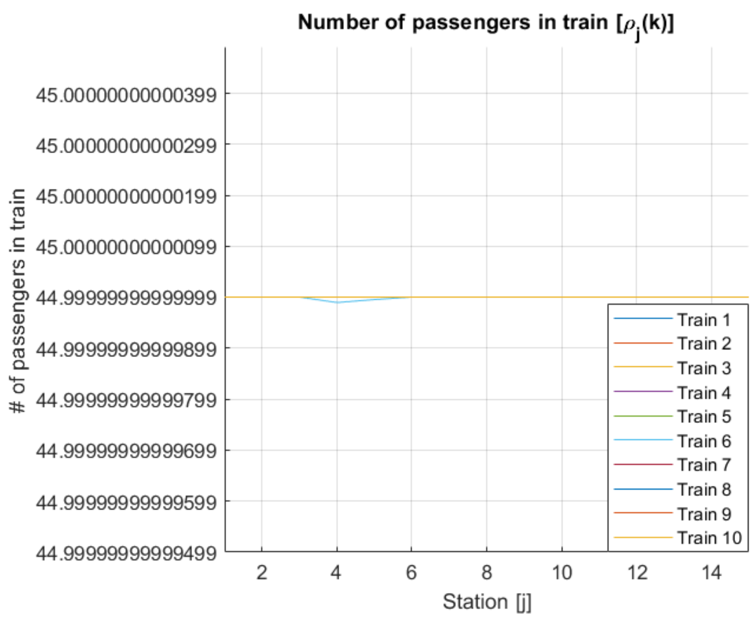


Figure B-13: Number of passengers in each train after leaving each station of the simulation of the controlled Augmented URS with 15 stations and 10 trains, disturbed with disturbance 3

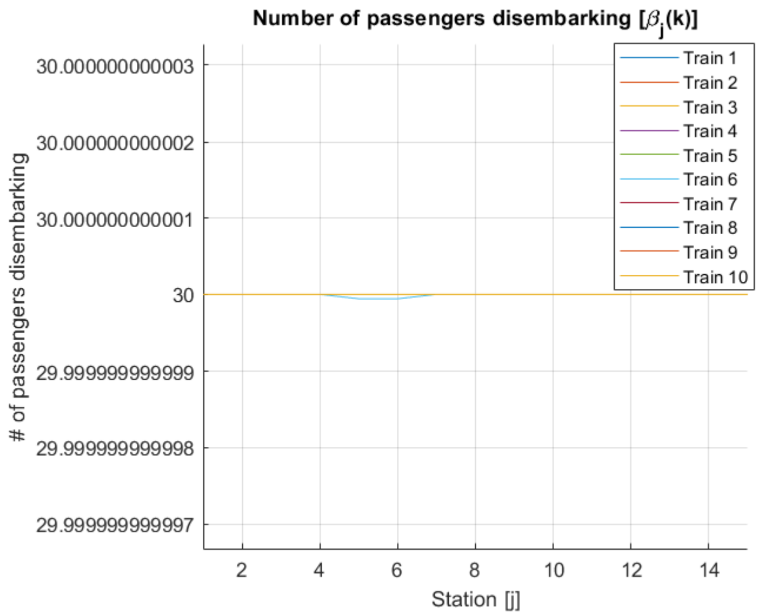


Figure B-14: Number of passengers disembarking each train at each station of the simulation of controlled the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 3

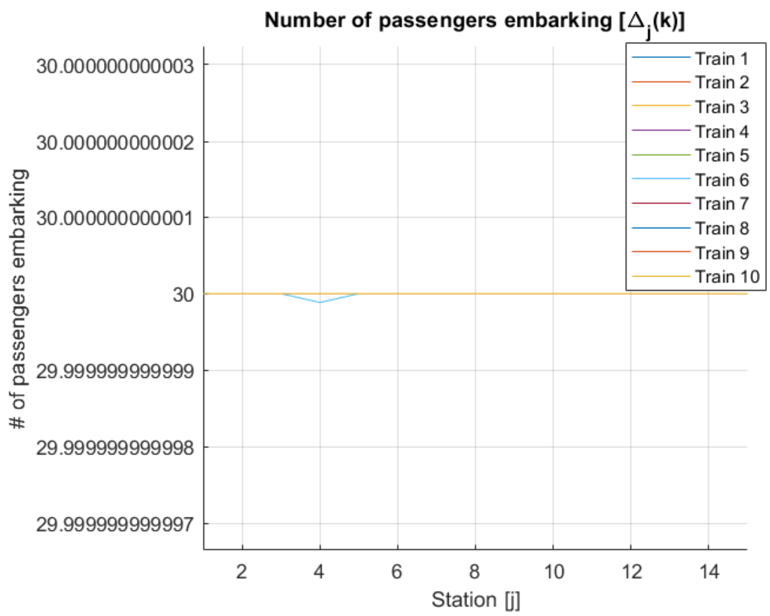


Figure B-15: Number of passengers embarking each train at each station of the simulation of the controlled Augmented URS with 15 stations and 10 trains, disturbed with disturbance 3

B-4 Controlled system 4 - surge in embark rate b

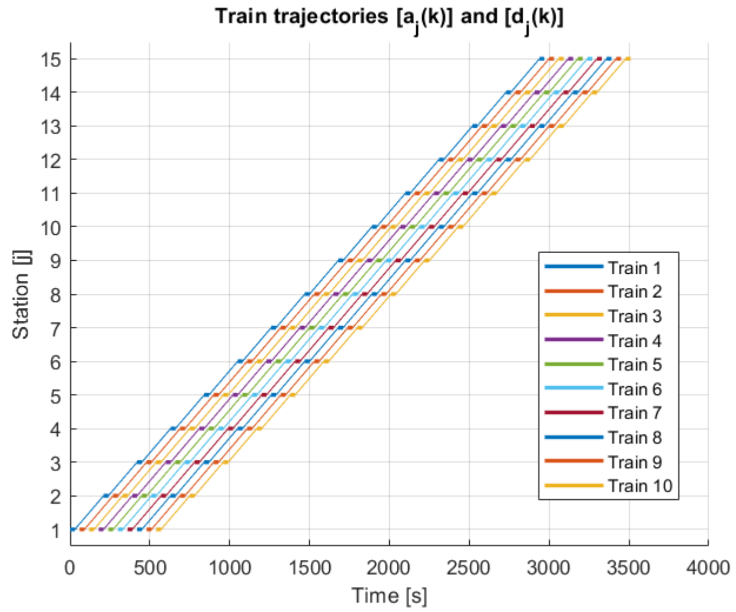


Figure B-16: Train trajectories of the controlled simulation of the Augmented URS with 10 trains and 15 stations, disturbed with disturbance 4

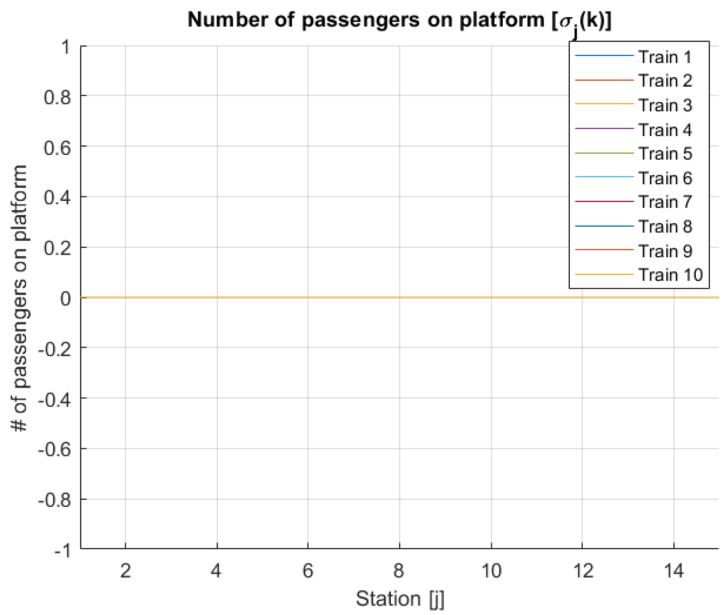


Figure B-17: Number of passengers left on the platform of each station after each train has departed of the simulation of the controlled AURS with 15 stations and 10 trains, disturbed with disturbance 4

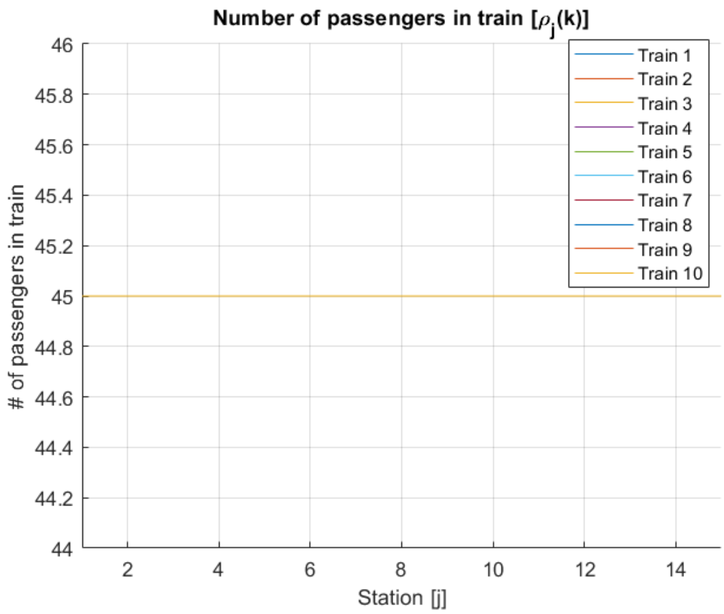


Figure B-18: Number of passengers in each train after leaving each station of the simulation of the controlled Augmented URS with 15 stations and 10 trains, disturbed with disturbance 4

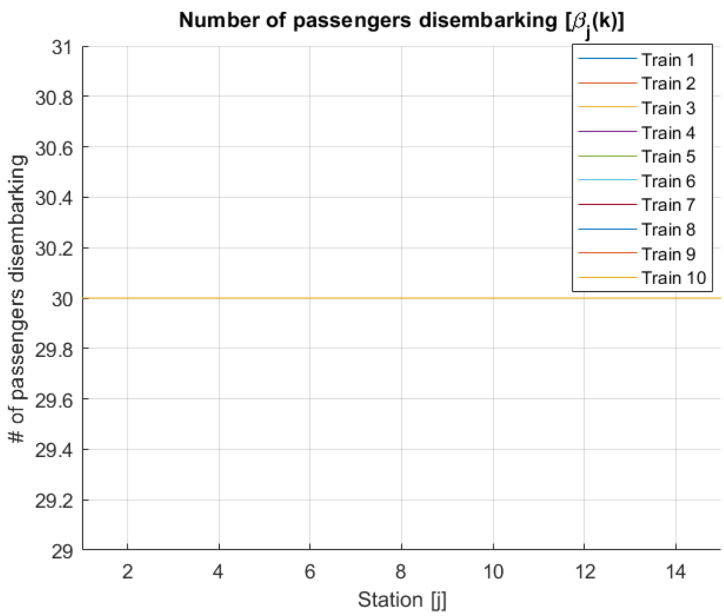


Figure B-19: Number of passengers disembarking each train at each station of the simulation of controlled the Augmented URS with 15 stations and 10 trains, disturbed with disturbance 4

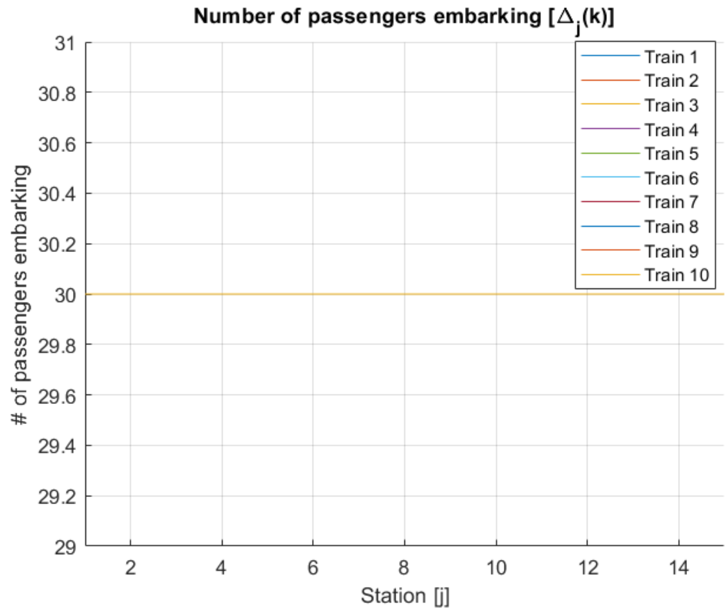


Figure B-20: Number of passengers embarking each train at each station of the simulation of the controlled Augmented URS with 15 stations and 10 trains, disturbed with disturbance 4

Appendix C

Fixed point analysis of the AURS

Table C-1: Analysis of 41 fixed points of the AURS with $J = 4$ stations

#	Multiplicity e.v. 1 M_θ	BB stable?	H_{eq} rank def.	$\mathcal{O}_\infty$ found	Iter
1	11	Yes	11	Yes	2
2	11	Yes	11	Yes	3
3	11	Yes	11	Yes	3
4	11	Yes	11	Yes	4
5	12	No	11	No	–
6	11	Yes	11	Yes	2
7	12	No	11	No	–
8	11	Yes	11	Yes	3
9	12	No	11	No	–
10	12	No	11	No	–
11	12	No	11	No	–
12	12	No	11	No	–
13	11	Yes	11	Yes	2
14	11	Yes	11	Yes	3
15	13	No	11	Yes	45
16	12	No	11	Yes	34
17	12	No	11	Yes	17
18	11	Yes	11	Yes	2
19	13	No	11	No	–
20	12	No	11	Yes	27
21	12	No	11	No	–

#	Multiplicity e.v. 1 M_θ	BB stable?	H_{eq} rank def.	$\mathcal{O}_\infty$ found	Iter
22	12	No	11	No	–
23	12	No	11	No	–
24	12	No	11	No	–
25	13	No	11	No	–
26	12	No	11	Yes	28
27	13	No	11	Yes	236
28	12	No	11	Yes	38
29	13	No	11	No	–
30	13	No	11	No	–
31	13	No	11	No	–
32	13	No	11	No	–
33	13	No	11	Yes	61
34	13	No	11	Yes	115
35	14	No	11	No	–
36	13	No	11	Yes	26
37	14	No	11	Yes	117
38	13	No	11	Yes	27
39	14	No	11	Yes	24
40	14	No	11	Yes	141
41	11	Yes	11	Yes	4

Bibliography

[1] S. Axler. *Linear Algebra Done Right*. Undergraduate Texts in Mathematics. Springer International Publishing, Cham, 2024.

[2] F. Borrelli, A. Bemporad, and M. Morari. *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.

[3] G. Cohen, D. Dubois, J. Quadrat, and M. Viot. A linear-system-theoretic view of discrete-event processes and its use for performance evaluation in manufacturing. *IEEE Transactions on Automatic Control*, 30(3):210–220, March 1985. Conference Name: IEEE Transactions on Automatic Control.

[4] S R Daams. Control Strategies for Max-Min-Plus-Scaling Systems. *Master's thesis, Delft University of Technology*, 2025.

[5] B. De Schutter and T. Van den Boom. On model predictive control for max-min-plus-scaling discrete event systems. *Technical Report Bds 00-04: Control Systems Engineering, Faculty of Information Technology and Systems*, 2000.

[6] B. De Schutter, T. van den Boom, J. Xu, and S.S. Farahani. Analysis and control of max-plus linear discrete-event systems: An introduction. *Discrete Event Dynamic Systems: Theory and Applications*, 30(1):25–54, 2020.

[7] B. De Schutter and T.J.J. Van Den Boom. MPC for continuous piecewise-affine systems. *Systems & Control Letters*, 52(3-4):179–192, July 2004.

[8] A Gupta. Max-plus-algebraic hybrid automata. *Doctoral thesis, Delft University of Technology*, 2023.

[9] B. Heidergott, G. J. Olsder, and J. van der Woude. *Max Plus at Work: Modeling and Analysis of Synchronized Systems: A Course on Max-Plus Algebra and Its Applications*. Princeton University Press, September 2014.

[10] J. P. Hespanha. *Linear Systems Theory: Second Edition*. Princeton University Press, February 2018.

[11] R. M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Mathematics*, 23(3):309–311, January 1978.

[12] J. Komenda, S. Lahaye, J.-L. Boimond, and T. Van Den Boom. Max-plus algebra in the history of discrete event systems. *Annual Reviews in Control*, 45:240–249, 2018.

[13] C. Leake, F. Baccelli, G. Cohen, G. J. Olsder, and J. Quadrat. Synchronization and Linearity: An Algebra for Discrete Event Systems. *The Journal of the Operational Research Society*, 45(1):118, January 1994.

[14] S. Markkassery, T. van den Boom, and B. De Schutter. Stability of Time-invariant Max-Min-Plus-Scaling Discrete-Event Systems with Diverse States. *IFAC-PapersOnLine*, 58(1):60–65, January 2024.

[15] S. Markkassery and B. De Schutter. Dynamics of Implicit Time-Invariant Max-Min-Plus-Scaling Discrete-Event Systems. *Unpublished*, 2024.

[16] S. Markkassery, T. Van Den Boom, and B. De Schutter. Eigenvalues of Time-invariant Max-Min-Plus-Scaling Discrete-Event Systems. In *2024 European Control Conference (ECC)*, pages 2017–2022, Stockholm, Sweden, June 2024. IEEE.

[17] S. A. Rosyada, Siswanto, and V. Y. Kurniawan. Bases in Min-Plus Algebra. pages 313–316. Atlantis Press, November 2021.

[18] T. van den Boom, A. Gupta, B. De Schutter, and R.E.S Beek. Max-Min-Plus-Scaling Systems in a Discrete-Event Framework with an Application in Urban Railway. *IFAC-PapersOnLine*, pages 7906–7911, January 2023.

[19] V. M. van Heijningen. Max-min-plus-scaling systems, real-life applications of an innovative framework. *Literature research, Delft University of Technology*, November 2024.

[20] Q. Zhao, D. Zheng, and X. Zhu. Structure properties of min-max systems and existence of global cycle time. *IEEE Transactions on Automatic Control*, 46(1):148–151, January 2001. Conference Name: IEEE Transactions on Automatic Control.

Glossary

List of Acronyms

AURS	Augmented Urban Railway System
CL	Closed-Loop
DES	Discrete Event System
LPP	Linear Programming Problem
MILP	Mixed Integer Linear Programming
MMP	Max-Min-Plus
MMPS	Max-Min-Plus-Scaling
OL	Open-Loop
URS	Urban Railway System