Control Strategies for Max-Min-Plus-Scaling Systems

An introduction on open-loop and closed-loop control

S.R. Daams

Master of Science Thesis

Delft University of Technology

Delft Center for Systems and Control

Control Strategies for Max-Min-Plus-Scaling Systems

An introduction on open-loop and closed-loop control

Master of Science Thesis

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

S.R. Daams

December 12, 2024

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 ME for acceptance a thesis entitled

Control Strategies for
Max-Min-Plus-Scaling Systems

by

S.R. Daams

in partial fulfillment of the requirements for the degree of

Master of Science Systems and Control

Dated: December 12, 2024

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

____
ir. S. Markkassery

Reader(s): ______________________
dr.ir. M.B. Mendel

Abstract

This thesis offers a detailed exploration of the integration of input signals and control mechanisms within max-min-plus-scaling (MMPS) systems, a subclass of discrete event (DE) systems. Unlike traditional control systems, which rely on continuous evolution modeled by differential equations, DE systems progress through the occurrence of discrete events. MMPS systems enhance this adaptability by encompassing maximization, minimization, scaling, and addition, creating a framework for modeling and managing various processes, including logistics networks and urban railway systems.

The primary objective of this thesis is to introduce input signals into MMPS systems and systematically investigate control strategies. This involves establishing a structure accommodating these input signals while preserving essential properties such as time invariance. The study examines both open-loop and closed-loop control strategies, focusing on the latter to implement optimization-based control to optimize system performance through effective feedback control.

This thesis is organized, beginning with the mathematical foundation of MMPS systems and progressing to the development of control methods. The implementation of these methods is validated through practical applications such as manufacturing systems and the urban railway system, demonstrating their effectiveness.

By advancing our understanding of control in MMPS systems, this research provides a systematic methodology that integrates control and illustrates how optimization-based techniques can enhance overall performance. The insights gained from this work lay a groundwork for future research, potentially extending beyond transportation systems to other discrete event-driven industries. Engaging with this research has the potential to control numerous fields, promising innovative solutions and improved efficiencies across sectors.

Table of Contents

Preface and Acknowledgements

Dear reader,

Over the past year, I have been working on my final project at Delft University of Technology, my Master's Thesis. Initially, the thought of focusing on the same subject for an entire year seemed like an endless journey into challenging mathematical problems. I found myself explaining my work to others countless times, often leaving them confused when I mentioned using a different kind of algebra. However, dedicating myself to a single topic over such a long period has fostered a deep sense of commitment. What once felt like an endless path turned out to be an incredible learning experience.

I would like to begin by expressing my sincere gratitude to my supervisors, Ton van den Boom and Sreeshma Markkassery, who have guided me throughout this thesis process.

Sreeshma, thank you for all our meetings. In the early stages of my thesis, it was especially challenging to grasp the theory behind this subject. Your extra meetings and willingness to answer my questions were invaluable. I wish you the best of luck and look forward to following your future work in this field.

Ton, from the very beginning, you were able to inspire my interest in this subject. The enthusiasm and energy you brought to our meetings were contagious, leaving me feeling more motivated each time. Your guidance played a crucial role in helping me achieve my goals and enjoy the process. I'm excited to read the finalized book on max-min-plus-scaling systems.

I would also like to thank Olaf and my friends from "Ongecontroleerd," who all positively impacted my academic achievements. Olaf, your help during our Bachelor's assignments was greatly appreciated. To everyone in "Ongecontroleerd," thank you for helping and keeping me motivated through the challenging courses the Master Systems and Control has to offer. I wish you all the best with your own Master's Theses.

Lastly, I want to express my heartfelt gratitude to my parents and sister. Although you may not fully understand the topics I study, your unwavering support has been invaluable, not only throughout my academic journey but throughout my entire life.

Thank you all.

Stef

“You see, Jude, in life, sometimes nice things happen to good people.”
— *Hanya Yanagihara, A Little Life*

Chapter 1

Introduction

This chapter introduces the research topics addressed in this thesis. Section 1-1 presents relevant background information. Section 1-2 discusses the motivation for this research, based on the literature review [1] conducted prior to this thesis, from which the research topics emerged. Additionally, this section outlines the approach used to investigate the research questions. Finally, Section 1-3 provides an overview of the structure of this thesis.

1-1 Background

In today's world, systems and control theory plays an increasingly vital role, driven by the rising demand for automation across various industries. The importance of this engineering discipline extends far beyond what many initially recognize. Control systems are integral to a wide range of applications, including energy distribution, transportation, manufacturing, medical devices, and even in managing economic processes such as unemployment and inflation. A critical aspect of effectively implementing control is the accurate modeling of these systems. Models capture the dynamics of control systems, allowing for simulation and analysis. Typically, these models are described by differential equations within conventional algebra (also known as plus-times algebra), where system evolution is influenced by internal and external factors over time.

However, there exists a subclass of systems that operate under a different mathematical framework: dioid algebra. These systems, known as discrete event (DE) systems, evolve based on the occurrence of discrete events rather than continuous time. In this context, events refer to sudden changes in a process [2]. Examples of DE systems include logistics networks, manufacturing systems, and urban railway systems [3].

When DE systems are modeled using conventional algebra, they often exhibit non-linear behavior, making the implementation of control more challenging and less flexible than with linear systems. Linear systems are foundational in classical control theory due to their simplicity and the wide applicability of linear control methods [4]. This challenge has led to the development of a subclass of DE systems that can be described as linear under max-plus algebra [3]. Max-plus algebra differs from conventional algebra by replacing addition and multiplication with maximization and addition, respectively.

Further research into this area has led to the extension of max-plus linear (MPL) systems to max-min-plus-scaling (MMPS) systems, which also incorporate minimization and scaling operations. Early studies on MMPS systems were conducted by [5, 6]. More recently, [7] introduced a mathematical framework for modeling MMPS systems by combining max-plus and min-plus algebra with scaling factors. Additional work by [8] explored key characteristics of MMPS systems, particularly the importance of time invariance, while [9] examined the stability of MMPS systems. However, much of the existing research focuses on autonomous systems, which are unaffected by external control. This presents a promising opportunity to investigate how control can be introduced to MMPS systems.

Incorporating inputs into MMPS systems introduces complexities that require careful consideration. Properties such as time invariance, as discussed in [8], must be maintained. Moreover, there are multiple ways to introduce input channels into DE systems, each requiring a tailored approach when applied to MMPS models. After structuring the input signal integration, this thesis will explore both open-loop and closed-loop control strategies for MMPS systems. The goal of this research is to provide a comprehensive methodology for adding input signals to MMPS systems and to apply various types of control to these systems.

1-2 Problem description

After defining the background of this research and identifying the research opportunity, we can describe the research problems for this thesis. The research questions are stated to guide the investigation conducted in this master's thesis. The research gap around the introduction of input signals and the application of control in MMPS systems gives rise to the following research questions.

1-2-1 Research questions

1. How can input signals be systematically integrated into MMPS systems, considering both temporal and quantity-based signals?
   - (a) How can the existing ABCD canonical form be extended to accommodate input signals?
   - (b) What constraints must be applied to the input structure to preserve key properties, such as time-invariance?
2. What is a systematic approach to implementing control strategies in MMPS systems?
   - (a) What are the effects and limitations of using open-loop control in MMPS systems?
   - (b) What are the effects and limitations of using closed-loop control in MMPS systems?
3. How can closed-loop control strategies be employed to regulate the dynamic behavior of MMPS systems?

(a) What specific objectives are pursued by implementing closed-loop control in MMPS systems?

(b) What methods and criteria are most effective for designing and optimizing closed-loop controllers for MMPS systems?

(c) How can closed-loop control mechanisms be tailored and applied to improve performance in urban railway systems?

1-2-2 Approach

In the first subsection of this chapter, we discussed the motivation and background for the research questions outlined above. Before addressing these questions, however, it is essential to thoroughly investigate the topic. To this end, we will first provide a comprehensive overview of the mathematical foundations of dioid algebra and MMPS systems. Following this, we will extend the MMPS system model, drawing inspiration from conventional algebra, to incorporate input signals, addressing research question 1.(a). An analysis of the MMPS with input signals will then be conducted, considering the properties outlined in [8], to identify the requirements needed to preserve key system properties, thereby answering research question 1.(b). This will result in the development of a systematic approach to implementing input signals in MMPS systems, fully addressing research question 1.

Next, the extended MMPS model with input signals will be used to validate control strategies. We will explore open-loop and closed-loop control in conventional algebra to identify parallels between classical control theory and control in MMPS systems. Leveraging these insights, along with the answers to research question 1, we will define several open-loop and closed-loop control strategies for MMPS systems, answering research question 2.

The closed-loop models developed in response to research question 2 will then serve as the foundation for formulating control goals for the closed-loop systems. We will define control objectives, such as minimizing the system's growth rate or tracking a reference signal answering question 3.(a). Optimization problems will be constructed to identify optimal controllers for these objectives, addressing research question 3.(b). Finally, the insights gained from this research will be applied to evaluate closed-loop control possibilities and optimize the performance of a closed-loop urban railway system, providing, an answer to research question 3.(c).

1-3 Outline

This thesis is organized as follows. Chapter 1 introduces the subject, providing a background and defining the research questions. It also outlines the approach for addressing these questions and concludes with an overview of the thesis structure. Chapter 2 lays the mathematical groundwork, presenting the theory of dioid algebra and the fundamentals of MMPS systems. In Chapter 3, the MMPS canonical form is introduced with the inclusion of input signals, where three distinct input strategies are developed, each defined by unique mathematical characteristics. Chapter 4 examines open-loop control, applying it across the three input strategies to analyze its effects on the system. Practical examples are included for additional clarity. Chapter 5 expands the control framework to closed-loop control, focusing on state feedback. This chapter includes analyses of critical properties and illustrates closed-loop control in MMPS systems through examples. Chapter 6 introduces optimization-based control, outlining control objectives and a key concept in MMPS systems: the existence of distinct regions. The chapter then focuses on applying optimization-based closed-loop control to achieve various objectives. Chapter 7 introduces model predictive control (MPC), a popular optimization-based control approach. It demonstrates how MPC can be applied to an MPL system by incorporating an MMPS controller to create a closed-loop MMPS system. Chapter 8 explores the urban railway system (URS), presenting simulations of both stable and unstable initial conditions. Using insights from previous chapters, optimization-based closed-loop control is applied to enhance system performance through stabilization and optimizing an objective. Chapter 9 concludes the thesis by summarizing its contributions to the field, while Chapter 10 suggests directions for future research.

Chapter 2

Max-min-plus-scaling (MMPS) discrete-event systems

This chapter provides an introduction to the field of dioid algebra and max-min-plus-scaling (MMPS) systems. In the early 1980's max-plus algebra was introduced as a framework to model discrete event (DE) systems [10]. Section 2-1 elaborates on these types of systems. Section 2-2 introduces dioid algebra, a more general concept of max-plus algebra, defining its properties and the use of operators for dioids. In Section 2-3, the definition of MMPS systems is provided with an explanation of the operations used and how these systems can be modeled. Additionally, several canonical forms of MMPS systems relevant to this research are presented. Finally, Section 2-4 offers a comprehensive analysis of MMPS systems, covering key properties and the definition of stability.

2-1 Discrete-event systems

Discrete-event (DE) systems form a large class of dynamical systems in which the evolution of the system is specified by the occurrence of discrete events [11]. This is different to well known discrete time (DT) systems which evolve based on a sampling time. A sampling time is not changing over a time series making the difference between points constant. In DE systems, the difference between events does not have to be the same for different steps. An example of such a system is an urban railway system (URS). In [7], the event cycle (k) is modeled as the train number. Each train will go from station 1 to J but during this simulation. The distance between trains 1 and 2 are not always the same. Therefore, the difference in distance between trains can change for every event cycle, or in the provided case study, the differences in arrival times at several stations might differ.

The changing difference between event cycles results in non-linear behavior when modeling this type of system in conventional (plus-times) algebra. However, there exists a mathematical frameworks that can linearize a certain class of DE systems. This mathematical framework includes dioid algebra, which will be discussed in the next section.

2-2 Dioid algebra

In dioid algebra, a dioid is defined by the following definition:

Definition 2.1. *(Dioid, [10]). A dioid is a set $\mathcal{R}$ endowed with two operations: addition ($\oplus$) and multiplication ($\otimes$) and a set of properties stated in Appendix A.1.*

Using Definition 2.1, two kinds of algebra will be defined; max-plus and min-plus algebra. The overall sets of both algebra's are defined by:

$$\mathcal{R}_{\max} = (\mathbb{R}_{\varepsilon}, \oplus, \otimes, \varepsilon, e) \ , \mathcal{R}_{\min} = (\mathbb{R}_{\top}, \oplus', \otimes', \top, e) \tag{2-1}$$

for which $\varepsilon = -\infty, \top = \infty$ and $e = 0$. $\mathbb{R}_{\varepsilon}$ is the set $\mathbb{R} \cup \{\varepsilon\}$, $\mathbb{R}_{\top}$ is the set $\mathbb{R} \cup \{\top\}$ and $\mathbb{R}_c = \mathbb{R} \cup \{\varepsilon\} \cup \{\top\}$, where $\mathbb{R}$ is the set of real numbers. In this research, often the set $\mathcal{R}$ is used, which can be either $\mathbb{R}$, $\mathbb{R}_{\varepsilon}$, $\mathbb{R}_{\top}$ or $\mathbb{R}_c$. The remaining entries in the sets $\mathcal{R}_{\max}$ and $\mathcal{R}_{min}$ are the operators: max-plus addition ($\oplus$), min-plus addition ($\oplus'$), max-plus multiplication ($\otimes$) and min-plus multiplication ($\otimes'$). The use of these operators are defined as follows:

Definition 2.2. *(Use of operators in dioid algebra, [2]). For the elements $a, b \in \mathcal{R}$. The operators are defined as:*

$$\begin{aligned} a \oplus b &= \max(a, b) \\ a \oplus' b &= \min(a, b) \\ a \otimes b &= a \otimes' b = a + b \end{aligned}$$

Rewriting a model from conventional (plus-times) algebra into dioid algebra is advantageous, as it preserves the applicability of basic operations. These properties are outlined in Appendix A.1.

When defining dioid systems in matrix notation, three operations are used frequently: matrix summation, matrix multiplication and the power of a matrix. These operations are defined as follows:

Definition 2.3. *(Matrix summation, matrix multiplication and the power of a matrix, [2]). The sum of matrices A,$B \in \mathcal{R}^{n\times m}$, denoted by $A \oplus B$, is defined as:*

$$[A \oplus B]_{ij} = [A]_{ij} \oplus [B]_{ij} = \max([A]_{ij}, [B]_{ij}) \tag{2-2}$$

for $i \in n$ and $j \in m$. The multiplication of matrices $A \in \mathcal{R}^{n\times l}$ and $B \in \mathcal{R}^{l\times m}$, denoted by $A \otimes B$, is defined by:

$$[A \otimes B]_{ik} = \bigoplus_{j=1}^{l} [A]_{ij} \otimes [B]_{jk} = \max_{j\in l}\{[A]_{ij} + [B]_{jk}\} \tag{2-3}$$

for $i \in n$ and $k \in m$. Note that matrix multiplications are generally not commutative, i.e.: $A \otimes B \neq B \otimes A$. The power of a matrix $A \in \mathcal{R}^{n\times n}$ with $n, k \in \mathbb{Z}_+$ as:

$$A^{\otimes^k} = \underbrace{A \otimes A \otimes \cdots \otimes A}_{k \ times} \tag{2-4}$$

Dioid algebra can be used to formulate linear systems. For example, we can model a DE system in dioid algebra with the characteristics of using synchronization and no choice. Synchronization means that the next event can only start as soon as all previous events are finished. No choice means that the system is deterministic in a way that it does not have to choose between several possibilities. These types of systems have nonlinear behavior in conventional algebra but when modeled in dioid algebra they can be written into a set of linear equations following the next definition.

Definition 2.4. *(Max-plus linear (MPL) system, [3]). A discrete-event system with only synchronization and no choice as described above can be modeled by a dioid algebraic model of the following form called a max-plus linear (MPL) system:*

$$\begin{aligned} x(k) &= A \otimes x(k-1) \oplus B \otimes u(k) \\ y(k) &= C \otimes x(k) \end{aligned} \tag{2-5}$$

where $A \in \mathcal{R}^{n\times n}, B \in \mathcal{R}^{n\times m}, C \in \mathcal{R}^{l\times n}$ and $k \in \mathbb{Z}_+$ with event counter k, number of outputs l and number of inputs m. The resulting set of equations have a linear relation in dioid algebra.

Similar to plus-times algebra, dioid algebra also includes the concepts of eigenvalues and eigenvectors. In this context, we use additive eigenvalues and eigenvectors, defined as follows:

Definition 2.5. *(Additive eigenvalues and eigenvectors in dioid algebra, [2]). Let $A \in \mathcal{R}^{n\times n}$ be a square matrix. If $\lambda \in \mathcal{R}$ is a scalar and $v \in \mathcal{R}^n$ is a vector that contains at least one finite element such that:*

$$A \otimes v = \lambda \otimes v \tag{2-6}$$

then λ is called an eigenvalue and v an eigenvector of A associated with eigenvalue λ.

In a dioid system, a stable additive eigenvalue indicates the rate of a systems growth. Therefore, this eigenvalue is often referred to as the system's growth rate.

2-3 Max-min-plus-scaling (MMPS) systems

MMPS systems generalize dioid linear systems and are particularly useful for modeling a wide range of discrete-event systems. The same mathematical symbols as those used in dioid algebra will be applied here. MMPS functions are a combination of different operations defined as:

Definition 2.6. *(Max-min-plus-scaling (MMPS) functions, [7]). MMPS functions use four operators: maximization, minimization, addition and multiplication which are combined in the formula:*

$$f_{\text{MMPS}} := x_i|\alpha|\max(f_k, f_l)|\min(f_k, f_l)|f_k + f_l|\beta \cdot f_k \tag{2-7}$$

with $\alpha \in \mathcal{R}$, $\beta \in \mathbb{R}$ and f_k, f_l are MMPS functions over the set $\mathcal{R}$. The notation "|" means "or" and is a recursive definition. The above statement holds componentwise for vector-valued MMPS functions.

The MMPS functions can be used to construct MMPS systems. Consider the following vector:

$$\chi(k) = [x^T(k), x^T(k-1), \ldots, x^T(k-M), u^T(k), w^T(k)]^T, \tag{2-8}$$

where $\chi \in \mathcal{X} \subseteq \mathcal{R}^{n_p}$, $x \in \mathcal{R}^n$ represents the state, $u \in \mathcal{R}^q$ the control input, and $w \in \mathcal{R}^z$ an external signal. MMPS systems can be formulated as state-space models in the form:

$$x(k) = f(\chi(k)), \tag{2-9}$$

where f is a function as defined in Definition 2.6 with variables χ. For explicit systems, the vector changes to:

$$\chi(k) = [x^T(k-1), \ldots, x^T(k-M), u^T(k), w^T(k)]^T, \tag{2-10}$$

and for autonomous systems, we have:

$$\chi(k) = [x^T(k), x^T(k-1), \ldots, x^T(k-M)]^T. \tag{2-11}$$

In the following subsection, we will describe the states of an MMPS system, as well as how to use the operators defined in Definition 2.6 within system equation to model an MMPS system in a structured way.

2-3-1 State and operator description

An MMPS system within a DE framework features states that can be described in two distinct ways [12]. The first type is temporal, which depends on the start and end times of operations within the event cycle k. For instance, in the context of the urban railway system (URS) discussed in 2-1, this refers to the moment a train arrives at a station. The second type represents quantities, such as the number of passengers in a train, which can lead to delays if an excessive number of people need to disembark. For this research, we define the state of an MMPS system as follows:

Definition 2.7. *(State of an MMPS system, [12]). The state of an MMPS systems includes two types combined in the vector:*

$$x = \begin{bmatrix} x_t \\ x_q \end{bmatrix} \tag{2-12}$$

with x_t the temporal signal state and x_q the quantity signal state. For $[x_t]_i$, we define i the time instant at which an event will occur for the k-th time and similar $[x_q]_j$ defines the value of the quantity j at event step k.

In the system equations of MMPS systems appear four basic operators, namely maximization, minimization, addition and scaling. All operators will be elaborated upon in the next definitions.

Definition 2.8. *(Addition I: processing, [7]).*

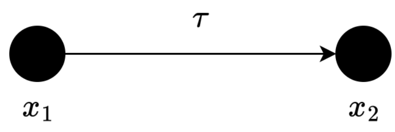


Figure 2-1: Operator example 1

The arrow in Figure 2-1 represents an operation with processing time τ *and the starting and ending times* x_1 *and* x_2 *for an event cycle* k*. This relation can be described by the plus-operation:* $x_2(k) = x_1(k) + \tau$*.*

Definition 2.9. *(Maximization I: sequential processing (no concurrency), [7]). In another situation of Figure 2-1, there are two operations in the same resource that needs to be finished before the next operation can take place (no concurrency). Let* u_1 *be the earliest possible starting time of* x_1 *for the cycle* k*, then the starting time* x_1 *is given by the max-operation:* $x_1(k) = \max(x_1(k-1) + \tau, u_1(k))$*.*

Definition 2.10. *(Maximization II: synchronization, [7]).*

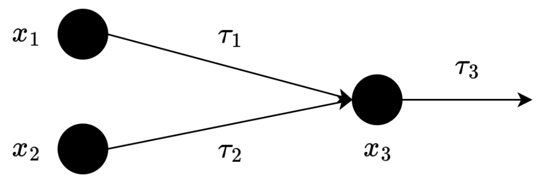


Figure 2-2: Operator example 2

In Figure 2-2, there exist a third operation x_3 *which starts when both previous operations are finished. The starting time* x_3 *is given by the max-operation:* $x_3(k) = \max(x_1(k)+\tau_1, x_2(k)+\tau_2)$*.*

Definition 2.11. *(Minimization: competition, [7]). Consider Figure 2-2 again. This time,* x_3 *will start when one of the previous operations are finished and the product is delivered. The starting time* x_3 *is now given by the min-operation:* $x_3(k) = \min(x_1(k) + \tau_1, x_2(k) + \tau_2)$*.*

Definition 2.12. *(Scaling I: state-dependent processing time, [7]). Consider Figure 2-1 again. We have processing time* τ *which equals an affine function of the state* x*. Such that,* $\tau(k) = \alpha + \beta^T \cdot x(k)$ *with* $\alpha \in \mathbb{R}_+$ *and* $\beta \in \mathbb{R}_+^n$*. Therefore, both are non-negative and* n *is the dimension of the state. The relation between* x_1 *and* x_2 *now includes a scaling-operation:* $x_2(k) = x_1(k) + \alpha + \beta^T \cdot x(k)$*.*

Definition 2.13. *(Scaling II: splitting quantities, [7]). In Figure 2-3, an operator splits the quantity state* x_1 *into two new quantities* x_2 *and* x_3 *with a ratio* η *and* $(1-\eta)$*. The quantities are given by a scaling-operation:* $x_2(k) = \eta \cdot x_1(k)$ *and* $x_3(k) = (1-\eta) \cdot x_1(k)$*.*

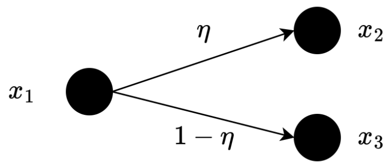


Figure 2-3: Operator example 3

In the following subsection, we will explore various methods of representing MMPS functions within MMPS systems.

2-3-2 Canonical forms of MMPS systems

An MMPS system can be expressed in either implicit or explicit format. An MMPS system in implicit format utilizes the state vector from Eq. (2-8), where the state evolution relies on states within the same event cycle. This format is generally less desirable, as it necessitates information about the current state. However, most MMPS systems can be converted to an explicit format through substitutions, allowing the state evolution to depend solely on past information. This conversion may lead to a significant increase in system size or result in a nested structure.

In contrast, systems presented in explicit form utilize the state vector from Eq. (2-10), where the next state is determined only by previous states and external signals.

A canonical form involves rewriting the system in a standard manner that preserves its behavior [12]. While several canonical forms exist for modeling MMPS systems, this subsection will focus solely on those that are relevant to the present research.

The first two definitions describe systems with different orders of operators:

Definition 2.14. *(Conjunctive form of an MMPS system, [12]). A conjunctive MMPS system describes a model of the form:*

$$x(k) = \min_{i=1,\ldots,K} \max_{j=1,\ldots,n_i} \left(\alpha_{i,j}^T \cdot p(k) + \beta_{i,j} \right) \tag{2-13}$$

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

Definition 2.15. *(Disjunctive form of an MMPS system, [12]). A disjunctive MMPS systems describes a model of the form:*

$$x(k) = \max_{i=1,\ldots,L} \min_{j=1,\ldots,m_i} \left(\sigma_{i,j}^T \cdot p(k) + \rho_{i,j} \right) \tag{2-14}$$

for some integers $L, m_1, \ldots, m_L$, vectors $\sigma_{i,j}$, and real numbers $\rho_{i,j}$.

The distinction between both is the order of the maximization and minimization. Every system in conjunctive form can be rewritten into disjunctive and vice versa [13]. However, there is redundancy when interchanging between forms.

The next form to be defined is the disjunctive matrix form, which is the most general format and will be used extensively throughout this research.

Definition 2.16. *(Implicit ABCD canonical form, [12]). The implicit ABCD canonical form describes an MMPS system in the following form:*

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k))) \tag{2-15}$$

with $A \in \mathcal{R}^{n \times m}$, $B \in \mathcal{R}^{m \times p}$, $C \in \mathcal{R}^{p \times n}$, $D \in \mathcal{R}^{p \times n}$, $x \in \mathcal{R}^n$ *and* $k \in \mathbb{Z}^+$. *The system can be made explicit by defining* $D = 0$.

The ABCD canonical form makes no distinction between the different types of signals discussed in Section 2-3-1. Therefore, we define a more specified form.

Definition 2.17. *(Implicit ABCD canonical form with diverse states, [9]). The implicit ABCD canonical form with diverse states describes an MMPS system in the following form:*

$$\begin{bmatrix} x_t(k) \\ x_q(k) \end{bmatrix} = \underbrace{\left[\begin{array}{c|c} A_t & \varepsilon \\ \hline \varepsilon & A_q \end{array}\right]}_{A} \otimes \Bigg(\underbrace{\left[\begin{array}{c|c} B_t & \top \\ \hline \top & B_q \end{array}\right]}_{B} \otimes' \Bigg(\underbrace{\left[\begin{array}{c|c} C_{11} & C_{12} \\ \hline C_{21} & C_{22} \end{array}\right]}_{C} \cdot \begin{bmatrix} x_t(k-1) \\ x_q(k-1) \end{bmatrix} + \underbrace{\left[\begin{array}{c|c} D_{11} & D_{12} \\ \hline D_{21} & D_{22} \end{array}\right]}_{D} \cdot \begin{bmatrix} x_t(k) \\ x_q(k) \end{bmatrix} \Bigg)\Bigg) \tag{2-16}$$

where $x_t \in \mathcal{R}^{n_t}, x_q \in \mathcal{R}^{n_q}, A_t \in \mathcal{R}^{n_t \times m_t}, A_q \in \mathcal{R}^{n_q \times m_q}$ $B_t \in \mathcal{R}^{m_t \times p_t}, B_q \in \mathcal{R}^{m_q \times p_q}, C_{11}, D_{11} \in \mathcal{R}^{p_q \times n_t}, C_{12}, D_{12} \in \mathcal{R}^{p_t \times n_q}$, $C_{21}, D_{21} \in \mathcal{R}^{p_q \times n_t}$, *and* $C_{22}, D_{22} \in \mathcal{R}^{p_q \times n_q}$. *The values* n_t *and* n_q *represent the amount of temporal and quantity signal states.*

To rewrite systems from one to another canonical form. Some properties of the maximization and minimization operators can be used which are discussed in Appendix A.2. In the next subsection, an example will be provided that evaluates the steps on how to model an MMPS system.

2-3-3 Modeling of an MMPS system

Using the information of the previous subsections, we can model an MMPS system. In this subsection, an example will be provided that elaborates upon the translation from a schematic overview of an DE system to its system equations, and from modeling the system equations into the ABCD canonical form. Suppose we have the situation from Figure 2-4 where a production system is schematically visualized.

The model exists of two machines (M), traveling times (t) and production times (d). The formula's for machine 1 equal:

$$\begin{aligned} x_1(k) &= \max(x_1(k-1) + 5, x_2(k-1)) \\ d_1(k) &= \alpha \cdot (x_2(k-1) - x_1(k-1)) \\ x_2(k) &= x_1(k) + d_1(k) \end{aligned} \tag{2-17}$$

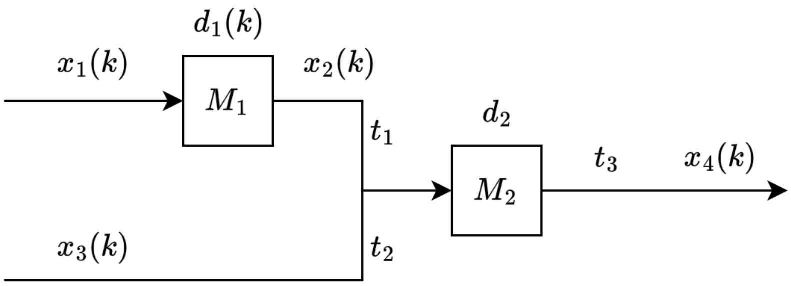


Figure 2-4: Schematic overview of a production system

In these set of system equations, the start of the machine (x_1) is based on the finishing time of the previous product and the incoming stream. The production time is based on a scaling multiplied with the previous production time and finally, the finishing time is a summation between the starting and production times. When combining these equations and using the fact that $\max(A, B) + C = \max(A + C, B + C)$, we get:

$$\begin{aligned} x_2(k) &= \max(x_1(k-1) + 5 + \alpha \cdot (x_2(k-1) - x_1(k-1)), x_2(k-1)) \\ &\qquad + \alpha \cdot (x_2(k-1) - x_1(k-1)) \\ &= \max((1-\alpha) \cdot x_1(k-1) + \alpha \cdot x_2(k-1) + 5, -\alpha \cdot x_1(k-1) \\ &\qquad + (1+\alpha) \cdot x_2(k-1)) \end{aligned} \tag{2-18}$$

Then there is a third incoming line of products (x_3) which will be used together with the output of machine 1 (x_2) in competition to create the final output (x_4) given by the equations:

$$\begin{aligned} x_3(k) &= x_3(k-1) + 4 \\ x_4(k) &= \min(x_2(k) + t_1 + d_2, x_3(k) + t_2 + d_2) + t_3 \\ &= \min(x_2(k) + t_1 + d_2 + t_3, x_3(k-1) + 4 + t_2 + d_2 + t_3) \end{aligned} \tag{2-19}$$

The total set of system equations are as follows:

$$\begin{aligned} x_1(k) &= \max(x_1(k-1) + 5, x_2(k-1)) \\ x_2(k) &= \max((1-\alpha) \cdot x_1(k-1) + \alpha \cdot x_2(k-1) + 5, -\alpha \cdot x_1(k-1) + (1+\alpha) \cdot x_2(k-1)) \\ x_3(k) &= x_3(k-1) + 4 \\ x_4(k) &= \min(x_2(k) + t_1 + d_2 + t_3, x_3(k-1) + 4 + t_2 + d_2 + t_3) \end{aligned} \tag{2-20}$$

The system will be maintained in implicit form. It can be transformed into explicit form by substituting the formula for x_2 into the system equations of x_4. This substitution results in an MMPS system in conjunctive ($\min - \max$) form, which cannot be directly expressed in the explicit ABCD canonical form. Converting it to disjunctive ($\max - \min$) form is a tedious process; therefore, we will represent the system in the implicit ABCD canonical form:

$$\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} = \underbrace{\begin{bmatrix} 5 & 0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 5 & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 \end{bmatrix}}_{A} \otimes (\underbrace{\begin{bmatrix} 0 & \top & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top & \top \\ \top & \top & 0 & \top & \top & \top \\ \top & \top & \top & 0 & \top & \top \\ \top & \top & \top & \top & 0 & \top \\ \top & \top & \top & \top & 4+t_2+d_2+t_3 & t_1+d_2+t_3 \end{bmatrix}}_{B}$$

$$\otimes' (\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1-\alpha & \alpha & 0 & 0 \\ -\alpha & 1+\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \\ x_4(k-1) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}}_{D} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix})) \quad (2\text{-}21)$$

The schematic overview of the production system presented in Figure 2-4 has been reformulated into an implicit MMPS system in ABCD canonical form. In Section 3-3, we will explore various strategies for incorporating input signals into the system equations of MMPS systems, demonstrating how to model them in a format similar to the ABCD canonical form.

2-4 Analysis of MMPS systems

In this section, we will analyze both implicit and explicit MMPS systems concerning several key properties: time-invariance, monotonicity, non-expansiveness and stability. These properties are significant for the following reasons. According to [14], time-invariance indicates that if all events in the system are shifted by the same amount, the system's dynamics will remain unchanged. For example, in a railway system, whether a train departs today or tomorrow, it will follow the same operational dynamics. Monotonicity implies that the system is non-decreasing; thus, if events are delayed, this delay will not lead to compensatory adjustments in subsequent events. Non-expansiveness is essential for constraining the system's dynamics, as highlighted by [15]. This section will define the necessary conditions for these properties based on the works of [12] and [8]. Finally, the stability of MMPS systems will be discussed, as analyzing stability is essential for ensuring that these systems remain bounded and will not grow exponentially.

2-4-1 Requirements for explicit MMPS systems

In this subsection, we will establish the requirements for explicit MMPS systems to ensure that the properties of time-invariance, monotonicity, and non-expansiveness hold. We will utilize the explicit form of the ABCD canonical form defined in Definition 2.17, where $D = 0$. Furthermore, the vectors $\mathbf{1}$ and $\mathbf{0}$ are defined as the unit and zero vector of appropriate dimensions.

To define time-invariance, the property of homogeneity is essential, a homogeneous system is defined as follows:

Definition 2.18. *(Homogeneous system, [8]). Consider a system $x(k+1) = f(x(k))$. This system is called homogeneous if there holds:*

$$f(x(k) + h \cdot \mathbf{1}) = f(x(k)) + h \cdot \mathbf{1} \tag{2-22}$$

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

In other words, a time or event jump added at the state is equal to the same addition to the output of the system. The definition of time-invariance in a system is as follows:

Definition 2.19. *(Time-invariant systems, [8]). A system $x(k) = f(x(k-1))$, where x is a temporal signal, is time-invariant for any delay $h \in \mathbb{R}$ if it holds that:*

$$x(k+1) + h \cdot \mathbf{1} = f(x(k) + h \cdot \mathbf{1}) \tag{2-23}$$

This basically means that delaying a state at instance k results in the exact same delay for $k+1$ and there are no other changes in the system dynamics.

Following [8], an MMPS system is time-invariant if and only if it is homogeneous. Therefore, we want to prove homogeneity. Consider an MMPS system in explicit ABCD canonical form following Definition 2.17 with only temporal signals:

$$x_t(k) = A \otimes (B \otimes' (C \cdot x_t(k-1))) \tag{2-24}$$

Then we can substitute the system inside Definition 2.18:

$$\begin{aligned} &A \otimes (B \otimes' (C \cdot (x_t(k-1) + h \cdot \mathbf{1}))) \\ &\quad = A \otimes (B \otimes' (C \cdot x_t(k-1) + C \cdot h \cdot \mathbf{1})) \end{aligned} \tag{2-25}$$

To prove equality, we use the fact that an max-min-plus (MMP) system is always homogeneous [8]. An MMP system is defined as follows:

Definition 2.20. *(Max-min-plus (MMP) system, [16, 17]). An MMP system is described by expressions in which three kind of operations are used: maximization, minimization and addition. Therefore, an MMP function can be described as [17]:*

$$f_{\text{MMP}} := x_i | \alpha | \max(f_k, f_l) | \min(f_k, f_l) | f_k + f_l \tag{2-26}$$

with $\alpha \in \mathcal{R}$ and f_k, f_l are MMP functions over the set $\mathcal{R}$. The notation "|" means "or" and is a recursive definition. Note that the difference with an MMPS function defined in Definition 2.6 is the scaling operation. The MMP functions can be written in a matrix format following [16]:

$$x(k) = A \otimes (B \otimes' x(k-1)) \tag{2-27}$$

with $A \in \mathcal{R}^{n \times m}$, $B \in \mathcal{R}^{m \times p}$, $x \in \mathcal{R}^n$ and $k \in \mathbb{Z}^+$.

An MMP system is an explicit MMPS system with no scaling such that C is an identity matrix. Therefore, the inequality for an MMP system is equal to:

$$A \otimes (B \otimes' (x(k-1) + h \cdot \mathbf{1})) = A \otimes (B \otimes' x(k-1)) + h \cdot \mathbf{1} \tag{2-28}$$

Combining equations Eq. (2-25) and Eq. (2-28). An MMPS system with only temporal signals is homogeneous if the following equality applies:

$$\begin{aligned} &A \otimes (B \otimes' (C \cdot x_t(k-1) + C \cdot h \cdot \mathbf{1})) \\ &\quad = A \otimes (B \otimes' (C \cdot x_t(k-1))) + C \cdot h \cdot \mathbf{1} \end{aligned} \tag{2-29}$$

Therefore, the system is homogeneous when:

$$\sum_j [C]_{ij} = 1, \, \forall i \tag{2-30}$$

In other words, when the sum of all scaling factors of matrix C in a row is equal to 1. However, as earlier described, states of MMPS systems can exist of both temporal and quantity signals. To define time-invariance in this situation, we introduce partly additive homogeneous systems:

Definition 2.21. *(Partial additive homogeneity, [9]). Consider the systems:*

$$\begin{aligned} x_t(k) &= f_t(x_t(k-1), x_q(k-1)) \\ x_q(k) &= f_q(x_t(k-1), x_q(k-1)) \end{aligned} \tag{2-31}$$

where f_t and f_q are mappings, the state x_t corresponds to the time for event k and x_q for the quantity of event k. This system is partly additive homogeneous if for any real number h

$$\begin{pmatrix} f_t(x_t(k) + h, x_q(k)) \\ f_q(x_t(k) + h, x_q(k)) \end{pmatrix} = \begin{pmatrix} f_t(x_t(k), x_q(k)) + h \\ f_q(x_t(k), x_q(k)) \end{pmatrix} \tag{2-32}$$

Previously we defined that an explicit MMPS system with only temporal signals is homogeneous if all scaling factors in a row sum up to 1 in the C matrix. For defining time-invariance in MMPS with temporal and quantity signal states, we are using the canonical form of Definition 2.17. The equality for temporal states from Eq. (2-29) is extended to the following two equalities:

$$\begin{aligned} &A_t \otimes (B_t \otimes' (C_{11} \cdot (x_t(k-1) + h \cdot \mathbf{1}) + C_{12} \cdot x_q(k-1))) \\ &\quad = A_t \otimes (B_t \otimes' (C_{11} \cdot x_t(k-1) + C_{12} \cdot x_q(k-1))) + h \cdot \mathbf{1} \\ &A_q \otimes (B_q \otimes' (C_{21} \cdot (x_t(k-1) + h \cdot \mathbf{1}) + C_{22} \cdot x_q(k-1))) \\ &\quad = A_q \otimes (B_q \otimes' (C_{21} \cdot x_t(k-1) + C_{22} \cdot x_q(k-1))) \end{aligned} \tag{2-33}$$

Again using the fact that an MMP systems is always homogeneous [8]. The equalities can be written as:

$$\begin{aligned}
&A_t \otimes (B_t \otimes' (C_{11} \cdot x_t(k-1) + C_{12} \cdot x_q(k-1) + C_{11} \cdot h \cdot \mathbf{1})) \\
&\quad = A_t \otimes (B_t \otimes' (C_{11} \cdot x_t(k-1) + C_{12} \cdot x_q(k-1))) + C_{11} \cdot h \cdot \mathbf{1} \\
&A_q \otimes (B_q \otimes' (C_{21} \cdot x_t(k-1) + C_{22} \cdot x_q(k-1) + C_{21} \cdot h \cdot \mathbf{1})) \\
&\quad = A_q \otimes (B_q \otimes' (C_{21} \cdot x_t(k-1) + C_{22} \cdot x_q(k-1)))
\end{aligned} \tag{2-34}$$

Therefore, it is required that $C_{11} \cdot h \cdot \mathbf{1} = h \cdot \mathbf{1}$ and $C_{21} \cdot h \cdot \mathbf{1} = \mathbf{0}$ which result in the following requirement for the C matrix:

$$\sum_{s=1}^{n_t} [C_{11}]_{rs} = 1, \forall r \text{ and } \sum_{s=1}^{n_q} [C_{21}]_{rs} = 0, \forall r \tag{2-35}$$

In other words for the scaling factors in each row of the C_{11} matrix should sum up to 1 and for C_{21} to 0. For the other two entries in the C matrix there are no restrictions to satisfy time-invariance. The proof is gathered from [9].

Next, the monotonicity of MMPS systems is analyzed based on the findings in [8]. For a system to be called monotone the following must hold:

$$\text{if } x \leq y \text{ then } f(x) \leq f(y) \tag{2-36}$$

An MMPS system with only temporal signals is monotonic if and only if $[C]_{ij} \geq 0 \ \forall i, j$. To prove this, we use the fact that an MMP system is monotonic [18]:

$$\begin{aligned}
A \otimes (B \otimes' x(k-1)) &\leq A \otimes (B \otimes' y(k-1)) \\
\text{when } x(k-1) &\leq y(k-1)
\end{aligned} \tag{2-37}$$

This means an MMPS system with only temporal signals is monotonic if:

$$\begin{aligned}
A \otimes (B \otimes' (C \cdot x_t(k-1))) &\leq A \otimes (B \otimes' (C \cdot y_t(k-1))) \\
\text{when } x_t(k-1) &\leq y_t(k-1)
\end{aligned} \tag{2-38}$$

This is true if:

$$\begin{aligned}
C \cdot x_t(k-1) &\leq C \cdot y_t(k-1) \\
\text{when } x_t(k-1) &\leq y_t(k-1)
\end{aligned} \tag{2-39}$$

Equation 2-39 satisfies iff $[C]_{ij} \geq 0 \ \forall i, j$. The proof is gathered from [8] and based on MMPS systems with only temporal signals. The analysis of monotonicity for MMPS with both temporal and quantity signals is outside the scope of this research.

Finally, the non-expansive property will be discussed based on the findings in [8]. For this property, it is assumed that the system is already time-invariant. Then a system is non-expansive in the ℓ-norm if:

$$||f(x) - f(y)||_\ell \leq ||x - y||_\ell \tag{2-40}$$

We use that an MMP system is non-expansive, the proof is outside of this research and can be found in [18]. Then a time-invariant MMPS system is non-expansive if $|[C]_{ij}| \leq 1 \; \forall i, j$. This can be shown by using the property that an MMP system is non-expansive such that:

$$\begin{aligned} ||A \otimes (B \otimes' x(k-1) - A \otimes (B \otimes' y(k-1))|| \leq ||x(k-1) - y(k-1)|| \\ ||A \otimes (B \otimes' (x(k-1) - y(k-1)))|| \leq ||x(k-1) - y(k-1)|| \end{aligned} \tag{2-41}$$

In dioid algebra, the operations $+$ and $-$ are distributive over $\otimes$, $\otimes'$. Then for the infinity norm, an MMPS is non-expansive if:

$$A \otimes (B \otimes' (C \cdot (x(k-1) - y(k-1))))|| \leq ||x(k-1) - y(k-1)|| \tag{2-42}$$

Let $w(k-1) = x(k-1) - y(k-1)$ and by using 2-41 we can simply to:

$$||C \cdot w(k-1)|| \leq ||w(k-1)|| \tag{2-43}$$

Assume that $|[C]_{ij}| \leq 1$. Then $\forall i$:

$$\sum_j c_{ij} w_j(k-1) \leq \max(|w_j(k-1)|) \tag{2-44}$$

This inequality only applies for $|[C]_{ij}| \leq 1$ because for any $|[C]_{ij}| > 1$ the inequality is violated. This proof only applies to MMPS systems with only temporal signals. Non-expansiveness of MMPS systems with both temporal and quantity signal states are outside the scope of this research.

2-4-2 Requirements for implicit MMPS systems

For implicit systems, we are only evaluating the property of time-invariance. This property is crucial for MMPS systems because time shifts should not influence the dynamics of the systems. We use the implicit ABCD canonical form of Definition 2.17. Following [19], an implicit MMPS systems is time-invariant when:

$$\begin{aligned} \sum_{j=1:n_t} ([C_{11}]_{ij}) + \sum_{j=1:n_t} ([D_{11}]_{ij}) = 1, \forall i \\ \sum_{j=1:n_t} ([C_{21}]_{ij}) + \sum_{j=1:n_t} ([D_{21}]_{ij}) = 0, \forall i \end{aligned} \tag{2-45}$$

The proof is provided in [19] and is an extension of the situation with explicit systems described in Section 2-4-1. The requirements for monotonicity and non-expansiveness of implicit MMPS systems are outside the scope of this research.

2-4-3 Stability of MMPS systems

For evaluating the stability of an MMPS systems, we are mainly focusing on defining eigenvalues, eigenvectors and boundedness of the MMPS systems. The introduction of eigenvalues and eigenvectors of MMPS systems is an extended version of Definition 2.5. The vectors **1** and **0** are defined as the unit and zero vectors of appropriate dimensions.

Definition 2.22. *(Additive eigenvalue and eigenvector, [19]). The time-invariant DE system:*

$$\begin{aligned} x(k) &= f(p(k)),\ x \in \mathcal{R}^n \\ f &: \mathcal{R}^n \to \mathcal{R}^n \end{aligned} \tag{2-46}$$

has an additive eigenvalue if there exists a real number $\lambda \in \mathbb{R}$ and a vector $v \in \mathbb{R}^n$ such that

$$\begin{aligned} f(v) &= v + \lambda \cdot s \\ s &= [\mathbf{1}_{n_t}^T, \mathbf{0}_{n_q}^T]^T \end{aligned} \tag{2-47}$$

where n_t and n_q are the number of temporal and quantity signal states. The scalar λ is then called an eigenvalue and the vector v is called an eigenvector.

Time-invariant explicit MMPS systems can be linearized following an algorithm introduced in [8], this computation will be used later. First, the linearized MMPS system has multiplicative eigenvalues and eigenvectors which can be defined as:

Definition 2.23. *(Multiplicative eigenvalue and eigenvector, [19]). The normalized time-invariant DE system:*

$$\tilde{x}(k) = M \cdot \tilde{x}(k-1),\ \tilde{x} \in \mathbb{R}^n \text{ and } M \in \mathbb{R}^{n \times n} \tag{2-48}$$

has an multiplicative eigenvalue if there exists a real number $\mu \in \mathbb{R}$ and a vector $w \neq 0 \in \mathbb{R}^n$ such that

$$M \cdot w = \mu \cdot w \tag{2-49}$$

The scalar μ is then called an eigenvalue and the vector w is called an eigenvector.

For the remainder of this research, the additive eigenvalue and eigenvector will also be referred to as the growth rate and fixed-point. The multiplicative eigenvalue and eigenvector will be referred to as the eigenvalue and eigenvector.

For defining internal stability of an MMPS system, we are introducing the max-plus Hilbert's projective norm:

Definition 2.24. *(Max-plus Hilbert projective norm, [2]). The max-plus Hilbert projective norm of a vector $x \in \mathbb{R}^n$ in max-plus algebra is defined as:*

$$\|x\|_{\mathbb{P}} = \max_{i \in \underline{n}}(x_i) - \min_{j \in \underline{n}}(x_j) \tag{2-50}$$

This norm looks at the maximal difference between vectors and therefore evaluates if the states do not diverge. Following [9], we define a time-invariant MMPS system with multiple growth rates in a normalized form.

Definition 2.25. *(Normalized MMPS system, [9]). An MMPS system in normalized form can be written as:*

$$\tilde{x}_\theta(k) = \tilde{A}_\theta \otimes (\tilde{B}_\theta \otimes' (C \cdot \tilde{x}_\theta(k-1))) \tag{2-51}$$

for $\theta \in \{1, \ldots, S\}$ with S the amount of growth rates. The matrices have structure $\tilde{A}_\theta \in \mathcal{R}^{n \times m}$,$\tilde{B}_\theta \in \mathcal{R}^{m \times p}$ and $C \in \mathcal{R}^{p \times n}$. The normalization process can be found in [9].

The normalized matrices consist of both a temporal and quantity part when working with MMPS systems that include both types of signals. The normalized matrices have the form:

$$\tilde{A}_\theta = \left[\begin{array}{c|c} \tilde{A}_{t\theta} & \varepsilon \\ \hline \varepsilon & \tilde{A}_{q\theta} \end{array}\right], \tilde{B}_\theta = \left[\begin{array}{c|c} \tilde{B}_{t\theta} & \top \\ \hline \top & \tilde{B}_{q\theta} \end{array}\right] \tag{2-52}$$

For all future definitions, the types of signals can be split in temporal and quantity but for simplicity we use the combined state vector which results in the combined matrices $\tilde{A}_\theta$ and $\tilde{B}_\theta$. Next, we define a region Ω_θ that contains all vectors $x \in \mathcal{R}^n$ such that:

$$\tilde{A}_\theta \otimes (\tilde{B}_\theta \otimes' (C \cdot x(k))) = G_{A_\theta} \cdot G_{B_\theta} \cdot C \cdot x(k) \tag{2-53}$$

The "G" matrices are footprint matrices and are defined as:

Definition 2.26. *(Footprint matrices, [9]). The footprint are composed by looking at the entries of the normalized system matrices.*

$$[G_{A_\theta}]_{ij} = \begin{cases} 1 & \text{if } \left[\tilde{A}_\theta\right]_{ij} = 0 \\ 0 & \text{if } \left[\tilde{A}_\theta\right]_{ij} < 0 \end{cases}, \quad [G_{B_\theta}]_{jl} = \begin{cases} 1 & \text{if } \left[\tilde{B}_\theta\right]_{jl} = 0 \\ 0 & \text{if } \left[\tilde{B}_\theta\right]_{jl} > 0 \end{cases} \tag{2-54}$$

These footprint matrices define the location of zero's in the normalized matrix. They can be used to define a linear programming (LP) problem to find a possible growth rate and a corresponding equilibrium point [9]. The footprint matrices can also be used to reformulate a normalized MMPS system as a linear system in plus-times algebra.

Definition 2.27. *(Normalized MMPS system as linear system, [9]). For all $\tilde{x}_\theta(k) \in \Omega_\theta$ and k a positive integer we get:*

$$\begin{aligned} \tilde{x}_\theta(k) &= M_\theta \cdot \tilde{x}_\theta(k-1) \\ M_\theta &= G_{A_\theta} \cdot G_{B_\theta} \cdot C \end{aligned} \tag{2-55}$$

Based on the preservation of properties of the M matrix, the matrix will have at least one eigenvector equal to one [9]. Then, following [20], an autonomous DE system is max-plus bounded buffer stable if for every initial state, $x_0 \in \mathbb{R}^n$, there exist a bound $M(x_0) \in \mathbb{R}$ such that the states are bounded in Hilbert's projective norm.

Furthermore, we use a notion from [21] that the system is stable when the system matrix M_θ has eigenvalues less than or equal to one and all Jordan blocks corresponding to magnitude one are 1×1. When an eigenvalue is larger than 1 or there are Jordan blocks of not size 1×1 corresponding to magnitude 1, the system is unstable. Note that stable linearized systems in this case means that states are not growing but are also not necessarily equal to 0, because of the existence of eigenvalues of magnitude 1. Applying Hilbert's projective norm results in boundedness of state vector $||\tilde{x}_\theta||_\mathbb{P}$ since:

$$\begin{aligned} ||x_\theta||_\mathbb{P} &= ||\tilde{x}_\theta + x_{e\theta} + \lambda_\theta \cdot k \cdot \mathbf{1}||_\mathbb{P} \\ &= ||\tilde{x}_\theta + x_{e\theta}||_\mathbb{P} \leq ||\tilde{x}_\theta||_\mathbb{P} + ||x_{e\theta}||_\mathbb{P} \end{aligned} \tag{2-56}$$

where $x_{e\theta}$ is the equilibrium point of the original autonomous MMPS systems when the growth rate is λ_θ. Hence, the MMPS system is max-plus bounded buffer stable at the temporal growth rate λ_θ in the region Ω_θ [9].

Finally, we define a special type of MMPS systems called topical systems. A system is topical when it is time-invariant, monotonic and non-expansive. Following [8], topical systems have 1 eigenvalue which is unique. Also, this eigenvalue is always stable. For this reason, it is worthwhile to analyse MMPS systems based on these three properties.

Chapter 3

Adding inputs to MMPS systems

The current state of the art concerning max-min-plus-scaling (MMPS) systems primarily consists of research focused on autonomous systems. Autonomous systems operate independently, without external influences. To date, no mathematical foundation has been established for incorporating inputs into MMPS systems. This chapter proposes an MMPS framework that includes inputs. Section 3-1 introduces the addition of input signals to continuous-time systems. Section 3-2 extends the ABCD canonical form by incorporating input signals into the MMPS system creating the ABCDE canonical form. Furthermore, the algebraic properties of MMPS systems, discussed in Section 2-4, are re-analyzed for the newly defined ABCDE canonical form. Section 3-3 presents three input strategies that cover all possible methods of adding input signals to MMPS system equations. Each input strategy is expressed in the ABCDE canonical form, with examples demonstrating how input signals can be included in the system equations.

3-1 Input signals in continuous time systems

In this section, a very general introduction of adding inputs to systems in conventional algebra is discussed. The information in this section is based on [22]. The most general description of a continuous time (CT) control system is given by the following equations:

$$\begin{aligned} \dot{x} &= f(t, x, u) \\ y &= h(t, x, u) \end{aligned} \tag{3-1}$$

where f and h are functions of time t, states x and inputs u. The outputs are denoted by y. We speak of an autonomous system when the functions are time-invariant and without control resulting in the form:

$$\dot{x} = f(x) \tag{3-2}$$

In autonomous form, you can only look into the behaviour of your system unable to influence it. However, this notation is in CT format. It is already discussed in Section 2-1 that MMPS systems are discrete event (DE) systems. The notation of an autonomous, DE system is given by:

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

with k the event counter. The autonomous system can be expanded to a form where the system can be influenced. Input signal $u(k)$ is added such that:

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

This form accommodates both linear and non-linear functions. In special cases, where the DE system is time-invariant and is a linear combination of states and inputs. The DE system can be written into the commonly used state space form:

$$\begin{aligned} x(k+1) &= A \cdot x(k) + B \cdot u(k) \\ y(k) &= C \cdot x(k) + D \cdot u(k) \end{aligned} \tag{3-5}$$

This form has some resemblance to the ABCD canonical form used in MMPS systems because of the linear combination of states and matrices. The linear state-space form is taken as inspiration for an ABCD canonical form with additional input entries. In the next subsection, this form will be introduced.

3-2 General framework for MMPS systems with input signals

To define a general form accommodating input signals, we expand upon Definition 2.17 of Section 2-3-2 introducing the ABCD canonical form. For now, we assume that the input signal only consists of $u(k)$. Later, when looking into closed-loop control, also $u(k-1)$ will be added. We define the ABCDE canonical form where also input signals can be implemented inside the system:

Definition 3.1. *(Implicit ABCDE Canonical form). The ABCD canonical form of Definition 2.17 will be extended with an additional input matrix (E) multiplied with input vectors. The input vector will also consist of both temporal and quantity input signals.*

$$\begin{aligned} \begin{bmatrix} x_t(k) \\ x_q(k) \end{bmatrix} = \underbrace{\left[\begin{array}{c|c} A_t & \varepsilon \\ \hline \varepsilon & A_q \end{array}\right]}_{A} \otimes \Bigg(\underbrace{\left[\begin{array}{c|c} B_t & \top \\ \hline \top & B_q \end{array}\right]}_{B} \otimes' \Bigg(\underbrace{\left[\begin{array}{c|c} C_{11} & C_{12} \\ \hline C_{21} & C_{22} \end{array}\right]}_{C} \cdot \begin{bmatrix} x_t(k-1) \\ x_q(k-1) \end{bmatrix} \\ + \underbrace{\left[\begin{array}{c|c} D_{11} & D_{12} \\ \hline D_{21} & D_{22} \end{array}\right]}_{D} \cdot \begin{bmatrix} x_t(k) \\ x_q(k) \end{bmatrix} + \underbrace{\left[\begin{array}{c|c} E_{11} & E_{12} \\ \hline E_{21} & E_{22} \end{array}\right]}_{E} \cdot \begin{bmatrix} u_t(k) \\ u_q(k) \end{bmatrix} \Bigg)\Bigg) \end{aligned} \tag{3-6}$$

For the remainder of this chapter, we will be working with explicit MMPS systems and therefore, $D = 0$.

By employing this general form, various types of inputs can be integrated into the system. In the next section, we will evaluate different methods for incorporating input signals into MMPS systems. However, before modeling these systems, we must define requirements for the E matrix to ensure that the properties discussed in Section 2-4 are preserved.

3-2-1 Analysis of MMPS systems with input signals

The three properties from Section 2-4 will be validated on the newly defined ABCDE canonical form in Definition 3-6. The analysis will focus on the explicit case where $D = 0$, as this allows for evaluating the influence of the input matrix E. Thus, analyzing explicit systems is considered sufficient for this research. The three properties under discussion are monotonicity, non-expansiveness, and time-invariance. The vectors $\mathbf{1}$ and $\mathbf{0}$ are defined as the unit and zero vectors of appropriate dimensions.

The first property to be discussed is monotonicity. Lemma 2 in [8] establishes the monotonicity of the autonomous MMPS system. This lemma can be extended to show that an MMPS system with additional inputs remains monotonic if:

$$\begin{aligned} A \otimes (B \otimes' (C \cdot x_1(k-1) + E \cdot u(k))) &\leq A \otimes (B \otimes' (C \cdot x_2(k-1) + E \cdot u(k))) \\ \text{when } x_1(k-1) &\leq x_2(k-1) \end{aligned} \tag{3-7}$$

Assuming equal inputs are applied to both systems, and given that max-min-plus (MMP) systems are inherently monotonic [18], the inequality simplifies to:

$$\begin{aligned} C \cdot x_1(k-1) + E \cdot u(k) &\leq C \cdot x_2(k-1) + E \cdot u(k) \\ \text{when } x_1(k-1) &\leq x_2(k-1) \end{aligned} \tag{3-8}$$

Generally, the additional inputs have no influence on the monotonicity of the systems. The system will be monotone if all entries of C satisfy $[C]_{i,j} \geq 0$. Next, Lemma 3 in [8] evaluates the non-expansiveness property. A similar situation arises where the system is non-expansive when:

$$\begin{aligned} &||A \otimes (B \otimes' ((C \cdot x_1(k-1) + E \cdot u(k)) - (C \cdot x_2(k-1) + E \cdot u(k))))|| \\ &\qquad \leq ||x_1(k-1) - x_2(k-1)|| \\ &||A \otimes (B \otimes' (C \cdot (x_1(k-1) - x_2(k-1))))|| \\ &\qquad \leq ||x_1(k-1) - x_2(k-1)|| \end{aligned} \tag{3-9}$$

The additional inputs are omitted in the equations, resulting in the same inequality as in [8]. Therefore, when assuming equal inputs in both systems, the non-expansiveness of the system is independent of the input structure. The system is non-expansive when: $|[C]_{i,j}| \leq 1$.

For time-invariance, we need to delve deeper into the equations of the system. As per [8], we understand that an autonomous MMPS system is time-invariant when:

$$A \otimes (B \otimes' (C \cdot (x(k-1) + h \cdot \mathbf{1}))) = A \otimes (B \otimes' (C \cdot x(k-1))) + h \cdot \mathbf{1} \tag{3-10}$$

for any $h \in \mathbb{R}$. This accounts for a system with only temporal signals. However, MMPS system can consist of both temporal and quantity signals. We define a vector: $s = [\mathbf{1}_{n_t}^T, \mathbf{0}_{n_q}^T]^T$ where n_t the amount of temporal signals and n_q the amount of quantity signals. The system is time-invariant when it is partly additive homogeneous, such that for some $h \in \mathbb{R}$:

$$\begin{aligned} &A \otimes (B \otimes' (C \cdot (x(k-1) + h \cdot s))) \\ &= A \otimes (B \otimes' (C \cdot x(k-1))) + h \cdot s \end{aligned} \tag{3-11}$$

For now, we assume that the input vector is not a function of previous states. In that case, extending the equations from Eq. (3-11) with inputs and using Eq. (3-10), we derive that the system is time-invariant for some $h \in \mathbb{R}$ when:

$$\begin{aligned} &A \otimes (B \otimes' (C \cdot (x(k-1) + hs) + E \cdot (u(k) + h \cdot s))) \\ &= A \otimes (B \otimes' (C \cdot x(k-1)) + E \cdot u(k))) + (C + E) \cdot h \cdot s \end{aligned} \tag{3-12}$$

From this equation we know that time-invariance applies when: $(C+E) \cdot h \cdot s = h \cdot s$. Therefore, the system is time-invariant when:

$$\begin{aligned} &\sum_{j=1:n_t} [C_{11}]_{ij} + \sum_{l=1:n_{u_t}} [E_{11}]_{il} = 1 \ \forall i \\ &\sum_{j=1:n_t} [C_{21}]_{ij} + \sum_{l=1:n_{u_t}} [E_{21}]_{il} = 0 \ \forall i \end{aligned} \tag{3-13}$$

The other entries of the C and E matrices have no requirements. With this information, we start to explore the design of different input strategies. These strategies can be implemented through various uses of the input matrix (E). An overview of the input strategies is provided in the next section.

3-3 Input strategies for MMPS systems

In this section, three types of techniques to include input signals in MMPS systems will be discussed. Generally, the explicit form of the MMPS system with inputs from Definition 3.1 will be used. The input strategies will be divided based on mathematical differences. When discussing input strategies we speak of two types of inputs; temporal and quantity signal inputs. Temporal signal inputs can be identified by its growing nature, quantity signal inputs are bounded. Three different input strategies will be covered by using schematic overviews, system equations and multiple examples.

3-3-1 Introduction of input strategy 1

The first input strategy discussed, covers the situation that inputs already exist in the original system equations. A schematic representation can be found in Figure 3-1

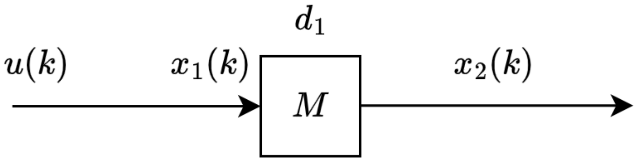


Figure 3-1: Schematic overview for input strategy 1

The schematic overview consists of a machine (M) with production time (d_1). For the entering lane of products, we use the sequential processing operation. Writing out the system equations from Figure 3-1 gives:

$$\begin{aligned} x_1(k) &= \max(x_2(k-1), u(k)) \\ x_2(k) &= x_1(k) + d_1 \end{aligned} \tag{3-14}$$

The starting time of the machine $(x_1(k))$ is dependent on the previous finishing time $(x_2(k-1))$ and the input stream $(u(k))$. The input can be used to force the machine to start working on the new cycle. In the example of Figure 3-1 all states and inputs are temporal signals. However, it is also possible that there is a similar combination of quantity signals states with quantity signal inputs. Cross combinations of different types of signals are not likely using this input strategy because it will change the nature of the states; a temporal signal state will become bounded when being influenced by a quantity signal input and a quantity signal state will be growing when being subjected to a temporal signal input.

The general structure of input strategy 1 is defined as follows:

Definition 3.2. *(General definition of input strategy 1). We use the explicit canonical form of Definition 3.1:*

$$\begin{bmatrix} x_t(k) \\ x_q(k) \end{bmatrix} = \underbrace{\left[\begin{array}{c|c} A_t & \varepsilon \\ \hline \varepsilon & A_q \end{array}\right]}_{A} \otimes (\underbrace{\left[\begin{array}{c|c} B_t & \top \\ \hline \top & B_q \end{array}\right]}_{B} \otimes' (\underbrace{\left[\begin{array}{c|c} C_{11} & C_{12} \\ \hline C_{21} & C_{22} \end{array}\right]}_{C} \cdot \begin{bmatrix} x_t(k-1) \\ x_q(k-1) \end{bmatrix} + \underbrace{\left[\begin{array}{c|c} E_{11} & E_{12} \\ \hline E_{21} & E_{22} \end{array}\right]}_{E} \cdot \begin{bmatrix} u_t(k) \\ u_q(k) \end{bmatrix})) \tag{3-15}$$

Because input strategy 1 covers the situation that input signals are existing in the original system equations. No additional augmentation of system matrices is necessary. However, it is important to preserve time-invariance such that Eq. (3-13) applies.

To further illustrate the design approach of input strategy 1, the system from Figure 3-1 will be used as an example.

Example 3.1. *Production unit example for input strategy 1*

Consider the production unit from Figure 3-1 with u representing the time instant at which a system will be fed a product for the k-th time. Then x_1 denotes the time instance at which the machine starts for the k-th time, and x_2 represents the time the machine finishes. Therefore, both states are temporal signals. The production time (d_1) is replaced with a state dependant production time ($d_1(k)$). The corresponding formulas for the production unit are:

$$\begin{aligned} x_1(k) &= \max(x_2(k-1), u(k)) \\ x_2(k) &= x_1(k) + d_1(k) \\ d_1(k) &= \alpha \cdot (x_1(k) - x_1(k-1)) \end{aligned} \tag{3-16}$$

The scaling factor α is a variable dependent on the production machine. Rewriting the system equations in explicit form, we obtain the MMPS system:

$$\begin{aligned} x_1(k) &= \max(x_2(k-1), u(k)) \\ x_2(k) &= \max(-\alpha \cdot x_1(k-1) + (1+\alpha) \cdot x_2(k-1), -\alpha \cdot x_1(k-1) + x_2(k-1) + \alpha \cdot u(k), \\ &\qquad -\alpha \cdot x_1(k-1) + \alpha \cdot x_2(k-1) + u(k), -\alpha \cdot x_1(k-1) + (1+\alpha) \cdot u(k)) \end{aligned} \tag{3-17}$$

These system equations can be incorporated into the general MMPS system structure. All scaling factors multiplied with states will be placed in the C_{11} matrix, and all scaling factors multiplied with inputs will be placed in the E_{11} matrix. We obtain the following matrices:

$$C = C_{11} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ -\alpha & 1+\alpha \\ -\alpha & 1 \\ -\alpha & \alpha \\ -\alpha & 0 \end{bmatrix}, \; E = E_{11} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \alpha \\ 1 \\ 1+\alpha \end{bmatrix} \tag{3-18}$$

It is evident that the requirement for time-invariance from Eq. (3-13) holds because: $\sum_i([C_{11}]_{1i} + [E_{11}]_{1i}) = 1$ for all rows, independent of the value assigned to α. The final MMPS system can be formed by implementing C_{11} as the C matrix and E_{11} as the E matrix because this system only has temporal signal states and inputs. For the production unit example, the A and B matrices have no additional information and can be modeled as:

$$A = \begin{bmatrix} 0 & 0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & 0 & 0 & 0 \end{bmatrix}, \; B = \begin{bmatrix} 0 & \top & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top & \top \\ \top & \top & 0 & \top & \top & \top \\ \top & \top & \top & 0 & \top & \top \\ \top & \top & \top & \top & 0 & \top \\ \top & \top & \top & \top & \top & 0 \end{bmatrix} \tag{3-19}$$

Conclusively, this input strategy can be used to model MMPS systems that have already input signals inside its system equations. In the next chapter, control will be applied on this input strategy to evaluate the execution of this design.

3-3-2 Introduction of input strategy 2

The second input strategy covers situations where the input signal is used to influence an MMPS system after the scaling step of the system equations. Therefore, the input will be added during either the maximization or minimization step. A schematic overview of a situation is given in Figure 3-2.

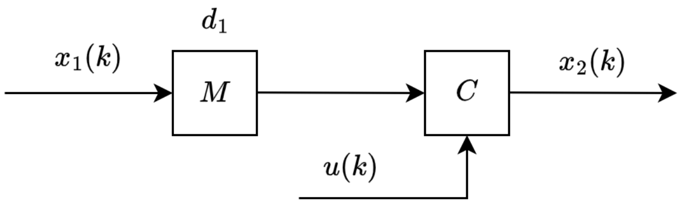


Figure 3-2: Schematic overview for input strategy 2

The schematic overview consists of a machine (M) with production time (d_1). After production the product will go through a control device (C) to create state x_2. The input is used for influencing purposes. The system equation from Figure 3-2 is as follows:

$$x_2(k) = \min(x_1(k) + d_1, u(k)) \tag{3-20}$$

In this case, whenever the production unit is too slow, the control device can implement an additional input to let the whole machine continue. This could be useful if there is a malfunction in the machine. This additional input can also be placed in the maximization step, in that case, it will slow down the system.

The input signal can be either a temporal and quantity signal. However, cross combinations of states with different types of input signals is again not likely when using this input strategy for the same reason as input strategy 1. Input strategy 2 will result in an expansion of the A, B and C system matrices. The general structure is defined as follows:

Definition 3.3. *(General definition of input strategy 2). For this input strategy, we distinguish two situations. First, where an additional input is added during the minimization step also referred to as the competition operation. An input using competition is able to force the system to a lower value or to upper bound the system. The matrix structure is defined as:*

$$x(k) = A \otimes (\begin{bmatrix} B & B_u \end{bmatrix} \otimes' (\begin{bmatrix} C \\ 0 \end{bmatrix} \cdot x(k-1) + \begin{bmatrix} 0 \\ E \end{bmatrix} \cdot u(k))) \tag{3-21}$$

The additional part of the B matrix (B_u) applies the input in the system equations. The augmentation of C and E makes sure that the input is fed through the system unaffected.

Similarly, additional inputs can be added during the maximization step also referred to as the synchronization operation. Which can be used to lower bound the system or force the system to increase its growth rate. The matrix structure is defined as:

$$x(k) = \begin{bmatrix} A & A_u \end{bmatrix} \otimes (\begin{bmatrix} B & \top \\ \top & B_I \end{bmatrix} \otimes' (\begin{bmatrix} C \\ 0 \end{bmatrix} \cdot x(k-1) + \begin{bmatrix} 0 \\ E \end{bmatrix} \cdot u(k))) \tag{3-22}$$

The additional part of the A matrix (A_u) adds the input to the maximization. The augmentation of B with an identity matrix (B_I) feeds through the input unaffected. Note that identity in max-plus is equal to zero's on the diagonal.

To further illustrate input strategy 2, an example will be used.

Example 3.2. *Numerical example for input strategy 2*

A possible system equation when evaluating input strategy 2 could be:

$$x(k) = \max(x(k-1) + 6, u(k)) \tag{3-23}$$

The system equation can be modeled in the structure of Eq. (3-22) as follows:

$$x(k) = \underbrace{\begin{bmatrix} 6 & 0 \end{bmatrix}}_{A} \otimes (\underbrace{\begin{bmatrix} 0 & \top \\ \top & 0 \end{bmatrix}}_{B} \otimes' (\underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{C} \cdot x(k-1) + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{E} \cdot u(k))) \tag{3-24}$$

In this case, both the C and E matrices gain an additional row, which consequently causes the A and B matrices to expand as well. The zero entry in the A matrix introduces the input to the system, while the augmentation of the B matrix allows the input to pass through to the maximization step unaffected.

3-3-3 Introduction of input strategy 3

Input Strategy 3 addresses a scenario where inputs are introduced directly during the scaling step of an MMPS system. A schematic representation of this strategy is provided in Figure 3-3.

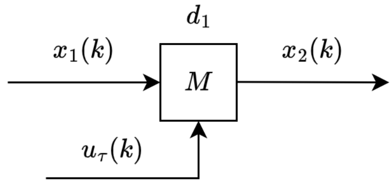


Figure 3-3: Schematic overview for input strategy 3

The primary distinction between Input Strategy 2 and Input Strategy 3 lies in where the input signal is added to the system. In the previous strategy, inputs were incorporated after the scaling phase, additional to the maximization or minimization.

In Input Strategy 3, as shown in Figure 3-3, inputs are added during the production phase, modifying the production time directly. For instance, with a machine (M) that has a production time of d_1, the input production time is represented by $u_\tau(k)$. The system equation can be expressed as follows:

$$x_2(k) = x_1(k) + d_1 + u_\tau(k) \tag{3-25}$$

It's important to note that u_τ can take negative values as long as the combined production time: $d_1+u_\tau(k)$, remains positive. Since the input is added during scaling, it must be bounded to avoid exponential behaviour. Therefore, the input should either represent a time difference or a quantity signal. Time differences help maintain the system's time-invariance, preventing exponential growth during scaling. A time difference is defined as follows:

Definition 3.4. *(Time difference). In cases where direct temporal signal inputs would violate time-invariance and cause exponential behavior, time differences must be used. For time differences, the input matrix E must satisfy the following conditions:*

$$\begin{aligned} &\sum_i([E_{11}]_{li}) = 0\ \forall l \\ &\sum_r([E_{21}]_{tr}) = 0\ \forall t \end{aligned} \tag{3-26}$$

Additionally, there must be at least two input signals, as having fewer would result in a total input of zero. These requirements ensure that the input remains bounded.

By combining the concept of time differences with the requirement for bounded inputs, we can define the structure of Input Strategy 3:

Definition 3.5. *(General definition of input strategy 3). Using the explicit canonical form from Definition 3.1, the system can be described as follows:*

$$\begin{aligned} \begin{bmatrix} x_t(k) \\ x_q(k) \end{bmatrix} = \underbrace{\left[\begin{array}{c|c} A_t & \varepsilon \\ \hline \varepsilon & A_q \end{array}\right]}_{A} \otimes (\underbrace{\left[\begin{array}{c|c} B_t & \top \\ \hline \top & B_q \end{array}\right]}_{B} \otimes' (\underbrace{\left[\begin{array}{c|c} C_{11} & C_{12} \\ \hline C_{21} & C_{22} \end{array}\right]}_{C} \cdot \begin{bmatrix} x_t(k-1) \\ x_q(k-1) \end{bmatrix} \\ + \underbrace{\left[\begin{array}{c|c} E_{11} & E_{12} \\ \hline E_{21} & E_{22} \end{array}\right]}_{E} \cdot \begin{bmatrix} u_t(k) \\ u_q(k) \end{bmatrix})) \end{aligned} \tag{3-27}$$

Input signals are categorized into two types:

1. *Temporal signal inputs, which must satisfy the time difference requirements.*

2. *Quantity signal inputs, which do not impose time-invariance constraints as they are inherently bounded.*

To better illustrate this, consider the following example:

Example 3.3. *Numerical example for input strategy 3*

The implementation of this input strategy can be achieved either by designing specific input signals, which can be either time differences or quantity signals, or by utilizing state feedback. The concept of state feedback will be introduced in Section 5-1. For now, we will add a quantity signal input to the system depicted in Figure 3-3.

If the input signal $u_\tau(k)$ represents a quantity signal, the system equations can be written as:

$$\begin{aligned} x_2(k) &= x_1(k) + d_1 + u_\tau(k) \\ u_\tau(k) &= u_q(k) \end{aligned} \tag{3-28}$$

In cases where d_1 is not controllable, it becomes part of either the A or B matrix, resulting in the following MMPS system:

$$\begin{aligned} x_2(k) &= x_1(k) + d_1 + u_q(k) \\ x_2(k) &= \underbrace{\begin{bmatrix} 0 \end{bmatrix}}_{A} \otimes (\underbrace{\begin{bmatrix} d_1 \end{bmatrix}}_{B} \otimes' (\underbrace{\begin{bmatrix} 1 \end{bmatrix}}_{D} \cdot x_1(k) + \underbrace{\begin{bmatrix} 1 \end{bmatrix}}_{E} \cdot u_q(k))) \end{aligned} \tag{3-29}$$

Since this input consists solely of quantity signals, the system remains time-invariant.

3-3-4 Overview and comparison of all the input strategies

In this section, a clear overview will be provided for all possible input structures that MMPS systems can have. In the previous sections, three strategies are discussed that can be distinguished based on their mathematical differences:

- Input Strategy 1: Inputs are inside the original system equations
- Input Strategy 2: Inputs are added during the maximization or minimization step
- Input Strategy 3: Inputs are added during the scaling step

The schematic overviews of all three situations are visualized in Figure 3-4.

The three strategies can be characterized from left to right by system equations of the form:

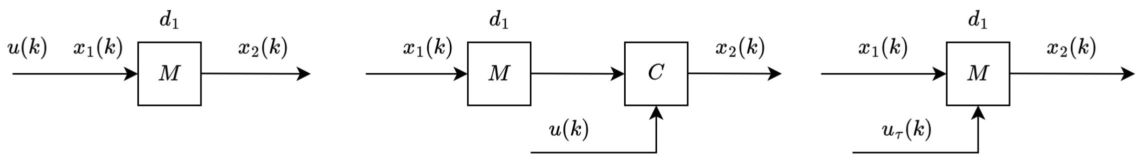


Figure 3-4: Schematic Overview All Input Strategies

Summary of input strategy 1

- System equation:

$$x(k) = \max(x(k-1), u(k))$$

- Matrix structure:

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + E \cdot u(k)))$$

- Input matrix (E) part of original system equations

Summary of input Strategy 2

- System equation:

$$x(k) = \max(x(k-1), u(k))$$

- Matrix structures:

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

$$x(k) = \begin{bmatrix} A & A_u \end{bmatrix} \otimes (\begin{bmatrix} B & \top \\ \top & B_I \end{bmatrix} \otimes' (\begin{bmatrix} C \\ 0 \end{bmatrix} \cdot x(k-1) + \begin{bmatrix} 0 \\ E \end{bmatrix} \cdot u(k))) \qquad (3\text{-}30)$$

- Input matrix (E) free to choose as long as time-invariance property preserved

Summary of input strategy 3

- System equation:

$$x(k) = x(k-1) + u(k)$$

- Matrix structure:

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + E \cdot u(k)))$$

- Input matrix (E) free to choose as long as time-invariance property preserved

Finally, for each input strategy it is discussed which type of signal can be combined with which type of state. The overview of possible combinations is summarized in Table 3-1, Table 3-2 and Table 3-3.

Table 3-1: Input-state combinations input strategy 1

	Temporal signal input	**Quantity signal input**
Temporal signal state	Possible	Not possible
Quantity signal state	Not possible	Possible

Table 3-2: Input-state combinations input strategy 2

	Temporal signal input	**Quantity signal input**
Temporal signal state	Possible	Not possible
Quantity signal state	Not possible	Possible

Table 3-3: Input-state combinations input strategy 3

	Temporal signal input	**Quantity signal input**
Temporal signal state	Possible*	Possible
Quantity signal state	Possible*	Possible

*Only possible if time difference is added to the system.

Chapter 4

Open-loop control of explicit MMPS systems

In the previous chapter, input strategies for implementing input signals in max-min-plus-scaling (MMPS) systems were introduced. When these inputs are used to influence the system, it is referred to as control. This chapter focuses solely on explicit MMPS systems, meaning that for the ABCDE canonical form, $D = 0$. The structure of this chapter is as follows: Section 4-1 introduces the concept of open-loop control and explores the relationship between classical control and control for MMPS systems. The subsequent Sections 4-2, 4-3 and 4-4 apply open-loop control to the input strategies presented in Section 3-3.

4-1 Introduction to open-loop control

In conventional plus-times algebra, time-invariant discrete-time systems can be written in the following form:

$$\begin{aligned} x(k+1) &= f(x(k), u(k)) \\ y(k) &= h(x(k), u(k)) \end{aligned} \tag{4-1}$$

The functions f and h can be either linear or nonlinear functions. The resulting set of system equations is also called a process. The process can be controlled by designing the input to reach specific values. When the input is a sequence of values that are not designed based on information of the process, it is called open-loop control. An open-loop system is schematically visualised in Figure 4-1.

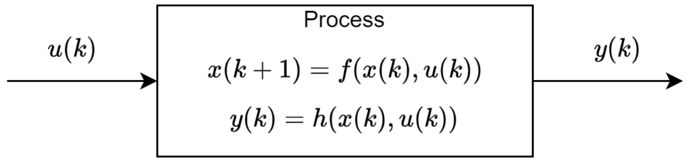


Figure 4-1: Schematic overview of an open-loop control system

This section will cover open-loop control techniques applied on explicit MMPS systems based on the input strategies described in Section 3-3. When looking at open-loop control of MMPS systems, we see that an explicit MMPS system can be described in a similar way compared to conventional algebra. In Figure 4-2 a schematic overview is given of the input-output relation in MMPS systems:

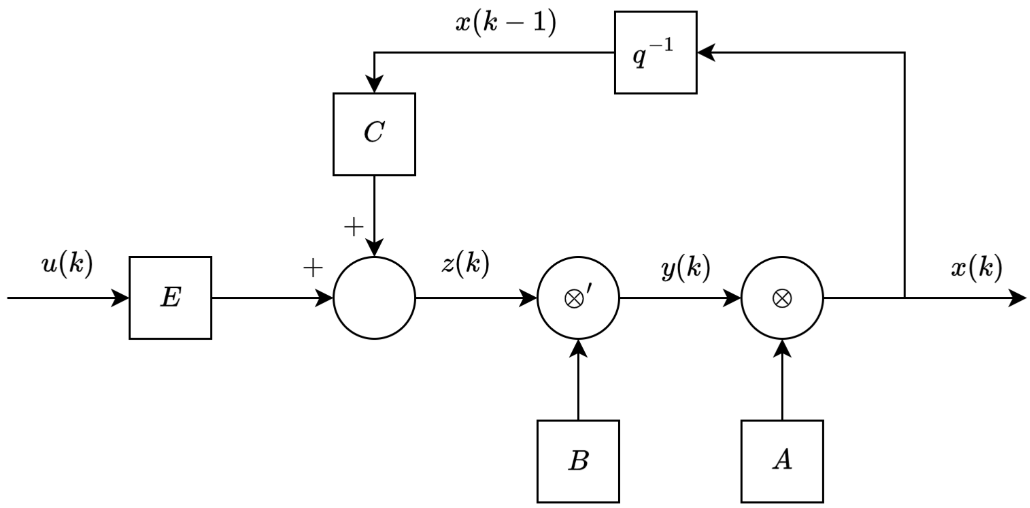


Figure 4-2: Schematic overview of an explicit MMPS system with input signals

where q^{-1} represents a time delay as $x(k-1)$ is used to generate the next event. Figure 4-2 consists of a similar input-output structure as Figure 4-1. Therefore, we can make the same overview of the process in an MMPS system. The schematic is given in Figure 4-3.

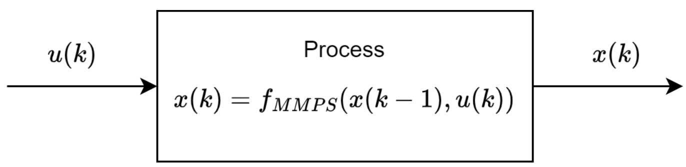


Figure 4-3: Schematic overview of an open-loop MMPS system

In the following sections, control is applied on MMPS systems with designs following the input structures of Section 3-3.

4-2 Open-loop control using input strategy 1

Input strategy 1 covers the situation where input signals are not additional but already part of the system equations of the MMPS systems, the system can not run autonomously. To evaluate the effect of applying control, we look at the same example as in Section 3-3-1.

Example 4.1. *Open-loop control of a production system using input strategy 1*

The production system is schematically visualized in Figure 4-4.

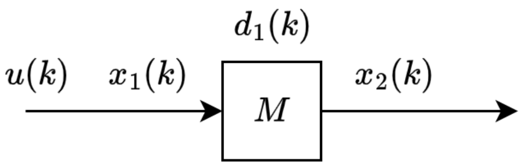


Figure 4-4: Schematic overview of a production system

In Figure 4-4, the output is defined as x_2. The machine (M) starts at x_1 and the machine will work for a state-dependent time ($d_1(k)$). The implicit MMPS system is defined by the following set of equations:

$$\begin{aligned} x_1(k) &= \max(x_2(k-1), u(k)) \\ x_2(k) &= x_1(k) + d_1(k) \\ d_1(k) &= \alpha \cdot (x_1(k) - x_1(k-1)) \end{aligned} \tag{4-2}$$

Substitution of the states provides the following explicit MMPS system:

$$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & 0 & 0 & 0 \end{bmatrix}}_{A} \otimes (\underbrace{\begin{bmatrix} 0 & \top & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top & \top \\ \top & \top & 0 & \top & \top & \top \\ \top & \top & \top & 0 & \top & \top \\ \top & \top & \top & \top & 0 & \top \\ \top & \top & \top & \top & \top & 0 \end{bmatrix}}_{B}$$

$$\otimes'(\underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \\ -\alpha & 1+\alpha \\ -\alpha & 1 \\ -\alpha & \alpha \\ -\alpha & 0 \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 1 \\ 0 \\ \alpha \\ 1 \\ 1+\alpha \end{bmatrix}}_{E} \cdot \begin{bmatrix} u(k) \end{bmatrix})) \tag{4-3}$$

The production unit is simulated with a scaling factor of $\alpha = 0.8$ and a dynamic input signal. The simulation can be found in Figure 4-5.

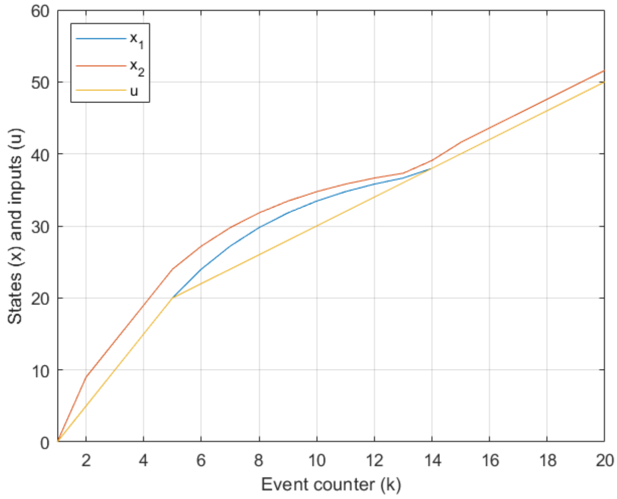


Figure 4-5: Open-loop control of a production system using input strategy 1

The system finds in both situations a stable equilibrium because the growth rate and difference between states is constant. When the input shifts, it takes some time to find its new equilibrium because of the scaling factor (α).

When designing this input structure on different systems, a similar approach can be used. The scaling factors of the inputs have to be placed inside the right entry of the input matrix and time-invariance must be ensured such that Eq. (3-13) holds.

4-3 Open-loop control using input strategy 2

The second control technique discussed involves input strategy 2, where an input is added during the maximization or minimization step. These inputs can either guide the system to specific values or impose boundaries. This section will focus on the latter.

By imposing boundaries on the MMPS system, it is able to withstand unwanted behaviour because the bounds control the system in a way that the states will remain within certain limits. As already described in Section 3-3-2, the system matrices need to be augmented to add inputs using this strategy. The extent to which the system matrices must be expanded depends on the type of boundary. In this section, we define two types of bounds which also can be combined, namely:

Imposing boundaries on the MMPS system can effectively eliminate unwanted behaviors by ensuring that the system states remain within defined limits. As discussed in Section 3-3-2, the system matrices must be augmented to incorporate this input strategy, with the extent of augmentation depending on the type of boundary applied. In this section, we define two primary types of boundaries, which can also be combined:

- Control input utilized as a maximal boundary
- Control input utilized as a minimal boundary

- A combination of control inputs serving as both maximal and minimal boundaries

The following subsections will explore each of these three possibilities in detail.

4-3-1 Control input utilized as a maximal boundary

Introducing an input as a maximal boundary ensures that, in the event of a sudden increase in the system's growth rate during a cycle, the system will be constrained by a maximum growth rate. This situation can arise, for example, in a production system facing a bottleneck, where imposing a maximum limit helps manage the process flow and maintain operations. The following definition is provided to place maximal bounds on MMPS systems:

Definition 4.1. *(Input signal utilized as a maximal boundary). The matrix structure from the competition operation of Definition 3.3 will be used:*

$$x(k) = A \otimes (\begin{bmatrix} B & B_u \end{bmatrix} \otimes' (\begin{bmatrix} C \\ 0 \end{bmatrix} \cdot x(k-1) + \begin{bmatrix} 0 \\ E \end{bmatrix} \cdot u(k))) \tag{4-4}$$

Two situations are defined, adding a bound to the temporal and to the quantity signals states. For temporal signal states, the input entries will be placed in the E_{11} part of the E matrix and B_u will have a non-infinity value in the row where the input signal must be applied.

Similarly, for maximal boundaries on quantity signals, the input entries are placed inside the E_{22} part of the E matrix and the B_u matrix has a non-infinity entry in the row where the input signal must be applied.

Generally, the non-infinity values of the B_u matrix are equal to 0 to have no further effect on the input signals.

When writing out the system equations of Eq. (4-4), we obtain the following set of equations:

$$\begin{aligned} x(k) &= A \otimes (\min(z_1(k) + B, z_2(k) + B_u)) \\ z_1(k) &= C \cdot x(k-1) \\ z_2(k) &= E \cdot u(k) \end{aligned} \tag{4-5}$$

The maximal bound is established because, when z_1 increases significantly, it causes z_2 to dominate the minimization. This effectively creates an upper bound on the system.

It is important to note that after the minimization step, the state will still be influenced by the values of A. Therefore, if the input acts as a strong boundary, it is necessary to subtract the maximum value of A since the maximum of each state is given by:

$$\max(x(k)) = E \cdot u(k) + \min(B_u) + \max(A) \tag{4-6}$$

In the next section, we will discuss the use of inputs to impose minimal bounds.

4-3-2 Control input utilized as a minimal boundary

In situations where the growth rate significantly decreases during an event cycle in the simulation, a minimal boundary can be implemented to bound the system. The minimal boundary is added during the maximization step and will therefore be placed in the A matrix instead of the B matrix. The general definition of adding minimal temporal signal bounds is as follows:

Definition 4.2. *(Input signal utilized as a minimal boundary). The matrix structure from the synchronization operation of Definition 3.3 will be used:*

$$x(k) = \begin{bmatrix} A & A_u \end{bmatrix} \otimes (\begin{bmatrix} B & \top \\ \top & B_I \end{bmatrix} \otimes' (\begin{bmatrix} C \\ 0 \end{bmatrix} x(k-1) + \begin{bmatrix} 0 \\ E \end{bmatrix} u(k))) \tag{4-7}$$

Two situations are defined, adding a minimal bound to the temporal and to the quantity signals states. For temporal signal states, the input entries will be placed in the E_{11} part of the E matrix. The identity matrix augmented inside B will consist of some zero's to pass on the input signals unaffected. Finally, A_u will have a non-infinity value in the row where the input must be applied.

Similarly, for minimal boundaries on quantity signals, the input entries are placed inside the E_{22} part of the E matrix, the B_I matrix will consist of zero's on the diagonal and the non-infinity entries of the A_u matrix need to be in the rows where the inputs must be applied.

Generally, the non-infinity values of the A_u matrix will be equal to 0 to have no further effects on the input signals.

When writing out the system equations of Eq. (4-7), we obtain the following set of equations:

$$\begin{aligned} x(k) &= \max(y_1(k) + A, y_2(k) + A_u) \\ y_1(k) &= C \cdot x(k-1) + B \\ y_2(k) &= E \cdot u(k) + B_I \end{aligned} \tag{4-8}$$

This configuration establishes a minimal boundary because, if y_1 becomes small, y_2 will dominate the maximization, effectively serving as a lower bound for the system. The minimal values in this system can then be expressed as:

$$\min(x(k)) = E \cdot u(k) + \max(A_u) + \min(B_I) \tag{4-9}$$

As long as the non-infinity entries in A_u and B_I are defined zero, the minimum will be unaffected. In the next subsection, we will combine both discussed bounds and evaluate an example.

4-3-3 Control inputs utilized as maximal and minimal boundaries

It is possible to incorporate both maximal and minimal boundaries within the same system for both temporal and quantity signal states. We combine Definition 4.1 and Definition 4.2 to get the following:

Definition 4.3. *(Input signals utilized as a maximal and minimal boundary). Several input signals are used to compute both maximal and minimal boundaries. The matrix structure for the competition and synchronization property are combined as follows:*

$$x(k) = \begin{bmatrix} A & A_u \end{bmatrix} \otimes (\begin{bmatrix} B & B_u \\ \top & B_I \end{bmatrix} \otimes' (\begin{bmatrix} C \\ 0 \end{bmatrix} \cdot x(k-1) + \begin{bmatrix} 0 \\ E \end{bmatrix} \cdot u(k))) \tag{4-10}$$

The A_u and B_u entries apply the bounds in the correct location. The augmentation of the remaining matrices is used to pass on information unaffected.

The structure of applying both types of bounds to the system will be illustrated in an example.

Example 4.2. *Open-loop control of a numerical example using input strategy 2*

We define the following MMPS system with two temporal and one quantity state:

$$\begin{bmatrix} x_{t1}(k) \\ x_{t2}(k) \\ x_q(k) \end{bmatrix} = \underbrace{\left[\begin{array}{cc|c} 4 & 2 & \varepsilon \\ 1 & 3 & \varepsilon \\ \hline \varepsilon & \varepsilon & 6 \end{array}\right]}_{A} \otimes (\underbrace{\left[\begin{array}{cc|c} 4 & 1 & \top \\ 2 & 4 & \top \\ \hline \top & \top & 5 \end{array}\right]}_{B} \otimes' \underbrace{\left[\begin{array}{cc|c} -0.75 & 1.75 & 4 \\ -0.75 & 1.75 & 0.2458 \\ \hline -4 & 4 & 0.1 \end{array}\right]}_{C} \cdot \begin{bmatrix} x_{t1}(k-1) \\ x_{t2}(k-1) \\ x_q(k-1) \end{bmatrix})) \tag{4-11}$$

The MMPS system will be augmented and two additional temporal and quantity input signals will be added. The additional matrices are defined as follows:

$$A_u = \left[\begin{array}{c|c} 0 & \varepsilon \\ 0 & \varepsilon \\ \hline \varepsilon & 0 \end{array}\right], B_u = \left[\begin{array}{cc|cc} 0 & \top & \top & \top \\ 0 & \top & \top & \top \\ \hline \top & \top & 0 & \top \end{array}\right], B_I = \left[\begin{array}{cc|cc} \top & 0 & \top & \top \\ \hline \top & \top & \top & 0 \end{array}\right], E = \left[\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right] \tag{4-12}$$

which can be used to model the MMPS system with maximal and minimal bounds on both its temporal and quantity signal states:

$$x(k) = \begin{bmatrix} A & A_u \end{bmatrix} \otimes (\begin{bmatrix} B & B_u \\ \top & B_I \end{bmatrix} \otimes' (\begin{bmatrix} C \\ 0 \end{bmatrix} \cdot x(k-1) + \begin{bmatrix} 0 \\ E \end{bmatrix} \cdot u(k))) \tag{4-13}$$

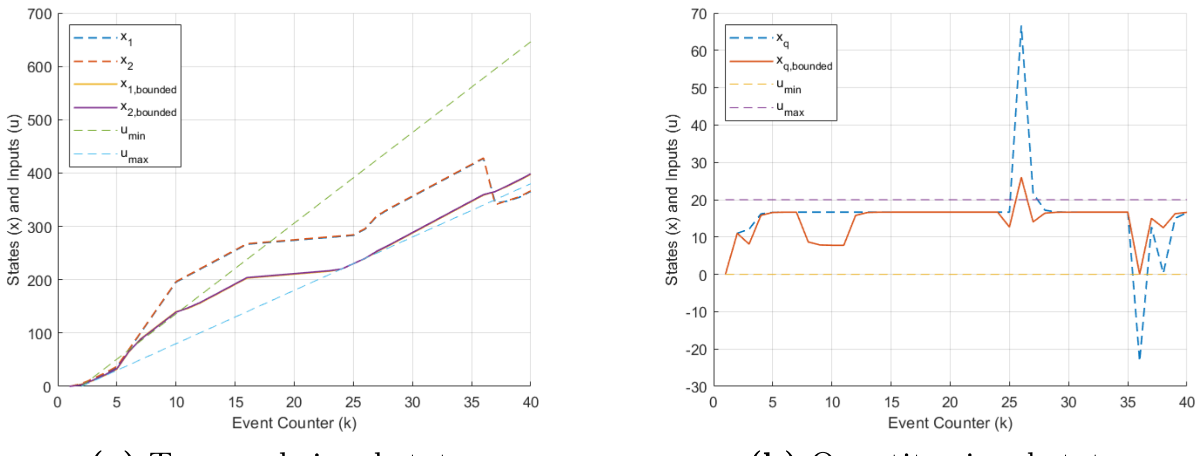
The following input signals are used as bounds for the system:

$$
\begin{aligned}
u_{t1}(k) &= 17 + u_{t1}(k-1) \\
u_{t2}(k) &= 10 + u_{t2}(k-1) \\
u_{q1}(k) &= 20 \\
u_{q2}(k) &= 0
\end{aligned}
\tag{4-14}
$$

The input signals consist of two temporal input signals that continuously increase and two quantity input signals that remain constant. To visualize the boundaries in a simulation, disturbance signals are introduced to the system. The $u_{t,dist}$ values are added to the temporal signal states for a period of k, while the $u_{q,dist}$ values are added to the quantity signal state for one event cycle. The disturbance signals are defined as follows:

$$
\begin{aligned}
u_{t,dist}(k) &= 20, \text{ for } k = \{5, \ldots, 9\} \\
u_{t,dist}(k) &= -10, \text{ for } k = \{16, \ldots, 22\} \\
u_{q,dist}(k) &= 50, \text{ for } k = 25 \\
u_{q,dist}(k) &= -40, \text{ for } k = 35
\end{aligned}
\tag{4-15}
$$

In Figure 4-6, two systems are simulated. The system from Eq. (4-11) and the disturbances from Eq. (4-15) as the dotted lines and the bounded system from Eq. (4-13) with input and disturbance signals from Eq. (4-14) and Eq. (4-15) in straight lines.



(a) Temporal signal states **(b)** Quantity signal states

Figure 4-6: Open-loop control of a numerical example using input strategy 2

Note that, as clarified in Section 4-3-1, the bounded system can exceed the upper bound. This occurs due to the addition of the entries from the A matrix after the minimization step. Furthermore, it can be observed that the system is effectively bounded because the unbounded systems significantly exceed the limits defined by $u_{\min}$ and $u_{\max}$.

4-4 Open-loop control using input strategy 3

The third input strategy adds input signals during the scaling step. The design of open-loop control using input strategy 3 will be evaluated based on an example.

Example 4.3. *Open-loop control of a production system using input strategy 3*

Suppose we have an MMPS system with the schematic overview of the production system visualized in Figure 4-7.

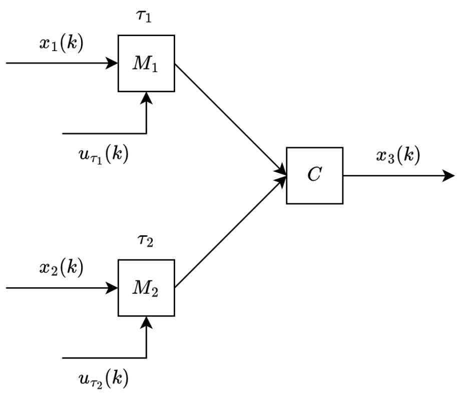


Figure 4-7: Schematic overview of a production system for input strategy 3

The system has two machines (M_1 and M_2) with its own respective production times (τ). Both machines can be influenced by an input signal ($u_\tau(k)$). The control device (C) uses the competition operation resulting in the system equation:

$$x_3(k) = \min(x_1(k) + \tau_1 + u_{\tau_1}(k), x_2(k) + \tau_2 + u_{\tau_2}(k)) \tag{4-16}$$

This system is implicit but can be made explicit by the following substitutions:

$$\begin{aligned} x_1(k) &= x_1(k-1) + 3 \\ x_2(k) &= x_2(k-1) + 4 \end{aligned} \tag{4-17}$$

The scalars are used to add a small delay in between starting times of the machines. Combine Eq. (4-16) and Eq. (4-17) to obtain an open-loop MMPS system:

$$\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & \varepsilon & \varepsilon \\ \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & 0 \end{bmatrix}}_{A} \otimes (\underbrace{\begin{bmatrix} 2 & \top & \top & \top \\ \top & 3 & \top & \top \\ \top & \top & 3+\tau_2 & 2+\tau_1 \end{bmatrix}}_{B} \otimes' \\ (\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}}_{E} \cdot \begin{bmatrix} u_{\tau_1}(k) \\ u_{\tau_2}(k) \end{bmatrix})) \quad (4\text{-}18)$$

The input solely consists of quantity signals, ensuring that time invariance is not violated. The production times are chosen as $\tau_1 = 3$ and $\tau_2 = 2$. The initial conditions are set to $x_0 = [10, 0]^T$, indicating that the first machine experiences a delay at the start. The simulation for varying input signals is visualized in Figure 4-8.

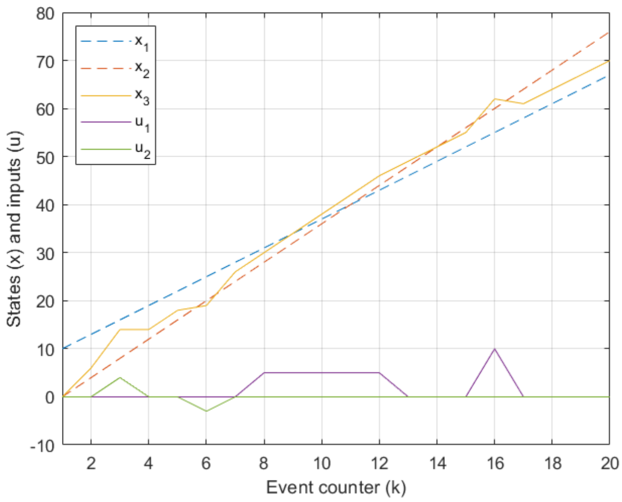


Figure 4-8: Open-loop control of a production system using input strategy 3

In the simulation of Figure 4-8, it can be observed that the input value influences the output of the system (x_3). Both positive and negative input changes affect the output. Additionally, it is possible to apply a constant input; in this case, the growth rate of the system will change for as long as the input is applied.

Chapter 5

Closed-loop control of explicit MMPS systems

In the previous chapter, open-loop control was introduced and applied to all the input strategies discussed in Section 3-3. Open-loop control refers to a scenario where the system is influenced by an input that does not depend on the system's own states or processes. While this method can be effective in some cases, it is limited by its inability to adapt to real-time changes within the system. It is possible to design an input strategy that utilizes real-time information from the system's processes, leading to the concept of closed-loop control, where the system's output is continuously monitored and used to adjust the input accordingly. This chapter focuses solely on explicit max-min-plus-scaling (MMPS) systems, meaning that for the ABCDE canonical form, $D = 0$. Section 5-1 provides a general introduction to closed-loop control. The subsequent Sections 5-2, 5-3, and 5-4 apply closed-loop control to the input strategies presented in Section 3-3. Finally, Section 5-5 demonstrates how closed-loop control can be used to stabilize an initially unstable MMPS system.

5-1 Introduction to closed-loop control

An open-loop control system consists of a process with an input stream $u(k)$ and an output stream $y(k)$. When applying closed-loop control, the input is replaced by a formula including information of the system. For this to work, information about the state values must be available. This can be obtained through sensors or other equipment. If state information is unavailable, an observer can estimate the states, but this requires the system to be observable. Observability theory is beyond the scope of this research; interested readers can refer to [23] and [24]. For this work, we assume full state information is available.

A substitution used for the input stream when applying closed-loop control are state feedback functions. A state feedback function, without an additional reference signal, can be described by the following form [24]:

$$u(k) = -K \cdot x(k) \tag{5-1}$$

This function is also referred to as a controller. The state feedback gain (K) can be designed to affect some states differently than others. A schematic overview of the new process and controller setup can be found in Figure 5-1.

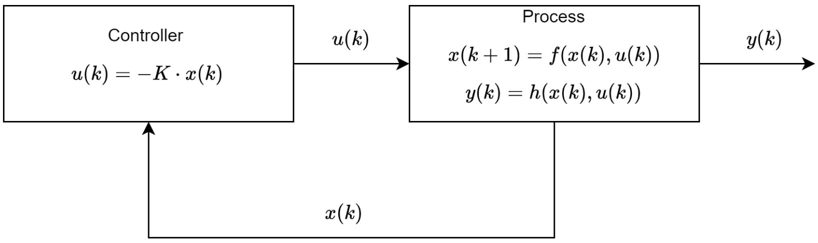


Figure 5-1: Schematic overview of a closed-loop control system

Compared to the schematic overview in Figure 4-1, the input is now replaced with a state feedback function, making the system closed-loop. Similarly, MMPS systems can also be converted to closed-loop configurations. A controller can be designed to utilize information from previous states; such a controller can generally be defined as follows:

$$u(k) = f_{\mathrm{u}}(x(k-1)) \tag{5-2}$$

Here, we denote f_u as a function containing the previous state. This function may also include a multiplication matrix (K) similar to Figure 5-1, or a reference value ($r(k)$). The closed-loop MMPS system is schematically represented in Figure 5-2.

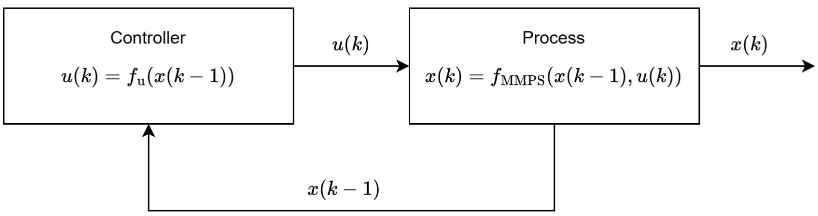


Figure 5-2: Schematic overview of a closed-loop explicit MMPS system

The function used for the controller depends on the specific goals of the system. However, we can define a very general form of the input function that encompasses all situations. Therefore, we use the following function for the input:

$$u(k) = G \otimes (H \otimes' (K \cdot x(k-1) + L \cdot u(k-1) + R \cdot r(k))) \tag{5-3}$$

where $G \in \mathcal{R}^{n\times m}$, $H \in \mathcal{R}^{m\times p}$, $K \in \mathcal{R}^{p\times n_x}$,$L \in \mathcal{R}^{p\times n_u}$,$R \in \mathcal{R}^{p\times r}$, $x \in \mathcal{R}^{n_x}$,$u \in \mathcal{R}^{n_u}$,$k \in \mathbb{Z}^+$ and $r \in \mathbb{R}^r$. The vector $r(k)$ represents a reference signal and K is a state feedback matrix. Using this equation, we can define a general closed-loop MMPS system that is implicit in $u(k)$.

Definition 5.1. *(General closed-loop structure of an explicit MMPS system). The following matrix structure is defined for a closed-loop MMPS system that is implicit in $u(k)$:*

$$\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} A & \varepsilon \\ \varepsilon & G \end{bmatrix} \otimes \left(\begin{bmatrix} B & \top \\ \top & H \end{bmatrix} \otimes' \left(\begin{bmatrix} C & E & 0 & 0 \\ K & 0 & L & R \end{bmatrix} \cdot \begin{bmatrix} x(k-1) \\ u(k) \\ u(k-1) \\ r(k) \end{bmatrix} \right)\right) \tag{5-4}$$

In the next sections, closed-loop control is applied on the three defined input strategies discussed in Section 3-3.

5-2 Closed-loop control using input strategy 1

For evaluating closed-loop control of an MMPS system using input strategy 1, we are again using the production system of Example 4.1 and the schematic overview of Figure 4-4.

Example 5.1. *Closed-loop control of a production system using input strategy 1*

Recall from Eq. (4-3) that the production system is modeled as an MMPS system in the following manner:

$$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & 0 & 0 & 0 \end{bmatrix}}_{A} \otimes \left(\underbrace{\begin{bmatrix} 0 & \top & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top & \top \\ \top & \top & 0 & \top & \top & \top \\ \top & \top & \top & 0 & \top & \top \\ \top & \top & \top & \top & 0 & \top \\ \top & \top & \top & \top & \top & 0 \end{bmatrix}}_{B} \otimes' \left(\underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \\ -\alpha & 1+\alpha \\ -\alpha & 1 \\ -\alpha & \alpha \\ -\alpha & 0 \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 1 \\ 0 \\ \alpha \\ 1 \\ 1+\alpha \end{bmatrix}}_{E} \cdot [u(k)] \right)\right) \tag{5-5}$$

The production unit is dependent on a growing input signal. Therefore, from Eq. (5-3), we are going to design an input signal that uses both a feedback matrix (K) and a reference signal ($r(k)$). The previous input will not be used inside this controller such that $L = 0$. Combining Definition 5.1 and the input signal from Eq. (5-3) with $G = H = 0$. A closed-loop system using input strategy 1 is obtained:

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + E \cdot K \cdot x(k-1) + E \cdot R \cdot r(k))) \tag{5-6}$$

For the production system of Eq. (5-5), this results in the following closed-loop system:

$$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = A \otimes (B \otimes' (\underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \\ -\alpha & 1+\alpha \\ -\alpha & 1 \\ -\alpha & \alpha \\ -\alpha & 0 \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ K \\ 0 \\ K \cdot \alpha \\ K \\ K \cdot (1+\alpha) \end{bmatrix}}_{E \cdot K} \cdot x_1(k-1) + \underbrace{\begin{bmatrix} 0 \\ R \\ 0 \\ R \cdot \alpha \\ R \\ R \cdot (1+\alpha) \end{bmatrix}}_{E \cdot R} \cdot r(k))) \tag{5-7}$$

The system matrices A and B are unaffected by the substitution. The system should preserve the time-invariance property, since x_1 and x_2 are both temporal signals, all rows of the combined matrix $C + E \cdot K$ should equal 1. This is the case for $K = 1$. For this system, we design the reference signal as bounded variables and therefore it will not influence time-invariance. The system is simulated with a varying reference value $r(k)$ and can be found in Figure 5-3.

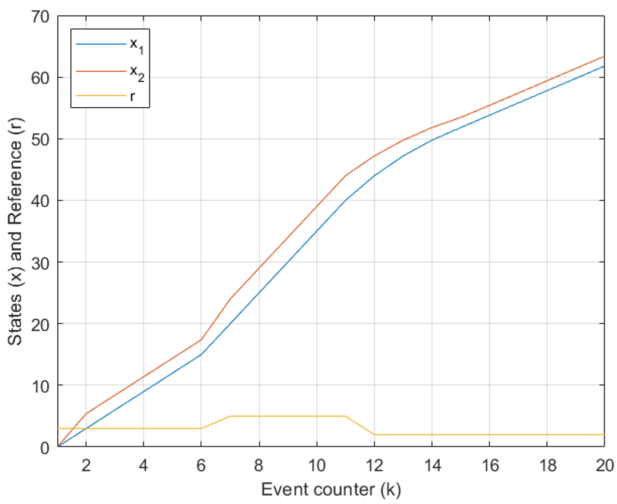


Figure 5-3: Closed-loop control of a production system using input strategy 1

The example of Figure 5-3 covers a situation where the original input was already in the original system equations. Therefore, the system corresponds to input strategy 1. Closed-loop control is applied by using a state feedback controller, the system remains stable and the reference value can be used to speed up or slow down the system. In the next section, closed-loop control will be evaluated on input strategy 2.

5-3 Closed-loop control using input strategy 2

To apply closed-loop control using input strategy 2, a controller similar to 4-3 can be employed. In Section 3-3-2, it was explained that input strategy 2 can be used either to steer the system toward specific values or to impose bounds on it. This section will focus on the latter case, where the closed-loop MMPS systems become bounded due to the inputs signals.

By combining the matrix structure from input strategy 2 with the closed-loop framework from Definition 5.1, we can establish a similar framework for input strategy 2 that creates state-dependent boundaries. These boundaries are advantageous because they do not need to be predefined, allowing the system to adapt more effectively to changes during operation. For instance, the growth rate of an MMPS system may fluctuate over several events before returning to its original rate. If boundaries are set for only one growth rate, this could lead to issues. State-dependent boundaries resolve this by keeping the system's variations between events within limits. A general structure for implementing state-dependent boundaries is defined as follows:

Definition 5.2. *(State-dependent boundaries for MMPS systems). Combining the structure of adding boundaries to MMPS systems of Definition 4.3 with the structure of the general closed-loop MMPS system of Definition 5.1, we obtain a general structure that creates state-dependent boundaries on MMPS systems:*

$$\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} A & A_u \\ \varepsilon & 0 \end{bmatrix} \otimes (\begin{bmatrix} B & B_u & \top \\ \top & B_I & \top \\ \top & \top & 0 \end{bmatrix} \otimes' (\begin{bmatrix} C & 0 & 0 & 0 \\ 0 & E & 0 & 0 \\ K & 0 & L & R \end{bmatrix} \cdot \begin{bmatrix} x(k-1) \\ u(k) \\ u(k-1) \\ r(k) \end{bmatrix})) \quad (5\text{-}8)$$

Substitution and defining that $L = 0$ makes the system explicit in $u(k)$. The final structure to create state-dependent boundaries is given by:

$$x(k) = \begin{bmatrix} A & A_u \end{bmatrix} \otimes (\begin{bmatrix} B & B_u \\ \top & B_I \end{bmatrix} \otimes' (\begin{bmatrix} C & 0 \\ E \cdot K & E \cdot R \end{bmatrix} \cdot \begin{bmatrix} x(k-1) \\ r(k) \end{bmatrix})) \quad (5\text{-}9)$$

The reference values $r(k)$ can be used to define the bounds and need to be bounded values itself to preserve time-invariance.

To further illustrate the introduction of state-dependent bounds, Example 4.2 will be re-used to evaluate closed-loop control using input strategy 2.

Example 5.2. *Closed-loop control of a numerical example using input strategy 2*

Recall from Eq. (4-11) that the MMPS system used in Example 4.2 can be described as follows:

$$\begin{bmatrix} x_{t1}(k) \\ x_{t2}(k) \\ x_q(k) \end{bmatrix} = \underbrace{\left[\begin{array}{cc|c} 4 & 2 & \varepsilon \\ 1 & 3 & \varepsilon \\ \hline \varepsilon & \varepsilon & 6 \end{array}\right]}_{A} \otimes (\underbrace{\left[\begin{array}{cc|c} 4 & 1 & \top \\ 2 & 4 & \top \\ \hline \top & \top & 5 \end{array}\right]}_{B} \otimes' \underbrace{\left[\begin{array}{cc|c} -0.75 & 1.75 & 4 \\ -0.75 & 1.75 & 0.2458 \\ \hline -4 & 4 & 0.1 \end{array}\right]}_{C} \cdot \begin{bmatrix} x_{t1}(k-1) \\ x_{t2}(k-1) \\ x_q(k-1) \end{bmatrix})) \tag{5-10}$$

State-dependent bounds will be imposed on both temporal signal states. The quantity signal states will be bounded in the same way as in Example 4.2. The following set of matrices are used to design the closed-loop system of Eq. (5-9):

$$A_u = \left[\begin{array}{cc|c} 0 & \varepsilon & \varepsilon \\ \varepsilon & 0 & \varepsilon \\ \hline \varepsilon & \varepsilon & 0 \end{array}\right], B_u = \left[\begin{array}{cccc|cc} 0 & \top & \top & \top & \top & \top \\ \top & 0 & \top & \top & \top & \top \\ \hline \top & \top & \top & \top & 0 & \top \end{array}\right], B_I = \left[\begin{array}{cccc|cc} \top & \top & 0 & \top & \top & \top \\ \top & \top & \top & 0 & \top & \top \\ \hline \top & \top & \top & \top & \top & 0 \end{array}\right],$$
$$E \cdot K = \left[\begin{array}{cc|c} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}\right], E \cdot R = \left[\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right] \tag{5-11}$$

The state-dependent bounds on the temporal signal states and the bounds on the quantity signal states are all placed on the system by defining bounded values for the reference signal:

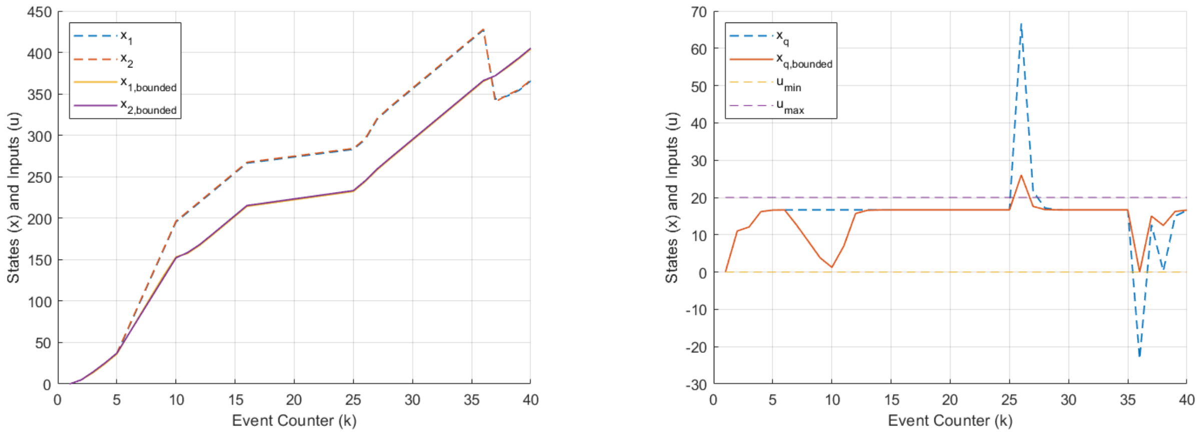
$$r(k) = \begin{bmatrix} \lambda_{t,up} & \lambda_{t,down} & \lambda_{q,up} & \lambda_{q,down} \end{bmatrix}^T \tag{5-12}$$

with $\lambda_{t,up} = 20, \lambda_{t,down} = 2, \lambda_{q,up} = 20, \lambda_{q,down} = 0$.

Through the closed-loop structure, the maximal difference per event cycle (k) in the temporal signal stats will be bounded between 2 and 20. The quantity signal states are bounded between 0 and 20. Again, note that the absolute upper bounds are higher because of the additional entries in A. Similar to Section 4-3, the system will be influenced by some disturbance signals to illustrate the effect of the bounds on the system. The disturbance signals are defined as follows:

$$\begin{aligned} u_{t,dist}(k) &= 20, \text{ for } k = \{5, \dots, 9\} \\ u_{t,dist}(k) &= -10, \text{ for } k = \{16, \dots, 22\} \\ u_{q,dist}(k) &= 50, \text{ for } k = 25 \\ u_{q,dist}(k) &= -40, \text{ for } k = 35 \end{aligned} \tag{5-13}$$

In Figure 5-4, the dotted lines illustrate the disturbed unbounded system, while the straight lines represent the disturbed bounded system. The bounded system does not exceed its bounded growth rate while the disturbed system crosses them. The quantity signal state also remains within bounds for the bounded system. Therefore, it can be concluded that the framework with adding state-dependent boundaries is applied correctly.



(a) Temporal signal states **(b)** Quantity signal states

Figure 5-4: Closed-loop control of a numerical example using input strategy 2

5-4 Closed-loop control using input strategy 3

For applying closed-loop control using input strategy 3, we have to design a controller that uses time differences in the feedback scheme. Therefore, we have to carefully consider the requirements set for time-differences defined in Definition 3.4. A general structure for closed-loop control using input strategy 3 can be defined as follows:

Definition 5.3. *(General structure of closed-loop control using input strategy 3). The general closed-loop structure for an explicit MMPS system defined in Definition 5.1 will be used:*

$$\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} A & \varepsilon \\ \varepsilon & 0 \end{bmatrix} \otimes (\begin{bmatrix} B & \top \\ \top & 0 \end{bmatrix} \otimes' (\begin{bmatrix} C & E & 0 & 0 \\ K & 0 & L & R \end{bmatrix} \cdot \begin{bmatrix} x(k-1) \\ u(k) \\ u(k-1) \\ r(k) \end{bmatrix})) \tag{5-14}$$

For ensuring time differences are added to the system, the state feedback matrix must be designed such that:

$$\sum_{j=1:n_t} [K]_{ij} = 0, \, \forall i \tag{5-15}$$

Often, using strategy 3, the reference matrix R will be zero. Furthermore, the previous input is not part of the new input such that: $L = 0$. After substitution of the input, making the *system explicit in $u(k)$, the general structure for closed-loop control using input strategy 3 is defined as:*

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + E \cdot K \cdot x(k-1))) \tag{5-16}$$

The application of closed-loop control using input strategy 3 will be further illustrated using an example.

Example 5.3. *Closed-loop control of a production system using input strategy 3*

The production system that will be evaluated in this example is visualized in Figure 5-5.

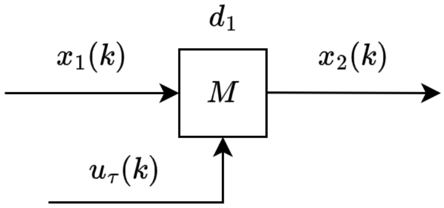


Figure 5-5: Schematic overview of a production system for input strategy 3

The output ($x_2(k)$) of the system is defined by the following system equation:

$$x_2(k) = x_1(k) + d_1 + u_\tau(k) \tag{5-17}$$

The following closed-loop controller is designed that uses state feedback and preserves time-invariance following Eq. (5-15):

$$u_\tau(k) = \underbrace{\begin{bmatrix} -\alpha & \alpha \end{bmatrix}}_{K} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \end{bmatrix} \tag{5-18}$$

The controller adjusts the time interval between the previous start and finish times of the machine. The system equation in Eq. (5-17) can be made explicit with the substitution $x_1(k) = x_1(k-1)+3$, indicating that the machine will start three time units after its previous starting time. The controller in Eq. (5-18) is applied to the output. By modeling the system equations within the closed-loop structure defined in Definition 5.3, we obtain:

$$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \underbrace{\begin{bmatrix} 5 & \varepsilon \\ \varepsilon & 5+\tau \end{bmatrix}}_{A} \otimes (\underbrace{\begin{bmatrix} 0 & \top \\ \top & 0 \end{bmatrix}}_{B} \otimes' (\underbrace{\begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \end{bmatrix}$$
$$+ \underbrace{\begin{bmatrix} 0 & 0 \\ -\alpha & \alpha \end{bmatrix}}_{E \cdot K} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \end{bmatrix})) \tag{5-19}$$

The system will be simulated with a production time of $\tau = 3$ and two different values for α. The simulation is visualized in Figure 5-6. It can be observed that the controller adjusts the difference between x_1 and x_2, yet the systems remain stable in both cases. The state x_1 is identical for both systems, as it is independent of α. For more complex systems, closed-loop control using input strategy 3 proves to be highly beneficial. This will be verified in the next chapter, where closed-loop systems will be optimized to achieve specific objectives. In the following section, we first examine the possibility of stabilizing an unstable MMPS system.

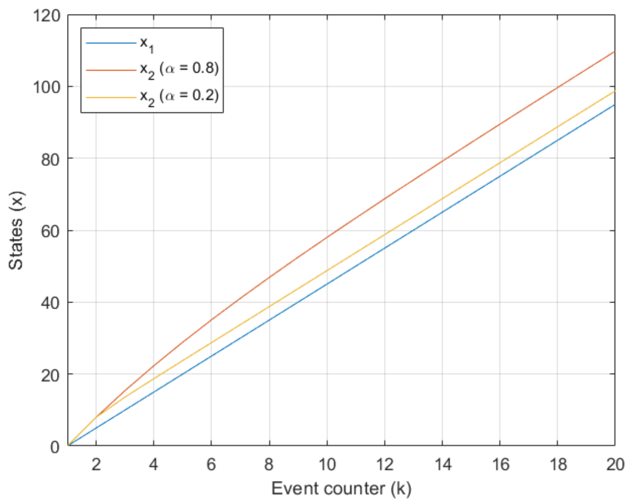


Figure 5-6: Simulation closed-loop control of MMPS system using input strategy 3

5-5 Stabilize an unstable explicit MMPS system

Recall, from Section 2-4-3, the definition of stability in an MMPS system. Definition 2.27, describes the formulation of a normalized MMPS system that can be written as:

$$\begin{aligned} \tilde{x}_\theta(k) &= M_\theta \cdot \tilde{x}_\theta(k-1) \\ M_\theta &= G_{A_\theta} \cdot G_{B_\theta} \cdot C \end{aligned} \tag{5-20}$$

Following [21], an MMPS system is stable if all eigenvalues of M_θ are less than or equal to one and all Jordan blocks corresponding to magnitude one are 1×1. In the following example, we will show that the autonomous system is unstable and that the controller has a stabilizing effect.

Example 5.4. *Stabilize an unstable production system using closed-loop control*

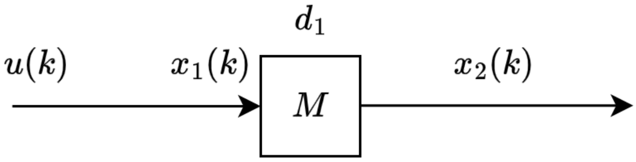


Figure 5-7: Schematic overview production system

The schematic representation of the production systems is visualized in Figure 5-7. The system consists of a machine (M), two temporal signal states ($x(k)$), an input signal ($u(k)$) that feeds the system for the k-th time and an event-depending processing time ($d(k)$). For the first state, we use the sequential processing operation such that:

$$x_1(k) = \max(x_2(k-1), u(k)) \tag{5-21}$$

To make the system autonomous, we assume that the input is fed every τ time units.

$$u(k) = x_1(k-1) + \tau \tag{5-22}$$

The event-depending processing time ($d(k)$) increases linearly with the difference between the present starting time ($x(k)$) and the previous starting time ($x_1(k-1)$), with a minimum value $d_{\min}$ and a maximum value $d_{\max}$:

$$d(k) = \min(\max(\alpha(x_1(k) - x_1(k-1)), d_{\min}), d_{\max}) \tag{5-23}$$

where $\alpha \in \mathbb{R}$. Finally, the finishing time is defined using processing:

$$x_2(k) = x_1(k) + d(k) \tag{5-24}$$

This is an MMPS system that can be modeled in explicit disjunctive form. The substitutions required to reach this form are straightforward and are assumed negligible to document in detail. The explicit MMPS system can be written as:

$$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \underbrace{\begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 & 0 \\ 0 & 0 & 0 & 0 & d_{\min} & \tau + d_{\min} \end{bmatrix}}_{A} \otimes (\underbrace{\begin{bmatrix} 0 & d_{\max} & \top & \top \\ \top & \alpha \cdot \tau & \top & \top \\ \top & \top & \tau & \tau + d_{\max} \\ \top & \top & \top & \alpha \cdot \tau \\ \top & 0 & \top & \top \\ \top & \top & \top & 0 \end{bmatrix}}_{B}$$

$$\otimes' (\underbrace{\begin{bmatrix} -\alpha & 1+\alpha \\ 0 & 1 \\ 1-\alpha & \alpha \\ 1 & 0 \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \end{bmatrix})) \tag{5-25}$$

when the system is modeled with the following parameters: $\alpha = 1$, $\tau = 1.5$, $d_{\max} = 10$, and $d_{\min} = 2$, two unstable growth rates emerge: $\theta_1 = 2$ and $\theta_2 = 10$ using the algorithm from [9]. Several footprint matrices associated with both growth rates contain multiple zeros in their normalized forms, causing these matrices to have more than one entry equal to "1." According to [19], this suggests that the footprint matrices may define two distinct linearized regions within the system. The specific footprint matrices corresponding to each growth rate are as follows:

$$\text{for } \theta_1 : G_{A_1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, G_{A_2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, G_B = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5-26}$$

$$\text{for } \theta_2 : G_A = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, G_{B_1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, G_{B_2} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$G_{B_3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, G_{B_4} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5-27}$$

Each growth rate has multiple footprint matrices, this means that the system consists of multiple regions with each a different growth rate. The initial state of the system decides which growth rate the system follows. This unstable behavior can also be recognized in the linearized system of Eq. (5-20). Using the algorithm of [9], the multiplicative eigenvalues of both growth rates are computed: $\mu_{\theta_1} = \mu_{\theta_2} = [0.382, 2.618]^T$. Based on the definition of stability from [21], the autonomous system has two unstable growth rates.

The unstable behavior can be verified with a simulation of the autonomous system using different initial state values. In Figure 5-8, the autonomous system is simulated for three different initial states, each resulting in different growth rates.

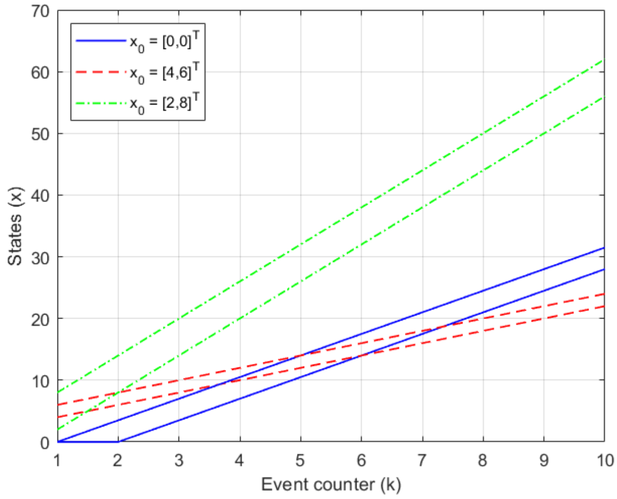


Figure 5-8: Simulation autonomous production system from multiple initial conditions

To stabilize the system, the next step is to create a closed-loop system. We are going to implement the closed-loop structure of input strategy 3 from Definition 5.3, where:

$$\begin{aligned} x(k) &= A \otimes (B \otimes' (C \cdot x(k-1) + E \cdot u(k))) \\ u(k) &= K \cdot x(k-1) \end{aligned} \tag{5-28}$$

We must define matrices E and K such that they have a stabilizing purpose. The goal is to achieve the same growth rate for different initial conditions. The following controller will be used:

$$u(k) = \beta \cdot (x_2(k-1) - x_1(k-1)) \tag{5-29}$$

This controller scales the difference between the previous finishing and starting times of the system. The input is added to the previous finishing time, resulting in the following matrices:

$$E = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, K = \begin{bmatrix} -\beta & \beta \end{bmatrix} \tag{5-30}$$

A scaling factor: $\beta = 0.75$ results in a closed-loop system with a single growth rate. Namely, $\lambda = 3.5$. The new footprint matrices are as follows:

$$G_{A_\lambda} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, G_{B_\lambda} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5-31}$$

By implementing this controller, the multiplicative eigenvalues of M have become: $\mu = [0, 1]^T$ making the growth rate $\lambda = 3.5$ a stable one.

To verify the stability of the closed-loop system, it is simulated again with the same initial conditions as shown in Figure 5-8. The results of the simulation can be found in Figure 5-9, where all the different initial conditions lead to the system following the stable growth rate $\lambda = 3.5$.

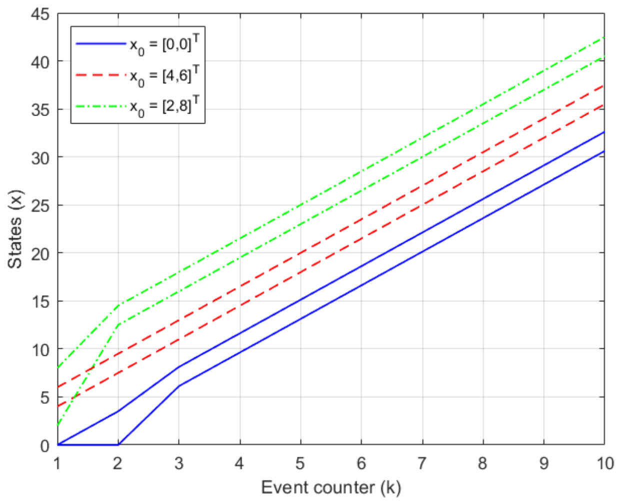


Figure 5-9: Simulation closed-loop production system from mulitple initial conditions

The controller used in this example is designed through human tuning. In the next chapter, an algorithm will be developed to optimize the feedback matrices to achieve several objectives.

Chapter 6

Optimization-based closed-loop control of explicit MMPS systems

In the previous chapter, the effects of applying closed-loop control across all input strategies is discussed. This chapter extends closed-loop control of explicit max-min-plus-scaling (MMPS) systems by defining feedback matrices using an optimization-based method. Section 6-1 introduces optimization-based control, detailing the closed-loop form that will be utilized, along with the optimization goals and the optimization algorithm. Additionally, the concept of regions in MMPS systems is introduced. Section 6-2 defines a method to identify the boundaries between regions, which can be used to either constrain the system within a region or force it to switch between regions. Section 6-3 applies optimization-based control with the objective of minimizing the system's closed-loop growth rate. Section 6-4 optimizes closed-loop control to minimize the difference between the output and a reference signal. Finally, Section 6-5 presents an optimization of the closed-loop structure that forces the system to switch between stable growth rates.

6-1 Introduction to optimization-based closed-loop control

In this section, we will introduce several key concepts that will be utilized in the subsequent sections, where an optimization-based control approach is applied to MMPS systems to achieve various objectives.

For this chapter, we will employ the general closed-loop structure outlined in Definition 5.1, setting the matrices G and H to zero. This results in the following matrix structure:

$$\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} A & \varepsilon \\ \varepsilon & 0 \end{bmatrix} \otimes (\begin{bmatrix} B & \top \\ \top & 0 \end{bmatrix} \otimes' (\begin{bmatrix} C & E & 0 & 0 \\ K & 0 & L & R \end{bmatrix} \cdot \begin{bmatrix} x(k-1) \\ u(k) \\ u(k-1) \\ r(k) \end{bmatrix})) \tag{6-1}$$

The input functions of all algorithms defined in this chapter will be independent of their previous inputs, resulting in $L = 0$. Furthermore, no reference signal will be incorporated into the closed-loop form, so $R = 0$. After substitution, the explicit closed-loop system utilized in this chapter is expressed as:

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + E \cdot K \cdot x(k-1))) \tag{6-2}$$

The time-invariance property must be carefully considered, ensuring that the following condition holds at all times:

$$\begin{aligned} \sum_{j=1:n_t} [C_{11}]_{ij} + [E_{11}]_{ij} \cdot [K]_{ij} = 1, \forall i \\ \sum_{j=1:n_q} [C_{21}]_{ij} + [E_{21}]_{ij} \cdot [K]_{ij} = 0, \forall i \end{aligned} \tag{6-3}$$

Next, we will discuss the concept of a region within an MMPS. In Section 2-4-3, we introduced the idea of a topical system. Topical systems are characterized as monotonic, time-invariant, and non-expansive. However, it is important to note that not all MMPS systems qualify as topical; non-topical systems may exhibit multiple stable or unstable growth rates.

When analyzing optimal closed-loop control, it is essential to examine the region in which a specific growth rate applies. This region within an MMPS system can be defined as follows:

Definition 6.1. *(Region of an MMPS system). An autonomous MMPS system in a stable configuration exhibits a constant growth rate. This growth rate is determined by the multiplication in the scaling step, combined with the addition of constant factors from the maximization and minimization steps. The state evolution for a stable autonomous MMPS system can be expressed as follows:*

$$\begin{aligned} x(k) &= x_{ss} + C \cdot x(k-1) \\ x_{ss} &= [A]_{ij} + [B]_{ij} \end{aligned} \tag{6-4}$$

where x_{ss} is the steady-state column vector containing the constant values derived from the A and B matrices. The region of an MMPS system is defined as the multidimensional plane within which the system evolves according to the above equations, provided it is not influenced by control mechanisms that would cause it to exit this region.

When optimizing the closed-loop structure of an MMPS system for specific objectives, our goal is to ensure that the system either remains within its initial region or is forced to switch to a different region. The conditions for achieving this objective can be determined by examining the boundaries of the regions in MMPS systems. These boundaries can be established through the definition of inequalities, which will be further elaborated in the next section.

Additionally, we define three optimization goals for closed-loop MMPS systems. These goals will be stated initially and then explained in greater detail:

1. Minimize the closed-loop growth rate.
2. Follow a reference signal.
3. Change between stable growth rates.

First, we discuss minimizing the closed-loop growth rate. In physical applications of MMPS systems, such as manufacturing, it may be desirable for a machine to operate as efficiently as possible. This can be achieved by minimizing the system's growth rate.

Second, we consider tracking a reference signal. For instance, in a manufacturing system, information about product demand may be available. When demand is low, production time should decrease accordingly; conversely, if demand increases, the system should respond to speed up the production process to minimize the difference between the number of products produced and the demand.

Third, we address the goal of switching between stable growth rates. As mentioned earlier, a system may exhibit multiple growth rates. In some cases, it is advantageous to design a closed-loop controller that enables the system to switch between these rates.

Finally, to solve the optimization problems, we will utilize the `fmincon` function in MATLAB, which is designed to find minima for constrained problems. Due to the non-linearity in the state evolution, we require a solver that accommodates non-linear functions. The `fmincon` function will be employed in combination with the Sequential Quadratic Programming (SQP) algorithm, which effectively handles inequality constraints. SQP solves a series of quadratic sub-problems that approximate the original non-linear problem [25]. This method is iterative and optimizes the variables in each iteration.

A downside of SQP is its sensitivity to the initial guess, which may lead to local optima rather than global optima. However, given the scale of the examples used in this research, this should not pose a significant issue.

6-2 Find boundaries of MMPS systems

As previously defined in Definition 6.1, autonomous MMPS systems in a stable configuration exhibit a constant state evolution. The equations that characterize this state evolution are referred to as the dominant equations of that region, defined as follows:

Definition 6.2. *(Dominant Equations). Suppose we have an MMPS system of the form:*

$$x(k) = A \otimes (B \otimes (C \cdot x(k-1))) \tag{6-5}$$

Under stable conditions, where the growth rate is constant, the state will evolve based on specific entries from the A and B matrices. These entries, combined with the scaled states, form the dominant equations. As long as the system remains in the same stable configuration, these equations will dictate the state evolution. They can be expressed as:

$$x = a_{i,j} + b_{i,j} + z \tag{6-6}$$

where $x = [x_1, x_2, \dots, x_n]^T$ *is the state vector and* $z = [z_1, z_2, \dots, z_n]^T$ *is a vector of the scaled states, with* $z = C \cdot x$*. The values* $a_{i,j}$ *and* $b_{i,j}$ *represent the dominant entries in the A and B matrices.*

The set of dominant equations corresponds to a multidimensional plane, referred to as the region defined in Definition 6.1. This region can be either stable or unstable. A stable region implies that when the states of the MMPS system lie within it, they will eventually converge to the set of dominant equations, evolving according to the associated growth rate. In stable conditions, this region is invariant, meaning the system will not exit it without external influence. Conversely, if the states of an MMPS system are situated in an unstable region, they will converge, within a finite number of steps, to a stable region.

It is essential for the system to exhibit at least two growth rates when determining the bounds of a region. To identify the set of dominant equations corresponding to a specific growth rate, we utilize structure matrices. To find the structure matrices, we employ the normalized system matrices $\tilde{A}$ and $\tilde{B}$ as defined in [9] and in Equation 5-20. A structure matrix enables us to efficiently find the dominant values of the A and B matrices and is defined as follows:

Definition 6.3. *(Structure matrices). The structure matrices filter out the non-zero values of the normalized system matrices such that:*

$$[F_{A_\theta}]_{ij} = \begin{cases} 0 & \text{if } \left[\tilde{A}_\theta\right]_{ij} = 0 \\ \varepsilon & \text{if } \left[\tilde{A}_\theta\right]_{ij} \neq 0 \end{cases}, \quad [F_{B_\theta}]_{jl} = \begin{cases} 0 & \text{if } \left[\tilde{B}_\theta\right]_{jl} = 0 \\ \top & \text{if } \left[\tilde{B}_\theta\right]_{jl} \neq 0 \end{cases} \tag{6-7}$$

The structure matrices have entries equal to zero at the positions of the dominant values. This characteristic is advantageous because, when combined with the original system matrices, it yields a matrix containing only the essential data needed to derive the dominant equations. This resulting matrix is referred to as the dominant entry matrix and is defined as follows:

Definition 6.4. *(Dominant entry matrices). The structure matrix F for each stable growth rate is added to the original system matrices via matrix addition:*

$$\begin{aligned} Z_{A_\theta} &= F_{A_\theta} + A \\ Z_{B_\theta} &= F_{B_\theta} + B \end{aligned} \tag{6-8}$$

This operation is possible because the structure matrix always has the same dimensions as its corresponding system matrix. The resulting matrix Z is called the dominant entry matrix.

When deriving the boundaries, we encounter two possible objectives: either to remain within the initial region or to transition to a different region. The following two definitions establish the inequalities required for each of these situations.

Definition 6.5. *(Inequalities for staying within the initial region). Consider an MMPS system with an initial growth rate λ_1. To ensure the system remains within this initial region, we define λ_1 as the desired growth rate and denote the alternative growth rates by λ_N, where $\lambda_N = [\lambda_2, \dots, \lambda_n]$. The partial solutions corresponding to each growth rate are given by:*

$$\begin{aligned} x_\theta &= Z_{A_\theta} \otimes (Z_{B_\theta} \otimes' z) \\ y_\theta &= Z_{B_\theta} \otimes' z \end{aligned} \tag{6-9}$$

where $\theta = [\lambda_1, \lambda_N]$. The system equations that must hold for remaining within the initial region can be expressed as:

$$\begin{aligned} \min(y_{\lambda_1}, y_{\lambda_N}) &= y_{\lambda_1} \\ \max(x_{\lambda_1}, x_{\lambda_N}) &= x_{\lambda_1} \end{aligned} \tag{6-10}$$

This leads to the following set of inequalities for ensuring that the system stays within the initial region:

$$
\begin{aligned}
Z_{B_{\lambda_N}} \otimes' z &> Z_{B_{\lambda_1}} \otimes' z \\
Z_{A_{\lambda_1}} \otimes (Z_{B_{\lambda_1}} \otimes' z) &> Z_{A_{\lambda_N}} \otimes (Z_{B_{\lambda_1}} \otimes' z)
\end{aligned} \tag{6-11}
$$

Definition 6.6. *(Inequalities for switching between regions). Consider an MMPS system where we wish to transition from an initial growth rate, λ_1, to a new growth rate, λ_2. The partial solutions associated with each region's growth rate are defined as:*

$$
\begin{aligned}
x_\theta &= Z_{A_\theta} \otimes (Z_{B_\theta} \otimes' z) \\
y_\theta &= Z_{B_\theta} \otimes' z
\end{aligned} \tag{6-12}
$$

where $\theta = [\lambda_1, \lambda_2]$. To successfully switch from the region associated with λ_1 to that of λ_2, the following system equalities must apply:

$$
\begin{aligned}
&\min(y_{\lambda_1}, y_{\lambda_2}) = y_{\lambda_2}, \\
&\max(x_{\lambda_1}, x_{\lambda_2}) = x_{\lambda_2}.
\end{aligned} \tag{6-13}
$$

Consequently, this yields the set of inequalities necessary for the region change:

$$
\begin{aligned}
Z_{B_{\lambda_1}} \otimes' z &> Z_{B_{\lambda_2}} \otimes' z \\
Z_{A_{\lambda_2}} \otimes (Z_{B_{\lambda_2}} \otimes' z) &> Z_{A_{\lambda_1}} \otimes (Z_{B_{\lambda_2}} \otimes' z)
\end{aligned} \tag{6-14}
$$

To further illustrate the application of the algorithm, consider the following example, where we derive the inequalities required to force an MMPS system to switch regions.

Example 6.1. *Derive inequalities to switch region for an MMPS system*

The following explicit MMPS system with three temporal signal states will be used:

$$
\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 3 & 4 \\ 4 & 5 & 1 \\ 4 & 5 & 2 \end{bmatrix}}_{A} \otimes (\underbrace{\begin{bmatrix} 1 & 3 & 4 \\ 4 & 5 & 1 \\ 4 & 5 & 2 \end{bmatrix}}_{B} \otimes' (\underbrace{\begin{bmatrix} -0.4 & 1.4 & 0 \\ 0.8 & 0.8 & -0.6 \\ -0.4 & -0.4 & 1.8 \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \end{bmatrix})) \tag{6-15}
$$

The system's growth rates, derived using the algorithm from [9], are: $\lambda = \{5.4, 4.8, 6\}$. Of these, only two are stable: $\lambda_1 = 5.4$ and $\lambda_2 = 4.8$. The corresponding structure matrices for these stable growth rates are as follows:

$$
\begin{aligned}
F_{A,\lambda_1} &= \begin{bmatrix} 0 & \varepsilon & \varepsilon \\ 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 \end{bmatrix}, F_{B,\lambda_1} = \begin{bmatrix} \top & \top & 0 \\ \top & \top & 0 \\ \top & \top & 0 \end{bmatrix} \\
F_{A,\lambda_2} &= \begin{bmatrix} \varepsilon & \varepsilon & 0 \\ 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 \end{bmatrix}, F_{B,\lambda_2} = \begin{bmatrix} 0 & \top & \top \\ 0 & \top & \top \\ 0 & \top & \top \end{bmatrix}
\end{aligned} \tag{6-16}
$$

The dominant entry matrices can be computed by using matrix addition: $Z_A = F_A + A$ and $Z_B = F_B + B$. This results in the matrices:

$$Z_{A,\lambda_1} = \begin{bmatrix} 5 & \varepsilon & \varepsilon \\ 5 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 5 \end{bmatrix}, Z_{B,\lambda_1} = \begin{bmatrix} \top & \top & 4 \\ \top & \top & 1 \\ \top & \top & 2 \end{bmatrix}$$
$$Z_{A,\lambda_2} = \begin{bmatrix} \varepsilon & \varepsilon & 5 \\ 5 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 5 \end{bmatrix}, Z_{B,\lambda_2} = \begin{bmatrix} 1 & \top & \top \\ 4 & \top & \top \\ 4 & \top & \top \end{bmatrix} \tag{6-17}$$

From this set of matrices, we can produce the set of inequalities using Definition 6.6. After simulating the system, it is shown that λ_2 is the initial growth rate. Therefore, we want to transition from λ_2 to λ_1. This results in the following set of inequalities:

$$\begin{aligned} z_1 + 1 &> z_3 + 4 \\ z_1 + 4 &> z_3 + 1 \\ z_1 + 4 &> z_3 + 2 \\ z_3 + 9 &> z_3 + 7 \\ z_3 + 9 &> z_3 + 4 \\ z_3 + 7 &> z_3 + 4 \end{aligned} \tag{6-18}$$

Upon examination, it becomes evident that certain inequalities are redundant or overlap with others. Consequently, this set of inequalities can be reduced to:

$$z_1 + 1 > z_3 + 4 \tag{6-19}$$

When this inequality is satisfied, all other inequalities will also hold.

6-3 Control objective: Minimize growth rate

In this section, an algorithm will be introduced such that the growth rate of a closed-loop MMPS system can be minimized. The explicit closed-loop system of Eq. (6-2) will be used.

When the MMPS system has multiple growth rates, it is essential to define inequalities that ensure the system remains within its initial region. These inequalities are derived using the algorithm outlined in Definition 6.5. They will be expressed as a set of linear equations of the following form:

$$A_z \cdot z(k) + b > 0 \tag{6-20}$$

where:

$$z(k) = C \cdot x(k-1) + E \cdot K \cdot x(k-1) \tag{6-21}$$

In this context, A_z contains values equal to -1 or 1 corresponding to the sign of the entries in $z = [z_1, \ldots, z_n]^T$, with n representing the number of z values, and b is a vector of remaining scalars.

The objective of the optimization problem is to minimize the growth rate by optimizing the state feedback matrix K. The objective function is defined as the sum of the absolute state differences between events:

$$\|x(k) - x(k-1)\|_1 \tag{6-22}$$

Several equality and inequality constraints must be combined into a comprehensive constraint function. Firstly, the system should follow the state evolution as described by:

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + E \cdot K \cdot x(k-1))) \tag{6-23}$$

Secondly, if there exists more than one stable eigenvalue, it is essential to ensure that the system remains within its initial stable region. This is represented by the constraint:

$$A_z \cdot (C + E \cdot K) \cdot x(k) + b > 0 \tag{6-24}$$

Furthermore, to maintain time-invariance, the following equality constraint is added:

$$\sum_j [K]_{ij} = 0 \quad \forall i \tag{6-25}$$

In some cases, it may be necessary to establish bounds on the values of the feedback matrix. This can be accomplished with the following two constraints:

$$\begin{aligned} [K]_{ij} &\leq M \quad \forall i, j \\ [K]_{ij} &\geq -M \quad \forall i, j \end{aligned} \tag{6-26}$$

Lastly, we require that the system is always growing and we need to ensure that all states and inputs remain positive. The complete optimization problem is summarized in the following definition:

Definition 6.7. *(Algorithm for minimizing the growth rate of an MMPS system). The optimization problem of minimizing the growth rate can be achieved by applying the following algorithm:*

$$\begin{aligned}
&\min_{K} \|x(k) - x(k-1)\|_1 \\
\textit{subject to:}\quad & x(k) = A \otimes (B \otimes' ((C + E \cdot K) \cdot x(k-1))) \\
& A_z \cdot (C + E \cdot K) \cdot x(k) + b > 0 \\
& \sum_j [K]_{ij} = 0 \quad \forall i \\
& [K]_{ij} \leq M \quad \forall i, j \\
& [K]_{ij} \geq -M \quad \forall i, j \\
& x(k) - x(k-1) \geq 0 \\
& u(k) \geq 0 \\
& x(k) \geq 0
\end{aligned} \tag{6-27}$$

The optimization problem computes the optimal state feedback matrix K to minimize the closed-loop growth rate.

The algorithm will be validated through an example. The example demonstrates minimizing the closed-loop growth rate of the initially unstable production system and discussed in Section 5-5.

Example 6.2. *Minimize the closed-loop growth rate of an unstable production system*

In Section 5-5, an unstable MMPS system was stabilized through the design of a state feedback controller. However, this controller was tuned manually, resulting in a suboptimal solution. In this example, the optimal feedback matrix is computed that minimizes the closed-loop growth rate while also stabilizing the system.

The same system matrices and parameters are employed as in Eq. (5-25). Recall that the autonomous system configured using these parameters has two unstable growth rates: $\theta_1 = 2$ and $\theta_2 = 10$. The optimal feedback matrix that minimizes the closed-loop growth rate of the production system is derived using the algorithm outlined in Definition 6.7.

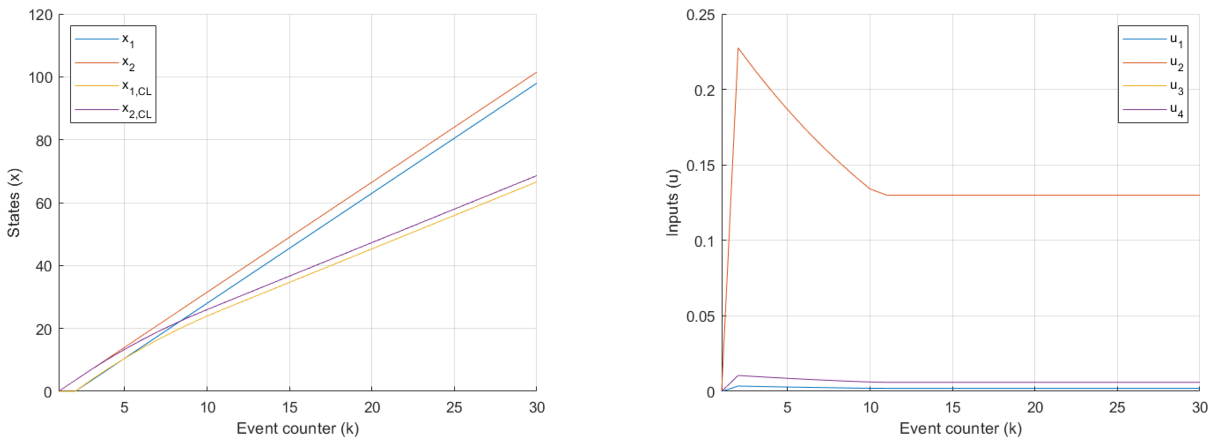
The bounds on the feedback matrix, denoted as M, are set sufficiently large to ensure they do not influence the optimization process. We will use the same set of initial state values as those presented in Section 5-5. The combinations of initial conditions with their corresponding growth rates are as follows:

$$\begin{aligned}
x_{0_1} &= \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \text{ with: } \lambda_1 = 3.5 \\
x_{0_2} &= \begin{bmatrix} 2 \\ 8 \end{bmatrix}, \text{ with: } \lambda_2 = 6 \\
x_{0_3} &= \begin{bmatrix} 4 \\ 6 \end{bmatrix}, \text{ with: } \lambda_3 = 2.5
\end{aligned} \tag{6-28}$$

By optimizing the closed-loop structures using the algorithm, the following feedback matrices are obtained, with the matrix E at each time step being a diagonal matrix of size 4×4:

$$K_1 = \begin{bmatrix} -0.001 & 0.001 \\ -0.065 & 0.065 \\ -0.003 & 0.003 \\ -0.003 & 0.003 \end{bmatrix}, K_2 = \begin{bmatrix} 0 & 0 \\ -0.15 & 0.15 \\ -0.096 & 0.096 \\ -0.004 & 0.004 \end{bmatrix}, K_3 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -0.004 & 0.004 \\ -0.002 & 0.002 \end{bmatrix} \tag{6-29}$$

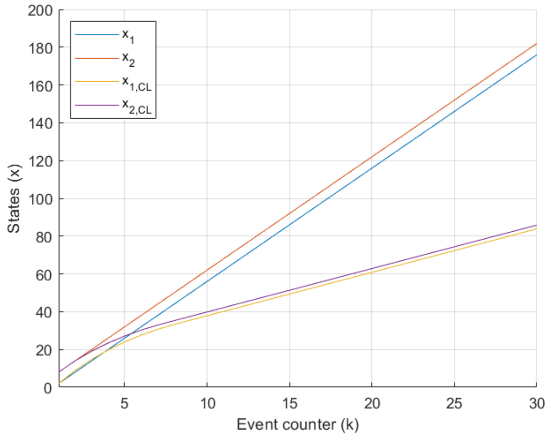
The state evolution is structured in a closed-loop configuration as defined in 6-23. For each closed-loop system, the corresponding autonomous system is also simulated to evaluate the differences in behavior. The simulations for the three systems are presented in Figure 6-1, Figure 6-2, and Figure 6-3.



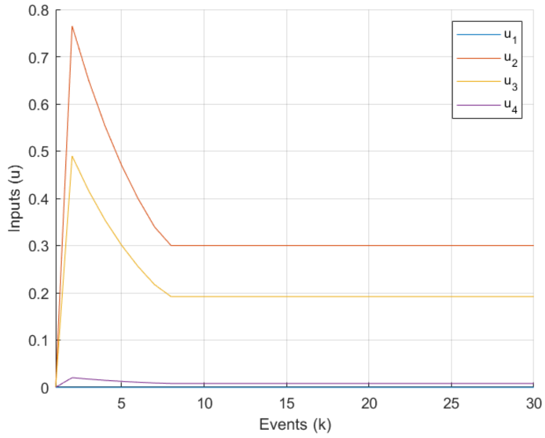
(a) States: $x(k)$ **(b)** Inputs: $u(k)$

Figure 6-1: Simulation autonomous and closed-loop production system from $x_0 = [0, 0]^T$

In Figure 6-1, it can be observed that the closed-loop system exhibits a lower growth rate than the open-loop system when starting from the initial condition $x_0 = 0$. Figure 6-2 shows a similar scenario, but the difference in growth rates is even more pronounced, with larger input signals and a greater number of input signals utilized compared to the first situation. Finally, Figure 6-3 indicates that the closed-loop growth rate remains the same as the initial growth rate, with only small positive inputs that do not significantly affect the growth rate. This suggests that the initial growth rate for $x_0 = [4, 6]^T$ is already the minimal growth rate.

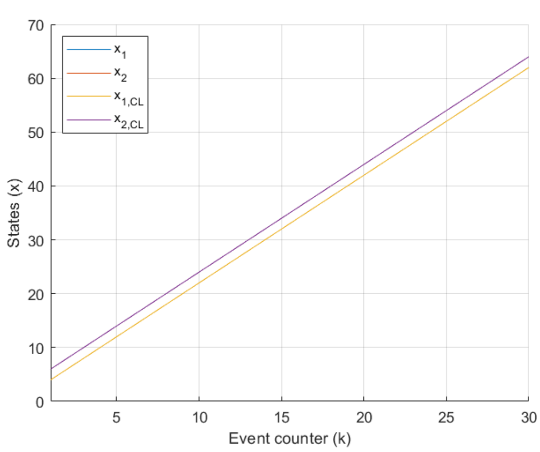


(a) States: $x(k)$

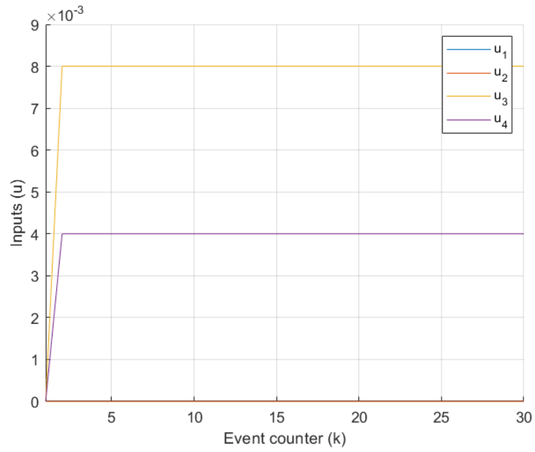


(b) Inputs: $u(k)$

Figure 6-2: Simulation autonomous and closed-loop production system from $x_0 = [2, 8]^T$



(a) States: $x(k)$



(b) Inputs: $u(k)$

Figure 6-3: Simulation autonomous and closed-loop production system from $x_0 = [4, 6]^T$

6-4 Control objective: Follow reference signal

In this section, we introduce an algorithm aimed at minimizing the distance between the system's states and an external reference signal. This algorithm is similar to the one used for minimizing the closed-loop growth rate, as defined in Definition 6.7. For the state evolution, we will employ the explicit closed-loop system format given in Eq. (6-2).

The primary distinction between this approach and the previous one lies in the definition of the objective function. In this case, the goal is to minimize the absolute difference between the system states and the external reference signal $r(k)$. We define the objective function as follows:

$$\|x(k) - r(k)\|_1 \tag{6-30}$$

The constraints will be identical to those outlined in Section 6-3. It is important to note that this optimization occurs offline, enabling the computation of the optimal state feedback matrix K based on prior knowledge of the reference signal. While there are methods to calculate the optimal input at each step of the simulation, this topic falls outside the scope of the current research. The algorithm for optimizing the feedback matrix to minimize the difference between the states and the reference signal is defined as follows:

Definition 6.8. *(Algorithm for minimizing the difference between states and reference signal). The optimization problem of minimizing the difference between states and reference signals can be achieved by applying the following algorithm:*

$$\begin{aligned}
&\min_K \|x(k) - r(k)\|_1 \\
\textit{subject to:}\quad & x(k) = A \otimes (B \otimes' ((C + E \cdot K) \cdot x(k-1))) \\
& A_z \cdot (C + E \cdot K) \cdot x(k) + b > 0 \\
& \sum_j [K]_{ij} = 0 \quad \forall i \\
& [K]_{ij} \leq M \quad \forall i, j \\
& [K]_{ij} \geq -M \quad \forall i, j \\
& x(k) - x(k-1) \geq 0 \\
& u(k) \geq 0 \\
& x(k) \geq 0
\end{aligned} \tag{6-31}$$

The optimization problem computes the optimal state feedback matrix K to minimize the difference between the states and an external reference signal $r(k)$.

The algorithm will be validated through an example.

Example 6.3. *Minimize the difference between the states and an external reference signal for a numerical example of an MMPS system*

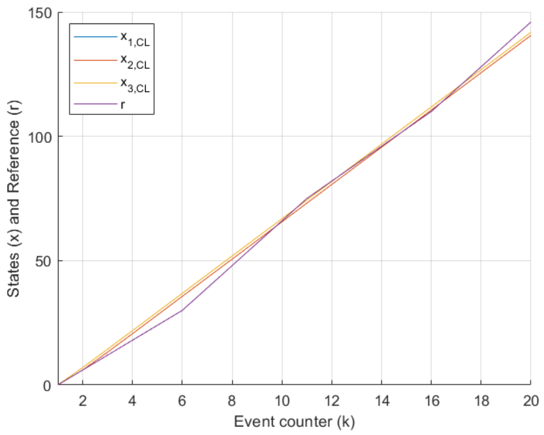
Suppose we have the following explicit MMPS system:

$$\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} = \underbrace{\begin{bmatrix} 5 & 3 & 5 \\ 5 & 1 & 1 \\ 1 & 4 & 5 \end{bmatrix}}_{A} \otimes (\underbrace{\begin{bmatrix} 1 & 3 & 4 \\ 4 & 5 & 1 \\ 4 & 5 & 2 \end{bmatrix}}_{B} \otimes' (\underbrace{\begin{bmatrix} 0.2 & 0.8 & 0 \\ 0.5 & 0.4 & 0.1 \\ 0.1 & 0.4 & 0.5 \end{bmatrix}}_{C} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \end{bmatrix})) \tag{6-32}$$

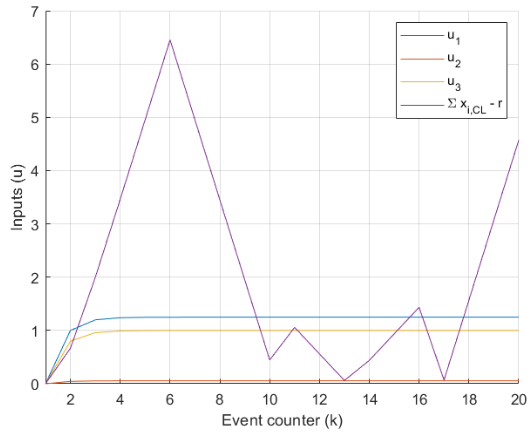
The autonomous MMPS system has a growth rate equal to $\lambda = 6.3$. As the system is topical and possesses only one growth rate, we do not need to worry about the system leaving its initial region. During optimization, we assume that the state feedback matrix K is unbounded. The following optimal closed-loop structure is obtained using the algorithm of Definition 6.8:

$$E = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, K = \begin{bmatrix} 0.21 & -0.43 & 0.21 \\ 0.26 & -0.52 & 0.26 \\ 0.32 & -0.64 & 0.31 \end{bmatrix} \tag{6-33}$$

A simulation of the reference signal and closed-loop state evolution can be found in Figure 6-4. The reference signal varies over time, and consequently, the optimized feedback matrix is designed to keep the difference between the states and the reference minimal. The closed-loop growth rate is equal to $\lambda = 7.5$. In Figure 6-4(b), it is evident that while the average difference between states and the reference varies, it generally remains low.



(b) $x_{i,CL}(k)$ and $r(k)$



(b) $u(k)$ and $\sum x_{i,CL}(k) - r(k)$

Figure 6-4: Simulation reference signal and closed-loop system

6-5 Control objective: Switch stable growth rates

In this section, the goal shifts from maintaining the same region as the initial autonomous system to actively steering the system into a different region through closed-loop control. To achieve this, we will utilize the explicit closed-loop system format provided in Eq. (6-2) for the state evolution.

The set of inequalities that triggers the system to switch its growth rates is defined according to Definition 6.6. These inequalities are formulated as linear equations in the following manner:

$$A_z \cdot z(k) > b \tag{6-34}$$

where:

$$z(k) = (C + E \cdot K) \cdot x(k-1) \tag{6-35}$$

Using the set of inequalities, we can determine Δz for each inequality, representing the minimum distance required between z-values to trigger the system to switch. Subsequently, we can calculate Δu using the following formula:

$$\begin{aligned} \Delta z &= \Delta z_{initial} + \Delta u \\ \Delta z &> b \end{aligned} \tag{6-36}$$

The initial value ($\Delta z_{\text{initial}}$) will be obtained from the autonomous system. $\Delta z_{\text{initial}}$ represents the initial difference between z values so that we do not introduce more difference than necessary. Δu is the difference we need to add to realize the switch in growth rate, implemented using the control structure such that:

$$\Delta u = E \cdot u(k) \tag{6-37}$$

We are going to use state feedback for the controller, where $u(k)$ is a function of the previous states: $u(k) = K \cdot x(k-1)$.

The last step is determining where to place the inputs in the E matrix. It is important to note that, after the addition of the input, the z-values first undergo minimization. Because of this, the following problem could occur:

$$\begin{gathered} \Delta z = z_1 - z_3 > 5 \\ \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} z_3 + 6 \\ z_2 \\ z_3 \end{bmatrix} \text{ but } z_2 < z_1, z_3 \\ \min(b_{i,j} + z_1, b_{i,j} + z_2, b_{i,j} + z_3) = b_{i,j} + z_2 \end{gathered} \tag{6-38}$$

This means that while the inequalities hold, the value z_2, which was previously irrelevant, has now become dominant. To resolve this issue, we check if the inequality ensures that the correct state becomes dominant. The positive z-term will be equal to 1 in the temporary input matrix E_{temp}. We then check if the following equality applies:

$$y = B \otimes' (z_{temp} + \Delta z \cdot E_{temp}) = Z_B \otimes' z_{temp} = y_{check} \tag{6-39}$$

Here, z_{temp} is a vector containing a solution for a random event in the simulation of the autonomous system, and Z_B is the dominant entry matrix corresponding to the desired growth rate. If the equality holds, it ensures that no other state will become dominant when using the input matrix E_{temp}. Otherwise, we need to adjust the input matrix.

One way to address this issue is by adding the input to all states except the one that needs to become dominant. However, this approach requires more controllable states, and in physical situations, not all states may be controllable. Therefore, in some cases, it might be impossible to switch between stable growth rates.

An example will be used to validate the approach to switch between the stable growth rates of an MMPS system.

Example 6.4. *Switch between stable growth rates of a numerical example of an MMPS system*

The same explicit MMPS system with three temporal signal states will be used as in Example 6.1. The MMPS system is defined in Eq. (6-15).

In Example 6.1, it is derived that only one inequality is able to force the system to switch growth rates. The only required inequality is:

$$z_1 + 1 > z_3 + 4 \tag{6-40}$$

Rewriting the inequality in the linear format of 6-34 results in the following:

$$\underbrace{\begin{bmatrix} 1 & 0 & -1 \end{bmatrix}}_{A_z} \cdot \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} > 3 = b \tag{6-41}$$

Therefore, according to Eq. (6-36), Δz should be larger than 3. The next step is to find Δu using Eq. (6-36). The autonomous MMPS system is simulated to find the initial differences between z_1 and z_3, which is equal to: $\Delta z_{\text{initial},13} = -5.4$. Resulting in the inequality for Δu:

$$\begin{aligned} &\Delta u > b - \Delta z_{initial,13} \\ &\Delta u > 8.4 \end{aligned} \tag{6-42}$$

For the derivation of the E matrix, we have to check if the additional input will not affect the minimal solution in an undesired way, as previously discussed. The equality of Eq. (6-39) is verified. Unfortunately, we find that $y \neq y_{\text{check}}$, which means we have to extend the input matrix to:

$$E = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \tag{6-43}$$

For the addition of the input signals that will execute the switch in growth rate, the difference between x_1 and x_2 will be used. This result in the following state feedback structure:

$$u(k) = \underbrace{\begin{bmatrix} \beta & -\beta & 0 \end{bmatrix}}_{K} \cdot \begin{bmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \end{bmatrix} \tag{6-44}$$

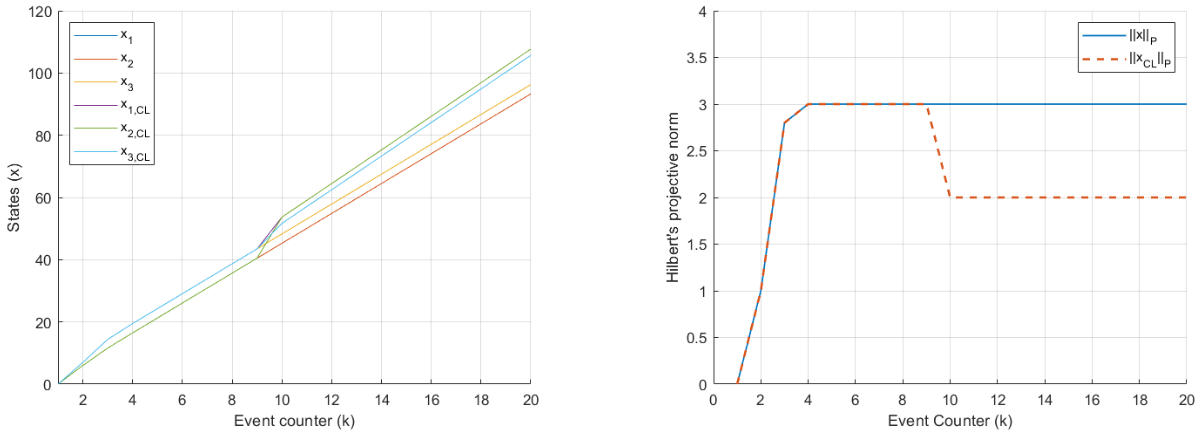
Following Eq. (6-42), the input added through the controller should be at least equal to 8.4. Therefore, the scaling factor β can be calculated by:

$$\beta = \frac{\Delta u}{\Delta x_{\text{initial}}} + \text{tolerance} \tag{6-45}$$

where $\Delta x_{\text{initial}}$ is the stationary difference in states that are multiplied by β, and a small tolerance is added to ensure that the value will be larger than the exact boundary. In this example, β is equal to 2.801, providing the state feedback matrix:

$$K = \begin{bmatrix} 2.801 & -2.801 & 0 \end{bmatrix} \tag{6-46}$$

In Figure 6-5(a), we visualize the simulation results for both the autonomous and closed-loop systems. At event $k = 10$, control is introduced, forcing the system to transition from its initial growth rate of $\lambda_2 = 4.8$ to a new growth rate of $\lambda_1 = 5.4$. Figure 6-5(b) presents the Hilbert projective norm for both systems, demonstrating that the system reaches a new equilibrium at exactly $k = 10$.



(a) States: $x(k)$ **(b)** Projective norm: $||x||_{\mathbb{P}}$

Figure 6-5: Simulation autonomous and closed-loop system

Chapter 7

Model predictive control (MPC) for max-plus linear (MPL) systems

In this chapter, we evaluate max-plus linear (MPL) systems instead of max-min-plus-scaling (MMPS) systems. MPL systems represent a simplified form of MMPS systems because they are linear in max-plus algebra. The state evolution of MPL systems is defined in Definition 2.4. Section 7-1 provides an introduction to the optimization-based control method that will be used to control MPL systems: model predictive control (MPC). Section 7-2 defines the MPC algorithm applied to control MPL systems. The application of MPC for MPL systems is further discussed through an example.

7-1 Introduction to model predictive control (MPC)

In this section, a short introduction to MPC for time-driven systems is provided. In the next section, we are going to make the relation between time-driven and MPL systems because there are many similarities. This section is based on [22] where more extensive research on MPC can be found.

Consider a state space system in conventional algebra of the form:

$$\begin{aligned} x(k) &= A \cdot x(k-1) + B \cdot u(k) \\ y(k) &= C \cdot x(k) \end{aligned} \tag{7-1}$$

where the state is denoted by $x \in \mathbb{R}^n$, the input $u \in \mathbb{R}^{n_u}$ and the output $y \in \mathbb{R}^{n_y}$ with n_u the amount of inputs and n_y the amount of outputs.

Generally, an MPC controller predicts the optimal control input (u^*) at each time step (t) over a finite horizon $N_p : u_t, u_{t+1}, \ldots, u_{t+N_p-1}$. The first input of the optimal input sequence will be applied to the system: u_t^*, afterwards the horizon shifts one time step, such that the new optimal input sequence will be computed for $t+1, \ldots, t+N_p$.

The optimal control sequence is computed by minimizing a cost function over prediction window (N_p) subject to some constraints [26]. The structure of the cost function differs in research, we are going to evaluate the form discussed in [13] where the cost function is defined as:

$$J(k) = J_{out}(k) + \beta \cdot J_{in}(k) \tag{7-2}$$

where J_{out} reflects the reference tracking error, J_{in} represents the control effort, and β is a non-negative weight parameter. Various options for reference tracking errors and control efforts can be found in [13].

Typically, the MPC input is held constant from a certain point onward in the prediction horizon, such that $u^*(k+j) = u^*(k+N_c-1)$ for $j = N_c, \ldots, N_p - 1$, where N_c denotes the control horizon. This strategy reduces computational complexity and generally yields a smoother control signal [27].

Due to the linearity in equation Eq. (7-1), future output values can be computed by recursively substituting the equations. In matrix notation, this can be expressed as:

$$\tilde{y}(k) = \tilde{C} \cdot x(k-1) + \tilde{D} \cdot \tilde{u}(k) \tag{7-3}$$

with vectors:

$$\tilde{y}(k) = \begin{bmatrix} \hat{y}(k \mid k) \\ \vdots \\ \hat{y}(k+N_p-1 \mid k) \end{bmatrix}, \tilde{u}(k) = \begin{bmatrix} u(k) \\ \vdots \\ u(k+N_p-1) \end{bmatrix}, \tilde{r}(k) = \begin{bmatrix} r(k) \\ \vdots \\ r(k+N_p) \end{bmatrix} \tag{7-4}$$

and state prediction matrices:

$$\tilde{C} = \begin{bmatrix} C \cdot A \\ C \cdot A^2 \\ \vdots \\ C \cdot A^{N_p} \end{bmatrix}, \tilde{D} = \begin{bmatrix} C \cdot B & 0 & \ldots & 0 \\ C \cdot A \cdot B & C \cdot B & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C \cdot A^{N_p-1} \cdot B & C \cdot A^{N_p-2} \cdot B & \ldots & C \cdot B \end{bmatrix} \tag{7-5}$$

Additional to optimizing a cost function, MPC is able to implement constraint in the optimization problem. The linear constraint is typically depicted by:

$$E(k) \cdot \tilde{u}(k) + F(k) \cdot \tilde{y}(k) + G(k) \cdot \tilde{r}(k) \leq h(k) \tag{7-6}$$

where $E(k) \in \mathbb{R}^{l \times n_u N_p}$, $F(k) \in \mathbb{R}^{l \times n_y N_p}$, $G(k) \in \mathbb{R}^{l \times n_y N_p}$ and $h(k) \in \mathbb{R}^l$ for some integer l. Minimizing the cost criterion subject to the linear constraint and the control horizon results in a convex quadratic optimization problem, which can be solved very efficiently [27].

7-2 Model predictive control (MPC) for max-plus linear (MPL) systems

The MPL system defined in Definition 2.4 and the linear state space system of Eq. (7-1) are very similar. The plus-times algebra is changed with max-plus algebra but the system remains linear. Therefore, we can derive state expectations in a similar way.

Suppose we have an MPL system with horizon N_p, we define the following vectors:

$$\tilde{y}(k) = \begin{bmatrix} \hat{y}(k \mid k) \\ \vdots \\ \hat{y}(k+N_p-1 \mid k) \end{bmatrix}, \tilde{u}(k) = \begin{bmatrix} u(k) \\ \vdots \\ u(k+N_p-1) \end{bmatrix}, \tilde{r}(k) = \begin{bmatrix} r(k) \\ \vdots \\ r(k+N_p) \end{bmatrix} \tag{7-7}$$

with $\tilde{u}$ the optimized input sequence, $\tilde{y}$ the estimated output and $\tilde{r}$ the desired reference signal. Again, we define an estimated output function of:

$$\tilde{y}(k) = \tilde{C} \otimes x(k-1) \oplus \tilde{D} \otimes \tilde{u}(k) \tag{7-8}$$

with $\tilde{C}$ and $\tilde{D}$, the prediction matrices:

$$\tilde{C} = \begin{bmatrix} C \otimes A \\ C \otimes A^{\otimes^2} \\ \vdots \\ C \otimes A^{\otimes^{N_p}} \end{bmatrix}, \tilde{D} = \begin{bmatrix} C \otimes B & \varepsilon & \dots & \varepsilon \\ C \otimes A \otimes B & C \otimes B & \dots & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ C \otimes A^{\otimes^{N_p-1}} \otimes B & C \otimes A^{\otimes^{N_p-2}} \otimes B & \dots & C \otimes B \end{bmatrix} \tag{7-9}$$

In this section, we will utilize the standard MPC problem as defined in [27], which is formulated as follows:

Definition 7.1. *(Standard MPC problem for MPL systems, [27]).*

$$\begin{aligned} &\min_{\tilde{u}(k),\tilde{y}(k)} J(k) = \min_{\tilde{u}(k),\tilde{y}(k)} J_{out}(k) + \beta \cdot J_{in}(k) \\ \textit{subject to: } &\tilde{y}(k) = \tilde{C} \otimes x(k-1) \oplus \tilde{D} \otimes \tilde{u}(k) \\ &E(k) \cdot \tilde{u}(k) + F(k) \cdot \tilde{y}(k) + G(k) \cdot \tilde{r}(k) \leq h(k) \\ &\Delta u(k+j) \geq 0 \quad \textit{for } j = 0, \dots, N_c - 1 \\ &\Delta^2 u(k+j) = 0 \quad \textit{for } j = N_c, \dots, N_p - 1 \end{aligned} \tag{7-10}$$

where J_{out} represents the tracking error and J_{in} the input effort with scaling factor β. The first constraint represents the state prediction. The second is the general constraint. The final 2 make sure inputs are growing and applies the control horizon cut-off to the system.

The cost function that will be used in this section is gathered from [27]. The cost function where n_y is the amount of outputs and n_u the amount of inputs is defined as:

$$\begin{aligned} J_{out}(k) &= \sum_{j=0}^{N_p-1} \sum_{i=1}^{n_y} \max(y_i(k+j \mid k) - r_i(k+j), 0) = \sum_{i=1}^{n_y N_p} \max(\tilde{y}_i(k) - \tilde{r}_i(k), 0) \\ J_{in}(k) &= \sum_{j=0}^{N_p-1} \sum_{i=1}^{n_u} (r_i(k+j) - u_i(k+j)) = \sum_{i=1}^{n_u N_p} (\tilde{r}_i(k) - \tilde{u}_i(k)) \end{aligned} \tag{7-11}$$

In the context of a manufacturing system, this cost function represents a strategy where raw materials are introduced into the system as late as possible, thereby ensuring that the internal buffer levels are kept as low as possible.

In the MPL-MPC problem that we will evaluate, we assume it to be unconstrained. Thus, the second constraint from Definition 7.1 is omitted. This allows for the analytical derivation of a closed-loop expression for the unconstrained MPL-MPC problem.

To express the tracking error J_{out} as a function of $\tilde{u}$, we substitute the tracking error from Eq. (7-11) along with the output expectation formula from Eq. (7-3). This yields the following result [28]:

$$J_{out}(\tilde{u}(k)) = \sum_{i=0}^{N_p-1} \max((\tilde{C} \otimes x(k-1) \oplus \tilde{D} \otimes \tilde{u}(k))_i - \tilde{r}_i(k), 0) \tag{7-12}$$

This tracking error is a convex function of $\tilde{u}(k)$. The reference term $\tilde{r}(k)$ has been added in J_{in} to obtain a bounded objective function of J_{in}. It has no influence on the optimization. Then we define the optimal solution of the MPL-MPC problem based on Lemma 1 of [28]

Lemma 7.1. *Assume $\beta < 1/N_p$, and define:*

$$\begin{aligned} \tilde{u}(k) &= [u(k-1), u(k-1), \ldots, u(k-1)]^T \\ \tilde{z}(k) &= [z^T(k \mid k), z^T(k+1 \mid k), \ldots, z^T(k+N_p-1 \mid k)]^T \\ &= \tilde{C} \otimes x(k-1) \oplus \tilde{D} \otimes \tilde{u}(k) \oplus \tilde{r}(k) \end{aligned} \tag{7-13}$$

and consider the optimization problem:

$$\tilde{u}^*(k) = \arg\max_{\tilde{u}(k)} \sum_{l=1}^{N_p} \tilde{u}_l(k), \tag{7-14}$$

subject to:

$$\begin{aligned} \tilde{D}_j \otimes \tilde{u}(k) &\leq z(k+j \mid k), \text{ for } j = 0, \ldots, N_p-1 \\ u(k+j) &\geq u(k+j-1), \text{ for } j = 0, \ldots, N_p-1 \end{aligned} \tag{7-15}$$

where $\tilde{D}_j$ denotes the jth row of $\tilde{D}$. Then $\tilde{u}^*$ is the optimal solution of the original MPL-MPC problem. The proof of this lemma is outside the scope of this research and can be found in [28].

Based on this optimization problem with constraints, we define the second lemma that derives the optimal input sequence [28].

Lemma 7.2. *Optimal input sequence for MPL-MPC problem*

$$u^*(k+j \mid k) = \begin{cases} \min\limits_i \left(z(k+i \mid k) - \tilde{D}_{ij} \right) & \textit{for } j = N_\mathrm{p} - 1 \\ \min\left(\min\limits_i \left(z(k+i \mid k) - \tilde{D}_{ij} \right), u^*(k+j+1 \mid k) \right) & \textit{for } j = 1, \ldots, N_\mathrm{p} - 2 \end{cases} \tag{7-16}$$

Then $\tilde{u}(k)$ is the optimal solution. Again, the proof is assumed to be outside the scope of this research but can be found in [28].

The optimal solution can be written as a min-plus expression [27].

$$\tilde{u}^*(k) = (-\tilde{D}^T) \otimes' \tilde{z}(k) \otimes' S \otimes' \tilde{u}^*(k) \tag{7-17}$$

where:

$$S = \begin{bmatrix} \top & 0 & \ldots & \top \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ \top & \ldots & & \top \end{bmatrix} \tag{7-18}$$

This expression is implicit; we can make it explicit by applying the min-plus Kleene star product to S. The Kleene star product operation is detailed in [10]. The explicit expression for the optimal input is defined as follows:

$$\begin{aligned} \tilde{u}^*(k) &= S^* \otimes' (-\tilde{D}^T) \otimes' \tilde{z}(k) \\ \tilde{z}(k) &= [z^T(k \mid k), z^T(k+1 \mid k), \ldots, z^T(k+N_p-1 \mid k)]^T \\ &= \tilde{C} \otimes x(k-1) \oplus \tilde{D} \otimes \tilde{u}(k) \oplus \tilde{r}(k) \end{aligned} \tag{7-19}$$

where:

$$S^* = E' \otimes' S \otimes' S^{\otimes'^2} \otimes \cdots = \begin{bmatrix} \top & 0 & \ldots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ \top & \ldots & & \top \end{bmatrix} \tag{7-20}$$

The resulting controller can be formulated as an MMPS function by substituting $\tilde{z}(k)$ inside the optimal input equation and by defining system matrices $\bar{H}$ and $\bar{G}$.

Definition 7.2. *(Optimal input sequence as an MMPS function, [27]). After substitution of $\tilde{z}$, we get:*

$$\tilde{u}^*(k) = \bar{0} \otimes' (-\tilde{D}^T) \otimes' (\tilde{C} \otimes x(k-1) \oplus \tilde{D} \otimes \bar{0} \otimes u(k-1) \oplus \tilde{r}(k)) \tag{7-21}$$

This can be written as a conjunctive MMPS function as:

$$\begin{aligned} \tilde{u}^*(k) &= \bar{H} \otimes' (\bar{G} \otimes (\begin{bmatrix} \bar{K} & \bar{L} & \bar{R} \end{bmatrix} \cdot \begin{bmatrix} x(k-1) \\ u(k-1) \\ \tilde{r}(k) \end{bmatrix})) \\ \text{with: } \bar{H} &= \bar{0} \otimes' (-\tilde{D}^T) \\ \bar{G} &= [\tilde{C}, \tilde{D} \otimes \bar{0}^T, d_\otimes(0)] \\ \bar{K} &= \begin{bmatrix} K \\ 0 \\ 0 \end{bmatrix}, \bar{L} = \begin{bmatrix} 0 \\ L \\ 0 \end{bmatrix}, \bar{R} = \begin{bmatrix} 0 \\ 0 \\ R \end{bmatrix} \end{aligned} \tag{7-22}$$

The zero vector, $\bar{0} = [0, \ldots, 0]$, is a zero row vector of length N_p. The diagonal matrix, $d_\otimes(0)$, has size $N_p \times N_p$ with zeros on the diagonal providing information of future reference values to the minimization problem.

The controller itself is a conjunctive MMPS function, which means that the closed-loop system itself is an MMPS system that can be defined by:

Definition 7.3. *(Implicit closed-loop conjunctive MMPS system). The following MPL system with conjunctive MMPS controller:*

$$\begin{aligned} x(k) &= A \otimes x(k-1) \oplus B \otimes u(k) \\ \tilde{u}^*(k) &= \bar{H} \otimes' (\bar{G} \otimes (\begin{bmatrix} \bar{K} & \bar{L} & \bar{R} \end{bmatrix} \cdot \begin{bmatrix} x(k-1) \\ u(k-1) \\ \tilde{r}(k) \end{bmatrix})) \end{aligned} \tag{7-23}$$

Can be written in the general implicit closed-loop form:

$$\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} I_{\otimes'} & \top \\ \top & \bar{H} \end{bmatrix} \otimes' (\begin{bmatrix} A & B & \varepsilon \\ \varepsilon & \varepsilon & \bar{G} \end{bmatrix} \otimes \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ \bar{K} & 0 & \bar{L} & \bar{R} \end{bmatrix} \cdot \begin{bmatrix} x(k-1) \\ u(k) \\ u(k-1) \\ \tilde{r}(k) \end{bmatrix}) \tag{7-24}$$

The MPL-MPC problem defined in this section will be validated through an example.

Example 7.1. *Closed-loop control of an MPL system by solving the MPL-MPC problem*

The simple manufacturing system from [29] will be evaluated. The manufacturing system is schematically represented in Figure 7-1.

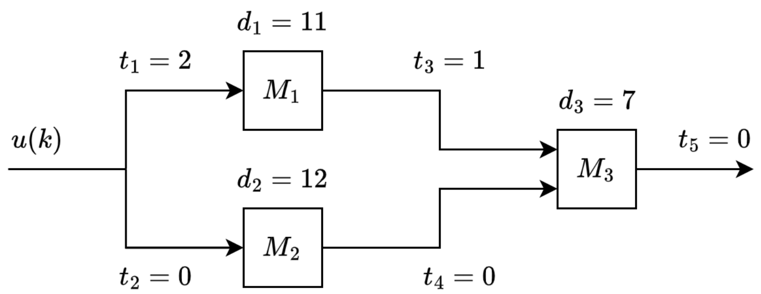


Figure 7-1: Simple Manufacturing System

The system consists of three production machines: M_1, M_2 and M_3. The raw material will be fed to the system through machines M_1 and M_2, then processed and send to M_3 where the product is assembled. The processing times (d) and transportation times (t) are depicted in Figure 7-1. Other transportation and set-up times are assumed negligible. The system can be modelled as a MPL system in the following way:

$$
\begin{aligned}
x(k) &= \underbrace{\begin{bmatrix} 11 & \varepsilon & \varepsilon \\ \varepsilon & 12 & \varepsilon \\ 23 & 24 & 7 \end{bmatrix}}_{A} \otimes x(k-1) \oplus \underbrace{\begin{bmatrix} 2 \\ 0 \\ 14 \end{bmatrix}}_{B} \otimes u(k) \\
y(k) &= \underbrace{\begin{bmatrix} \varepsilon & \varepsilon & 7 \end{bmatrix}}_{C} \otimes x(k)
\end{aligned} \tag{7-25}
$$

The input $u(k)$ represent the time at which a batch of raw material is fed to the system, $x_i(k)$ the time at which a machine M_i starts working and $y(k)$ the time at which a finished products leaves the system. The goal is to define an optimal input sequence that minimizes the differences between output and reference signal $r(k)$. The reference signal might represent a change in demand of the product. A lower growth rate represents more demand and vice versa.

The following closed-loop problem will be solved using the matrices $\bar{G}$ and $\bar{H}$, as defined in Definition 7.1, which include the state prediction matrices $\tilde{C}$ and $\tilde{D}$. We obtain:

$$
\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} I_{\otimes'} & \top \\ \top & \bar{H} \end{bmatrix} \otimes' \left(\begin{bmatrix} A & B & \varepsilon \\ \varepsilon & \varepsilon & \bar{G} \end{bmatrix} \otimes \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ I & 0 & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \cdot \begin{bmatrix} x(k-1) \\ u(k) \\ u(k-1) \\ \tilde{r}(k) \end{bmatrix} \right) \tag{7-26}
$$

The closed-loop system will be evaluated for two different prediction horizons: N_p such that the effect of the MPC controller can be verified. The simulation of the reference signal and two closed-loop systems with $N_p = 2$ and $N_p = 6$ is visualized in Figure 7-2.

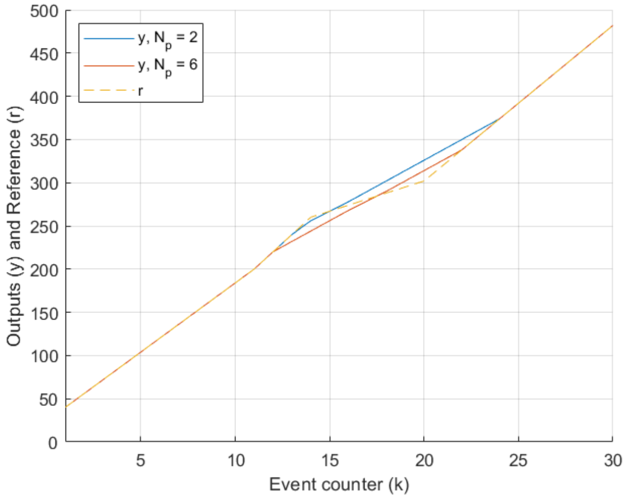


Figure 7-2: Simulation reference signal and closed-loop systems

In Figure 7-2, the closed-loop system effectively tracks the reference, but when demand increases, the manufacturing system reaches its minimum production time. A system with a longer prediction horizon can anticipate demand increases, allowing it to accelerate before the demand spike occurs. Consequently, the difference between the reference and output is smaller for higher horizons. This is illustrated in Figure 7-3, which shows the output-reference difference for both closed-loop systems. The better performance on this aspect can be verified in the total sum of the error which for both system is equal to:

$$N_p = 2 : \sum_{k=1}^{30} \|y(k) - r(k)\| = 110,\ N_p = 6 : \sum_{k=1}^{30} \|y(k) - r(k)\| = 70 \tag{7-27}$$

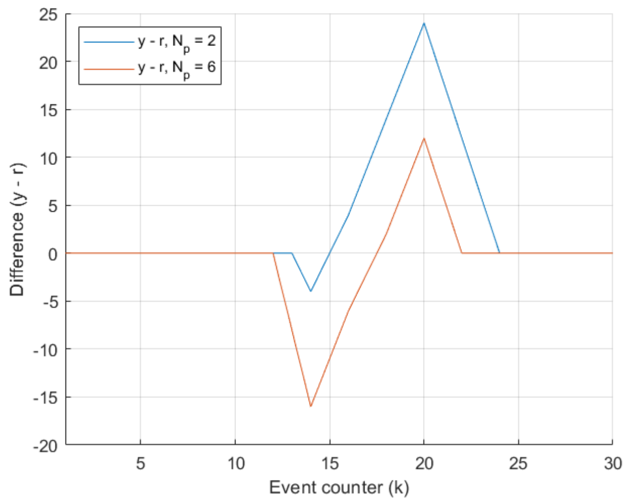


Figure 7-3: Difference reference signal and output closed-loop systems

Chapter 8

Case study: Urban railway system

In this chapter, the real life example of the urban railway system (URS) will be evaluated that is first introduced as max-plus linear parameter varying system in [30] and afterwards adopted in [31, 32]. However, all these researches used the max-plus linear parameter varying formats. The URS as max-min-plus-scaling (MMPS) was first introduced in [33] and afterwards also used and extended in [7, 19]. Section 8-1 introduces the URS with all its parameters and system equations. Also, based on [19], the URS is modeled in the implicit ABCD canonical form. In Section 8-2, the system is simulated without control for both a stable and unstable configuration. Section 8-3 defines an optimization problem to control the URS. This section covers multiple steps including the addition of control inputs, making the system closed-loop and defining the optimization problem. Section 8-4 uses the optimization problem to apply optimization-based closed-loop control on the URS to minimize the amount of passengers waiting at the train stations. Also, the stability of the closed-loop system is evaluated. Finally, Section 8-5 givens an analysis on the eigenvalues and eigenvectors of the URS.

8-1 Introduction of the urban railway system

This section introduces the urban railway system (URS), which is modeled as a partially homogeneous discrete-event (DE) max-min-plus-scaling (MMPS) system. The description of the URS is based on the model presented in [19]. The parameters of the URS model used throughout this chapter are summarized in Table 8-1. There, the four entries of the state of an URS are described as well. Namely, the arrival time ($a_j(k)$), the departure time ($d_j(k)$), the amount of passengers in the train ($\rho_j(k)$) and the amount of passengers at a station after a train leaves ($\sigma_j(k)$). The first two are identified as temporal signal states and the latter two as quantity signal states. The state vector for station j and train k is defined as:

$$x_j(k) = [a_j(k), d_j(k), \rho_j(k), \sigma_j(k)]^T \tag{8-1}$$

Table 8-1: List of model parameters used in the URS

Description	Parameter
Trains	k
Stations	j
Arrival time at station j of train k	$a_j(k)$
Departure time at station j of train k	$d_j(k)$
Number of passengers in train k when leaving station j	$\rho_j(k)$
Number of passengers at station j when train k leaves	$\sigma_j(k)$
Maximum capacity of the trains	ρ_{max}
Running time between consecutive stations	τ_r
Speed of passengers arriving at station	e
Passenger boarding speed	b
Passenger disembarking speed	f
Fraction of passengers disembarking at each station	β
Headway time	τ_H

For each simulation, Train 0 and Station 1 are initialized independently from the other trains and stations. This initialization process is based on a set of specific parameters, defined in Table 8-2.

Table 8-2: List of parameters initializing the URS

Description	Parameter
Difference in arrival times of consecutive trains at station 1	$\bar{\tau}$
Dwell time at station 1	τ_d
Departure time at station j of train 0	$\bar{d}_j$
Number of passengers in train 0 when leaving station j	$\bar{\rho}_j$
Number of passengers at station j when train 0 leaves	$\bar{\sigma}_j$

The derivation of the system equations of the URS is outside of the scope of this research but interested readers can have a look at [33]. The definition of the system equations are split in three parts: the evolution of station 1, the evolution of train 0 and the evolution of the other trains and stations. The evolution of station 1 is defined by the following set of equations:

$$\begin{aligned} a_1(k) &= a_1(k-1) + \bar{\tau} \\ d_1(k) &= a_1(k) + \tau_d \\ \rho_1(k) &= \rho_1(k-1) \\ \sigma_1(k) &= 0 \end{aligned} \tag{8-2}$$

The evolution of a train is based on the previous one. Therefore, to define the values of train 1, we need an initialization of train 0. The evolution of train 0 is defined as follows:

$$
\begin{aligned}
a_j(0) &= 0 \\
d_j(0) &= \bar{d}_j \\
\rho_j(0) &= \bar{\rho}_j \\
\sigma_j(0) &= \bar{\sigma}_j
\end{aligned} \tag{8-3}
$$

Finally, we have the evolution of the remaining trains and stations. The state evolution for $j > 1$ and $k > 0$ is defined as follows:

$$
\begin{aligned}
a_j(k) &= \max(d_{j-1}(k) + \tau_r, d_j(k-1) + \tau_H) \\
d_j(k) &= \min(\mu_1 a_j(k) + \mu_2 \rho_{j-1}(k) + \mu_3 \sigma_j(k-1) + (1-\mu_1) d_j(k-1), \gamma_1 + a_j(k) + \gamma_2 \rho_{j-1}(k)) \\
\rho_j(k) &= (1-\beta)\rho_{j-1}(k) + b(d_j(k) - a_j(k) - \frac{\beta}{f}\rho_{j-1}(k)) \\
\sigma_j(k) &= \sigma_j(k-1) + e(d_j(k) - d_j(k-1)) - b(d_j(k) - a_j(k) - \frac{\beta}{f}\rho_{j-1}(k))
\end{aligned} \tag{8-4}
$$

where $\mu_1 = \frac{b}{b-e}$, $\mu_2 = \frac{b}{b-e}\frac{\beta}{f}$, $\mu_3 = \frac{1}{b-e}$, $\gamma_1 = \frac{1}{b}\rho_{max}$ and $\gamma_2 = \frac{\beta}{f} - \frac{1-\beta}{b}$. The system equations will be modeled as an implicit MMPS system in the ABCD canonical form of Definition 2.17 based on the findings in [19]. The definition of the system matrices is separated in two steps. The system matrices corresponding to station 1 and the other stations. The system matrices for station 1, based on the system equations of Eq. (8-2) are defined as follows:

$$
\bar{A}_1 = \left[\begin{array}{ccccc}
0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon \\
\hline
\varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & 0
\end{array}\right],
\bar{B}_1 = \left[\begin{array}{cccccc}
\bar{\tau} & \top & \top & \top & \top & \top \\
\top & 0 & \top & \top & \top & \top \\
\top & \top & \tau_d & \top & \top & \top \\
\hline
\top & \top & \top & \top & 0 & \top \\
\top & \top & \top & \top & \top & 0
\end{array}\right]
$$

$$
\bar{C}_1 = \left[\begin{array}{cccc}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{array}\right], \quad
\bar{D}_1 = \left[\begin{array}{cccc}
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\hline
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0
\end{array}\right] \tag{8-5}
$$

The system matrices of the other stations, based on the system equations in Eq. (8-4) are defined as follows:

$$
\bar{A}_j = \left[\begin{array}{ccc|cc}
\tau_r & \tau_H & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon \\
\hline
\varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & 0
\end{array}\right],
\bar{B}_j \left[\begin{array}{cccc|cc}
0 & \top & \top & \top & \top & \top \\
\top & 0 & \top & \top & \top & \top \\
\top & \top & 0 & \gamma_1 & \top & \top \\
\hline
\top & \top & \top & \top & 0 & \top \\
\top & \top & \top & \top & \top & 0
\end{array}\right]
$$

$$\bar{C}_j = \left[\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & (1-\mu_1) & 0 & \mu_3 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & -e & 0 & 1 \end{array}\right], \bar{D}_{j,c} = \left[\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \mu_1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ \hline -b & b & 0 & 0 \\ b & e-b & 0 & 0 \end{array}\right], \bar{D}_{j,p} = \left[\begin{array}{cc|cc} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \mu_2 & 0 \\ 0 & 0 & \gamma_2 & 0 \\ \hline 0 & 0 & -b\gamma_2 & 0 \\ 0 & 0 & b\frac{\beta}{f} & 0 \end{array}\right] \tag{8-6}$$

The D matrix is split in two parts: the matrix $D_{j,c}$ contains information of the current station, for example: $a_j(k)$. The $D_{j,p}$ contains information of the previous station, for example: $a_{j-1}(k)$. All provided system matrices in Eq. (8-5) and Eq. (8-6) can be combined in one big set of system matrices:

$$\bar{A} = \begin{bmatrix} \bar{A}_1 & \varepsilon & \dots & \varepsilon \\ \varepsilon & \bar{A}_2 & & \vdots \\ \vdots & & \ddots & \varepsilon \\ \varepsilon & \dots & \varepsilon & \bar{A}_J \end{bmatrix}, \bar{B} = \begin{bmatrix} \bar{B}_1 & \top & \dots & \top \\ \top & \bar{B}_2 & & \vdots \\ \vdots & & \ddots & \top \\ \top & \dots & \top & \bar{B}_J \end{bmatrix},$$
$$\bar{C} = \begin{bmatrix} \bar{C}_1 & 0 & \dots & 0 \\ 0 & \bar{C}_2 & & \vdots \\ 0 & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \bar{C}_J \end{bmatrix}, \bar{D} = \begin{bmatrix} \bar{D}_{1,c} & 0 & \dots & 0 \\ \bar{D}_{2,p} & \bar{D}_{2,c} & & \vdots \\ 0 & \ddots & \ddots & 0 \\ 0 & \dots & \bar{D}_{J,p} & \bar{D}_{J,c} \end{bmatrix} \tag{8-7}$$

The states of the URS evolve according to the following formula:

$$x_j(k) = \bar{A} \otimes (\bar{B} \otimes (\bar{C} \cdot x(k-1) + \bar{D} \cdot x(k))) \tag{8-8}$$

In this chapter, also the stability of the URS will be discussed and this will follow the criteria defined in Section 2-4-3, where an MMPS system is considered max-plus bounded buffer stable if, for any initial state $x_0 \in \mathbb{R}^n$, the system's states remain bounded in Hilbert's projective norm as described in Definition 2.24. However, for the URS, we aim to demonstrate the stability of only a specific simulation corresponding to a set of initial conditions. Thus, the analysis will focus on whether the temporal signal states in the simulation are bounded in Hilbert's projective norm. Additionally, the quantity signal states should remain bounded in the infinity norm to prevent unbounded growth in the system.

The Hilbert's projective norm defined in Definition 2.24 calculates the absolute difference between temporal states, in case of the URS, the norm will be calculated using:

$$\|x\|_{\mathbb{P}} = d_j(k) - a_j(k),\ k \in \mathbb{Z}_+ \tag{8-9}$$

As the departure time of a train k at station j is always larger than the arrival time, no absolute value is necessary. This difference is also referred to as the dwell time at a station. The boundedness of the quantity signal states will be verified using the infinity norm calculated as follows:

$$\begin{aligned} \|x_i(k)\|_\infty &= \max(x_{1,2}(k)),\ k \in \mathbb{Z}_+ \\ x_1 &= [\rho_1, \rho_2, \ldots, \rho_j]^T \\ x_2 &= [\sigma_1, \sigma_2, \ldots, \sigma_j]^T \end{aligned} \tag{8-10}$$

If both states result in bounded values, we speak of a stable simulation of the URS. Furthermore, to assess the system's performance, we will evaluate the number of passengers waiting at station j when train k arrives, denoted as $p_j^{\text{wait}}(k)$. This value will be used as an objective function in Section 8-3-5. For now, it is introduced to analyze the performance of the URS. The number of waiting passengers is calculated as follows:

$$p_j^{\text{wait}}(k) = e \cdot (a_j(k) - d_j(k-1)) + \sigma_j(k-1) \tag{8-11}$$

This expression accounts for both the number of passengers arriving at station j between the arrival times of two consecutive trains and the number of passengers left behind by the previous train at station j.

8-2 Simulation of an autonomous URS

In this section, two simulations of the autonomous URS will be evaluated. In both cases, the train schedule of the URS evolves based on the situation where no control is applied. The difference in the two simulations origins in the initial conditions. The parameters of Table 8-1 are in both simulations equal to the following values:

$$\rho_{max} = 150, \quad \tau_r = 180, \quad e = 0.5, \quad b = 2, \quad f = 2, \quad \beta = 0.5, \quad \tau_H = 30 \tag{8-12}$$

The initial conditions of the two simulations include both a stable and an unstable configuration.

8-2-1 Simulation of the autonomous URS using stable initial conditions

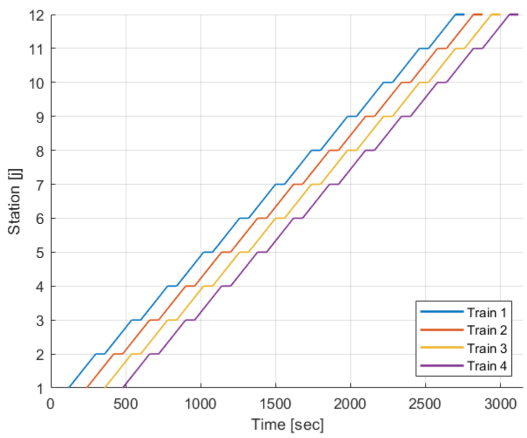
For the stable configuration, we define the parameters of Table 8-2 as follows:

$$\tau_d = 0, \quad \bar{\tau} = 120, \quad \bar{d_j} = 240, \quad \bar{\rho_j} = 120, \quad \bar{\sigma_j} = 0 \tag{8-13}$$

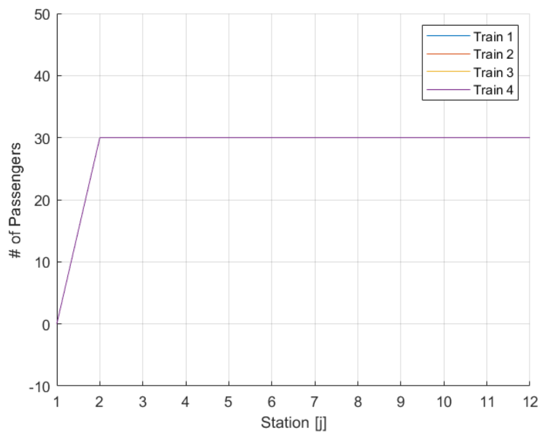
These parameters result in the following system equations for the evolution in station 1 and train 0:

$$\textbf{For}\quad \textbf{j = 1:}\quad \begin{aligned} a_1(k) &= a_1(k-1) + 120 \\ d_1(k) &= a_1(k) + 0 \\ \rho_1(k) &= 120 \\ \sigma_1(k) &= 0 \end{aligned} \qquad \textbf{For}\quad \textbf{k = 0:}\quad \begin{aligned} a_j(0) &= 0 \\ d_j(0) &= (j-1)\cdot 240 \\ \rho_j(0) &= 120 \\ \sigma_j(0) &= 0 \end{aligned} \tag{8-14}$$

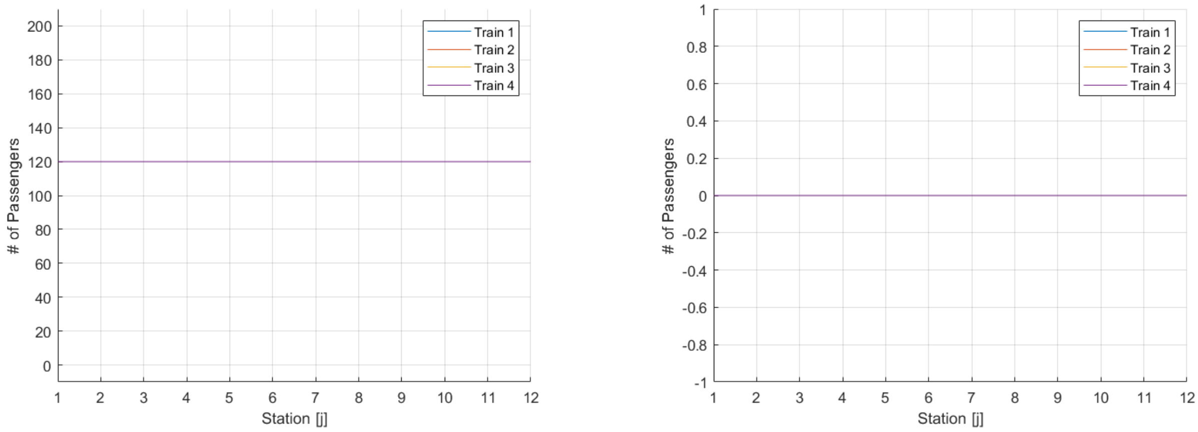
In Figure 8-1, the simulation of the autonomous URS with stable initial conditions is visualized. Figure 8-1(a) shows the place-time evolution of all the trains. The evolution is a combination of arrival times, dwell times and departure times. It can be seen that for the stable initial conditions, the difference between the trains remains constant over all the stations. Figure 8-1(b) shows the amount of people waiting at station j when train k arrives, which directly finds an equilibrium. Figure 8-1(c) shows the amount of people in the train and Figure 8-1(d), the amount of passengers at station j after train k leaves. Both states remain constant for all trains and stations.



(a) $a_j(k), \tau_{\text{dwell},j}(k)$ and $d_j(k)$



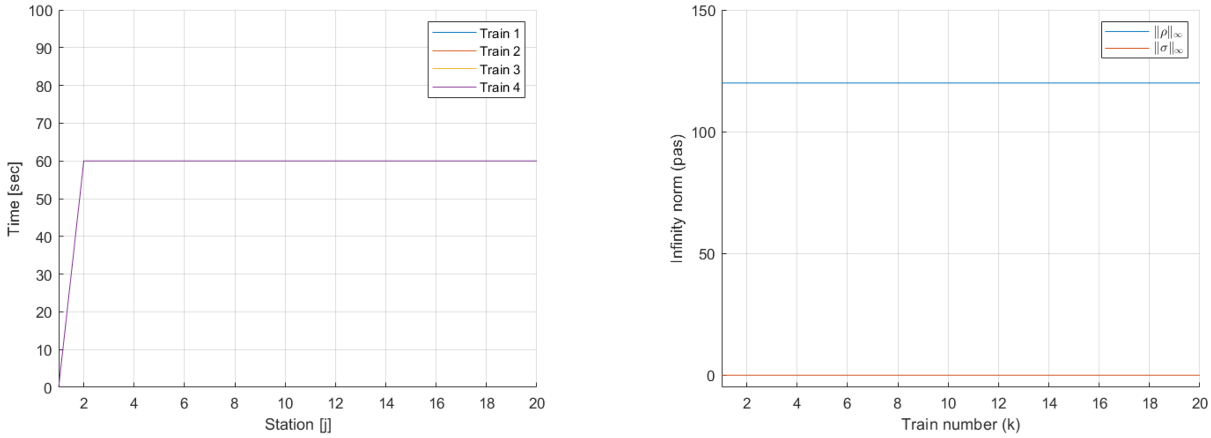
(b) $p_j^{\text{wait}}(k)$



(c) $\rho_j(k)$ **(d)** $\sigma_j(k)$

Figure 8-1: States and objective for the autonomous URS simulated from stable initial conditions

To verify the stability of this system, the Hilbert's projective norm of the temporal signal states following Eq. (8-9) and the infinity norm of the quantity signal states following Eq. (8-10) is derived. In Figure 8-2(a), the Hilbert's projective norm is visualized for all trains. There can be verified that the system's temporal states remain bounded. In Figure 8-2(b), the infinity norm of the quantity signal states is visualized, they also remain bounded. Therefore, it can be concluded that the initial conditions of Eq. (8-14) result in a stable simulation of the autonomous URS.



(a) $\|x\|_{\mathbb{P}}$ **(b)** $\|x_i(k)\|_\infty$

Figure 8-2: Hilbert's projective norm and infinity norm for the simulation of the autonomous URS from stable initial conditions

8-2-2 Simulation of the autonomous URS using unstable initial conditions

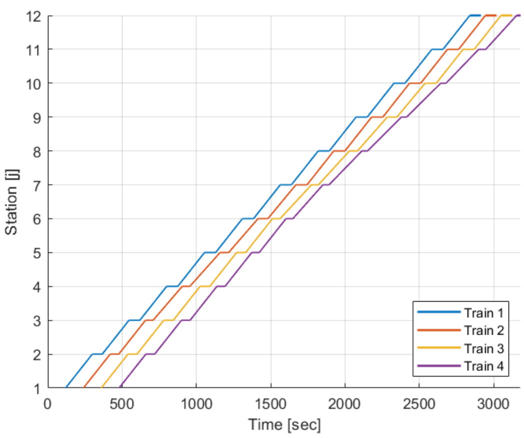
For the unstable configuration, we define the parameters of Table 8-2 as follows:

$$\tau_d = 0, \quad \bar{\tau} = 120, \quad \bar{d_j} = 240, \quad \bar{\rho_j} = 120, \quad \bar{\sigma_j} = 10 \tag{8-15}$$

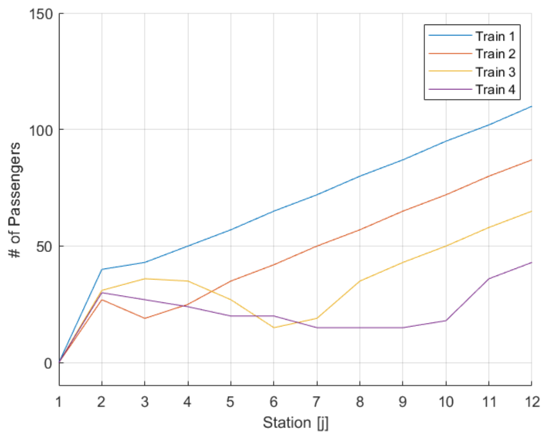
These parameters result in the following system equations for the evolution in station 1 and train 0:

$$\begin{aligned} &\textbf{For} \quad \textbf{j = 1:} \quad a_1(k) = a_1(k-1) + 120 && \textbf{For} \quad \textbf{k = 0:} \quad a_j(0) = 0 \\ &\qquad\qquad\quad d_1(k) = a_1(k) + 0 && \qquad\qquad\quad d_j(0) = (j-1)\cdot 240 \\ &\qquad\qquad\quad \rho_1(k) = 120 && \qquad\qquad\quad \rho_j(0) = 120 \\ &\qquad\qquad\quad \sigma_1(k) = 0 && \qquad\qquad\quad \sigma_j(0) = 10 \end{aligned} \tag{8-16}$$

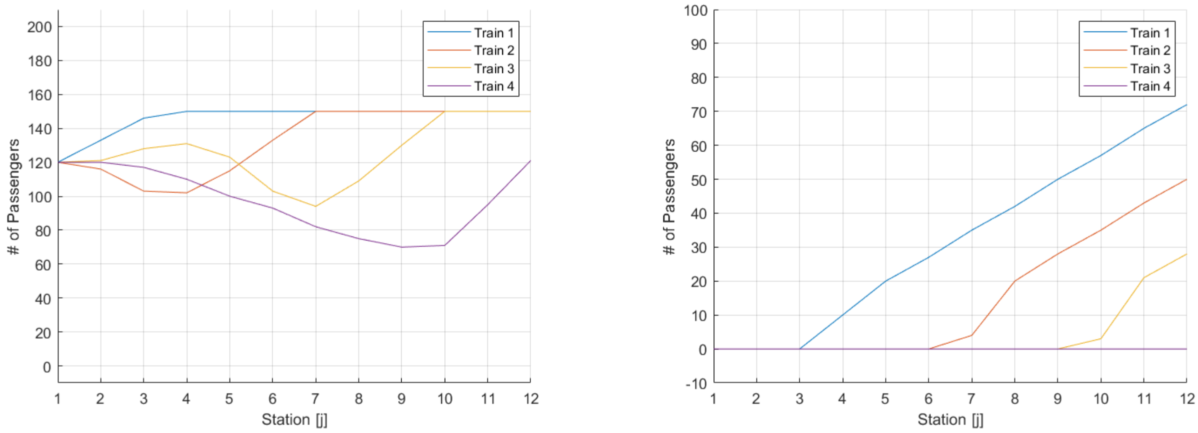
In Figure 8-3, the simulation of the autonomous URS with unstable initial conditions is visualized. Figure 8-3(a) shows the place-time evolution of all the trains. It can be seen that the difference between consecutive trains remains not constant. Figure 8-3(b) shows the amount of people waiting at station j when train k arrives, which is increasing for each train over time. Figure 8-3(c) shows the amount of people in the train, which quickly reaches its maximum for some of the trains. Figure 8-3(d)s shows the amount of passengers at station j after train k leaves, which, similar to Figure 8-3(b), keeps increasing the more stations a train passes.



(a) $a_j(k), \tau_{\text{dwell},j}(k)$ and $d_j(k)$



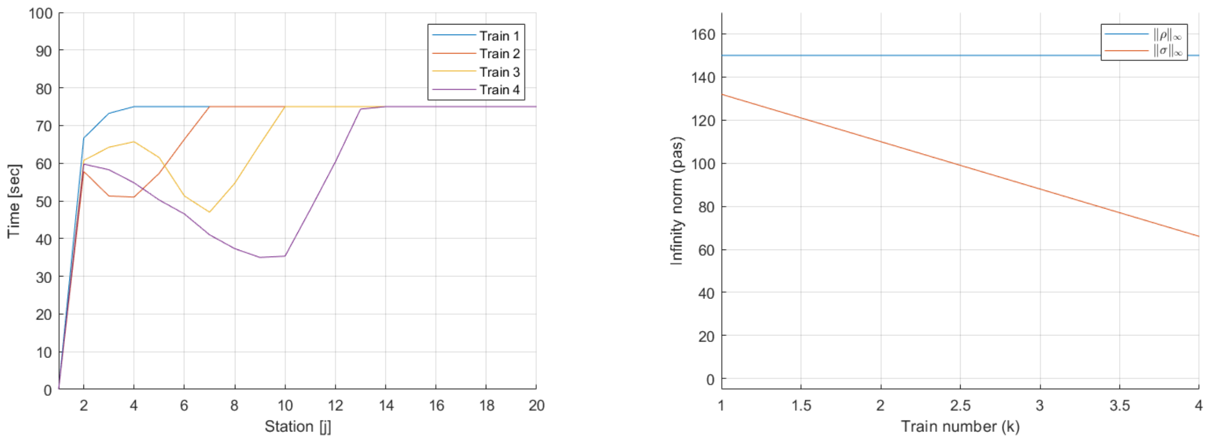
(b) $p_j^{\text{wait}}(k)$



(c) $\rho_j(k)$ **(d)** $\sigma_j(k)$

Figure 8-3: States and objective for the autonomous URS simulated from unstable initial conditions

The stability will again be evaluated based on the boundedness of the states. In Figure 8-4(a), the Hilbert's projective norm is visualized for all trains. There can be seen that the Hilbert's projective norm finds an equilibrium over time which is bounded. In Figure 8-4(b), the infinity norm of the quantity signal states is visualized, there can be seen that the norm for amount of passengers in the train is maximized for the whole simulation, which is bounded by its maximum. However, the amount of passengers at station j after train k leaves ($\sigma_j(k)$) will increase unbounded. The simulation is executed for twelve stations, and the introduction of more stations results in an unbounded increase of the infinity norm. This can also be verified in Figure 8-3(d) where the value increases unbounded by adding more stations. Therefore, it can be concluded that the unstable initial conditions from Eq. (8-16) result in an unstable simulation of the autonomous URS.



(a) $\|x\|_{\mathbb{P}}$ **(b)** $\|x_i(k)\|_\infty$

Figure 8-4: Hilbert's projective norm and infinity norm for the simulation of the autonomous URS from unstable initial conditions

When evaluating the difference between initial conditions, the origin of the instability can be recognized. The $\bar{\sigma}_j$ value is not zero for the unstable initial conditions. This value affects the number of people waiting at station j when train 1 arrives. Causing train 1 to reach its maximum capacity after a few stations.

In Figure 8-3(b), the amount of passengers waiting at the station keeps increasing over time under the unstable initial conditions. In the next section, the goal is to define an optimization problem to minimize the passengers waiting at a station while also stabilizing the simulation of the URS.

8-3 Defining an optimization problem for the URS

In this section, an optimization problem will be defined for the URS that minimizes the amount of passengers waiting at a station when a train arrives. The optimization problem will be defined in several steps. First, the implementation of control to the URS will be discussed. Also, the URS in ABCD canonical form of Eq. (8-8) will be extended with an input matrix. Second, the open-loop URS with input signals will be made closed-loop by using state feedback. Then, the objective and constraint function will be defined. Finally, the optimization problem will be presented and the selection of an optimization algorithm is discussed.

8-3-1 Adding input signals to the URS

The addition of input signals is based on [33]. The input signals will be used to influence the states of the system. The system equations for the URS are the same as depicted in Eq. (8-4).

The URS is a scheduling problem. Therefore, we are able to implement a control effort on the traveling speeds of the trains [33]. In previous cases, it was assumed that the running time between stations (τ_r) is equal in all situations. However, within boundaries, a train can speed up or slow down on a trajectory between two stations. This results in a new equation for the arrival time:

$$a_j(k) = \max(d_{j-1}(k) + \tau_r + u_{1,j}(k), d_j(k-1) + \tau_h) \tag{8-17}$$

The control effort $u_{1,j}(k)$ will be introduced as a quantity signal input because it is added to a temporal signal: $d_{j-1}(k)$. The input will be a bounded variable that only slight in- or decreases the time of train between two stations: $\tau_r + u_{1,j}(k)$. Hence a quantity signal.

Let us also introduce a second input $u_{2,j}(k)$; this input can decrease or increase the headway time τ_H between consecutive trains. The second input is added to the arrival time in the following manner:

$$a_j(k) = \max(d_{j-1}(k) + \tau_r + u_{1,j}(k), d_j(k-1) + \tau_h + u_{2,j}(k)) \tag{8-18}$$

The second input is also a quantity signal for the same reason as the first input. The input is added to a temporal signal, in this case the departure time of the previous train at station j: $d_j(k-1)$. The input is used to influence the headway time so again can only slightly in- or decrease that: $\tau_h + u_{2,j}(k)$. Hence also a quantity signal.

Both input signals being quantity signals means that the addition of both inputs does not affect time-invariance. Because we found in Section 3-2-1 that only temporal signal inputs influence time-invariance.

The new function for the arrival times and the implementation of the input signals will be used to make the system closed-loop. This is discussed in the next subsection.

8-3-2 Deriving a closed-loop URS using state feedback

Recall from Definition 3.1 that we can write an implicit open-loop system in the following form:

$$x(k) = A \otimes (B \otimes' (C \cdot x(k-1) + D \cdot x(k) + E \cdot u(k))) \tag{8-19}$$

The input values are designed as quantity signals which means there is no requirement for time-invariance. For the first input we design a closed-loop function that scales the dwell time at the previous station:

$$u_{j,1}(k) = \alpha_1 \cdot (d_{j-1}(k) - a_{j-1}(k)) \tag{8-20}$$

This means that if the dwell time at the previous station is large, the train was waiting for a long period at the previous station and will increase its speed to arrive faster at the following station. In that case, α_1 will be negative. Oppositely, if the dwell-time is small because the station are almost empty. In that case, the train can slow down such that it will not be ahead of schedule, resulting in a positive α_1. The second input will be scaling the headway time between two trains following the function:

$$u_{j,2}(k) = \alpha_2 \cdot (d_j(k-1) - d_{j-1}(k)) \tag{8-21}$$

The input scales the difference in departure times for a train $k-1$ at station j and the next train k at a previous station $j-1$ calculating the headway time. Depending on the sign of $u_{2,j}(k)$ the trains will be pushed closer or away of each other. In matrix format, we can write the state feedback functions in the following way:

$$\begin{bmatrix} u_{j,1}(k) \\ u_{j,2}(k) \end{bmatrix} = \begin{bmatrix} -\alpha_1 & \alpha_1 & 0 \\ 0 & -\alpha_2 & \alpha_2 \end{bmatrix} \cdot \begin{bmatrix} a_{j-1}(k) \\ d_{j-1}(k) \\ d_j(k-1) \end{bmatrix} \tag{8-22}$$

However, the state matrix exists of both explicit and implicit state values. Which means that in the closed-loop system both additional values to the C and D matrix are needed. Therefore, we define two separate input functions:

$$\begin{aligned} u_{j,1}(k) &= K_1 \cdot x_j(k-1) \\ u_{j,2}(k) &= K_2 \cdot x_j(k) \end{aligned} \tag{8-23}$$

The total closed-loop system of the URS can be described as follows:

$$\begin{bmatrix} x_j(k) \\ u_{j,1}(k) \\ u_{j,2}(k) \end{bmatrix} = \begin{bmatrix} \bar{A} & \varepsilon & \varepsilon \\ \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & 0 \end{bmatrix} \otimes (\begin{bmatrix} \bar{B} & \top & \top \\ \top & 0 & \top \\ \top & \top & 0 \end{bmatrix} \otimes' (\begin{bmatrix} \bar{C}_j & \bar{D}_j & \bar{E}_1 & \bar{E}_2 \\ K_1 & 0 & 0 & 0 \\ 0 & K_2 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_j(k-1) \\ x_j(k) \\ u_{j,1}(k) \\ u_{j,2}(k) \end{bmatrix} \tag{8-24}$$

Final step is to derive the input matrices $\bar{E}_1$, $\bar{E}_2$ and the size of the state feedback matrices K_1 and K_2. The feedback will applied for $j > 1$ such that the initialization of station 1 remains unaffected. Then, the total closed-loop matrices for the scaling: $\bar{C}_{CL}$ and $\bar{D}_{CL}$ are defined as:

$$\begin{aligned} \bar{C}_{CL} &= \begin{bmatrix} \bar{C}_1 & 0 & \dots & 0 \\ 0 & \bar{C}_2 + E_1 \cdot K_1 & & \vdots \\ 0 & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \bar{C}_J + E_1 \cdot K_1 \end{bmatrix} \\ \bar{D}_{CL} &= \begin{bmatrix} \bar{D}_{1,c} & 0 & \dots & 0 \\ \bar{D}_{2,p} + E_2 \cdot K_2 & \bar{D}_{2,c} & & \vdots \\ 0 & \ddots & \ddots & 0 \\ 0 & \dots & \bar{D}_{J,p} + E_2 \cdot K_2 & \bar{D}_{J,c} \end{bmatrix} \end{aligned} \tag{8-25}$$

with for each state the input and feedback matrices defined as:

$$E_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, K_1 = \begin{bmatrix} 0 & \alpha_2 & 0 & 0 \end{bmatrix}, E_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, K_2 = \begin{bmatrix} -\alpha_1 & \alpha_1 & 0 & 0 \\ 0 & -\alpha_2 & 0 & 0 \end{bmatrix} \tag{8-26}$$

Note that in 8-25, the top left part of $\bar{C}_{CL}$ has no input because of the initialization and that the control in the $\bar{D}_{CL}$ is added to the lower triangular part of the matrix. Therefore, we can define the total input matrices: $\bar{E}_1$, $\bar{E}_2$ in the following way:

$$\bar{E}_1 = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & E_1 & & \vdots \\ 0 & \ddots & \ddots & 0 \\ 0 & \dots & 0 & E_1 \end{bmatrix}, \bar{E}_2 = \begin{bmatrix} 0 & 0 & \dots & 0 \\ E_2 & 0 & & \vdots \\ 0 & \ddots & \ddots & 0 \\ 0 & \dots & E_2 & 0 \end{bmatrix} \tag{8-27}$$

The $\bar{A}$ and $\bar{B}$ matrices remain unaffected because the input signals are only during the scaling step, corresponding to input strategy 3 of Section 3-3. The URS is a an implicit closed-loop system following the state evolution:

$$x(k) = \bar{A} \otimes (\bar{B} \otimes' (\bar{C}_{CL} \cdot x(k-1) + \bar{D}_{CL} \cdot x(k))) \tag{8-28}$$

8-3-3 Definition of the objective function

The goal is to find the optimal values of K_1 and K_2 that minimize passenger waiting times. To achieve this, an objective function needs to be carefully defined. The objective function, which was presented in Eq. (8-11), involves minimizing the amount of passengers waiting at each station. This waiting is determined by the incoming flow of passengers e and the passengers left behind by the previous train, represented by $\sigma_j(k-1)$.

In addition to minimizing the waiting time, we introduce a penalty on the control effort. The purpose of this penalty is to prevent the optimization from selecting excessively large control inputs, which could lead to impractical or inefficient solutions. The total objective function, incorporating both passenger waiting and control effort, is defined as:

$$\begin{aligned} J(k) &= \|p_j^{wait}(k)\|_1 + \beta_1 \|u_{1,j}(k)\|_1 + \beta_2 \|u_{2,j}(k)\|_1 \\ p_j^{wait}(k) &= e \cdot (a_j(k) - d_j(k-1)) + \sigma_j(k-1) \end{aligned} \tag{8-29}$$

The objective function is a linear problem and we are using the 1-norm because we want to minimize the total values of the waiting passengers and control effort. The values β_1 and β_2 are weight parameters which scale the importance of the number of passengers waiting compared to minimizing the control efforts.

8-3-4 Definition of the constraints

In this subsection, the physical limitations of the URS will be formulated as constraint functions such that the simulation of the closed-loop system will remain realistic.

The first constraint entails the maximum capacity ($\rho_{\max}$) of the trains in the URS. The maximum capacity is already implemented in the system equations of the URS and therefore no additional constraint are necessary.

The second constraint entails non-negativity of the states. The temporal signal states including arrival and departure time and the quantity signal states based on number of people can both never be negative. Furthermore, the arrival and departure times are real values while the amount of passengers should be integer values. Therefore, the following constraints will be added to the system:

$$\begin{aligned}
&a_j(k), d_j(k), \rho_j(k), \sigma_j(k) \geq 0 \\
&a_j(k), d_j(k) \in \mathbb{R} \\
&\rho_j(k), \sigma_j(k) \in \mathbb{Z}_+
\end{aligned} \tag{8-30}$$

Next, the number of people waiting at a station calculated in the objective function should also be a non-negative integer value such that:

$$\begin{aligned}
&p_j^{wait}(k) \geq 0 \\
&p_j^{wait}(k) \in \mathbb{Z}_+
\end{aligned} \tag{8-31}$$

Furthermore, there is a limitation on the speed of the train and we do not want two trains to collide and therefore we introduce bounds on the running and headway times of the trains. Also, the running and headway times should be real values:

$$\begin{aligned}
\tau_r^{lb} \leq \tau_r + u_{1,j}(k) \leq \tau_r^{ub} \\
\tau_H^{lb} \leq \tau_H + u_{2,j}(k) \leq \tau_H^{ub} \\
u_{1,j}(k), u_{2,j}(k) \in \mathbb{R}
\end{aligned} \tag{8-32}$$

The minimization of the passengers waiting could result in trains having a very small dwell time: $d_j(k) - a_j(k)$, the time a train waits at a certain station. To give passengers the opportunity to enter the train, we add a lower bound on the dwell time using the following constraint function:

$$\tau_{dwell,\min} \leq a_j(k) - d_j(k) \tag{8-33}$$

Finally, we might want to bound the values of the feedback matrices to also prevent the optimization from selecting excessively large control signals. Therefore, we add the constraint:

$$\|[K_{1,2}]_{ij}\| \leq M \; \forall i, j \tag{8-34}$$

with $M \in \mathbb{R}$.

8-3-5 Summary of the optimization problem

An optimization problem for the URS to minimize the amount of passengers waiting at station j when train k arrives can be computed by combing all information in this section. The optimization problem is defined as follows:

Definition 8.1. *(Optimization algorithm for the URS). The optimal feedback matrices K_1 and K_2 for minimizing the amount of passengers waiting at a station j when train k arrives for the URS are derived by minimizing:*

$$\begin{aligned}
\min_{K_1,K_2} & \|p_j^{wait}(k)\|_1 + \beta_1\|u_{1,j}(k)\|_1 + \beta_2\|u_{2,j}(k)\|_1 \\
\textit{subject to} \quad & x(k) = \bar{A} \otimes (\bar{B} \otimes (\bar{C}_{CL} \cdot x(k-1) + \bar{D}_{CL} \cdot x(k))) \\
& \tau_r^{lb} \leq \tau_r + u_{1,j}(k) \leq \tau_r^{ub} \\
& \tau_H^{lb} \leq \tau_H + u_{2,j}(k) \leq \tau_H^{ub} \\
& \tau_{dwell,\min} \leq a_j(k) - d_j(k) \\
& [K_{1,2}]_{ij} \leq M \quad \forall i,j \\
& [K_{1,2}]_{ij} \geq -M \quad \forall i,j \\
& 0 \leq a_j(k), d_j(k), \rho_j(k), \sigma_j(k), p_j^{wait}(k) \\
& \rho_j(k), \sigma_j(k), p_j^{wait}(k) \in \mathbb{Z}_+
\end{aligned} \tag{8-35}$$

When using the optimization problem, the set of parameters defined in Table 8-3 must be initialized to obtain a realistic optimum.

Table 8-3: Parameters for the optimization problem

Description	Parameter
Initial guess optimization variable 1	$\alpha_{1,0}$
Initial guess optimization variable 2	$\alpha_{2,0}$
Scaling factor objective function 1	β_1
Scaling factor objective function 2	β_2
Minimum dwell time	$\tau_{dwell,min}$
Lower bound on running time between consecutive stations	τ_r^{lb}
Upper bound on running time between consecutive stations	τ_r^{ub}
Lower bound on headway time between consecutive trains	τ_H^{lb}
Upper bound on headway time between consecutive trains	τ_H^{ub}
Bound on state feedback matrices	M

This optimization problem for the minimization of the amount of passengers waiting will find the optimal values for the state feedback matrices. The problem will be solved using the genetic algorithm (GA). The GA is used instead of the Sequential Quadratic Programming (SQP) algorithm because the objective function contains an integer variable.

The standard SQP algorithm does not support integer optimization. Following [25], we can instead use heuristic research techniques. Downside to this is that there is no guaranteed convergence to a global optimum but on average results are good. Therefore, for the optimization of the URS the genetic algorithm will be used. More information on genetic algorithms can be found in [34]. The initial values for the optimization problem in Table 8-3 can be varied to search for different local optima.

8-4 Minimize passengers waiting for the URS

In this section, we are going to apply optimization-based closed-loop control on the URS using the algorithm defined in Definition 8-3. The same unstable initial conditions will be used as for Section 8-2-2. Therefore, the evolution of station 1 and train 0 will be the same as defined in 8-16. In Section 8-2-2 is defined that using these initial conditions to simulate an autonomous URS results in an unstable simulation.

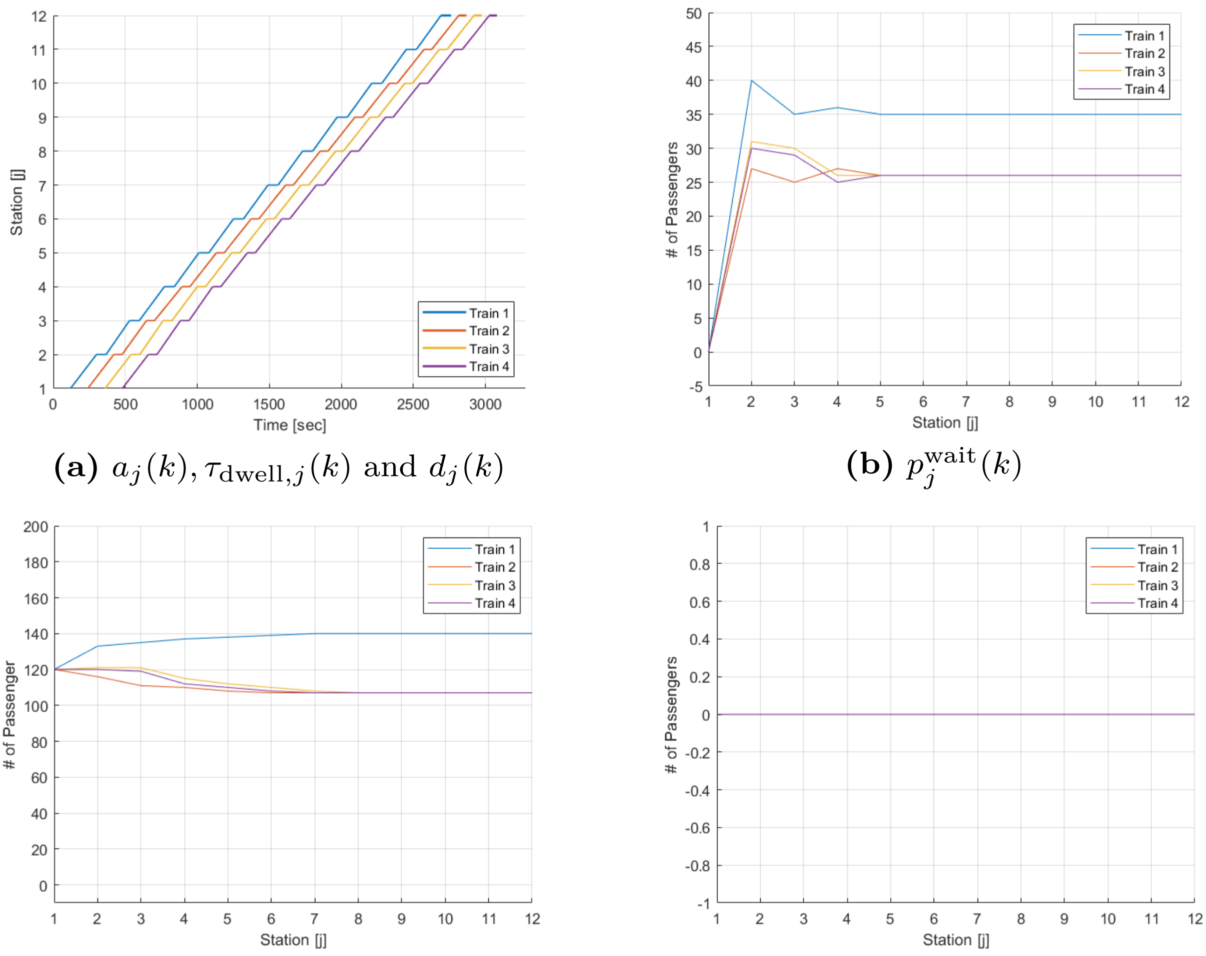
The optimal closed-loop solution for minimizing the amount of passengers waiting is obtained by using the following initialization of the optimization problem:

$$\begin{aligned} &\alpha_{1,0} = \alpha_{2,0} = 0, \quad \beta_1 = \beta_2 = 0.01, \quad \tau_{dwell,min} = 30, \\ &\tau_r^{lb} = 160, \quad \tau_r^{ub} = 230, \quad \tau_H^{lb} = 20, \quad \tau_H^{ub} = 80, \quad M = 1 \end{aligned} \tag{8-36}$$

Using MATLAB, an optimal solution for the feedback matrices K_1 and K_2 are computed and equal to:

$$K_1 = \begin{bmatrix} 0 & 0.1715 & 0 & 0 \end{bmatrix}, \; K_2 = \begin{bmatrix} 0.2822 & -0.2822 & 0 & 0 \\ 0 & -0.1715 & 0 & 0 \end{bmatrix} \tag{8-37}$$

In Figure 8-5, the states of the closed-loop URS and the amount of passengers waiting is visualized. In Figure 8-5(a) can be seen that the distance between trains differ and so there is control effort on the states. Figure 8-5 (b) shows that the amount of passengers waiting becomes constant over time, in comparison to the autonomous system with unstable conditions where the passengers waiting increased unbounded. Figure 8-5(c) shows that also the amount of passengers in the train over time becomes a constant value and remains well under its maximum capacity. Furthermore, Figure 8-5(d) shows that never any passengers are left behind at the station.



(a) $a_j(k), \tau_{\text{dwell},j}(k)$ and $d_j(k)$ **(b)** $p_j^{\text{wait}}(k)$

(c) $\rho_j(k)$ **(d)** $\sigma_j(k)$

Figure 8-5: Closed-loop simulation of the URS for unstable initial conditions, for $J = 12$ and $K = 4$

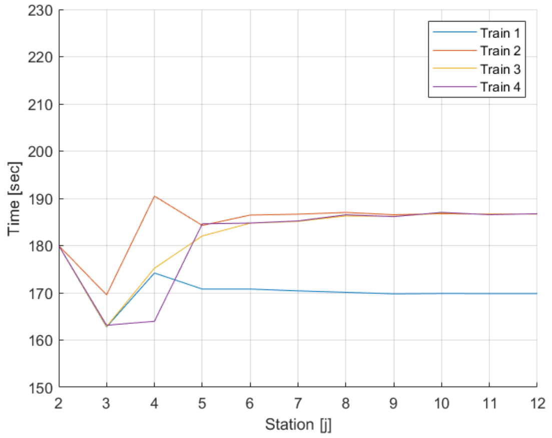
In Figure 8-6(a) the running time between consecutive stations is visualized to evaluate the effect of the first input signal. The running time is calculated using the formula:

$$\tau_{r,j}(k) = a_j(k) - d_{j-1}(k) = \tau_r + u_{1,j}(k) \tag{8-38}$$

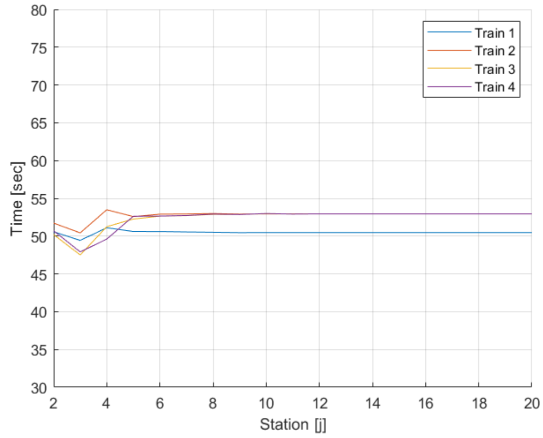
In Figure 8-6(a), note that the running time between stations 1 and 2 for a train k is placed at station 2. Furthermore, we can verify that the bounds on the running time, defined in Table 8-3, are validated. Figure 8-6 (b) visualizes the effect of the second input signal, the headway time, calculated using the formula:

$$\tau_{H,j}(k) = \tau_H + u_{2,j}(k) \tag{8-39}$$

The headway is increased due to positive input values but they remain well within the defined bounds of Table 8-3.



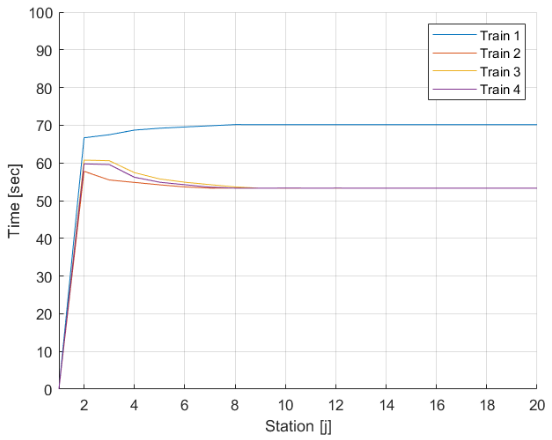
(a) $\tau_{r,j}$



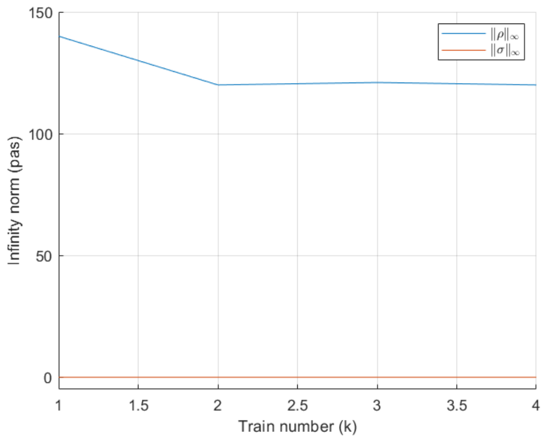
(b) $\tau_{H,j}$

Figure 8-6: Running and headway time of the closed-loop URS

The stability of the closed-loop URS will again be evaluated based on the boundedness of the states. In Figure 8-7(a), the Hilbert's projective norm is visualized for all trains. There can be seen that the Hilbert's projective norm finds an equilibrium over time which is bounded. In Figure 8-7(b), the infinity norm of the quantity signal states are bounded, similar to the autonomous URS with stable initial conditions. The amount of passengers in the trains never reach their maximum and therefore there will be no passengers left behind at a station as well. It can be concluded that the closed-loop structure stabilizes the simulation of the URS for the unstable initial conditions.



(a) $\|x\|_{\mathbb{P}}$



(b) $\|x_i(k)\|_\infty$

Figure 8-7: Hilbert's projective norm and infinity norm for the simulation of the autonomous URS from unstable initial conditions

Finally, the objective can evaluated in 8-5(b), the amount of passengers waiting at the station is settling at 35 persons for train 1 and the remaining trains converge to 26 passengers. This is even less than the amount of passengers waiting for the autonomous URS with stable initial conditions. Therefore, it can be concluded that the solution of the optimization problem in Definition 8-3 results in an improved performance.

8-5 Eigenvalues and eigenvectors of the urban railway system

This chapter finishes with an analysis of the URS related to the eigenvalues and eigenvectors of the autonomous and closed-loop system. The definition of eigenvalues and eigenvectors in MMPS systems can be found in Definition 2.22.

An eigenvector of the URS is defined as follows:

$$v_j = \begin{bmatrix} a_j & d_j & \rho_j & \sigma_j \end{bmatrix}^T \tag{8-40}$$

where j is the number of stations. For the computation of the eigenvalues and eigenvectors, the power algorithm will be used. The power algorithm in this research is based on the findings in [35]. Recall from Definition 2.22 that any time-invariant discrete event (DE) system has an additive eigenvalue and eigenvector if there exists a real number $\lambda \in \mathbb{R}$ and a vector $v \in \mathbb{R}^n$ such that:

$$\begin{aligned} f(v) &= v + \lambda \cdot s \\ s &= [\mathbf{1}_{n_t}^T, \mathbf{0}_{n_q}^T]^T \end{aligned} \tag{8-41}$$

Then, given a finite initial condition, we say that a system, after a number of steps, ends up in a periodic behavior, if there exist integers p, q with $p > q \geq 0$ and $c \in \mathbb{R}$ such that: $x(p) = x(q) + c$. In this case, the average weight or the eigenvalue is defined by: $\lambda = \frac{c}{p-q}$ [35]. The eigenvalues and eigenvectors always have to be finite. Combining the information we can define the power algorithm used in this section for computing the eigenvalues and eigenvectors of the URS.

Definition 8.2. *Power Algorithm for computing eigenvalues and eigenvectors of DE systems [35]*
The algorithm is based on two assumptions. First, for any initial state $x(0)$, the system ends up in a periodical behavior. Second, every periodic behavior has the same average weight. Then, the eigenvalues and eigenvectors can be computed using the following algorithm:

1. *Take an Arbitrary initial value $x(0)$*
2. *Iterate the DE system until there are integers p, q with $p > q \geq 0$ and a real number c such that $x(p) = x(q) + c$*
3. *Define the eigenvalue: $\lambda = \frac{c}{p-q}$*
4. *Define a candidate eigenvector: $v = \bigotimes_{j=1}^{p-q} (\lambda^{\otimes(p-q-j)} \otimes x(q + j - 1))$*

5. *If $f(v) = \lambda + v$ then v is a correct eigenvector and the algorithm can stop. If not follow the next step*
6. *Take $x(0) = v$ as new initial state and restart the state evolution until for some r it holds that $x(r+1) = \lambda + x(r)$ then $x(r)$ is an eigenvector of the system for the eigenvalue λ*

First, the eigenvalue and eigenvector of the URS using the stable initial conditions that are used in Section 8-2-1 will be computed. Figure 8-1 shows periodic behavior and in Section 8-2-1 is defined that the simulation is stable. The URS has two temporal values that should be growing entities and two quantity signal values that should remain bounded. The eigenvalue and eigenvector of the autonomous systems with stable initial conditions denoted with subscript stable calculated through the algorithm of Definition 8.2 provides the following values:

$$\begin{aligned}\lambda_{\text{stable}} &= 120\\ v_{\text{stable}} &= [300, 360, 120, 0, 540, 600, 120, 0, 780, 840, 120, 0]^T\\ &= [0, 60, 120, 0, 240, 300, 120, 0, 480, 540, 120, 0]^T\end{aligned} \tag{8-42}$$

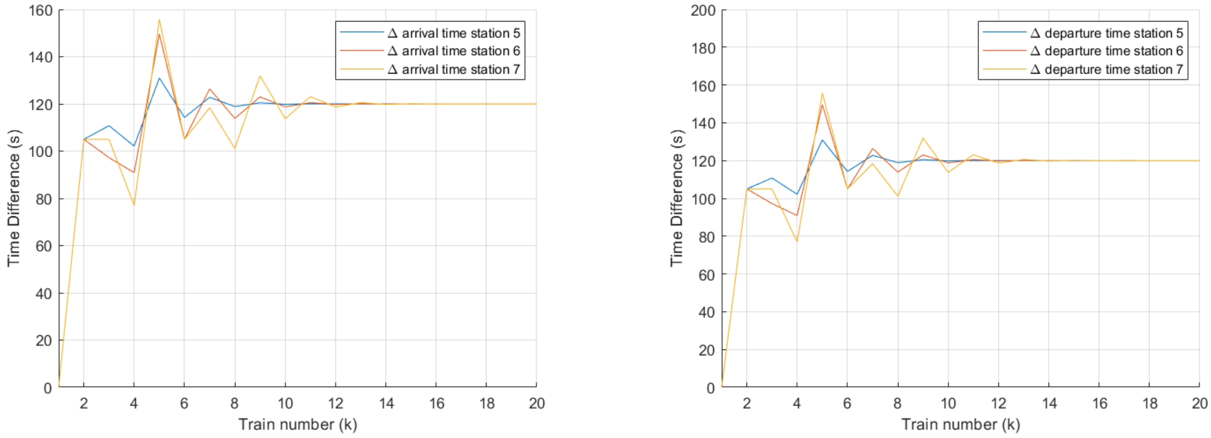
The eigenvector is in line with Eq. (8-40) and Eq. (8-41). The four states are recurring in the vector and s has a recurring structure: $s = [1, 1, 0, 0]^T$ for each station. The URS is simulated for three stations and therefore the eigenvector has twelve values; four values for each station.

Next, we are going to evaluate the eigenvalues and eigenvectors of the autonomous URS with unstable initial conditions. In Section 8-2-2 was verified that the unstable conditions result in an unstable simulation. However, this does not mean that the whole system itself is inherently unstable. To derive the eigenvalue of the system, we have to find if the unstable initial conditions, over time, result in a constant state evolution. In Figure 8-8, the difference in arrival and departure times for train k between two stations is simulated. The differences are computed such that the $\Delta a_j(k) = a_j(k) - a_{j-1}(k)$.

In Figure 8-8 can be seen, for stations $5, 6$ and 7, that the system settles at a constant difference over time and therefore the eigenvalue of the unstable initial conditions is equal to: $\lambda_{\text{unstable}} = 120$. Increasing the station number still results in convergence to the eigenvalue. However it will take more trains passing before it converges.

For a candidate eigenvector following the algorithm defined in Definition 8.2, we have to look into the region where the eigenvalue is active. Only in that region, an eigenvector can be found such that the step of the power algorithm in Definition 8.2 holds. An example of such eigenvector is:

$$\begin{aligned}\lambda_{\text{unstable}} &= 120\\ v_{\text{unstable}} &= [1740, 1800, 120, 0, 1980, 2040, 120, 0, 2220, 2280, 120, 0]^T\\ &= [0, 60, 120, 0, 240, 300, 120, 0, 480, 540, 120, 0]^T\end{aligned} \tag{8-43}$$

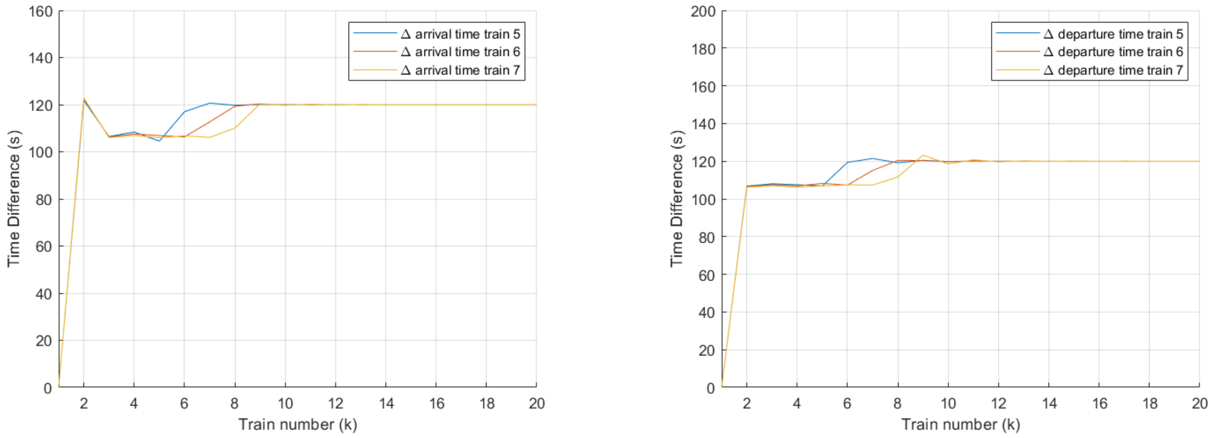


(a) $a_j(k) - a_{j-1}(k)$ **(b)** $d_j(k) - d_{j-1}(k)$

Figure 8-8: Difference in arrival and departure times between stations

This is the same eigenvector as Eq. (8-42) which makes sense because the system itself is not unstable, its initial conditions make it unstable and result in an unbounded growth of the quantity signal states.

Finally, we are going to evaluate the eigenvalues and eigenvectors of the closed-loop URS and discuss if the closed-loop structure alters them. First, we are again verifying if the system will converge to an eigenvalue.



(a) $a_j(k) - a_{j-1}(k)$ **(b)** $d_j(k) - d_{j-1}(k)$

Figure 8-9: Difference in arrival and departure times between stations, closed-loop system

In Figure 8-9 can be seen that the difference in arrival and departure times for train k between two stations converges to the same eigenvalue as for the autonomous systems: $\lambda_{CL} = 120$.

For a candidate eigenvector, we take the same approach as for the autonomous URS with unstable initial conditions. We look at the region where the eigenvalue is active. An example of such eigenvector is:

$$\begin{aligned}\lambda_{\text{CL}} &= 120 \\ v_{\text{CL}} &= [4595.2, 4648.4, 106, 0, 4836.1, 4889.2, 106, 0, 5075.7, 5128.7, 106, 0]^T \\ &= [0, 53.2, 106, 0, 240.9, 294, 106, 0, 480.5, 533.5]^T \end{aligned} \tag{8-44}$$

This is a different eigenvector compared to the autonomous URS systems. The difference is due to the change in constant running and dwell time of the system in its new found equilibrium. From Figure 8-6, the equilibrium values for the running and dwell time can be derived. In Figure 8-6(a), all running times, except train 1, go to an equilibrium value of: $\tau_{r,j} = 186,7$ and in Figure 8-6(b) the dwell time, except train 1, finds an equilibrium value of: $\tau_{dwell,j} = 53.3$. This values are similar to the evolution of the temporal states of v_{unstable}, except for some rounding errors. Finally, the new equilibrium value of the passengers in the train can be found in 8-5(c) and is equal to: $\rho_j(k) = 106$.

It can be concluded that the closed-loop structure does not affect the eigenvalue of the URS but does change the eigenvector of the system.

Chapter 9

Conclusions and contributions

In this chapter, we will reflect on the work done in this thesis and evaluate the research questions stated in Section 1-2. The conclusions will be separated in three parts based on the three main research questions. Section 9-1 covers the first set of research question relating the topic of integrating input signals in max-min-plus-scaling (MMPS) systems in a systematic way. Section 9-2 discusses the implementation of both open-loop and closed-loop control to MMPS systems. Section 9-3 discussed the results on the research done to achieve several control objectives using closed-loop control on MMPS systems. Finally, Section 9-4 gives an overview of the contributions this thesis provides to the field of System and Control.

9-1 On the integration of input signals for MMPS systems

After giving a thorough introduction to the field of dioid algebra and MMPS systems, the foundation was built to investigate the possibilities to include input signals inside the system equations of MMPS systems. The first research questions from Section 1-2 were states as:

- How can input signals be systematically integrated in MMPS systems, considering both temporal and quantity-based signals?
    1. How can the existing ABCD canonical form be extended to accommodate input signals?
    2. What constraints must be applied to the input structure to preserve key properties, such as time-invariance and stability?

For the first research sub-question, inspiration is drawn from the linear state space system in conventional algebra. An additional input matrix (E) and input vector are added to the ABCD canonical form for MMPS systems. The input matrix is added inside the scaling step of the process of MMPS systems. By doing this, we are able to accommodate inputs anywhere in the system. However, before analysing the different types of input structures. An analysis is done on the newly formed ABCDE canonical form were properties such as time-invariance, monotonicity and non-expansiveness are evaluated. This analysis provides constraints for the input matrix (E) in combination with the type of input signal added to the system such that these properties are preserved, answering the second sub-question.

After finding an MMPS model that accommodates input signals and an analysis is done to define the constraints on the system to preserve properties, the main question could be answered. It could be concluded that there are three distinctive ways of adding input signals to the MMPS systems. These three input strategies can be distinguished based on differences in their mathematical foundation. Finally, a summary is provided with a systematic overview for each input strategy providing typical system equations, constraints on the input matrix and the possibility of which type of input signal could be added through the strategy, answering the first main research question.

9-2 On the implementation of control for MMPS systems

After identifying three input strategies that can used to implement input signals to MMPS system. The next step was to use these strategies to apply control on the MMPS systems. The second set of research questions defined in Section 1-2 were stated as:

- What is a systematic approach to implementing control strategies in MMPS systems?
    1. What are the effects and limitations of using open-loop control in MMPS systems?
    2. What are the effects and limitations of using closed-loop control in MMPS systems?

For first research sub-question, open-loop control is generally introduced and all three input strategies were evaluated by applying open-loop control. For each input strategies the effects of control is evaluated based on the findings in the matrix format and the examples provided in each section.

For the second research sub-question, open-loop control is extended to closed-loop control where the principle of state feedback is introduced. A general state feedback function for MMPS systems is defined that accommodates all types of controllers. This newly defined feedback function was implemented in the three input strategies again providing a systematic approach for each strategy on the matrix format of the closed-loop systems. Each closed-loop system was evaluated by using examples finding the effects and limitations for closed-loop control on each input strategy, answering research question 2.

9-3 On control objectives for closed-loop MMPS systems

In the previous research question of this thesis, we developed a systematic framework of implementing closed-loop control on MMPS distinguishing three input strategies. Next, this framework is used to reach several control objectives for MMPS systems. The final set of research questions defined in Section 1-2 were stated as:

- How can closed-loop control strategies be employed to regulate the dynamic behavior of MMPS systems?
  1. What specific objectives are pursued by implementing closed-loop control in MMPS systems?
  2. What methods and criteria are most effective for designing and optimizing closed-loop controllers for MMPS systems?
  3. How can closed-loop control mechanisms be tailored and applied to improve performance in urban railway systems?

For answering the first research sub-question, control objectives are defined for optimization-based control methods. An MMPS system, in most cases, represents the cycle of a product or machine through a process. Often, the lower the growth rate, the faster the system operates. Therefore, the first goal is to minimize the growth rate of the MMPS system under certain circumstances. Secondly, it might be desirable to closely follow a reference signal that changes over time and find the most efficient path over time. Therefore, the second research goal includes minimizing the difference between outputs and reference signal. Finally, some MMPS systems consist of multiple regions. It might be desirable to switch from one to another stable region to follow another stable growth rate over time. Concluding the final research goal.

The research goals will be reached by implementing optimization based closed-loop controllers and answering research sub-question 2. To obtain this, an optimization algorithm is defined that consists of a cost function that is based on the research goal and a set of constraints ensuring properties of MMPS systems are preserved. This algorithm is successfully implemented on examples to reach all of the defined research goals, finding solutions that are optimal. Afterwards, a well-known optimisation based-control method is introduced and applied on max-plus linear (MPL) systems. The optimal input constructed is an MMPS system making the total system a closed-loop MMPS system that is able to optimize its input based on future reference values.

Finally, all knowledge gathered from this thesis is combined in evaluating the case study of the urban railway system (URS), answering research sub-question 3. The URS is simulated for a stable and unstable initial condition. An optimization algorithm is computed and an objective to minimize the amount of passengers waiting at a station before a train arrives is defined. Using closed-loop control, an optimal feedback matrix is computed that minimizes the objective. Also, this closed-loop structure is able to let the unstable initial conditions converge a lot faster to an equilibrium compared to the autonomous case and is able to stabilize the system.

9-4 Contributions

This thesis is a contribution to the field of systems and control and more specifically discrete event and max-min-plus-scaling systems through the following results:

- A schematic way is described to implement input signals in MMPS systems by defining the ABCDE canonical form

- Distinguishing three input strategies of adding input signal to MMPS systems based on mathematical differences
- Introducing the concept of open-loop and closed-loop control on MMPS systems
- Defining the concept of regions in MMPS systems and computing an algorithm that finds the boundaries between regions
- Creating optimization-based closed-loop control algorithms to reach several objectives such as minimizing the closed-loop growth rate, minimizing the difference between output and reference signal and forcing the system to switch between stable regions
- Applying an optimization-based control technique to improve the performance of the URS

Chapter 10

Recommendations for future work

This chapter presents several recommendations for future work, building on the findings and methodologies explored in this thesis. Each suggestion aims to deepen the understanding of max-min-plus-scaling (MMPS) systems, broaden their potential applications, or enhance the methodological approaches within this field. Additionally, specific recommendations are provided to improve the closed-loop structure of the urban railway system (URS).

- **Establish requirements for monotonicity and non-expansiveness of explicit MMPS with both temporal and quantity signals**

This thesis defines requirements for monotonicity and non-expansiveness of explicit MMPS systems with only temporal signals. Future work could investigate whether similar requirements can be established for systems incorporating both temporal and quantity signals.

- **Define monotonicity and non-expansiveness requirements for implicit MMPS systems**

While monotonicity and non-expansiveness are well-defined for explicit MMPS systems, similar definitions for implicit systems remain unexplored. Examining these properties in implicit MMPS systems would be very useful. Particularly in the situation where implicit systems are time-invariant, monotonic and non-expansive, also known as topical. Then the system has always exactly one growth rate.

- **Analyse closed-loop eigenvalues and eigenvectors in MMPS systems**

The algorithm that computes eigenvalues and eigenvectors for autonomous MMPS systems can also be used to study the closed-loop systems. See if the addition of feedback matrices or reference signals in the closed-loop system matrices affect the eigenvalues and eigenvectors.

- **Reformulate the optimization problem as a linear programming problem (LPP)**

MMPS systems can be represented by linear equations, enabling reformulation of the optimization problem as a linear programming problem (LPP). This reformulation could eliminate the need for complex algorithms like Sequential Quadratic Programming (SQP) or genetic algorithm (GA), simplifying solution approaches and enhancing computational efficiency.

- **Experiment with alternative input signals in the URS**

In this thesis, two specific input signals were introduced to control the URS. Future work could explore additional input signals at various points in the URS system equations, leading to alternative closed-loop structures. Potentially, improving the performance of the URS.

- **Conduct simulations with variable parameters**

The initial conditions of the URS were modified in this study, yet the impact of varying parameters, such as passenger arrival rates at different stations, remains an area for future exploration. This would better simulate real-world conditions, where station dynamics vary significantly.

- **Develop more real-world examples of MMPS systems**

Currently, few real-world applications of MMPS systems exist in the literature. Creating practical examples could not only fill this gap but also offer new tools for problem-solving and decision-making processes. Applications could include transportation systems, production scheduling, or queuing networks, where MMPS modelling could offer a distinctive advantage.

Appendix A

Mathematical properties

In this Appendix, mathematical properties regarding dioid algebra can be found. These mathematical properties can be used to define similarities between dioid and plus-times algebra. Furthermore, several properties of max and min operators are discussed, which can be used to rewrite MMPS system from one to another canonical form.

A-1 Algebraic properties of dioid algebra

In Section XX, the definition was already discussed, where a dioid uses two operations namely addition ($\oplus$) and multiplication ($\otimes$). This algebra has a lot of properties that also account in plus-times algebra. The properties are described in max-plus algebra but also apply for min-plus. The properties are gathered from [10].

Associativity

If the same operator is used twice or more after one another, the order does not matter.

$$\begin{aligned} \forall a, b, c \in \mathbb{R}_\varepsilon : \quad & a \oplus (b \oplus c) = (a \oplus b) \oplus c \\ \forall a, b, c \in \mathbb{R}_\varepsilon : \quad & a \otimes (b \otimes c) = (a \otimes b) \otimes c \end{aligned} \tag{A-1}$$

Commutativity

Changing the order of the operation will not change the result.

$$\forall a, b \in \mathbb{R}_\varepsilon : \quad a \oplus b = b \oplus a \quad \text{and} \quad a \otimes b = b \otimes a. \tag{A-2}$$

Distributivity of $\otimes$ over $\oplus$

When the $\otimes$ operator is used on $\oplus$ operations, all elements in the $\oplus$ operation will be multiplied by the value relating to $\otimes$.

$$\forall a, b, c \in \mathbb{R}_\varepsilon : \quad a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c). \tag{A-3}$$

Existence of a zero element

$$\forall a \in \mathbb{R}_\varepsilon : \quad a \oplus \varepsilon = \varepsilon \oplus a = a. \tag{A-4}$$

Existence of a unit element

$$\forall a \in \mathbb{R}_\varepsilon : \quad a \otimes e = e \otimes a = a. \tag{A-5}$$

The zero is absorbing for $\otimes$

The zero is absorbing in combination with the $\otimes$ operator.

$$\forall a \in \mathbb{R}_\varepsilon : \quad a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon. \tag{A-6}$$

Idempotency of $\oplus$

The continuous use of the $\oplus$ operation on the same element will not change the previous result.

$$\forall a \in \mathbb{R}_\varepsilon : \quad a \oplus a = a. \tag{A-7}$$

The final property is particularly important. This means that in dioid systems information can not be obtained using inverse equations. For example when you have:

$$\forall a, b \in \mathbb{R}_\varepsilon : \quad a \oplus b = b. \tag{A-8}$$

The information of a is lost in the process and can not be obtained by just the information of b.

A-2 Properties of max and min operators

The properties in this appendix are based on information from [12]. Let $\alpha, \beta, \gamma, \delta \in \mathbb{R}$. Since minimization is distributive with respect to maximization we have that:

$$\min(\alpha, \max(\beta, \gamma)) = \max(\min(\alpha, \beta), \min(\alpha, \gamma)) \tag{A-9}$$

and thus,

$$\begin{aligned} \min(\max(\alpha, \beta), \max(\gamma, \delta)) &= \max(\min(\alpha, \gamma), \min(\alpha, \delta), \min(\beta, \gamma), \min(\beta, \delta)) \\ \max(\min(\alpha, \beta), \min(\gamma, \delta)) &= \min(\max(\alpha, \gamma), \max(\alpha, \delta), \max(\beta, \gamma), \max(\beta, \delta)) \end{aligned} \tag{A-10}$$

Furthermore, addition is distributive with respect to minimization and maximization. Which gives that:

$$\begin{aligned} \min(\alpha,\beta)+\min(\gamma,\delta) &= \min(\alpha+\gamma,\alpha+\delta,\beta+\gamma,\beta+\delta) \\ \max(\alpha,\beta)+\max(\gamma,\delta) &= \max(\alpha+\gamma,\alpha+\delta,\beta+\gamma,\beta+\delta) \end{aligned} \tag{A-11}$$

To switch between just the min and max operators we can use that:

$$\max(\alpha,\beta) = -\min(-\alpha,-\beta) \tag{A-12}$$

Finally, if we have a positive $\rho \in \mathbb{R}$, then:

$$\begin{aligned} \rho\cdot\max(\alpha,\beta) &= \max(\rho\cdot\alpha,\rho\cdot\beta) \\ \rho\cdot\min(\alpha,\beta) &= \min(\rho\cdot\alpha,\rho\cdot\beta) \end{aligned} \tag{A-13}$$

Bibliography

[1] S.R. Daams. *Max-Min-Plus-Scaling Systems: General Concepts, Stability Properties and Potential Control Strategies.* Literature Research, Delft University of Technology, 2024.

[2] B. Heidergott, G.J. Olsder, and J. van der Woude. *Max Plus at Work.* Princeton University Press, 2005.

[3] B. De Schutter and T.J.J. van den Boom. Max-plus algebra and max-plus linear discrete event systems: An introduction. In *Proceedings of the 9th International Workshop on Discrete Event Systems (WODES'08)*, Göteborg, Sweden, pp. 36–42, May 2008.

[4] G. Cohen, S. Gaubert, and J.-P. Quadrat. *Max-plus algebra and system theory: Where we are and where we go now.* Annual Reviews in Control (IFAC, Elsevier), 1999.

[5] B. De Schutter and T.J.J. van den Boom. Model predictive control for max-min-plus-scaling systems. In *Proceedings of the 2001 American Control Conference*, (Arlington, Virginia), pp. 319-324, June 2001.

[6] B. De Schutter and T. van den Boom. *On model predictive control for max-min-plus-scaling discrete event systems.* Systems Engineering, vol 19, 2000.

[7] T. van den Boom, A. Gupta, B. De Schutter, and R. Beek. Max-min-plus-scaling systems in a discrete-event framework with an application in urban railway. In *Proceedings of the 22th IFAC World Congress*, (Yokohama, Japan), July 2023.

[8] 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. IEEE, 2024.

[9] 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 Workshop on Discrete Event Systems, vol. 58, no. 1, 60–65., 2023.

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

[11] S.S. Farahani. *Approximation methods in stochastic max-plus systems.* PhD thesis, Delft University of Technology, 2012.

[12] T. van den Boom, S. Markkassery, A. Gupta, and B. De Schutter. *"Max-Min-Plus-Scaling Systems in a discrete event framework: Modeling, Control, and Scheduling.* Technical Report, Delft University of Technologies, 2024.

[13] B. De Schutter and T.J.J. van den Boom. Model predictive control for max-min-plus-scaling systems - efficient implementation. In *Proceedings of the 6th International Workshop on Discrete Event Systems (WODES'02) (M. Silva, A. Giua, and J.M. Colom, eds.)*, Zaragoza, Spain, pp. 343–348, Oct. 2002.

[14] J Gunawardena. *From max-plus algebra to nonexpansive mappings: a nonlinear theory for discrete event systems.* Theoretical Computer Science, vol. 293, no. 1, pp. 141–167, 2003.

[15] J. Cochet-Terrasson, S. Gaubert, and J. Gunawardena. *Dynamics of min-max functions.* Technical Report HPL-BRIMS-97-13, Hewlett-Packard Laboratories, 1997.

[16] J. van der Woude. *A characterization of the eigenvalue of a general (min,max,+)-system.* Discrete Event Dynamic Systems, vol. 11, pp. 203-210, 2001.

[17] J. Gunawardena. *Min-max functions.* Discrete Event Dynamic Systems, vol. 4, pp. 377-407, 1994.

[18] Q. Zhao, D.Z. Zheng, and X. Zhu. *Structure properties of min-max systems and existence of global cycle time*, chapter vol. 46, no. 1, pp. 148–151. IEEE Transactions on Automatic Control, 2001.

[19] S. Markkassery, T. van den Boom, and B. De Schutter. *Dynamics of Implicit Time-Invariant Max-Min-Plus-Scaling Discrete-Event Systems.* To be submitted, 2024.

[20] A. Gupta. *Max-plus-algebraic hybrid automata: Beyond synchronisation and linearity.* PhD thesis, Delft University of Technology, Jan. 2023.

[21] J.P. Hespanha. *Linear Systems Theory.* Princeton University Press, 2018.

[22] J.B. Rawlings, D.Q. Mayne, M.M. Diehl, and S. Barbara. *Model Predictive Control: Theory, Computation, and Design 2nd Edition.* Nob Hill Publishing, 2020.

[23] B. Friedland. *Control system Design: An Introduction to State-space Methods.* Dover Publications, 2005.

[24] K.J. Astrom and R.M. Murray. *Feedback Systems: An Introduction for Scientists and Engineers.* Princeton University Press, Princeton and Oxford, 2009.

[25] T.J.J. van den Boom and B. De Schutter. *Optimization for System and Control.* Lecture Notes for the Course SC 42056, Delft University of Technology, 2024.

[26] J. Komenda, S. Lahaye, S.-L. Boimond, and T. van den Boom. *Max-plus algebra in the history of discrete event systems.* Annual Reviews in Control, 45, pp. 240–249., 2018.

[27] T. van den Boom and B. De Schutter. *Model predictive control of manufacturing systems with max-plus algebra.* Chapter 12 in Formal Methods in Manufacturing (J. Campos, C. Seatzu, and X. Xie, eds.), Industrial Information Technology, CRC Press, ISBN 978-1466561557, pp. 343–378, Feb. 2014.

[28] T. van den Boom, B. De Schutter, and I. Necoara. *On MPC for max-plus-linear systems: Analytic solution and stability.* Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005 (CDCECC'05), Seville, Spain, pp. 7816–7821, Dec. 2005.

[29] B. De Schutter and T. van den Boom. *Model predictive control for max-plus-linear discrete event systems.* Automatica, vol. 37, pp. 1049–1056, July, 2001.

[30] B. Zwerink Arbonés. *An exploratory study in max-plus linear parameter varying systems.* MSc Thesis, Delft University of Technology, 2019.

[31] M. Abdelmoumni, A. Gupta, J. van der Woude, and T. van den Boom. *Max-plus linear parameter-varying systems: An application to urban railway.* Internal Report, 2021.

[32] M. Abdelmoumni. *Max-plus linear parameter varying systems - A framework and solvability.* MSc Thesis, Delft University of Technology, 2021.

[33] R.E.S. Beek. *Max-Plus Linear Parameter Varying Systems.* Delft University of Technology, MSc Thesis, Delft University of Technology, 2022.

[34] M. Mitchell. *An introduction to genetic algorithms.* Cambridge, Mass. : MIT Press, 1996.

[35] Subiono and J. van der Woude. *Power algorithms for (max, +)- and bipartite (min,max, +)-systems.* Discrete Event Dynamic Systems: theory and applications, vol. 10, no. 4, pp. 369–389, 2000.

Glossary

List of Acronyms

CT	continuous time
DE	discrete event
DT	discrete time
GA	genetic algorithm
LP	linear programming
ME	Mechanical Engineering
MMP	max-min-plus
MMPS	max-min-plus-scaling
MPC	model predictive control
MPL	max-plus linear
SQP	Sequential Quadratic Programming
URS	urban railway system

List of Symbols

N_c	Control horizon
N_p	Prediction horizon
Δ	Difference operator, $\Delta x_{1,2} = x_1 - x_2$
λ	Additive eigenvalue or growth rate
$\mathbb{P}$	Max-plus Hilbert projective norm
$\mathbb{R}$	Set of real numbers
$\mathbb{R}_\top$	Set of real numbers including $\top$
$\mathbb{R}_\varepsilon$	Set of real numbers including ε
$\mathbb{Z}_+$	Set of positive Integers

$\mathcal{R}$	Either $\mathbb{R}, \mathbb{R}_\varepsilon$ or $\mathbb{R}_\top$
μ	Multiplicative eigenvalue
$\oplus$	Max-plus addition operator ("o-plus")
$\oplus'$	Min-plus addition operator ("o-plus-prime")
$\otimes$	Max-plus multiplication operator ("o-times")
$\otimes'$	Min-plus multiplication operator ("o-times-prime")
$\top$	Min-plus zero element, $\varepsilon = \infty$
ε	Max-plus zero element, $\varepsilon = -\infty$
n_q	Amount of quantity signals
n_t	Amount of temporal signals
v	Additive eigenvector or fixed-point
w	Multiplicative eigenvector