Matrix spans in max-plus algebra and a graph-theoretic approach to switching max-plus linear systems.

Vangelis Peter Kalamboukis

Master of Science Thesis

Delft University of Technology

Delft Center for Systems and Control

Matrix spans in max-plus algebra and a graph-theoretic approach to switching max-plus linear systems.

Master of Science Thesis

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

Vangelis Peter Kalamboukis

May 11, 2018

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

Abstract

The relation between graph theory and max-plus algebra has been well studied since the inception of max-plus algebra. It has been shown that any square matrix over the max-plus semiring can be represented as a weighted directed graph. Furthermore, properties of these matrices, such as irreducibility and its (unique) eigenvalue, can be determined by its graph-theoretical interpretation. However, this graph-theoretical interpretation has not yet been extended to SMPL systems.

Switching max-plus-linear (SMPL) systems are an extension of max-plus-linear systems (MPL) for modelling discrete-event systems. While for MPL systems the system is described by one max-plus-linear state equation and one max-plus-linear output equation, for SMPL systems the system is described by more than one mode of operation, each consisting of its own unique max-plus-linear state equation and max-plus-linear output equation. The different modes allow for more efficient modelling of changes to the structure of the system. The switching between the different modes of operation can be deterministic, stochastic or a combination of the two.

Due to the fact that max-plus algebra is an idempotent algebra and there is no opposite operation to max-plus addition, vectors spaces in max-plus algebra cannot be defined in the same way as for conventional algebra. As a result, determining the span of matrices has to be performed in a different way than for matrices in conventional algebra as matrix ranks are also defined in a different way. Determining the span of matrices in both max-plus algebra and conventional algebra is important as it allows for the calculation of the set of states that can be accessed (reached) by MPL systems and LTI systems respectively.

The purpose of this thesis is three-fold: firstly, a method is developed for accurately determining the span of max-plus matrices, secondly, this method is applied to MPL systems with the purpose of determining the set of accessible states for autonomous and non-autonomous MPL systems and establishing the necessary conditions for structural controllability by making use of its graphical representation and thirdly, to model

SMPL systems and also establish properties such as structural controllability by means of a graph-theoretic framework.

Table of Contents

Acknowledgements **ix**

1 Introduction **1**
1-1 Motivation . . . 1
1-2 Contribution . . . 3
1-3 Outline . . . 3

2 Graph Theory **5**
2-1 Introduction . . . 5
2-2 Directed Weighted Graphs . . . 6
2-3 Bipartite Graphs . . . 9
2-4 Signal-Flow Graph . . . 10
2-5 Dynamic Graph . . . 11
2-5-1 Coloured Dynamic Graph . . . 12

3 Graph Theory in Linear System Theory **15**
3-1 Introduction . . . 15
3-2 Structural Controllability . . . 16
3-3 Structural Controllability in terms of the Dynamic Graph . . . 18

4 Max-Plus Algebra **19**
4-1 Basic Definitions . . . 19
4-2 Spectral Theory . . . 21
4-3 Asymptotic Growth Rate and Cycle-Time Vector . . . 24
4-4 Max-Plus Geometry . . . 26
4-4-1 Obtaining the span of non-square matrices. . . . 34

5 Max-Plus-Linear (MPL) Systems & Switching Max-Plus-Linear (SMPL) Systems 41

5-1 Max-plus linear (MPL) systems . . . 41

5-2 Switching max-plus linear (SMPL) systems . . . 45

5-2-1 Annotation of switching sequences . . . 48

5-2-2 Graphical Representation of SMPL systems . . . 49

6 Structural Controllability for MPL and SMPL systems. 51

6-1 Introduction . . . 51

6-2 Controllability and Structural Controllability for MPL systems. . . . 52

6-2-1 Accessibility for Autonomous Systems . . . 54

6-2-2 Accessibility for Non-Autonomous Systems . . . 57

6-3 Structural Controllability for SMPL systems . . . 62

6-3-1 Structurally finite SMPL systems. . . . 62

6-3-2 Non-structurally finite SMPL systems . . . 66

7 Conclusion and Future Work 69

7-1 Conclusion . . . 69

7-2 Future Work . . . 70

Bibliography 71

Glossary 75

List of Acronyms . . . 75

List of Symbols . . . 75

List of Figures

2-1 An example of an undirected graph with four nodes and six edges. 6

2-2 An example of a directed graph. 7

2-3 An example of a bipartite graph. 9

2-4 An example of a signal-flow graph that is spanned by a cactus. 11

2-5 A dynamic graph for the A and B matrices of (2-3). 12

2-6 An example of a coloured dynamic graph. 13

3-1 The signal flow graph of Example 3.2.1. 17

4-1 span(V) of Example 4.3 for $v_3 = 0$. Closure (cyan): Polyhedral set Γ_1 described by $\{(-1 \leq v_2 - v_1 \leq 1) \wedge ((1 \leq v_3 - v_2 \leq 3))\}$ Connections & extreme points: Polyhedral sets Γ_2, Γ_3 $\{((1 \leq v_2 - v_1 \leq 3) \wedge (v_2 - v_3 = -3)) \vee ((-1 \leq v_3 - v_2 \leq 1) \wedge (v_2 - v_1 = -1))\}$ 33

4-2 span(V) (cyan,black) and span($V^{\clubsuit}$) (blue) of Example 4.3 for $v_3 = 0$. 34

4-3 span(V) of Example 4.4 for $v_3 = 0$. 38

4-4 $span^{-}(V^{\clubsuit})$ in comparison to $span(V)$ of Example 4.4. 39

5-1 A simple production system. 42

5-2 A simple production system. 46

5-3 A dynamic graph for the SMPL system of (6-20)-(6-21). 50

6-1 An example of a coloured dynamic graph. 66

List of Tables

Acknowledgements

I would like to thank both my supervisors Dr.ir. A.J.J. van den Boom and Dr.ir. J.W. van der Woude for their invaluable assistance during the course of this thesis. Our meetings throughout this period have been one of the highlights of my brief academic life. I would also like to thank the final member of the committee Prof.dr.ir. B.H.K. De Schutter.

My parents for standing by me no matter the circumstances. My sister for reminding me that I am never alone. My grandparents for granting me the opportunity to leave a rock of the Aegean and find my place in the world and $\Gamma\iota\alpha\gamma\iota\alpha$ for teaching me there is no greater virtue than patience.

My friends Nick, Vangelis, Mimos and Kostas for treating me no differently even though I have been away for 7 years. Youssef for being the best classmate I have ever had. Jordan, Pavlos and Makis for putting up with me. Chris and Constantine for being the ever-present element throughout my student life in Swansea and Delft.

Finally, I would like to thank God.

Delft, University of Technology
May 11, 2018

Vangelis Peter Kalamboukis

Dedicated to Grandma and Grandpa (Margaret & Peter Sivil) who passed away during the course of this Thesis and without whom none of this would have been possible.

Chapter 1

Introduction

1-1 Motivation

Plato said that knowledge is certain and infallible and thus, knowledge is the only thing true and beautiful. For something to be true, he continues, it must be a product of reason and not perception. Such a product of reason is mathematics. With the intention of finding truth in this world, humans have always attempted to express the physical world through a mathematical lens. Even though Plato's statement is open to argument, there is no denying that mathematical modelling of the physical world has been of great benefit.

Mathematical modelling (or simply modelling) offers a road map for a better understanding of the world. To the present day, a great amount of effort is being afforded to developing more efficient and more realistic models. Graphs are a mathematical modelling tool that have risen to prominence due to their natural, visual representation and the powerful combinatorial properties [1] they possess.

Graph theory, the study of graphs, has been applied to a wide range of applications over the years. Social systems, physical systems, biochemistry, computer science and scheduling are some examples of fields where graph theory has been applied. Scheduling, in particular, will be the application of interest with regards to this thesis.

In the present day where automation is an integral part of everyday life, scheduling has become an increasingly popular field of research. Scheduling can be defined as the allocation of limited resources to a set of jobs with the purpose of meeting certain criteria. Jobs are sequences of operations, where each operation is performed on a particular resource. To further illustrate this point, think of a railway network. A train that begins at station A, stops at stations B and C and terminates at station D, can be thought of as a job. The tracks between the stations are the resources, as no two trains can operate on the same track for safety reasons. Travelling from one station to another can be thought of as the operation. So, the job of going from A to D consists of performing the three operations $A \to B, B \to C, C \to D$. If more trains were introduced to the network, performing the same or different jobs, then conflicts could arise due to competing for the same resource at the same time. By designing and implementing a schedule the objective is to have an optimized system with no conflicts.

Such systems where progression is reliant upon completion of distinct operations that are asynchronous in regards to each other, are called discrete event systems (DES). In contrast to discrete time systems where the difference in time between two consecutive time steps is always the same, in DES the evolution of the state is dependent on previous events. Consequently, two consecutive time steps of the system may not have the same time difference. The main obstacle that is encountered when trying to model such systems is that usually they are nonlinear in conventional algebra. Nonetheless, a certain class of DES can become linear when modelled in the max-plus algebra. Linear systems in max-plus algebra are called Max-Plus-Linear (MPL) systems.

Max-plus algebra has been used for the analysis and modelling of DES since the landmark book by Baccelli et al. [2]. The main advantage is that when this class of DES is modeled in max-plus algebra, it becomes linear. In addition to this, various results from conventional linear system theory have been adapted to max-plus algebra, thus providing a mathematical framework for an in-depth analysis of these systems. A disadvantage of MPL systems, however, is that a change in the structure of the system cannot be modelled, as the structure is fixed. With the purpose of overcoming this, Switching Max-Plus-Linear (SMPL) systems were introduced by van den Boom and De Schutter [3]. SMPL systems are a collection of MPL systems, called modes, where switching is allowed between different modes. By using different modes of operation for modelling, it is possible to model potential changes in the structure of the system.

Hence, the introduction of graph theory. Graphs are an efficient and powerful tool for analyzing structural properties of systems. They have been used extensively [4, 5, 6, 7, 8] for the structural analysis of conventional linear systems. It has been shown that properties like controllability and observability can be established through a graph-theoretical approach.

In addition to this, graphs already play a prominent role in spectral theory of max-plus matrices. The eigenvalue of a square matrix and whether a matrix is irreducible or not can be determined directly through the graphical representation of the matrices. Subsequently, it would be of interest to enhance the graph-theoretical approach to MPL and SMPL systems by investigating whether a framework could be developed with the purpose of establishing properties such as controllability in the max-plus setting.

1-2 Contribution

Among the contributions of this thesis is the use of a graphical representation for the modelling of SMPL systems and the use of a graph-theoretical approach for the establishment of structural controllability for MPL and SMPL systems. Conditions for establishing controllability of MPL systems through timed event graphs already exist [2]. In this thesis they are translated to a different type of graph, namely directed graphs. Additionally, the dynamic graph is employed for the purpose of modelling SMPL systems and establishing controllability of such systems. To our knowledge, no such graph-theoretical representation of SMPL systems exists. Finally, a method is developed for determining the span of max-plus matrices and is applied on MPL systems, for the purpose of establishing the set of values that can be achieved by the system. A distinction is made between MPL systems in which a control input is present and for systems that do not include a control input.

1-3 Outline

Chapter 2 provides an introduction to graph theory and all the relevant concepts that will be used. Different types of graphs, such as the weighted directed graph, the bipartite graph, the signal-flow graph and the dynamic graph will all be presented. In Chapter 3 an overview of how graph-theoretical concepts have been used for the structural analysis of conventional linear system will be given. Max-plus algebra and its connection to graph theory will be introduced in Chapter 4. Moreover, in Section 4-4 a method will be presented for establishing the span of max-plus matrices. While, in Chapter 5 the focus will shift to MPL and SMPL systems. Examples of such models will be given and sufficient conditions for them to be stabilizable will also be presented. Furthermore, a graph-theoretic representation of SMPL systems will be defined. Finally, in Chapter 6 sufficient conditions for controllability of MPL and SMPL systems will be derived through a graph-theoretical point of view, and the method presented in Section 4-4 will be applied to MPL systems.

Chapter 2

Graph Theory

2-1 Introduction

Graph theory is a branch of mathematics that studies graphs. Graphs are formed by vertices (also called nodes) and edges (also called arcs) connecting the vertices. More formally, a graph is a pair of sets (V, E), where $V(G)$ is the set of vertices and $E(G)$ is the set of edges for the graph G. Some basic concepts for normal (undirected) graphs are defined below [9].

- The two vertices u and v are end vertices of the edge (u, v).
- An edge $e = (u, v)$ is *incident* with the vertices u and v.
- An edge of the form (v,v) is a *loop*.
- A graph with no edges is called *null*.
- A graph with no vertices is called an *empty graph*.
- Edges are adjacent if they share a common end vertex.
- Vertices have degrees, $d(v)$, defined as the number of edges with v as an end vertex.
- A vertex that has no edges is called *isolated*.
- The number of vertices contained in a vertex set is termed the *cardinality* of the vertex set and is denoted by $|.|$.

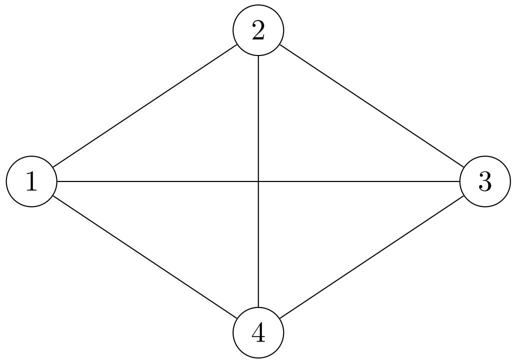

Figure 2-1: An example of an undirected graph with four nodes and six edges.

A sequence of edges that begins from node u and ends at node v is termed a *walk* from u to v, an example of this in Figure 2-1 would be $\{(1,2),(2,4),(4,1),(1,3)\}$ being a walk from 1 to 4. If moreover, a walk from u to v does not pass through any vertex more than once it is called a *path*. By definition a path is always a walk, although the opposite is not always true. Returning to Figure 2-1, a path from 1 to 3 could be $\{(1,2),(2,3)\}$ or $\{(1,2),(2,4),(4,3)\}$ (of different lenghts) but not $\{(1,2),(2,4),(4,1),(1,3)\}$ as node 1 is traversed twice. If a path exists from a node u to a node v, we say these nodes are *connected*. The number of edges in the path that are required in order to reach v from u is the *length* of the path. A path will be written as $p(u,v,k)$ with u being the starting vertex, v the end vertex and k the length of the path, k may be omitted depending on the circumstances. Additionally, if the initial vertex is also the terminal vertex of the path then the path is called a *cycle* or *circuit*. Furthermore, a *cycle family* is a set of cycles that do not have any vertices in common, they are also termed *vertex disjoint cycles*. The number of edges contained in this family is the *length* of the family. Finally, a connected graph is defined by the following property. Any vertex $u \in V(G)$ can be reached by any vertex $v \in V(G)$, i.e. there is a path (sequence of edges) connecting any vertex in the vertex set to any other in the same set. Graphs that contain no circuits are also called a *tree*. A *tree* is a connected forest.

2-2 Directed Weighted Graphs

Directed graphs (or digraphs) are a category of graphs where the edges are ordered. This means that the edges (u,v) and (v,u) are different, they have the opposite orientation. Therefore, for an edge $(u,v) \in E(G)$, while v can be reached from u, the opposite is not also true unless, $(v,u) \in E(G)$. A directed graph is called a *weighted directed graph (or weighted digraph)* if a weight $w(u,v) \in \mathbb{R}$ is associated with all edges $(u,v) \in E(G)$.

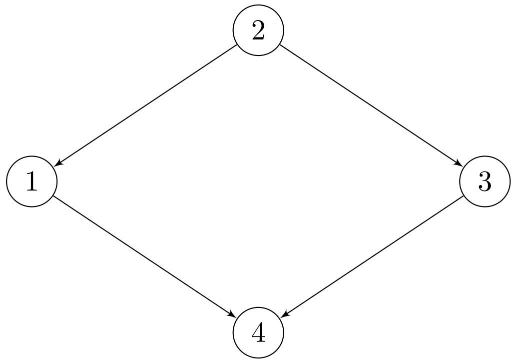

Figure 2-2: An example of a directed graph.

Below, some of the concepts defined above for normal graphs will be extended to directed graphs.

- For the edge $(u, v) \in E(G)$, we call (u, v) an outgoing edge at u and an incoming edge at v.
- Vertices have outgoing and incoming degrees denoted $od(v)$ and $id(v)$ respectively. The outgoing degree of a vertex $v \in V(G)$ is defined as the number of outgoing edges from v, while the incoming degree is defined as the number of incoming edges at v.
- A vertex that has no outgoing edges is called *isolated*.
- Because edges are directed, if a path exists from a vertex u to a vertex v the opposite, unlike undirected graphs, may not be true. A path may not exist from v to u.
- If a path exists from vertex u to vertex v, we say v can be *reached* by u.
- Differently to undirected graphs, two vertices $u, v \in V(G)$ are strongly connected if u can be reached by v and v can be reached by u.
- A cycle is termed *elementary* if, limited to the cycle, each of its vertices has an outgoing and incoming degree equal to one.
- Graph G is strongly connected if every pair of vertices is strongly connected.

Graphs that are not strongly connected can be partitioned into subgraphs that are strongly connected. All the vertices that belong to subgraphs $(V_1, E_1), \ldots, (V_g, E_g)$ that are strongly connected, are also strongly connected. Any subgraph $(V_j, E_j), j \in \{1, 2, \ldots, g\}$, forms a strongly connected graph. Additionally, the partition of a graph into strongly connected subgraphs covers all the vertices of the graph G, isolated vertices form a class of their own (i.e. they are connected only with themselves). Subsequently, according to this partition of the graph into subgraphs, no two vertices that belong to different subgraphs (classes) can be strongly connected. That is, for

$$u, v \in V(G), \quad u \in (V_u), \quad v \in (V_v), \quad V_u, V_v \subset V(G)$$

where V_u, V_v are the node sets of different strongly connected subgraphs, as defined above, then either

$$(u, v) \in E(G) \quad or \quad (v, u) \in E(G) \quad or \quad (u, v), (v, u) \notin E(G)$$

we call the subgraphs that arise from such partitions of a graph *maximally strongly connected subgraphs* (m.s.c.s.).

Furthermore, the notion of the cyclicity of a graph is now defined. The cyclicity of a graph is defined differently for strongly connected graphs and not strongly connected graphs. If a graph is strongly connected then the cyclicity is defined as the greatest common divisor of the lengths of all (elementary) cycles in the graph. On the other hand, if the graph is not strongly connected then the cyclicity of the graph is equal to the least common multiple of the cyclicities of all maximal strongly connected subgraphs.

Following on, we will call the *direct predecessors* of a vertex $u \in V(G)$, the vertices that have outgoing edges that end at u,

$$\pi(u) \quad \stackrel{\text{def}}{=} \quad \{v \in V(G) : (v, u) \in E(G)\}$$

while, *predecessors* will be the term used for the set of vertices that have paths which reach u,

$$\pi^+(u) \quad \stackrel{\text{def}}{=} \quad \{v \in V(G) : p(v, u) \neq \varnothing\}$$

with $p(v, u)$ being the set of paths from v to u.

In a similar way we denote the set for the *direct successors* of vertex $u \in V(G)$,

$$\xi(u) \quad \stackrel{\text{def}}{=} \quad \{v \in V(G) : (u, v) \in E(G)\}$$

and the *successors* as,

$$\xi^+(u) \quad \stackrel{\text{def}}{=} \{v \in V(G) : p(u, v) \neq \varnothing\}$$

The notion of *Menger-type linking's* will now be presented [6]. Consider a directed graph $G = (V, E; U, Y)$ with its vertex set V a union of three distinct vertex sets. More formally, $V = (U \cup X \cup Y)$, where $X = \{x_1, \cdots, x_n\}$, $U = \{u_1, \cdots, u_m\}$ and $Y = \{y_1, \cdots, y_o\}$. In this case it is assumed that there are no incoming edges for any vertex $u_i \in U$ or any outgoing edges for a vertices $y_i \in Y$. The vertex set U will be called *entrance* and the set Y *exit*. A *Menger-type linking* from U to Y is a set of pairwise vertex-disjoint directed paths from a vertex in U to a vertex in Y. The amount of directed paths that are included in this linking are the size of the linking. If the size of the linking is maximal then it is a *maximum linking* furthermore, if $|U| = |Y|$, a linking of size $|U|$ is called a *complete linking*. Finally, a *separator* of (U, Y) is such a subset of V that intersects any directed path from a vertex in U to a vertex in Y.

2-3 Bipartite Graphs

A bipartite graph, $G_b = (V^+, V^-; E)$ is a graph consisting of two disjoint vertex sets V^+ and V^- with edge set E. In a bipartite graph, vertices belonging to the same vertex set can not be adjacent, i.e. vertices in edge set V^+ can only be adjacent to vertices in V^- and vice versa. A subset of edges is a *matching* if any two edges of the subset do not have a common vertex. The number of edges is named *cardinality* of the matching while, a matching with maximal cardinality is a *maximum matching*. The size of a maximum matching is denoted by $m(G)$. If the maximum matching includes all the vertices of both sets (V^+ and V^-) then it is called a *perfect matching*. For a perfect matching to exist the two vertex sets must contain the same number of vertices.

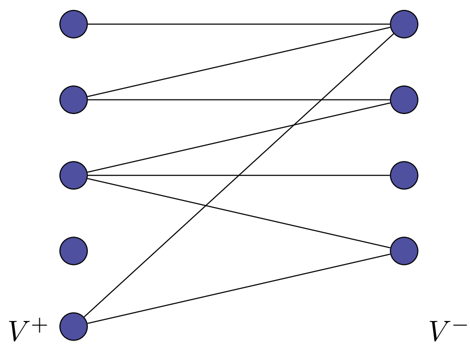

Figure 2-3: An example of a bipartite graph.

Bipartite graphs are of interest as any matrix can be represented as a bipartite graph. By defining $Col(A) = V^+$, $Row(A) = V^-$, for $A \in \mathbb{R}^{n\times n}$ and $E = ((j,i)|A_{ij} \neq 0)$. This is of interest due to the fact that algebraic properties of a matrix can be directly deducted from the bipartite representation of it. Murota [6] proves that the term-rank of a matrix is equal to the maximum matching of its bipartite graph representation. The matching on the bipartite graph of a matrix shows the position of elements in a matrix. If the bipartite graph has a perfect matching then no row or columns is equal to zero. In particular the rank of matrix where only the position of the elements in the matrix is considered and not their values, is termed the term-rank of a matrix. Note that, by definition the term-rank is always greater or equal to the generic-rank of a matrix.

2-4 Signal-Flow Graph

The *signal-flow graph* is the graphical representation of the state-space of systems that have the following form

$$\dot{x} = Ax + Bu, \tag{2-1}$$
$$y = Cx \tag{2-2}$$

where $A \in \mathbb{R}^{n \times n}$ is the state matrix, $B \in \mathbb{R}^{n \times m}$ the input matrix and $C \in \mathbb{R}^{o \times n}$ the output matrix.

The signal-flow graph provides an efficient way to model the structure of the system and to derive properties such as structural controllability and structural observability. We will denote by $G(A, B, C)$ the signal-flow graph that represents all of the state space and by $G(A, B)$ we will denote the signal-flow graph that models only Equation (2-1).

The signal-flow graph $G(A, B, C)$ is a directed graph with vertex set $V(G) = V_A \cup V_B \cup V_C$ and edge set $E(G) = E_A \cup E_B \cup E_C$. The vertex set is partitioned into three distinct subsets, the *state vertices* corresponding to the the states of the system $V_A = \{v_1, \cdots, v_n\}$, the *input vertices* that correspond to the inputs of the system $V_B = \{u_1, \cdots, u_m\}$ and the *output vertices* corresponding to the outputs of the system $V_C = \{y_1, \cdots, y_o\}$. The edges set, is partitioned in a similar way with $E_A = \{(x_j, x_i) | a_{ij} \neq 0\}$ being the set of edges between state vertices, $E_B = \{(u_j, x_i) | b_{ij} \neq 0\}$ being the set of edges from input vertices to state vertices and $E_C = \{(x_j, y_i) | c_{ij} \neq 0\}$ being the set of edges from state vertices to output vertices. It is evident that no edges exist from state vertices to input vertices, from input vertices to output vertices or from output vertices to any other vertex set. The signal-flow graph representation $G(A, B)$ of Equation (2-1) omits the output vertices V_C, in addition to edge set E_C.

Graph-theoretical properties of the signal-flow graph that will play a vital role in the next chapter will now be discussed. We will call a state vertex x_j in the signal-flow graph $G(A, B)$ *reachable* if it can be reached via a directed path originating from an input vertex. If no such path exists then the vertex will be termed *unreachable*. Furthermore, a directed path that begins from an input vertex will be called a *stem*. The first vertex of a stem is the *root*, while the final vertex is the *top* of the stem. A *bud* is an elementary cycle with an additional edge (i, j) where j is a vertex of the cycle and i is not. The added edge (i, j) is named the *distinguished edge of the bud*. Should a graph be spanned by a specific arrangement of buds and stems, then such a graph can be called a *cactus*. The formal definition of a *cactus* as given in [5] follows

Definition 2.1. *A cactus is a subgraph defined successively in the following way. A stem is a cactus. Given a stem S_0 and buds $B_1, B_2, \cdots, B_\beta$, then $S_0 \cup B_1 \cup B_2 \cup \cdots \cup B_\beta$ is a cactus if for every j $(1 \leq j \leq \beta)$ the first vertex of the distinguished edge of B_j is not the top of S_0 and is the only vertex belonging to both B_j and $S_0 \cup B_1 \cup B_2 \cup \cdots \cup B_{i-1}$. We will say a graph is spanned by a cactus if all vertices are part of the cactus. A set of disjoint cactuses is termed a cacti.*

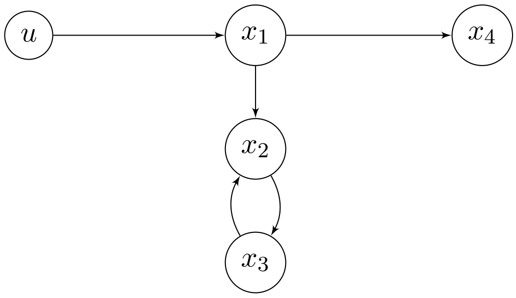

Figure 2-4: An example of a signal-flow graph that is spanned by a cactus.

Figure 2-4 depicts a signal flow graph that is spanned by a cactus. The corresponding matrices A and B for this signal-flow graph are given below.

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 \\ x_{21} & 0 & x_{23} & 0 \\ 0 & x_{32} & 0 & 0 \\ x_{41} & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} u_1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{2-3}$$

We will see in the next chapter that if the signal-flow graph of a system that has the form of (2-1) is spanned by a cacti, then we can infer a specific property (structural controllability) of this system.

2-5 Dynamic Graph

The *dynamic graph* is the graphical representation of the evolution of a discrete-time system over a certain period. Discrete-time systems are systems whose evolution depends on events that may or may not have an equal timing between them. The dynamic graph was first introduced by Kazuo Murota [6, 10] as an alternative graphical representation to the signal-flow graph for linear control systems. Because the dynamic graph models the system over a period of time, it also allows for modeling potential changes in the structure of the system. This property differentiates the dynamic graph representation from the signal-flow graph representation, as it is possible to model potential changes in the structure of the system and thus, provides more flexibility. Thus, the dynamic graph can be used to model both systems with fixed structure and switching systems (will be introduced in Chapter 5). More formally, the dynamic graph is defined as follows.

For $k \geq 1$ the dynamic graph of time-span k for a state-space system is defined to be $G_0^k = (X_0^k \cup U_1^k, E_0^{k-1})$ with,

$$X_0^k = \cup_{t=0}^k X^t, \quad X^t = \{x_i^t | i = 1, ..., n\}(t = 0, 1, ..., k),$$

$$U_1^k = \cup_1^k U^t, \quad U^t = \{u_j^t | j = 1, ..., m\}(t = 1, ..., k),$$
$$E_0^{k-1} = \{(x_j^t, x_i^{t+1}) | A_{ij}^t \neq 0; t = 0, 1, ..., k-1\} \cup \{(u_j^t, x_i^{t+1}) | B_{ij}^t \neq 0; t = 1, ..., k\}$$

The state and input matrices A and B respectively, are denoted as A^t and B^t for the purpose of also including systems whose structure may change over time. Examples of such systems are switching systems or systems where the stochasticity is associated with the structure of the system (i.e. the edges of the signal flow graph and not the weights).

An example of a dynamic graph for a time span of $k = 3$ can be seen in Figure 2-5 for the matrices A and B of (2-3). The edges that exist between the state vertices (denoted by x) are the entries of the A matrix while the edges originating at an input vertex (denoted by u) are the edges associated with the input matrix B.

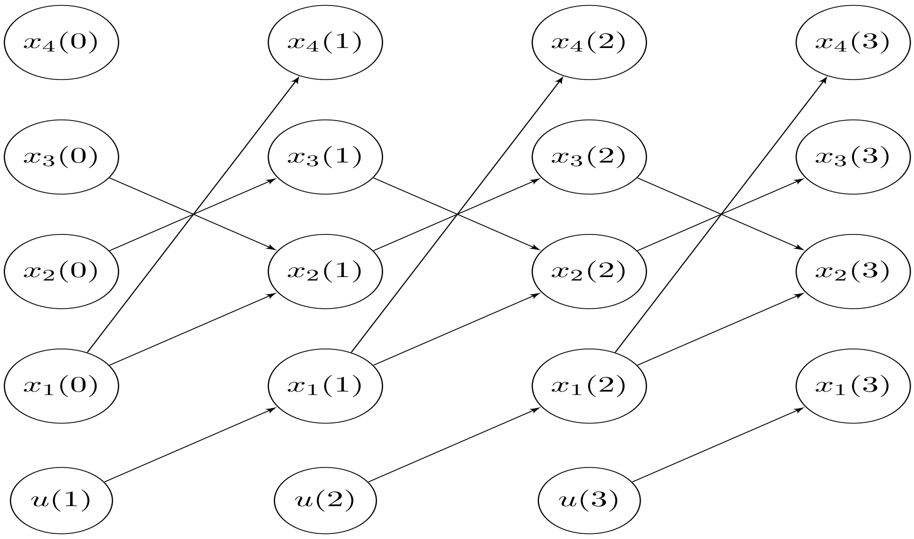

Figure 2-5: A dynamic graph for the A and B matrices of (2-3).

In the next chapter we will see how the dynamic graph can be used to derive structural properties of a linear system.

2-5-1 Coloured Dynamic Graph

The *coloured dynamic graph* is a dynamic graph in which vertices are associated with one or more colours. By a *colour* we mean a subset of the vertex set which contains vertices that share a common property. For the case of this thesis, this property will be whether vertices of the dynamic graph are part of a path originating from an input vertex.

Definition 2.2. *A colour of a dynamic graph, denoted* $col_{i,k}, i, k \in \mathbb{N}^+$*, is a subset of the state vertex set of the dynamic graph that is,* $col_i \subseteq X^k$*. Each colour,* $col_{i,k}$*, is a set that contains all the state vertices at time event* k *that can be reached by a path originating from an input from time event* i*. Any vertex can be associated with more than one colour or with no colours. More formally,*

$$col_{i,k} = \{x_j(k) \in col_{i,k} \quad | \quad \mathcal{P}(u_i, x_j) \neq \emptyset, \quad j \in \{1, ..., n\}, \quad i \in \{1, ..., k\}\} \tag{2-4}$$

To be more specific, take the coloured dynamic graph depicted in Figure 2-6. This figure depicts a dynamic graph of a switching system. In a switching system the structure of the system may change over time. We will associate with the set $col_{1,2}$ all the (state) vertices at time event 2 that can be reached by a path(s) originating from an input vertex (or vertices generally) at time event 1. For this that would be $\{x_1\}$; highlighted in orange. We can also define more colours, like $col_{2,2}$ for vertices that belong to a path originating from the input at time event 2, this set would be $\{x_2\}$; highlighted in blue. Coloured dynamic graphs will prove useful for the modeling of SMPL systems. The concept will become more clear in chapters 5 and 6.

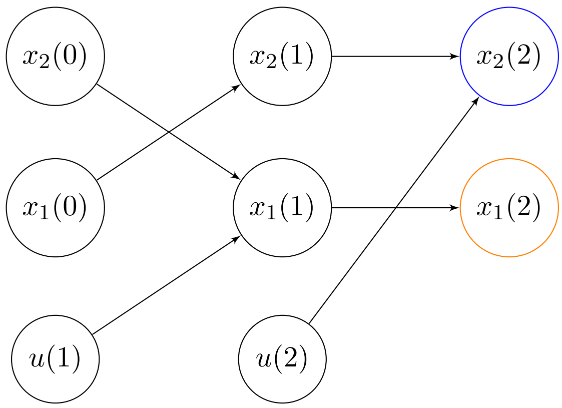

Figure 2-6: An example of a coloured dynamic graph.

Chapter 3

Graph Theory in Linear System Theory

Graph theory has been widely applied for modelling different systems. What is of particular interest however, is the use of a graph-theoretic approach for the derivation of properties in linear systems, more specifically, of structural controllability and observability. This provides sufficient motivation for approaching similar problems encountered in the max-plus semiring (to be defined in Chapter 4) in a similar way. In this chapter the graph-theoretic approach to linear control systems will be presented. In Section 3-1 a brief overview of linear systems and the property of controllability will be given, Section 3-2 will overview the notion of structural controllability as given by Lin [4] and in the end, in Section 3-3, similar definitions of controllability will be presented as given by Murota [6]. Although structural controllability is the main subject of this Chapter, these notions can be further extended to observability through the duality theorem.

3-1 Introduction

Consider a linear time-invariant (LTI) system described by the following equations

$$\dot{x} = Ax + Bu, \quad x(0) = x_0, \tag{3-1}$$

$$y = Cx, \tag{3-2}$$

where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$ and $C \in \mathbb{R}^{o \times n}$.

Controllability of a system refers to the property of the system to steer any initial state $x_0 \in \mathbb{R}^n$ at time zero to any final state $x_f \in \mathbb{R}^n$ in a finite amount of time $T > 0$.

The reachable set of a system R_T at time $T > 0$ is the set of all states $x(T)$ that can be reached from initial state zero by any control input. The system is said to be controllable if the reachable set is equal to the set of real numbers, that is $R_T = \mathbb{R}^n$. Obviously, controllability is a very fundamental property of a system as it shows whether a system can be brought to a desirable state within a finite amount of time. Moreover, Kalman [11] showed that a system defined by (A, B) is controllable if and only if the controllability matrix

$$K = \begin{pmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{pmatrix} \tag{3-3}$$

has full row rank. Showing that matrix K has full row rank is equivalent to showing that the range of the matrix is equal to $\mathbb{R}^n$. If this is the case, one can steer any initial state $x_0 \in \mathbb{R}^n$ at time 0 to any final state $x_f \in \mathbb{R}^n$ at time $T > 0$.

3-2 Structural Controllability

Structural controllability is a graph-theoretic concept first introduced by Lin [4] in 1974 for LTI systems with one input. It has since been extended to systems with multiple inputs [8]. Even though it is a graph-theoretic concept it has clear algebraic implications. To begin with, for structural controllability, the system matrices (A, B), as defined in Equation (3-1), are considered to be structured matrices. What is meant by this is that the elements of these matrices are either taken as independent parameters over the field of real numbers or they are set to zero. In other words, this means we know the structure of the system (i.e. the elements which are equal to zero). The physical meaning of this can be interpreted in the following way, we know whether connections between different parameters exist, even if they cannot be precisely measured and we also have knowledge of the absence of connection between other parameters. In reality this is often the case for well defined physical systems.

If a system is considered to be structurally controllable then it is possible to select the independent parameters (A, B) in such a way that the system is considered controllable in the traditional sense (Equation 3-3). Structural controllability is a generic property of the system and most often implies controllability. A potential loss of controllability can only materialize in rare circumstances - such a case can arise when the parameters of the system are not independently defined [5]. Two examples borrowed from this paper will further illustrate the notion of structural controllability.

Example 3.1. Consider the system

$$\begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \\ \dot{x_3}(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ \alpha_{21} & 0 & 0 \\ 0 & \alpha_{32} & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} b_1 \\ 0 \\ 0 \end{bmatrix} u(t)$$

the controllability matrix is

$$K = [B \quad AB \quad A^2B] = b_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha_{21} & 0 \\ 0 & 0 & \alpha_{32}\alpha_{21} \end{bmatrix}$$

has $rank(K) = 3 = n$ and the system is controllable. This will always be the case if the weights α_{21}, α_{32} and b_1 are non-zero. Hence, controllability is invariant under different values for the elements as long as they are not zero.

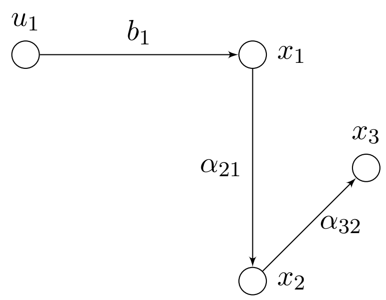

Figure 3-1: The signal flow graph of Example 3.2.1.

□

Example 3.2. For the second example consider the system

$$\begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \\ \dot{x_3}(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ \alpha_{21} & 0 & 0 \\ \alpha_{31} & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} b_1 \\ 0 \\ 0 \end{bmatrix} u(t)$$

the controllability matrix is

$$K = [B \quad AB \quad A^2B] = b_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha_{21} & 0 \\ 0 & \alpha_{31} & 0 \end{bmatrix}$$

and has $rank(K) = 2 < n$. This shows the system is uncontrollable regardless of the values of α_{21}, α_{31} and b_1. □

Now that the notion of structural controllability has been explained, the formal theorem for establishing structural controllability of a system is given below.

Theorem 3.1. *(Lin's Structural Controllability Theorem)*
The following three statements are equivalent:

1. *An LTI system (A, B) is structurally controllable.*

2. (a) *The signal-flow graph $G(A, B)$ has no unreachable vertices.*
   (b) *The signal-flow graph $G(A, B)$ is spanned by disjoint cycles and stems in such a way that all vertices of the graph belong to either a stem or a cycle.*
3. *The signal-flow graph $G(A, B)$ is spanned by a cacti.*

The first condition ($2a$) that the signal-flow graph has no unreachable vertices has a straightforward interpretation. If a vertex is not accessible from an input vertex then it cannot be influenced by the input and as a result it cannot be controlled. The second condition $2(b)$ that the signal-flow graph is spanned by disjoint cycles and stems guarantees that the structured matrix $[A; B], \quad A \in \mathbb{R}^{n\times n}, \quad B \in \mathbb{R}^{n\times m}$ has generic rank equal to n. By generic rank of a structured matrix we mean the maximum rank that can be attained for this matrix as a function of its independent parameters.

3-3 Structural Controllability in terms of the Dynamic Graph

Murota in his book [6] establishes a connection between the structural controllability of discrete LTI systems and the number of Menger-type linkings contained in the dynamic graph of the system. The discrete version of the LTI system of (3-1) is given by

$$\dot{x}(k) = Ax(k) + Bu(k), \quad x(0) = x_0, \tag{3-4}$$

$$y(k) = Cx(k),$$

And the connection established between Menger-type linkings and structural controllability is given by the following Theorem.

Theorem 3.2. *A discrete system in the standard form (3-4) is structurally controllable if and only if there exists in the dynamic graph G_0^n of time-span n a Menger-type vertex-disjoint linking of size n from U_0^{n-1} to X^n.*

A direct consequence of this theorem is that the controllability matrix of the system has full generic rank. If it did not the system would not be structurally controllable. Another important point of interest is the fact that if the controllability matrix has full generic rank it also has full term-rank [6]. Recall from Section 2-3 that the term-rank of a matrix is equal to the maximum matching of its bipartite representation. It is important to note that the opposite is not true, i.e. if a system could have full term-rank this does not guarantee that it would also have full generic-rank.

Being able to establish structural controllability by means of the dynamic graph is of particular interest in the scope of this thesis. This is due to the fact that the modelling framework of dynamic graphs makes it feasible to model potential changes in the structure of the system over time. As will be seen in Chapter 5 this is exactly the case for Switching Max-Plus Linear systems where the structure of the system changes over time making it impossible to model the system or derive its structural properties through the signal-flow graph.

Chapter 4

Max-Plus Algebra

Discrete event systems (DESs) in which there is synchronisation but no concurrency or choice are traditionally nonlinear in conventional algebra. However, these systems can be described by models that are linear in max-plus algebra. Max-plus algebra is an algebra with maximisation and addition as its basic operations over the idempotent semiring $\mathbb{R}_{max}$, that is, the union of real numbers with minus infinity. The chapter is mainly based on [12]. Section 4-1 goes over the basic definitions and notions related to max-plus algebra succeeding this is Section 4-2 which gives an overview of spectral theory for max-plus algebra. Section 4-3 introduces the notion of the asymptotic growth rate and the cycle-time vector while, Section 4-4 provides an overview of vector spaces for max-plus algebra in addition to developing a method for determining the span of max-plus and min-plus matrices.

4-1 Basic Definitions

Max-plus algebra has been developed for the description and evaluation of discrete-event systems. It is the semi-ring over the union of real numbers and minus infinity. More formally, let $\varepsilon \overset{\text{def}}{=} -\infty$, $e \overset{\text{def}}{=} 0$ and $\mathbb{R}_{max} \overset{\text{def}}{=} \mathbb{R} \cup \{\varepsilon\}$. For $a, b \in \mathbb{R}_{max}$ the max-plus addition $\oplus$ and multiplication $\otimes$ are defined as:

$$a \oplus b \overset{\text{def}}{=} \max(a, b) \qquad \text{and} \qquad a \otimes b \overset{\text{def}}{=} a + b \tag{4-1}$$

For any $a \in \mathbb{R}_{max}$

$$\tag{4-2}$$

$$a \oplus \varepsilon = \varepsilon \oplus a = a \qquad \text{and} \qquad a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon \tag{4-3}$$

Powers in max-plus algebra are similar to conventional algebra. For $a \in \mathbb{R}_{max}$ and $n \in \mathbb{N}$:

$$x^{\otimes n} \overset{\text{def}}{=} x \otimes x \otimes ... \otimes x, n \geq 1$$

Which in conventional algebra is

$$x^{\otimes k} = \underbrace{x + x + ... + x}_{k\ times} = k \times x$$

.
The aforementioned concepts can be further extended to matrices. For any $A \in \mathbb{R}_{max}^{n\times m}$ and $B \in \mathbb{R}_{max}^{n\times m}$ we have:

$$[A \oplus B]_{ij} = a_{ij} \oplus b_{ij} = \max(a_{ij}, b_{ij})$$

While, for matrices $C \in \mathbb{R}_{max}^{n\times p}$ and $D \in \mathbb{R}_{max}^{p\times m}$ the matrix product is defined as

$$[C \otimes D]_{ik} = \bigoplus_{j=1}^{p} c_{ij} \otimes d_{jk} = \max_{1\leq j\leq p} (c_{ij} + d_{jk})$$

For $A \in \mathbb{R}_{max}^{n\times n}$, the matrix powers are defined by

$$A^{\otimes k} = \underbrace{A \otimes A \otimes ... \otimes A}_{k\ times}$$

In addition to this, a matrix is called regular if it has at least one entry different than ε in every row. Finally, the max-plus-algebraic zero matrix $\mathcal{E}$ is defined as $[\mathcal{E}]_{i,j} = \varepsilon$, while the max-plus-algebraic identity matrix is

$$[E_{ij}] = \begin{cases} e & i = j \\ \varepsilon & i \neq j \end{cases}$$

Many algebraic properties of conventional algebra also extend to the premises of max-plus algebra. The main difference is the fact that, in contrast to conventional algebra, max-plus algebra is idempotent. This and the other algebraic properties of max-plus algebra are presented in the following list.

- Associativity:

$$\forall a, b, c \in \mathbb{R}_{max} : a \oplus (b \oplus c) = (a \oplus b) \oplus c$$
$$\forall a, b, c \in \mathbb{R}_{max} : a \otimes (b \otimes c) = (a \otimes b) \otimes c$$

- Commutativity:

$$\forall a, b \in \mathbb{R}_{max} : a \oplus b = b \oplus a \qquad \text{and} \qquad a \otimes b = b \otimes a$$

- Distributivity of $\otimes$ and $\oplus$:

$$\forall a, b, c \in \mathbb{R}_{max} : a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$$

- Presence of a zero element:

$$\forall a \in \mathbb{R}_{max} : a \oplus \varepsilon = \varepsilon \oplus a = a$$

- Presence of a unit element:

$$\forall a \in \mathbb{R}_{max} : a \otimes e = e \otimes a = a$$

- Zero is absorbing for $\otimes$:

$$\forall a \in \mathbb{R}_{max} : a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon$$

- Idempotency:

$$\forall a \in \mathbb{R}_{max} : a \oplus a = a$$

Max-plus algebra is not the only idempotent algebra that exists. Min-plus , max-times and min-max, to name a few, are some of the most prominent idempotent algebras, other than max-plus. In min-plus algebra the max operator is substituted with the min operator for addition. Min-plus algebra and max-plus algebra are isomorphic. Due to the two algebras being isomorphic, all notions, theorems and lemmas of max-plus algebra can be extended to min-plus algebra. The min-plus algebra is a semiring over the union of real numbers and infinity, $\mathbb{R}_{min} = \mathbb{R} \cup \{\infty\}$. The neutral element is now $\varepsilon^- = \infty$ while the identity element remains the same as in max-plus algebra $e = 0$. By convention, $\varepsilon \otimes \varepsilon^- = \varepsilon$ and $\varepsilon^- \otimes \varepsilon = \varepsilon^-$. For $a, b \in \mathbb{R}_{min}$ min-plus addition $\oplus'$ and multiplication $\otimes'$ are defined as:

$$a \oplus' b \stackrel{\text{def}}{=} \min(a, b) \qquad \text{and} \qquad a \otimes' b \stackrel{\text{def}}{=} a + b \tag{4-4}$$

Multiplications and addition can be extended to vectors and matrices in a similar manner as in max-plus. Lastly the set $\bar{\mathbb{R}}$, will denote the union of real numbers with infinity and minus infinity. That is, $\bar{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$.

4-2 Spectral Theory

Any square matrix $A \in \mathbb{R}_{max}^{n \times n}$ can also be represented as a weighted directed graph, named the *communication graph* and defined as $G(A) = (V(A), E(A))$ where $V(A) \stackrel{\text{def}}{=} \{1, ..., n\}$ is the set of vertices (i.e. the number of vertices is equal to the order of the matrix A) and $E(A) \overset{\text{def}}{=} \{u, v \in V(A), (u, v) \in E(A) | a_{vu} \neq \varepsilon\}$ is the set of edges where each edge is associated with a weight equal to the value of that parameter in the matrix A (i.e. if $a_{vu} = 2$ then there exists an edge from u to v with weight 2). This is of particular interest as a lot of underlying properties of the matrix A can be revealed through its graph structure.

Such a property is whether the matrix $A \in \mathbb{R}_{max}^{n \times n}$ is reducible or irreducible. If the communication graph of the matrix is strongly connected then matrix A is said to be irreducible. On the other hand, if not, then it is called reducible. As a result an irreducible matrix is a strongly connected graph and all vertices in this graph belong to the same m.s.c.s. However, if A is reducible then it contains more than one m.s.c.s., and the matrix can be written, after a possible relabeling of the vertices, in its *normal form* with respect to its m.s.c.s.'s.

$$\begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1p} \\ \mathcal{E} & A_{22} & \cdots & A_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{E} & \mathcal{E} & \cdots & A_{pp} \end{pmatrix} \tag{4-5}$$

Here each diagonal block matrix A_{jj} corresponds to the vertices of the same m.s.c.s. and p is the number of m.s.c.s.'s contained in the communication graph of A. Since blocks A_{jj} correspond to a m.s.c.s. all diagonal blocks are also irreducible. The remaining matrices A_{ij} correspond to edges that connect vertices that belong to the m.s.c.s. $[j]$, $[j]$ denoting the j^{th} m.s.c.s., to vertices that belong to $[i]$.

Another close relation exists between the powers of A and its communication graph $G(A)$. More precisely, elements of the k^{th} power of A yield the maximal weight of a path of length k, should that exist, between the vertices of those elements. That is, if the element $[A^{\otimes k}]_{ji}$ is not ε, then this element represents the maximal weight of a path of length k from vertex i to vertex j.

In accordance to the preceding property, for $A \in \mathbb{R}_{max}^{n \times n}$, let

$$A^{+} \overset{\text{def}}{=} \bigoplus_{k=1}^{\infty} A^{\otimes k} \tag{4-6}$$

where any element $[A^{+}]_{ij}$ represents the maximal weight of any path from j to i.

Furthermore, one of the most important properties that can be derived from graphs is the existence of an eigenvalue if a circuit exists in the communication graph of the matrix. The main theorems and lemmas will be introduced below, for proofs and more information, the reader is referred to [12]. Firstly, the definition of an eigenvalue and eigenvector will be given in max-plus algebra and secondly, a lemma to show that average circuit weights are potential eigenvalue(s).

Definition 4.1. *Let $A \in \mathbb{R}_{max}^{n \times n}$ be a square matrix. If $\mu \in \mathbb{R}_{max}$ is a scalar and $v \in \mathbb{R}_{max}^{n}$ is a vector that contains at least one finite element such that*

$$A \otimes v = \mu \otimes v$$

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

Lemma 4.2. *Let $A \in \mathbb{R}^{n\times n}_{max}$ have finite eigenvalue μ. Then, a circuit γ exists in G(A) such that*

$$\mu = \frac{|\gamma|_w}{|\gamma|_l}$$

where, $|\gamma|_w$ is the sum of weights of the circuit and $|\gamma|_l$ is the length of the circuit.

According to this lemma possible eigenvalues can be found from the average weight of circuits. However, this lemma does not provide any further information on which circuit(s) weight(s) are equal to the eigenvalue. With this in mind the following concepts are defined.

A *critical circuit* is defined as the circuit that has the maximal average weight. Consequently, the *critical graph*, designated by $G^c(A) = (V^c(A), E^c(A))$ of a matrix (A), is the graph consisting of those vertices and edges that belong to the critical circuits of $G(A)$. By combining these two concepts and the previous lemma, the following lemma can be proved.

Lemma 4.3. *Let $A \in \mathbb{R}^{n\times n}_{max}$ have finite maximal average circuit weight λ. Then, λ is an eigenvalue of A and for any vertex v in $G^c(A)$ it holds that $[A^*_\lambda]_{.\eta}$ is an eigenvector associated with λ.*

Where,

$$[A_\lambda]_{uv} = a_{uv} - \lambda \tag{4-7}$$

$$A^*_\lambda \overset{\text{def}}{=} E \oplus A^+_\lambda = \bigoplus_{k\geq 0} A^{\otimes k}_\lambda \leftrightarrow [A^*_\lambda]_{.\eta} = [E \oplus A^+_\lambda]_{.v} \tag{4-8}$$

A_λ is also called the *normalized matrix*, while the operator (*) is called the *Kleene star*. The *Kleene star* of a matrix exists only if all cycles of the communication graph of the matrix have non-positive weights.

The above lemma is of great importance as it establishes the existence of an eigenvalue and its corresponding eigenvector, provided that the maximal average circuit weight exists and is finite. Moreover, irreducibility of A already establishes that the maximal average circuit weight is finite. Consequently, the eigenvalue of irreducible matrices always exists and is also unique. Finally, we call the set of vertices of $G(A)$ that correspond to finite entries of v (the eigenvector) the *support* of v. In general the support of a vector, corresponds to its the indices of its elements that are finite.

The aforementioned definitions and lemmas show the important correlation that exists between max-plus algebra and graph theory. Underlying structural properties of a matrix such as the eigenvalue and eigenvector can be directly determined from the communication of the matrix. In the following section a new concept will be introduced and its relationship to graph theory will be analysed.

4-3 Asymptotic Growth Rate and Cycle-Time Vector

The asymptotic growth rate or cycle-time vector of $x(k)$ is the quantitative asymptotic behaviour of $x(k)$, when generated by a sequence such as

$$x(k+1) = A \otimes x(k) \tag{4-9}$$

for all $k \geq 0, A \in \mathbb{R}^{n \times n}_{max}$ and $x(0) = x_0 \in \mathbb{R}^n_{max}$. It is formally defined as

Definition 4.4. *Let $\{x(k) : k \in \mathbb{N}\}$ be a sequence in $\mathbb{R}^n_{max}$, and assume that for all $j \in V(A)$ the quantity η_j, defined by*

$$\lim_{k \to \infty} \frac{x_j(k)}{k}$$

exists. The vector $\eta = (\eta_1, \eta_2, \cdots, \eta_n)^T$ is called the cycle time vector of the sequence x(k). If all η_j's have the same value, this value is also called the asymptotic growth rate of the sequence x(k).

The asymptotic growth rate of each state (or the cycle-time vector of the whole system) describes the limiting behaviour of each state (or the whole system). What is of most importance is to quantify the effect of the initial condition on the evolution of $x(k)$ and what this evolution is in the case when A is reducible and as a result there is no common eigenvalue for all states. The dependency of the asymptotic growth rate on the initial condition is examined in Theorem 4.5.

Theorem 4.5. *Consider the recurrence relation $x(k+1) = A \otimes x(k)$ for $k \geq 0$, with $A \in \mathbb{R}^{n \times n}_{max}$ a square regular matrix and x_0 as initial condition. If $x_0 \in R^n$ is a particular initial condition such that the asymptotic growth rate exists, then the asymptotic growth rate exists and has the same value for any initial condition $y \in \mathbb{R}^n$*

As a result, should the limit exist, its value has no dependence on the initial condition, regardless of whether A is irreducible or not. This shows that after a finite period of time the effect of the initial condition will fade away and the evolution of the states can be completely described by the eigenvector of A. In the ensuing lemma the conditions for the existence of the asymptotic growth rate for irreducible matrices are established.

Lemma 4.6. *Consider the recurrence relation $x(k+1) = A \otimes x(k)$ for $k \geq 0$ and $A \in \mathbb{R}^{n \times n}_{max}$ a square irreducible matrix with eigenvalue $\lambda \in \mathbb{R}$. Then*

$$\lim_{k \to \infty} \frac{x_v(k)}{k} = \lambda,$$

for any initial condition $x(0) = x_0 \in \mathbb{R}^n$ and for all $v \in V(A)$.

A direct consequence of the above lemma is the fact that should the eigenvalue of an irreducible matrix exist and be finite,then it is equal to the asymptotic growth rate, something that can be established and calculated from the communication graph of A. The asymptotic growth rate may also be referred to in literature as max-plus exponent or Lyapunov exponent.

With the purpose of showing that the cycle-time vector exists for regular reducible matrices, the normal form of A is considered as shown in Equation 4-5. Then let $x(k)$ be partitioned according to the normal form of A

$$\begin{pmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_p(k) \end{pmatrix}$$

where each $x_j(k)$ for $j \in p$ is a vector of appropriate size. Having partitioned the state vector in this way, it allows for the evolution of each state of Equation 4-9 to be described by the following recurrence relation

$$x_j(k+1) = A_{jj} \otimes x_j(k) \oplus \bigoplus_{i=j+1}^{p} A_{ji} \otimes x_j(k) \tag{4-10}$$

Each block matrix A_{jj} is either irreducible and describes a m.s.c.s. of A or it is equal to the max-plus zero matrix.

The asymptotic growth rate of irreducible matrices has been defined and proved to exist, it follows that each m.s.c.s. of a graph, since it is irreducible, also has an asymptotic growth rate. Furthermore, all the vertices that belong to the same m.s.c.s. have the same asymptotic growth rate. The asymptotic growth rate of a m.s.c.s. can be obtained by finding the maximum eigenvalue among that m.s.c.s. and the preceding m.s.c.s. of the graph (i.e. the m.s.c.s.'s for which (a) path(s) exists that reach the m.s.c.s. in question).

The cycle-time vector then is the vector that contains the asymptotic growth rates of all states. In other words, in a regular reducible matrix each state evolves with respect to the m.s.c.s. and as a result not all states evolve with the same rate. As can be deduced, it is possible through the communication graph of A, $G(A)$, to analyse and evaluate the limiting behaviour of each state as the irreducibility, eigenvalue and cycle-time vector (or asymptotic growth rate in the case when A is irreducible) can be determined directly from it.

The cycle-time vector is one of the most important properties for control of max-plus linear (MPL) or switching max-plus linear (SMPL) systems and its importance will be shown in the next chapter when MPL and SMPL systems will be presented.

4-4 Max-Plus Geometry

In this section, an overview of the basic notions of max-plus geometry will be presented. Following this, a method will be introduced with the objective of adequately representing the span of max-plus matrices. As will be shown, the span of max-plus matrices can be represented as the union of polyhedral sets. This method will be further used in Chapter 6 with the intention of accurately determining the set of accessible states for MPL systems.

Due to the lack of an opposite operation to max-plus addition and the fact that max-plus algebra is idempotent, vector spaces cannot be defined in a similar way to conventional algebra. The subspaces created by a vector, or the addition of two or more vectors are referred to as *semimodules* or *max-plus convex cones* or *max-plus polyhedra*. Both terms (semimodules and convex cones) will be used interchangeably for the internal representation, while the term max-plus polyhedra will be used for the external representation. What internal and external representations are will be explained in due course and once some preliminaries have first been introduced. For now the terms convex cones and semimodules will be used interchangeably. They are the equivalent of vector spaces for max-plus algebra.

Consider a subset $S \subseteq \mathbb{R}^n_{max}$. Then this subset is a max-plus convex cone if

$$\alpha \otimes u \oplus \beta \otimes v \in S$$

$\forall u, v \in S$ and $\alpha, \beta \in \mathbb{R}$. A max-plus convex cone (hereafter, we will omit "max-plus" as a convex cone will always refer to a max-plus convex cone) is *finitely generated* if for any $x \in S$, x can be expressed as $x = \bigoplus_{i=1}^{s} \alpha_i \otimes v_i$. The set of vectors $\{v_1, \dots, v_s\}$ is the *generating set* of S and $\alpha_1, \dots, \alpha_s \in \mathbb{R}_{max}$. A generating set is called *minimal* if any vector v_i belonging to the generating set cannot be expressed as a linear combination of the other generators, that is for some $\alpha_j, j \neq i$

$$v_i \neq \bigoplus_{j \neq i} \alpha_j \otimes v_j$$

The set of minimal generators is also called *bases.* Vectors that are not a linear combination of each other are called *(weakly) independent.* All vectors contained in a minimal generating set are therefore independent.

The set of all max-plus linear combinations of a set of vectors V will be denoted as $span(V)$. Furthermore, if $span(V) = S$ then V is a generating set of S. It is therefore evident that the column span of a max-plus matrix $V \in \mathbb{R}^{n \times m}_{max}$ (this is also true for square matrices) is a convex cone, if all the vectors are linearly dependent then the cone generated is just the scalar multiples of these vectors. The generators of a convex cone are unique up to a scalar multiplication (in the max-plus sense) of the vectors or a reordering of the vectors [13]. The minimal number of generating vectors for a convex cone is called the *dimension* of the cone [13]. A method for determining whether the columns of a matrix or the vectors of a generating set are linearly independent, and thus the dimension of the cone, was given by [14] and is presented below in the form of a theorem.

Theorem 4.7. *Let $V \in \overline{\mathbb{R}}^{n\times m}$ ($\overline{\mathbb{R}}$ defined in page 21) be a matrix with columns $(v_{.1}, \ldots, v_{.m})$ and $\mathcal{V}$ be the matrix produced by $-V^T \otimes' V$ after changing the diagonal elements to ε. Then for all $i \in [1, \ldots, m]$ the column $v_{.i}$ is equal to the i^{th} column of $V \otimes \mathcal{V}$ if and only if $v_{.i}$ is a combination (in the max-plus sense) of other columns of V. The elements of the i^{th} column of $\mathcal{V}$ then provide the coefficients to express the max-combination.*

The advantage presented by Theorem 4.7 is that it allows us to determine the minimal generating set of the cone generated by the columns of any matrix $V \in \mathbb{R}_{max}^{n\times m}$. This is particularly useful when considering large rectangular matrices ($m >> n$) due to the fact that, the columns of a matrix that are not part of the minimal generating set of the cone can be eliminated. This may be the case for the controllability matrix introduced in Chapter 6. Theorem 4.7 will be illustrated by an example taken from [13] (Example 3.4.3)

Example 4.1. Consider,

$$V = \begin{pmatrix} 1 & 1 & 2 & \varepsilon & 5 \\ 1 & 0 & 4 & 1 & 5 \\ 1 & \varepsilon & -1 & 1 & 0 \end{pmatrix}$$

then

$$-V^T \otimes' V = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 0 & -\varepsilon \\ -2 & -4 & 1 \\ -\varepsilon & -1 & -1 \\ -5 & -5 & 0 \end{pmatrix} \otimes' \begin{pmatrix} 1 & 1 & 2 & \varepsilon & 5 \\ 1 & 0 & 4 & 1 & 5 \\ 1 & \varepsilon & -1 & 1 & 0 \end{pmatrix}$$
$$= \begin{pmatrix} 0 & -1 & -2 & 0 & -1 \\ 0 & \varepsilon & 1 & \varepsilon & 4 \\ -3 & \varepsilon & 0 & \varepsilon & 1 \\ 0 & \varepsilon & -2 & \varepsilon & -1 \\ -4 & \varepsilon & -3 & \varepsilon & 0 \end{pmatrix}$$

So,

$$V \otimes \mathcal{V} = \begin{pmatrix} 1 & 0 & 2 & 1 & 5 \\ 1 & 0 & 2 & 1 & 5 \\ 1 & 0 & -1 & 1 & 0 \end{pmatrix}$$

As a result,

$$v_{.1} = 0 \otimes v_{.2} \oplus -3 \otimes v_{.3} \oplus 0 \otimes v_{.4}$$
$$v_{.5} = 4 \otimes v_{.2} \oplus 1 \otimes v_{.3} \oplus -1 \otimes v_{.4}$$

and the generating set is $v_{.2}, v_{.3}, v_{.4}$ □

This is also one of the definitions for the column rank of a matrix however, unlike conventional algebra, many definitions of rank exist in max-plus algebra. For more information on linear dependence and matrix ranks over the max-plus semiring the reader is referred to [15]. Generators are also referred to as *extreme points* and the formal definition follows [16].

Definition 4.8. *Let $S \subset \mathbb{R}^n_{max}$ be a cone. A vector $x \in S$ is an extreme point of S if the following property holds*

$$x = u \oplus v, \quad u, v \in S \Longrightarrow x = u \quad or \quad x = v.$$

If x is an extreme point of S, then the set $x = \{\lambda \otimes x | \lambda \in \mathbb{R}_{max}\}$ is an extreme ray of S.

A direct consequence is that the extreme rays are also the generating set of a convex cone. It is now evident why it is required for generating vectors to be independent, as scaled and dependent vectors already belong to the convex cone and therefore do not "augment" the cone.

In a similar way to conventional convex cones, max-plus convex cones can either be represented internally, in terms of their extreme points and rays, or externally, in terms of the intersection of (closed) max-plus half-spaces. A max-plus *half-space* is a set defined in the following way

$$\mathcal{H} = \{x \in \mathbb{R}^n_{max} | \oplus_{1 \le i \le n} \alpha_i \otimes x_i \le \oplus_{1 \le j \le n} \beta_j \otimes x_j\} \tag{4-11}$$

$\alpha, \beta \in \mathbb{R}^n_{max}$, and a max-plus *affine half-space* is a set of the form

$$\mathcal{H} = \{x \in \mathbb{R}^n_{max} | (\oplus_{1 \le i \le n} \alpha_i \otimes x_i) \oplus c \le (\oplus_{1 \le j \le n} \beta_j \otimes x_j) \oplus d\} \tag{4-12}$$

with $\alpha, \beta \in \mathbb{R}^n_{max}, \quad c, d \in \mathbb{R}_{max}$. Half-spaces are max-plus cones [17]. In other words, max-plus cone can be externally represented as a set of linear inequalities (in the conventional sense) or as a set of linear inequalities in the max-plus sense. This set of linear inequalities determines the span of the matrix as well as what values can be achieved by $x(k)$ in a sequence of the form (4-9).

Our objective is to represent the span of max-plus matrices as sets of linear inequalities (in the conventional sense). This is equivalent to representing the span of max-plus matrices as the union of non-overlapping polyhedral sets (in the conventional sense).

Polyhedral sets, Γ, are the solutions to systems of linear inequalities. Note that equalities can also be represented as two inequalities.

$$\Gamma = \{Mx \preceq b\} \tag{4-13}$$

x being a $n \times 1$ vector, M a $q \times n$ matrix and b a $q \times 1$ vector, and with $\preceq$ being a vector operator that serves as either $<$ or $\leq$. A method for representing the span of max-plus matrices as a union of non-overlapping polyhedral sets is introduced in the sequel. However, some necessary notions and operations will be first introduced.

A rearrangement of the columns of a matrix is called a *column-permutation* of the columns. A permutation can also be applied to the rows of a matrix (called *row permutation*. When the terms row or column are omitted then a permutation will refer to a permutation of both the rows and columns. A permutation of a matrix $V \in \mathbb{R}_{max}^{n\times n}$ will be denoted as ϖ, ϖ being the new arrangement. A *weight* can be associated with a permutation and is calculated in the following way

$$w(\varpi, V) = \bigotimes_{i\in\{1,\ldots,n\}} v_{i,\varpi(i)} \tag{4-14}$$

The permutation(s) with the greatest weight is also referred to as maximal permutation. A maximal permutation is not necessarily unique. The maximal permutation of a matrix V can be found in $O(n^3)$ time using the Hungarian method [13, 18].

The focus will now shift to column-permutations and some of their properties. A column-permutation does not alter the span of the matrix. A **column-permutation** of a matrix V will be denoted as $V^{\{v_i,\ldots,v_j\}}$ with $\{v_i,\ldots,v_j\}$ being the new arrangement of the columns with respect to the original arrangement. Moreover, multiplying a vector by a scalar $\lambda \in \mathbb{R}$ does not alter the span of the vector as a scalar multiple of a vector belongs to the same extreme ray and as a result has the same contribution in regards to the generation of a cone. This leads to the question whether there exist a permutation and multiplication of the columns of a matrix that reveals information about the max-plus polyhedra defined by the span of the columns of a matrix.

A matrix will be referred to as being in *definite form* if all its diagonal entries are equal to e (zero) and the maximal cycle mean (eigenvalue) of the matrix is not greater than e. Note that for all matrices in definite form the Kleene star exists. Furthermore, for all matrices that have at least one element in every row and column not equal to ε there exists a permutation and scaling of the columns that brings the matrix to definite form. As can be deducted by the definition of the definite form, only square matrices can be brought to this form. The case for rectangular matrices will also be presented later on. Furthermore, only regular square matrices (at least one element in every row is **not** ε) are considered. Although this may seem like a limitation it is not, as real systems modelled in max-plus algebra nearly always are regular.

Example 4.2. Consider the matrix V and its permutation (312)

$$V = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 3 & -1 \\ 2 & 6 & -2 \end{pmatrix}, \qquad V^{\{312\}} = \begin{pmatrix} 0 & 0 & 0 \\ -1 & 1 & 3 \\ -2 & 2 & 6 \end{pmatrix}$$

the second and third column can now be scaled by subtracting 1 and 6 respectively to obtain

$$V' = \begin{pmatrix} 0 & -1 & -6 \\ -1 & 0 & -3 \\ -2 & 1 & 0 \end{pmatrix}$$

The matrix V' is the definite form of V. All its diagonal entries are equal to zero and the maximal cycle mean is also equal to zero. □

The column permutation(s) that bring a matrix V to definite form are the same as the maximal permutation(s) of the matrix. However, instead of permuting both the rows and columns, as is the case with the maximal permutation, only the columns are rearranged. As a result, in order to obtain the feasible column-permutations for bringing a matrix to definite form, one has to first calculate the maximal permutation (as shown in (4-14)) and then apply that permutation as a **column-permutation**.

The method for obtaining the span of square matrices will now be presented. It was initially inspired by the work of Butkovic [19] and Sergeev [20]. In their papers they show that the eigenspace of definite matrices can be represented by a set of linear inequalities obtained from the Kleene star of the matrix. However, they are interested only in the eigenspace of definite matrices. We extend this method in a way that allows for the complete calculation of the span of any square matrix. Furthermore, we extend this method to rectangular matrices and also present a way for obtaining an over-approximation of the span of any matrix. Finally, we represent the span of any max-plus matrix as the union of polyhedral sets.

The definite form of a matrix may not be unique that is to say, that more than one column-permutation (and therefore scaling) may exist that brings a matrix to definite form. This form is of interest due to the fact that entries of this form that are equal to entries of its Kleene star define the inequalities of the closure of the span of the original matrix. By closure, we refer to the area of the span that is generated by linear combinations of all vectors of the generating set. Edges with the same orientation and same weight in the communication graphs of the definite form and its Kleene star define inequalities of the following form

$$\{u, v \in V(G), (u, v) \in E(G) \ \& \ E(G^*), w(u, v) = w^*(u, v)\} \ \rightarrow v - u \geq w(u, v) \tag{4-15}$$

The set of inequalities defined from the communication graphs of the definite form and its Kleene star are the external representation of the closure of a max-plus cone. In addition to this, an important result from [20] (Proposition 10) shows that the closures of all definite forms of any matrix, with at least one permutation of finite weight, are the same. Consequently, it makes no difference which permutation is chosen if more than one permutation exists that brings a matrix to definite form. The inequalities that define the closure of a matrix also form one of the polyhedral sets that describe the span of the matrix.

Nonetheless, the polyhedral defined by the inequalities of the closure does not express the whole span of a matrix, which is the objective. Unless the definite form is equal to its Kleene star then the inequalities will not include at least one of the extreme points and its connection to the closure. As explained in the prequel, the extreme points are the generators of the cone and their connection to the closure is the combination of at most $n-1$ generators (assuming a matrix $V \in \mathbb{R}_{max}^{n \times n}$). While the extreme points and their connection are part of the span of the matrix they are not included in the inequalities of the closure. However, this can be overcome with the method presented in the sequel.

Assume that for a square matrix $V \in \mathbb{R}_{max}^{n \times n}$ the inequalities of its closure have been obtained by bringing it to definite form, calculating its Kleene star and identifying the common elements. Furthermore, assume that the definite form of the matrix is not equal to its Kleene star as this would mean that all the extreme points (or generators) are included in the closure. The next step is to compare the generators to the inequalities of the closure. Recall that it does not matter whether we take the generators of the original matrix or its definite form as they lie on the same extreme ray and subsequently will violate the same inequalities. After comparing the generators with the inequalities, some of the inequalities will be violated. The inequality(ies) that are violated can then be augmented to include the extreme points. However, these inequalities are not just augmented arbitrarily as they lie on the line(s)(or a plane in a higher dimension than 2) that connect the extreme point to the closure. These lines(or planes) are obtained by setting the inequalities of the closure to equalities and finding which are satisfied for the generator in hand. As a result, the union of the inequalities of the closure with the augmented inequalities that lie on the lines defined above, fully characterize the span of the matrix. Every extreme point and its connection to the closure can be characterized by a (conventional) polyhedral set. Consequently, the union of the polyhedral sets defined by the extreme points and their connections with the polyhedral set defined by the closure of the matrix fully describe the span of the matrix. This will be further illustrated in the example presented in the sequel.

Example 4.3. The Kleene star of the matrix V' of example 4.2 is equal to

$$V^* = \begin{pmatrix} 0 & -1 & -4 \\ -1 & 0 & -3 \\ 0 & 1 & 0 \end{pmatrix}$$

The elements of a vector v are denoted as $v = (v_1, v_2, v_3)^T$. The definite form V' and the Kleene star V^* of V have four equal edges that provide the following inequalities

$$\begin{aligned} v_2 - v_1 &\geq -1 \\ v_1 - v_2 &\geq -1 \\ v_3 - v_2 &\geq 1 \\ v_2 - v_3 &\geq -3 \end{aligned}$$

which define the closure of the matrix V. Therefore, the closure can be described by the polyhedral set $\Gamma_1 = \{M_1 v < b_1\}$, where

$$M_1 = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \quad b_1 = \begin{pmatrix} 1 \\ 1 \\ -1 \\ 3 \end{pmatrix}$$

The next step is to compare the generators to the inequalities obtained above. The first generators considered is $(0, -1, -2)^T$. This generator violated the inequality $v_3 - v_2 \geq 1$ as $v_3 - v_2 = -1$. Additionally, the generator satisfies the equality $v_2 - v_1 = -1$. So the span is augmented by adding the inequality $-1 \leq v_3 - v_2 \leq 1$ for $v_2 - v_1 = -1$. Now the next generator is considered $(-1, 0, 1)^T$, this generator satisfies all the inequalities of the closure so it does not augment the span. Finally, the third generator $(-6, -3, 0)$ violated the inequality $v_2 - v_1 \geq 1$ and satisfies the equality $v_2 - v_3 = -3$, so the span is augmented by adding the inequality $1 \leq v_2 - v_1 \leq 3$ for $v_2 - v_3 = -3$. So the two polyhedral sets, Γ_2 and Γ_3, that are obtained from the two extreme points that lie outside the closure and their connections to the closure are given by $\Gamma_2 = \{M_2 v \leq b_2\}$ and $\Gamma_3 = \{M_3 v \leq b_3\}$ respectively, with

$$M_2 = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \quad b_2 = \begin{pmatrix} 1 \\ -1 \\ 1 \\ 1 \end{pmatrix}$$
$$M_3 = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \quad b_3 = \begin{pmatrix} -1 \\ 3 \\ -3 \\ 3 \end{pmatrix}$$

So, in total the span of the matrix V can be characterized by $\Gamma = \cup_{i=1}^{3} \Gamma_i$. The cross section of the span of the matrix for $v_3 = 0$ is depicted in Figure 4-1. The Figure depicts all the values that the elements v_1 and v_2 can attain when $v_3 = 0$.

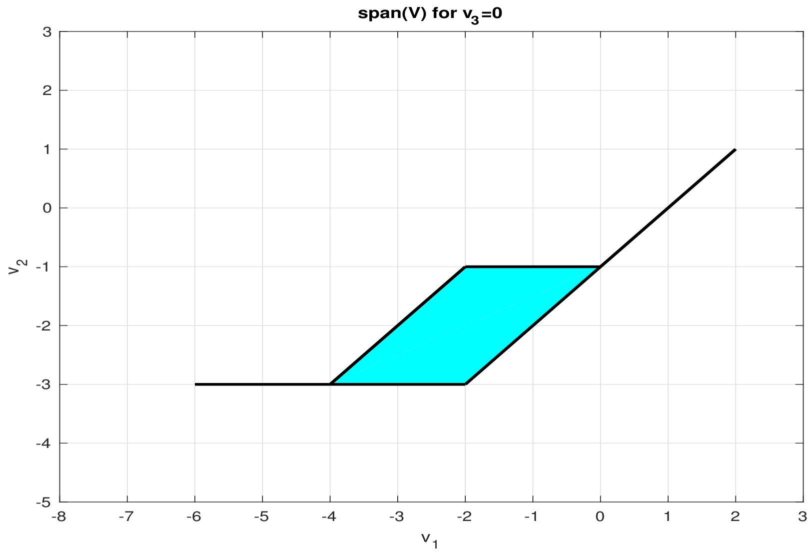

Figure 4-1: span(V) of Example 4.3 for $v_3 = 0$.
Closure (cyan): Polyhedral set Γ_1 described by $\{(-1 \leq v_2 - v_1 \leq 1) \wedge ((1 \leq v_3 - v_2 \leq 3))\}$
Connections & extreme points: Polyhedral sets Γ_2, Γ_3 $\{((1 \leq v_2 - v_1 \leq 3) \wedge (v_2 - v_3 = -3)) \vee ((-1 \leq v_3 - v_2 \leq 1) \wedge (v_2 - v_1 = -1))\}$

□

Calculating the min-plus span of a square matrix, $span^{-}(A)$, can be done in the same way as for max-plus matrices. The two modifications that need to be made in order to account for the min operation instead of the max operation, are follows. Firstly, instead of requiring the smallest (in contrast to max-plus) eigenvalue of the definite form to be negative or zero, now it is required to be zero or positive. This occurs because the Kleene star for min-plus matrices is only defined for matrices that have cycle weights positive or equal to zero. Secondly, the inequalities defined by the edges of the min-plus Kleene star of the min-plus definite form of a matrix have an opposite orientation to the max-plus case. More specifically,

$$\{u, v \in V(G), (u, v) \in E(G) \;\&\; E(G^*), w(u, v) = w^*(u, v)\} \;\rightarrow v - u \leq w(u, v) \quad (4\text{-}16)$$

instead of, $v - u \geq w(u, v)$ (4-15), as is the max-plus case.

Before considering the case of non-square matrices an important result from Prou [21] (Proposition 3.3) will be presented. Let $V \in \mathbb{R}_{max}^{n \times m}$, then define $V^{\clubsuit} \in \bar{\mathbb{R}}^{n \times n}$ as $V^{\clubsuit} = V \otimes -V^T$. The proposition then states that

$$span(V^{\clubsuit}) \subset span(V) \subset span^{-}(V^{\clubsuit}) \quad (4\text{-}17)$$

Moreover, since only regular matrices are considered, the diagonal entries of $V^{\clubsuit}$ will be equal to zero since the diagonal entries are the max-plus multiplication of the same vector with opposite sign (i.e. $v^T \otimes -v$). As it is regular each row (and thus column of $-V^T$) will have at least one finite element. Another important property of this matrix is that $V^♣ = V^{♣**}$ [21] (V^{**} denoting the min-plus Kleene star). As a result, $V^♣$ is already in min-plus definite form and its span generates a closure that includes the span of V.

In essence only the second part of relation (4-16), $span(V) \subset span^-(V^♣)$, will be used. The reason for this is that the matrix $V^♣$ may contain elements equal to ∞, as it belongs to $\bar{\mathbb{R}}$ and not $\mathbb{R}_{max}$, thus rendering it impossible to bring it to definite form and calculate its (max-plus) span with the methodology presented above. However, the second part of the relation provides advantages. Such an advantage is that it allows us to obtain an upper bound on the span of a rectangular matrix by calculating the min-plus span of a square matrix. As will be indicated in the next subsection, this is particularly useful for matrices $V \in \mathbb{R}_{max}^{n\times m}$, where $m >> n$. For a better understanding of the second part of relation (4-16), Figure 4-2 depicts the min-plus span of $V^♣$, for V of Example 4.3, in comparison to the actual span (the actual span is included in the min-plus span, it is depicted in cyan for the purpose of creating a contrast between the actual span and min-plus span of $V^♣$).

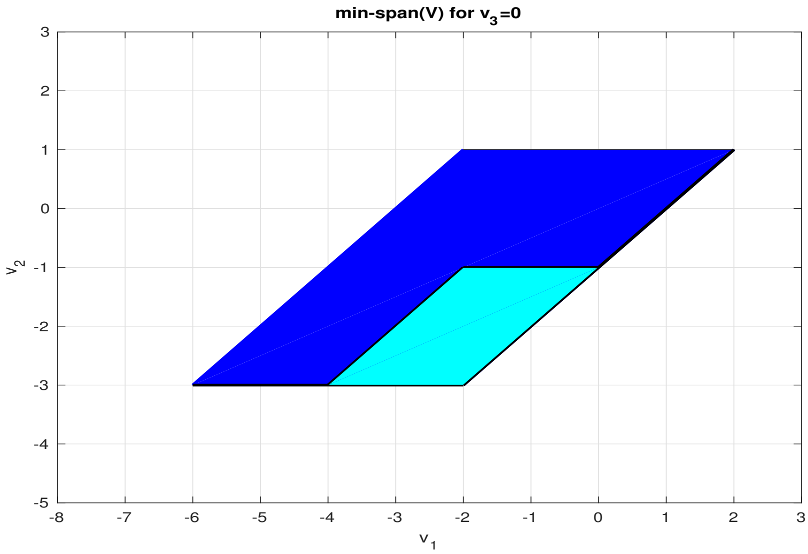

Figure 4-2: span(V) (cyan,black) and span($V^♣$) (blue) of Example 4.3 for $v_3 = 0$.

4-4-1 Obtaining the span of non-square matrices.

The case for non-square matrices will now be considered. As can be deducted, only square matrices can be brought to definite form. Rectangular matrices do not possess a main diagonal and therefore the methodology presented above has to be adjusted. A distinction will be made for matrices that have less columns than rows and matrices that have more columns than rows.

Case 1 ($m > n$). The method for precisely calculating the span of a matrix $A \in \mathbb{R}_{max}^{n\times m}, m > n$, is to calculate the span of all square submatrices with linearly independent columns of A. The overall span is then the union of these spans. In this thesis the term submatrix will refer to a matrix that is obtained from the original matrix after an erasure of a number of columns. A submatrix will be denoted as, $A_{[i_1,\ldots,i_j]}, i \in [1,\ldots,m], j \in \mathbb{N}^+$, i_j symbolizing the column(s) of the original matrix that have been omitted.

As was seen above, since the subspace created by the columns of a matrix is a convex-cone, it is generated by its extreme rays (i.e. the linearly independent columns of the matrix). By considering the span of all linearly independent square submatrices of a matrix, it is possible to obtain all the closures, extreme points and connections generated by the linearly independent columns. Because some combinations of columns are the same between the different submatrices, some of the closures, extreme points and connections defined by the spans overlap. In addition to this, unlike square matrices, for rectangular matrices the closure of the span may be defined by more than one polyhedral set. This is occurs because the overall closure of a rectangular matrices is the union of the polyhedral sets that define the closure for every square submatrix. An example follows to better demonstrate the process.

Example 4.4. Consider the matrix V

$$V = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 2 & 3 & 4 \\ 2 & 4 & 6 & 8 \end{pmatrix}$$

The objective is to determine the span generated by its columns. In this case it is easy to see that the columns of the matrix are linearly independent. For more complex cases Theorem 4.7 can be used to determine the linearly independent columns. Since this matrix has four linearly independent columns it also has 4 square submatrices. The first submatrix to be considered is

$$V_{[1]} = \begin{pmatrix} 0 & 0 & 0 \\ 2 & 3 & 4 \\ 4 & 6 & 8 \end{pmatrix}$$

where the first column has been omitted. The definite form and its Kleene star then are

$$V'_{[1]} = \begin{pmatrix} 0 & -3 & -8 \\ 2 & 0 & -4 \\ 4 & 3 & 0 \end{pmatrix}, \quad {V'_{[1]}}^* = \begin{pmatrix} 0 & -3 & -7 \\ 2 & 0 & -4 \\ 5 & 3 & 0 \end{pmatrix}$$

The span of this submatrix is then given by the following polyhedral sets

$$span(V_{[1]}) = \begin{cases} ((2 \le v_2 - v_1 \le 3) \vee (-4 \le v_2 - v_3 \le -3)) \\ \vee((3 \le v_2 - v_1 \le 4) \wedge (v_2 - v_3 = -4)) \\ \vee((-3 \le v_2 - v_3 \le -2) \wedge (v_2 - v_1 = 2) \end{cases}$$

or in polyhedral form $\Gamma_{[1],1}, \Gamma_{[1],2}, \Gamma_{[1],3}$, with the polyhedral sets defined as follows

$$M_{[1],1} = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \quad b_{[1],1} = \begin{pmatrix} -2 \\ 3 \\ -3 \\ 4 \end{pmatrix}$$

$$M_{[1],2} = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \quad b_{[1],2} = \begin{pmatrix} -3 \\ 4 \\ -4 \\ 4 \end{pmatrix}$$

$$M_{[1],3} = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \quad b_{[1],3} = \begin{pmatrix} -2 \\ 2 \\ -2 \\ 3 \end{pmatrix}$$

In a similar manner the spans for submatrices $V_{[2]}, V_{[3]}, V_{[4]}$ are obtained to be

$$span(V_{[2]}) = \begin{cases} ((1 \leq v_2 - v_1 \leq 3) \wedge (-4 \leq v_2 - v_3 \leq -3)) \\ \vee((3 \leq v_2 - v_1 \leq 4) \wedge (v_2 - v_3 = -4)) \\ \vee((-1 \leq v_2 - v_3 \leq -3) \wedge (v_2 - v_1 = 1)) \end{cases}$$

$$span(V_{[3]}) = \begin{cases} ((1 \leq v_2 - v_1 \leq 2) \wedge (-4 \leq v_2 - v_3 \leq -2)) \\ \vee((2 \leq v_2 - v_1 \leq 4) \wedge (v_2 - v_3 = -4)) \\ \vee((-2 \leq v_2 - v_3 \leq -1) \wedge (v_2 - v_1 = 1)) \end{cases}$$

$$span(V_{[4]}) = \begin{cases} ((1 \leq v_2 - v_1 \leq 2) \vee (-3 \leq v_2 - v_3 \leq -2)) \\ \vee((2 \leq v_2 - v_1 \leq 3) \wedge (v_2 - v_3 = -3)) \\ \vee((-2 \leq v_2 - v_3 \leq -1) \wedge (v_2 - v_1 = 1)) \end{cases}$$

Obviously some of the spans defined overlap. To begin with, only the polyhedral sets that define the closure(s) are considered. Notice that, the closures obtained from $V_{[1]}$ and $V_{[4]}$ are already included in the closures of $V_{[2]}$ and $V_{[3]}$ respectively. Correspondingly, the closure of the overall matrix V is obtained from taking the union of the closures defined by $V_{[2]}$ and $V_{[3]}$. So for the closure we have

$$clo(V) = \begin{cases} ((1 \leq v_2 - v_1 \leq 3) \wedge (-4 \leq v_2 - v_3 \leq -3)) \\ \vee((1 \leq v_2 - v_1 \leq 2) \wedge (-3 \leq v_2 - v_3 \leq -2)) \end{cases}$$

or in polyhedral form $\Gamma_{clo} = \cup_{i=1}^{2} \Gamma_{clo,i}$

$$M_{clo,1} = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \quad b_{clo,1} = \begin{pmatrix} -1 \\ 3 \\ -3 \\ 4 \end{pmatrix}$$

$$M_{clo,2} = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \quad b_{clo,2} = \begin{pmatrix} -1 \\ 2 \\ -2 \\ 3 \end{pmatrix}$$

The next step now is to consider all the boundary rays defined by the submatrices. These boundary rays are checked against the inequalities of the closure (above). If the boundary rays do not violate these inequalities then no modification is made to the span of the matrix. If however this is not the case, then the boundary rays of the submatrices are added as boundary rays to the span of the overall matrix. We finally obtain

$$span(V) = \begin{cases} ((1 \leq v_2 - v_1 \leq 3) \wedge (-4 \leq v_2 - v_3 \leq -3)) \\ \vee((1 \leq v_2 - v_1 \leq 2) \wedge (-3 \leq v_2 - v_3 \leq -2)) \\ \vee((3 \leq v_2 - v_1 \leq 4) \wedge (v_2 - v_3 = -4)) \\ \vee((-2 \leq v_2 - v_3 \leq -1) \wedge (v_2 - v_1 = 1)) \end{cases}$$

or in polyhedral form $\Gamma = \Gamma_{clo} \cup \Gamma_{con}$, where $\Gamma_{con} = \cup_{i=1}^{2} \Gamma_{con,i}$ and the polyhedral sets $\Gamma_{con,1}$, $\Gamma_{con,2}$ are given by

$$M_{con,1} = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \quad b_{con,1} = \begin{pmatrix} -3 \\ 4 \\ -4 \\ 4 \end{pmatrix}$$

$$M_{con,2} = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix}, \quad b_{con,2} = \begin{pmatrix} -1 \\ 1 \\ -1 \\ 2 \end{pmatrix}$$

$span(V)$ is depicted in Figure 4-3.

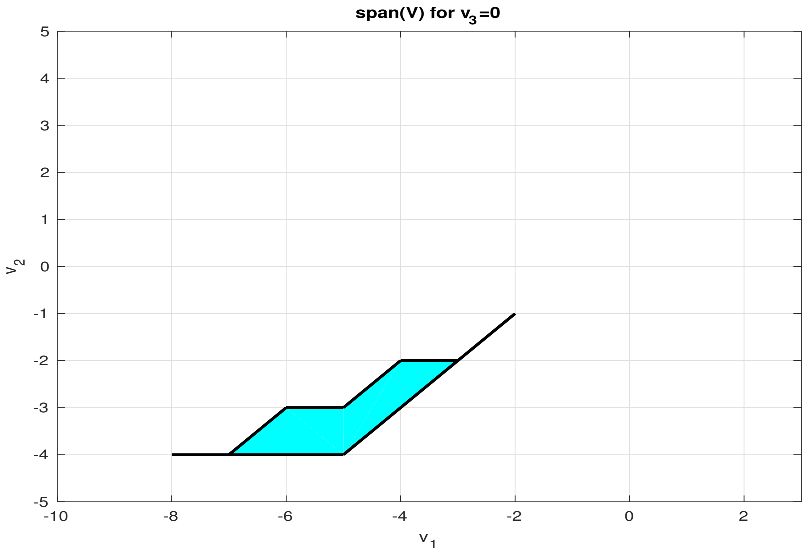

Figure 4-3: span(V) of Example 4.4 for $v_3 = 0$.

□

Although it is possible with this method to precisely determine the span of a rectangular matrix, it may become computationally inefficient for matrices of a large order or for matrices with $m >> n$. By using the property of relation (4-17) it is possible to compute an over-approximation of the span of a rectangular matrix with a one-shot computation. Another benefit is that the min-plus span of $V^\clubsuit$ can always be expressed as one polyhedral set. This may be of particular interest when the goal is to determine whether all the state vertices remain bounded with respect to each other. However, if precision is of greater importance it is preferable to use the method of Example 4.4. Which method is used depends on whether the objective is precision or specific information regarding the span of a matrix. The example in the sequel demonstrates how the min-plus method can be used for rectangular matrices.

Example 4.5. Consider again the matrix V of Example 4.4

$$V = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 2 & 3 & 4 \\ 2 & 4 & 6 & 8 \end{pmatrix}$$

The next step is to calculate $V^\clubsuit$

$$V^\clubsuit = \begin{pmatrix} 0 & -1 & -2 \\ 4 & 0 & -1 \\ 8 & 4 & 0 \end{pmatrix}$$

as was stated above this matrix coincides with its min-plus Kleene star. So its min-plus span is given by

$$span^{-}(V^{\clubsuit}) = \begin{cases} 1 \leq v_2 - v_1 \leq 4 \\ 1 \leq v_3 - v_2 \leq 4 \\ 2 \leq v_3 - v_1 \leq 8 \end{cases}$$

or in polyhedral form, $\Gamma^{\clubsuit}$

$$M^{\clubsuit} = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \\ 1 & 0 & -1 \\ -1 & 0 & 1 \end{pmatrix}, \quad b^{\clubsuit} = \begin{pmatrix} -1 \\ 4 \\ -1 \\ 4 \\ -2 \\ 8 \end{pmatrix}$$

The min-plus span of $V^{\clubsuit}$ in comparison to the exact span of the matrix is depicted in Figure 4-4.

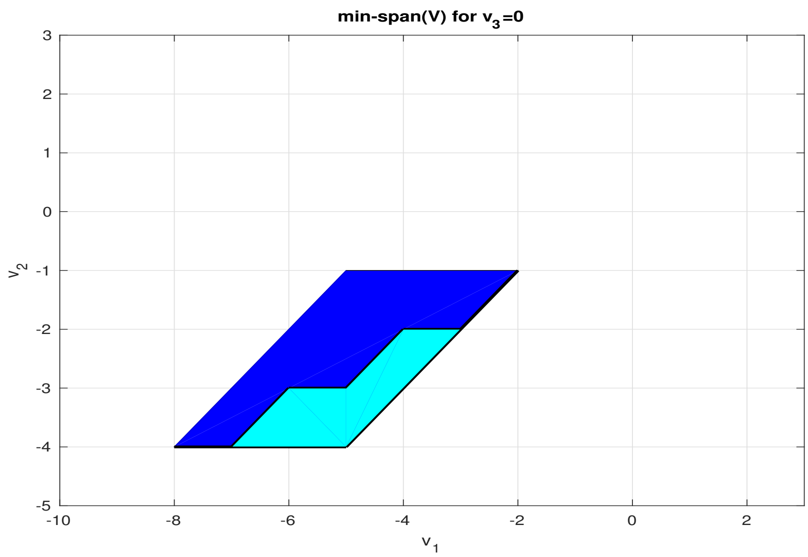

Figure 4-4: $span^{-}(V^{\clubsuit})$ in comparison to $span(V)$ of Example 4.4.

□

Even though the min-plus span defines a closure that contains the span of the original matrix and does not precisely calculate it, it is computationally less expensive. Furthermore, if a state vertex is unbounded in the original matrix, it will remain unbounded in min-span of the $\clubsuit$ matrix, and vice versa.

Case 2 (m<n). The difference between this case and the previous one is that matrices of this type do not have square submatrices. Nonetheless, this can be overcome by augmenting the matrix with max-plus multiples of the linearly independent columns of the matrix. Adding columns that are max-plus multiples of the already existing columns of a matrix does not alter the span of the matrix, because both vectors are part of the same extreme ray and as a result do not contribute to the span of the matrix. The min-plus method can be also applied to matrices of this type. Although, as in the previous, the span obtained by the min-plus method could be an over approximation of the actual span of the matrix.

Example 4.6. Consider the matrix V,

$$V = \begin{pmatrix} 0 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 1 & \varepsilon & \varepsilon \\ 2 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 1 & \varepsilon & \varepsilon \end{pmatrix}$$

the two columns that are equal to ε are replaces by max-plus multiples of the existing columns

$$V_{aug} = \begin{pmatrix} 0 & \varepsilon & 1 & \varepsilon \\ \varepsilon & 1 & \varepsilon & 2 \\ 2 & \varepsilon & 3 & \varepsilon \\ \varepsilon & 1 & \varepsilon & 2 \end{pmatrix}$$

its definite form and its respective Kleene star are

$$V^{'} = V^{'*} = \begin{pmatrix} 0 & \varepsilon & -2 & \varepsilon \\ \varepsilon & 0 & \varepsilon & 0 \\ 2 & \varepsilon & 0 & \varepsilon \\ \varepsilon & 0 & \varepsilon & 0 \end{pmatrix}$$

Finally, the inequalities defining the span are

$$span(V) = \begin{cases} 2 \leq x_3 - x_1 \leq 2 \\ 0 \leq x_4 - x_2 \leq 0 \end{cases}$$

or in polyhedral form

$$M^{\clubsuit} = \begin{pmatrix} 1 & 0 & -1 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & -1 & 0 & 1 \end{pmatrix}, \quad b^{\clubsuit} = \begin{pmatrix} -2 \\ 2 \\ 0 \\ 0 \end{pmatrix}$$

□

Chapter 5

Max-Plus-Linear (MPL) Systems & Switching Max-Plus-Linear (SMPL) Systems

Railway networks, production system with fixed routing and communication networks are just some examples of systems that can be modelled with the use of max-plus algebra. These systems are all discrete event systems (DES) in which there is no concurrency or choice but, there is synchronization. Even though these systems are inherently nonlinear in conventional algebra, they become linear when modeled within the max-plus semiring.

5-1 Max-plus linear (MPL) systems

Scheduling has been defined as 'the allocation of resources over time to perform a collection of tasks' [22]. The processing of a job on a machine is called operation and each job consists of a set of operations and an order of operations. The most common objective is to minimize the time required for the completion of all jobs, subject to retaining the order of operations.

DES that have a fixed schedule can be characterized by an MPL model. These systems have no concurrency or choice but synchronization is possible. By no concurrency we mean that no two operations can take place simultaneously on the same machine, or in terms of a railway network example, two trains would not be able to occupy the same block of a track simultaneously. What is meant by synchronization is that machines wait for the previous operation to finish before beginning their new operation. In a production system this could be interpreted as a machine having to finish processing on its current product in order to begin processing on a new product. Moreover, the implication of no choice is that the order of operations for a job is fixed and cannot change. In a railway network example this would mean that a train follows a fixed route and does not alter its sequence of stations except for unforeseeable circumstances. Such DES can be modeled by an MPL system of the form

$$x(k) = A \otimes x(k-1) \oplus B \otimes u(k) \tag{5-1}$$

$$y(k) = C \otimes x(k) \tag{5-2}$$

with $A \in \mathbb{R}_{max}^{n \times n}$, $B \in \mathbb{R}_{max}^{n \times m}$ and $C \in \mathbb{R}_{max}^{o \times n}$. The number of states is equal to n, of inputs to m, of outputs to o and the event counter is k. At this point it is important to note that usually the state, input and output vectors ($x(k)$, $u(k)$, $y(k)$) represent time instants at which events occur for the k^{th} time. The output is often the state of the system and as a result it is a common theme to exclude the output equation (5-2) and describe a system only by its state equation (5-1). Systems that do not have an input are called *autonomous systems*. In order for the concept of MPL systems to be better understood we will provide an example taken from [23].

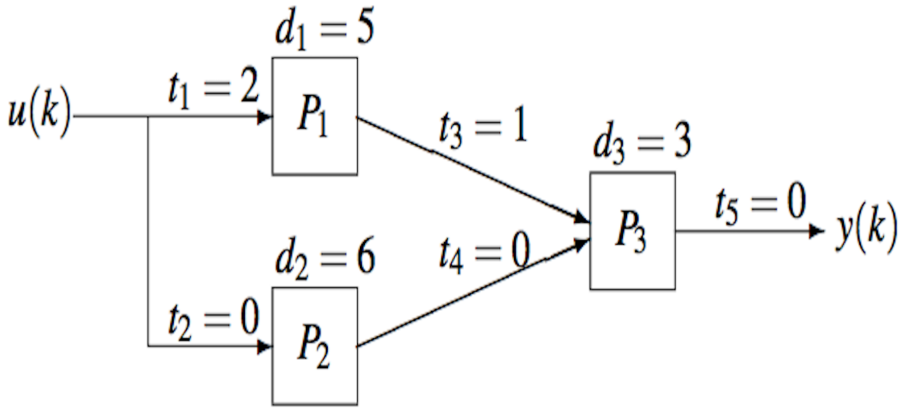

Figure 5-1: A simple production system.

Example 5.1. Consider the system depicted in Figure 5-1. Three processing units, P_1, P_2 and P_3 comprise this production system. Raw material is fed and then split, to P_1 and P_2 where it is processed. Once the process has finished the processed material of P_1 and P_2 respectively is assembled at P_3. The processing times for each processing unit are depicted on the figure by the letter d and the respective subscript. Additionally, the transportation time between the processing units is also represented on the figure by the letter t. Only once the previous product on a processing unit has been completed, can the processing unit begin working on a new product. The state, input and output variables are defined as follows.

- $x_i(k)$: time instant at which the i^{th} processing unit start working for the k^{th} time.
- $u(k)$: time instant at which raw material is fed into the system for the k^{th} time.
- $y(k)$: time instant at which the k^{th} product leaves the system.

In order for P_1 to begin working on a product for the k^{th} time it needs to have raw material available and to have finished working on the product of the previous time instant $(k-1)$. This can be expressed as

$$x_1(k) = \max(x_1(k-1) + 5, u(k) + 2)$$

The other states can be expressed in a similar way.

$$\begin{aligned}
x_2(k) &= \max(x_2(k-1) + 6, u(k) + 0) \\
x_3(k) &= \max(x_1(k) + 5 + 1, x_2(k) + 6 + 0, x_3(k-1) + 3) \\
&= \max(x_1(k-1) + 11, x_2(k-1) + 12, x_3(k-1) + 3, u(k) + 8) \\
y(k) &= x_3(k) + 3 + 0
\end{aligned}$$

Notice for $x_3(k)$, that the constraint of the processed material from unit P_1 being available before P_3 can begin the processing, is expressed as $\max(x_1(k) + 5 + 1)$. This is due to the fact that, $x_1(k)$ represents the instant that unit P_1 *begins* its operations.As a result, we must also add the time value for the processing time of P_1 (d_1) in addition to the transportation time (t_3). The same is also true for the constraint corresponding to unit P_2. The MPL system can then be expressed in the form of (5-1)-(5-2) in the following way

$$\begin{aligned}
x(k) &= \begin{pmatrix} 5 & \varepsilon & \varepsilon \\ \varepsilon & 6 & \varepsilon \\ 11 & 12 & 3 \end{pmatrix} \otimes x(k-1) \oplus \begin{pmatrix} 2 \\ 0 \\ 8 \end{pmatrix} \otimes u(k) \\
y(k) &= \begin{pmatrix} \varepsilon & \varepsilon & 3 \end{pmatrix} \otimes x(k)
\end{aligned}$$

where $x(k) = \begin{pmatrix} x_1(k) & x_2(k) & x_3(k) \end{pmatrix}^T$. □

In MPL systems the eigenvalue of the state matrix A can be regarded as the minimum period required for a schedule in order for it to be stable. In other words, the eigenvalue indicates the period it takes for each node to become active while the eigenvector indicates the order of activation of each node associated with it. Going again back to the production system example, if a production system is represented by a max-plus model, such as the one in the example above, and the eigenvalue is equal to λ then, any schedule with a period shorter than λ $(< \lambda)$ will be unstable. However, it is common practice to have a period a bit larger than the eigenvalue with the purpose of enhancing the robustness properties of the system.

An important aspect for systems of the form (5-1)-(5-2), is whether they are stable and if not, whether they can be stabilized. A system is stable, as stated above, if the period of the system is greater than the eigenvalue. If this is the case then the buffer levels of the system will remain bounded. Lets assume that a reference signal $r(k)$ exists for the system, describing the due dates of the finished product for the example above. The buffer levels of an MPL system are then expressed as follows

$$y(k) - r(k) \leq M_{yr} \tag{5-3}$$

$$r(k) - y(k) \leq M_{ry} \tag{5-4}$$

$$y(k) - u(k) \leq M_{yu} \tag{5-5}$$

Alternatively, a definition of stability can be given with regards to the boundedness of the internal buffer levels [24]. By boundedness of the internal buffer levels we refer to the fact that the state trajectory remains bounded. That is to say, there exists a finite $\delta > 0$ for which the number of material/train/parts (depending on the system being modelled) in the buffer is always less than δ. More formally,

$$\exists \delta \quad \textit{such that} \quad |x_i(k) - x_j(k)| \leq \delta, \quad \forall i, j, k \tag{5-6}$$

Condition (5-3) indicates that the delay between the output and the due date remains bounded, condition (5-4) shows that the amount of parts in the output will remain bounded and lastly, condition (5-5) guarantees that the time between the starting date $u(k)$ and the output date $y(k)$ also remains bounded. A direct consequence of this is that max-plus linear systems are not inherently stable or unstable but rather, the stability of these systems is also a function of the input and reference signal. For the example given above this can be interpreted in the following way. In order for the system to be stable the asymptotic slope of the reference signal needs to be greater than the eigenvalue of the system. For a production line this can be interpreted as not having more requests for completed products than the amount of products the production line can produce. Additionally, the input must be chosen in such a way that the overall period of the system is greater than the eigenvalue, while at the same time, guaranteeing that the products are finished before the due dates. This ensures firstly, that the due dates are met meaning there is no delay and secondly, that the buffers of the system remain bounded. With respect to the previous example, an unbounded buffer would constitute feeding in material every second. Then the amount of raw material waiting to be processed would tend to infinity as time tends to infinity. If all the previous conditions hold true simultaneously then the system is stable. However, how can we determine a priori whether a system is stabilizable. The answer to this question is given in [2] by the following theorem.

Theorem 5.1. *Any structurally controllable and structurally observable system can be made internally stable by output feedback.*

Consequently, we can determine whether a MPL system is feedback stabilizable by checking whether it is structurally controllable and observable. The notions of structural controllability and observability and how they can be derived for MPL systems will be discussed in detail in the next chapter, where an overall overview of the notion of controllability for MPL systems will be given. For now it is important to note that for a system to be feedback stabilizable it is sufficient to show that it is structurally controllable and observable. For structurally controllable and observable systems condition (5-6) already entails that the output and input remain bounded [24].

Nevertheless, the fixed structure of MPL systems renders the modeling of a large variety of DES impossible. This is due to the fact that, the fixed structure of a MPL system does not allow for a variation in the order of events to be modeled or for the termination of a synchronization. This provides challenges in many real life applications. For example, it would not be possible to model the scheduling of a printer that needs to change from printing A4 size of paper to A3 by a MPL system. This occurs because the structure of the system and the amount of time it would take to print varies depending on the size of paper that needs to be printed. In the interest of overcoming such challenges switching max-plus linear (SMPL) systems were introduced [3]. These systems will be presented in the next section.

5-2 Switching max-plus linear (SMPL) systems

With the purpose of modeling a change in structure, order or a break in the synchronization SMPL systems were introduced by van den Boom and De Schutter [3]. Such systems allow for switching between different modes of operation. Each mode is a separate MPL system described by its own state, input and output matrices. The switching between different modes of operation can be either deterministic, stochastic or a combination of the two [25].

As opposed to conventional switched systems which are most commonly used in hybrid theory to describe a combination of continuous and discrete dynamics, SMPL systems are DESs. The max-plus linear state space model that describes such systems in mode $l(k) \in \{1, \cdots, L\}$, L denoting the number of modes, is given by Equations (5-7)-(5-8)

$$x(k) = A^{l(k)} \otimes x(k-1) \oplus B^{l(k)} \otimes u(k) \tag{5-7}$$

$$y(k) = C^{l(k)} \otimes x(k) \tag{5-8}$$

where matrices $A^{l(k)}$, $B^{l(k)}$ and $C^{l(k)}$ are the system matrices for mode $l(k)$. From this point on, when referring to SMPL systems, only SMPL systems with state matrices of equal order will be considered. That is, $A^{l(k)} \in \mathbb{R}^{n\times n}_{max}, \quad \forall l(k) \in \{1, \cdots, L\}$. The number of inputs and outputs may vary between different modes.

By switching between different modes it is possible to model a wider range of DESs and to incorporate the potential modeling changes in the structure of the system, in the order of events or in the synchronization requirements. As was the case with MPL systems an example will be presented, taken from [3], for a better understanding of the concept of SMPL systems.

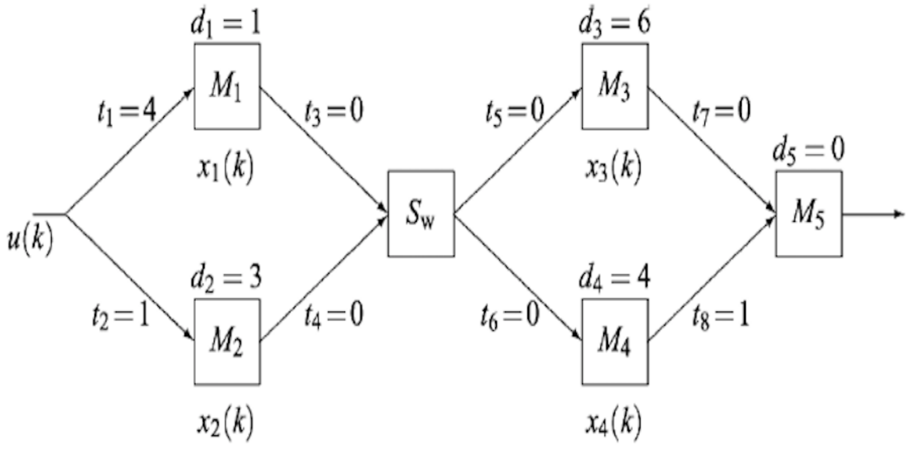

Figure 5-2: A simple production system.

Example 5.2. Consider the production system depicted in Figure 5-2. There are five processing units denoted M_1, M_2, M_3, M_4 and M_5. The raw material is fed into M_1 and M_2 where it is processed. The processed material can now be fed into either M_3 or M_4, which perform the same operation with different processing times (that is both machines can process the material provided from M_1 or M_2). As a result, the processed products exiting M_1 and M_2 are driven to a switching mechanism S_w, that directs the first product received to the slower machine M_3 and the second product to M_4. The states and inputs are defined in a similar way to the previous example. A system such as this can me modelled as a SMPL system with two modes, one mode for when M_1 finishes first and one mode for when M_2 finishes first. If M_1 finishes first (which will be mode - 1 in this example) then the material of M_1 will be fed to M_3 and the material of M_2 to M_4. As a result, in order for M_3 to begin its operation M_1 needs to have finished processing and for M_3 to have finished its previous operation so we have (we neglect the transportation times as they are zero from the machine to the switching mechanism),

$$x_3(k) = \max(x_1(k) + d_1, x_3(k-1) + d_3)$$

where,

$$x_1(k) = \max(x_1(k-1) + d_1, u(k) + t_1)$$

so we substitute to obtain,

$$\begin{aligned} x_3(k) &= \max(x_1(k-1) + 2d_1, x_3(k-1) + d_3, u(k) + d_1 + t_1) \\ &= \max(x_1(k-1) + 2, x_3(k-1) + 6, u(k) + 5) \end{aligned}$$

The second mode is also modeled in the same way. The SMPL system is then

$$x(k) = \begin{pmatrix} 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon \\ 2 & \varepsilon & 6 & \varepsilon & \varepsilon \\ \varepsilon & 6 & \varepsilon & 4 & \varepsilon \\ 8 & 11 & 12 & 9 & \varepsilon \end{pmatrix} \otimes x(k-1) \oplus \begin{pmatrix} 4 \\ 1 \\ 5 \\ 4 \\ 11 \end{pmatrix} \otimes u(k) \quad (mode-1) \tag{5-9}$$

$$x(k) = \begin{pmatrix} 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 6 & 6 & \varepsilon & \varepsilon \\ 2 & \varepsilon & \varepsilon & 4 & \varepsilon \\ 7 & 11 & 12 & 9 & \varepsilon \end{pmatrix} \otimes x(k-1) \oplus \begin{pmatrix} 4 \\ 1 \\ 5 \\ 4 \\ 10 \end{pmatrix} \otimes u(k) \quad (mode-2) \tag{5-10}$$

And the switching mechanism is defined as

$$\begin{pmatrix} z_1(k) \\ z_2(k) \end{pmatrix} = \begin{pmatrix} x_1(k) + d_1 \\ x_2(k) + d_2 \end{pmatrix} \tag{5-11}$$

$$= \begin{pmatrix} \max(x_1(k-1) + 2, u(k) + 5) \\ \max(x_2(k-1) + 6, u(k) + 4) \end{pmatrix} \tag{5-12}$$

and the sets

$$\mathcal{L}_1 = \{z \in \mathbb{R}^2_{max} | z_1 \leq z_2\} \tag{5-13}$$

$$\mathcal{L}_2 = \{z \in \mathbb{R}^2_{max} | z_1 > z_2\} \tag{5-14}$$

The variables z_1 and z_2 are the time instant at which machines M_1 and M_2, respectively, *complete* their product for the k^{th} time. So for the first mode, the set $\mathcal{L}_1$ corresponds to when $z_1 \leq z_2$ or in other words, when the processed material from M_1 reached the switching mechanism first. Conversely, the second set $\mathcal{L}_2$ is associated with when M_2 finishes processing first, when $z_2 > z_1$. □

The switching between the different modes, as mentioned above, can either be deterministic or stochastic or a combination of the two (for the example above it is deterministic). If the switching is deterministic then it depends on deterministic variables such as the previous state $x(k-1)$, the input variable $u(k)$, and(or) a (complementary) control variable $v(k)$. In contrast, if the switching is stochastic it will depend on stochastic variables. The switching can also be a combination of deterministic and stochastic variables. In this case the switching may depend on the previous state, input variable and control variable in addition to stochastic variables. Even when the switching purely depends on deterministic variables, a switching sequence can be implemented because it optimizes certain parameters of interest or because it was forced by the situation.

As SMPL systems are not wholly characterized by one state matrix, the concept of maximum autonomous growth rate is used as a substitute to the max-plus eigenvalue. The definition of the maximum autonomous growth rate is given below [25]

Definition 5.2. *Consider an SMPL system of the form (5-7)-(5-8) with system matrices* $A^{(l)}$*,* $l \in \{1, \cdots, L\}$*. Define the matrices* $A_{\alpha}^{(l)}$ *with* $[A_{\alpha}^{(l)}]_{ij} = [A^{(l)}]_{ij} - \alpha$*. Define the set* $\mathcal{S}_{\text{fin},n}$ *of all* $n \times n$ *max-plus permutation matrices with finite diagonal entries, so* $\mathcal{S}_{\text{fin},n} = \{S | S = diag_{\oplus}(s_1, \cdots, s_n), s_i \text{ is finite}\}$*. The maximum autonomous growth rate* ζ *is defined by*

$$\zeta = \min\{\alpha | \exists S \in \mathcal{S}_{\text{fin},n} \quad \text{such that} \quad [S \otimes A_{\alpha}^{(l)} \otimes S^{\otimes -1}]_{ij} \leq 0 \quad \forall i, j, l\} \tag{5-15}$$

The maximum autonomous growth rate can be seen as the equivalent of choosing the largest max-plus eigenvalue of a system matrix in MPL systems. Moreover, the maximum autonomous growth rate exists and is finite for any SMPL system.

As is the case with the max-plus eigenvalue for MPL systems, the maximum autonomous growth rate plays an important role in the control of SMPL systems. As shown in [25] an SMPL can be stabilized if it fulfills the three conditions which are presented below. Note that, as for MPL systems, the structural controllability and observability have not yet been defined as a more comprehensive review of the notion (for SMPL systems also) will be given in the next chapter.

Theorem 5.3. *Consider an SMPL system with maximum autonomous growth rate* ζ *and consider a reference signal with growth rate* ρ*. Define the matrices* $A_{\rho}^{(l)}$ *with* $[A_{\rho}^{(l)}]_{i,j} = [A^{(l)}]_{i,j} - \rho$*. Now if*

1. $\rho > \zeta$
2. *the system is structurally controllable. and*
3. *the system is structurally observable,*

then any input signal

$$u(k) = \rho k + \mu(k), \quad \text{where} \quad \mu_i(k) \leq \mu_{max}, \forall i, \forall k$$

for a finite value μ_{max}*, will stabilize the SMPL system.*

A direct consequence of Theorem 5.3 is that for an SMPL system to be stabilizable it is sufficient to show conditions (1-3) above. Additionally, it is shown that structural controllability and observability are sufficient conditions for the feedback stabilizability of MPL systems. This partly motivates the subject of the next chapter.

5-2-1 Annotation of switching sequences

A brief description of how switching sequences will be annotated will be presented here. A switching sequence will be denoted by the vector

$$s^{(\sigma)} = \{[s_1, \ldots, s_t]^T | s_k \in \{l_1, \ldots, l_L\}, k = \{1, \ldots, t\}\}, \sigma \in \{1, \ldots, F\}$$

where F is the number of all feasible successive sequences, L recall is the number of modes of the system and t is the time-span of the sequence. The parameter $s_k^{(\sigma)}$ denotes the active mode of sequence σ at step k. The time-span of each sequence may vary. However, we can say that $t_{min} \leq t \leq t_{max}$, with t_{min} being the smallest time-span among all switching sequences and t_{max} being the largest time-span among switching sequences.

Finally we define the set $\mathcal{F}$, as the set of all feasible switching sequences for a given SMPL system.

$$\mathcal{F} = \{s^{(\sigma)} \in \mathcal{F}, \text{if and only if}, s^{(\sigma)} \text{is a feasible switching sequence of the SMPL system.}\} \tag{5-16}$$

Whether a switching sequence is feasible or not depends on the system being modelled. Consequently, it is not always possible to determine whether a switching sequence is feasible or not without information about the system being modelled.

5-2-2 Graphical Representation of SMPL systems

Due to the fact that the structure changes over time, SMPL systems cannot be associated with signal-flow graphs. This obstacle can be overcome by using the dynamic graph or the coloured dynamic graph (introduced in Section 2-5). For a given switching sequence over a number of steps k the SMPL system can be modelled as a dynamic graph. For a switching sequence of modes $s^{(\sigma)} = \{\{s_1, \cdots, s_t\}, \quad s_k \in \{l_1, \cdots, l_L\}, \quad k \in \{1, \cdots, t\}\}$, over a number of steps $k > 0$, the dynamic graph of the SMPL system is $\tilde{G}(A^{(s^{(\sigma)})}, B^{(s^{(\sigma)})})$,

$$V(\tilde{G}(A)) = X_0^{t+1} = \cup_{k=0}^{t+1} X^k, \quad X^k = \{x_i^k | i = 1, ..., n\}(k = 0, 1, ..., t+1),$$

$$V(\tilde{G}(B)) = U_1^t = \cup_1^t U^k, \quad U^k = \{u_j^k | j = 1, ..., m\}(k = 1, ..., t),$$

$$E(\tilde{G}(A, B)) = E_0^t = \{(x_j^k, x_i^{k+1}) | A_{ij}^{(s_k)} \neq 0; k = 0, 1, ..., t\} \cup \{(u_j^k, x_i^{k+1}) | B_{ij}^{(s_k)} \neq 0; k = 1, ..., t\}$$

Consider an SMPL system with two modes and system matrices

$$A^{(1)} = \begin{bmatrix} 1 & \varepsilon \\ \varepsilon & 1 \end{bmatrix}, \quad B^{(1)} = \begin{bmatrix} 0 \\ \varepsilon \end{bmatrix} \tag{5-17}$$

$$A^{(2)} = \begin{bmatrix} 1 & \varepsilon \\ 1 & \varepsilon \end{bmatrix}, \quad B^{(2)} = \begin{bmatrix} \varepsilon \\ 0 \end{bmatrix} \tag{5-18}$$

The dynamic graph of this system for a sequence $s = \{l_1, l_2, l_1\}$ is depicted in Figure 5-3.

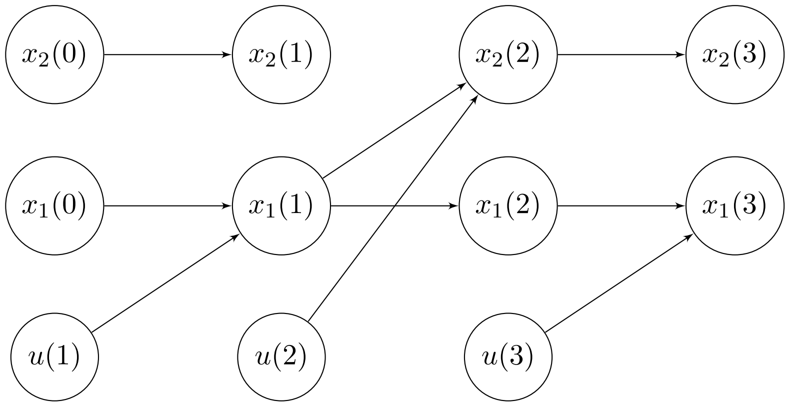

Figure 5-3: A dynamic graph for the SMPL system of (6-20)-(6-21).

Chapter 6

Structural Controllability for MPL and SMPL systems.

We have seen in Chapter 3 that directed graphs have been used in conventional linear system theory for the purpose of defining and deriving the property of structural controllability. In Section 6-1 an overview of the current definitions that exist for controllability in MPL systems will be given. In Section 6-2 controllability will be defined for max-plus linear systems and then the notion will be further extended to SMPL systems in Section 6-3.

6-1 Introduction

As was seen in the previous chapter, being able to determine whether a system is structurally controllable and/or observable is of great importance as they are sufficient conditions for a system to be stabilizable. However, due to the very nature of MPL systems it is not possible to define controllability in the same way as for conventional linear systems. Controllability in conventional algebra implies that one can steer any state to any value within the vector space of the controllability matrix. However, due to the nature of max-plus algebra it is impossible for the vector space of a max-plus matrix (*semimodule*) to include all of $\mathbb{R}^n_{max}$. In fact, semimodules of the controllability matrix of max-plus linear systems can be associated with a cone structure over the $\mathbb{R}_{max}$ semiring, as was seen in 4-4 as, due to the lack of an inverse operation for max-plus addition, it is unfeasible to decrease the values of states. This can be seen to have a clear physical meaning, the states of max-plus linear systems represent event timings making it impossible for a state to decrease in value as that would represent going back in time.

This created the need to define controllability and structural controllability in a different way for MPL and SMPL systems. Baccelli et all [2] defined structural controllability and observability in terms of a timed event graph, while Cofer [26] defines states to be controllable if they can be arbitrarily delayed. Gaubert [27] said that a state is reachable only if it belongs to the semimodule of the controllability matrix. However, computing whether a state belongs to the semimodule of a matrix is a computationally difficult task.

The definitions for structural controllability of Baccelli et all. and controllability of Cofer are in fact the same, but stated differently. In contrast, Gaubert's definition of controllability is more relevant to specific states than the whole system and is more in line with a geometric approach. In the sequel we will focus more on the definition of controllability as given by Baccelli et all and Cofer while, in subsections 6-2-1 and 6-2-2 we will focus more on Gaubert's definition of controllability by making use of the method described in 4-4 for obtaining the span of matrices.

6-2 Controllability and Structural Controllability for MPL systems.

Baccelli et all. defined controllability in terms of a timed event graph and required that every internal transition (equivalent to a state vertex) can be reached by a path from at least one input transition (equivalent to an input vertex). The consequence of this is that the synchronization constraints imposed on all states are a function of the input. If all the synchronization constraints of all the states are a function of the input (or inputs) then they can be influenced by the input and thus they can be arbitrarily delayed, which is the definition of controllability given for MPL systems by Cofer. Notice that no distinction is made based on whether these states can be influenced independently. These two definitions also have an algebraic interpretation. By defining the reachability matrix for the k^{th} event as,

$$K_k = [B \quad A \otimes B \quad \cdots \quad A^{\otimes k-1} \otimes B], \quad k \in \mathbb{N}^+ \tag{6-1}$$

the states of the MPL system at the k^{th} event can be expressed as

$$x(k) = A^{\otimes k} \otimes x(0) \oplus K_k \otimes U_k \tag{6-2}$$

where, $U_k = [u^T(k) \quad u^T(k-1) \cdots u^T(1)]^T$. Another important differentiation of MPL systems compared to LTI systems is the fact that, unlike conventional systems, the semimodule of the reachability matrix $K_k, k = n-1$, for a system with state matrix $A \in \mathbb{R}^{n \times n}_{max}$, may not actually represent the whole semimodule that can be achieved. The cause of this is that if the columns of the B matrix do not enter the eigenspace of the A matrix, then they will enter a period which could infinitely increase the semimodule as $k \to \infty$. The meaning of this is that if a certain state $x(k)$ does not belong to the semimodule at K_k, this does not mean it will not belong to the semimodule $K_{k+c}, c \geq 1$. Consequently, unlike LTI systems where controllability is a property of the system, for MPL systems it is a property of the state.

The definition of structural controllability has a clear implication on the reachability matrix. If every state vertex can be reached by an input vertex then the reachability matrix is row-finite. A matrix is termed row-finite if there exists at least one finite element in every row. That is, for a matrix $V \in \mathbb{R}_{max}^{n \times m}$ for each row $i = 1, 2, \cdots n, \quad \bigoplus_{j=1}^{m} v_{ij} \in \mathbb{R}$. Subsequently, a system is structurally controllable, if and only if, the reachability matrix is row-finite.

Row-finiteness of the reachability matrix has a direct implication on the signal-flow graph of the MPL system. If the reachability matrix is row finite then every state vertex of the system can be reached by a path originating in an input vertex. Recall that the max-plus power of a matrix $A^{\otimes k}$, shows the max weight of paths of length k that exist in the communication graph of A. Furthermore, by definition the communication graph of A is a subgraph of the signal-flow graph of the MPL system. As a result, one can now deduct that if $[A^{\otimes k} \otimes B]_{1m}$ has a finite value in the first row, then a path of length $(k+1)$ exists from an input vertex u to the first state of the system x_1.

Definition 6.1. *A MPL system of the form (5-1) is termed to be structurally controllable if every state $x_i, i \in \{1, ..., n\}$ can be influenced by at least one input $u_j, j \in \{1, ..., m\}$.*

Accordingly, Theorem 6.2 in the sequel will provide sufficient conditions for structural controllability of a MPL system.

Theorem 6.2. *For an MPL system of the form (5-1), the following three conditions (1-3) are equivalent*

1. *The MPL system (5-1) is structurally controllable.*
2. *The reachability matrix K_n is row-finite.*
3. *In the signal-flow graph of (5-1), each state vertex can be reached from a path originating at an input vertex.*

Theorem 6.2 allows us to express structural controllability in terms of the signal-flow graph. Structural controllability, as is the case with LTI systems, is an inherent property of the system and thus does not depend on the initial condition or the weights of the edges. It only depends on the structure of the system or, in other words, what is of importance in terms of structural controllability is whether an edge or path exists between two vertices and not the weight of that path or edge. Therefore, the property of structural controllability can also be applied to stochastic MPL systems, as long as the stochasticity is associated with the weights of the edges (or values of the A and B matrices) and not with the structure of the system.

Moreover, an important point worth mentioning is that structural controllability does not provide information on whether we can achieve a certain value for the states given an initial condition but, on whether we can influence the states regardless of the initial condition. This is a very important distinction to make. The problem of deriving the set of states over $\mathbb{R}_{max}$ that can be achieved given an initial condition (or a set of initial conditions) has been tackled by Katz and Gaubert [28, 29] and more recently by Adzkiya et al [30, 31, 32]. Katz implements a geometric approach with the purpose of determining the maximal set of initial conditions for which the states always stay in that semimodule, while Adzikya makes use of the one to one correspondence between MPL systems and Piece-wise Affine (PWA) models to calculate both the forwards and backwards reachability sets.

The next point of interest with regards to this thesis is to determine the set of accessible states. The term accessible states refers to the values the elements of a state vector x can achieve. For example, it would be beneficial to determine whether a certain desirable state $x_{des} = (x_1, \ldots, x_n)$ can be accessed (achieved). This can be determined by checking whether the desirable state belongs to the semimodule of the reachability matrix. Deciding whether a state is accessible differs between autonomous and non-autonomous systems so a distinction will be made.

6-2-1 Accessibility for Autonomous Systems

The evolution of the state for autonomous systems is described by the following equation.

$$x(k) = A \otimes x(k-1) \tag{6-3}$$

with $A \in \mathbb{R}_{max}^{n \times n}$.

Determining the set of accessible states for autonomous systems is significantly easier than for non-autonomous systems. This occurs because for a square matrix the semimodule can be efficiently determined by the method described in Section 4-4. As a result, determining whether a desired state is accessible is equivalent to determining whether the desired state belongs to $span(A)$. The only requirement for the application of the method is that the state matrix A is row-finite which is nearly always the case for real life systems.

In addition to this, it does not matter whether the columns of the A matrix are linearly independent. Even if all the columns belong to the same extreme ray, the method will still determine the semimodule generated by that ray. Subsequently, for the purpose of determining the set of accessible states it is sufficient to calculate the polyhedra produced by the columns of the state matrix. Some examples follow for clarification.

Example 6.1. Consider the autonomous system described by the following state matrix A,

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

It can be observed that the third column is just the max-plus addition of the first and second column. The permutation $A^{\{2,1,3\}}$ is applied and then the matrix is brought to definite form.

$$A^{'} = \begin{pmatrix} 0 & -1 & 0 \\ -2 & 0 & -1 \\ -1 & 1 & 0 \end{pmatrix}$$

with the Kleene star being

$$A^{'*} = \begin{pmatrix} 0 & 1 & 0 \\ -2 & 0 & -1 \\ -1 & 1 & 0 \end{pmatrix}$$

Therefore, the closure of the matrix is defined by,

$$-2 \leq x_2 - x_1 \tag{6-4}$$
$$-1 \leq x_3 - x_1 \leq 0 \tag{6-5}$$
$$1 \leq x_3 - x_2 \leq 1 \tag{6-6}$$

The second column (of A or $A^{'}$, recall that the columns of both matrices belong to the the same extreme ray) violates inequality (6-5), as $x_3 - x_1 = 2$, and satisfies the equality $x_3 - x_2 = 1$ (which for this particular example is always the case as $x_3 - x_2$ is always equal to one). So, overall the set of accessible states and the span of the system matrix are given by

$$span(A) = \begin{cases} -2 \leq x_2 - x_1 \\ -1 \leq x_3 - x_1 \leq 2 \\ 1 \leq x_3 - x_2 \leq 1 \end{cases}$$

□

Example 6.2. Consider the autonomous system described by the following state matrix A,

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{pmatrix}$$

It can be easily seen that all the columns are max-plus multiples of each other. There is no need for a permutation and the definite form of the matrix and its Kleene star are

$$A^{'} = \begin{pmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{pmatrix}$$

$$A^{'*} = \begin{pmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{pmatrix}$$

The inequalities defining the span are,

$$span(A) = \begin{cases} 1 \leq x_2 - x_1 \leq 1 \\ 2 \leq x_3 - x_1 \leq 2 \\ 1 \leq x_3 - x_2 \leq 1 \end{cases}$$

The fact that in this case the inequalities are in fact equalities is attributed to the fact that the cone of the matrix A is of dimension one. □

The main benefit of this method is that it enables a one-shot computation of the span of a matrix and therefore, its set of accessible states. Moreover, it may also provide information on the lower and upper bound of the qualitative difference between different state vertices. This is not to say that if no bound exists the difference will grow asymptotically. However, if a bound does exist then the divergence between distinct state vertices remains bounded. Concluding a priori whether or not the state vertices remain bounded with respect to each other is of particular interest, as systems in which one state vertex grows infinitely larger than the other state vertices are unstable. The following lemma provides a condition under which all state vertices remain bounded with respect to each other.

Lemma 6.3. *All state vertices, $x_1, \ldots, x_n$, of a matrix $A \in \mathbb{R}_{max}^{n \times n}$ remain bounded with respect to one another if and only if the directed graph defined by the common edges of the definite form and its Kleene star is strongly connected. That is, the graph $G(A^{'} \cup A^{'*}) = (V(A), E(A^{'} \cup A^{'*}))$, $E(A^{'} \cup A^{'*}) = \{(i,j) \in E(A^{'}), E(A^{'*}) | w(i,j)^{'} = w(i,j)^{'*}\}$ is strongly connected*

Proof. If the directed graph is strongly connected then a path exists from any state vertex to any other state vertex. Recall that edges in the communication graph of the definite form define an inequality of the following form $x_j - x_i \geq w_{ji}$. So if a path exists $x_i \to x_{i+1} \to x_{i+2}$ a relation of the following form holds, $x_{i+2} \geq x_i + w(i, i+2)$. Since the graph is strongly connected such a relation holds for all state vertices. As a result a bound exists between all state vertices. Even though these inequalities only define the closure of the span and not the span in its entirety, it does not alter the result. This is due to the fact that only the closure can remain unbounded, as extreme points with coordinates at $-\infty$ cannot by definition contribute to the closure. □

Nonetheless, it is important to again highlight that the difference between the state vertices will not necessarily grow to infinity if the graph is not strongly connected. This depends on the initial condition and the eigenspace of the matrix among other things.

Another important benefit of the method is that it may allow for a computation of the range of values for certain state vectors, if a desired value is wanted for a specific state vertex. Consider for instance, that in Example 6.1 $x_2 = 1$ is desired. A computation will show that if $x_2 = 1$ then x_3 will be equal to 2 and x_1 will range between zero and three $x_1 \in [0, 3]$. If furthermore, a set of constraints for a given system was expressed as a polyhedra or square matrix, then it is attainable to decide whether the system can comply with the constraints.

6-2-2 Accessibility for Non-Autonomous Systems

The evolution of the state of the system for non-autonomous systems is described by an equation of the form

$$x(k) = A \otimes x(k-1) \oplus B \otimes u(k) \tag{6-7}$$

with $A \in \mathbb{R}_{max}^{n\times n}$ and $B \in \mathbb{R}_{max}^{n\times m}$. Moreover, it will be assumed that the system is always structurally controllable and subsequently the reachability matrix is row-finite.

As stated in the sequel the matrix K_k (6-1) is called the reachability matrix. If the initial condition is equal to ε, $x(0) = \varepsilon$, then the set of accessible states for an event $k \in \mathbb{N}^+$ is equal to the span of the reachability matrix K_k for that event step [27]. Nevertheless, if the initial condition is not ε then depending on the choice of control the states may lie in a region between the span of the state matrix A and the span of the reachability matrix. If no control is applied then the set of accessible states is characterized by the span of the state matrix. If control is applied then, depending on the system and the control, some state vertices may be influenced, all state vertices may be influenced or no state may be influenced. State vertices that are influenced by a control input tend towards the span of the reachability matrix. What is meant by influence is not that the states vertices can be reached by an input vertex, but that the control input alters the value of the state vertex in some next time step. More practically, that for $x_i(k), i \in V(A)$ the following relationship holds,

$$a_{i.} \otimes x_i(k-1) < b_{i.} \otimes u \tag{6-8}$$

$a_{i.}$ and $b_{i.}$ denoting the i^{th} row of the A and B matrix respectively.

Another challenge with regards to determining the set of accessible states for non-autonomous systems is that the dimension of the cone generated by the columns of the reachability matrix may grow with every time event. Consequently, it could be the case that in order to determine the set of accessible state for every time event, the span of the reachability matrix has to also be calculated at every time event. The reachability matrix can be interpreted as a set of autonomous systems with state matrix A and initial conditions the columns of the input matrix B. In fact, for every initial condition of a sequence of the form (6-3) one of two things may happen. Either the starting vector enters the eigenspace, the semimodule defined by the eigenvectors of a matrix, of $span(A)$, or it enters a periodic regime [12]. A periodic regime is a set of vectors $x^1, \ldots, x^l \in \mathbb{R}^n_{max}, l \geq 1$ such that for a scalar $\alpha \in \mathbb{R}$ the following relation holds

$$\alpha \otimes x^1 = A \otimes x^l \quad \text{and} \quad x^{i+1} = A \otimes x^i, \quad i \in [1, l-1] \tag{6-9}$$

Determining whether a vector enters the eigenspace or a periodic regime is a computationally difficult task, especially for large order systems, and it is associated with the cyclicity of the critical graph of a matrix and the number of strongly connected components of a matrix [33]. Furthermore, there is a significant difference associated with whether the initial condition reaches the eigenspace or not. If it does reach the eigenspace, then all the vectors thereafter will be max-plus multiples of the eigenvector and thus the dimension of the cone is equal to the number of events it takes for the starting vector to reach the eigenspace. However, should the initial condition enter a periodic regime, depending on the periodic regime, the dimension of the cone could continue increasing. Nevertheless, note that if $span(A)$ remains bounded (if Lemma 6.3 holds for A) then there is **no** periodic regime for any initial condition that keeps increasing the dimension of the cone, as the differences between the elements are bounded. This is not to say that if $span(A)$ is not bounded a periodic regime exists that increases the dimension of the cone. Rather, if $span(A)$ is bounded then it is guaranteed that no periodic regime exists that increases the dimension. This is an important distinction to make.

So, if the initial condition is $x(0) = \varepsilon$, the set of accessible states is determined by the span of the reachability matrix. If the dimension of the semimodule defined by the columns of the reachability matrix does not grow indefinitely, i.e. if the columns of B either enter the eigenspace of A or enter a periodic regime that does not augment the cone or if A satisfies Lemma 6.3, then the set of accessible states is solely determined by the span of the reachability matrix $K_{\kappa+1}$, for all $k \geq \kappa$. In this case κ signifies the time event after which the dimension of the cone remains unchanging. For all matrices and for any initial condition the following relation holds

$$\kappa \leq s(A) \tag{6-10}$$

$s(A)$ signifying the cyclicity of the critical graph of A.

As a consequence, in order to determine the set of accessible states for a system of the form (6-7) with initial condition $x(0) = \varepsilon$, and with a reachability matrix that has a fixed dimension as $k \to \infty$, one has to calculate the span of the matrix $K_{s(A)+1}$. A method for obtaining the span of such matrices was presented in Section 4-4-1. Again, depending on whether the objective is to show that the buffer levels remain bounded or if the objective is to determine the range of values for the state vertices at a given time event, the min-plus method can be used or the submatrix method can be used respectively. Since, it is highly unlikely that systems of this kind will fulfill the condition $m >> n$ (due to the reasons explained in the prequel), for systems of this kind it is preferable and less computationally expensive to determine the exact span of the reachability matrix as it provides more information than the min-plus span.

However, if the case is that a(the) column(s) of B enters a periodic regime that keeps augmenting the cone then the set of accessible state for time event k is determined only by $span(K_k)$. Hence, as k keeps increasing so does the size of the reachability matrix. So in contrast to the previous case, for k large it could be preferable to only approximate its span by using the min-plus method. Obviously, this depends on the system, its properties (such as the order of the system) and the objective in hand.

Example 6.3. Consider the MPL system described in Example 5.1

$$x(k) = \begin{pmatrix} 5 & \varepsilon & \varepsilon \\ \varepsilon & 6 & \varepsilon \\ 11 & 12 & 3 \end{pmatrix} \otimes x(k-1) \oplus \begin{pmatrix} 2 \\ 0 \\ 8 \end{pmatrix} \otimes u(k)$$

The span of matrix A is given by

$$x_3 - x_1 \geq 6$$
$$x_3 - x_2 \geq 6$$

Since, $G(A^{'*})$ is not strongly connected the closure of the span is unbounded and a periodic regime **may** exist that keeps augmenting the dimension of the cone. In fact, assuming no control and initial condition $x(0) = [0\ 10\ 0]^T$, we have $x(100) = (500\ 610\ 616)^T$ and $x(200) = (1000\ 1210\ 1216)^T$. As can be seen even though the difference between x_2 and x_3 remains bounded they both diverge from x_1 as k grows larger. We now consider the reachability matrix,

$$K_8 = \begin{pmatrix} 2 & 7 & 12 & 17 & 22 & 27 & 32 & 37 & 42 \\ 0 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 \\ 8 & 13 & 18 & 24 & 30 & 36 & 42 & 48 & 54 \end{pmatrix}$$

It is evident that the after $k = 2$ the column of B enters a periodic regime with $\alpha = (5\ 6\ 6)^T$. Consequently, it is not possible to determine a global set of accessible states as this set depends on each time event and is not an inherent property of the system. Nonetheless, after calculating $span^{-}(K_8^{\clubsuit})$ we obtain

$$span^{-}(K_8^{\clubsuit}) = \begin{cases} -2 \leq x_2 - x_1 \leq 6 \\ 6 \leq x_3 - x_1 \leq 12 \\ 6 \leq x_3 - x_2 \leq 8 \end{cases}$$

Which shows that the span of the reachability matrix remains bounded, even though the bounds change with every time event. So, if for instance the objective of the designer is for $x_2 - x_1 \leq 4$ (i.e. processing unit 2 starts working at most 4 time units after processing unit 1) one can deduct that a control input must be applied at least every 6 time events (assuming $x(0) = \varepsilon$) as

$$span^{-}(K_{6}^{\clubsuit}) = \begin{cases} -2 \leq x_2 - x_1 \leq 4 \\ 6 \leq x_3 - x_1 \leq 10 \\ 6 \leq x_3 - x_2 \leq 8 \end{cases}$$

Because the bounds change with every time event, for $K_{\kappa}\kappa > 6$, the bounded region will exceed $x_2 - x_1 \leq 4$ (e.g. for $span^{-}(K_{8}^{\clubsuit})$ we have $x_2 - x_1 \leq 6$) and thus the possibility will exist that the objective cannot be met. However, since the boundary of $span^{-}(K_{6}^{\clubsuit})$ is $x_2 - x_1 \leq 4$, the objective of $x_2 - x_1$ will always be met if a control input is applied at least every six time events. □

The case where the initial condition is $x(0) \neq \varepsilon$ is a bit more complicated than when the initial condition is equal to ε. As stated previously, this occurs because the state vertices may be in $span(A)$ or $span(K_k)$ depending on whether (6-8) is satisfied for each state vertex. Accordingly, the state $x(k)$ only belongs to the span of the reachabilitty matrix if condition (6-8) is satisfied for all state vertices $x_i(k)$. Therefore, if (6-8) holds for all state vertices the set of accessible states can be determined in the same way as for $x(0) = \varepsilon$, by calculating the span of the reachability matrix K_k. If condition (6-8) is not satisfied for all state vertices then the set of accessible states is determined by calculating $span(KA)$, where

$$KA = [A \quad B] \tag{6-11}$$

The reason for only considering the input matrix B and not the reachability matrix K_k is because, with the exception of the columns of B, all the other columns of the reachability matrix already belong to $span(A)$ as they are max-plus multiples of A ($[A \otimes B \ldots A^{\otimes k} \otimes B]$). As a result, $span(KA)$ includes not only the spans of A and B but also all the linear combinations between the columns of B and the columns of A. This regions has to be considered as it may be the case that $x(k)$ belongs to neither $span(A)$ or $span(B)$ but to $span(KA)$. This will be illustrated in the following example.

Example 6.4. Consider an MPL system with state and input matrix as given below

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}$$

It is fairly straightforward to see that $span(A) = \{x_1 = x_2 = x_3\}$ (as the cone is of dimension one) and that $span(B) = \{x_1 = x_2 - 1, x_2 = x_3 - 1, x_1 = x_3 - 2\}$ (as the cone is again of dimension 1).

Now, assume initial condition $x(0) = (1 \quad 1 \quad 1)^T$ and control input $u(1) = 0$. The state at time-event one is then

$$x(1) = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$$

It can be immediately deducted that $x(1)$ violates both $span(A)$ and $span(B)$. The matrix KA is now taken into consideration.

$$KA = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

In this particular case, since the first three columns of the matrix are the same we can ommit one column to obtain

$$KA = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 2 \end{bmatrix}$$

The span is calculated in a similar way as in the previous examples and we obtain

$$span(KA) = \begin{cases} (0 \leq x_2 - x_1 \leq 0) \vee ((0 \leq x_2 - x_1 \leq 1) \wedge (x_3 - x_2 = 1) \\ 0 \leq x_3 - x_2 \leq 1 \\ 0 \leq x_3 - x_1 \end{cases}$$

It can be seen that indeed $x(1)$ belongs to $span(KA)$. □

By considering the span of matrix KA it is possible to determine the set of accessible states of the system regardless of the initial condition. This is particularly useful when no information is provided with regard to whether (6-8) is satisfied.

In conclusion, the method described in Section 4-4 can be applied to autonomous systems and non-autonomous with the purpose of determining the set of accessible states, regardless of the initial condition. Accurately describing the set of accessible states depends on the structure of the system, the initial condition and the weights of the edges associated with input vertices. Nevertheless, it is always possible to determine the set of inaccessible states. Subsequently, this method is more suited to modelling MPL systems for which the designer may have a choice over the selection of certain parameters, and can thus modify them if necessary to fit some constraints or desired dynamics. Furthermore, for non-autonomous systems this methodology can provide information on the time events where control inputs have to be applied with the purpose of satisfying some constraints. On a final note, it is also worth mentioning that this method also shows whether the difference between the elements remains bounded and whether they will remain bounded after a control input has been applied. Recall that systems in which the difference between the elements of the state vector grow asymptotically are unstable. By determining whether the span of the matrix is bounded or not, it is possible to conclude that the system will remain stable with respect to the buffer zones between the state vertices.

6-3 Structural Controllability for SMPL systems

Switching max-plus linear systems provide an added level of complexity as the structure of the system changes over time. Moreover, it could be the case that even though the modes of the system are uncontrollable as stand alone MPL systems, the overall SMPL system is controllable for a certain switching sequence. Consequently, it is meaningless to study the controllability properties of each individual mode as they do not provide any information with regards to the overall SMPL system. The idea behind structural controllability for SMPL systems is similar to MPL systems. A SMPL system is structurally controllable if all state vertices of the last mode of a sequence can be reached by a path originating at an input vertex (not necessarily an input vertex of the same mode).

Before proceeding further we will have to distinguish between structural controllability for finite SMPL systems and for non finite SMPL systems. A structurally finite system is a system with the following property [25].

Lemma 6.4. *An SMPL system of the form (5-7)-(5-8) is structurally finite if and only if the matrix*

$$H^{(l)} = \begin{bmatrix} A^{(l)} & B^{(l)} & \varepsilon \\ \varepsilon & \varepsilon & C^{(l)} \end{bmatrix} \tag{6-12}$$

is row-finite for all modes, $\forall l \in \{1, ..., L\}$.

It is important to remark that usually physical systems are structurally finite [25]. Nevertheless, the case of non-structurally finite systems will still be considered. Whether a SMPL system is structurally finite or not has implications on its dynamic graph and as a result, also on its controllability.

6-3-1 Structurally finite SMPL systems.

Boom and De Schutter [25] define structurally finite SMPL systems to be structurally controllable in the following way. For all feasible successive switching sequences, namely $\forall s^{(\sigma)} \in \mathcal{F}, \quad \sigma \in \{1, ..., F\}$, the matrices

$$\Delta^{(\sigma)}(s^{(\sigma)}) = [A^{(s_t^{(\sigma)})} \otimes A^{(s_{t-1}^{(\sigma)})} \otimes \cdots \otimes A^{(s_2^{(\sigma)})} \otimes B^{(s_1^{(\sigma)})} \quad \cdots \quad A^{(s_t^{(\sigma)})} \otimes B^{(s_{t-1}^{(\sigma)})} \quad B^{(s_t^{(\sigma)})}] \tag{6-13}$$

are row-finite. If this is case then the system is termed structurally controllable.

The fact that a system is structurally finite has a direct consequence on the dynamic graph of the system. From Lemma 6.4 it is obvious that, for the system to be structurally finite, the matrix $[A^{l(k)}, B^{l(k)}]$ must be row-finite for all modes. If all the matrices $[A^{l(k)}, B^{l(k)}]$ are row-finite, then all the state vertices of the dynamic graph (with the exception of the initial state vertices $x(0)$) have an incoming arc from either a state vertex of the previous time event or from an input vertex. Subsequently, one can infer that if after a certain number of events $k > 0$ all the state vertices can be reached by some input vertex, then all the state vertices for time events $\kappa \geq k$ can always be reached by a path originating from an input vertex. Because the system is structurally finite, all state vertices for events $\kappa \geq k$ will always have incoming paths from either a state vertex, for which it is already known that it can be reached from an input vertex, or from an input vertex of that time event.

Furthermore, the switching sequence must also be accounted for. As the switching sequence determines how the structure of the system changes over time it plays a vital role in determining whether the system is structurally controllable. Therefore, for structurally finite SMPL systems we will define two types of structural controllability. The first type of structural controllability will determine whether a system is structurally controllable for a given switching sequence and the second type of structural controllability will determine whether a system is structurally controllable for all switching sequences.

Definition 6.5. *An SMPL system of the form (5-7) is said to be* ***sequentially structurally controllable*** *if for a given feasible switching sequence of the modes* $s^{(\sigma)} = \{[s_1^{(\sigma)}, \cdots, s_t^{(\sigma)}]^T \quad | \quad s_k^{(\sigma)} \in [l_1, \cdots, l_L], \quad k = \{1, \cdots, t^{(\sigma)}\}\}$*, all state vertices at* $t = k$ *can be reached by a path originating at an input vertex.*

Definition 6.6. *An SMPL system is* ***structurally controllable*** *if it is sequentially structurally controllable for all feasible switching sequences. That is, it is sequentially structurally controllable* $\forall s^{(\sigma)} \in \mathcal{F}$*.*

Since the system is structurally finite, if it is sequentially structurally controllable for a given switching sequence, then it will remain structurally controllable for all time events $\kappa \geq t$.

The coloured dynamic graph will be used with the purpose of giving a sufficient condition for structural controllability of structurally finite SMPL systems. With the coloured dynamic graph, as opposed to the signal-flow graph, we are able to model the change of structure in a system over time. This is very advantageous as it grants the opportunity to check whether all states are reachable after a finite amount of time. As shown, this is equivalent to showing that the sequence is structurally controllable.

In the coloured dynamic graph the colours are associated with the input(s) of each mode. For example, the set $col_{1,t}^{(\sigma)}$ (colour 1 of switching sequence σ) includes all the state vertices of mode s_t (the last mode of the sequence) that can be reached by a path originating from an input vertex of mode s_1. In the same way the rest of the colours are defined

$$col_{i,t}^{(\sigma)} = \{x_j(t) \in col_{i,t}^{(\sigma)} \quad | \quad \mathcal{P}(u_i, x_j) \neq \emptyset, \quad j \in \{1, ..., n\}, \quad i \in \{1, ..., t^{(\sigma)}\}\} \tag{6-14}$$

where $\mathcal{P}(u_i, x_j)$ denotes the set of paths beginning from input vertices of the active mode at the k^{th} time event and terminating at a state vertex x_j. The number of colours is equal to the time-span of the sequence. We can now say that

Theorem 6.7. *For an SMPL system of the form (5-7) and a given switching sequence $s^{(\sigma)} \in \mathcal{F}$, the following conditions (1) to (3) are equivalent:*

1. *The system (5-7) is sequentially structurally controllable for sequence $s^{(\sigma)}$.*
2. *The union of the sets of all colours associated with the coloured dynamic graph $\tilde{G}(A^{s_{t+1}^{(\sigma)}}, B^{s_t^{(\sigma)}})$ of (5-7), includes all state vertices.*
$$col^{(\sigma)} = col_{1,t}^{(\sigma)} \cup col_{2,t}^{(\sigma)} \cup \cdots \cup col_{t,t}^{(\sigma)} = \{1, \cdots, n\} \tag{6-15}$$
3. *The matrix*
$$\Delta(s^{(\sigma)}) = [A^{(s_t^{(\sigma)})} \otimes A^{(s_{t-1}^{(\sigma)})} \otimes \cdots \otimes A^{(s_2^{(\sigma)})} \otimes B^{(s_1^{(\sigma)})} \quad \cdots \quad A^{(s_t^{(\sigma)})} \otimes B^{(s_{t-1}^{(\sigma)})} \quad B^{(s_t^{(\sigma)})}] \tag{6-16}$$
is row-finite.

Proof Conditions (1) and (3) are equivalent from [25]. As a result we only need to show that conditions (2) and (3) are also equivalent.

If the union of the colour sets contains all the vertices, then each state vertex $x_j(t)$ can be reached by a path originating from an input vertex. Assume the set $col_{1,t}$ (indicating all the state vertices at time event t that can be reached by an input from s_1) contains only one state vertex, x_j, then state vertex, x_j, can be reached from the input of the first mode. This means that for a time span t the state vertex, x_j, can be influenced from the input of the first mode. This in turn implies that the vector

$$[A^{(s_t^{(\sigma)})} \otimes A^{(s_{t-1}^{(\sigma)})} \otimes \cdots \otimes A^{(s_2^{(\sigma)})} \otimes B^{(s_1^{(\sigma)})}]$$

has a finite entry at the j^{th} position, $[A^{(s_t^{(\sigma)})} \otimes A^{(s_{t-1}^{(\sigma)})} \otimes \cdots \otimes A^{(s_2^{(\sigma)})} \otimes B^{(s_1^{(\sigma)})}]_{j.} \neq \varepsilon$.

Subsequently, if all state vertices can be reached from an input vertex of some mode over a time span t then the matrix

$$[A^{(s_t^{(\sigma)})} \otimes A^{(s_{t-1}^{(\sigma)})} \otimes \cdots \otimes A^{(s_2^{(\sigma)})} \otimes B^{(s_1^{(\sigma)})} \quad \cdots \quad A^{(s_t^{(\sigma)})} \otimes B^{(s_{t-1}^{(\sigma)})} \quad B^{(s_t^{(\sigma)})}]$$

will be row-finite, which is equivalent to (1). □

The Lemma in the sequel, comes as a natural extension of Theorem 6.7.

Lemma 6.8. *For an SMPL system of the form (5-7) and the set $\mathcal{F} = \{\{s^{(1)}, ..., s^{(F)}\} | \sigma \in \{1, \cdots, F\}\}$, being the set of all feasible successive switching sequences of the system, the following conditions (1) to (3) are equivalent:*

1. *The system (5-7) is structurally controllable.*

2. *The intersection of the sets of colours associated with every feasible switching sequence covers all state vertices.*

$$col^1 \cap col^2 \cap \cdots \cap col^F = \{1, \cdots, n\} \tag{6-17}$$

3. *For all feasible successive switching sequence,* $\forall \sigma \in \{1, \cdots, F\}$*, the matrices*

$$\Delta^{(\sigma)}(s^{(\sigma)}) = [A^{(s_t^{(\sigma)})} \otimes A^{(s_{t-1}^{(\sigma)})} \otimes \cdots \otimes A^{(s_2^{(\sigma)})} \otimes B^{(s_1^{(\sigma)})} \quad \cdots \quad A^{(s_t^{(\sigma)})} \otimes B^{(s_{t-1}^{(\sigma)})} \quad B^{(s_t^{(\sigma)})}] \tag{6-18}$$

are row-finite $\forall \sigma$.

Proof The proof comes as a direct consequence of Theorem 6.7. The equivalence of conditions (1) and (3) is shown in [25]. So, as for the proof of Theorem 6.7, we only need to show the equivalence of conditions (2) and (3).

If the intersection of the sets of all colours covers all the vertices, then the union of the colour sets of all switching sequences will cover all the vertices. This means that, $col^{(\sigma)} = \{1, \cdots, n\}, \quad \forall \sigma$. So, as a consequence of Theorem 6.7, the matrices

$$\Delta^{(\sigma)}(s^{(\sigma)}) = [A^{(s_t^{(\sigma)})} \otimes A^{(s_{t-1}^{(\sigma)})} \otimes \cdots \otimes A^{(s_2^{(\sigma)})} \otimes B^{(s_1^{(\sigma)})} \quad \cdots \quad A^{(s_t^{(\sigma)})} \otimes B^{(s_{t-1}^{(\sigma)})} \quad B^{(s_t^{(\sigma)})}] \tag{6-19}$$

will be row-finite for all feasible switching sequences, $\forall \sigma$. Which is condition (3). □

Example 6.5. Consider an SMPL system with two modes and system matrices

$$A^{(1)} = \begin{bmatrix} 1 & \varepsilon \\ \varepsilon & 1 \end{bmatrix}, \quad B^{(1)} = \begin{bmatrix} 0 \\ \varepsilon \end{bmatrix} \tag{6-20}$$

$$A^{(2)} = \begin{bmatrix} \varepsilon & 1 \\ 1 & \varepsilon \end{bmatrix}, \quad B^{(2)} = \begin{bmatrix} 0 \\ \varepsilon \end{bmatrix} \tag{6-21}$$

Assume we have two feasible switching sequences, namely $\sigma_1 = \{l_1, l_2, l_1\}$ and $\sigma_2 = \{l_2, l_1, l_2\}$. The coloured dynamic graph for switching sequence σ_1 is depicted in Figure 6-1

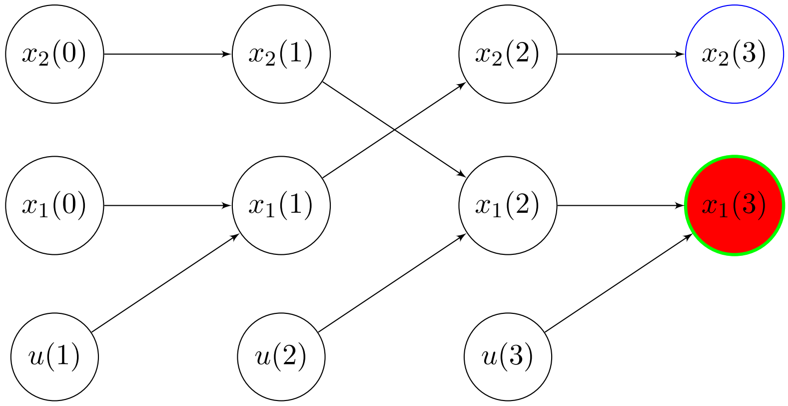

Figure 6-1: An example of a coloured dynamic graph.

We have the following colours, $col_{1,3}^{\sigma_1} = \{x_2\}$ (highlighted in blue on the Figure), $col_{2,3}^{\sigma_1} = \{x_1\}$ (highlighted in green) and finally $col_{3,3}^{\sigma_1} = \{x_1\}$ (highlighted in red). As a result, we have

$$col^{\sigma_1} = col_{1,3}^{\sigma_1} \cup col_{2,3}^{\sigma_1} \cup col_{3,3}^{\sigma_1} = \{x_1, x_2\}$$

and the switching sequence σ_1 is sequentially structurally controllable. Doing the same for σ_2 we obtain

$$col^{\sigma_2} = col_{1,3}^{\sigma_2} \cup col_{2,3}^{\sigma_2} \cup col_{3,3}^{\sigma_2} = \{x_1, x_2\}$$

so, switching sequence σ_2 is also sequentially structurally controllable. Finally, from Lemma 6.8 we have

$$col^{\sigma_1} \cap col^{\sigma_2} = \{x_1, x_2\}$$

so the overall SMPL system is structurally controllable. □

Theorem 6.7 and Lemma 6.8 allow for determining whether a SMPL system is structurally controllable or not from a graph-theoretic point of view. The coloured dynamic graph comes as a natural modelling tool for structurally finite SMPL systems. However, for non-structurally finite SMPL systems the normal dynamic graph must be used.

6-3-2 Non-structurally finite SMPL systems

When an SMPL system is not structurally finite then the matrices $[A^{l(k)}, B^{l(k)}]$ will not be row-finite for all modes. The interpretation of this, in contrast to structurally finite systems, is that even if a system is structurally controllable after a finite number of steps $k > 0$, this does not guarantee that it will remain structurally controllable for time events $\kappa > k$. Because not all matrices $[A^{l(k)}, B^{l(k)}]$ are row-finite, some state vertices do not have an incoming arc. If a state vertex does not have an incoming arc, then it is unreachable and cannot be influenced. This in turn implies that the system is not structurally controllable. Subsequently, for non-structurally finite SMPL systems, structural controllability is not an inherent property of the system but rather a property indicative of a single time event. With this in mind, structural controllability for non-structurally finite SMPL systems is defined in the following ways.

Definition 6.9. *A non-structurally finite SMPL system of the form (5-7) is* ***structurally controllable*** *at event step k, if all states $x_j(k), j \in \{1, \cdots, n\}$ can be influenced by the inputs.*

Definition 6.10. *A non-structurally finite SMPL system of the form (5-7) is* ***structurally controllable*** *for a period τ, if for a certain interval of events $\tau = \{\{k_1, ..., k_\tau\}, k_i \in \mathbb{N}^+, i \in \{1, \cdots, \tau\}\}$ all states $x_j(k_i)$ can be influenced by the input.*

Due to the properties of non-structurally finite SMPL system it is preferential to model them by normal dynamic graphs. This makes it possible to give sufficient conditions for the structural controllability of the system.

Theorem 6.11. *For a non-structurally finite SMPL system of the form (5-7), and a feasible switching sequence σ the following conditions are equivalent:*

1. *The system (5-7) is structurally controllable at time event k of switching sequence σ.*
2. *On the dynamic graph $\tilde{G}(A^{s_k^{(\sigma)}}, B^{s_k^{(\sigma)}})$ of the SMPL system at time event k, all state vertices $x_j(k)$ can be reached by a path originating at an input vertex.*

Proof If at time event k all state vertices can be reached by a path originating at an input vertex, then they can be influenced by the input. Thus at time event k the system is structurally controllable. □

With the purpose of giving sufficient conditions for structural controllability over a period of events, the following Lemma is given.

Lemma 6.12. *For a non-structurally finite SMPL system of the form (5-7), a feasible switching sequence σ and a period $\tau = [k_1, \cdots, k_\tau]$, the following conditions are equivalent:*

1. *The system (5-7) is structurally controllable for period τ of switching sequence σ.*
2. *On the dynamic graph $\tilde{G}(A^{s_k^{(\sigma)}}, B^{s_k^{(\sigma)}})$ of the SMPL system, all state vertices $x_j(k), \forall k \in \{k_1, \cdots, k_\tau\}$ can be reached by a path originating at an input vertex.*

Proof This lemma is a direct consequence of Theorem 6.11

Remark Notice that Theorem 6.11 and Lemma 6.12 can also be applied to structurally finite systems. This is because they are generalizations of Theorem 6.7 and Lemma 6.8. Nevertheless, for structurally finite SMPL systems with a large order and a large number of modes Theorem 6.7 and Lemma 6.8 are preferred as they are computationally less demanding.

Since non-structurally finite systems imply that either the state matrix or the reachability matrix of a mode are not regular matrices, they nearly never occur in practice. The framework described above however, could also be applied to stochastic SMPL systems where the stochasticity can be associated with the edges of the system.

Chapter 7

Conclusion and Future Work

This thesis began with an overview of some important concepts in graph theory and how they are applied to conventional LTI systems. Following this, max-plus algebra was presented and a graph-theoretical approach was implemented for determining whether MPL and SMPL are structurally controllable. In addition to this, a method was developed for determining the span of max-plus matrices and then applied to autonomous and non-autonomous systems with the purpose of obtaining the set of accessible states. Section 7-1 is centered around the conclusions of this thesis, while Section 7-2 discusses suggestions for future work.

7-1 Conclusion

All in all, throughout this thesis it has been shown that graphs provide for a simple and efficient way to model MPL and SMPL systems. Moreover, as was seen in the case of LTI systems, structural properties of the underlying systems can be extracted from the graph-theoretical models of these systems. Unlike conventional systems, the notion of structural controllability does not provide rank information for the reachability matrices of MPL and SMPL systems, it does however guarantee that these matrices are row-finite. This condition is adequate for the structural controllability of systems over the max-plus semiring.

Furthermore, a method was presented for expressing the span of square matrices as a set of linear inequalities in the conventional sense. For square matrices this was achieved by bringing the matrix to definite form and obtaining its closure in addition to implementing the part of the span that is associated with the generators of the semimodule and their connection to the closure. For non-square matrices the span was obtained by calculating the span of all submatrices associated with the linearly independent columns of the original matrix. This method is simple and efficient for square matrices but becomes more complex for rectangular matrices that have a large number of columns. This can be potentially overcome by calculating an approximation of the span. This is achieved by making use of the min-plus method presented in Section 4-4.

In addition to this, a framework was developed, in conjuction with the method of Section 4-4, for establishing the set of accessible states for autonomous and non-autonomous MPL systems. By using this framework, it is possible to establish the range of values the state vectors can achieve, as well as determining through a graph-theoretic notion whether the qualitative difference between elements of the state vector remain bounded with respect to one another.

7-2 Future Work

Based on the work done in this thesis, recommendations for future work are presented below :

- **Accessible states of SMPL systems.** Accurately obtaining the set of accessible states for SMPL systems is much harder than for MPL systems. This is due to the fact that SMPL systems may not have a fixed reachability matrix, as the reachability matrix in SMPL systems depends on the switching sequence. As a result the set of accessible states varies if the switching sequence also changes. An additional challenge that is present in SMPL systems is the fact that prior knowledge is required regarding the amount of time the system remains in each mode. This is the case because, like for non-autonomous systems, the span associated with the reachability matrix of each mode may vary depending on the number of time events the system remains in that mode. An efficient method for determining the set of accessible states for SMPL systems has not, to our knowledge, yet been developed.
- **Graphical representation of SMPL systems.** In this thesis, the dynamic graph was used for representing SMPL systems in a graph-theoretic manner. It was then used for establishing the property of structural controllability for SMPL systems and can be further extented in the same way for structural observability. However, the edges of the dynamic graph were not weighted. It would be of particular interest if it was possible to determine properties of the SMPL system, such as the maximum autonomous growth rate, through the dynamic graph representation of the SMPL system.
- **Set of observable states.** Another point of potential interest, could be translating the method of Section 4-4, with the purpose of attaining the set of observable states. In such a case, the set of the observable states would be represented as a set of conventional linear inequalities, as was done for the controllable case.

Bibliography

[1] S. Pirzada, "Applications of graph theory," *PAMM*, vol. 7, no. 1, pp. 2070013–2070013, 2007.

[2] F. Baccelli, G. Cohen, G. J. Olsder, and J.-P. Quadrat, "Synchronization and linearity: an algebra for discrete event systems," 1992.

[3] T. J. van den Boom and B. De Schutter, "Modelling and control of discrete event systems using switching max-plus-linear systems," *Control engineering practice*, vol. 14, no. 10, pp. 1199–1211, 2006.

[4] C.-T. Lin, "Structural controllability," *IEEE Transactions on Automatic Control*, vol. 19, no. 3, pp. 201–208, 1974.

[5] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *Nature*, vol. 473, no. 7346, pp. 167–173, 2011.

[6] K. Murota, *Systems analysis by graphs and matroids: structural solvability and controllability*, vol. 3. Springer Science & Business Media, 2012.

[7] K. Reinschke, F. Svaricek, and H.-D. Wend, "On strong structural controllability of linear systems," in *Decision and Control, 1992., Proceedings of the 31st IEEE Conference on*, pp. 203–208, IEEE, 1992.

[8] R. Shields and J. Pearson, "Structural controllability of multiinput linear systems," *IEEE Transactions on Automatic control*, vol. 21, no. 2, pp. 203–212, 1976.

[9] K. Ruohonen, *Graph Theory*. 2013.

[10] K. Murota, *Matrices and matroids for systems analysis*, vol. 20. Springer Science & Business Media, 2009.

[11] R. E. Kalman, "Mathematical description of linear dynamical systems," *Journal of the Society for Industrial and Applied Mathematics, Series A: Control*, vol. 1, no. 2, pp. 152–192, 1963.

[12] B. Heidergott, G. J. Olsder, and J. Van der Woude, *Max Plus at work: modeling and analysis of synchronized systems: a course on Max-Plus algebra and its applications*. Princeton University Press, 2014.

[13] P. Butkovič, *Max-linear systems: theory and algorithms*. Springer Science & Business Media, 2010.

[14] R. Cuninghame-Green, "Lecture notes in economics and mathematical systems," in *Minimax algebra*, Springer Verlag, 1979.

[15] M. Akian, S. Gaubert, and A. Guterman, "Linear independence over tropical semirings and beyond," *Contemporary Mathematics*, vol. 495, p. 1, 2009.

[16] S. Gaubert and R. D. Katz, "The minkowski theorem for max-plus convex sets," *Linear Algebra and its Applications*, vol. 421, no. 2-3, pp. 356–369, 2007.

[17] S. Gaubert and R. D. Katz, "Minimal half-spaces and external representation of tropical polyhedra," *Journal of Algebraic Combinatorics*, vol. 33, no. 3, pp. 325–348, 2011.

[18] R. E. Burkard and E. Cela, "Linear assignment problems and extensions," in *Handbook of combinatorial optimization*, pp. 75–149, Springer, 1999.

[19] P. Butkovic, "Simple image set of (max,+) linear mappings," *Discrete Applied Mathematics*, vol. 105, no. 1-3, pp. 73–86, 2000.

[20] S. Sergeev, "Max-plus definite matrix closures and their eigenspaces," *Linear algebra and its applications*, vol. 421, no. 2-3, pp. 182–201, 2007.

[21] J.-M. Prou and E. Wagneur, "Controllability in the max-algebra," *Kybernetika*, vol. 35, no. 1, pp. 13–24, 1999.

[22] K. R. Baker, "Scheduling a full-time workforce to meet cyclic staffing requirements," *Management Science*, vol. 20, no. 12, pp. 1561–1568, 1974.

[23] B. De Schutter and T. J. van den Boom, "Model predictive control for max-plus-linear discrete event systems," *Automatica*, vol. 37, no. 7, pp. 1049–1056, 2001.

[24] I. Necoara, B. De Schutter, T. J. van den Boom, and H. Hellendoorn, "Stable model predictive control for constrained max-plus-linear systems," *Discrete Event Dynamic Systems*, vol. 17, no. 3, pp. 329–354, 2007.

[25] T. J. van den Boom and B. De Schutter, "Modeling and control of switching max-plus-linear systems with random and deterministic switching," *Discrete Event Dynamic Systems*, vol. 22, no. 3, pp. 293–332, 2012.

[26] D. D. Cofer and V. K. Garg, "Supervisory control of real-time discrete-event systems using lattice theory," *IEEE Transactions on Automatic Control*, vol. 41, no. 2, pp. 199–209, 1996.

[27] S. Gaubert, "Methods and applications of (max,+) linear algebra," in *Annual Symposium on Theoretical Aspects of Computer Science*, pp. 261–282, Springer, 1997.

[28] R. D. Katz, "Max-plus (a, b)-invariant spaces and control of timed discrete-event systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 2, pp. 229–241, 2007.

[29] S. Gaubert and R. Katz, "Reachability and invariance problems in max-plus algebra," in *Positive Systems*, pp. 15–22, Springer, 2003.

[30] D. Adzkiya, B. De Schutter, and A. Abate, "Forward reachability computation for autonomous max-plus-linear systems.," in *TACAS*, pp. 248–262, 2014.

[31] D. Adzkiya, B. De Schutter, and A. Abate, "Backward reachability of autonomous max-plus-linear systems," *IFAC Proceedings Volumes*, vol. 47, no. 2, pp. 117–122, 2014.

[32] D. Adzkiya, B. De Schutter, and A. Abate, "Computational techniques for reachability analysis of max-plus-linear systems," *Automatica*, vol. 53, pp. 293–302, 2015.

[33] J. Mairesse, "A graphical approach of the spectral theory in the (max,+) algebra," *IEEE Transactions on Automatic Control*, vol. 40, no. 10, pp. 1783–1789, 1995.

Glossary

List of Acronyms

MPL	Max-Plus-Linear
SMPL	Switching Max-Plus-Linear
DES	Discrete Event Systems