MAX-PLUS ALGEBRA FOR GENETIC ALGORITHMS

By

Joseph Ackora-Prah B.Sc (Hons), M.Sc

A Thesis submitted to the Department of Mathematics, Kwame Nkrumah University of Science and Technology in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY IN MATHEMATICS

College of Science

August 2012

Declaration

I hereby declare that this submission is my own work towards the PhD and that, to the best of my knowledge, it contains no material previously published by another person nor material which has been accepted for the award of any other degree of the University, except where due acknowledgment has been made in the text.

JOSEPH ACKORA-PRAH
PG 5142610 .... 8/2/13

Student Name and ID Signature Date

Certified by:

....

Supervisor's Name Signature Date

Certified by:

....

Head of Dept. Name Signature Date

Abstract

We investigate the redesigning of the general Genetic Algorithms (GAs) using concepts from max-plus algebra. Our formulation presents a general outlook which affords a comprehensive analysis of genetic algorithms and ensures that maximun fitness function is obtained by summing the functional values of all chromosomes in the search space.

We showed that the population generation dynamics of GAs can be formulated using a max-plus linear recursive equation and this yields a sequence of better solutions for the next generation each time. We illustrated how the non-linear iterative system $x_i^{k+1} = \max_{j \in n}\{a_{ij} + x_j^k\}$ in our genetic algorithm can be linear in the max-plus sense. We note that if the population is too low, the investigation may cover too little of the search space to find the optimum solution. Our model was able to withstand large populations so that the optimum solution would not be trapped in a local optimum. This is shown by the fact that there are no restriction on number of chromosomes in the population.

Our formulation makes use of the stable growth max-plus equation $\lambda^{(k+1)} = \lambda \otimes x^{(k)}$ which normalizes the GA system and makes it stable with constant population. Again, the model addresses some of the disadvantages of GAs.

Our formulation uses equations from max-plus algebra to present a more general model to genetic algorithms and to the best of our knowledge, this is the first mathematical framework in genetic algorithms. The model gives a real understanding of the effects of parameter changes on the properties of the GAs and can be used for analysis, optimization, control and to predict the behavior of the GAs in future works.

Acknowledgements

I would like to acknowledge and extend my heartfelt gratitude to my Supervisor, Professor Anthony Yaw Aidoo of Eastern Connecticut State University (ECSU, USA) who has made the completion of this thesis possible. This thesis would have remained a dream had it not been for his constructive suggestions during the planning and development of this research work. I just cannot find words to thank him for his patient guidance, enthusiastic encouragement and useful critiques of this work.

The completion of this thesis has only been possible due to the support of the administration and staff of Eastern Connecticut State University. I would like to express my appreciation to the President and the entire staff of this great institution. I would also like to offer my special thanks to Dean of Arts and Social Sciences (ECSU) Dr.Cid Carmen, and the Vice-President (ECSU) Dr Rhona Free for their understanding and support at all times. I am particularly grateful for the assistance, motivation and encouragement given to me by Professor Bonsu Osei (Stac), Mathematics and Computer

Science department, ECSU. Prof well done and may God bless you. I would like to express my very great appreciation to Professor Mizan Khan (former Chair), Professor Christian Yankov, Professor Marsha Davies (current Chair) all of Mathematics and Computer Science Department ECSU, USA for their support and contributions.

I am greatly indebted to my colleague, a friend and a father, Dr. Emmanuel Osei-Frimpong (Mathematics Department, KNUST) who initiated this whole idea of studying abroad for me. I also wish to thank Dr. S. K. Amponsah, Professor I. K. Dontwi and Mr. Kwaku Darkwa all of Mathematics Department, KNUST for their inspiration.

I wish to acknowledge the very special support provided by my sweetness, Mrs. Vida Ackora-Prah. I want to say that I am grateful for your prayers and for looking after the children in my absence. Finally, my sincere appreciation goes to all my friends and to God, who made all things possible.

Contents

Abstract i

Acknowledgements iii

Introduction 1

1 Max-plus Algebra 4

1.1 Preliminary Concepts and Algebraic Properties 4

1.2 Max-plus Linear Algebra 9

1.2.1 Max-plus exponential functions 15

1.2.2 Determinants 16

1.2.3 Cramer's Rule 17

1.2.4 Linear Dependence and Independence 18

1.2.5 Graph Theory in Max-plus 20

1.2.6 Max-plus polynomials 22

1.2.7 Solution of Systems of Equations in Max-plus 25
1.2.8 Solution of systems of linear equations and inequalities in max-plus 31
1.2.9 Max-Linear Program with equation and inequality constraints 38
1.2.10 Eigenvalues and Eigenvectors 41
1.2.11 The Transitive Closures 43
1.2.12 Linear Programming Approach to Finding Eigenvalues in Max-plus Algebra 45
1.2.13 Cayley-Hamilton Theorem and the Max-plus Characteristic Equation 49

2 Some Applications of Max-plus Algebra 51
2.1 Introduction 51
2.2 Control Problems 52
2.2.1 Finite and Infinite-time horizon optimal control problems 54
2.2.2 Max-plus Finite Element Method For Optimal Control Problems 57
2.3 Optimal Control Approach for Max-plus Linear Discrete Event Systems (DES) 62

2.3.1 Max-plus linear state space models 62
2.3.2 Model Predictive Control (MPC) 63
2.3.3 Max-plus Linear Input-Output (MPLIO) Systems . . . 64
2.3.4 Model Predictive Control for MPLIO Systems 65
2.3.5 Asymptotic growth rate analysis and Limiting behaviour in max-plus 69

3 Genetic Algorithms 73
3.1 Introduction 73
3.1.1 Initialization and Chromosome Representation 76
3.1.2 Selection Mechanism 77
3.1.3 Genetic Operators 78
3.2 Schema 82
3.2.1 Schema in GA 83
3.2.2 Termination 86

4 A Max-plus Model for Genetic Algorithms 88
4.1 Introduction 88
4.1.1 Max-plus Search Space and Fitness 89
4.1.2 Max-plus GA steps 91

4.1.3 Max-plus Model For Population Dynamics in Genetic Algorithms 95

4.1.4 Eigenvalues And Eigenvectors On Population 99

4.1.5 Convergence of the Max-plus GA 101

4.1.6 The Max-plus GA Complete Model 102

4.1.7 Why Max-plus Genetic Algorithms 104

5 Conclusion, Open Problems and Extensions **108**

5.1 Conclusion 108

5.2 Open Problems and Extensions 110

Introduction

In this thesis, we investigate the redesigning of the general Genetic Algorithms (GAs) using concepts from max-plus algebra. Max-plus algebra is an analogue of conventional linear algebra developed on the operations of $\oplus$ and $\otimes$. The algebraic structure is a semi-ring whose elements are the usual real numbers along with $\epsilon = -\infty$ and $e = 0$, where $\oplus$ represents taking the maximum and $\otimes$ is the standard addition. A GA is an evolutionary heuristic search and optimization algorithm that works by iterative application of evolutionary operators on population of solutions and mimics the basic natural evolution cycle (Holland, 1975). Parameters and operators such as the initial population, the probability and type of crossover, probability and type of mutation, stopping criteria, the type of selection operator and the fitness function affect the performance of the GA. They are also inter-related and form a system.

Max-plus algebra is a relatively new concept in mathematics used mostly by the control theory community. Despite this, several other interesting applications have emerged. We give a brief overview of some of the most interesting applications here.

Quite recently, non-linear models that describe the behavior of discrete event systems have been described by models that are linear in the max-plus algebra (Boom et al., 2004). Previous work of McEneaney, (see McEneaney, 2004) shows that Hamilton-Jacobi-Bellman partial differential equations are based on the max-plus formulation of these problems. Max-plus algebra has also been used as an algebra for optimal control of dynamical systems and several authors have already developed methods to compute optimal control sequences for max-plus-linear discrete event systems, see for example (Baccelli at al., 1992).

In this thesis we examine the dynamics of genetic algorithms in terms of the max-plus algebra. We define a function called the max-plus fitness in Genetic Algorithms, which determines the fittest chromosome in the search space to begin the next generation, propose that the search space is a max-plus idempotent commutative semi-ring and present a novel max-plus linear model to describe the genetic algorithm processes. This model allows us to translate the many properties, concepts and techniques of the genetic algorithms in a simplified manner and offers a compact way of describing the dynamics of the iterative processes in the genetic algorithms. This theoretical approach of analysis provides a useful framework for propagating a population of candidate solutions such that an optimal solution can be obtained in an evolutionary computation process. In addition, it provides an important basis for further analytical studies on the efficiency of the genetic algorithm procedure.

The thesis is organized as follows: In chapter one we discuss some key properties of the max-plus algebra and how these properties are used to analyze some linear algebra concepts. Solutions of systems of equations and inequalities are also introduced. Chapter two presents some of the applications of the max-plus algebra in various fields of mathematics such as control problems, discrete event systems and global convergence. In chapter three, we review GAs as search techniques, which have emerged to meet the global optimization needs in a complex search space. In chapter four, we formulate a max-plus model for the genetic algorithms. Chapter five concludes and suggests open problems and extensions of the study.

Chapter 1

Max-plus Algebra

1.1 Preliminary Concepts and Algebraic Properties

In this chapter, we provide the preliminary definitions, concepts and algebraic properties of the max-plus algebra. We also introduce the concept of matrices, vectors, linear dependence and independence, eigenvalues and eigenvectors in the max-plus algebraic environment. We denote by $\mathbb{N}$, the set of natural numbers, $\mathbb{R}$ the set of real numbers, $\varepsilon = -\infty$, $e = 0$, and $\mathbb{R}_{max} = \mathbb{R} \cup \{\varepsilon\}$. We give an overview of the max-plus algebra by defining the operations $\oplus$ and $\otimes$ by $a \oplus b = \max(a, b)$ and $a \otimes b = a + b$ where $a, b \in \mathbb{R}_{max}$. The set

$\mathbb{R}_{max}$ equipped with the operations $\oplus$, $\otimes$ and the elements ε and e is called a max-plus algebra and is denoted by $\mathcal{R}_{max} = (\mathbb{R}_{max}, \oplus, \otimes, \varepsilon, e)$. We see clearly that $\max(a, -\infty) = \max(-\infty, a) = a$ and $a + (-\infty) = (-\infty) + a = -\infty$ for all $a \in \mathbb{R}_{max}$. We show that $a \oplus \varepsilon = \varepsilon \oplus a = a$ where ε is an additive identity for $\oplus$. Similarly $a \otimes e = e \otimes a = a$ for all $a \in \mathbb{R}_{max}$ and $e = 0$ is the multiplicative identity. Again $a \otimes \varepsilon = \varepsilon \otimes a$ shows that ε is an absorbing element under $\otimes$ in max-plus algebra. We show that $\oplus$ and $\otimes$ in the max-plus algebra obey some of the properties similar to '$+$' and '$\times$' in the conventional algebra. For example, for $a, b, c \in \mathbb{R}_{max}$ we have

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

which means that $\otimes$ is distributive over $\oplus$. The list of the algebraic properties of the max-plus algebra is given below.

1. Commutativity: $\forall a, b \in \mathbb{R}_{max} : a \oplus b = b \oplus a$ and $b \otimes a = a \otimes b$

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

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

4. Existence of a zero element: $\forall a \in \mathbb{R}_{max} : a \oplus \varepsilon = \varepsilon \oplus a$

5. Existence of a unit element: $\forall a \in \mathbb{R}_{max} : a \otimes e = e \otimes a = a$

6. The zero is absorbing for $\otimes$: $\forall a \in \mathbb{R}_{max} : a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon$

7. Idem-potency of $\oplus$: $\forall a \in \mathbb{R}_{max} : a \oplus a = a$

8. Let $m, n \in \mathbb{N}$ and $a \in \mathbb{R}_{max}$, then $a^{\otimes m} \otimes a^{\otimes n} = ma + na = (m+n)a = a^{\otimes(m \otimes n)}$

9. Let $m, n \in \mathbb{N}$ and $a \in \mathbb{R}_{max}$, then $(a^{\otimes m})^{\otimes n} = (ma)^{\otimes n} = nma = a^{\otimes(m^{\otimes n})}$

10. Let $m \in \mathbb{N}$ and $a \in \mathbb{R}_{max}$, then $a^{\otimes m} \otimes b^{\otimes m} = (a \otimes b)^{\otimes m}$

If $a \in \mathbb{R}_{max}$ and $n \in \mathbb{N}$, then $a^{\otimes n} = \overbrace{a \otimes a \otimes \ldots \otimes a}^{n\ times}$. In max-plus algebra exponentiation reduces to the conventional multiplication, that is: $a^{\otimes n} = na$. We note the existence of order under $\oplus$ in max-plus algebra as in the case of the conventional real number system. We state the following definition due to Baccilli (Baccilli et al., 1992):

$$a \leq b \text{ if } a \oplus b = b.$$

and we have Gondran and Minoux (Gondran and Minoux, 1984) requiring that $a = b \oplus c$ and $b = a \oplus d \;\Rightarrow\; a = b$ and in this instance we have $\oplus$

inducing an ordering $\geq$ on the element of the max-plus algebra in either case as $a \geq b \Leftrightarrow a = a \oplus b$ and $a \geq b \Leftrightarrow \exists c \in \mathbb{R}_{max} : a = b \oplus c$. In either case, the relation $\geq$ must be a partial ordering, that is, $a \geq b$ and $b \geq a \Rightarrow a = b$ and $\geq$ is transitive.

Dioid Algebra

Dioid algebra is a powerful mathematical tool that allows a linear description of systems that have non-linear representation. For our purpose, the importance of dioids is seen in chapter two and in particular discrete event systems, where synchronization phenomena predominates. The max-plus algebra is one of the many algebraic structures which are called dioids, however it has several useful properties, which are not necessarily present in all dioids. We begin the definition of dioid by introducing monoid, the simple algebraic unit from which dioids are formed.

Definition

A monoid $(M, \oplus)$ is an algebraic structure consisting of a set of elements, M and an operation $\oplus$ on the elements of M such that:

(i) M is closed with respect to $\oplus$: $a, b \in M \wedge a \oplus b = c \Rightarrow c \in M$.

(ii) M is associative with respect to $\oplus$: $\forall a, b, c \in M : (a \oplus b) \oplus c = a \oplus (b \oplus c)$.

(iii) M has a zero or identity element, $\varepsilon \in M$ such that: $\forall a \in M : \varepsilon \oplus a = a \oplus \varepsilon = a$.

We note that a commutative monoid is one in which $\forall a, b \in M : a \oplus b = b \oplus a$.

Definition

A dioid $(D, \oplus, \otimes)$ is an algebraic structure consisting of a set D with a pair of associated operations $\otimes$ and $\oplus$ such that:

(i) $(D, \oplus)$ is a commutative monoid with identity element ε.

(ii) $(D, \oplus)$ is a monoid with identity element e.

(iii) ε is absorbing for $\otimes$, that is $\forall a \in D : a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon$

(iv) $\otimes$ is both left and right distributive over $\oplus$

$$\forall a, b, c \in D : c \otimes (a \oplus b) = (c \otimes a) \oplus (c \otimes b)$$

$$\forall a, b, c \in D : (a \oplus b) \otimes c = (a \otimes c) \oplus (b \otimes c)$$

(v) $\oplus$ is idempotent: $\forall a \in D : a \oplus a = a$.

We say that the dioid is commutative when $\otimes$ is commutative. When a dioid is closed for infinite series of $\oplus$ operations and the left and right distributive properties apply to these infinite series, then the dioid is said to be complete.

1.2 Max-plus Linear Algebra

In this section we define inverse and permutation matrices in terms of the max-plus algebra. The pair of operations $(\otimes, \oplus)$ is extended to matrices and vectors formally in the same way as in the conventional linear algebra. We consider $A = (a_{ij})$, $B = (b_{ij})$ and $C = (c_{ij})$ as matrices with elements from R_{max} of compatible sizes, then $C = A \oplus B$ if $c_{ij} = a_{ij} \oplus b_{ij}$ for all i, j; $C = A \otimes B$ if $c_{ij} = \Sigma_k^{\oplus} a_{ik} \otimes b_{kj} = \max_k(a_{ik} + b_{kj})$ for all i, j and $\alpha \otimes A = A \otimes \alpha = (\alpha \otimes a_{ij})$ for $\alpha \in R_{max}$. The transpose of a matrix A^T is as defined in conventional algebra that is: $(a)_{ij}^T = (a)_{ji}$. We denote by I the square matrix called the unit matrix, whose diagonal entries are 0 and the off-diagonals are ε. Then we have

$$I = \begin{pmatrix} 0 & \varepsilon & \cdots\cdots\cdots & \varepsilon \\ \varepsilon & 0 & \varepsilon\cdots\cdots\cdots & \varepsilon \\ \vdots & \vdots & \cdots\cdots\cdots & \vdots \\ \varepsilon & \varepsilon & \varepsilon & 0 \end{pmatrix}$$

and obviously $A \otimes I = I \otimes A = A$ if I is of a compatible dimension. This shows that in max-plus algebra, the identity matrix is an identity with respect to $\otimes$. For a square matrix in max-plus algebra and positive integer k, the

k^{th} power of A will be denoted by $A^{\otimes k}$which is the iterated product $A^{\otimes k} = \overbrace{A \otimes A \otimes ... \otimes A}^{k\ times}$ and by definition, $A^0 = I$ for any square matrix A. We note that $\otimes$ distributes over $\oplus$ for matrices and as usual $\oplus$ is commutative but $\otimes$ is not. Also $\oplus$ is idempotent in $\mathbb{R}_{max}^{n\times n}$ since we have $A \oplus A = A$. This means that $\mathbb{R}_{max}^{n\times n}$ is another idempotent semi-ring in which $\otimes$ is non-commutative.

A square matrix is called diagonal denoted by $diag(\lambda_1, \lambda_2, ..., \lambda_n)$, or $D(\lambda_i)$ if its diagonal entries are $\lambda_1, \lambda_2, ..., \lambda_n \in \mathbb{R}_{max}$ and off-diagonal entries are ε that is:

$$D(\lambda_i) = \begin{pmatrix} \lambda_1 & \varepsilon & \cdots\cdots\cdots & \varepsilon \\ \varepsilon & \lambda_2 & \varepsilon\cdots\cdots\cdots & \varepsilon \\ \vdots & \vdots & \cdots\cdots\cdots & \vdots \\ \varepsilon & \varepsilon & \varepsilon & \lambda_n \end{pmatrix}$$

A permutation matrix is a matrix in which each row and each column contains exactly one entry equal to e and all other entries are equal to ϵ. If $\sigma : \{1, 2, ..., n\} \rightarrow \{1, 2, ..., n\}$ is a permutation we define the max-plus permutation matrix as $P_\sigma = [p_{ij}]$ where

$$p_{ij} = \begin{cases} e & \text{if } i = \sigma(j) \\ \varepsilon & \text{if } i \neq \sigma(j) \end{cases}$$

So the j_{th} column of P_σ has e in the $\sigma(j)^{th}$ row.

Left multiplication by P_σ permutes the rows of a matrix, so that the i^{th} row of A appears as the $\sigma(i)^{th}$ row of $P_\sigma \otimes A$. We state and prove the following theorem to show that the matrix $A \in \mathbb{R}_{max}^{n \times n}$ has a right inverse if there is a permutation.

Theorem 1.1. *$A \in \mathbb{R}_{max}^{n \times n}$ has a right inverse if and only if there is a permutation σ and values $\lambda_i > \epsilon, i \in \{1, 2, ..., n\}$ and such that $A = P_\sigma \otimes D(\lambda_i)$.*

Proof. Let the matrix $B \in \mathbb{R}_{max}^{n \times n}$ such that $A \otimes B = I$. Then we have

$$\max_k(a_{ik} + b_{ki}) = e \;=\; 0 \text{ for each } i \tag{1.1}$$

$$\max_k(a_{ik} + b_{kj}) = \varepsilon \;=\; -\infty \text{ for all } i \neq j \tag{1.2}$$

We note from equation (1.1) that $\exists\; k$ for each i such that $(a_{ik} + b_{ki}) = e$ and so we have a function $k = \theta(i)$ with $a_{i\theta(i)} > \varepsilon$ and $b_{i\theta(i)} > \varepsilon$. We see form equation (1.2) that

$$a_{i\theta(j)} = \varepsilon \text{ for all } i \neq j. \tag{1.3}$$

Since $a_{i\theta(i)} > \varepsilon = a_{i\theta(j)}$ for $i \neq j$, it follows that θ is injective and therefore a permutation. From equation (1.3), $a_{i\theta(i)}$ is the only entry of the $\theta(i)^{th}$ column of A that is not ε. Now let's denote $P_\theta \otimes A$ by $\tilde{A}$, then $\theta(i)^{th}$ row of Ã is the i^{th} row of A whose entry is greater than ε in the $\theta(i)^{th}$ column. All the diagonal elements of Ã are greater than ε and A has only one non-ε element in each column. This is also true for Ã that is $P_\theta \otimes A = \tilde{A} = D(\lambda_i)$ with $\lambda_i = a_{\theta^{-1}(i)i} > \varepsilon$. Let $\sigma = \theta^{-1}$ and since $P_\sigma \otimes P_\theta = I$, it follows that $A = P_\sigma \otimes D(\lambda_i)$. Conversely we assume that $A = P_\sigma \otimes D(\lambda_i)$ and $\lambda_i > \varepsilon$. If this is true then we let let $B = D(-\lambda_i) \otimes P_{\sigma^{-1}}$ which implies that $A \otimes B = P_\sigma \otimes D(\lambda_i) \otimes D(-\lambda_i) \otimes P_{\sigma^{-1}} = P_\sigma \otimes P_{\sigma^{-1}} = I$ Therefore $A \otimes B = I$ and B is the right inverse of A. □

Any matrix which can be obtained from the unit or diagonal matrix by permuting the rows and/or columns will be called a generalized permutation matrix. We note also that the generalized permutation matrices are special in max-plus algebra than in the conventional linear algebra since they are the only matrices having inverses. Clearly, if an inverse to the matrix A exists then it is unique. This is shown by the following theorems.

Theorem 1.2. *For $A, B \in \mathbb{R}^{n \times n}_{max}$ if $A \otimes B = E$ then $B \otimes A = E$, and B is uniquely determined by A.*

Proof. From theorem (1.1) we have $A = P_\sigma \otimes D(\lambda_i)$. We see that $\tilde{B} = D(-\lambda_i) \otimes P_{\sigma^{-1}}$ is a left inverse of A. If $A \otimes B = I$, then $\tilde{B} = \tilde{B} \otimes (A \otimes B) = (\tilde{B} \otimes A) \otimes$

$\tilde{B} = I \otimes B = B$, implying that B is uniquely determined by A. $\square$

Theorem 1.3. *If $A \in \mathbb{R}_{max}^{n \times n}$ and $B \in \mathbb{R}_{max}^{n \times n}$ are invertible then $A \otimes B$ is invertible.*

Proof. From theorems (1.1) and (1.2) we have $A = P_{\sigma_a} \otimes D(\lambda_i^a)$ and $D(\lambda_i^b) \otimes P_{\sigma_b}$ Then $A \otimes B = P_{\sigma_a} \otimes D(\lambda_i^a) \otimes D(\lambda_i^b) \otimes P_{\sigma_b}$. Since the product of two diagonal matrices is a diagonal matrix we have: $A \otimes B = P_{\sigma_a} \otimes D(\lambda_i^a \otimes \lambda_i^b) \otimes P_{\sigma_b}$ Therefore $A \otimes B$ is a permuted diagonal matrix and hence $A \otimes B$ is invertible. $\square$

Theorem 1.4. *Suppose $A \in \mathbb{R}_{max}^{n \times n}$ and let $L_A : \mathbb{R}_{max}^n \to \mathbb{R}_{max}^n$ be a linear map $L_A(x) \to A \otimes x$. The the following are equivalent:*

1. *$A = P_\sigma \otimes D(\lambda_i)$ for some permutation $\lambda_i > \epsilon$*
2. *L_A is surjective*
3. *A has a right inverse:$A \otimes B = E$*
4. *A has a right inverse:$B \otimes A = E$*
5. *L_A is injective*

Proof. We have already proved that $1 \Leftrightarrow 3$. The proof that $1 \Leftrightarrow 3$, $2 \Leftrightarrow 3$, $1 \Leftrightarrow 2$, $1 \Leftrightarrow 4 \Leftrightarrow 5$ are all elementary. We are left to prove that $5 \Leftrightarrow 1$. To see this, we let L_A be injective and define the sets $F_i = j : a_{ji} > \varepsilon$ and

$G_i = j : a_{jk} > \varepsilon$ for some $k \neq i$ and claim that $F_i \not\subseteq G_i$. We suppose by contradiction that $F_i \subset G_i$. We show that this is a contradiction to the fact that L_A is injective. Let $x = [x_k]$ where

$$x_k = \begin{cases} e & \text{if } k \neq i \\ \varepsilon & \text{if } k = i \end{cases}$$

Define $b = A \otimes x = \bigoplus_{k \neq i} a_{*k}$, where a_{*k} denotes the k_{th} column of A and suppose that $j \in F_i$, then $j \in G_i$. Therefore there exists $k \neq i$ for which $a_{jk} > \varepsilon$. This means that we have $b_j \geq a_{jk} > \varepsilon$. Since $a_{ji} > \varepsilon$, then we can find $\alpha_j > \varepsilon$ such that $\alpha_j \otimes a_{ji} \leq b_j$. If $j \notin F_i$, then $a_{ji} = \varepsilon$. Therefore $\alpha \otimes a_{ji} \leq b_j$ for all j. Thus we have $\alpha \otimes a_{*i} \leq b$ and $A \otimes [x \oplus \alpha \otimes e_i] = [A \otimes x] \oplus [\alpha \otimes A \otimes e_i] = b \oplus \alpha \otimes a_{*j} = b$.

So for $\tilde{x} = x \oplus \alpha \otimes e_i$, $L_A\tilde{x} = L_A(x)$. But $x_i = \varepsilon < \tilde{x} = \alpha$, contradicting the fact that L_A is injective which proves our claim. We note that for each i there exists $j = \sigma(i)$ such that $a_{ji} > \varepsilon$ but $a_{jk} = \varepsilon$ for all $k \neq i$. In other words a_{ji} is the only entry not equal to ε in the $j = \sigma(i)$ row. But if $j = \sigma\acute{i}$ then it follows that $i = \acute{i}$ and σ is injective. This means that σ is a permutation. Therefore for each row j there is a unique column $i(j = \sigma(i))$ such that a_{ji} is the only entry not equal to ε. For each column i and any row k with $k \neq \sigma(i)$ we know that $k = \sigma(\acute{i})$ for some $\acute{i} \neq i$. This means a_{ki}

is not the unique non$-\varepsilon$ entry in the k^{th} row, so $a_{ji} = \varepsilon$. Therefore $a_{\sigma(i)i}$ is the only non$-\varepsilon$ entry in column i. Hence A is a permuted diagonal matrix and $A = P_\sigma \otimes D(\lambda_i)$ for some permutation $\lambda_i > \epsilon$. $\square$

1.2.1 Max-plus exponential functions

In this section we look at the operations $\oplus$ and $\otimes$ on exponential functions as induced by the conventional algebra. This concept is the basis of many proofs in the max-plus algebra. Infact it has been used to generalize the Cramer's rule and the Cayley-Hamilton's theorem as we shall see in the later sections of this chapter. If $\psi : (0, \infty) \to (0, \infty)$ and $\lambda \in (-\infty, \infty)$ then we define $\psi \asymp e^{s\lambda}$ to mean $\lim_{s\to\infty} s^{-1} ln(\psi) = \lambda$. We note in the conventional algebra that $ln(0) = -\infty$ and $e^{-\infty} = 0$. We state the following theorem which will be applied to the discussion of dominance and permanent of matrices in the next section.

Theorem 1.5. *If $f \asymp e^{sa}$ and $g \asymp e^{sb}$, then $f + g \asymp e^{s(a\oplus b)}$ and $fg \asymp e^{s(a\otimes b)}$*

Proof. We see that $\lim_{s\to\infty} s^{-1} ln(fg) = \lim_{s\to\infty} s^{-1} ln(f) + \lim_{s\to\infty} s^{-1} ln(g) = a + b = a \otimes b$ that is, $fg \asymp e^{s(a\oplus b)}$. Now we have $max(f, g) \leq f + g \leq 2max(f, g)$. This means that $\lim_{s\to\infty} s^{-1} ln(max(e^{sa}, e^{sb})) \leq \lim_{s\to\infty} s^{-1} ln(e^{sa} + e^{sb}) \leq \lim_{s\to} s^{-1}[ln(max(e^{sa}, e^{sb})) + ln(2)]$.

By applying the squeeze theorem we see that $\lim_{s\to\infty} s^{-1}ln(f+g) = max(a,b) = a \oplus b$ since $\lim_{s\to\infty} s^{-1}ln(max(e^{sa}, e^{sb})) = max(a,b)$. $\square$

1.2.2 Determinants

In this section we discuss the max-plus algebra determinant with reference to two related quantities, the permanent of A and the dominance of A and indicate the difference between it and the conventional algebra determinant. The max-plus determinant has no direct analogue with the conventional due to the absence of additive inverses. In conventional algebra $A \in \mathbb{R}^{nxn}$ has $detA = \Sigma_{\sigma\in P_n} sgn(\sigma)\Pi_{i=1}^{n} a_{i\sigma(i)}$ where P_n is the set of all permutations of $\{1,2,...,n\}$ and $sgn(\sigma)$ is the sign of the permutations σ. In the max-plus algebra the permanent of A, called $perm(A)$ and the dominant of A, $dorm(A)$ are two related quantities, which take over the role of the determinant. $Perm(A)$ is defined similarly to the determinant but with $sgn(\sigma)$ omitted , see (Olsder et al, 1998). For $A \in \mathbb{R}_{max}^{nxn}$ the permanent of A is defined to be $perm(A) = \bigoplus_{\sigma\in P_n} \bigotimes_{i=1}^{n}(a_{i\sigma(i)})$, with $\sigma \in P_n$ as the set of all permutation of $\{1,2,...,n\}$. The dominant of A is given by

$$dom(A) = \begin{cases} \lim_{s\to\infty} \frac{1}{s} ln \mid \det\,(e^{sA}) \mid & \text{if } \det\,(e^{sA}) \neq 0 \\ \epsilon & \text{if } \det\,(e^{sA}) = 0 \end{cases}$$

We state the following theorems which characterize permanents and dominants without proofs.

Theorem 1.6. *1.* $dom(A) \leq perm(A)$

2. If $A \in \mathbb{R}^{n \times n}_{max}$ *is invertible, then* $dom(A) \neq \epsilon$

3. If $A \in \mathbb{R}^{n \times n}_{max}$ *is invertible, then* $dom(A) = perm(A)$

4. If $A \in \mathbb{R}^{n \times n}_{max}$ *is invertible, then* $dom(A \otimes B) = dom(A) \otimes dom(B)$ *and* $perm(A \otimes B) = perm(A) \otimes perm(B)$.

1.2.3 Cramer's Rule

In this section, we present the max-plus formulation for the Cramer's rule, the method in linear algebra used to solve systems of linear equations. We note that the dominant is a refined version of permanent and leads to the result such as the max-plus analogue of the Cramer's rule. When A is a non-singular matrix, then in conventional algebra, Cramer's rule gives a solution to the linear matrix equation $Ax = b$ as follows:

$$x_i = \frac{\det(a_{*1}, \ldots a_{*i-1}, b, a_{*i+1} \ldots a_{*n})}{\det A}, \quad i = 1, 2, \ldots, n$$

where a_{*j} denotes the j^{th} column of A and $1 \leq j \leq n$, (Farlow,2009). The max-plus analogue of the Cramer's rule for $Ax = b$ is $A \otimes x = b$

and is given by $x_i \otimes dom(A) = dom(a_{\bullet 1}, ... a_{\bullet i-1}, b, a_{\bullet i+1} ... a_{\bullet n})$. Note that $(a_{\bullet 1}, ... a_{\bullet i-1}, b, a_{\bullet i+1} ... a_{\bullet n})$ is matrix A with its i^{th} column replaced by the vector b. Unlike the conventional algebra, $dom(A)$ is not sufficient to produce a solution. An additional condition is needed that is, $sign(a_{\bullet 1}, ... a_{\bullet i-1}, b, a_{\bullet i+1} ... a_{\bullet n}) = sign(A)$ for all $1 \leq i \leq n$. To define $sign(A)$, let P_n be the set of permutation $\sigma : \{1, 2, ..., n\} \rightarrow \{1, 2, ..., n\}$ and let $t_1, t_2, ..., t_L$ be all possible values such that: $t_i = \bigoplus_1^n (a_{i,\sigma(i)})$ for some $\sigma \in P_n$. Let $S_i = \{\sigma \in P_n : t_i = \bigoplus_{i=1}^n (a_{i,\sigma(i)})$ for some $\sigma \in P_n\}$, $S_{ie} = \{\sigma \in S_i : \sigma \in P_n^e\}$, $S_{ie} = \{\sigma \in S_i : \sigma \in P_n^o\}$ $k_{ie} = | S_{ie} |$ and $k_{io} = | S_{io} |$.
Then $sign(A) = 1$ if $k_{ie} - k_{io} \geq 0$ and $sign(A) = -1$ if $k_{ie} - k_{io} \leq 0$. If $dom(A) = \epsilon$ then $sign(A) = \epsilon$. From the above definition we can write $det(e^{sA}) = \Sigma_i^n (k_{ie} - k_{io}) e(st_i)$. We note that if $sign(A) \neq \epsilon$, then $sign(A) det(e^{sA}) > 0$ for all sufficiently large s. If $sign(a_{\bullet 1}, ... a_{\bullet i-1}, b, a_{\bullet i+1} ... a_{\bullet n}) = sign(A)$ for all i and $dom(A) > \epsilon$ then $x_i \otimes dom(A) = dom(a_{\bullet 1}, ... a_{\bullet i-1}, b, a_{\bullet i+1} ... a_{\bullet n})$ yields a solution to $A \otimes x = b$.

1.2.4 Linear Dependence and Independence

The concept of linear independence and dependence has its analogue in the max-plus sense (Gaubert, 1997). Since max-plus algebra is an idempotent semi-ring we need the definition of semi-module to explain linear dependence and independence. A semi-module is essentially a linear space over a semi-ring. Semi-modules and sub-semimodules are analogous to modules and submodules over rings. A set $V \subset \mathbb{R}^n_{max}$ is a commutative idempotent semi-ring over $\mathbb{R}_{max}$ if it is closed under $\oplus$ and scaler multiplication, that is, $\alpha \otimes v \in V$ and $u \oplus v \in V$ for all $\alpha \in \mathbb{R}_{max}$ and $u, v \in \mathbb{R}^n_{max}$ (Hogben et al, 2007). A finitely generated semi-module $V \subset \mathbb{R}^n_{max}$ is the set of all linear combinations of a finite set $\{u_1, u_2, ..., u_r\}$ of vectors in $\mathbb{R}^n_{max}$:

$$V = \{\bigoplus_{i=1}^{r} \alpha_i \otimes u_i \mid \alpha_1, \alpha_2, ..., \alpha_r \in \mathbb{R}_{max}\}$$

An element x can be written as a finite linear combination of elements of $F \subseteq V$ if $x = \bigoplus_{f \in F} \lambda_f \otimes f$ for some $\lambda_f \in \mathbb{R}_{max}$ such that $\lambda_f = \varepsilon$ for all but finitely many $f \in F$. There are different interpretations of linear dependence and independence but we consider their definitions due to (Akian et al, 2007).

Definition

1. A set p of vectors $\{v_1, v_2, ..., v_p\} \in \mathbb{R}^n_{max}$ is linearly dependent if the set $\{1, 2, ..., p\}$ can be partitioned into disjoint subsets I and K such that for $j \in I \cup K$ there exist $\alpha_j \in \mathbb{R}_{max}$, not all equal to ϵ and

$$\bigoplus_{i \in I} \alpha_i v_i = \bigoplus_{k \in K} \alpha_k v_k$$

2. A set p of vectors $\{v_1, v_2, ..., v_p\} \in \mathbb{R}^n_{max}$ is linearly independent if for all disjoint subsets I and K of $\{1, 2, ..., p\} j \in I \cup K$ and all $\alpha_j \in \mathbb{R}_{max}$ we have

$$\bigoplus_{i \in I} \alpha_i v_i \neq \bigoplus_{k \in K} \alpha_k v_k$$

unless $\alpha = \varepsilon$ for all $j \in I \cup K$.

1.2.5 Graph Theory in Max-plus

Many results in max-plus can be attributed to the theory of graphs in max-plus. Graph theory plays an important role in obtaining the maximum cycle mean, which is used to solve eigenvalue and eigenvector problems. A directed graph is an ordered pair $D = (V, E)$ where V is a non-empty set (of nodes) and $E \subseteq V \times V$ (the set of arcs). A sub-digraph of D is any digraph $D' = (V', E')$ such that $V' \subseteq V$ and $E' \subseteq E$. If $e = (u, v) \in E$ for some

$u, v \in V$ then we say that e is leaving u and entering v. Any arc of the form (u, u) is called a loop (Butkovič, 2008).

Let $D = (V, E)$ be a given digraph. A sequence $\pi = (v_1, ..., v_p)$ of nodes in D is called a path (in D) if $p = 1$ or $p > 1$ and $(v_i, v_{i+1}) \in E$ for all i=1,...,p-1. The node v_1 is called the starting node and v_p the end node of π. The number $p - 1$ is called the length of π that is $l(\pi)$ and thus Π is a $u - v$ path. If there is a $u - v$ path in D then v is said to be reachable from u that is $u \to v$. A path $(v_1, ..., v_p)$ is called a cycle if $(v_1 = v_p)$ and $p > 1$. If there is no cycle in D then D is called acyclic.

A digraph is strongly connected if $u \to v$ for all nodes u, v in D. A sub-digraph D' of D is called a strongly connected component of D if it is a maximal strongly connected sub-digraph of D.

If $A = (a_{ij}) \in \mathbb{R}^{n \times n}$ then the symbol $F_A(Z_A)$ will denote the digraphs with the node set N and arc sets $E = \{(i, j); a_{ij} > \epsilon\}$ $(E = (i, j); a_{ij} = 0)$. $F_A(Z_A)$ will be called the finiteness (zero) digraph of A. If F_A is strongly connected then A is called irreducible and reducible otherwise. If $A = (a_{ij}) \in \mathbb{R}^{n \times n}_{max}$ is irreducible and $n > 1$ then A is called doubly $\mathbb{R}$-astic. and if $A = (a_{ij}) = \mathbb{R}^{n \times n}$ is row or column $\mathbb{R}$-astic then F_A contains a cycle.

A weighted digraph $D = (V, E, w)$ where (V, E) is a digraph and $w : E \to \mathbb{R}$.

If $\pi = (v_1, ..., v_p)$ is a path in (V, E, w) then the weight of π is $w(\pi) = w(v_1, v_2) + w(v_2, v_3) + ... + w(v_{p-1}, v_p)$ if $p > 1$ and ϵ if $p = 1$. A cycle σ is called positive if $w(\sigma) > 0$. In contrast, σ is called a zero cycle if $w(v_k, v_{k+1}) = 0$ for all $k = 1, ..., p-1$. Given $A = (a_{ij}) \in \mathbb{R}^{n \times n}_{max}$ let D_A denote the weighted digraph (N, E, w) where $F_A = (N, E)$ and $w(i, j) = a_{ij}$ for all $(i, j) \in E$. If $\pi = (i_1, ..., i_p)$ is a path in D_A then we denote $w(\pi, A) = w(\pi)$ and it follows that $w(\pi, A) = a_{i_1 i_2} + a_{i_2 i_3} + ... + a_{i_{p-1} i_p}$ if $p > 1$ and ϵ if $p = 1$ (Carré , 1971).

1.2.6 Max-plus polynomials

In this section we introduce max-plus polynomials and its application in modeling Discrete Event Systems like the max-plus-linear input-output systems, (Baccelli et al., 1992). We let the operator γ be such that $\gamma z(k) = z(k-1)$, then the max-plus polynomial is $P(\gamma) = p_0 \otimes \gamma^0 \oplus p_1\gamma_1 \oplus ... \oplus p_n \otimes \gamma^n$ and we have

$$\begin{aligned} P(\gamma)z(k) &= (p_0 \otimes \gamma^0 \oplus p_1\gamma_1 \oplus ... \oplus p_n \otimes \gamma^n)z(k) \\ &= p_0 \otimes \gamma^0 z(k) \oplus p_1\gamma_1 z(k) \oplus ... \oplus p_n \otimes \gamma^n z(k) \\ &= p_0 \otimes z(k) \oplus p_1 \otimes z(k-1) \oplus ... \oplus p_n \otimes z(k-n) \end{aligned}$$

Let P, Q and R be three max-plus polynomials such that

$$\begin{aligned} P(\gamma) &= p_0 \otimes \gamma^0 \oplus p_1\gamma_1 \oplus ... \oplus p_n \otimes \gamma^n \\ Q(\gamma) &= q_0 \otimes \gamma^0 \oplus q_1\gamma_1 \oplus ... \oplus q_n \otimes \gamma^n \\ R(\gamma) &= r_0 \otimes \gamma^0 \oplus r_1\gamma_1 \oplus ... \oplus r_m \otimes \gamma^m \end{aligned}$$

Then the max-plus product and sum for polynomials are defined as follows:

$$\begin{aligned} P(\gamma) \oplus Q(\gamma) &= p_0 \otimes \gamma^0 \oplus p_1\gamma^1 \oplus ... \oplus p_n \otimes \gamma^n \oplus q_0 \otimes \gamma^0 \oplus q_1\gamma_1 \oplus ... \oplus q_n \otimes \gamma^n \\ &= \bigoplus_{i=0}^{n} (p_i \oplus q_i) \otimes \gamma_i \\ P(\gamma) \otimes R(\gamma) &= (p_0 \otimes \gamma^0 \oplus p_1\gamma^1 \oplus ... \oplus p_n \otimes \gamma^n) \otimes (r_0 \otimes \gamma^0 \oplus r_1\gamma_1 \oplus ... \oplus r_m \otimes \gamma^m) \\ &= \bigoplus_{i=0}^{n} \bigoplus_{j=0}^{m} (p_i \otimes r_j) \otimes \gamma^{i+j} \end{aligned}$$

Let P, Q, R be three max-plus polynomials and z and w be two signals, then we can observe the following properties of the max-plus polynomial expressions:

$$\begin{aligned} P(\gamma)z(k) \oplus Q(\gamma)z(k) &= (P(\gamma) \oplus Q(\gamma))z(k) \\ P(\gamma)z(k) \oplus P(\gamma)w(k) &= P(\gamma)(z(k) \oplus w(k)) \\ P(\gamma)(R(\gamma)z(k)) &= (P(\gamma) \otimes R(\gamma))z(k) \end{aligned}$$

We consider systems that can be described by the input-output relation:

$$y(k) = a_1 \otimes y(k-1) \oplus a_2 \otimes y(k-2) \oplus ... \oplus a_n \otimes y(k-n)$$

$$\oplus \; b_0 \otimes uk \oplus b_1 \otimes u(k-1) \oplus ... \oplus b_m \otimes u(k-m)$$

This can be written in polynomial form as $y(k) = A(\gamma)y(k) \oplus B(\gamma)u(k)$ where $A(\gamma)$ and $B(\gamma)$ are polynomial operators

$$\begin{aligned} A(\gamma) &= a_1 \otimes \gamma^1 \oplus a_2\gamma^2 \oplus ... \oplus a_n \otimes \gamma^n \\ B(\gamma) &= b_0 \otimes \gamma^0 \oplus b_1\gamma^1 \oplus ... \oplus b_m \otimes \gamma^m \end{aligned}$$

A system of multivariate polynomial equalities and inequalities in the max-plus algebra is defined as a set of integers $\{m_k\}$ and sets of coefficients $\{a_{ki}\}, \{b_k\}$ and $\{c_{kij}\}$ with $i \in \{1, ..., m_k\}, j \in \{1, ..., n\}$ and $k \in \{1, ..., p_1, p_1 + 1, ..., p_1 + p_2\}$ such that

$$\begin{aligned} \bigoplus_{i=1}^{m_k} a_{ki} \otimes \bigoplus_{j=1}^{n} x_j^{\otimes c_{kij}} &= b_k \quad \text{for } k = 1, 2, ..., p_1, \\ \bigoplus_{i=1}^{m_k} a_{ki} \otimes \bigoplus_{j=1}^{n} x_j^{\otimes c_{kij}} &\leq b_k \quad \text{for } k = p_1 + 1, ..., p_1 + p_2. \end{aligned}$$

In conventional algebra the equations can be written as:

$$\begin{aligned} \max_{i=1,...,m_k} (a_{ki} + \Sigma_{j=1}^{n} c_{kij} x_j) &= b_k \quad \text{for } k = 1, 2, ..., p_1, \\ \max_{i=1,...,m_k} (a_{ki} + \Sigma_{j=1}^{n} c_{kij} x_j) &\leq b_k \quad \text{for } k = p_1 + 1, ..., p_1 + p_2. \end{aligned}$$

1.2.7 Solution of Systems of Equations in Max-plus

The theory of linear systems of equations for the max-plus method is developed in this section after Gordran and Minoux, 1984. Generally, we would like to be able to solve the matrix equation $A \otimes \mathbf{x}=\mathbf{b}$. Although there are some similarities in solving systems of equations in max-plus and the conventional algebra method, the operation $\oplus$ creates some interesting differences. We consider the equivalent system of equations in the conventional algebra to first get an idea of how to solve the system. We can write $A \otimes \mathbf{x}=\mathbf{b}$ as the following detailed matrix equation and the equivalent system of max-plus equations as follows:

$$A \otimes \mathbf{x}=\mathbf{b} \equiv \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \otimes \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

That is:

$$\begin{aligned} (a_{11} \otimes x_1) \oplus (a_{12} \otimes x_2) \oplus \cdots \oplus (a_{1n} \otimes x_n) &= b_1 \\ (a_{21} \otimes x_1) \oplus (a_{22} \otimes x_2) \oplus \cdots \oplus (a_{2n} \otimes x_n) &= b_2 \end{aligned}$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots \qquad \vdots$$

$$(a_{m1} \otimes x_1) \oplus (a_{m2} \otimes x_2) \oplus \cdots \oplus (a_{mn} \otimes x_n) \; = \; b_m$$

The following system must be solved simultaneously as follows:

$$max\{(a_{11} + x_1), (a_{12} + x_2), \cdots, (a_{1n} + x_n)\} = b_1$$

$$max\{(a_{21} + x_1), (a_{22} + x_2), \cdots, (a_{2n} + x_n)\} = b_2$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots \qquad \vdots$$

$$max\{(a_{m1} + x_1), (a_{m2} + x_2), \cdots, (a_{mn} + x_n)\} = b_m$$

We first consider the case that a solution exists and some of the entries of $\mathbf{b}$ are $-\infty$. Without loss of generality, we can reorder the equations so that the finite entries of $\mathbf{b}$ occur first:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \otimes \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ b_k \\ -\infty \\ \vdots \\ -\infty \end{pmatrix}$$

This gives the following system of equations:

$$max\{(a_{11}+x_1),(a_{12}+x_2),\cdots,(a_{1n}+x_n)\}=b_1$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots$$

$$max\{(a_{k1}+x_1),(a_{k2}+x_2),\cdots,(a_{kn}+x_n)\}=b_k$$

$$max\{(a_{k+1,1}+x_1),(a_{k+1,2}+x_2),\cdots,(a_{k+1,n}+x_n)\}=-\infty$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots$$

$$max\{(a_{n1}+x_1),(a_{n2}+x_2),\cdots,(a_{nn}+x_n)\}=-\infty$$

Rearranging the variables so that those $j's$ such that $a_{k+1,j},\ldots,a_{m,j}=-\infty$ occur first we have:

$$\begin{pmatrix} A_1 & | A_2 \\ A_3 & | A_4 \end{pmatrix}\begin{pmatrix} x_1 \\ \vdots \\ x_l \\ x_{l+1} \\ x_n \end{pmatrix}=\begin{pmatrix} b_1 \\ \vdots \\ b_k \\ -\infty \\ \vdots \\ -\infty \end{pmatrix}$$

Let the dimensions of A_1 be $k \times l$. Let $\mathbf{b}' = \begin{pmatrix} b_1 \\ \vdots \\ b_k \end{pmatrix}$ and $\mathbf{x}' = \begin{pmatrix} x_1 \\ \vdots \\ x_l \end{pmatrix}$.

We note that if $A \otimes \mathbf{x} = \mathbf{b}$ has a solution, then $x_{n+1} = x_n = -\infty$, and $A \otimes \mathbf{x}' = \mathbf{b}'$. Thus, $A \otimes \mathbf{x} = \mathbf{b}$ has a solution if and only if $\mathbf{x}'$ is a solution to $A_1 \otimes \mathbf{x}' = \mathbf{b}'$ and solutions to $A \otimes \mathbf{x} = \mathbf{b}$ are:

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}' \\ -\infty \\ \vdots \\ -\infty \end{pmatrix}.$$

Therefore, the solution of a system with infinite entries in $\mathbf{b}$ can be reduced to that of a system where all the entries in $\mathbf{b}'$ are finite. Hence we restrict our attention to the system $A \otimes \mathbf{x} = \mathbf{b}$ where all the entries of $\mathbf{b}$ are finite. If there has to be a solution to the system of max-plus equations, then $a_{ij} + x_j \leq b_i$ for all $i \in \{1, ..., m\}$ and $j \in \{1, ..., n\}$. To search for a solution to the system, we first consider each component of $\mathbf{x}$ separately. For example, if there is a solution x_1 to the system, then $a_{i1} + x_1 \leq b_i$ for $i = 1, ..., m$. Thus $x_1 \leq b_i - a_{i1}$ for each i, leading to the following system of upper bounds on x_1:

$$x_1 \leq b_1 - a_{11}$$

$$x_1 \leq b_2 - a_{21}$$

$$\vdots \qquad \vdots$$

$$x_1 \leq b_m - a_{m1}$$

The solution of this system of inequalities must satisfy

$$x_1 \leq min\{(b_1 - a_{11}), (b_2 - a_{21}), \cdots, (b_m - a_{m1})\}$$

Similarly, $x_2, ..., x_n$, will give us the following system of inequalities on the entries of $\mathbf{x}$:

$$x_1 \leq min\{(b_1 - a_{11}), (b_2 - a_{21}), \cdots, (b_m - a_{m1})\}$$

$$x_2 \leq min\{(b_1 - a_{12}), (b_2 - a_{22}), \cdots, (b_m - a_{m2})\}$$

$$\vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots$$

$$x_n \leq min\{(b_1 - a_{1n}), (b_2 - a_{2n}), \cdots, (b_m - a_{mn})\}$$

This leads to a candidate for a solution to $A \otimes \mathbf{x}=\mathbf{b}$, That is:

$$\mathbf{x}' = \begin{pmatrix} x_1' \\ x_2' \\ \vdots \\ x_n' \end{pmatrix} \text{ where}$$

$$x_1' \leq min\{(b_1 - a_{11}), (b_2 - a_{21}), \cdots, (b_m - a_{m1})\}$$

$$x_2' \leq min\{(b_1 - a_{12}), (b_2 - a_{22}), \cdots, (b_m - a_{m2})\}$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots$$

$$x_n' \leq min\{(b_1 - a_{1n}), (b_2 - a_{2n}), \cdots, (b_m - a_{mn})\}$$

We define the discrepancy matrix $D_{A,\mathbf{b}}$ as follows:

$$D_{A,\mathbf{b}} = \begin{pmatrix} b_1 - a_{11} & b_1 - a_{12} & \cdots & b_1 - a_{1n} \\ b_2 - a_{21} & b_2 - a_{22} & \cdots & b_2 - a_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ b_m - a_{m1} & b_m - a_{m2} & \cdots & b_m - a_{mn} \end{pmatrix}$$

$D_{A,\mathbf{b}}$ is a matrix with all the upper bounds of the x_i's and each x_i' can be found by taking the minimum of the ith column of $D_{A,\mathbf{b}}$.

1.2.8 Solution of systems of linear equations and inequalities in max-plus

In this section we consider the problem consisting of a system of linear equations and inequalities in max-plus algebra and present necessary and sufficient conditions for its solvability.

Let $A = (a_{ij}) \in \mathbb{R}^{k \times n}, C = (c_{ij}) \in \mathbb{R}^{r \times n}, b = (b_1, ..., b_k)^T \in \mathbb{R}^k$ and $d = (d_1, ..., d_r)^T \in \mathbb{R}^r$. A one sided max-linear system with both equations and inequalities is of the form:

$$\begin{aligned} A \otimes x &= b \\ C \otimes x &\leq d \end{aligned} \tag{1.4}$$

The following notation will be used throughout this section:

$$\begin{aligned} M &= \{1, 2, ..., m\} \\ N &= \{1, 2, ..., n\} \\ R &= \{1, 2, ..., r\} \\ K &= \{1, 2, ..., k\} \\ J &= \{j \in N; \bar{\bar{x}}_j \geq \bar{x}_j\} \\ L &= N \setminus J \end{aligned}$$

$$
\begin{aligned}
a^{-1} &= -a, \forall a \in \mathbb{R} \\
S(A,B) &= \{x \in \mathbb{R}^n; A \otimes x = b\} \\
M_j &= \{k \in M; b_k \otimes a_{kj}^{-1} = \min_{i \in M}(b_i \otimes a_{ij}^{-1})\} \ \forall \ j \in N \\
\overline{x}_j &= \min_{i \in M}(b_i \otimes a_{ij}^{-1}) \ \forall \ j \in N \\
\overline{x}_j &= \min_{i \in K}(b_i \otimes a_{ij}^{-1}) \ \forall \ j \in N \\
\overline{\overline{x}}_j &= \min_{i \in R}(d_i \otimes c_{ij}^{-1}) \ \forall \ j \in N \\
K_j &= \{k \in K; b_k \otimes a_{kj}^{-1} = \min_{i \in K}(b_k \otimes a_{kj}^{-1})\} \ \forall \ j \in N \\
\overline{x} &= (\overline{x}_1, ..., \overline{x}_n)^T \\
\overline{\overline{x}} &= (\overline{\overline{x}}_1, ..., \overline{\overline{x}}_n)^T \\
S(A,C,b,d) &= \{x \in \mathbb{R}^n; A \otimes x = b \ \text{ and } \ C \otimes x \leq d\} \\
S(A,b,\leq) &= \{x \in \mathbb{R}^n; A \otimes x \leq b\} \\
S(C,d,\leq) &= \{x \in \mathbb{R}^n; C \otimes x \leq d\}
\end{aligned}
$$

We also define the vector $\hat{x} = (\hat{x}_1, \hat{x}_2, ..., \hat{x}_n)^T$, where

$$
\hat{x}_j = \begin{cases} \overline{x}_j & \text{if } j \in J \\ \overline{\overline{x}}_j & \text{if } j \in L \end{cases}
$$

and $N_{\hat{x}} = \{j \in N; \hat{x}_j = \overline{x}_j\}$. Since much attention has been given to one-sided systems of linear equations and systems of inequalities in max-plus algebra, we present the combination of the two system as one and the aim is to analyze the existence and uniqueness of solution to such systems . The following theorems show how the existence of a unique solution to max-linear system $A \otimes x = b$ is described and how the system containing both equations and inequalities can be solved. We state the following theorems and without proofs (Aminu, 2011, Cuninghame-Green, 1979).

Theorem 1.7. *Let $A = (a_{ij}) \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. Then $x \in S(A, b)$ if and only if*

$$(i)\ x \leq \overline{x} \text{ and}$$

$$(ii) \bigcup_{j \in N_x} M_j = M \text{ where } N_x = \{j \in N; x_j = \overline{x}_j\}$$

$$(iii)\ x \in S(A, b, \leq) \text{ if and only if } x \leq \overline{x}.$$

Theorem 1.8. *Let $A = (a_{ij}) \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}_m$. Then $S(A, b) = \{\overline{x}\}$ if and only if*

$$(i) \bigcup_{j \in N} M_j = M \text{ and}$$

$$(ii) \bigcup_{j \in N'} M_j \neq M \text{ for any } N' \subseteq N, N' \neq N$$

Theorem 1.9. *Let $A = (a_{ij}) \in \mathbb{R}^{k \times n}, C = (c_{ij}) \in \mathbb{R}^{r \times n}, b = (b_1, ..., b_k)^T \in \mathbb{R}^k$ and $d = (d_1, ..., d_r)^T \in \mathbb{R}^r$. Then the following three statements are equivalent:*

$$(i)\ S(A, C, b, d) \neq \emptyset$$

$$(ii)\;\; \hat{x} \in S(A, C, b, d)$$

$$(iii)\;\; \bigcup_{j\in J} K_j = K$$

Proof. $(i) \implies (ii)$. Let $x \in S(A, C, b, d)$, therefore $x \in S(A, b)$ and $x \in S(C, d, \leq)$. Since $x \in S(C, d, \leq)$, it follows from theorem 1.7(iii) that $x \leq \overline{\overline{x}}$. Now that $x \in S(A, b)$ and also $x \in S(C, d, \leq)$, we need to show that $\overline{\overline{x}}_j \geq \overline{x}_j \;\; \forall j \in N_x$ (that is $N_x \subseteq J$). Let $j \in N_x$ then $x_j = \overline{x}_j$. Since $x \in S(C, d, \leq)$ we have $x \leq \overline{\overline{x}}$ and therefore $\overline{x}_j \leq \overline{\overline{x}}_j$ thus $j \in J$. Hence $N_x \subseteq J$ and by theorem 1.7, $\bigcup_{j\in J} K_j = K$. This also proves $(i) \implies (iii)$.

$(iii) \implies (i)$. Suppose $\bigcup_{j\in J} K_j = K$. Since $\hat{x} \leq \overline{\overline{x}}$ we have $\hat{x} \in S(C, d, \leq)$. Also $\hat{x} \leq \overline{x}$ and $N_{\hat{x}} \supseteq J$ gives $\bigcup_{j\in N_{\hat{x}}} K_j \supseteq \bigcup_{j\in J} K_j = K$. Hence $\bigcup_{j\in N_{\hat{x}}} K_j = K$, therefore $\hat{x} \in S(A, b)$ and $\hat{x} \in S(C, d, \leq)$. Hence $\hat{x} \in S(A, C, b, d)$ (that is $S(A, C, b, d) \neq \emptyset$) and this proves $(iii) \implies (ii)$. $\square$

Theorem 1.10. *Let $A = (a_{ij}) \in \mathbb{R}^{k\times n}$, $C = (c_{ij}) \in \mathbb{R}^{r\times n}$, $b = (b_1, ..., b_k)^T \in \mathbb{R}^k$ and $d = (d_1, ..., d_r)^T \in \mathbb{R}^r$. Then $x \in S(A, C, b, d)$ if and only if*

$$(i)\;\; x \;\leq\; \hat{x} \;\; \textit{and}$$

$$(ii)\; \bigcup_{j\in N_x} K_j \;=\; K \;\; \textit{where} \;\; N_x = \{j \in N; x_j = \overline{x}_j\}$$

Proof. Let $x \in S(A, C, b, d)$, then $x \leq \overline{x}$ and $x \leq \overline{\overline{x}}$. Since $\hat{x} = \overline{x} \oplus \overline{\overline{x}}$ we have $x \leq \hat{x}$. Also $x \in S(A, C, b, d)$ implies that $x \in S(C, d, \leq)$. It follows from theorem (1.7) that $\bigcup_{j\in N_x} K_j = K$.

Suppose that $x \leq \hat{x} = \overline{x} \oplus \overline{\overline{x}}$ and $\bigcup_{j\in N_x} K_j = K$. It follows from theorem (1.7) that $x \in S(A, C, b, d)$, and also by theorem (1.8) that $x \in S(C, d, \leq)$. Thus $x \in S(A, b) \cap S(C, d, \leq) = x \in S(A, C, b, d)$. $\square$

Let $|X|$ be the number of elements of the set X.

Lemma 1: Let $A = (a_{ij}) \in \mathbb{R}^{k\times n}, C = (c_{ij}) \in \mathbb{R}^{r\times n}, b = (b_1, ..., b_k)^T \in \mathbb{R}^k$ and $d = (d_1, ..., d_r)^T \in \mathbb{R}^r$. If $|S(A, C, b, d)| = 1$ then $|S(A, b)| = 1$.

Proof. Suppose $|S(A, C, b, d)| = 1$, that is $S(A, C, b, d) = \{x\}$ for any $x \in \mathbb{R}_n$. Since $S(A, C, b, d) = \{x\}$ we have $x \in S(A, b)$ and thus $S(A, b) = \emptyset$. For contradiction, suppose $|S(A, b)| > 1$. We need to check the following two cases: (i) $L \neq \emptyset$ and (ii) $L = \emptyset$ where $L = N \setminus J$, and show in each case that $|S(A, C, b, d)| > 1$.

Proof of case (i): Suppose that L contains only one element say $n \in N$ that is $L = \{n\}$. Since $x \in S(A, C, b, d)$, it follows from theorem (1.9) that $\hat{x} \in S(A, C, b, d)$. That is $x = \hat{x} = (\overline{x}_1, ..., \overline{x}_n, \overline{\overline{x}}_n) \in S(A, C, b, d)$. It can be seen that $\overline{\overline{x}}_n \leq \overline{x}_n$ and any vector of the form $z = (\overline{x}_1, ..., \overline{x}_{n-1}, \alpha) \in S(A, C, b, d)$, where $\alpha \leq \overline{\overline{x}}_n$. Hence $|S(A, C, b, d)| > 1$. If L contains more than one element, then the proof is done in a similar way. Case (ii): that is $L = \emptyset (J = N)$: Suppose $J = N$. Then we have $\hat{x} = \overline{x} \leq \overline{\overline{x}}$. Suppose without loss of generality that $x, x\prime \in S(A,b)$ such that $x \neq x\prime$. Then $x \leq \overline{x} \leq \overline{\overline{x}}$ and also $x\prime \leq \overline{x} \leq \overline{\overline{x}}$. Thus, $x, x\prime \in S(C,d,\leq)$. Consequently, $x, x\prime \in S(A,C,b,d)$ and $x \neq x\prime$. Hence $|S(A,C,b,d)| > 1$. $\square$

Theorem 1.11. *Let $A = (a_{ij}) \in \mathbb{R}^{k\times n}, C = (c_{ij}) \in \mathbb{R}^{r\times n}, b = (b_1, ..., b_k)^T \in \mathbb{R}^k$ and $d = (d_1, ..., d_r)^T \in \Re^r$. If $|S(A,C,b,d)| = 1$ then $J = N$.*

Proof. Suppose $|S(A,C,b,d)| = 1$. It follows from theorem (1.7) that $\bigcup_{j\in J} K_j = K$. Also, $|S(A,C,b,d)| = 1$ implies that $|S(A,b)| = 1$ from Lemma 1. Moreover, $|S(A,b)| = 1$ implies that $\bigcup_{j\in J} K_j = K$ and $\bigcup_{j\in N'} K_j \neq K, N' \subseteq N, N' \neq N$. Since $J \subseteq N$ and $\bigcup_{j\in J} K_j = K$, we have $J = N$. $\square$

Corollary 1: Let $A = (a_{ij}) \in \mathbb{R}^{k\times n}, C = (c_{ij}) \in \mathbb{R}^{r\times n}, b = (b_1, ..., b_k)^T \in \mathbb{R}^k$ and $d = (d_1, ..., d_r)^T \in \mathbb{R}^r$. If $|S(A,C,b,d)| = 1$ then $S(A,C,b,d) = \{\overline{x}\}$.

Proof: The statement follows from Theorem 1.7 and Lemma 1.

Corollary 2: Let $A = (a_{ij}) \in \mathbb{R}^{k\times n}, C = (c_{ij}) \in \mathbb{R}^{r\times n}, b = (b_1, ..., b_k)^T \in \mathbb{R}^k$ and $d = (d_1, ..., d_r)^T \in \mathbb{R}^r$. Then the following three statements are equivalent:

$$(i)\ \ |S(A,C,b,d)| = 1$$

$$(ii)\ \ |S(A,b)| = 1 \ \text{ and } \ J = N$$

$$(iii)\ \ \bigcup_{j\in J} K_j = K \text{and } \ \bigcup_{j\in J'} K_j \neq K, \ \text{ for every } \ J' \subseteq J, J' \neq J, \ \text{ and } \ J = N.$$

Proof. $(i) \implies (ii)$ Follows from Lemma(1) and Theorem 1.11. $(ii) \implies (i)$ Let $J = N$, therefore $\overline{x} \leq \overline{\overline{x}}$ and thus $S(A,b) \cap S(C,d,\leq) = S(A,b)$. Hence $|S(A,C,b,d)| = 1$.

$(ii) \implies (iii)$ Suppose that $S(A,B) = \{x\}$ and $J = N$. It follows from Theorem 1.8 that $\bigcup_{j\in N} K_j = K$ and $\bigcup_{j\in N'} K_j \neq K, N' \subseteq N, N' \neq N$. Since $J = N$ the statement follows from Theorem 1.8.

$(iii) \implies (ii)$ It is immediate that $J = N$ and the statement now follows from Theorem 1.8. $\square$

Theorem 1.12. *Let $A = (a_{ij}) \in \mathbb{R}^{k\times n}, C = (c_{ij}) \in \mathbb{R}^{r\times n}, b = (b_1, ..., b_k)^T \in \mathbb{R}^k$ and $d = (d_1, ..., d_r)^T \in \mathbb{R}^r$. If $|S(A,C,b,d)| > 1$ then $|S(A,C,b,d)|$ is infinite.*

Proof. Suppose $|S(A,C,b,d)| > 1$. By Corollary 2 we have $\bigcup_{j\in J} K_j = K$, and for some $J \subseteq N, J \neq N$ (that is $\exists j \in N$ such that $\overline{x}_j > \overline{\overline{x}}_j$). Now $J \subseteq N$ and $\bigcup_{j\in J} K_j = K$, Theorem 1.8 implies that any vector $x = (x_1, x_2, ..., x_n)^T$ of the form:

$$x_j = \begin{cases} \overline{x}_j & \text{if } j \in J \\ y \leq \overline{\overline{x}}_j & \text{if } j \in L \end{cases}$$

is in $S(A,C,b,d)$, and the statement follows. From Theorem 1.10 we can say that the number of solutions to the one-sided system containing both equations and inequalities can only be $1, 0, \infty$. □

1.2.9 Max-Linear Program with equation and inequality constraints

In this section, we show how max-plus can be applied to optimization. In particular we show formulation of Max-linear program with equality and inequality constraints. We present a polynomial algorithm for solving max-linear programs whose constraints are max-linear equations and inequalities. This algorithm does not increase the number of constraints and thus decrease the computational complexity.

Suppose $f \in \mathbb{R}^n$ and let $f(x) = f^T \otimes x$ be defined on $\mathbb{R}^n$. Then,

(i) $f(x)$ is max-linear (Gondran and Minoux, 1984), i.e. $f(\lambda \otimes x \oplus \mu \otimes y) = \lambda \otimes f(x) \oplus \mu \otimes f(x)$ for every $x, y \in \mathbb{R}^n$.

(ii) $f(x)$ is isotone (Gondran and Minoux, 1984), i.e. $f(x) \leq f(y)$ for every $x, y \in \mathbb{R}^n, x \leq y$.

Suppose that the vector $f = (f_1, f_2, ..., f_n)^T \in \mathbb{R}^n$ is given. The task of minimizing(maximizing) the function:

$$f(x) = f^T \otimes x = \max(f_1 + x_1, f_1 + x_2, ..., f_n + x_n) \text{ subject to}$$

$$A \otimes x = b$$

$$C \otimes x \leq d$$

is called max-linear program with one-sided equations and inequalities. The set of optimal solutions are denoted by $S^{min}(A, C, b, d)$ and $S^{max}(A, C, b, d)$ respectively. We note that it would be possible to convert equations to inequalities and conversely but this would result in an increase in the number of constraints or variables and thus increasing the computational complexity. The following algorithm does not require any new constraint or variable and solves the problem. We let $(A \otimes x)_i = \max_{j \in N}(a_{ij} + x_j)$ and let $x_j = f(x)$ for some $j \in N$, then x_j is called active. Also a variable is called active on the constraint equation if the value $(A \otimes x)_i$ is attained at the term x_j. This follows from the previous theorem that $\hat{x} \in S^{max}(A, C, b, d)$. We now present a polynomial algorithm which finds $x \in S^{min}(A, C, b, d)$ or recognizes that $S^{min}(A, C, b, d) = \emptyset$. Since $x \in S(A, C, b, d)$ or $S(A, C, b, d) = \emptyset$, we assume in the following algorithm that $S(A, C, b, d) \neq \emptyset$ and also $S^{min}(A, C, b, d) \neq \emptyset$. We state the algorithm as follow:

Algorithm (Max-linear program with one-sided equations and inequalities)

Input: $f = (f_1, f_2, ..., f_n)^T \in \mathbb{R}^n$,$b = (b_1, b_2, ..., b_k)^T \in \mathbb{R}^k$, $d = (d_1, d_2, ..., d_r)^T \in$

$\mathbb{R}^r, A = (a_{ij} \in \mathbb{R}^{k\times n}$ and $C = (c_{ij}) \in \mathbb{R}^{r\times n}$.

Output: $x \in S^{min}(A, C, b, d)$

1. Find $\overline{x}, \overline{\overline{x}}, \hat{x}$ and $K_j, j \in J; J = \{j \in N; \overline{\overline{x}}_j \geq \overline{x}_j\}$.

2. $x := \hat{x}$

3. $H(x) := \{j \in N; f_j + x_j = f(x)\}$

4. $J := J \setminus H(x)$

5. If $\bigcup_{j\in J} K_j \neq K$ then stop $(x \in S^{min}(A, C, b, d))$

6. Set x_j small enough for every $j \in H(x)$

7. Go to 3

Max-Linear Program with two sided constraints

Here, we show an extension of the previous linear programming formulation in max-plus algebra as a special case. Suppose $c = (c_1, c_2, ..., c_m)^T$, $d = (d_1, d_2, ..., d_m)^T \in \mathbb{R}^m$, $A = (a_{ij})$ and $B = (b_{ij}) \in \mathbb{R}^{m\times n}$ are given matrices and vectors, then the system $A\otimes x \oplus c = B \otimes x \oplus d$ is called non-homogeneous two-sided max-linear system (Butkovič and Aminu, 2009) . The following maximization problem is the max-linear program with two sided constraints.

$$f(x) = f^T \otimes x = \max(f_1 + x_1, f_1 + x_2, ..., f_n + x_n) \text{ subject to}$$

$$A \otimes x \oplus c = B \otimes x \oplus d$$

where $f = (f_1, f_2, ..., f_n)^T \in \mathbb{R}^n \in \mathbb{R}^n$, $c = (c_1, c_2, ..., c_m)^T$, $d = (d_1, d_2, ..., d_m)^T \in \mathbb{R}^m$, $A = (a_{ij})$ and $B = (b_{ij}) \in \mathbb{R}^{m \times n}$ are given matrices and vectors.

1.2.10 Eigenvalues and Eigenvectors

In max-plus algebra the max-plus eigenvalue and eigenvectors have a graph theoretical interpretation. We explain this graph theoretical interpretations through a series of definitions. For an $n \times n$ matrix A, the digraph(or directed graph) of A is the graph with vertices $1, ..., n$ where there is a directed arc from i to j with weight a_{ij} if and only if $a_{ij} \neq -\infty$. A path is a sequence of distinct vertices $i_1, i_2, ..., i_k$ such that there is an arc from i_j to i_{j+1} for $j = 1, ..., k-1$. The weight of a path is the sum of the weights of the arcs that make up that path (Sergreev, 2007). The digraph D_A is strongly connected if there is a path from any vertex to any other vertex. If D_A is strongly connected, then we say the matrix A is irreducible.

Definitions

1. A cycle σ, is a sequence $i_1, i_2, ..., i_k$ of distinct vertices such that $i_1 \to i_2, i_2 \to i_3, ..., i_k \to i_1$ of adjacent arcs in the digraph that starts and ends at the same vertex and does not travel through any other vertex more than

once. This can be described as a sequence of vertices. The number of arcs in a cycle is called its length l_σ. Note that for any $\sigma, l_\sigma \leq n$.

2. A loop is a cycle with length 1. For a cycle σ, the sum of its arc weights divided by the length l_σ, is called the mean $M(\sigma)$.

We are interested in the maximum of these cycle means, where the maximum is taken over all circuits in the matrix A.

3. For a matrix A with distinct cycles $\sigma_1, \sigma_2, ..., \sigma_n$, we define the maximum cycle mean by $\mu(A) = M(\sigma_i)$. A graph that contains only the cycles with the maximum cycle mean is called a critical graph.

We note that the maximum cycle mean of a matrix is of fundamental importance in max-plus algebra because for any square matrix it is the greatest max-algebraic eigenvalue.

The max-plus eigenvalue-eigenvector problem $A \otimes x = \lambda \otimes x$ for $A = (a_{ij}) \in \mathbb{R}^{n \times n}_{max}$ where λ is the eigenvalue of A and x, the corresponding eigenvector was one of the early problems of max-algebra. Here, we discuss only the case where A does not contain the element ε as this will involve some combinatorial features of the eigen-problem. We note that for every $A = (a_{ij}) \in \mathbb{R}^{n \times n}_{max}$ there is a unique value of $\lambda = \lambda(A)$, called the eigenvalue of A to which there is an $x \in \mathbb{R}^n_{max}$ satisfying the equation $A \otimes x = \lambda \otimes x$.

The unique eigenvalue is the maximum cycle mean in D_A that is

$$\lambda(A) = \max_{\sigma} \frac{\omega(A, \sigma)}{l(\sigma)}$$

where $\sigma = (i_1, ..., i_k)$ denotes an elementary cycle (that is a cycle with no repeated node except the first and the last one) in D_A, $\omega(A, \sigma) = a_{i_1 i_2} + ... + a_{i_k i_1}$ is the weight of σ and $l(\sigma) = k$ is the length of σ. The maximization is taken over elementary cycles of all lengths in D_A, including the loops. The computation of the maximum cycle mean is difficult since the number of cycles is very large in general. The best known method currently is Karp's algorithm which is based on the following theorem:

Theorem 1.13. *If $A = (a_{ij}) \in \mathbb{R}_{max}^{n \times n}$ is irreducible then*

$$\lambda(A) = \max_{j \in N} \min_{k \in N} \frac{F_{n+1}(j) - F_k(j)}{n + 1 - j}$$

where $F_k(j)$ is the maximum weight of an $s - j$ path of length k.

For proof see (Karp, 1978).

1.2.11 The Transitive Closures

We discuss transitive closures which are of fundamental importance in max-algebra and enable us discuss to non-trivial solutions. We define the infinite series as follows:

$$\Gamma(A) = A \otimes A^2 \otimes A^3 \otimes ... \tag{1.5}$$

$$\Delta(A) = I \oplus \Gamma(A) = I \oplus A \oplus A^2 \oplus A^3 \oplus ... \tag{1.6}$$

where $A \in \mathbb{R}^{n\times n}_{max}$ and show how $\Gamma(A)$ and $\Delta(A)$ can be used for finding a solution to the equations $A \otimes x = x$ and $A \otimes x \leq \lambda \otimes x$. The matrix $\Gamma(A)$ is called the weak transitive closure of A if these series converge, and $\Delta(A)$ is the strong transitive closure of A. We use these matrices to describe all non-trivial solutions (if any) to the max-plus equation $A \otimes x = x$ in the case of $\Gamma(A)$ and all finite solutions to $A \otimes x \leq \lambda \otimes x$ in the case of $\Delta(A)$. To see this, we propose that (1.5) converges if and only if $\lambda(A) \leq 0$. If $\lambda(A) \leq 0$ then $\Gamma(A) = A \oplus A^2 \oplus A^3 \oplus ... \oplus A^k$ for every $k \geq n$. If A is irreducible and $n > 1$ then $\Gamma(A)$ is finite. We consider the matrix $A^2 = A \otimes A$ where its elements are $\Sigma^{\oplus}_{k\in N} a_{ik} \otimes a_{kj} = \max_{k\in N}(a_{ik} + a_{kj})$ that is the greatest weights of paths of length 2 and in general A^k represents greatest weights of paths of length k. Hence if $\lambda(A) \leq 0$ then it implies $A^k \leq A \oplus A^2 \oplus A^3 \oplus ... \oplus A^n$ for every $k \geq 1$ and therefore $\Gamma(A)$ for any matrix with $\lambda(A) \leq 0$ exists and is equal to $A \oplus A^2 \oplus A^3 \oplus ... \oplus A^n$. Again if $\lambda(A) \leq 0$ we have $\Delta(A) = I \oplus \Gamma(A) = I \oplus A \oplus A^2 \oplus A^3 \oplus ... \oplus A^k$ for every $k \geq n-1$ which implies that $A \otimes \Delta(A) = A \otimes (I \oplus A \oplus A^2 \oplus A^3 \oplus ... \oplus A^{k-1} =$

$A \oplus A^2 \oplus A^3 \oplus ... \oplus A^n = \Gamma(A) \leq \Delta(A)$, that is every column of $\Delta(A)$ is a solution to $A \otimes x \leq x$

Similarly if $\lambda(A) \leq 0$ we have $A \otimes \Gamma(A) = A \otimes (A \oplus A^2 \oplus A^3 \oplus ... \oplus A^n) = A^2 \oplus A^3 \oplus ... \oplus A^{n+1} \leq A \oplus A^2 \oplus A^3 \oplus ... \oplus A^{n+1} = \Gamma(A)$ which also implies that every column of $\Gamma(A)$ is also a solution to $A \otimes x \leq x$. If $\lambda(A) = 0$, then at least one column of $\Gamma(A)$ is a solution to $A \otimes x = x$. We note that $A \otimes x \leq \lambda \otimes x$ has a finite solution if and only if $\lambda(A) \leq \lambda$.

1.2.12 Linear Programming Approach to Finding Eigenvalues in Max-plus Algebra

We consider a linear programming approach of finding the eigenvalue and the corresponding eigenvector in the max-plus environment. We consider the matrix $A \in \mathbb{R}^{n \times n}$ and the equation $A \otimes x = \lambda \otimes x$, where x is an eigenvector and λ an eigenvalue. If

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \quad and \quad x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix},$$

then

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \otimes \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \lambda \otimes \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

Rewriting the above matrix equation in max-plus we have:

$$\begin{aligned} (a_{11} \otimes x_1) \oplus (a_{12} \otimes x_2) \oplus (a_{13} \otimes x_3) &= \lambda + x_1 \\ (a_{21} \otimes x_1) \oplus (a_{22} \otimes x_2) \oplus (a_{23} \otimes x_3) &= \lambda + x_2 \\ (a_{31} \otimes x_1) \oplus (a_{32} \otimes x_2) \oplus (a_{33} \otimes x_3) &= \lambda + x_3 \end{aligned}$$

Using the standard max-plus notation, this is written as:

$$max\{(a_{11} + x_1), (a_{12} + x_2), (a_{13} + x_3)\} = \lambda + x_1$$

$$max\{(a_{21} + x_1), (a_{22} + x_2), (a_{23} + x_3)\} = \lambda + x_2$$

$$max\{(a_{31} + x_1), (a_{32} + x_2), (a_{33} + x_3)\} = \lambda + x_3.$$

This leads to the following system of upper bounds on $a_{ij} + x_j$ for $i, j = 1, 2, 3$

$$a_{11} + x_1 \leq \lambda + x_1,$$

$$a_{12} + x_2 \leq \lambda + x_1,$$

$$a_{13} + x_3 \leq \lambda + x_1,$$

$$a_{21} + x_1 \leq \lambda + x_2,$$

$$a_{22} + x_2 \leq \lambda + x_2,$$

$$a_{23} + x_3 \leq \lambda + x_2,$$

$$a_{31} + x_1 \leq \lambda + x_3,$$

$$a_{32} + x_2 \leq \lambda + x_3,$$

$$a_{33} + x_3 \leq \lambda + x_3.$$

And this can be written in matrix form as:

$$A \otimes \mathbf{x} = x \otimes \lambda = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & 1 \\ 1 & 0 & -1 & 1 \\ -1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & -1 & 1 \\ -1 & 0 & 1 & 1 \\ 0 & -1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \lambda \end{pmatrix} \geq \begin{pmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \\ a_{31} \\ a_{32} \\ a_{33} \end{pmatrix} \qquad (1.7)$$

Let H_n be an $n^2 \times (n+1)$ matrix in the conventional algebra, then the simplex method solves the linear programming problem

$$\text{Primal}: \quad \max \ b^T y \ \text{ subject to } \ H_n^T y = c, \ \ y \geq 0, \text{and}$$

$$\text{Dual}: \quad \min \ c^T z \ \text{ subject to } \ H_n z \geq b \tag{1.8}$$

We note the formulation (1.7) is the same as (1.8) so we can use (1.8) to find the eigenvalue and an eigenvector of A by linear programming. We need to minimize $c^T z$ so that $H_n z \geq b$. Let $c = [0, 0, ..., 0, 1]^T$, then $c^T z = \lambda$ and the sum of all entries of $H_n z$ is equal to $n^2\lambda = n^2 c^T z$.

$$H_n z \geq b \ \Rightarrow \ n^2 c^T z \geq \Sigma_{i=1}^{n^2} b_i$$

$$\Rightarrow \ c^T z \geq \frac{\Sigma_{i=1}^{n^2} b_i}{n^2}$$

Hence minimizing $c^T z$ is the same as maximizing $\frac{\Sigma_{i=1}^{n^2} b_i}{n^2}$. We note that $\frac{\Sigma_{i=1}^{n^2} b_i}{n^2}$ is $b^T y$, where $y = \frac{1}{n^2}$ and this gives $H_n^T y = c$. Linear programming guarantees that an eigenvalue and the eigenvector can be found in a polynomial time.

1.2.13 Cayley-Hamilton Theorem and the Max-plus Characteristic Equation

The characteristic equation of a matrix equation in conventional algebra is used to determine the eigenvalue of a matrix. Let C_n^k be the set of all subsets of k elements of the set $\{1, 2, ..., n\}$. If A is an $n \times n$ matrix and $\varphi \subset \{1, 2, ..., n\}$, the submatrix obtained by removing all rows and columns of A except those denoted by φ is denoted by $A_{\varphi\varphi}$. The following theorem guarantees the determination of eigenvalues from the characteristic equation.

Theorem 1.14. *Suppose $A \in \mathbb{R}^{n \times n}$, if*

$$det(\lambda I - A) = \lambda^n + c_1\lambda^{n-1} + ... + c_{n-1}\lambda + c_n \quad then$$

$$A^n + c_1 A^{n-1} + ... + c_{n-1}A + c_n I = 0$$

where the c_k are given by $c_k = (-1)^k \sum_{\sigma \in C_n^k} \det A_{\sigma\sigma}$.

The Cayley-Hamilton Theorem can be translated into Max-plus algebra. Consider the matrix e^{sA}, the characteristic polynomial of the matrix-valued function e^{sA} is given by

$$det(\lambda I - e^{sA}) = \lambda^n + \gamma_1(s)\lambda^{n-1} + ... + \gamma_{n-1}(s)\lambda + \gamma_n(s) \tag{1.9}$$

with coefficients $\gamma_k(s) = (-1)^k \sum_{\varphi \in C_n^k} \det(e^{sA_{\varphi\varphi}})$.

Therefore $(e^{sA})^n + \gamma_1(s)(e^{sA})^{n-1} + \ldots + \gamma_{n-1}(s)e^{sA} + \gamma_n I = 0$. This is the result of applying the above theorem to the matrix e^{sA}. The Max-plus characteristic equation of A is also found by rearranging (1.9) and removing the $\gamma_k(s)\lambda^{n-k}$ for which $\lambda_k(s)$ has a negative leading coefficient to the right side, replace λ with $e^{s\lambda}$. The reason for doing this is that subtraction is not defined in the max-plus algebra. The max-plus characteristic equation of $A \in \mathbb{R}_{max}^{n\times n}$ is defined as $\lambda^{\otimes n} \oplus \bigoplus_{k\in l} d_k \otimes \lambda^{\otimes n-k} = \bigoplus_{k\in j} d_k \otimes \lambda^{\otimes n-k}$ and A satisfies its own characteristic equation $A^{\otimes n} \oplus \bigoplus_{k\in l} d_k \otimes A^{\otimes n-k} = \bigoplus_{k\in j} d_k \otimes A^{\otimes n-k}$.

Chapter 2

Some Applications of Max-plus Algebra

2.1 Introduction

In chapter one, we discussed some key properties of the max-plus algebra and how these properties are used to analyze some linear algebra concepts. In this chapter, we present some applications of the max-plus algebra as one of the many idempotent semi-rings, which have been considered in various field of mathematics. Max-plus algebra has found applications in many areas such as optimization, mathematical physics, algebraic geometry and combinatorics.

It has also been used extensively in control theory, discrete event processes, telecommunication networks, machine scheduling, manufacturing systems, traffic control and parallel processing systems (Gaubert, 1997).

Max-plus serves as a powerful tool required to analyze the highly nonlinear equations that arise in most applied models in Mathematics. Many of the model equations formulated for describing the phenomena of these applications are nonlinear in the conventional sense. When max-plus is applied, the highly nonlinear equations are transformed into linear in the max-plus algebra. This is one of the reasons why it is used in various areas of application. We present some of the applications of the max-plus algebra to finite and infinite horizon optimal control problems, finite element method for control problems, linear discrete event systems and asymptotic growth rate analysis.

2.2 Control Problems

Some entirely new classes of numerical methods for the solution of nonlinear control problems like Hamilton-Jacobi-Bellman Partial Differential Equations (HJB PDEs) are based on max-plus formulation of these problems. One of the approaches to solving nonlinear control problems is Dynamic Programming (DP) (Bertsekas, 1987). The Dynamic Programming Principle (DPP) is an operator mapping the the value function (optimal cost as a function of system state) at one time to the value function at a later time. In the continuous-time case, if one takes the limit in the DPP as the time-interval goes to zero, the Dynamic Programming Equation (DPE) is obtained. For most continuous-time/continuous-space problems, this DPE takes the form of an HJB PDE which is a nonlinear first-order PDE.

Max-plus methods work directly with the DPP rather than the limit PDE. The DPP corresponds to an operator and the time-step parameterized group of DP operators for a problem is a semi-group whose generator is the HJB PDE. The naturalness of the max-plus method for solution of deterministic control problems (and HJB PDEs) is reflected by the fact that these operators are max-linear operators. The solution of time-dependent problems reduces to propagation by the linear operator, steady-state problems reduce to eigenfunction problems corresponding to the linear operator (Basser, 1991).

2.2.1 Finite and Infinite-time horizon optimal control problems

In this subsection we discuss finite and infinite-time horizon applications in optimal control problems. Let the state at time t be denoted by ξ_t, the state space be $\mathbb{R}^n$, and points in the state space by $x \in \mathbb{R}^n$. One obvious category of problems appropriate for the max-plus analysis are the finite time-horizon optimal control problems. Let the dynamics and initial condition be

$$\begin{aligned} \dot{\xi} &= f(\xi, u) \\ \xi_s &= x, \end{aligned}$$

where u_t is a control input process taking values in the set $U \subseteq \mathbb{R}^k$. The mathematics for solution of associated HJB PDE will be the same regardless of the real-world interpretation. Let $\mathcal{U}$ be the space of functions or control space for the input. If the finite time-horizon problem is over time interval $[s, T]$, where T is the terminal time, then the space of input functions is $u \in \mathcal{U}_{[s,T]}$. For example, one might have $\mathcal{U}_{[s,T]} = \{u : [s,T] \to U | u \text{ is measurable}\}$ if U is compact. If U is not compact (e.g., $U = \mathbb{R}^k$), then one might take $\mathcal{U}_{[s,T]} = \left\{u : [s,T] \to U \mid \int_{[s,T]} | u_t|^2 dt < \infty\right\}$. It will generally be of interest to consider optimization of a payoff (or cost criterion) through choice of this input process. In the case where u represents a controller, we may interpret the problem as an optimal control problem with the given cost criterion. Such a criterion would typically take a form such as

$$J(s, x; u) = \int_s^T L(\xi_t, u_t)dt + \phi(\xi_T)$$

(McEneaney, 2004) Other problem formulations where one takes suprema or limit suprema over time are considered next. To maximize the payoff, the value function of this problem would be

$$V(s, x) = \sup_{u \in \mathcal{U}} J(s, x; u)$$

Problems where one maximizes over the control space is usually amenable to max-plus methods. If instead, one wishes to minimize, then the min-plus algebra would be appropriate. A particularly interesting class of problems will be the robust/H_∞ infinity time-horizon control problems. The typical dynamics in such cases take the form

$$\begin{aligned} \dot{\xi} &= f(\xi, v(\xi)) + \sigma(\xi)u \\ \xi_0 &= x \in \mathbb{R}^n. \end{aligned}$$

where $v(x)$ is the fixed feedback controller, $g(\xi) = f(\xi, v(\xi))$ represents the nominal dynamics (that is, the dynamics in the absence of a disturbance process), u represents an input disturbance process. If the range space of u were $\mathbb{R}^k$, then the disturbance space (also referred to as the control space of u) is given by:

$$\mathcal{U} = L_2([0,\infty);\mathbb{R}^k) = \{u : [0,\infty) \to \mathbb{R}^k | u_{[0,T]} \in L_2(0,T) \ \forall \ T \in [0,\infty)\}$$

where $L_2(0,T)$ is the set of square-integrable functions over $[0,T]$. This system is said to satisfy an H_∞ criterion with disturbance parameter $\gamma \in (0,\infty)$ if there exists a locally bounded $\beta : \mathbb{R}^n \to [0,\infty)$ where $\beta(0) = 0$ such that

$$\int_0^T L(\xi_t)dt \leq \beta(x) + \frac{\gamma^2}{2}\|u\|^2_{[0,T]} = \beta(x) + \frac{\gamma^2}{2}\int_0^T |u|^2 dt$$

for all $T \in [0,\infty)$ and all $u \in \mathcal{U}$. $L(\cdot)$ is assumed nonnegative and often takes the form of a quadratic. We note that a function is locally bounded if it is bounded on compact sets. The associated value function is given by:

$$\begin{aligned} W(x) &= \sup_{u\in\mathcal{U}} \sup_{T\in[0,\infty)} \int_0^T (L(\xi_t) - \frac{\gamma^2}{2}|u|^2)dt \\ &= \sup_{T\in[0,\infty)} \sup_{u\in\mathcal{U}} \int_0^T (L(\xi_t) - \frac{\gamma^2}{2}|u|^2)dt \\ &= \lim_{T\to\infty} \sup_{u\in\mathcal{U}} \int_0^T (L(\xi_t) - \frac{\gamma^2}{2}|u|^2)dt \end{aligned}$$

see for example (Lions, 1982).

2.2.2 Max-plus Finite Element Method For Optimal Control Problems

We now develop the max-plus finite element method to solve finite deterministic optimal control problem. Let the optimal control problem be

$$\text{maximize} \int_0^T \ell(x(s), u(s))ds + \phi(x(T)) \tag{2.1}$$

subject to the following constraints

$$\dot{x}(s) = f(x(s),\ u(s)),\ \ x(0) = x,\ \ x(s) \in X,\ \ u(s) \in U \tag{2.2}$$

for all $0 \leq s \leq T$. $X \subset \mathbb{R}^n$ is the state space, $U \subset \mathbb{R}^m$ is the set of control values, $T > 0$ is the horizon and the initial condition is $x \in X$. We assume that the map $u(\cdot)$ is measurable, and that the map $x(\cdot)$ is continuous. Assume also that the instantaneous reward or Lagrangian $\ell : X \times U \to \mathbb{R}$, and the dynamics $f : X \times U \to \mathbb{R}^n$, are sufficiently regular maps, and that the terminal reward ϕ is a map $X \to \mathbb{R} \cup -\infty$. The value function v associates to any $(x, t) \in X \times [0, T]$ the supremum $v(x, t)$ of $\int_0^T \ell(x(s), u(s))ds + \phi(x(T))$, under the constraint (2.2), for $0 \leq s \leq t$. Under certain regularity assumptions, v is solution of the Hamilton-Jacobi equation of the form:

$$-\frac{\partial v}{\partial t} + H(x, \frac{\partial v}{\partial x}) = 0,\ \ (x, t) \in X \times (0, T] \tag{2.3}$$

with initial conditions

$$v(x,0) = \phi(x), \;\; x \in X, \tag{2.4}$$

where $H(x,p) = \sup_{u \in \mathcal{U}} \ell(x,u) + p.f(x,u)$ is the Hamiltonian of the problem (Lions, 1982). The evolution semi-group S^t of (2.3) and (2.4) associates to any map ϕ the function $v^t = v(.,t)$, where v is the value function of the optimal control problem (2.1). The evolution semi-group S^t is max-plus linear (Kolokoltsov et al, 1997) that is for all maps f,g from X to $\mathbb{R}_{max}$, and for all $\lambda \in \mathbb{R}_{max}$, we have

$$\begin{aligned} S^t(f \oplus g) &= S^t f \oplus S^t g \\ S^t(\lambda f) &= \lambda S^t f \end{aligned}$$

where $f \oplus g$ is the map $x \to f(x) \oplus g(x)$, and λf denotes the map $x \to \lambda \otimes f(x)$.

Max-plus finite element solution method

In this section, we look at the max-plus finite element method to solve problem (2.1). Applying the property of semi-groups $S^{t+t'} = S^t.S^{t'}$, for $t, t' > 0$, we have the following recursive equation:

$$v^{t+\delta t} = S^{\delta t} v^t, \;\; t = 0, \delta t, ..., T - \delta \tag{2.5}$$

where $v^0 = \phi$ and $\delta = \frac{T}{N}$, for some N. Let $\mathcal{W}$ be semi-module $\mathbb{R}_{max}$ of functions from X to $\mathbb{R}_{max}$ such that $\phi \in \mathcal{W}$ and for all $v \in \mathcal{W}$, $t > 0$, $S^t v \in \mathcal{W}$. Let $\mathcal{Z}$ be a semi-module of test functions from X to $\mathbb{R}_{max}$, then the max-plus scalar product is $\langle u|v\rangle = \sup_{x\in X} u(x) \otimes v(x)$, for all functions $u, v : X \to \mathbb{R}_{max}$. Let equation (2.5) be replaced by the following:

$$\langle z|v^{t+\delta}\rangle = \langle z|S^{\delta}v^t\rangle, \ \forall z \in \mathcal{Z} \tag{2.6}$$

for $t = 0, \delta t, ..., T - \delta$, with $v^{\delta}, ..., v^T \in \mathcal{W}$. The above equation (2.6) can be used to define a notion of solution to Hamilton-Jacobi equations (Kolokoltsov et al, 1988). Let a semi-module $\mathcal{W}_h \subset \mathcal{W}$ be generated by the family $\{w_i\}_{1\leq i\leq p}$. The function w_i is called the finite elements. We approximate v^t by $v_h^t \in \mathcal{W}_h$ that is $v^t \simeq v_h^t = \bigotimes_{i=1}^{p} w_i\lambda_i^t$, where $\lambda_i^t \in \mathbb{R}_{max}$. We also consider a semi-module $\mathcal{Z}_h \subset \mathcal{Z}$ generated by the family $\{z_j\}_{1\leq i\leq q}$, then the functions $z_1, ..., z_q$ act as test functions. We replace equation (2.6) by the following equation:

$$\langle z_j|v_h^{t+\delta}\rangle = \langle z_j|S^{\delta}v_h^t\rangle, \ \forall \ 1 \leq i \leq q \tag{2.7}$$

for $t = 0, \delta t, ..., T - \delta$, with $v_h^0 = \phi_h \simeq \phi$ and $v_h^t \in \mathcal{W}_h, t = 0, \delta t, ..., T$. Equation (2.7) need not have a solution therefore we look for the maximal sub-solution, that is the maximal solution $v_h^{t+\delta} \in \mathcal{W}_h$ of

$$\langle z_j | v_h^{t+\delta} \rangle \leq \langle z_j | S^\delta v_h^t \rangle, \ \forall 1 \leq i \leq q \tag{2.8}$$

We also consider the approximate value function v_h^0 at time 0 the maximal solution $v_h^0 \in \mathcal{W}_h$ of

$$v_h^0 \leq v^0 \tag{2.9}$$

We denote by W_h the max-plus operator from $\mathbb{R}^p_{max}$ to $\mathcal{W}$ with matrix $W_h = col(w_i)_{1\leq i \leq p}$, and by Z_h^* the max-plus linear operator from $\mathcal{W}$ to $\mathbb{R}^q_{max}$ whose transpose matrix is $Z_h = col(z_j)_{1\leq i \leq q}$. This implies that $W_h \lambda = \bigotimes_{i=1}^p w_i \lambda_i$ for all $\lambda = (\lambda_i)_{i=1,...,p} \in \mathbb{R}^p_{max}$, and $(Z_h^* v)_j = \langle z_j | v \rangle$ for all $v \in \mathcal{W}$ and $j = 1, ..., q$. The maximal solution $v_h^{t+\delta} \in W_h$ of (2.8) and (2.9) is given by $v_h^{t+\delta} = S^\delta v_h^t$, where $S_h^\delta = \Pi_{W_h}^{Z_h^*} \cdot S^\delta$.

Let $v_h^t \in W_h$ be the maximal solution of (2.8) and (2.9), for $t = 0, \delta t, ..., T$. Then for every $t = 0, \delta t, ..., T$, there exists $\lambda^t \in \mathbb{R}^p_{max}$ such that $v_h^t = W_h \lambda^t$. Moreover the maximal λ^t satisfying these conditions verifies the recursive equation

$$\lambda^t = (Z_h^* W_h) \tag{2.10}$$

and the initial conditions $\lambda^0 = W_h$. For $1 \leq i \leq p$ and $1 \leq j \leq q$, we define

$$(A_h)_{ji} = \langle z_j | W_i \rangle \tag{2.11}$$

$$(B_h)_{ji} = \langle z_j | S^\delta w_i \rangle \qquad (2.12)$$

where A_h and B_h are respectively the matrices of the max-plus linear operators $Z_h^* W_h$ and $Z_h^* S^\delta W_h$. Equation (2.10) may be written explicitly for $1 \leq i \leq p$ as

$$\lambda_i^t = \min_{1\leq j\leq q} (-1(A_h)_{ji} + \max_{1\leq j\leq q}((B_h)_{jk} + \lambda_k^{t-\delta})).$$

This recursion may be interpreted as the dynamic programming equation of a deterministic zero-sum two players game with finite action and state spaces. The max-plus finite element method can be summarized as follows:

1. Choose $\delta = \frac{T}{N}$ and the finite elements $(w_i)_{1\leq i\leq p}$ and $(z_j)_{1\leq j\leq q}$,
2. Compute the matrix A_h by (2.11) and the matrix B_h by (2.12),
3. Compute $\lambda^0 = W_h$ and $v_h^0 = W_h \lambda^0$.
4. For $t = \delta, 2\delta, ..., T$, compute $\lambda^t = A_h$ and $v_h^t = W_h \lambda^t$

then v_h^t approximates the function at time t, v^t.

2.3 Optimal Control Approach for Max-plus Linear Discrete Event Systems (DES)

2.3.1 Max-plus linear state space models

DES can be modeled by a max-plus algebraic model of the form:

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

see for example (Heidergott et al, 2006).

The vector x represents the state, u the input vector and y the output vector of the system. The components of the input, output and the state are event times and k is an event counter. For a manufacturing system, $u(k)$ would represent the time at which raw material is fed to the system for the k^{th} time, $x(k)$ the time at which the machine start processing the k^{th} batch of intermediate products, and $y(k)$ the time at which the k^{th} batch of finished products leaves the system. To determine the input-output behavior of the DES we have the following:

$$\begin{aligned} x(1) &= A \otimes x(0) \oplus B \otimes u(1) \\ x(2) &= A \otimes x(1) \oplus B \otimes u(2) \end{aligned}$$

$$= A^{\otimes^2} \otimes x(0) \oplus A \otimes B \otimes u(1) \oplus B \otimes u(2)$$

and so on, which yields $x(k) = A^{\otimes^k} \otimes x(0) \oplus \bigoplus_{i=1}^{k} A^{\otimes^{k-i}} \otimes B \otimes u(i)$, for $k = 1, 2, \ldots$. Hence

$$y(k) = C \otimes A^{\otimes^k} \otimes x(0) \oplus \bigoplus_{i=1}^{k} A^{\otimes^{k-i}} \otimes B \otimes u(i) \tag{2.13}$$

for $k = 1, 2, \ldots$.

Let $y_1 = \{y_1(k)\}_{k=1}^{\infty}$ be the output sequence that corresponds to the input sequence $u(1) = \{u_1(k)\}_{k=1}^{\infty}$ with initial condition $x_1(0)$ and let $y_2 = \{y_2(k)\}_{k=1}^{\infty}$ be the output sequence that corresponds to the input sequence $u(2) = \{u_2(k)\}_{k=1}^{\infty}$ with initial condition $x_2(0)$. Let $\alpha, \beta \in \mathbb{R} \cup \{-\infty\}$ it follows from equation (2.13) that the output sequence that corresponds to the input sequence $\alpha \otimes u_1 \oplus \beta \otimes u_2 = \{\alpha \otimes u_1(k) \oplus \beta \otimes u_2(k)\}_{k=1}^{\infty}$ is given by $\alpha \otimes y_1 \oplus \beta \otimes y_2$.

2.3.2 Model Predictive Control (MPC)

MPC is a very popular controller design method in the process industry. Process industry is always characterized by changes in production parameters, and MPC has shown to respond effectively to these parameter changes in many practical process control problems and applications. MPC makes use of a finite horizon which allows the inclusion of such additional inputs. It also uses linear discrete event system models for the process to be controlled. A cost criterion J is formulated that reflects the reference tracking error J_{out} and the control effort J_{in}.

$$\begin{aligned} J &= J_{out} + \lambda J_{in} \\ &= \Sigma_{j=1}^{N_p} \parallel (\hat{y}(k+j \mid k) - r(k+j) \parallel^2 \\ &+ \lambda \Sigma_{j=1}^{N_p} \parallel u(k+j-1) \parallel^2 \end{aligned} \quad (2.14)$$

where $\hat{y}(k+j \mid k)$ is the estimate of the output at time step $k+j$, r is a reference signal λ is a non-negative scalar, and N_p is the prediction horizon. The input is taken to be constant from a certain time on, that is: $u(k+j) = u(k+N_c-1)$ for $j = N_c, ..., N_p - 1$ where N_c is the control horizon. The use of a control horizon leads to a reduction of the number of optimization variables which gives a smoother controller signal and a stabilizing effect.

2.3.3 Max-plus Linear Input-Output (MPLIO) Systems

We consider systems that can be described by the input-output relation:

$$\begin{aligned} y(k) &= a_1 \otimes y(k-1) \oplus a_2 \otimes y(k-2) \oplus ... \oplus a_n \otimes y(k-n) \\ &\oplus b_0 \otimes uk \oplus b_1 \otimes u(k-1) \oplus ... \oplus b_m \otimes u(k-m) \end{aligned} \quad (2.15)$$

This can be written in polynomial form as

$$y(k) = A(\gamma)y(k) \oplus B(\gamma)u(k) \tag{2.16}$$

where $A(\gamma)$ and $B(\gamma)$ are polynomial operators, that is:

$$A(\gamma) = a_1 \otimes \gamma^1 \oplus a_2\gamma_2 \oplus ... \oplus a_n \otimes \gamma^n \tag{2.17}$$

$$B(\gamma) = b_0 \otimes \gamma^0 \oplus b_1\gamma_1 \oplus ... \oplus b_m \otimes \gamma^m \tag{2.18}$$

Discrete event systems (DES) that can be described by this model will be called max-plus linear input-output(MPLIO) systems. The input $u(k)$ contains the instants at which the input events occur for the k^{th} time, and the output $y(k)$ contains the time instants at which the output events occur for the k^{th} time. Note that for a manufacturing system, $u(k)$ contains the time instants at which the k^{th} batch of raw material is fed to the system, and $y(k)$ the time instants at which the k^{th} batch of finished product leaves the system. The entries of the polynomials $A(\gamma)$ and $B(\gamma)$ are varying in time due to slow changes in the system.

2.3.4 Model Predictive Control for MPLIO Systems

In this subsection we consider a cost criterion $J(k)$ that reflect the output and input cost functions $J_{out}(k)$ and $J_{in}(k)$ respectively in the event period

$[k, k+N_p-1]$ as $J(k) = J_{out}(k) + \lambda J_{in}(k)$ in which

$$
\begin{aligned}
J_{out}(k) &= \Sigma_{j=0}^{N_{p-1}} \max(\hat{y}(k+j \mid k) - r(k+j), 0) \\
J_{in}(k) &= -\Sigma_{j=0}^{N_{p-1}} u(k+j)
\end{aligned}
$$

where N_p is the prediction horizon and λ is a weighting parameter, $\hat{y}(k+j \mid k)$ is the prediction of the output signal $y(k+j)$, k is the event step and $r(k)$ is the due date signal. $J_{out}(k)$ reflect the due date error and $J_{in}(k)$ is used to penalize a large input buffer (De Schutter and Boom, 2001). In order to calculate the optimal MPC input signal, we need to make predictions of the output signal.

Theorem 2.1. *Consider an MPLIO system (2.3)-(2.5). For any non-negative integer j, there exist polynomials:*

$$
\begin{aligned}
C_j(\gamma) &= c_{1,j} \otimes \gamma^1 \oplus c_{2,j}\gamma^2 \oplus ... \oplus c_{n,j} \otimes \gamma^n \\
D_j(\gamma) &= d_{0,j} \otimes \gamma^0 \oplus d_{1,j}\gamma^1 \oplus ... \oplus d_{m-1,j} \otimes \gamma^{m-1} \\
F_j(\gamma) &= f_{0,j} \otimes \gamma^0 \oplus f_{1,j}\gamma^1 \oplus ... \oplus f_{j,j} \otimes \gamma^j
\end{aligned}
$$

such that

$$\hat{y}(k+j \mid k) = C_j(\gamma)y(k) \oplus D_j(\gamma)u(k-1) \oplus F_j(\gamma)u(k+j) \tag{2.19}$$

Proof. : (De Schutter and Boom, 2001). Define

$$\begin{aligned}
C_0(\gamma) &= A(\lambda) \\
D_0(\gamma) &= b_1 \otimes \gamma^0 \oplus b_2 \otimes \gamma^1 \oplus \ldots \oplus b_m \otimes \gamma^{m-1} \\
F_0(\gamma) &= b_0 \otimes \gamma^0
\end{aligned}$$

and for $j < 0$, $C_j(\gamma) = \gamma^{-j}$, $\quad D_j(\gamma) = \epsilon$, $\quad F_j(\gamma) = \epsilon$ then (2.19) is satisfied for $j = 0$ since $y(k) = A(\gamma)y(k) \oplus B(\gamma)u(k) = C_0(\gamma)y(k) \oplus D_0(\gamma)u(k-1) \oplus F_0(\gamma)u(k)$ and for $j < 0, i > 0$, we have $y(k-i) = \gamma^i y(k)$ Let for $j \in Z, j > 0$, the polynomials $C_{j-\ell}(\gamma), D_{j-\ell}(\gamma)$ and $F_{j-\ell}(\gamma)$ for all $\ell \in Z, \ell > 0$ be such that $\hat{y}(k+j-\ell \mid k) = C_{j-\ell}(\gamma)y(k) \oplus D_{j-\ell}(\gamma)u(k-1) \oplus F_{j-\ell}(\gamma)u(k+j-\ell)$ then

$$\begin{aligned}
\hat{y}(k+j \mid k) &= A(\gamma)\hat{y}(k+j \mid k) \otimes B(\gamma)u(k+j) \\
&= a_1\hat{y}(k+j-1 \mid k) \oplus a_2\hat{y}(k+j-2 \mid k) \oplus \ldots \oplus a_n\hat{y}(k+j-n \mid k) \\
&\oplus B(\gamma)u(k+j) \\
&= \bigoplus_{\ell=1}^{n} a_\ell \otimes (C_{j-\ell}(\gamma)y(k) \oplus D_{j-\ell}(\gamma)u(k-1) \oplus F_{j-\ell}(\gamma)u(k+j-\ell)) \\
&\oplus B(\gamma)u(k+j) \\
&= \bigoplus_{\ell=1}^{n}(a_\ell \otimes (C_{j-\ell}(\gamma))y(k) \oplus \bigoplus_{\ell=1}^{n}(a_\ell \otimes (D_{j-\ell}(\gamma))u(k-1) \\
&\oplus \bigoplus_{\ell=1}^{n}(a_\ell \otimes F_{j-\ell}(\gamma^\ell))u(k+j) \oplus B(\gamma)u(k+j)
\end{aligned}$$

$$
\begin{aligned}
&= \bigoplus_{\ell=1}^{n}(a_\ell \otimes (C_{j-\ell}(\gamma))y(k) \oplus \bigoplus_{\ell=1}^{n}(a_\ell \otimes (D_{j-\ell}(\gamma))u(k-1) \\
&\oplus \bigoplus_{\ell=1}^{n}(a_\ell \otimes F_{j-\ell}(\gamma^\ell) \oplus B(\gamma))u(k+j)
\end{aligned}
$$

Let $B_j^{fut}(\gamma)$ and $B_j^{past}(\gamma)$ be two polynomials for $j < m$, then

$$
\begin{aligned}
B_j^{fut}(\gamma) &= b_0 \otimes \gamma^0 \oplus b_1 \otimes \gamma^1 \oplus ... \oplus b_j \otimes \gamma^j \\
B_j^{past}(\gamma) &= b_{j+1} \otimes \gamma^0 \oplus b_{j+2} \otimes \gamma^1 \oplus ... \oplus b_m \otimes \gamma^{m-i-1}
\end{aligned}
$$

for $j \geq m$:

$$B_j^{fut}(\gamma) = B(\gamma) \qquad B_j^{past}(\gamma) = \epsilon$$

then for all $j \in Z, j > 0$

$$
\begin{aligned}
B(\gamma)u(k+j) &= B_j^{fut}(\gamma)u(k+j) \oplus B_j^{past}(\gamma) \otimes \gamma^{j+1}u(k+j) \\
&= B_j^{fut}(\gamma)u(k+j) \oplus B_j^{past}(\gamma)u(k-1)
\end{aligned}
$$

and so

$$
\begin{aligned}
\hat{y}(k+j \mid k) &= \bigoplus_{\ell=1}^{n}(a_\ell \otimes (C_{j-\ell}(\gamma))y(k) \oplus \bigoplus_{\ell=1}^{n}(a_\ell \otimes (D_{j-\ell}(\gamma))u(k-1) \\
&\oplus \bigoplus_{\ell=1}^{n}(a_\ell \otimes F_{j-\ell}(\gamma^\ell) \oplus B(\gamma))u(k+j) \oplus B_j^{past}(\gamma)u(k-1) \\
&= C_j(\gamma)y(k) \oplus D_j(\gamma)u(k-1) \oplus F_j(\gamma)u(k+j) \quad \text{where}
\end{aligned}
$$

$$C_j(\gamma) = \bigoplus_{\ell=1}^{n}(a_\ell \otimes C_{j-\ell}(\gamma))$$

$$
\begin{aligned}
D_j(\gamma) &= B_j^{past}(\gamma) \oplus \bigoplus_{\ell=1}^{n} (a_\ell \otimes D_{j-\ell}(\gamma)) \\
F_j(\gamma) &= B_j^{fut}(\gamma) \oplus \bigoplus_{\ell=1}^{n} (a_\ell \otimes F_{j-\ell}(\gamma) \otimes \gamma^\ell
\end{aligned}
$$

□

We note that the expression $C_j(\gamma)y(k) \oplus D_j(\gamma)u(k-1)$ in equation (2.22) depends on event previous steps and $F_j(\gamma)u(k+j)$ depends on present and future values of the input signal.

2.3.5 Asymptotic growth rate analysis and Limiting behaviour in max-plus

In this subsection we discuss how max-plus results can be used for global convergence of Genetic algorithms. We consider the sequences $\{x(k) : k \in \mathbb{N}\}$ be generated by $x(k+1) = A \otimes x(k)$ where $A \in \mathbb{R}_{max}^{n \times n}$ and $x(0) \in \mathbb{R}_{max}^n$ is the initial condition. Then the sequences can be described as $x(k) = A^{\otimes} \otimes x(0)$. The following results are from (Basser and Bernhard, 1991).

Definition

Let $\{x(k) : k \in \mathbb{N}\}$ be a sequence in $\mathbb{R}_{max}^n$ and assume that for all $j \in n$ the quantity κ_j, defined by $\lim_{k\to\infty} \frac{x_j(k)}{k}$, exists. Then the vector $\kappa = (\kappa_1, \kappa_2, ..., \kappa_n)^T$ is called the cycle time vector of the sequence $x(k)$. If all the $\kappa_j's$ have the same value, then this value is also called the asymptotic growth rate of the sequence $x(k)$. If A is irreducible with unique eigenvalue λ and associated eigenvector v, then for $x(0) = v$ it follows that $x(k) = A^{\otimes k} \otimes x(0) = \lambda^{\otimes k} \otimes v$. This implies that for any $j \in n$, $\lim_{k\to\infty} \frac{x_j(k)}{k} = \lambda$ and the asymptotic growth rate of $x(k)$ coincides with the eigenvalue of A. We state and prove the following theorem which confirms our claim.

Theorem 2.2. *Consider the recurrence relation $x(k+1) = A \otimes x(k), k \geq 0, x(0) = x_0 \in \mathbb{R}^n_{max}$ arbitrary. Then $\lim_{k\to\infty} \frac{x_j(k)}{k} = \lambda, \forall j = 1, ..., n.$*

Proof. Let v be an eigenvector of A such that $x_0 = v$ then, $x(k) = \lambda^{\otimes k} \otimes v \Rightarrow x(k) = k\lambda + v \Rightarrow \frac{x(k)}{k} = \lambda + \frac{v}{k} \Rightarrow \lim_{k\to\infty} \frac{x_j(k)}{k} = \lambda, \forall j = 1, ..., n.$ □

We note that for a regular matrix A, the cycle time vector of $x(k)$ exists and it is independent of the initial vector $x(0)$. Furthermore, if A is irreducible, then the asymptotic growth rate of any $x_j(k), j \in n$ is equal to the eigenvalue of A. The ℓ^∞-norm of a vector $v \in \mathbb{R}^n$ is defined as the maximum of the absolute value of all the entries of v. That is $\|v\|_\infty = \max_{i\in n} |v_i|$ for every $v \in \mathbb{R}^n$. We note that the ℓ^∞-norm of a vector in $\mathbb{R}^n_{max}$ may be infinite. This happens if at least one of its components is equal to ε. The following theorems will be used in analyzing the overall convergence of the max-plus genetic algorithm model for the best solution.

Theorem 2.3. *Let $A \in \mathbb{R}_{max}^{n\times n}$ be a regular (not necessarily square) matrix, then $\|(A \otimes u) - (A \otimes v)\|_\infty \leq \|u - v\|_\infty$ for every $u, v \in \mathbb{R}^n$.*

Proof. We first note that $(A \otimes u), (A \otimes v) \in \mathbb{R}^m$ are finite vectors. Let $\varphi = \|(A \otimes u) - (A \otimes v)\|_\infty$ then there exist an $i_0 \in m$ such that $\varphi = |[(A \otimes u) - (A \otimes v)]_{i_0}|$ where i_0 is the index of the entry with the maximum absolute value in $(A \otimes u) - (A \otimes v)$. Assume that $\varphi = [(A \otimes u) - (A \otimes v)]_{i_0} \geq 0$; then $\varphi = \max_{j\in n}(a_{i_0 j} + u_j) - \max_{l\in n}(a_{i_0 l} + v_l)$ by max-plus matrix multiplication. Hence there exist a $j_0 \in n$ such that $\varphi = (a_{i_0 j_0} + u_{j_0}) - \max_{l\in n}(a_{i_0 l} + v_l) \leq (a_{i_0 j_0} + u_{j_0}) - (a_{i_0 j_0} + v_{j_0}) = u_{j_0} - v_{j_0}$, where $l = j_0$. This implies that $\varphi \leq u_{j_0} - v_{j_0} \leq \max_{j\in n}(u_j - v_j) \leq \max_{j\in n}|u_j - v_j| = \|u - v\|_\infty$. Therefore if $\varphi = [(A \otimes u) - (A \otimes v)]_{i_0} \geq 0$, then $\varphi \leq \|u - v\|_\infty$. The same can be shown if $\varphi = [(A \otimes u) - (A \otimes v)]_{i_0} \leq 0$. $\square$

The inequality $(A \otimes u) - (A \otimes v)\|_\infty \leq \|u - v\|_\infty$ is called the non-expansiveness in the ℓ^∞-norm of the mapping $u \in \mathbb{R}_{max}^n \to A \otimes u \in \mathbb{R}_{max}^m$. Repeated application of the above theorem for a regular square matrix A gives the following: $(A^{\otimes k} \otimes u) - (A^{\otimes} \otimes v)\|_\infty \leq \|u - v\|_\infty$. This means that the ℓ^∞-distance between $A^{\otimes k} \otimes u$ and $A^{\otimes k} \otimes v$ is bounded by $\|u - v\|_\infty$.

Non-expansiveness implies that the cycle-time vector, provided it exists for at least one initial vector, exists for any initial vector and is independent of the specific initial vector. Let $x(k, x(0))$ denote the vector $x(k)$ initiated by $x(0)$, then $x(k, x(0)) = A^{\otimes k} \otimes x(0)$.

Theorem 2.4. *Consider the recurrence relation $x(k+1) = A \otimes x(k)$ for $k \geq 0$, with $A \in \mathbb{R}_{max}^{n \times n}$ a square regular matrix and initial condition $x(0)$. If $x(0) \in \mathbb{R}^n$ is an initial condition such that $\lim_{k\to\infty} \frac{x(k,x(0))}{k}$ exists, then this limit exists and has the same value for any initial condition $y(0) \in \mathbb{R}^n$.*

Proof. Assume that $x(0) \in \mathbb{R}^n$ is such that $\lim_{k\to\infty} \frac{x(k,x(0))}{k} = \varphi$ with $\varphi \in \mathbb{R}^n$. For any $y_0 \in \mathbb{R}^n$ we have,

$$\begin{aligned} 0 &\leq \|\frac{x(k, y(0))}{k} - \frac{x(k, x(0))}{k}\|_\infty \\ &\leq \frac{1}{k}\|(A^{\otimes k} \otimes y(0)) - (A^{\otimes k} \otimes x(0))\|_\infty \\ &\leq \frac{1}{k}\|y(0) - x(0)\|_\infty \end{aligned}$$

Taking the limit as $k \to \infty$ in the above row of inequalities gives

$$\lim_{k\to\infty} \|\frac{x(k, y(0))}{k} - \frac{x(k, x(0))}{k}\|_\infty = 0$$

Therefore as $k \to \infty$ the ℓ^∞-distance between $\frac{x(k,x(0))}{k}$ and $\frac{x(k,y(0))}{k}$ tends to zero, which implies that κ is the cycle time vector for any initial value $y(0)$. □

Chapter 3

Genetic Algorithms

3.1 Introduction

Our goal in this thesis is to reformulate GA using max-plus algebra. In chapter 2, we introduced max-plus algebra and its properties. In this chapter we review GA as a search approach which has emerged to meet the global optimization needs in a complex search space. Genetic Algorithms fall under the broad classification of Evolutionary Algorithms. The basic ideas of GAs were first proposed by John Holland, (Holland, 1975). GAs are inspired by the mechanism of natural selection, a biological process in which stronger individuals are likely to be the winners in a competing environment. It presumes that the potential solution of a problem is an individual and can be represented by a set of parameters regarded as the genes of a chromosome and can be structured by a string of values in binary form. A function known as the fitness function and closely related to the objective function is used to determine the best chromosome for solving the problem. A fitter chromosome has the tendency to yield good-quality offspring. Initially a population pool of chromosomes has to be randomly set. The size of the population depends on the problem at hand, and some guidelines are given in (Mahford, 1994).

In each cycle of genetic operation, called an evolution process, a subsequent generation is created from the chromosome in the current population. This can only be done if a group of parent chromosomes are selected via a specific selection routine. The genes of the parents are to be mixed and recombined for the production of offspring in the next generation. It is expected that from this process of evolution (manipulation of genes), the better chromosome will create a larger number of offspring, and thus has a higher chance of surviving in the subsequent generation, emulating the survival of the fittest mechanism in nature. Algorithms in functional optimization are generally limited to convex regular functions. Many functions are however, non-differentiable, discontinuous, multi-modal and sampling methods have been used to optimize these functions. Whereas traditional search techniques use characteristics of the problem to determine the next sampling point (for example, gradients, Hessians, linearity, and continuity), stochastic search techniques make no such assumptions. Instead, the next sampled points are determined based on stochastic decision rules rather than a set of deterministic decision methods.

Genetic algorithms have been used to solve difficult problems with objective functions that do not possess nice properties such as continuity, differentiability, satisfaction of the Lipschitz Condition, (Goldberg, 1989). These algorithms maintain and manipulate a population of candidate solutions and implement the survival of the fittest strategy in their search for better solutions. This provides an implicit and explicit parallelism allowing for the exploitation of several promising areas of the solution or search space at the same time. The implicit parallelism is due to the schema theory developed by John Holland whereas the explicit one is from the manipulation of a population of points. GAs have also been shown to solve linear and non-linear problems by exploring all regions of the state space and exploiting promising areas through crossover, mutation, and selection operations applied to individuals in the population, (Michalewich, 1994). They require the determination of six fundamental issues namely: chromosome representation, the evaluation function, selection function, the genetic operators making up the reproduction function, the creation of the initial population and the stopping criteria. We describe each of these issues in the following sections.

3.1.1 Initialization and Chromosome Representation

The GA must be provided with an initial population. The population is generated randomly, covering the entire range of the search space (possible solutions). Since GAs can iteratively improve existing solutions, the initial population can be seeded with good solutions (i.e. solutions from other heuristics). A chromosome representation is needed to describe each individual in the population of interest. Each chromosome or individual is made up of a sequence of genes from a certain alphabet. An alphabet could consist of binary digits (0 and 1), integers, symbols (like A,B,C,D), matrices, etc. However, it has been shown that more natural representations are more efficient and produce better solutions (Michalewich, 1994). One useful representation of an individual or chromosome for function optimization involves genes or variables from an alphabet of floating point numbers with values within the variables upper and lower bounds.

3.1.2 Selection Mechanism

The selection operation involves the selection of good individuals and at the same time eliminates bad individuals from the population based on the evaluation of individual fitness. A probabilistic selection is performed based on the individuals fitness such that the better ones have an increased chance of being selected. There are several selection processes namely: Roulette wheel , Tournament, Elitist models, Scaling techniques and Ranking methods (Michalewich, 1994). The first selection method, roulette wheel, was developed by Holland in 1975. An example of a selection function is the probability function

$$p_s(x_i) = \frac{f(x_i)}{\sum_{i=1}^{k} f(x_k)}$$

on the population $\{x_1, ..., x_n\}$, where $f(x_i)$ equals the fitness of individual x_i. Ranking methods assign $p(x_i)$ based on the rank of solution x_i when all solutions are sorted. Normalized geometric ranking defines $p(x_i)$ for each individual by the following (Houck and Jones, 1994); P(selecting the i^{th} individual) is $q'(1-q)^{r-1}$ where q is the probability of selecting the best individual, r is the rank of the individual, where 1 is the best, p is the population size and q' is $\frac{q}{1-(1-q)^p}$

Tournament selection works by selecting j individuals randomly with replacement from the population and inserts the best of the j into the new population. The procedure is repeated until the required number of individuals have been selected.

3.1.3 Genetic Operators

Genetic operators are used to create new solutions based on the existing solutions in the population. There are two basic types of operators: crossover and mutation. The crossover is the most significant operation in the genetic search strategy. It determines the major behavior of the optimization process. There are several crossover schemes used such as one-point, two-point, multi-point. However, as a common criteria, any crossover scheme should ensure that the proper genes of good individuals be inherited by the new individuals of the next generation. A big crossover probability may improve genetic algorithms capability to search for new solution space.

On the other hand mutation's main function is to prevent losing single important gene segment to maintain the variety of solution population. Relatively small mutation probabilities are used. Inversion, for example is a special form of mutation. It is designed to carry out reordering operation and improve the local search ability for genetic algorithm. The application of these two basic types of operators and their derivatives depends on the chromosome representation used.

To illustrate this let X and Y be two binary m-dimensional row vectors denoting chromosomes (parents) from the population. We consider binary mutation and simple crossover for X and Y as follows:

$$x_i^{'} = \begin{cases} 1 - x_i, & \text{if } U(0,1) < p_m \\ x_i, & \text{otherwise} \end{cases}$$

We note that binary mutation flips each bit in every individual in the population with probability p_m according to the above equation. Simple crossover also generates a random number r from a uniform distribution from 1 to m, where m is a fixed positive integer, and creates two new individuals according to the following equations:

$$x_i^{'} = \begin{cases} x_i, & \text{if } i < r \\ y_i, & \text{otherwise} \end{cases} \tag{3.1}$$

$$y_i^{'} = \begin{cases} y_i, & \text{if } i < r \\ x_i, & \text{otherwise} \end{cases} \tag{3.2}$$

For X and Y real, the following operators are defined: uniform mutation, non-uniform mutation, multi-uniform mutation, boundary mutation, simple crossover, arithmetic crossover, and heuristic crossover. We discuss briefly the each of the above operators. We consider a_i and b_i as the lower and upper bounds respectively of the variable i. Uniform mutation selects randomly one variable, j, and sets it equal to a uniform random number $U(a_i, b_i)$ as follows:

$$x_i' = \begin{cases} U(a_i, b_i), & \text{if } i = j \\ x_i, & \text{otherwise} \end{cases}$$

In boundary mutation we randomly selects one variable, j, and sets it equal to either its lower or upper bound, where $r = U(0, 1)$, that is :

$$x_i' = \begin{cases} a_i, & \text{if } i = j, r < 0.5 \\ b_i, & \text{if } i \geq j, r < 0.5 \\ x_i, & \text{otherwise} \end{cases}$$

Non-uniform mutation randomly selects one variable, j and sets it equal to a non-uniform random number as follows:

$$x_i' = \begin{cases} x_i + (b_i - x_i)f(G), & \text{if } r_1 < 0.5 \\ x_i - (x_i + a_i)f(G), & \text{if } r_1 \geq 0.5 \\ x_i, & \text{otherwise} \end{cases}$$

$$\begin{aligned} \text{where} \quad f(G) &= (r_2(1 - \frac{G}{G-max}))^b \\ r_1, r_2 &= \text{a uniform random number between } (0, 1) \end{aligned}$$

$$G = \text{the current generation}$$

$$G_{max} = \text{the maximum number of generations}$$

$$b = \text{a shape parameter}$$

The multi-non-uniform operator applies the non-uniform operator to all of the variables in the parent X. Real-valued simple crossover is identical to the binary version presented above in equations (3.1) and (3.2). Arithmetic crossover produces two complimentary linear combinations of the parents, where $r = U(0,1)$ as follows:

$$X' = rX + (1-r)Y$$

$$Y' = (1-r)X + rY$$

Heuristic crossover produces a linear extrapolation of the two individuals. This is the only operator that makes use of the fitness function information. A new individual X' is created using the equation:

$$X' = X + r(X - Y) \tag{3.3}$$

$$Y' = X$$

where $r = U(0,1)$ and X is better than Y in terms of fitness. We define feasibility as follows:

$$\text{feasibility} = \begin{cases} 1, & \text{if } x_i' \geq a_i, x_i' \leq b_i \\ 0, & \text{otherwise} \end{cases}$$

If X' is infeasible, then generate a new random number r and create a new solution using equation (3.3) and stop otherwise.

3.2 Schema

A schema is a structural unit that represents a concept, situation, event, behavior and so on, in a generalized form that is, it contains an abstract representation of multiple instances of the same kind. In schema theory this unit is an internal data structure in the memory that organizes an individual's similar experiences. It is used to recognize similar and discriminate dissimilar new experiences, access the essential elements of the commonality, draw inferences, create goals and develop plans. For example, the schema of a CAR is a generalization of all the cars a particular individual has seen or experienced before. Though it does not contain all the details of any car seen, it contains the essentials, the core features and properties shared by almost all the cars. If an individual sees an object that shares the same elements and relations as stored in his or her CAR schema, the individual will recognize it as a car.

3.2.1 Schema in GA

The use of the schema theory was one of the earliest attempts to understand how genetic algorithm works. In schema theory, the search space is partitioned into subspaces of varying levels of generality and mathematical models are constructed which estimate how the individuals in the population belonging to a schema can be expected to grow in the next generation. The theory gave rise to the building block hypothesis which attempted to explain how a GA solves a problem by revealing that near optimal solutions were forged from small, low-order, better-than average schemata. A schema is a template made up of a string of $1s$, $0s$ and $*s$, where $*$ is used as a wild card that can be either 1 or 0. For example $H = 1**0*0$ is a schema which has 8 instances one of which is 101010. The number of non $*$, or defined bits in a schema, is called its order and is denoted by $o(H)$. In the example, H has order 3. In the classical sense, we consider schema to be collections of strings of a special kind. A schema is represented by an element of $\{0, 1, *\}^l$ when we have binary strings of fixed length l. The relationship between the schema s and the binary strings is given as $s_1...s_l \longleftrightarrow \{d_1...d_l : \forall j.s_j \neq * \Longrightarrow d_j = s_j\}$.

We let $\pi_j : \{0,1\}^l \to \{0,1\}$ be the projection to the jth coordinate, that is, $\pi_j(d_1...d_l) = d_j$, then we define schema s as:

$$s_j = \begin{cases} *, \text{ if } \pi_j(H) = \{0,1\} \\ 1, \text{ if } \pi_j(H) = \{1\} \\ 0, \text{ if } \pi_j(H) = \{0\} \end{cases}$$

for any non-empty collection H of strings (Vose, 1993). To illustrate this, we see that if $H = \{001011, 011111\}$, then every member of $0{*}1{*}11$ can be generated, including the strings 011011 and 001111 which are originally contained in H. Crossover is usually regarded as the central exploratory mechanism of the genetic algorithm and schema characterize the subsets which crossover explores. Crossover also has the ability of recombining schema. For example crossover may produce 001111 from the parents 011111 and 001011. This is an example of (members of) the schemas $***11*$ and $00****$ being recombined to form (a member of) the schema $00*11*$. Thought of in this way, crossover can assemble the building blocks represented by $***11*$ and $00****$ into something new, the composite $00*11*$. Given that exploration and recombination of schema are definitive properties of genetic algorithms, perhaps a theorem concerning schema could address these issues and relate them to the direction of genetic search. We state and discuss the Holland schema theorem.

Theorem 3.1. $E(m(H,t+1)) \geq \frac{\hat{u}(H,t)m(H,t)}{\bar{f}(t)}(1 - p_c\frac{d(H)}{l-1})((1-p_m)^{o}H)$

In the discussion that follows we use the term 'schema' to refer both to the template and to the set of instances it defines within a population. Now let us look at the expected number of instances of schema H as we iterate the Standard Genetic Algorithm. Let $m(H,t)$ be the number of instances of H at time t. Let $f(x)$ represent the fitness of chromosomes x, and $\bar{f}$ represent the average fitness at time t, or $\bar{f}(t) = \frac{\sum_{x \in S} f(x)}{n}$ where $n = |S|$. Let $\hat{u}(H,t)$ represent the average fitness of instances of H at time t, or

$$\hat{u}(H,t) = \frac{\sum_{x \in H} f(x)}{m(H,t)}.$$

If we completely ignore the effects of crossover and mutation, then we get the expected value $E(m(H,t+1)) = n\frac{\sum_{x \in H} f(x)}{\sum_{x \in S} f(x)} = \frac{\sum_{x \in H} f(x)}{\bar{f}(t)} = \frac{\hat{u}(H,t)m(H,t)}{\bar{f}(t)}$. We consider only the effects of crossover and mutation that lower the number of instances of H in the population. Then there will be a good lower bound on $E(m(H,t+1))$. Let $S_c(H)$ be the probability that a random crossover bit is between the defining bits of H and let p_c be the probability of crossover occurring. Then $S_c(H) = 1 - p_c(\frac{d(H)}{l-1})$. Let $S_m(H)$ be the probability of an instance of H remaining the same after mutation, then $S_m(H)$ is dependent on the order of H. If the probability of mutation is p_m, then $S_m(H) = (1 - p_m)^{\circ}(H)$. Substituting this in the above equation justifies the above theorem by John Holland. This theorem is often interpreted to mean that if $\hat{u} > \bar{f}(t)$, then there will be exponentially more instances of schema with low defining length and low order. The above theorem somehow justifies that all genetic algorithms follow a certain pattern called the building block hypothesis (Herrmann, 2011). In this pattern, schema with low order and low defining length are optimized. Crossover is used to turn the schema into higher order and higher defining length schema. The schema theorem provides a good advice on how to assure the growth of good sub-solutions and to sustain the growth to takeover the population.

3.2.2 Termination

Natural selection uses diversity in a population to produce adaptation. If there is no diversity then there is nothing for natural selection to work on. Since GAs imitate natural selection, we apply the same principles and use a measure of diversity to determine convergence. The following are some of the stopping criteria in GAs:

1. A common terminating criterion is a specified number of generations.

2. Another stopping strategy is the population convergence criteria. Genetic algorithms in general will force much of the entire population to converge to a single population.

3. When the sum of the deviations among the individuals becomes smaller than some specified threshold, the algorithm can be terminated.

4. The algorithm can also be terminated due to lack of improvement in the best solution over a specified number of generations.

5. Sometimes manual inspection helps us to end the process.

Some of the above conditions can be used in conjunction with each other to terminate the algorithm.

Chapter 4

A Max-plus Model for Genetic Algorithms

4.1 Introduction

We consider an optimization problem of the form $\max f(x)$ subject to a search space Ω, where $f(x)$ is a nonlinear function for which the usual gradient methods are not suitable. This type of optimization problem becomes a candidate for solution by GA. In this chapter, we develop the max-plus algebra model as a novel mathematical framework which formulates and explains the many concepts and related properties (operators) in Genetic Algorithms.

The model represents key attributes and characteristics of Genetic Algorithms which presents information about the component of the algorithm by breaking it down to its usable parts for further analysis since Genetic Algorithms do not possess strong mathematical formulations. This max-plus formulation for genetic algorithms has been developed for the following reasons:

(i) to provide a mathematical framework and a model for the population dynamics of the Genetic Algorithm processes,

(ii) to address some of the disadvantages of Genetic algorithms using our method, and

(iii) to predict or simulate what a real-world system will do in future since it is expensive, impractical and sometimes impossible to experiment directly with the system.

4.1.1 Max-plus Search Space and Fitness

We consider $\Omega = \{x_1, x_2, ..., x_n\}$ as a finite set of possible solutions of the GA and let Ω be known as the search space or solution space. Then each $x_i \in \Omega$, $1 \leq i \leq n$ is a chromosome or an individual. We claim that the GA search space is a commutative idempotent semi-ring. To see this, we note that $\oplus$ is associative in the search space, that is for all chromosomes $x_1, x_2, x_3 \in \Omega$, $x_1 \oplus (x_2 \oplus x_3) = (x_1 \oplus x_2) \oplus x_3$ and commutative with zero element ε, thus $x_1 \oplus \varepsilon = \varepsilon \oplus x_1 = x_1$. Again we see that $\otimes$ is associative, distributive over $\oplus$ and has unit element e since for all elements $x_1, x_2, x_3 \in \Omega$, $x_1 \otimes (x_2 \oplus x_3) = (x_1 \otimes x_2) \oplus (x_1 \otimes x_3)$ and $x_1 \otimes e = ex_1 \otimes x_1 = x_1$. Next, ε is absorbing for $\otimes$ since $x_1 \otimes \varepsilon = \varepsilon \otimes x_1 = \varepsilon$ for all $x_1 \in \Omega$. We also know that $\otimes$ is commutative for all chromosomes $x_1, x_2 \in \Omega$ since $x_1 \otimes x_2 = x_2 \otimes x_1$. Finally, $\oplus$ is idempotent, and so for all $x_1 \in \Omega$, $x_1 \oplus x_1 = x_1$. Hence the GA search space is a max-plus commutative idempotent semi-ring. The number of population members in each generation is represented by the vector of chromosomes $(x_1, x_2, ..., x_n)^T$.

Again, we consider the function given by $f : \Omega \to \mathbb{R}_{max}$ defined on some domain, where $\mathbb{R}_{max}$ is the max-plus algebra, and the domain is problem specific. Then the goal of the above function f is to evaluate the maximum functional value determined by a chromosome in the search space, that is $x_i \in \Omega$ such that $f(x_i) \to f_{max}(x_i)$ for all $1 \leq i \leq n$. The function f which calculates the maximum value of the solution space might be called Max-plus fitness function. The individual chromosome $x_i \in \Omega$ which gives the maximum fitness is the fittest chromosome to survive in the next generation and is denoted by x_{max}. We define fitness in max-plus as the sum of the functional values of all chromosomes in the search space Ω, that is:

$$\begin{aligned} f(x_1) \oplus f(x_2) \oplus \ldots \oplus f(x_n) &= \max(f(x_1), f(x_2), \ldots, f(x_n)) \\ &= \max_{i \in n} f(x_i) \\ &= arg \max_{i \in n} f(x_i) \end{aligned}$$

Thus given a search space $\Omega = \{x_1, x_2, \ldots, x_n\}$ and a function $f : \Omega \to \mathbb{R}_{max}$, we have $f_{max} = f(x_{max}) = arg \max_{i \in n} f(x_i)$ where x_{max} is the fittest chromosome and f_{max} is the maximum fitness.

4.1.2 Max-plus GA steps

We consider the initial population of chromosomes which are selected randomly based on the requirements imposed on the solution. Let the initial population be denoted by P, then

$$\mathbf{P} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

where $x_i \in \Omega$ is selected randomly. We note that for any GA, a chromosome representation is needed to describe each individual in the population of interest.

Evaluation Process

Individual chromosomal strengths of the population are determined by evaluating their fitness function values, $f(x_i)$. That is:

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \longrightarrow \begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{bmatrix}$$

where the fittest individual is determined by the highest $f(x_i)$ value. The above procedure may be explained in terms of max-plus to determine the fittest individual as follows:

$$\begin{aligned} f(x_1) \oplus f(x_2) \oplus ... \oplus f(x_n) &= \max(f(x_1), f(x_2), ..., f(x_n)) \\ &= \max_{i \in n} f(x_i) \end{aligned}$$

and $x_i = x_{max} \in \Omega$ is the fittest chromosome to survive in the next generation.

Selection Process

A probabilistic selection is performed based on the the fitness of the individual such that the better chromosomes have more advantage of being selected.

There are several selection schemes as indicated, (Kolokoltsov and Maslov, 1988). Roulette wheel for example was the first selection scheme developed by Holland in 1975, (De Schutter and van den Boom, 2001). The probability of each individual P_i is given by $P(\text{Chromosome } i \text{ is chosen}) = \frac{f(x_j)}{\Sigma_i^n f(x_i)}$ where $f(x_i)$ is fitness of chromosome i. The distribution of a randomly chosen chromosome (in population) after selection is given by $x(n) = \frac{\sigma(f(x_j)x_j(n)}{\Sigma_i^n \sigma(f(x_i))x_i(n)}$ where σ is the selection function. A new population is obtained where $y_i = x_j$, that is:

$$\mathrm{P}' = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

All of the chromosomes of P' are those of P and the expectation of the number of occurrences of any chromosome of P in P' is proportional to the number of occurrences of that chromosome in P times the chromosomes fitness value, that is $E(x_i) = x_i f(x_i)$. This is designed to imitate the principle of the survival of the fittest developed by Darwin.

Reproduction Process

The chromosomes of P' are grouped for mating according to some probabilistic rule and transformation $\mathcal{T}$. For instance the groups could be

$$\mathbf{Q}_1 = \begin{bmatrix} y_{i_1^1} \\ y_{i_2^1} \\ \vdots \\ y_{i_{q_1}^1} \end{bmatrix} \quad \mathbf{Q}_2 = \begin{bmatrix} y_{i_1^2} \\ y_{i_2^2} \\ \vdots \\ y_{i_{q_2}^2} \end{bmatrix} \dots \quad \mathbf{Q}_j = \begin{bmatrix} y_{i_1^j} \\ y_{i_2^j} \\ \vdots \\ y_{i_{q_j}^j} \end{bmatrix} \dots$$

under the crossover transformation process

$$\mathbf{Q}^{\mathcal{T}} = \begin{bmatrix} \mathcal{T}_1(y_{i_1^j}, y_{i_2^j}, \dots, y_{i_{q_j}^j}) \\ \mathcal{T}_2(y_{i_1^j}, y_{i_2^j}, \dots, y_{i_{q_j}^j}) \\ \vdots \\ \mathcal{T}_{q_j}(y_{i_1^j}, y_{i_2^j}, \dots, y_{i_{q_j}^j}) \end{bmatrix}.$$

This results in the next generation population of chromosomes

$$\mathbf{P}'' = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix}$$

We mutate P''. Mutation serves as a supporting operator for restoring lost genetic traits. Thus, with very small probability we replace z_i with w_i for some randomly chosen element and this gives a new population

$$P''' = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}$$

If we are not satisfied with the solution, we start all over with P''' as the initial population and the cycle is repeated a certain number of times depending on the stopping conditions imposed on the problem.

4.1.3 Max-plus Model For Population Dynamics in Genetic Algorithms

We know that in each generation, the population is a vector of chromosomes, thus

$$\mathbf{x}^{(k)} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix}$$

where $x_i^{(k)}$ is the number of chromosomes at generation k.

The probability that a member of the k^{th} population will survive to become a member of the $(k+1)^{th}$ population is given by P_k, where $0 \leq P_k \leq 1$, $k = 1, 2, ..., n-1$. Let b_k be the reproduction rate, then $b_k \geq 0$ for $k = 1, 2, ..., n$.

Then these numbers can be written in matrix form as follows:

$$T = \begin{bmatrix} b_1 & b_2 & b_3 & \cdots & b_{n-1} & b_n \\ P_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & P_2 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & P_{n-1} & 0 \end{bmatrix}$$

where T is called the projection matrix. Multiplying the above matrix T by the population vector $\mathbf{x}^{(k)}$ produces the population vector for the next generation. This procedure is represented as a recursive max-plus equation of the form

$$\mathbf{x}^{(k+1)} = T \otimes \mathbf{x}^{(k)} \tag{4.1}$$

where $\mathbf{x}^{(k+1)}$ is the population vector in the next generation, that is:

$$\mathbf{x}^{(k+1)} = \begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ \vdots \\ x_n^{(k+1)} \end{bmatrix} \tag{4.2}$$

and

$$\mathbf{x}^{(k)} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix}. \tag{4.3}$$

is the population vector with element $x_1^{(k)}, x_2^{(k)}, \ldots$ representing the current number of individuals. We present the above equation in Max-plus matrix representation which generate a linear system of equations as follows:

$$\begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ \vdots \\ x_n^{(k+1)} \end{bmatrix} = \begin{bmatrix} b_1 & b_2 & b_3 & \cdots & b_{n-1} & b_n \\ P_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & P_2 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & P_{n-1} & 0 \end{bmatrix} \otimes \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix}$$

which ends up in the following linear systems of equations

$$\begin{aligned} x_1^{(k+1)} &= (b_1 \otimes x_1^{(k)}) \oplus (b_2 \otimes x_2^{(k)}) \oplus \ldots \oplus (b_n \otimes x_n^{(k)}) \\ &= max[(b_1 + x_1^{(k)}), (b_2 + x_2^{(k)}), \ldots, (b_n + x_n^{(k)})] \\ x_2^{(k+1)} &= (P_1 \otimes x_1^{(k)}) \oplus (0 \otimes x_2^{(k)}) \oplus \ldots \oplus (0 \otimes x_n^{(k)}) \\ &= max[(P_1 + x_1^{(k)}), (0 + x_2^{(k)}), \ldots, (0 + x_n^{(k)})] \end{aligned}$$

$$\vdots$$

$$x_n^{(k+1)}$$

$$= (0 \otimes x_1^{(k)}) \oplus (0 \otimes x_2^{(k)}) \oplus ... \oplus (P_{n-1} \otimes x_{n-1}^{(k)}) \oplus (0 \otimes x_n^{(k)})$$

$$= max[x_1^{(k)}, x_2^{(k)}, ..., P_{n-1} + x_{n-1}^{(k)}, x_n^{(k)}]$$

We iterate to find the population distribution at any generation k as follows:

$$\begin{aligned}
\mathbf{x}^{(k+1)} &= T \otimes \mathbf{x}^{(k)} \\
&= T \otimes (T^{\otimes(k)} \otimes \mathbf{x}^{(0)}) \\
&= (T \otimes T^{\otimes(k)}) \otimes \mathbf{x}^{(0)} \\
&= \overbrace{T \otimes T \otimes ... \otimes T}^{k \ \ times} \otimes \mathbf{x}^{(0)} \\
&= T^{\otimes k} \otimes \mathbf{x}^{(0)}
\end{aligned}$$

We note that the numerical evaluation of $T^{\otimes k}$ is $k \times T$ in conventional algebra. Thus, if we know the initial generation vector

$$\mathbf{x}^{(0)} = \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ \vdots \\ x_n^{(0)} \end{bmatrix} \tag{4.4}$$

We can determine the next vector by multiplying $\mathbf{x}^{(0)}$ by an appropriate power of the projection matrix.

Alternatively equation (4.1) can be interpreted in the max-plus algebra as follows:

$$\begin{aligned} \mathbf{x}^{(n+1)} &= T \otimes \mathbf{x}^{(n)} \\ &= (T_{i1} \otimes x_1^{(n)}) \oplus ... \oplus (T_{in} \otimes x_n^{(n)}) \\ &= \bigoplus_{j=1}^{k} (T_{ij} \otimes x_j^{(n)}) \\ &= \max_{1 \leq j \leq k} (T_{ij} + x_j^{(n)}) \end{aligned}$$

The subsequent generations will be determined by the max-plus multiplication of the projection matrix T by an initial matrix from the search space related to the population dynamics of the problem. Hence $\mathrm{x}^{(n+1)} = \max_{1 \leq j \leq k}(T_{ij} + x_j^{(n)})$ is the fittest population to begin the next cycle.

4.1.4 Eigenvalues And Eigenvectors On Population

In this section we consider the notion of eigenvalues and eigenvectors in max-plus on the GA population. Linear systems have modes which are reached asymptotically when systems are stable in the conventional algebra and these modes are related to their eigenstructures. Similar notions exist for systems obeying the max-plus equation of the form $A \otimes \mathbf{x} = \lambda \otimes \mathbf{x}$. Let A be $n \times n$

matrix such that

$$A \otimes \mathbf{x} = \lambda \otimes \mathbf{x}$$

where λ is a scaler and $\mathbf{x}$ is an $n-$vector whose i^{th} element equals $\lambda \otimes \mathbf{x}_i$, which is equal to $\lambda + \mathbf{x}_i$, then λ is called the eigenvalue and $\mathbf{x}$ is the corresponding eigenvector of the matrix A. Starting from a specified initial population $\mathbf{x}^{(0)}$ as an eigenvector of A corresponding to the eigenvalue λ, and repeatedly applying equation (4.1) we see that

$$\mathbf{x}^{(1)} = A \otimes \mathbf{x}^{(0)} = \lambda \otimes \mathbf{x}^{(0)}$$

$$\mathbf{x}^{(2)} = A \otimes \mathbf{x}^{(1)} = A \otimes (\lambda \otimes \mathbf{x}^{(0)}) = \lambda^{\otimes 2} \otimes \mathbf{x}^{(0)}$$

and in general we have

$$\mathbf{x}^{(n)} = \lambda^{\otimes n} \otimes \mathbf{x}^{(0)}, \quad n = 0, 1, 2, ...$$

Since the population of chromosomes have the characteristics of a stable growth pattern that is, one which the chromosomal numbers remain the same for the generation process, the $(n+1)^{th}$ population vector must be a scalar multiple of the n^{th} population vector. Thus, equation (4.5) can be represented as

$$\mathbf{x}^{(n+1)} = T \otimes \mathbf{x}^{(n)} = \lambda \otimes \mathbf{x}^{(n)}$$

where λ is a scalar and hence

$$T \otimes \mathrm{x}^{(n)} = \lambda \otimes \mathrm{x}^{(n)} \tag{4.5}$$

The above max-plus equation (4.5) shows that we can obtain a population growth pattern in which the chromosome numbers remain constant in the next generation of the genetic algorithm.

4.1.5 Convergence of the Max-plus GA

To establish convergence, we expect that the algorithm tends to an optimum. We note in general that the search space in GA is simply a set without any norm or distance measure. Therefore we cannot anticipate convergence criteria saying that x^{n+1} tends to a limiting value (optimum) as n tends to infinity. Instead, we require that the best solution would be a measure of diversity of the chromosomes. To see this, we look at the max-plus asymptotic behavior of $\mathrm{x}^{(k)}$ quantitatively.

We consider $\{\mathrm{x}^{(k)} : k \in \mathbb{N}\}$ as a sequence in the max-plus algebra and assume that for all $j \in n$ the quantity η_j, defined by

$$\eta_j = \lim_{k\to\infty} \frac{x_j^{(k)}}{k} \tag{4.6}$$

exists, where

$$x_i^{(k+1)} = \max_{1\leq i\leq n}(A_{ij} + x_j^{(k)}), \;\; \forall 1 \leq i \leq n \tag{4.7}$$

Then the vector $\eta = (\eta_1, \eta_2, ..., \eta_n)^T$ is called the cycle time vector of the sequence $\mathbf{x}^{(k)}$. If all $\eta_j's$ have the same value, this value is called the asymptotic growth rate of the sequence $\mathbf{x}^{(k)}$. In this case the sum of deviations among individuals become smaller and smaller and there is lack of improvement in the best solution over a specified number of generations determined by the cycle time vector. We apply the max-plus asymptotic growth rate of the sequence $\mathbf{x}^{(n)}$ to terminate the genetic algorithm. We note that the above equation (4.7) is nothing but a linear system in the max-plus semi-ring $\mathbf{x}^{(n+1)} = A \otimes \mathbf{x}^{(n)}$. We apply the population convergence criteria of genetic algorithms and use a measure of diversity in the current population to predict the solution to the problem. We find the asymptotic growth rate of the sequence $\mathbf{x}^{(n)}$ of the population to determine the population diversity.

4.1.6 The Max-plus GA Complete Model

We present the complete max-plus GA formulation and indicate the interaction between the projection matrix and the generation vectors as shown below. The fitness function is evaluated by the max-plus equation

$$\begin{aligned} f(x_1) \oplus f(x_2) \oplus ... \oplus f(x_n) &= \max(f(x_1), f(x_2), ..., f(x_n)) \\ &= \max_{i \in n} f(x_i) \\ &= arg \max_{i \in n} f(x_i) \;\; \text{where} \;\; x_i \in \Omega \end{aligned}$$

The population dynamics for the generation is given by the max-plus equations

$$\begin{aligned} \mathbf{x}^{(k+1)} &= T \otimes \mathbf{x}^{(k)} \\ &= T \otimes (T^{\otimes(k)} \otimes \mathbf{x}^{(0)}) \\ &= (T \otimes T^{\otimes(k)}) \otimes \mathbf{x}^{(0)} \\ &= \overbrace{T \otimes T \otimes ... \otimes T}^{k \;\; times} \otimes \mathbf{x}^{(0)} \\ &= T^{\otimes k} \otimes \mathbf{x}^{(0)} \end{aligned}$$

We now consider that for a given $\mathbf{x}^{(0)}$, the sequence $\mathbf{x}^{(k)}$, $k = 1, 2, ...$ generated by the max-plus equation $\mathbf{x}^{(k+1)} = T \otimes \mathbf{x}^{(k)}$, $k \geq 0$ yield a sequence of increasingly better solutions indicated by the iterates $\mathbf{x}^{(k)}$ with the projection matrix. This is shown by the expression $\max_{1 \leq j \leq k}(T_{ij} + x_j(n))$ below, which shows the maximum of the interaction between the projection matrix and the current population vector and in max-plus, this is the fittest population to begin the next generation.

$$\begin{aligned}
\mathbf{x}^{(n+1)} &= T \otimes \mathbf{x}^{(n)} \\
&= (T_{i1} \otimes x_1^{(n)}) \oplus \ldots \oplus (T_{in} \otimes x_n^{(n)}) \\
&= \bigoplus_{j=1}^{k} (T_{ij} \otimes x_j^{(n)}) \\
&= \max_{1 \le j \le k} (T_{ij} + x_j^{(n)})
\end{aligned}$$

The equation below actually plays a larger role in max-plus algebra representation of convergence of the GA that is,

$$T \otimes \mathbf{x}^{(n)} = \lambda \otimes \mathbf{x}^{(n)} \text{ and}$$

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

where $\eta_j' s$ are the cycle-time vectors of the sequence $\mathbf{x}^{(k)}$ and

$$x_i^{(k)} = \max_{1 \le i \le n} (A_{ij} + x_j^{(k-1)}), \forall 1 \le i \le n$$

4.1.7 Why Max-plus Genetic Algorithms

In this subsection, we discuss what our max-plus genetic algorithm formulation has achieved. We state some of the disadvantages of the genetic algorithms and show how our method addresses them. Previous analysis of GA models used parameters which were specific to the type of problem. Many genetic algorithms do not fit this frameworks. Our formulation uses mathematical equations from max-plus algebra to present a more general model to genetic algorithms and to the best of our knowledge, this is the first mathematical framework in genetic algorithms.

One of the disadvantages of GAs is that the fitness function is crucial to the development of a suitable solution for the problem. The fitness function ensures the fittest member to begin the next generation so if poorly written, the GA may end up solving different problem entirely from the originally intended one or the solution will be far from the truth. Our max-plus algebra formulation ensures that maximum fitness function is obtained by summing the functional values of all chromosomes in the search space Ω, that is $\max_{i \in n} f(x_i)$ for which $x_i = x_{\max}$ is the fittest chromosomes. Hence from equation (4.1) we note that $\max_{1 \le j \le k}(T_{ij} + x_j^{(n)})$ is the fittest population to begin the next cycle. Thus the max-plus formulation solves the original intended problem.

Again in real life, many of the equations that are encountered with respect to the description of fitness functions in GAs are nonlinear in conventional algebra but become linear in the max-plus algebra. To illustrate the usefulness of the max-plus in a simple example, we consider the nonlinear iterative system given by $x_i^{(k+1)} = \max_{j\in n}\{a_{ij} + x_j^{(k)}\}$. We express this system in max-plus algebra as $x_i^{(k+1)} = \bigoplus_{j=1}^{n}(a_{ij} \otimes x_j^{(k)}), i = 1, 2, ..., n$ where $\bigoplus_{j=1}^{n}(a_{ij} \otimes x_j^{(k)}) = (a_{i1} \otimes x_1) \oplus (a_{i2} \otimes x_2) \oplus ... \oplus (a_{in} \otimes x_n)$, which is max-plus linear. If we let a_{ij} be elements of the matrix A such that $\mathbf{x}^{(k)} = A \otimes \mathbf{x}^{(k-1)}$, then from equation (4.1) we see that $\mathbf{x}^{(k)} = A^{\otimes k} \otimes \mathbf{x}^{(0)}$ and this is an example of a conventional nonlinear system which is linear in the max-plus sense.

We see that Genetic Algorithms are based on Darwin's theory of evolutions which mimics the basic natural evolution cycle, that is, natural selection or the survival of the fittest reproduction. The evolution processes result in a struggle for existence for which some combinations will survive while others perish. This is a clear example of trial and error method. We note that this trial and error method is not desirable due to the fact that money and time are wasted in the execution of this method. Since GAs do not possess any strong mathematical framework we think that our formulation is just in time to remedy this situation. Our max-plus algebra formulation which consist of mathematical equations is used to describe the behavior of GAs. The model gives a real understanding of the effects of parameter changes on the properties of the GAs and can be used for analysis, optimization, control and to predict the behavior of the GAs in future works.

Another issue with GAs is setting the correct population size. If the population is too low, the investigation may cover too little of the search space to find the optimum solution. Our model was able to withstand large populations so that the optimum solution would not be trapped in a local optimum. This is shown by the fact that there are no restriction on number of chromosomes in the population.

Another disadvantage is that a large number of parameters need to be adjusted, for example the kind of selection and crossover operator to use, the population size, the probabilities of applying a certain operator and the form of the fitness function. This fact results in a lengthy trial-and-error procedure whose purpose is to adjust the parameters of the GA and thus makes the method more complicated. To solve this our formulation makes use of the stable growth max-plus equation $\lambda^{(k+1)} = \lambda \otimes x^{(k)}$ which normalizes the GA system and makes it stable with constant population.

Chapter 5

Conclusion, Open Problems and Extensions

5.1 Conclusion

In this thesis, we have presented an overview of the max-plus algebra. We have shown that the algebra is an analogue of linear algebra developed on the pair of operations $(\oplus, \otimes)$. We have presented the use of the max-plus algebra in solution of systems of equations and inequalities, optimal control problems and some discrete event systems. In particular, we have shown how to find eigenvalues of a matrix using the max-plus discrepancy matrix and other related max-plus techniques which helped us to obtain important information on the dynamics of the genetic algorithm population. We have defined max-plus fitness function which determines the fittest chromosome in the search space to begin the next generation. We have proved that the search space is a commutative idempotent semi-ring since for all chromosomes $x_i \in \Omega$ there is associativity for $\oplus$ and $\otimes$ is distributive over $\oplus$; has unit and absorbing elements in the search space. We note that the possibility of working in a linear-like manner is based on the fact that the max-plus algebra is an idempotent semi-ring.

We have presented some of the disadvantages of genetic algorithms and shown how our formulation can help solve them. We are able to show by the max-plus algebra analysis that our iterate will yield a sequence of better solutions for the next generation each time. We have shown that the population generation dynamics of GAs can be modeled using a max-plus linear recursive equation and this is guaranteed by equation (4.1). We showed that by applying repeatedly the recursive equation yields a sequence of better solutions for the next generation at any time. This novel max-plus model explains a growth pattern in which the chromosomal numbers remain constant after obtaining the updated population for the next generation. By formulating the GA in this way we able to use the model to explain the concepts better and it turns out to be much more general and easier than the previously known methods. These results indicate a promising direction for further research into other operators in Genetic Algorithms and we recommend that the formulation be extend to other variants of evolutionary computations.

5.2 Open Problems and Extensions

In the previous sections, we discussed the formulation of GAs using concepts in max-plus algebra. In this section we present some of the issues that are related to our work and are worth investigating.

Multi-Objective Optimization Problems (MOP)

This thesis focused on a single fitness function as a single objective function subject to a search space. Quite recently, researchers have used GAs to solve MOP, where the objective function is a set of functions. We think that the analysis of our formulation will allow for the combination of the multiple objectives to form a scalar objective function through a max-plus linear combination. This method seem promising for formulating a GA that solves MOP in terms of a max-plus algebra model.

Formulation of Nonlinear Problems

Our max-plus method offers a clear understanding of polynomial formulation and its applications to discrete event models as indicated in chapter 1. We showed that the max-plus operations $\oplus$ and $\otimes$ on exponential functions are induced by the conventional algebra. Indeed this concept is the basis of the generalization of Cramer's rule and the Cayley-Hamilton's theorem in chapter 1. We expect that the max-plus concept can be extended to generalize problems involving rational functions and even trigonometric functions since we can rewrite these functions in terms of polynomial series using Taylor's theorem.

Execution of Model

Further work needs to be done in terms of actual execution of the model. We believe that there are many interesting dynamics that can be investigated. For example, we want to know if the model converges to the optimal solution in a reasonable time. We also want to possibly conduct post-optimality analysis on the parameters of the fitness function and the search space to see its effect on the max-plus model solution.

Variants of Evolutionary Computations

Another interesting open problem is to extend these max-plus results to other evolutionary algorithms (EAs) namely, Genetic Programming, Evolutionary Strategy, Evolutionary Programming, other derivative-free optimization and search methods like Simulated Annealing and Tabu Search. All these optimization techniques are similar and their functioning are based on the principle of natural selection, that is the survival of the fittest. All operate on fixed length strings, incorporate selection, mutation and recombination operators as in the case of Genetic Algorithms. The formulation of these problems using max-plus algebra, therefore, could yield promising results.

Bibliography

[1] AKIAN, M., BAPAT, R., GAUBERT, S., 2007. *Max-plus algebra: Handbook of Linear Algebra.* Discrete Mathematics and Its applications. Chapman and Hall CRC.

[2] AMINU, A., 2011. *Simultaneous Solution of Linear Equations and Inequalities in Max-Algebra*, Kybernetika-volume 47, number 2, pages 241-250.

[3] BACCELLI, F., COHEN, G., OLSDER, G. L., and QUADRAT, J.P. 1992. *Synchronization and Linearity: An algebra for discrete event systems*, Wiley, New York.

[4] BASSER, T., and BERNHARD, P. 1991. *H_∞-Optimal Control and Related Minimax Design Problems*, Birkhauser, Boston.

[5] BERTSEKAS, D.P. 1987. *Dynamic Programming, Deterministic and Stochastic Methods*, Prentice-Hall, Englewood Cliffs, NJ.

[6] BIGGS, N. 1993. *Algebraic Graph Theory* 2^{nd} ed. Cambridge University Press.

[7] BOOM, T.J.J van den, SCHUTTER, B., SCHULLERUS, G., and KREBS, V. 2004. *Adaptive model predictive control for max-plus-linear*

discrete event input-output systems, IEE *Proceedings-Control Theory and Applications*, vol.151, no.3, pp. 339-346.

[8] BUTKOVIČ, P. 2008. *Introduction to max-plus algebra*, The University of Birmingham, UK.

[9] BUTKOVIČ, P. 1994. Strong regularity of matrices-a survey of results *Discrete Applied Mathematics*, 48:45-68.

[10] BUTKOVIČ, P. 2003. *Max-algebra: the linear algebra of combinatorics*, Linear Algebra and Appl. 367, pp. 313-335.

[11] BUTKOVIČ, P.,and AMINU, A. 2009. *Introduction to Max-linear Programming*, IMA J. Management Mathematics 20, 3, 233-249.

[12] Carré, B.A., 1971. *An algebra for network routing problems*, J.Inst.Math.Appl.7 (273).

[13] CUNINGHAME-GREEN, R. 1979. *Minimax Algebra*. Notes on Economics and Mathematical Systems. Springer Verlag.

[14] De SCHUTTER, B., van den BOOM, T. 2001 *Model predictive control for max-plus linear discrete event systems.*, 37(7):1050-1056.

[15] FARLOW, K.G. 2009. *Max-plus algebra*, A thesis submitted to the Faculty of the Virginia Polytechnic and State University in partial fulfillment of the requirements for the degree of Masters in Mathematics.

[16] GAUBERT, S., and KATZ, R. 2007 *The Minkowski Theorem for max-plus convex sets, Linear Algebra and its Applications* 421 pp. 350-370.

[17] GAUBERT, S. 1997. *Methods and applications of max-plus linear algebra*, STACS 1997, Lecture notes in computer science **500**, Springer-Verlag, Berlin, pp. 261-282.

[18] GOLDBERG, D. 1989. *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison-Wesley.

[19] GONDRAN, M., and MINOUX, M. 1984. *Linear algebra in diodids: A survey of recent results*, Annals of Discrete Mathematics, 19 pp.130-160.

[20] HEIDERGOTT, B., OLSDER, G. L., van der WOUDE, J. 2006. *Max-plus at Work.* Princeton, New Jersey: Princeton University Press.

[21] HERRMANN, M. 2011 *The Schema Theorem and the Building Block Hypothesis*

[22] HOLLAND J.H., 1975 *Adaptation in Natural and Artificial Systems*, Ann. Arbor: The University of Michigan Press.

[23] HOGBEN, L., BRUALDI, R., GREENBAUM, A., and MATHIAS, R., 2007. *Handbook of linear algebra: Discrete Mathematics and its applications*, Volume 39, chapter 25, Chapman and Hall, ISBN 1-58488-510-6.

[24] HOUCK, C., JOINES, J. 1994. *On the use of non-stationary penalty functions to solve constrained optimization problems with genetic algorithms.* In 1994 IEEE International Symposium Evolutionary Computation, Orlando, Fl, pp. 579-584.

[25] KARP, R.M., 1978. *A characterization of the minimum cycle mean in a digraph*, Discrete Mathematics 23, 309-311.

[26] KOLOKOLTSOV, V.N., MASLOV, V. P. 1997. *Idempotent analysis and applications.* Kluwer Acad. Publisher.

[27] KOLOKOLTSOV, V.N., MASLOV, V. P. 1988. *The Cauchy problem for the homogeneous Bellman equation.* Soviet Math. Dokl., 36(2):326-330.

[28] LIONS, P.L., *Generalized Solutions of Hamilton-Jacobi Equations* Pitman.

[29] McENEANEY, W.M 2004. *Max-plus eigenvector methods for nonlinear H problems: error analysis*, SIAM J. Control Optimization.46 (2004),pp. 380-412

[30] MASLOV, V.P., 1987. *On a new principle of superposition for optimization problems* Russian Math. surveys, 42 pp. 43-54.

[31] MAHFORD S.W., 1994 *Population sizing for sharing methods*, Dept. Computer Science, Univ. of Illinois, IlliGAL Rep. 94005.

[32] MICHALEWICH, Z. 1994. *Genetic Algorithms + Data Structures = Evolution Programs. AI Series.* Springer-Verlag, New York.

[33] OLSDER, G.J.,and ROOS, C. 1988. *Cramer and Cayley-Hamilton in the max-algebra*, Linear Algebra and its Applications.101 (1988), pp. 87-108.

[34] SERGREEV, S. 2007 *Max-plus definite matrix closures and their eigenspaces*, Linear Algebra and Its Applications 421 pp. 182-201.

[35] VOSE, M.D. 1993 *A Critical Examination of the Schema Theorem*, University of Tennessee.