Finding a bounded mixed-integer solution to a system of dual network inequalities

P.Butkovič*
University of Birmingham

April 24, 2008

Abstract

We show that using max-algebraic techniques it is possible to generate the set of all solutions to a system of inequalities $x_i - x_j \geq b_{ij}$, $i, j = 1, ..., n$ using n generators. This efficient description enables us to develop a pseudopolynomial algorithm which either finds a bounded mixed-integer solution, or decides that no such solution exists.

AMS Classification [2000]: Primary 68Q25, 15A18; Secondary 90C10, 90C35, 15A36.

Keywords: Eigenvector; Dual Network Inequalities; Max-Algebra.

1 Introduction

This papers deals with the systems of inequalities of the form

$$x_i - x_j \geq b_{ij} \quad (i, j = 1, ..., n) \tag{1}$$

where $B = (b_{ij}) \in \mathbb{R}^{n \times n}$. In [19] the matrix of the left-hand side coefficients of this system is called the *dual network matrix*. It is the transpose of the constraint matrix of a circulation problem in a network (such as the maximum flow or minimum-cost flow problem) and inequalities of the form (1) therefore appear as dual inequalities for this type of problems. These facts motivate us to call (1) the *system of dual network inequalities* (SDNI). The aim of this paper is to show that using standard max-algebraic techniques it is possible to generate the set of all solutions to (1) (which is of size $n^2 \times n$) using n generators (Theorem 2.3). This description enables us then to find a *bounded mixed-integer solution* to the following system of dual network inequalities (BMISDNI), or to decide that there is no such solution:

$$x_i - x_j \geq b_{ij} \qquad (i, j \in N)$$

*School of Mathematics, University of Birmingham, Edgbaston, Birmingham B15 2TT, United Kingdom, p.butkovic@bham.ac.uk

$$u_j \geq x_j \geq l_j \qquad (j \in N)$$

$$x_j \text{ integer} \qquad (j \in J)$$

where $u = (u_1, ..., u_n)^T, l = (l_1, ..., l_n)^T \in \mathbb{R}^n$ and $J \subseteq N = \{1, ..., n\}$ are given. Note that without loss of generality u_j and l_j may be assumed to be integer for $j \in J$. This type of inequalities have been studied for instance in [19] where it has been proved that a related mixed-integer feasibility question is NP-complete. For similar problems see also [15].

We will show that in general, the application of max-algebra leads to a pseudopolynomial algorithm for solving BMISDNI. However, an explicit solution is proved in the case when B is integer (but still a mixed-integer solution is wanted). This implies that BMISDNI can be solved using $O(n^3)$ operations. Note that when $J = \emptyset$ then BMISDNI is polynomially solvable since it is a set of constraints of a linear program. When $J = N$ and B is integer then BMISDNI is also polynomially solvable since the matrix of the system is totally unimodular [16].

2 All solutions to SDNI

The system

$$x_i - x_j \geq b_{ij} \quad (i, j \in N)$$

is equivalent to

$$\max_{j \in N} (b_{ij} + x_j) \leq x_i \ (i \in N).$$

If we denote $u \oplus v = \max(u, v)$ and $u \otimes v = u + v$ for $u, v \in \overline{\mathbb{R}} := \mathbb{R} \cup \{-\infty\}$ then this reads $\sum_{j \in N}^{\oplus} b_{ij} \otimes x_j \leq x_i$ for $i \in N$ or (if we extend the operations $\oplus$ and $\otimes$ to matrices and vectors), equivalently

$$B \otimes x \leq x. \tag{2}$$

Being motivated by this observation we first summarize some basic concepts and results of max-algebra and then we present our main results.

By *max-algebra* we understand the analogue of linear algebra developed for the pair of operations $(\oplus, \otimes)$, extended to matrices and vectors. That is if $A = (a_{ij}),\ B = (b_{ij})$ and $C = (c_{ij})$ are matrices of compatible sizes with entries from $\overline{\mathbb{R}}$, we write $C = A \oplus B$ if $c_{ij} = a_{ij} \oplus b_{ij}$ for all i, j and $C = A \otimes B$ if $c_{ij} = \sum_k^{\oplus} a_{ik} \otimes b_{kj} = \max_k (a_{ik} + b_{kj})$ for all i, j. If $\alpha \in \overline{\mathbb{R}}$ then $\alpha \otimes A = (\alpha \otimes a_{ij})$. If $\alpha \in \mathbb{R}$ then the symbol α^{-1} stands for $-\alpha$.

The following isotonicity lemma is easily verified:

Lemma 2.1 *If $A \in \overline{\mathbb{R}}^{n \times n}$ and $x, y \in \overline{\mathbb{R}}^n$ then $x \leq y$ implies $A \otimes x \leq A \otimes y$.*

The letter I will stand for any square matrix whose diagonal entries are 0 and off-diagonal entries are $-\infty$. If A is an $n \times n$ matrix and k is a positive integer then the iterated product $A \otimes A \otimes ... \otimes A$ in which the symbol A appears

k-times will be denoted by A^k and $A^* = I \oplus A \oplus A^2 \oplus ... \oplus A^n$. Any set of the form

$$\{A \otimes z; z \in \mathbb{R}^n\}$$

is a finitely generated *max-algebraic linear subspace* (sometimes also called a *maxcone*) whose essentially unique basis can be found efficiently [7].

Given $A = (a_{ij}) \in \overline{\mathbb{R}}^{n \times n}$ the symbol D_A denotes the *associated digraph*, that is the arc-weighted digraph (N, E, w) where $E = \{(i,j)\,; a_{ij} > -\infty\}$ 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 $w(\pi, A) = a_{i_1 i_2} + a_{i_2 i_3} + ... + a_{i_{p-1} i_p}$ if $p > 1$ and $-\infty$ if $p = 1$. The number $p-1$ is called the *length* of π and $w(\pi, A)$ the *weight* of π. It can be easily seen that A^k is the matrix of greatest weights of paths of length k between all pairs of nodes in D_A. If $i_1 = i_p$ but $p > 1$ then π is called a *cycle*; it is called *positive* if $w(\pi, A) > 0$.

Max algebra has been studied by many authors and the reader is referred to [14], [1] or [4] for more information about max-algebra, see also [9], [10], [11], [18], [20], [8], [13], [12], [2], [3].

A basic problem in max-algebra, motivated for instance by the efforts to solve synchronisation problems in some industrial processes [9], [1] is:

EIGENVECTOR [EV]: Given $A \in \overline{\mathbb{R}}^{n \times n}$ find all $x \in \overline{\mathbb{R}}^n$, $x \neq (-\infty, ..., -\infty)^T$ such that $A \otimes x = \lambda \otimes x$ for some $\lambda \in \overline{\mathbb{R}}$.

EV has been studied since the 1960's and can now be efficiently solved [10] [11], [8], [1], [14], [4]. It is known that an $n \times n$ matrix may have up to n eigenvalues. The set of eigenvectors corresponding to a particular eigenvalue is a finitely generated max-algebraic linear subspace.

In this paper we only discuss finite (real matrices) but most of the results can be extended to matrices over $\overline{\mathbb{R}}$. If $A = (a_{ij}) \in \mathbb{R}^{n \times n}$ then A has a unique (max-algebraic) eigenvalue equal to the maximum cycle mean (notation $\lambda(A)$) of the associated digraph, that is

$$\lambda(A) = \max \frac{a_{i_1 i_2} + a_{i_2 i_3} + ... + a_{i_{p-1} i_p}}{p}$$

where the maximisation is taken over all p-tuples of indices from N, and $p = 1, 2, ..., n$. All eigenvectors are finite and the set of eigenvectors can easily be described. It follows from the definition of $\lambda(A)$ that $\lambda(A) \leq 0$ means that there are no positive cycles in D_A. It is known [1], [14] that in this case A^* is the matrix of greatest weights of paths between all pairs of nodes in D_A with added zero entries on the diagonal. This matrix can be found using standard $O(n^3)$ algorithms such as Floyd-Warshall's [16].

For $A \in \mathbb{R}^{n \times n}$ and $\mu \in \mathbb{R}$ we denote

$$Sol(A, \mu) = \{x \in \mathbb{R}^n; A \otimes x \leq \mu \otimes x\}.$$

Theorem 2.1 *([6], Cor.2.9) If $A \in \mathbb{R}^{n \times n}$ and $\mu \in \mathbb{R}$ then*

1. *$Sol(A, \mu) \neq \emptyset$ if and only if $\lambda(A) \leq \mu$.*

2. *If $Sol(A, \mu) \neq \emptyset$ then*

$$Sol(A, \mu) = \{(\mu^{-1} \otimes A)^* \otimes z; z \in \mathbb{R}^n\}$$

Remark 2.1 *It is known that $Sol(A, \mu)$ is actually the set of (max-algebraic) eigenvectors of the matrix*

$$I \oplus \mu^{-1} \otimes A.$$

Max-algebra also works with *dual operations*: $u \oplus' v = \min(u, v)$ and $u \otimes' v = u \otimes v$ for $u, v \in \mathbb{R}$ (the operators $\otimes$ and $\otimes'$ coincide for reals). The *conjugate* of a square matrix $A = (a_{ij})$ is $A^\sharp = (-a_{ji})$.

Theorem 2.2 *[9] If $A \in \mathbb{R}^{n \times n}, b \in \mathbb{R}^n$ and $z \in \mathbb{R}^n$ then*

$$A \otimes z \leq b \text{ if and only if } z \leq A^\sharp \otimes' b$$

Corollary 2.1 *If $A \in \mathbb{R}^{n \times n}$ and $v \in \mathbb{R}^n$ then $A \otimes \left(A^\sharp \otimes' v\right) \leq v$ and (by isotonicity) $A \otimes z \leq A \otimes \left(A^\sharp \otimes' v\right)$ for every z satisfying $A \otimes z \leq v$.*

We can now use Theorems 2.1 and 2.2 to describe all solutions to SDNI. In (2) we obviously have $\mu = 0$ and B plays the role of A. For simplicity we denote $Sol(B, 0)$ by $Sol(B)$. We start with an immediate transcription of Theorem 2.1.

Theorem 2.3 *If $B \in \mathbb{R}^{n \times n}$ then*

1. *$Sol(B) \neq \emptyset$ if and only if $\lambda(B) \leq 0$.*
2. *If $Sol(B) \neq \emptyset$ then*

$$Sol(B) = \{B^* \otimes z; z \in \mathbb{R}^n\}.$$

Hence the set of all solutions to SDNI is a finitely generated max-algebraic linear subspace.

Corollary 2.2 *The set of all solutions x to SDNI satisfying $x \leq u$ is*

$$\left\{B^* \otimes z; z \leq (B^*)^\sharp \otimes' u\right\}$$

and if this set is non-empty then the vector $B^* \otimes \left((B^*)^\sharp \otimes' u\right)$ is the greatest element of this set. Hence the inequality

$$l \leq B^* \otimes \left((B^*)^\sharp \otimes' u\right)$$

is necessary and sufficient for the existence of a solution to SDNI satisfying $l \leq x \leq u$.

Proof. It follows from (2) and Theorem 2.3 part 2. that solutions to SDNI are exactly the vectors of the form $B^* \otimes z, z \in \mathbb{R}^n$. Therefore solutions to SDNI satisfying $x \leq u$ are exactly the vectors $B^* \otimes z, B^* \otimes z \leq u$. By Theorem 2.2 this means the same as $B^* \otimes z, z \leq (B^*)^\sharp \otimes' u$ and the first part follows. For the second part realise that $B^* \otimes \left((B^*)^\sharp \otimes' u\right)$ is by Corollary 2.1 the greatest solution to SDNI satisfying $x \leq u$. ∎

3 Solving BMISDNI

We start by another corollary to Theorem 2.3.

Corollary 3.1 *A necessary condition for BMISDNI to have a solution is that $\lambda(B) \leq 0$. If this condition is satisfied then the BMISDNI is equivalent to finding a vector $z \in \mathbb{R}^n$ such that*

$$l \leq B^* \otimes z \leq u$$

and

$$\left(B^* \otimes z\right)_j \quad \text{integer for } j \in J.$$

Remark 3.1 *Recall that $\lambda(B) \leq 0$ means there is no positive cycle in D_B and in what follows we will assume that this condition is satisfied.*

Theorem 3.1 *Let $A \in \mathbb{R}^{n \times n}, b \in \mathbb{R}^n$ and $J \subseteq N$. Let $\tilde{b}$ be defined by*

$$\begin{aligned} \tilde{b}_j &= \lfloor b_j \rfloor \text{ for } j \in J, \\ \tilde{b}_j &= b_j \text{ for } j \notin J. \end{aligned}$$

Then the following are equivalent:

1. *There exists a $z \in \mathbb{R}^n$ such that $l \leq A \otimes z \leq b$ and*
$$\left(A \otimes z\right)_j \text{ integer for } j \in J.$$
2. *There exists a $z \in \mathbb{R}^n$ such that $l \leq A \otimes z \leq \tilde{b}$ and*
$$\left(A \otimes z\right)_j \text{ integer for } j \in J.$$
3. *There exists a $z \in \mathbb{R}^n$ such that $l \leq A \otimes z \leq A \otimes \left(A^\sharp \otimes' \tilde{b}\right)$ and*
$$\left(A \otimes z\right)_j \text{ integer for } j \in J.$$

Proof. 1. $\Longrightarrow$ 2.: If $\left(A \otimes z\right)_j \leq b_j$ and $\left(A \otimes z\right)_j$ is integer then $\left(A \otimes z\right)_j \leq \lfloor b_j \rfloor = \tilde{b}_j$ by the definition of the integer part.

2. $\Longrightarrow$ 1.: $\tilde{b}_j = \lfloor b_j \rfloor \leq b_j$ for $j \in J$ by definition and the statement follows.

2. $\Longrightarrow$ 3.: If $A \otimes z \leq \tilde{b}$ then by Theorem 2.2 $z \leq A^\sharp \otimes' \tilde{b}$ and by isotonicity (Lemma 2.1) $A \otimes z \leq A \otimes \left(A^\sharp \otimes' \tilde{b}\right)$.

3. $\Longrightarrow$ 2.: By Corollary 2.1 $A \otimes \left(A^\sharp \otimes' \tilde{b}\right) \leq \tilde{b}$ and so if $A \otimes z \leq A \otimes \left(A^\sharp \otimes' \tilde{b}\right)$ then also $A \otimes \left(A^\sharp \otimes' \tilde{b}\right) \leq \tilde{b}$. ■

Theorem 3.1 enables us to compile the following algorithm.

Algorithm 3.1 *BMISDNI*

Input: $B \in \mathbb{R}^{n \times n}, u, l \in \mathbb{R}^n$ and $J \subseteq N$

Output: x satisfying the BMISDNI conditions or an indication that no such vector exists.

[1] $A := B^*, x := u$

[2] $x_j := \lfloor x_j \rfloor$ for $j \in J$

[3] $z := A^\sharp \otimes' x$, $x := A \otimes z$

[4] If $l \nleqq x$ then stop (no solution)

[5] If $l \leq x$ and x_j integer for $j \in J$ then stop else go to [2]

Theorem 3.2 *Algorithm BMISDNI is correct and requires $O(n^3 + n^2 L)$ operations of addition, maximum, minimum, comparison and integer part, where*

$$L = \sum\nolimits_{j \in J} (u_j - l_j) .$$

Proof. If the algorithm terminates at step [4] then there is no solution by the repeated use of Theorem 3.1.

The sequence of vectors x constructed by this algorithm is non-increasing by Corollary 2.1 and hence $x = A \otimes z \leq u$ if it terminates at step [5]. The remaining requirements of the BMISDNI are satisfied explicitly due to the conditions in step [5].

Computational complexity: The calculation of B^* is $O(n^3)$ [16]. Each run of the loop [2]-[5] is $O(n^2)$. In every iteration at least one component of $x_j, j \in J$ decreases by one and the statement now follows from the fact that all x_j range between l_j and u_j. ∎

Example 3.1 *Let*

$$B = \begin{pmatrix} -2 & 2.7 & -2.1 \\ -3.8 & -1 & -5.2 \\ 1.6 & 3.5 & -3 \end{pmatrix}$$

$u = (5.2, 0.8, 7.4)^T, J = \{1, 3\}$ *(l is not specified). The algorithm BMISDNI will find:*

$$A = B^* = \begin{pmatrix} 0 & 2.7 & -2.1 \\ -3.6 & 0 & -5.2 \\ 1.6 & 4.3 & 0 \end{pmatrix}$$

$x = (5, 0.8, 7)^T$,

$$z = A^\sharp \otimes' x = \begin{pmatrix} 0 & 3.6 & -1.6 \\ -2.7 & 0 & -4.3 \\ 2.1 & 5.2 & 0 \end{pmatrix} \otimes' x = \begin{pmatrix} 4.4 \\ 0.8 \\ 6 \end{pmatrix}$$

$$x = A \otimes z = (4.4, 0.8, 6)^T .$$

Now $x_1 \notin \mathbb{Z}$ so the algorithm continues by another iteration: $x = (4, 0.8, 6)^T$,

$$z = A^\sharp \otimes' x = (4, 0.8, 6)^T$$

and

$$x = A \otimes z = (4, 0.8, 6)^T,$$

which is a solution to the BMISDNI (provided that $l \leq x$) since $x_1, x_3 \in \mathbb{Z}$ (otherwise there is no solution).

4 Solving BMISDNI for integer matrices

In this section we prove that a solution to the BMISDNI can be found explicitly if B is integer.

The following will be useful:

Theorem 4.1 *Let $A \in \mathbb{Z}^{n\times n}, b \in \mathbb{R}^n$ and $A \otimes x = b$ for some $x \in \mathbb{R}^n$. Let $J \subseteq N$ and $\tilde{b}$ be defined by*

$$\begin{aligned} \tilde{b}_j &= \lfloor b_j \rfloor \text{ for } j \in J \\ \tilde{b}_j &= b_j \text{ for } j \notin J. \end{aligned}$$

Then there exists an $\tilde{x} \in \mathbb{R}^n$ such that

$$A \otimes \tilde{x} \leq \tilde{b}$$

and

$$\left(A \otimes \tilde{x}\right)_j = \tilde{b}_j \text{ for } j \in J.$$

Proof. Let $k \in J$ be such that $b_k \notin \mathbb{Z}$. Since $b_k = \max_{j\in N}\left(a_{kj} + x_j\right)$, the set

$$S_k = \{s; a_{ks} + x_s > \lfloor b_k \rfloor\}$$

is non-empty and $x_s \notin \mathbb{Z}$ for every $s \in S_k$ since A is integer. Let $x^{(1)}$ be the vector defined by $x_j^{(1)} = \lfloor x_j \rfloor$ for $j \in S_k$ and $x_j^{(1)} = x_j$ otherwise. Clearly $x^{(1)} \leq x$ and so $A \otimes x^{(1)} \leq A \otimes x$ by Lemma 2.1. Let $r \in N$ be such that $\max_{j\in N}\left(a_{rj} + x_j\right) \in \mathbb{Z}$ (if any). Then $a_{rs} + x_s < \max_{j\in N}\left(a_{rj} + x_j\right)$ for all $s \in S_k$ since $x_s \notin \mathbb{Z}$. Therefore $\max_{j\in N}\left(a_{rj} + x_j^{(1)}\right) = \max_{j\in N}\left(a_{rj} + x_j\right)$. At the same time $\max_{j\in N}\left(a_{kj} + x_j^{(1)}\right) = \lfloor b_k \rfloor$ yielding that the number of indices r such that $\max_{j\in N}\left(a_{rj} + x_j^{(1)}\right) = \lfloor b_r \rfloor$ has increased by at least one compared to x. If there is still an index $k \in J$ such that $S_k \neq \emptyset$ then we repeat this construction and obtain $x^{(2)}, x^{(3)}, \ldots$. Since the number of indices r for which $\max_{j\in N}\left(a_{rj} + x_j\right) \in \mathbb{Z}$ increases at every step, this process stops after a finite number of steps with a vector $\tilde{x}$ satisfying the conditions in the theorem statement. ■

Corollary 4.1 *Under the assumptions of Theorem 4.1 and using the same notation, if* $\bar{x} = A^\sharp \otimes' \tilde{b}$ *then*

$$A \otimes \bar{x} \leq \tilde{b}$$

and

$$(A \otimes \bar{x})_j = \tilde{b}_j \text{ for } j \in J.$$

Proof. The inequality follows from Corollary 2.1. Let $\tilde{x}$ be the vector described in Theorem 4.1. By Theorem 2.2 we have $\tilde{x} \leq \bar{x}$ implying that

$$\tilde{b}_j = (A \otimes \tilde{x})_j \leq (A \otimes \bar{x})_j \leq \tilde{b}_j \text{ for } j \in J$$

which concludes the proof. ■

Our main result is:

Theorem 4.2 *Let* $B \in \mathbb{Z}^{n\times n}, \lambda(B) \leq 0, A = B^*, b = A \otimes \left(A^\sharp \otimes' u\right)$ *and* $\tilde{b}$ *be defined by*

$$\tilde{b}_j = \lfloor b_j \rfloor \text{ for } j \in J$$

and

$$\tilde{b}_j = b_j \text{ for } j \notin J.$$

Then the BMISDNI has a solution if and only if

$$l \leq A \otimes \left(A^\sharp \otimes' \tilde{b}\right),$$

and $\hat{x} = A \otimes \left(A^\sharp \otimes' \tilde{b}\right)$ *is then the greatest solution (that is* $y \leq \hat{x}$ *for any solution* y*).*

Proof. Note first that A is an integer matrix and we therefore may apply Corollary 4.1 to A.

"If": By Corollary 2.1 $\hat{x} \leq \tilde{b} \leq b \leq u$. Let us take in Corollary 4.1 (and Theorem 4.1) $x = A^\sharp \otimes' u$. Then $\hat{x} = A \otimes \bar{x}$ and so $\hat{x}_j \in \mathbb{Z}$ for $j \in J$.

"Only if": Let y be a solution. Then $y = A \otimes w \leq u$ for some $w \in \mathbb{R}^n$, thus by Theorem 2.2

$$w \leq A^\sharp \otimes' u$$

and so

$$y = A \otimes w \leq A \otimes \left(A^\sharp \otimes' u\right) = b.$$

Since $y_j \in \mathbb{Z}$ for $j \in J$ we also have

$$A \otimes w = y \leq \tilde{b}.$$

Hence by Theorem 2.2

$$w \leq A^\sharp \otimes' \tilde{b}$$

and by Lemma 2.1 then

$$l \leq y = A \otimes w \leq A \otimes \left(A^\sharp \otimes' \tilde{b}\right) = \hat{x}.$$

We also have $\hat{x} \leq \tilde{b} \leq b \leq u$ by Corollary 2.1 and $\hat{x}_j \in \mathbb{Z}$ for $j \in J$ by Corollary 4.1 as above, hence $\hat{x}$ is the greatest solution. ■

Example 4.1 *Let*

$$B = \begin{pmatrix} -2 & 2 & -2 \\ -3 & -1 & -4 \\ 1 & 3 & -3 \end{pmatrix}$$

$u = (3.5, 0.8, 5.7)^T, J = \{1, 3\}$ *(l is not specified). Theorem 4.2 provides:*

$$A = B^* = \begin{pmatrix} 0 & 2 & -2 \\ -3 & 0 & -4 \\ 1 & 3 & 0 \end{pmatrix}$$

$$A^\sharp \otimes' u = \begin{pmatrix} 0 & 3 & -1 \\ -2 & 0 & -3 \\ 2 & 4 & 0 \end{pmatrix} \otimes' u = \begin{pmatrix} 3.5 \\ 0.8 \\ 4.8 \end{pmatrix}$$

$$b = A \otimes \left(A^\sharp \otimes' u\right) = \begin{pmatrix} 3.5 \\ 0.8 \\ 4.8 \end{pmatrix}$$

$$\tilde{b} = \begin{pmatrix} 3 \\ 0.8 \\ 4 \end{pmatrix}$$

$$\hat{x} = A \otimes \left(A^\sharp \otimes' \tilde{b}\right) = (3, 0.8, 4)^T$$

By Theorem 4.2 $\hat{x}$ *is the greatest solution to the BMISDNI provided that* $l \leq \hat{x}$ *(otherwise there is no solution).*

5 A note on an application

As a by-product, this paper provides a solution technique for solving a scheduling-type of problems.

Consider a multiprocessor interactive system (of production, transportation, information technology, etc.) in which the individual processors work in stages and a processor, say P cannot start its work in a new stage until all or some of the processors have finished their activities necessary for P [10], [11], [14]. It is assumed that each of the processors $P_1, ..., P_n$ can work for all other processors simultaneously and that a processor starts all these activities as soon as it starts to work.

Let $x_i(r)$ denote the starting time of the r^{th}stage on processor i $(i = 1, ..., n)$ and let a_{ij} denote the duration of the operation at which the j^{th} processor prepares the component necessary for the i^{th} processor in the $(r+1)^{st}$ stage $(i, j = 1, ..., n)$. Then

$$x_i(r+1) = \max(x_1(r) + a_{i1}, ..., x_n(r) + a_{in})(i = 1, ..., n; r = 0, 1, ...)$$

or, in max-algebraic notation

$$x(r+1) = A \otimes x(r)(r = 0, 1, ...)$$

where $A = (a_{ij})$ is a *production matrix*. We say that the system is in a *steady state* [9] if it moves forward in regular steps, that is if for some λ we have $x(r+1) = \lambda \otimes x(r)$ for all r. This implies $A \otimes x(r) = \lambda \otimes x(r)$ for all r. Therefore the system is in a steady state in all stages if and only if for some λ, the starting times vector $x(0)$ is a solution to

$$A \otimes x = \lambda \otimes x.$$

For practical reasons it may be necessary to find the starting times for the individual processors within given bounds, for instance $u_j \geq x_j \geq l_j$ for all j. If an eigenvector within these bounds does not exist then it may be interesting to find a subeigenvector, that is an x satisfying

$$A \otimes x \leq \lambda \otimes x \tag{3}$$

and $u_j \geq x_j \geq l_j$ for all j (in this case a new stage at any processor starts within a given time limit λ after the beginning of the previous stage). Solvability of (3) is answered by Theorem 2.1 and once this is affirmative it remains to solve

$$B \otimes x \leq x$$

$$l \leq x \leq u$$

where $B = \lambda^{-1} \otimes A$. The answer to this question is given in Corollary 2.2.

Acknowledgement The author wishes to thank two anonymous referees for their constructive remarks that helped to improve the presentation of this paper.

References

[1] F.L.Baccelli, G.Cohen, G.-J.Olsder and J.-P.Quadrat, Synchronization and Linearity, John Wiley, Chichester, New York, 1992.

[2] R.B.Bapat, A max version of the Perron Frobenius Theorem, Linear Algebra and Its Applications 275/276 (1998) 3-18.

[3] R.B.Bapat, Pattern properties and spectral inequalities in max algebra, SIAM J on Matrix Analysis and Applications, 16:964-976, 1995.

[4] P.Butkovic, Max-algebra: the linear algebra of combinatorics? Linear Algebra and Its Applications 367 (2003) 313-335.

[5] R.E.Burkard and P.Butkovic, Max algebra and the linear assignment problem, Mathematical Programming, Ser.B 98: 415-429 (2003).

[6] P.Butkovic and H.Schneider, Applications of max-algebra to diagonal scaling of matrices, Electronic Journal of Linear Algebra 13 (2005) 262-273.

[7] P.Butkovic, H.Schneider and S.Sergeev: Generators, Extremals and Bases of Max Cones, Linear Algebra and Its Applications 421 (2007) 394-406.

[8] G.Cohen, D.Dubois, J.-P.Quadrat and M.Viot, A Linear-System-Theoretic View of Discrete-Event Processes and Its Use for Performance Evaluation in Manufacturing, IEEE Transactions on Automatic Control, Vol. AC-30, No 3, 1985.

[9] R.A.Cuninghame-Green, Describing industrial processes with interference and approximating their steady-state behaviour, Operations Research Quarterly 13 (1962) 95-100.

[10] R.A.Cuninghame-Green, *Minimax Algebra.* Lecture Notes in Economics and Mathematical Systems **166**, Berlin, Springer, 1979.

[11] R.A.Cuninghame-Green, Minimax Algebra and Applications in: Advances in Imaging and Electron Physics, vol. 90, pp. 1–121, Academic Press, New York, 1995.

[12] S.Gaubert, Théorie des systèmes linéaires dans les dioïdes, Thèse, Ecole des Mines de Paris, 1992.

[13] M.Gondran and M.Minoux, Valeurs propres et vecteur propres dans les dioïdes et leur interprétation en théorie des graphes, Bulletin de la direction des etudes et recherches, Serie C, Mathematiques et Informatiques, No 2 (1977) 25-41.

[14] B.Heidergott, G.J.Olsder and J. van der Woude (2006), Max Plus at Work: Modeling and Analysis of Synchronized Systems, A Course on Max-Plus Algebra, PUP.

[15] D.Hochbaum and J. Naor, Simple and fast algorithms for linear and integer programs with two variables per inequality, SIAM Journal on Computing, vol 23, No. 6 (1994) 1179-1192.

[16] C.H.Papadimitriou and K.Steiglitz, *Combinatorial Optimization-Algorithms and Complexity*, Dover Publications, 1998.

[17] D.R.Shier, A decomposition algorithm for optimality problems in tree-structured networks, Discrete Mathematics 6 (1973) 175-189.

[18] N.N.Vorobjov, Extremal Algebra of Positive Matrices, Elektronische Datenverarbeitung und Kybernetik 3 (1967) 39-71 (in Russian).

[19] M.Conforti, M.Di Summa, F.Eisenbrand and L.Wolsey, Network formulations of mixed-integer programs, CORE discussion paper 2006/117, Université Catholique de Louvain, Belgium.

[20] K.Zimmermann, Extremální algebra, *Výzkumná publikace Ekonomicko - matematické laboratoře při Ekonomickém ústavě ČSAV*, 46, Praha (1976) (in Czech).