

# Reduction and Analysis of a Max-Plus Linear System to a Constraint Satisfaction Problem for Mixed Integer Programming

Hajime Yokoyama, Hiroyuki Goto

Department of Industrial & System Engineering, Hosei University, Tokyo, Japan

Email: hajime.yokoyama.3m@stu.hosei.ac.jp, goto-h@hosei.ac.jp

**How to cite this paper:** Yokoyama, H. and Goto, H. (2017) Reduction and Analysis of a Max-Plus Linear System to a Constraint Satisfaction Problem for Mixed Integer Programming. *American Journal of Operations Research*, 7, 113-120.

<https://doi.org/10.4236/ajor.2017.72008>

**Received:** January 28, 2017

**Accepted:** March 12, 2017

**Published:** March 15, 2017

Copyright © 2017 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

This research develops a solution method for project scheduling represented by a max-plus-linear (MPL) form. Max-plus-linear representation is an approach to model and analyze a class of discrete-event systems, in which the behavior of a target system is represented by linear equations in max-plus algebra. Several types of MPL equations can be reduced to a constraint satisfaction problem (CSP) for mixed integer programming. The resulting formulation is flexible and easy-to-use for project scheduling; for example, we can obtain the earliest output times, latest task-starting times, and latest input times using an MPL form. We also develop a key method for identifying critical tasks under the framework of CSP. The developed methods are validated through a numerical example.

## Keywords

Max-Plus Algebra, Scheduling, Critical Path, Constraint Satisfaction Problems, Mixed Integer Programming

---

## 1. Introduction

This research develops a solution method for a project scheduling using max-plus algebra (MPA). Goto [1] developed a solution method for two types of linear equations in MPA:  $A \otimes x = b$  and  $x = A \otimes x \oplus b$ , where  $\oplus$  and  $\otimes$  are the max and plus operations in MPA. These equations, also referred to as max-plus-linear (MPL) equations, can be reduced to constraint satisfaction problems (CSPs) for mixed integer programming (MIP) [1]. MPA [2] [3] is an algebraic system wherein the max and plus operations are defined as addition and multiplication, respectively. MPL representation is an approach to model and analyze a class of discrete-event systems with structures of non-concurrency, synchroniza-

tion, parallel processing of multiple tasks, and so on. Such systems include production systems and project scheduling, the behavior of which is represented by linear equations in MPA. In project scheduling, the linear equation

$x = A \otimes x \oplus b$  is used to obtain the earliest start times  $x$ , while the equation  $A \otimes x = b$  to obtain the latest event occurrence times  $x$ . If both matrix  $A$  and vector  $b$  consist of constants only, then the optimal solution is a unique constants vector. There are already solution methods for such case [4]. On the other hand, if matrix  $A$  or vector  $b$  contain variables, then the optimal solution becomes a function of variables. For example, suppose that matrix  $A$  includes system parameters which correspond to the duration times. If matrix  $A$  or vector  $b$  is incorporated into other constraints, the existing solution methods cannot be used. Accordingly, Goto [1] developed a solution method for these equations by reducing the two MPL equations into CSPs for MIP. However, the reference does not provide a solution method for calculating other essential quantities in project scheduling.

Therefore, we newly develop a solution method for calculating the earliest output times to all output transitions, latest task-starting times, and latest input times, all of which are represented by an MPL from. In addition, we develop a method to identify critical tasks under the framework of CSP, which shall be a key method in this article.

## 2. Max-Plus Algebra

We define a set  $\mathbb{R}_{\max} = \mathbb{R} \cup \{-\infty\}$ , where  $\mathbb{R}$  is the whole real line. Then, for  $x, y \in \mathbb{R}_{\max}$ , we define the following two basic operators:

$$x \oplus y = \max(x, y), \tag{1}$$

$$x \otimes y = x + y. \tag{2}$$

Additionally, we define a set  $\bar{\mathbb{R}}_{\max} = \mathbb{R} \cup \{-\infty\} \cup \{+\infty\}$  to add the following two complementary operators:

$$x \wedge y = \min(x, y), \tag{3}$$

$$x \odot y = -x + y. \tag{4}$$

The priority of operators  $\otimes$  and  $\odot$  is higher than that of  $\oplus$  and  $\wedge$ . We shall denote the zero and unit elements for operators  $\oplus$  and  $\otimes$  by  $\varepsilon (= -\infty)$  and  $e (= 0)$ , respectively, and the unit element for operator  $\wedge$  by  $\bar{\varepsilon} (= +\infty)$ . If  $X, Y \in \mathbb{R}_{\max}^{n \times m}$ , and  $Z \in \mathbb{R}_{\max}^{m \times q}$ , then

$$[X \oplus Y]_{ij} = [X]_{ij} \oplus [Y]_{ij}, \tag{5}$$

$$[X \otimes Z]_{ij} = \max_{k=1}^n ([X]_{ik} \otimes [Z]_{kj}). \tag{6}$$

Moreover,

$$[X \wedge Y]_{ij} = [X]_{ij} \wedge [Y]_{ij}, \tag{7}$$

$$[X \odot Z]_{ij} = \min_{k=1}^n (-[X]_{ik} \odot [Z]_{kj}), \tag{8}$$

$$X^T \odot Z = X \setminus Z. \tag{9}$$

$\varepsilon$  is a matrix whose elements are all  $\varepsilon$ , and  $e$  is a matrix whose diagonal elements are  $e$  and off-diagonal elements are  $\varepsilon$ . If  $X \in \mathbb{R}_{\max}^{n \times n}$ , then  $X^*$  represents the Kleene staroperation defined below:

$$X^* = e \oplus X \oplus X^{\otimes 2} \oplus \dots \oplus X^{\otimes (s-1)} \tag{10}$$

where  $s(1 \leq s \leq n)$  is an instance that satisfies  $X^{\otimes (s-1)} \neq \varepsilon$  and  $X^{\otimes s} = \varepsilon$ .

### 3. Reduction of the Relevant Operators to CSPs

The addition and multiplication operators are reduced to CSPs for MIP. For  $x, y, z \in \mathbb{R}_{\max}$ ,

$$z = x \oplus y, \tag{11}$$

$$z = x \otimes y \tag{12}$$

are focused on. First, Equation (11) is reduced to a CSP. The resulting formulation is:

$$\begin{aligned} z &\geq x, \\ z &\geq y, \\ z &\leq x + M(1 - s_1), \\ z &\leq y + M(1 - s_2), \\ (s_1, s_2) &\in \{0, 1\}, \\ s_1 + s_2 &\geq 1, \end{aligned} \tag{13}$$

where  $s_1, s_2$  are binary variables, whilst  $M$  is a large positive constant called big-M. The big-M,  $s_1$ , and  $s_2$  play a role of switching the equal signs. By generalizing this result for multiple numbers, the addition  $z = \bigoplus_{i=1}^n x_i$  can be reduced to a CSP as follows:

$$\begin{aligned} z &\geq x_i && \forall i \in \mathcal{V}_n, \\ z &\leq x_i + M(1 - s_i) && \forall i \in \mathcal{V}_n, \\ s_i &\in \{0, 1\} && \forall i \in \mathcal{V}_n, \\ \sum_{k=1}^n s_k &\geq 1, \end{aligned} \tag{14}$$

where  $\mathcal{V}_n = \{1, 2, \dots, n\}$ . If  $X, Y \in \mathbb{R}_{\max}^{n \times m}$ , and  $Z \in \mathbb{R}_{\max}^{m \times r}$ , then the addition  $P = X \oplus Y$  can be formulated as follows:

$$\begin{aligned} p_{ij} &\geq x_{ij} && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m, \\ p_{ij} &\geq y_{ij} && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m, \\ p_{ij} &\leq x_{ij} + M(1 - s_{ij1}) && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m, \\ p_{ij} &\leq y_{ij} + M(1 - s_{ij2}) && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m, \\ (s_{ij1}, s_{ij2}) &\in \{0, 1\} && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m, \\ s_{ij1} + s_{ij2} &\geq 1 && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m. \end{aligned} \tag{15}$$

With respect to Equation (12), the multiplication of two numbers in MPA can be formulated as follows:

$$z = x + y. \tag{16}$$

Then, the multiplication of two matrices  $Q = Y \otimes Z$  can be reduced to:

$$\begin{aligned} q_{ij} &\geq y_{ik} + z_{kj} && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_r, \forall k \in \mathcal{V}_m, \\ q_{ij} &\leq y_{ik} + z_{kj} + M(1 - s_{ijk}) && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_r, \forall k \in \mathcal{V}_m, \\ s_{ijk} &\in \{0, 1\} && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_r, \forall k \in \mathcal{V}_m, \\ \sum_{l=1}^m s_{ijl} &\geq 1 && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_r. \end{aligned} \tag{17}$$

Next, we focus on the two complementary operators:

$$z = x \wedge y, \tag{18}$$

$$z = x \odot y. \tag{19}$$

In a similar manner to the reduction of  $x \oplus y$ , Equation (18) can be reduced to:

$$\begin{aligned} z &\leq x, \\ z &\leq y, \\ z &\geq x - M(1 - s_1), \\ z &\geq y - M(1 - s_2), \\ (s_1, s_2) &\in \{0, 1\}, \\ s_1 + s_2 &\geq 1. \end{aligned} \tag{20}$$

By generalizing this result for multiple numbers, the minimization  $z = \wedge_{i=1}^n x_i$  can be reduced as follows:

$$\begin{aligned} z &\leq x_i && \forall i \in \mathcal{V}_n, \\ z &\geq x_i - M(1 - s_i) && \forall i \in \mathcal{V}_n, \\ s_i &\in \{0, 1\} && \forall i \in \mathcal{V}_n, \\ \sum_{k=1}^n s_k &\geq 1, \end{aligned} \tag{21}$$

If  $X, Y \in \mathbb{R}_{\max}^{n \times m}$ , and  $Z \in \mathbb{R}_{\max}^{n \times r}$ , then the minimization of two matrices  $P = X \wedge Y$  can be formulated as follows:

$$\begin{aligned} p_{ij} &\leq x_{ij} && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m, \\ p_{ij} &\leq y_{ij} && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m, \\ p_{ij} &\geq x_{ij} - M(1 - s_{ij1}) && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m, \\ p_{ij} &\geq y_{ij} - M(1 - s_{ij2}) && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m, \\ (s_{ij1}, s_{ij2}) &\in \{0, 1\} && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m, \\ s_{ij1} + s_{ij2} &\geq 1 && \forall i \in \mathcal{V}_n, \forall j \in \mathcal{V}_m. \end{aligned} \tag{22}$$

Next, we focus on Equation (19), which can be reduced in a straightforward manner from the definition of  $\odot Z$  to:

$$z = -x + y. \tag{23}$$

Then, the pseudo division operation  $Q = X^T \odot Z$  can be formulated as follows:

$$\begin{aligned} q_{ij} &\leq -x_{ik} + w_{kj} && \forall i \in \mathcal{V}_m, \forall j \in \mathcal{V}_r, \forall k \in \mathcal{V}_n, \\ q_{ij} &\geq -x_{ik} + w_{kj} + M(1 - s_{ijk}) && \forall i \in \mathcal{V}_m, \forall j \in \mathcal{V}_r, \forall k \in \mathcal{V}_n, \\ s_{ijk} &\in \{0, 1\} && \forall i \in \mathcal{V}_m, \forall j \in \mathcal{V}_r, \forall k \in \mathcal{V}_n, \\ \sum_{l=1}^n s_{ijl} &\geq 1 && \forall i \in \mathcal{V}_m, \forall j \in \mathcal{V}_r. \end{aligned} \tag{24}$$

## 4. Solution Methods for MPL Equations

### 4.1. MPL Representation

After defining the following relevant matrices and vectors, we introduce the MPL equations taken up in references [5] and [6]:

- $n$ : number of tasks;
- $p$ : number of external outputs;
- $q$ : number of external inputs;
- $\mathbf{B} \in \mathbb{R}_{\max}^{n \times q}$ : input matrix,  $[\mathbf{B}]_{ij} = \{e : \text{if task } i \text{ has an input transition } j, \varepsilon : \text{otherwise}\}$ ;
- $\mathbf{C} \in \mathbb{R}_{\max}^{p \times n}$ : output matrix,  $[\mathbf{C}]_{ij} = \{e : \text{if task } j \text{ has an output transition } i, \varepsilon : \text{otherwise}\}$ ;
- $\mathbf{d} \in \mathbb{R}_{\max}^n$ : system parameter,  $[\mathbf{d}]_i$ : duration time in task  $i$ ;
- $\mathbf{u} \in \mathbb{R}_{\max}^q$ : input vector,  $[\mathbf{u}]_i$ : input time to external input  $i$ ;
- $\mathbf{y} \in \mathbb{R}_{\max}^p$ : output vector,  $[\mathbf{y}]_i$ : output time from external output  $i$ ;
- $\mathbf{x} \in \mathbb{R}_{\max}^n$ : state vector,  $[\mathbf{x}]_i$ : start or completion time of task  $i$ .

The earliest task-completion times of all tasks,  $\mathbf{x}_E$ , are calculated using

$$\mathbf{x}_E = \mathbf{A} \otimes \mathbf{x}_E \oplus \mathbf{b}, \tag{25}$$

Matrix  $\mathbf{A} \in \mathbb{R}_{\max}^{n \times n}$  is the weighted transition matrix, and vector  $\mathbf{b} \in \mathbb{R}_{\max}^n$  is the weighted input vector which satisfies the following relation

$\mathbf{b} = \mathbf{P} \otimes \mathbf{B} \otimes \mathbf{u}$ . The earliest output times to all output transitions,  $\mathbf{y}_E$ , are then calculated by

$$\mathbf{y}_E = \mathbf{C} \otimes \mathbf{x}_E. \tag{26}$$

Then, the latest task-starting times,  $\mathbf{x}_L$ , are calculated using

$$\mathbf{x}_L = \mathbf{A}^T \odot \mathbf{x}_L \wedge \mathbf{c} \tag{27}$$

Vector  $\mathbf{c} \in \mathbb{R}_{\max}^n$  is the weighted output vector which satisfies:

$\mathbf{c} = \mathbf{P} \setminus (\mathbf{C} \setminus \mathbf{y}_E)$ . The latest input times,  $\mathbf{u}_L$ , are calculated using  $\mathbf{x}_L$ :

$$\mathbf{u}_L = \mathbf{B}^T \odot \mathbf{x}_L. \tag{28}$$

As a consequence, the total floats of all tasks can be calculated using Equations (25) and (27):

$$\mathbf{m} = (\mathbf{x}_L + \mathbf{d}) - \mathbf{x}_E. \tag{29}$$

All tasks can be subsequently classified into two types according to  $[\mathbf{m}]_i = 0$  or  $[\mathbf{m}]_i > 0$ , where the former one is classified as a critical task whereas the latter one as a non-critical task.

### 4.2. Reduction of MPL Equations to CSPs

We consider a solution method for Equation (25). A simple approach is to relax the equation into the following inequality:

$$\mathbf{x}_E \geq \mathbf{A} \otimes \mathbf{x}_E \oplus \mathbf{b}. \tag{30}$$

The solution that has the smallest elements satisfying Equation (30), also called the least solution, is given by  $\mathbf{x}_E = \mathbf{A}^* \otimes \mathbf{b}$ . We can reduce Equation (30)

to the following CSP for MIP in the following manner:

$$\begin{aligned}
 x_{Ei} &\geq a_{ik} + x_{Ek} && \forall (i, k) \in \mathcal{E}_A, \\
 x_{Ei} &\leq a_{ik} + x_{Ek} + M(1 - s_{ik}) && \forall (i, k) \in \mathcal{E}_A, \\
 x_{Ei} &\geq b_i && \forall (i, k) \notin \mathcal{E}_A, \\
 x_{Ei} &\leq b_i + M(1 - s_{ik}) && \forall (i, k) \notin \mathcal{E}_A, \\
 s_{ik} &\in \{0, 1\} && \forall i, k \in \mathcal{V}_n, \\
 \sum_{l=1}^n s_{il} &\geq 1 && \forall i \in \mathcal{V}_n.
 \end{aligned} \tag{31}$$

If  $a_{ik} = \varepsilon$ , then  $a_{ik} + x_{Ekj} = \varepsilon$  follows. It is notable that we can compute  $A^* \otimes b$  directly without calculating  $A^*$ . After relaxing the equation, we reduce Equation (26) to a CSP for MIP with the help of Equation (17):

$$\begin{aligned}
 y_{Ei} &\geq C_{ik} + x_{Ek} && \forall i \in \mathcal{V}_p, \forall k \in \mathcal{V}_n, \\
 y_{Ei} &\leq C_{ik} + x_{Ek} + M(1 - s_{ik}) && \forall i \in \mathcal{V}_p, \forall k \in \mathcal{V}_n, \\
 s_{ik} &\in \{0, 1\} && \forall i \in \mathcal{V}_p, \forall k \in \mathcal{V}_n, \\
 \sum_{l=1}^n s_{il} &\geq 1 && \forall i \in \mathcal{V}_p.
 \end{aligned} \tag{32}$$

Next, we consider a solution method for Equation (27). A simple approach is to relax the equation into the following inequality:

$$\mathbf{x}_L \leq A^T \odot \mathbf{x}_L \wedge \mathbf{c}. \tag{33}$$

The solution that has the maximum elements satisfying Equation (33), also called the greatest solution, is given by  $\mathbf{x}_L = A^* \setminus \mathbf{c}$ . We can reduce Equation (33) to a CSP for MIP in the following manner:

$$\begin{aligned}
 x_{Li} &\leq -a_{ik} + x_{Lk} && \forall (i, k) \in \mathcal{E}_A, \\
 x_{Li} &\geq -a_{ik} + x_{Lk} - M(1 - s_{ik}) && \forall (i, k) \in \mathcal{E}_A, \\
 x_{Li} &\leq c_i && \forall (i, k) \notin \mathcal{E}_A, \\
 x_{Li} &\geq c_i - M(1 - s_{ik}) && \forall (i, k) \notin \mathcal{E}_A, \\
 s_{ik} &\in \{0, 1\} && \forall i, k \in \mathcal{V}_n, \\
 \sum_{l=1}^n s_{il} &\geq 1 && \forall i \in \mathcal{V}_n.
 \end{aligned} \tag{34}$$

If  $a_{ik} = \varepsilon$ , then  $-a_{ki} + x_{Lkj} = \varepsilon$  follows. Here, it is again remarkable that Equation (34) can compute  $A^* \setminus \mathbf{c}$  directly without calculating  $A^*$ . After relaxing the equation, we reduce Equation (28) to a CSP for MIP with the help of Equation (24):

$$\begin{aligned}
 u_{Li} &\leq -B_{ik} + x_{Lk} && \forall i \in \mathcal{V}_q, \forall k \in \mathcal{V}_n, \\
 u_{Li} &\geq -B_{ik} + x_{Lk} + M(1 - s_{ik}) && \forall i \in \mathcal{V}_q, \forall k \in \mathcal{V}_n, \\
 s_{ik} &\in \{0, 1\} && \forall i \in \mathcal{V}_q, \forall k \in \mathcal{V}_n, \\
 \sum_{l=1}^n s_{il} &\geq 1 && \forall i \in \mathcal{V}_q.
 \end{aligned} \tag{35}$$

Lastly, we focus on reducing Equation (29), the resulting formulation of which is as follows:

$$m_i = (x_{Li} + d_i) - x_{Ei} \quad \forall i \in \mathcal{V}_n. \tag{36}$$

Then, we reduce Equation (36) to a CSP for MIP as follows.

$$\begin{aligned}
 m_i - M\alpha_i &\leq 0 & \forall i \in \mathcal{V}_n, \\
 m_i - (1/M)\alpha_i &\geq 0 & \forall i \in \mathcal{V}_n, \\
 \alpha_i &\in \{0,1\} & \forall i \in \mathcal{V}_n.
 \end{aligned}
 \tag{37}$$

If the total float  $m_i$  is a real number, then Equation (37) can be computed by introducing a small positive constant  $(1/M)$ . If  $\alpha_i = 0$ , then task  $i$  is critical since  $m_i = 0$  follows. Conversely, if  $\alpha_i = 1$ , then task  $i$  is non-critical because  $(1/M) \leq m_i \leq M$  holds. This is an important and key technique to classify all tasks into either critical or non-critical.

### 5. Numerical Example

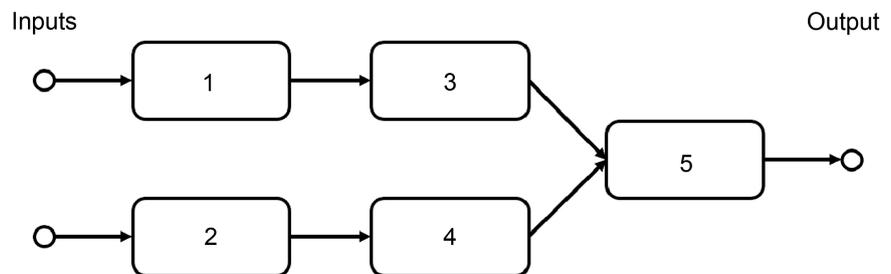
A numerical example is presented to validate the developed framework. We use a personal computer with the following execution environment:

- machine: Dell Optiplex 9020;
- CPU: Intel® Core™ i7-4790 3.60 GHz;
- OS: Microsoft Windows 7 Professional;
- memory: 4.0 GB.

To solve CSPs, we use SCIP version 3.2.1. **Figure 1** is a simple manufacturing system with five processes, two inputs, and one output, which is the same as taken in reference [1]. Each node represents a task having a processing time equal to the node number. We feed raw materials to inputs 1 and 2 at  $t = 3$  and  $t = 0$ , respectively.

$$\mathbf{d} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 4 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 5 & 5 & \varepsilon \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 4 \\ 2 \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} \bar{\varepsilon} \\ \bar{\varepsilon} \\ \bar{\varepsilon} \\ \bar{\varepsilon} \\ 7 \end{bmatrix}.$$

The constant vectors and a matrix,  $\mathbf{d}$ ,  $\mathbf{A}$ ,  $\mathbf{b}$ , and  $\mathbf{c}$ , reflect the processing times, precedence constraints, locations of the external inputs, and locations of the external outputs, the definitions of which appear in Section 4.1. Since the solver cannot treat the zero element  $\varepsilon$ , we need to set the two constants big-M and zero element  $\varepsilon$  carefully. In accordance with the procedure in reference [1], we set the constants to  $M = 70$  and  $\varepsilon = 20$ . The resulted values from the solver are as follows: the earliest task-completion times:  $\mathbf{x}_E = [4 \ 2 \ 7 \ 6 \ 12]^T$ , the earliest output time:  $y_E = 12$ , the latest task-starting times:  $\mathbf{x}_L = [3 \ 1 \ 4 \ 3 \ 7]^T$ ,



**Figure 1.** A simple manufacturing system with five processes (the same as [1]).

the latest input times:  $\mathbf{u}_L = [3 \ 1]^T$ , the total floats:  $\mathbf{m} = [0 \ 1 \ 0 \ 1 \ 0]^T$ , and the criticalities of all tasks:  $\boldsymbol{\alpha} = [0 \ 1 \ 0 \ 1 \ 0]^T$ . We hence were able to obtain solutions by reducing the MPL equations to CSPs for MIP.

## 6. Conclusion

This research has developed a solution method for reducing a project scheduling problem represented by a max-plus-linear form. The resulting formulation was constraint satisfaction problems for mixed-integer programming. We attained calculating the earliest output times, latest task-starting times, latest input times, and critical tasks. Moreover, we also developed a method to classify all tasks into either critical or non-critical by introducing a small positive constant  $(1/M)$ . Our future works include a proper setting of the small positive constant as well as considering resource contentions.

## References

- [1] Goto, H. (2017) Reduction of Max-Plus Algebraic Equations to Constraint Satisfaction Problems for Mixed Integer Programming. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science*, **E100-A**, 427-430.
- [2] Heidergott, B., Olsder, G.J. and Woude, L. (2006) Max Plus at Work: Modeling and Analysis of Synchronized Systems. Princeton University Press, New Jersey.
- [3] Baccelli, F., Cohen, G., Older, G.J., and Quadrat, J.P. (1992) Synchronization and Linearity. An Algebra for Discrete Event Systems, John Wiley & Sons, New York.
- [4] Goto, H., and Masuda, S. (2004) On the Properties of the Greatest Subsolution for Linear Equations in the Max-Plus Algebra. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science*, **E87-A**, 424-432.
- [5] Goto, H. (2013) Address a Project Scheduling Problem using Max-Plus Algebra. *Journal of the Society of Instrument and Control Engineers*, **52**, 1083-1089.
- [6] Yokoyama, H. and Goto, H. (2016) Resolution of Resource Contentions in the CCPM-MPL Using Simulated Annealing and Genetic Algorithm. *American Journal of Operations Research*, **6**, 480-488. <https://doi.org/10.4236/ajor.2016.66044>



**Submit or recommend next manuscript to SCIRP and we will provide best service for you:**

Accepting pre-submission inquiries through Email, Facebook, LinkedIn, Twitter, etc.

A wide selection of journals (inclusive of 9 subjects, more than 200 journals)

Providing 24-hour high-quality service

User-friendly online submission system

Fair and swift peer-review system

Efficient typesetting and proofreading procedure

Display of the result of downloads and visits, as well as the number of cited articles

Maximum dissemination of your research work

Submit your manuscript at: <http://papersubmission.scirp.org/>

Or contact [ajor@scirp.org](mailto:ajor@scirp.org)