Online MPL Scheduling of Backward Type for Repetitive Systems with MIMO-FIFO Structure

HIROYUKI GOTO, MASARU ONUMA and KAZUHIRO YAMADA
Department of Management and Information Systems Engineering
Nagaoka University of Technology
1603-1 Kamitomiokamachi, Nagaoka, Niigata 940-2188
JAPAN

Abstract: - This paper proposes an online scheduling method considering no-concurrency of resource with subsequent event. We focus on repetitive discrete event systems with MIMO (Multiple Inputs and Multiple Outputs)-FIFO structure. The behavior of such kind of systems can be described by linear equations in max-plus algebra, which is also called the MPL (Max-Plus Linear) systems. Conventional MPL representation formulates no-concurrency of resource with previous event and can provide the earliest starting times in internal facilities. We recently proposed a calculation method for a single job, however, the resource no-concurrency with the subsequent job is not considered. Thus, it may not give an optimal solution if a large number of jobs have to be handled. Therefore, we derive a general form of MPL representation that provides the latest starting times considering the no-concurrency with subsequent event. Also, we consider an effective rescheduling method which is applicable for cases where the system parameters are changed after the job commencement.

Key-Words: - Max-plus linear system, repetitive system, state-space representation, MIMO, earliest/latest time, rescheduling, forward/backward type

1 Introduction

There is an approach based on MPL representation as modeling and analysis for discrete event systems with MIMO-FIFO structure. It describes behaviors of systems with linear equations in max-plus algebra [1], [2] on which 'max' and '+' operations are defined as addition and multiplication, respectively. The equations are similar to the state-space representation in modern control theory. In the MPL representation, systems with synchronization of multiple events, no-concurrency of resources or repeatability can be stated simply [3].

Reference [4] revealed that the conventional MPL representation is equivalent to PERT in the field of OR and the maximum value of the state variables by which the same output value is obtained to the latest starting time. It also proposed a method for calculating the corresponding latest starting time and finding bottleneck processes. However, ref. [4] considers the no-concurrency of resource with previous event only. Since many existing production or transportation systems process or transport repeatedly with identical resources [5], [6], it is desirable to formulate or optimize systems by considering the no-concurrency of resource with subsequent job.

Therefore, this paper considers and proposes a MPL state-space representation taking into account the no-concurrency with subsequent job. Since the proposed method represents the internal states by those of the subsequent job, we call this as 'backward representation'.

2 Conventional MPL Representation

The conventional MPL representation taking into account the no-concurrency with previous event is briefly reviewed. For a preparation, relevant operators based on max-plus algebra are defined.

2.1 Basic operators

Denote the real field by $\boldsymbol{R}$. Max-plus algebra is an algebraic system in which 'max' and '+' operations are defined as addition and multiplication. In $\mathcal{D} = \boldsymbol{R} \cup \{-\infty\}$, define the following four operators.

$$x \oplus y = \max(x, y), \; x \otimes y = x + y$$
$$x \wedge y = \min(x, y), \; x \setminus y = -x + y$$

As in conventional algebra, the symbol for multiplication $\otimes$ is suppressed when no confusion is likely to arise. Operators for multiple terms are, if $m \le n$,

$$\bigoplus_{k=m}^{n} x_k = \max(x_m, \cdots, x_n) \quad , \quad \bigwedge_{k=m}^{n} x_k = \min(x_m, \cdots, x_n)$$

For matrices, if $\boldsymbol{X} \in \mathcal{D}^{m\times n}$, $[\boldsymbol{X}]_{ij}$ expresses the (i, j) -th element and $\boldsymbol{X}^T$ is the transposed matrix. For $\boldsymbol{X}, \boldsymbol{Y} \in \mathcal{D}^{m\times n}$,

$$[\boldsymbol{X} \oplus \boldsymbol{Y}]_{ij} = \max([\boldsymbol{X}]_{ij}, [\boldsymbol{Y}]_{ij}) \quad , \quad [\boldsymbol{X} \wedge \boldsymbol{Y}]_{ij} = \min([\boldsymbol{X}]_{ij}, [\boldsymbol{Y}]_{ij})$$

If $\boldsymbol{X} \in \mathcal{D}^{m\times l}$, $\boldsymbol{Y} \in \mathcal{D}^{l\times p}$,

$$[\boldsymbol{X} \otimes \boldsymbol{Y}]_{ij} = \bigoplus_{k=1}^{l}\left([\boldsymbol{X}]_{ik} \otimes [\boldsymbol{Y}]_{kj}\right) = \max_{k=1,\cdots,l}([\boldsymbol{X}]_{ik} + [\boldsymbol{Y}]_{kj})$$

$$[\boldsymbol{X} \odot \boldsymbol{Y}]_{ij} = \bigwedge_{k=1}^{l}\left([\boldsymbol{X}]_{ik} \setminus [\boldsymbol{Y}]_{kj}\right) = \min_{k=1,\cdots,l}(-[\boldsymbol{X}]_{ik} + [\boldsymbol{Y}]_{kj})$$

Priorities of operators $\otimes$, $\setminus$ and $\odot$ are higher than $\oplus$ and $\wedge$. These follow the next operation rules [2], [4]. If $\boldsymbol{X}, \boldsymbol{Y} \in \mathcal{D}^{l\times m}$, $\boldsymbol{Z} \in \mathcal{D}^{m\times l}$, $\boldsymbol{v}, \boldsymbol{w} \in \mathcal{D}^{m}$,

$$(\boldsymbol{X} \oplus \boldsymbol{Y}) \odot \boldsymbol{v} = (\boldsymbol{X} \odot \boldsymbol{v}) \wedge (\boldsymbol{Y} \odot \boldsymbol{v}) \tag{1}$$

$$\boldsymbol{X} \odot (\boldsymbol{v} \wedge \boldsymbol{w}) = (\boldsymbol{X} \odot \boldsymbol{v}) \wedge (\boldsymbol{X} \odot \boldsymbol{w}) \tag{2}$$

$$\boldsymbol{Y}^T \odot (\boldsymbol{Z}^T \odot \boldsymbol{v}) = (\boldsymbol{Z}\boldsymbol{Y})^T \odot \boldsymbol{v} \tag{3}$$

We denote the unit elements for $\oplus$, $\otimes$ and $\wedge$ by $\varepsilon\,(= -\infty)$, $e\,(= 0)$ and $\mathsf{T}\,(= +\infty)$, respectively. For matrices, let them be denoted as follows.

- $\boldsymbol{\varepsilon}_{mn}$: All elements are ε in $\boldsymbol{\varepsilon}_{mn} \in \mathcal{D}^{m\times n}$.
- $\boldsymbol{e}_m$: All diagonal elements are e and all off-diagonal elements are ε in $\boldsymbol{e}_m \in \mathcal{D}^{m\times m}$.
- T_{mn} : All elements are T in $\mathsf{T}_{mn} \in \mathcal{D}^{m\times n}$.

2.2 Max-plus linear system

We assume a production system in which all execution sequences are predetermined and the system has the following features.

- The number of facilities, external inputs and external outputs are n, p and q , respectively.
- Every jobs use all facilities and only once for each.
- Transportation times from external input to internal facility, between facilities and from internal facility to external output can be ignored, and the buffer capacities between these paths are infinite.

Additionally, assume the following constraints regarding the manufacturing sequences are imposed.

- While a facility is at work, the next job cannot be commenced.
- Facilities which have preceding ones cannot start manufacturing until the manufacturing in these facilities are completed.
- Facilities which have external inputs cannot start manufacturing until all required materials are supplied from these inputs.

Though we assume a production system as an example here, the same idea can also be applied to project management by regarding facility and job as work and project, respectively.

For the k -th job, denote the manufacturing starting time and the processing time in each facility by $[\boldsymbol{x}(k)]_i$ and $d_i(k)\,(\ge 0)$ $(1 \le i \le n)$, respectively. Moreover, the output time in each external output is denoted by $[\boldsymbol{y}(k)]_i$ $(1 \le i \le q)$, and set the initial condition to $\boldsymbol{x}(0) = \boldsymbol{\varepsilon}_{n1}$. In addition, define the following representation matrices $\boldsymbol{A}_k^0$, $\boldsymbol{F}_k$, $\boldsymbol{B}^0$ and $\boldsymbol{C}_k^0$ that specify structures of the system.

$$[\boldsymbol{A}_k^0]_{ij} = \begin{cases} d_i(k) & : \text{if } i = j \\ \varepsilon & : \text{otherwise} \end{cases}$$

$$[\boldsymbol{F}_k]_{ij} = \begin{cases} d_j(k) & : \text{if facility } i \text{ has a preceding facility } j \\ \varepsilon & : \text{if facility } i \text{ does not have preceding facilities} \end{cases}$$

$$[\boldsymbol{B}^0]_{ij} = \begin{cases} e & : \text{if facility } i \text{ has an external input } j \\ \varepsilon & : \text{if facility } i \text{ does not have external inputs} \end{cases}$$

$$[\boldsymbol{C}_k^0]_{ij} = \begin{cases} d_j(k) & : \text{if facility } j \text{ has an external output } i \\ \varepsilon & : \text{if facility } j \text{ does not have external outputs} \end{cases}$$

According to ref. [4], in facility i , $[\boldsymbol{A}_{k-1}^0 \boldsymbol{x}(k-1)]_i$, $[\boldsymbol{F}_k \boldsymbol{x}(k)]_i$ and $[\boldsymbol{B}_k^0 \boldsymbol{u}(k)]_i$ represent the manufacturing completion time of the previous job, the latest time among manufacturing completion times in preceding facilities and the latest time among feeding times from external inputs, respectively. Moreover, $[\boldsymbol{C}_k^0 \boldsymbol{x}(k)]_i$ states the latest time among manufacturing completion times in facilities attached to the i -th output. These results imply that the earliest starting times and output times in all facilities and external outputs can be summarized as shown below.

$$\boldsymbol{x}_E(k) = \boldsymbol{F}_k \boldsymbol{x}_E(k) \oplus \boldsymbol{A}_{k-1}^0 \boldsymbol{x}(k-1) \oplus \boldsymbol{B}_k^0 \boldsymbol{u}(k) \tag{4}$$

$$\boldsymbol{y}_E(k) = \boldsymbol{C}_k^0 \boldsymbol{x}(k) \tag{5}$$

where the suffix E stands for the earliest time. Moreover, eq. (4) can be transformed into the next form.

$$\boldsymbol{x}_E(k) = \boldsymbol{F}_k^* \boldsymbol{A}_{k-1}^0 \boldsymbol{x}(k-1) \oplus \boldsymbol{F}_k^* \boldsymbol{B}^0 \boldsymbol{u}(k) \tag{6}$$

where

$$\boldsymbol{F}_k^* = \boldsymbol{e}_n \oplus \boldsymbol{F}_k \oplus \boldsymbol{F}_k^2 \oplus \cdots \oplus \boldsymbol{F}_k^{l-1},\ \boldsymbol{F}_k^l = \boldsymbol{\varepsilon}_{nn} \tag{7}$$

and $^\exists l\ (1 \le l \le n)$ is an instance that depends on the precedence constraints of the system.

3 Backward Representation

A method for calculating the latest starting times and the earliest input times are considered. Since this method predicts the state variables iteratively by decreasing the event counter and giving the output times, we call this as 'backward representation'.

In obtaining the earliest starting times using eq. (6), a lower limit of predicted value of the state variables $\boldsymbol{x}(k)$ is considered by giving both the earliest manufacturing starting times of the previous job $\boldsymbol{x}(k-1)$ and the feeding times $\boldsymbol{u}(k)$. On the other hand, the latest starting times can be obtained by calculating the upper bound of the predicted value of the state variables $\boldsymbol{x}(k)$ with the estimated output times $\boldsymbol{y}(k)$ and the manufacturing starting times of the subsequent job $\boldsymbol{x}(k+1)$.

The constraints mentioned in 2.2 considers the no-concurrency with the previous job, which can be restated by taking into account the no-concurrency with the subsequent job in the following manner.

- The manufacturing completion time of the corresponding job would be equal or earlier than the manufacturing starting times in succeeding facilities.
- Facilities which have succeeding ones would complete manufacturing on or earlier than the manufacturing starting times in the facilities.
- Facilities which have external outputs would finish manufacturing on or earlier than the estimated output times.

We now consider deriving the latest manufacturing starting time of the k-th job in each facility. We utilize the same representation matrices $\boldsymbol{A}_k^0$, $\boldsymbol{F}_k$, $\boldsymbol{B}^0$ and $\boldsymbol{C}_k^0$, and the parameter for the processing time $d_i(k)$ defined in 2.2. For facility i, $\mathcal{Q}_i$ and $\mathcal{S}_i$ expresses a number set of attached external outputs and succeeding facilities, respectively. By setting the latest manufacturing starting time to $[\boldsymbol{x}_L(k)]_i$, the latest completion time in each facility can be formulated as follows.

$$\begin{aligned} &[\boldsymbol{x}_L(k)]_i + d_i(k) \\ &\quad = [\boldsymbol{x}(k+1)]_i \wedge (\bigwedge_{j \in \mathcal{S}_i} [\boldsymbol{x}_L(k)]_j) \wedge (\bigwedge_{j \in \mathcal{Q}_i} [\boldsymbol{y}(k)]_j) \end{aligned} \tag{8}$$

The first term of the right side represents the manufacturing starting time of the subsequent job. The second and third terms represent the manufacturing starting times in the succeeding facilities and the estimated output times in facilities with external outputs, respectively. By solving eq. (8) for $[\boldsymbol{x}_L(k)]_i$, the next relationship is obtained.

$$\begin{aligned} &[\boldsymbol{x}_L(k)]_i = ([\boldsymbol{x}(k+1)]_i - d_i(k)) \\ &\wedge (\bigwedge_{j \in \mathcal{S}_i} [\boldsymbol{x}_L(k)]_j - d_i(k)) \wedge (\bigwedge_{j \in \mathcal{Q}_i} [\boldsymbol{y}(k)]_j - d_i(k)) \\ &= ([\boldsymbol{x}(k+1)]_i - [\boldsymbol{A}_k^0]_{ii}) \wedge (\bigwedge_{j=1}^{n} [\boldsymbol{x}_L(k)]_j - [\boldsymbol{F}_k]_{ji}) \\ &\qquad \wedge (\bigwedge_{j=1}^{q} [\boldsymbol{y}(k)]_j - [\boldsymbol{C}_k^0]_{ji}) \\ &= [\boldsymbol{A}_k^{0^T} \odot \boldsymbol{x}(k+1)]_i \wedge [\boldsymbol{F}_k^T \odot \boldsymbol{x}_L(k)]_i \\ &\qquad \wedge [\boldsymbol{C}_k^{0^T} \odot \boldsymbol{y}(k)]_i \end{aligned} \tag{9}$$

Note that $\boldsymbol{x}(k+1)$ should not necessarily identical to the latest starting times $\boldsymbol{x}_L(k+1)$. Since eq. (9) holds true for all i $(1 \le i \le n)$, the latest starting times of all processes $\boldsymbol{x}_L(k)$ can be represented as follows.

$$\begin{aligned} &\boldsymbol{x}_L(k) \\ &\quad = \boldsymbol{F}_k^T \odot \boldsymbol{x}_L(k) \wedge \boldsymbol{A}_k^{0^T} \odot \boldsymbol{x}(k+1) \wedge \boldsymbol{C}_k^{0^T} \odot \boldsymbol{y}(k) \end{aligned} \tag{10}$$

Next, consider simplifying eq. (10). With the help of eqs. (1)-(3), substitute the entire right hand-side of eq. (10) for the first term $\boldsymbol{x}_L(k)$. This leads

$$\begin{aligned} &\boldsymbol{x}_L(k) = \{\boldsymbol{F}_k^T \odot [\boldsymbol{F}_k^T \odot \boldsymbol{x}_L(k) \wedge \boldsymbol{A}_k^{0^T} \odot \boldsymbol{x}(k+1) \\ &\quad \wedge \boldsymbol{C}_k^{0^T} \odot \boldsymbol{y}(k)]\} \wedge \boldsymbol{A}_k^{0^T} \odot \boldsymbol{x}(k+1) \wedge \boldsymbol{C}_k^{0^T} \odot \boldsymbol{y}(k) \\ &= \cdots \\ &= \boldsymbol{F}_k^{2^T} \odot \boldsymbol{x}_L(k) \wedge [\boldsymbol{A}_k^0(\boldsymbol{e}_n \oplus \boldsymbol{F}_k)]^T \odot \boldsymbol{x}(k+1) \\ &\qquad \wedge [\boldsymbol{C}_k^0(\boldsymbol{e}_n \oplus \boldsymbol{F}_k)]^T \odot \boldsymbol{y}(k) \end{aligned}$$

Repeating the same procedure derives the following simplified expression.

$$\begin{aligned} \boldsymbol{x}_L(k) &= \boldsymbol{F}_k^{3^T} \odot \boldsymbol{x}_L(k) \\ &\quad \wedge [\boldsymbol{A}_k^0(\boldsymbol{e}_n \oplus \boldsymbol{F}_k \oplus \boldsymbol{F}_k^2)]^T \odot \boldsymbol{x}(k+1) \\ &\quad \wedge [\boldsymbol{C}_k^0(\boldsymbol{e}_n \oplus \boldsymbol{F}_k \oplus \boldsymbol{F}_k^2)]^T \odot \boldsymbol{y}(k) \qquad (11) \\ &= \cdots \\ &= (\boldsymbol{A}_k^0 \boldsymbol{F}_k^*)^T \odot \boldsymbol{x}(k+1) \wedge (\boldsymbol{C}_k^0 \boldsymbol{F}_k^*)^T \odot \boldsymbol{y}(k) \end{aligned}$$

where $\boldsymbol{F}_k^*$ is the same as eq. (7).

Furthermore, a relationship between the latest input time and the state variable is obtained. Designating $\mathcal{W}_i$ as a number collection of processes attached to the i-th external input, the input variable $[\boldsymbol{u}_L(k)]_j$ can be represented as follows.

$$\begin{aligned} [\boldsymbol{u}_L(k)]_i &= \bigwedge_{j \in \mathcal{W}_i} [\boldsymbol{x}(k)]_j = \bigwedge_{j=1}^{n} ([\boldsymbol{x}(k)]_j - [\boldsymbol{B}^0]_{ji}) \\ &= \bigwedge_{j=1}^{n} ([\boldsymbol{B}^{0^T}]_{ij} \setminus [\boldsymbol{x}(k)]_j) = [\boldsymbol{B}^{0^T} \odot \boldsymbol{x}(k)]_i \end{aligned}$$

Since this holds true for all i $(1 \le i \le p)$, the latest input times $\boldsymbol{u}_L(k)$ can be obtained using the latest manufacturing starting times $\boldsymbol{x}_L(k)$, in the following manner.

$$\boldsymbol{u}_L(k) = \boldsymbol{B}^{0^T} \odot \boldsymbol{x}_L(k)$$

As these matters indicate, we obtained the MPL state-space representation taking into account the no-concurrency of resource with the subsequent job. In summarize, the MPL state-space representation of backward type consists the following two equations.

$$\boldsymbol{x}_L(k) = (\boldsymbol{A}_k^0 \boldsymbol{F}_k^*)^T \odot \boldsymbol{x}(k+1) \wedge (\boldsymbol{C}_k^0 \boldsymbol{F}_k^*)^T \odot \boldsymbol{y}(k) \qquad (12)$$

$$\boldsymbol{u}_L(k) = \boldsymbol{B}^{0^T} \odot \boldsymbol{x}_L(k) \qquad (13)$$

Equation (12) means that the latest manufacturing starting times are stated by the internal states of the subsequent job $\boldsymbol{x}(k+1)$ and the estimated output times $\boldsymbol{y}(k)$. If the manufacturing of the subsequent job starts sufficiently after the current job is completed, the state vector can be set to

$$\boldsymbol{x}(k+1) = \mathsf{T}_{n1}$$

and eq. (12) be simplified to

$$\boldsymbol{x}_L(k) = (\boldsymbol{C}_k^0 \boldsymbol{F}_k^*)^T \odot \boldsymbol{y}(k)$$

This is equivalent to the result in ref. [3] where the no-concurrency of resource with subsequent job is not considered. Furthermore, eq. (13) yields

$$\begin{aligned} \boldsymbol{u}_L(k) &= \boldsymbol{B}^{0^T} \odot [(\boldsymbol{C}_k^0 \boldsymbol{F}_k^*)^T \odot \boldsymbol{y}(k)] \\ &= (\boldsymbol{C}_k^0 \boldsymbol{F}_k^* \boldsymbol{B}^0)^T \odot \boldsymbol{y}(k) \end{aligned}$$

and is identical to ref. [4] which proposes a method for calculating the latest input times by which the manufacturing can be completed on the desired time.

As these issues imply, the backward MPL representation in this paper can be understood as a general extension of the previous researches.

4 Online Scheduling

In the previous section we assumed a situation where the system operates entirely along to the predetermined schedule. However, in practical cases, there may often be cases where the arrival time of material is delayed or the due date is put forward. Such kinds of changes would vary the relevant systems parameters, float times or location of bottleneck processes. Hence, it is essential to develop a rescheduling method which can be applied even when the job has commenced. This paper thus proposes a method for the backward MPL representation.

4.1 Rescheduling for backward type

Assume the relevant parameters such as estimated output time, manufacturing starting time or processing time is changed for some reason after the job commencement. We denote the changed variables by appending a symbol $[\tilde{\ }]$ as follows.

$$[\tilde{\boldsymbol{y}}]_i \ (1 \le i \le q),\ [\tilde{\boldsymbol{x}}]_i \ (1 \le i \le n)$$

where the event counter (k) is abbreviated for simplicity. If the processing times are changed, they then influence on the system, output and adjacency matrices.

$$\tilde{\boldsymbol{A}}^0,\ \tilde{\boldsymbol{C}}^0,\ \tilde{\boldsymbol{F}}$$

where the suffix k for representing the event counter is omitted again. Note that the input matrix $\boldsymbol{B}^0$ is invariant concerning its definition. Utilizing the above variables and representation matrices, we recalculate the changed state variables. First, set

$$[\tilde{\boldsymbol{x}}^{(0)}]_i = \mathsf{T} \ (= \infty) \qquad (14)$$

for all facilities i which require recalculation. The upper suffix (0) stands for a initial value for recalculating the state variable iteratively. Facilities which do not require recalculation mean that they are already commenced or their starting times are fixed. In a similar way to eq. (9), the latest starting time $[\tilde{\boldsymbol{x}}^{(1)}]_i$ in facility i can be represented in the next manner.

$$[\tilde{\boldsymbol{x}}^{(1)}]_i = [\tilde{\boldsymbol{A}}^{0^T} \odot \boldsymbol{x}(k+1)]_i \wedge [\tilde{\boldsymbol{F}}^T \odot \tilde{\boldsymbol{x}}^{(0)}]_i \wedge [\tilde{\boldsymbol{C}}^{0^T} \odot \tilde{\boldsymbol{y}}]_i \quad (15)$$

where facility i is located in lowermost stream whose all facilities in downstream do not require recalculation. There may be multiple elements. In a similar way to eq. (9), the latest starting time in upstream facilities are calculated iteratively as follows.

$$\begin{aligned}
\tilde{\boldsymbol{x}}^{(2)} &= \tilde{\boldsymbol{A}}^{0^T} \odot \boldsymbol{x}(k+1) \wedge \tilde{\boldsymbol{F}}^T \odot \tilde{\boldsymbol{x}}^{(1)} \wedge \tilde{\boldsymbol{C}}^{0^T} \odot \tilde{\boldsymbol{y}} \\
&= [\tilde{\boldsymbol{A}}^0 (\boldsymbol{e}_n \oplus \tilde{\boldsymbol{F}})]^T \odot \boldsymbol{x}(k+1) \wedge \tilde{\boldsymbol{F}}^{2^T} \odot \tilde{\boldsymbol{x}}^{(0)} \\
\vdots &\qquad \wedge [\tilde{\boldsymbol{C}}^0 (\boldsymbol{e}_n \oplus \tilde{\boldsymbol{F}})]^T \odot \tilde{\boldsymbol{y}} \\
\tilde{\boldsymbol{x}}^{(l-1)} &= \tilde{\boldsymbol{A}}^{0^T} \odot \boldsymbol{x}(k+1) \wedge \tilde{\boldsymbol{F}}^T \odot \tilde{\boldsymbol{x}}^{(l-2)} \wedge \tilde{\boldsymbol{C}}^{0^T} \odot \tilde{\boldsymbol{y}} \\
&= [\tilde{\boldsymbol{A}}^0 (\boldsymbol{e}_n \oplus \tilde{\boldsymbol{F}} \oplus \cdots \oplus \tilde{\boldsymbol{F}}^{l-2})]^T \odot \boldsymbol{x}(k+1) \\
&\quad \wedge \tilde{\boldsymbol{F}}^{l-1^T} \odot \tilde{\boldsymbol{x}}^{(0)} \\
&\quad \wedge [\tilde{\boldsymbol{C}}^0 (\boldsymbol{e}_n \oplus \tilde{\boldsymbol{F}} \oplus \cdots \oplus \tilde{\boldsymbol{F}}^{l-2})]^T \odot \tilde{\boldsymbol{y}} \\
\tilde{\boldsymbol{x}}^{(l)} &= (\tilde{\boldsymbol{A}}^0 \tilde{\boldsymbol{F}}^*)^T \odot \boldsymbol{x}(k+1) \wedge (\tilde{\boldsymbol{C}}^0 \tilde{\boldsymbol{F}}^*)^T \odot \tilde{\boldsymbol{y}}
\end{aligned} \quad (16)$$

where

$$\tilde{\boldsymbol{F}}^* = \boldsymbol{e}_n \oplus \tilde{\boldsymbol{F}} \oplus \cdots \oplus \tilde{\boldsymbol{F}}^{l-1},\ \tilde{\boldsymbol{F}}^l = \boldsymbol{\varepsilon}_{nn}$$

and the element number $[\cdot]_i$ is abbreviated for simplicity. Utilizing eqs. (15) and (16), the latest manufacturing starting times which require recalculation are determined iteratively towards upstream. Recalling eq. (14), the manufacturing starting times which require recalculation are set to T. Thus, the latest starting times of all facilities $\tilde{\boldsymbol{x}}_L$ for the rescheduling can be represented in the following way.

$$\tilde{\boldsymbol{x}}_L = \tilde{\boldsymbol{x}}^{(0)} \wedge \tilde{\boldsymbol{x}}^{(1)} \wedge \cdots \wedge \tilde{\boldsymbol{x}}^{(l)}$$

Utilizing eqs. (15) and (16), the coefficients of $\boldsymbol{x}(k+1)$, $\tilde{\boldsymbol{x}}^{(0)}$ and $\tilde{\boldsymbol{y}}$ for $\tilde{\boldsymbol{x}}_L$ are simplified as follows.

$$\begin{aligned}
&\tilde{\boldsymbol{A}}^{0^T} \odot \boldsymbol{x}(k+1) \wedge [\tilde{\boldsymbol{A}}^0 (\boldsymbol{e}_n \oplus \tilde{\boldsymbol{F}})]^T \odot \boldsymbol{x}(k+1) \wedge \\
&\qquad \cdots \wedge [\tilde{\boldsymbol{A}}^0 (\boldsymbol{e}_n \oplus \tilde{\boldsymbol{F}} \oplus \cdots \oplus \tilde{\boldsymbol{F}}^{l-1})]^T \odot \boldsymbol{x}(k+1) \\
&\quad = (\tilde{\boldsymbol{A}}^0 \tilde{\boldsymbol{F}}^*)^T \odot \boldsymbol{x}(k+1)
\end{aligned}$$

$$\begin{aligned}
&\tilde{\boldsymbol{x}}^{(0)} \wedge \tilde{\boldsymbol{F}}^T \odot \tilde{\boldsymbol{x}}^{(0)} \wedge \cdots \wedge \tilde{\boldsymbol{F}}^{l-1^T} \odot \tilde{\boldsymbol{x}}^{(0)} \\
&\quad = \tilde{\boldsymbol{F}}^{*T} \odot \tilde{\boldsymbol{x}}^{(0)}
\end{aligned}$$

$$\begin{aligned}
&\tilde{\boldsymbol{C}}^{0^T} \odot \tilde{\boldsymbol{y}}(k) \wedge [\tilde{\boldsymbol{C}}^0 (\boldsymbol{e}_n \oplus \tilde{\boldsymbol{F}})]^T \odot \tilde{\boldsymbol{y}} \wedge \\
&\qquad \cdots \wedge [\tilde{\boldsymbol{C}}^0 (\boldsymbol{e}_n \oplus \tilde{\boldsymbol{F}} \oplus \cdots \oplus \tilde{\boldsymbol{F}}^{l-1})]^T \odot \tilde{\boldsymbol{y}} \\
&\quad = (\tilde{\boldsymbol{C}}^0 \tilde{\boldsymbol{F}}^*)^T \odot \tilde{\boldsymbol{y}}
\end{aligned}$$

Therefore, the latest starting times of all facilities $\tilde{\boldsymbol{x}}_L(k)$ are obtained as shown below.

$$\begin{aligned}
\tilde{\boldsymbol{x}}_L(k) = (\tilde{\boldsymbol{A}}_k^0 \tilde{\boldsymbol{F}}_k^*)^T \odot \boldsymbol{x}(k+1) \wedge (\tilde{\boldsymbol{C}}_k^0 \tilde{\boldsymbol{F}}_k^*)^T \odot \tilde{\boldsymbol{y}}(k) \\
\wedge \tilde{\boldsymbol{F}}_k^{*T} \odot \tilde{\boldsymbol{x}}^{(0)}(k)
\end{aligned} \quad (17)$$

The above equation is a general form that is applicable for cases where the manufacturing starting times and/or processing times are changed and they may either smaller or greater than their original values. However, this form has also a disadvantage that all terms must be recalculated even if a single parameter is changed, which it may be undesirable in terms of calculation efficiency. Hence, in the next subsection, we consider a simple and efficient calculation method which can be applied for cases that are often common in practical systems.

4.2 Moving up the schedule

We consider a simple rescheduling method for cases where the manufacturing starting times or estimated output times are put forward whereas the processing times are invariant. First, consider a case where a certain manufacturing starting time has moved up from the initial value $\boldsymbol{x}_L$. For $\tilde{\boldsymbol{x}}^{(0)}$, we update elements whose manufacturing starting times are changed and fixed, and keep the original values for other facilities. Hence,

$$[\tilde{\boldsymbol{x}}^{(0)}]_i \le [\boldsymbol{x}_L]_i \text{ for all } i \ (1 \le i \le n) \quad (18)$$

is followed and the representation matrices are invariant as shown below.

$$\tilde{\boldsymbol{A}}^0 = \boldsymbol{A}^0,\ \tilde{\boldsymbol{C}}^0 = \boldsymbol{C}^0,\ \tilde{\boldsymbol{F}}^* = \boldsymbol{F}^* \quad (19)$$

Hence, $\tilde{\boldsymbol{x}}_L$ in eq. (17) can be obtained using eqs. (12) and (19) as follows.

$$\begin{aligned}
\tilde{\boldsymbol{x}}_L &= (\boldsymbol{A}^0 \boldsymbol{F}^*)^T \odot \boldsymbol{x}(k+1) \wedge (\boldsymbol{C}^0 \boldsymbol{F}^*)^T \odot \boldsymbol{y} \\
&\qquad \wedge \boldsymbol{F}^{*T} \odot \tilde{\boldsymbol{x}}^{(0)} \\
&= \boldsymbol{x}_L \wedge \boldsymbol{F}^{*T} \odot \tilde{\boldsymbol{x}}^{(0)}
\end{aligned}$$

Reference [3] proves that the diagonal elements of $\boldsymbol{F}^*$ are $e(=0)$. Thus,

$$[\boldsymbol{F}^{*T} \odot \tilde{\boldsymbol{x}}^{(0)}]_i = \bigwedge_{m=1}^{n} (-[\boldsymbol{F}^*]_{mi} + [\tilde{\boldsymbol{x}}^{(0)}]_m)$$
$$\leq -[\boldsymbol{F}^*]_{ii} + [\tilde{\boldsymbol{x}}^{(0)}]_i \leq -[\boldsymbol{F}^*]_{ii} + [\boldsymbol{x}_L]_i = [\boldsymbol{x}_L]_i$$

Since this holds true for all i $(1 \leq i \leq n)$, the following formula is obtained for calculating $\tilde{\boldsymbol{x}}_L(k)$.

$$\tilde{\boldsymbol{x}}_L(k) = \boldsymbol{F}_k^{*T} \odot \tilde{\boldsymbol{x}}^{(0)}(k) \tag{20}$$

Next, consider a case where a due date is put forward. It follows

$$[\tilde{\boldsymbol{y}}]_i \leq [\boldsymbol{y}]_i \text{ for all } i \ (1 \leq i \leq q)$$

and eq. (19) is also satisfied. Since all manufacturing starting times are not recalculated at this stage, set $\tilde{\boldsymbol{x}}^{(0)} = \boldsymbol{x}_L$. Utilizing eqs. (12) and (1)-(3), eq. (17) can be simplified in the following manner.

$$\begin{aligned}
\tilde{\boldsymbol{x}}_L &= (\boldsymbol{A}^0\boldsymbol{F}^*)^T \odot \boldsymbol{x}(k+1) \wedge \boldsymbol{F}^{*T} \odot \boldsymbol{x}_L \\
&\qquad \wedge (\boldsymbol{C}^0\boldsymbol{F}^*)^T \odot \tilde{\boldsymbol{y}} \\
&= (\boldsymbol{A}^0\boldsymbol{F}^*)^T \odot \boldsymbol{x}(k+1) \wedge (\boldsymbol{C}^0\boldsymbol{F}^*)^T \odot \tilde{\boldsymbol{y}} \\
&\quad \wedge \boldsymbol{F}^{*T} \odot [(\boldsymbol{A}^0\boldsymbol{F}^*)^T \odot \boldsymbol{x}(k+1) \wedge (\boldsymbol{C}^0\boldsymbol{F}^*)^T \odot \boldsymbol{y}] \\
&= (\boldsymbol{A}^0\boldsymbol{F}^* \oplus \boldsymbol{A}^0\boldsymbol{F}^*\boldsymbol{F}^*)^T \odot \boldsymbol{x}(k+1) \\
&\qquad \wedge (\boldsymbol{C}^0\boldsymbol{F}^*\boldsymbol{F}^*)^T \odot \boldsymbol{y} \wedge (\boldsymbol{C}^0\boldsymbol{F}^*)^T \odot \tilde{\boldsymbol{y}} \\
&= (\boldsymbol{A}^0\boldsymbol{F}^*)^T \odot \boldsymbol{x}(k+1) \wedge (\boldsymbol{C}^0\boldsymbol{F}^*)^T \odot \boldsymbol{y} \\
&\qquad \wedge (\boldsymbol{C}^0\boldsymbol{F}^*)^T \odot \tilde{\boldsymbol{y}} \\
&= \boldsymbol{x}_L \wedge (\boldsymbol{C}^0\boldsymbol{F}^*)^T \odot \tilde{\boldsymbol{y}}
\end{aligned} \tag{21}$$

where we utilized the next relationship.

$$\boldsymbol{F}^*\boldsymbol{F}^* = \boldsymbol{F}^*$$

Equation (20) means that the rescheduling can be accomplished only by operating $\boldsymbol{F}^{*T} \odot$ to the updated state variables $\tilde{\boldsymbol{x}}^{(0)}$ if one or more manufacturing starting times are put forward. Moreover, eq. (21) means that the updated manufacturing starting times can be calculating with a low computation load only if only the due date is moved up.

5. Conclusion

This paper derived a MPL representation of backward type taking into account the no-concurrency of resource with subsequent event and considered a method for obtaining the latest starting and input times. The relevant researches have focused on MPL representation of forward type that considers the no-concurrency with previous job. Thus, they often supplied a schedule which causes a delay for the due date if the schedules of corresponding and subsequent jobs are close together. On the other hand, since this research takes the no-concurrency with the subsequent job into consideration, such delays do not occur. Later, we proposed a rescheduling method for cases where the relevant parameters are changed after the job commencement. Moreover, we proposed a simplified method that is applicable for cases where the state variables or output values are lessened from their original ones.

The no-concurrency between previous or subsequent job can be taken into account in this paper. However, several practical systems process many jobs simultaneously, by lot unit for instance. In such cases the conventional or proposed method may not provide an effective schedule. It is required to involve an idea of model predictive control (MPC) to cope with this problem. This extension remains as a future work.

References:

[1] B. Heidergott, G. J. Olsder, and L. Woude, *Max Plus at Work: Modeling and Analysis of Synchronized Systems*, Princeton University Press, New Jersey, 2006.

[2] F. Baccelli, G. Cohen, G. J. Olsder, and J. P. Quadrat, *Synchronization and Linearity*, John Wiley & Sons, New York, 1992.

[3] G. Cohen, P. Moller, J. Quadrat, and M. Viot, Algebraic Tools for the Performance Evaluation of Discrete Event Systems, *Proceeding of the IEEE*, Vol.77, No.1, 1989, pp.39-58.

[4] H. Goto and S. Masuda, Scheduling Method for MIMO-Type Discrete Event Systems Utilizing Max-Plus Linear Representation, *Proceedings of Asia Pacific Industrial Engineering and Management Systems (APIEMS)*, 2006, pp. 1029-1037.

[5] B. Heidergott and R. De Veries, Towards a (Max, +) Control Theory for Public Transportation Networks, *Discrete Event Dynamic Systems: Theory and Applications*, Vol.11, 2001, pp.371-398.

[6] B. Schutter, and T. Boom, Model Predictive Control for Max-Plus-Linear Systems, *Automatica*, Vol.37, No.7, 2001, pp.1049-1056.