Using Max-Plus Algebra in The Flow Shop Scheduling

Subiono[1] and Nur Shofianah[2]

Abstract —**In this paper, it is discussed notion of max-plus algebra and their properties. A model of flow shop production system and analyze the dynamical behavior of the system for scheduling problems are derived by means of max-plus algebra. The solutions of these problems are that the optimal sequence of jobs and the regular scheduling are obtained.**

Keywords— **Max - Plus Algebra, flow shop, scheduling.**

I. INTRODUCTION

Discrete event systems can be used to study processes that are driven by occurrence of events. The relevant variables represent times at which events take place. In recent year both industry and the academic world have become interested to model, analyze and control complex systems such as manufacturing systems, traffic control system, and telecommunication network and so on. This kind of systems is typical examples of discrete event systems. One of the most characteristic features of a discrete event system is that its dynamics are event-driven in contrast with time driven. The behavior of a discrete event system is governed by events rather than by ticks of a clock. An event corresponds to the start or the end of an activity. If we consider of a production system then the possible events are the completion of a part on a machine, a buffer become empty, a machine breakdown and so on. Event occurs at discrete time instants. Interval between events are not necessarily identical, it can be deterministic or stochastic.

An introduction to Discrete Event Systems can be found in [3], and the application of traffic light system in [12]. Although in general discrete event systems lead to a non linear description in conventional algebra, there exists a subclass of discrete event systems for which this model becomes linear when we formulate it in the max-plus algebra notion. In this paper, we focus on of subclass of discrete event systems that is called max-plus algebra. This subclass can be used to analyze behavior of a system. The standard on the max-plus algebra approach to discrete event systems can be found in [1], and some results work of max-plus algebra in [2]. Some real application of max-plus algebra to transportation system can be found in [4, 8, 11], an assembly production process system in [7], and flow shop production process

with *n* job 2 machines in [10]. In this paper, we discuss flow shop *production* process and modify the model such that we get both a regular scheduling and a minimal makespan.

The model of the system by using max-plus algebra approach is derived; this approach will have an advantage. The system will be linear and non linear in conventional algebra [6]. The linearity makes the analysis on the behavior of system easy. Further in the following section, a brief introduction to max-plus algebra notion and notation is given. And as a motivation, we give an example application of max-plus algebra that is related to a periodic scheduling. This periodic scheduling associate is with an eigenvalue and an eigenvector of a square matrix in max-plus algebra. We use *Max-Plus Algebra Toolbox for Scilab* [9] to compute the eigenvalue and corresponding eigenvector of a square matrix in max-plus algebra. As conclusion, we give some notes of the discussion and the future work.

II. MAX-PLUS ALGEBRA AND SOME RELATED NOTATION

This section presents a brief introduction to max plus algebra that would be employed in the following discussion. The element of max-plus algebra are real number and $\varepsilon = -\infty$. The set $R \cup \{\varepsilon\}$ will be denoted by R_{max} with R is the set of real number. The basic operations of Max-Plus Algebra are *maximization* (denoted by symbol $\oplus$) and *addition* (denoted by symbol $\otimes$). With these two operations for all $x, y \in R_{max}$, we get:

$x \oplus y = \max\{x, y\}$ and $x \otimes y = x + y$ and, in the context of Max-Plus Algebra

$a^{\otimes b} := ab, \ a \in R_{max}$ and $b \in R$

So, we get $3^{\otimes 4} = 4(3) = 12$ and $12^{\otimes \frac{1}{2}} = -\frac{1}{2}(12) = -6$.

Note that: for every $x \in R_{max}$ that satisfies $x \oplus \varepsilon = x = \varepsilon \oplus x$ and $x \otimes 0 = x = 0 \otimes x$.

As in the conventional algebra, it is well known that a matrix with real number entries and two operations addition and multiplication. In the max-plus algebra R_{max}, two operations addition and multiplication are denoted by two symbols respectively $\oplus$ and $\otimes$. For two matrices over R_{max}, the addition of matrices $A, B \in R_{max}^{m \times n}$, be given by:

$[A \oplus B]_{i,j} = a_{i,j} \oplus b_{i,j} = \max\{a_{i,j}, b_{i,j}\}$ with i = 1,2,...,m and j = 1,2,...,n. And multiplication of two matrices $A \in R_{max}^{m \times p}$ and $B \in R_{max}^{p \times n}$ be given by $[A \otimes B]_{i,j} = \bigoplus_{k=1}^{p} a_{i,k} \otimes b_{k,j} = \max_{1 \le k \le p} \{a_{i,k} + b_{k,j}\}$ with i = 1,2,...,m and j = 1,2,...,n.

---
[1]Subiono is with the Department of Mathematics, Faculty of Mathematics and Natural Science, Institut Teknologi Sepuluh Nopember, Surabaya, 60111, Indonesia.
E-mail:subiono2008@matematika.its.ac.id
[2]Nur Shofianah is Graduated Student of Master Program Mathematics Department, Faculty of Mathematics and Natural Science, Institut Teknologi Sepuluh Nopember, Surabaya, 60111, Indonesia.

It is shown that there is an analogy between $\oplus$ and $+$ and, $\otimes$ and $\times$. Therefore we choose the symbols $\oplus$ and $\otimes$. And multiplication of two matrices in max-plus algebra is similar to multiplication of two matrices in conventional algebra, i.e. two operations multiplication and addition in the conventional algebra and "maximization ($\oplus$)" and "addition ($\otimes$)" in the max-plus algebra as replacement of respectively addition and multiplication that are used in the conventional algebra. With these symbols, the description of state space of a system be given by equations as follow:

$$x(k+1) = A \otimes x(k) \oplus B \otimes u(k+1)$$
$$y(k) = C \otimes x(k) \tag{1}$$

with the size of the matrices in equation (1) satisfies addition and multiplication of matrices.

Like conventional algebra, an eigenvalue and a corresponding eigenvector of a square matrix A of size n x n also exists in max-plus algebra, i.e. if we give the equation

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

In this case vector $x \in R_{max}^n$ and scalar $\lambda \in R$ are respectively called *an eigenvector* and a corresponding *eigenvalue* of the matrix A with vector $x \neq (\varepsilon, \cdots, \varepsilon)'$. The sign $'$ represents transpose. An algorithm to compute an eigenvalue and a corresponding eigenvector of a square matrix A can be found in [5].

Example 1: Let be a matrix

$$A = \begin{pmatrix} 1 & 2 & \varepsilon \\ 4 & \varepsilon & 3 \\ \varepsilon & 5 & \varepsilon \end{pmatrix},$$ it follows that, $\begin{pmatrix} 1 & 2 & \varepsilon \\ 4 & \varepsilon & 3 \\ \varepsilon & 5 & \varepsilon \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 2 \\ 3 \end{pmatrix} = \begin{pmatrix} 4 \\ 6 \\ 7 \end{pmatrix} = 4 \otimes \begin{pmatrix} 0 \\ 2 \\ 3 \end{pmatrix}$

It is shown that the eigenvalue and corresponding eigenvector of matrix A are respectively $\lambda = 4$ and x = (0 2 3)^T. Let be given matrix $A \in R_{max}^{n \times n}$, a directed graph of matrix A is denoted by G(A) = (E,V). Graph G(A) has n nodes (vertices), the set of all nodes of graph G(A) is denoted by V. An arc (edge) from node j to i occurs if $a_{i,j} \neq \varepsilon$, this arc is denoted by (j,i). The set of all arcs of graph G(A) is denoted by E. The weight of arc (j,i) is value of $a_{i,j}$, this one is denoted by $w(j,i) = a_{i,j}$. If $a_{i,j} = \varepsilon$, then arc (j,i) does not exist. A sequence of arc (i_1,i_2), (i_2,i_3),..., (i_{l-1},i_l) of a graph is called a *path*. A path is called *elementer* if it nodes has only one incoming and one outgoing arc. A circuit is a close elementer path, i.e.:
(i_1,i_2), (i_2,i_3),..., (i_{l-1},i_l). The weight of a path p = (i_1,i_2), (i_2,i_3),..., (i_{l-1},i_l) is denoted by $|p|_w$ with

$$|p|_w = \left(a_{i_2,i_1} + a_{i_3,i_2} + \cdots + a_{i_1,i_{l-1}} \right).$$

The length of a path p is the sum of arc in the path p and it is denoted by $|p|_l$. The average weight of a path p is the weight of path p divided by the length of path p, i.e.

$$\frac{|p|_w}{|p|_l} = \frac{\left(a_{i_2,i_1} + a_{i_3,i_2} + \cdots + a_{i_1,i_{l-1}} \right)}{1-1}.$$

Circuit mean is the average weight of a circuit. Any circuit with maximum circuit mean is called *critical circuit*. A graph is called *strongly connected* graph if a path exists for every node i to every node j. If graph G(A) is strongly connected, then the matrix A is called *irreducible*.

Fig. 1 is graph G(A) of Example 1. There are three circuits, i.e. (1,1); (1,2),(2,1) and (2,3),(3,2). Each circuit

has circuit mean: $\frac{1}{1}=1; \frac{4+2}{2}=3; \frac{5+3}{2}=4$. It is shown that the maximum circuit mean equals 4. This circuit is (2,3),(3,2) and this one is the *critical circuit* of grap G(A).

The graph G(A) is *strongly connected*. Some notion that have discussed have been implemented in the toolbox [9]. The interpretation of the eigenvalue and the corresponding eigenvector of Example 1 as follows: Let be given three activities (nodes of graph G(A)), that ones periodically operated. Each node has its own kind of activity. It is assumed that an activity at certain node can only start when all preceding nodes have finished their activities and send the results of these activities along the arcs to the current node. Thus, the arc corresponding to $a_{i,j}$ can be interpreted as an output for node j and simultaneously as input for node i. Suppose that this node i starts its activity as soon as all preceding nodes have sent their results to node i. So an activity only can starts its activity at *(k+1)*-th time if all preceding activities have finished their activities and sent their results at *k*-th time. Element $a_{i,j}$ of matrix A represents the sum of the activity time of node j and the travelling time from node j to node i. If element $a_{i,j} = \varepsilon$, then the activity i does not depend on the activity j. If $x_i(k)$ is the earliest epoch at which node i become active for the *k*-th time with i =1, 2, 3, then the evolution of the system of Example 1 is given by difference equation of 1st-orde

$$x(k+1) = A \otimes x(k), k = 0, 1, 2, \dots \tag{2}$$

If it is chosen x(0) as the eigenvector of matrix A, then system (2) will operate as periodic with periodicity equals the eigenvalue λ=4. This follows the equation:

$$\begin{aligned} x(k+1) &= A \otimes x(k) \\ &= \lambda^{\otimes(k+1)} \otimes x(0), k = 0, 1, 2, \dots \end{aligned} \tag{3}$$

and we get the regular activities sequence x(k) :

$$\begin{pmatrix} 0 \\ 2 \\ 3 \end{pmatrix}, \begin{pmatrix} 4 \\ 6 \\ 7 \end{pmatrix}, \begin{pmatrix} 8 \\ 10 \\ 11 \end{pmatrix}, \begin{pmatrix} 12 \\ 14 \\ 15 \end{pmatrix}, \begin{pmatrix} 16 \\ 18 \\ 19 \end{pmatrix}, \begin{pmatrix} 20 \\ 22 \\ 23 \end{pmatrix}, \begin{pmatrix} 24 \\ 26 \\ 27 \end{pmatrix} \cdots$$

For nonnegative number $M \geq 0$, let be a matrix $A_m \in R_{max}^{n \times n}$ for $0 \leq m \leq M$ and $x(m) \in R_{max}^n$ for $-M \leq m \leq -1$, then the difference equation of *Mth*-order will be written as follows:

$$x(k) = \bigoplus_{m=0}^{M} A_m \otimes x(k-m), \ k \geq 0. \tag{4}$$

The Equation (2) is a difference equation of 1st-orde and $A_0 = \varepsilon$. A difference equation of *Mth*-orde with $A_0 \neq \varepsilon$ can be transformed into a difference equation of 1st-order that is given by Equation (2), as follows

$$\tilde{X}(k+1) = \tilde{A} \otimes \tilde{X}(k), \quad k \geq 0, \tag{5}$$

with

$$\tilde{A} = \begin{pmatrix} A_0^* \otimes A_1 & A_0^* \otimes A_2 & \cdots & \cdots & A_0^* \otimes A_M \\ E & \varepsilon & \cdots & \cdots & \varepsilon \\ \varepsilon & E & \ddots & \cdots & \varepsilon \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \varepsilon & \varepsilon & \cdots & E & \varepsilon \end{pmatrix} \tag{6}$$

and

$$A_0^* = \bigoplus_{i=0}^{n-1} A_0^{\otimes i}, \tag{7}$$

Matrix E is identity matrix. This can be found in [2].

III. SCHEDULING OF FELLOW SHOP

In this section, how a Max-Plus model of flow shop production system is derived. It would be discussed

thoroughly. Such a model is given a flow shop production of n job $\{J_i\}$ that will be scheduled on m machines $\{M_k\}$ with $1 < i < n$ and $k = 1, 2$. Each job must be processed exactly at a time in each machine by means of the same machine ordering. Permutation flow shop is a special class of flow shop by which each job would have the same processing order in each machine. Assuming is no set-up time between job operations. A dynamical model of flow shop production is constructed in [10] and an example is given. It is shown that the model has a regular scheduling. But it is not an optimal scheduling due to the makespan that is greater than the minimal one. Therefore, in this paper, the model is modified in such a way that it has a regular scheduling and minimal makespan. Firstly, it begins with analyzing some important factors of the model to decide when starting job processing of each cyclic production. Then, the analyses of flow job as a cycle of (periodic) production and, a job notion will be replaced by an operation to make clearer job ordering. Thus, giving a flow shop of 2 job, 2 machines, job J_1 in machine M_1 and J_1 in M_2 respectively will be called first and second operation. Therefore, the ordering of the same job J_1 processed in the different machine M_1 and M_2 more clearly. If $x_{ij}(k+1)$ represents starting time of j-th operation in i-th machine for $(k+1)$-th period and p_{ij} represents time processing of j-th operation in i-th machine, then $x_{ij}(k+1)$ will be a maximization of :

Release time of i-th machine for k-th period, i.e. completion operations for k-th period in i-th machine, this will be denoted by $y_i(k)$, and $y_i(0)$ for $k = 0$.

It is assumed that initial time of resources in each job is ready for first period so $r_{ij}(0) = 0$ for $k = 0$. There are no preceding jobs for the first job, so we define $r_{ij}(k) = \bigotimes_{i=1}^{k} \lambda_i$ for $k = 1, 2, \dots$ For the first operation of second job and the next job, the initial time of ready resources is the sum of period time and processing time of all preceding jobs operation, i.e. $r_{ij}(k) = \bigotimes_{i=1}^{k} \lambda_i \bigotimes_{\substack{n \in P_j \\ m \in M}} p_{mn}$

with $\lambda_i = \lambda$ represents period time (the eigenvalue of transition matrix of the model before modified [10]) ; the set of all preceding j operation is denoted by P_j and M represents a machine. Therefore P_{mn} represents processing time of operation n in machine m with preceding process n is processing j in machine i.

The completion time of all preceding operation of j operation, includes:
Processed operations in each machine of $(k+1)$-th period. If P_j is set of all preceding j operation of corresponding job, then for $n \in P_j$ and $m \in M$, completion time of all preceding j operation of $(k+1)$-th period is given by $x_{mn}(k+1) + p_{mn}$. If j operation is the first operation of an job, then this one has no preceding operation.
Other scheduled operation ($l \neq j$ and $l \in P$) that early processing of j operation in i machine of $(k+1)$-th period is represented in $x_{il}(k+1) + p_{il}$.

So we can write $x_{il}(k+1)$ as
$$x_{ij}(k+1) = \max\{y_i(k), r_{ij}(k), p_{mn}(k+1) + x_{mn}(k+1), p_{il} + x_{il}(k+1)\}$$
or in the Max-Plus Algebra as
$$x_{ij}(k+1) = y_i(k) \oplus r_{ij}(k) \oplus p_{mn}(k+1) \otimes \qquad (8)$$
$$x_{mn}(k+1) \oplus p_{il} \otimes x_{il}(k+1)$$

Release time in each 1-machine and 2-machine represented
$$y_1(k) = \max\{x_{11}(k) + p_{11}, x_{13}(k) + p_{13}, \cdots$$
$$\cdots, x_{1(n-1)}(k) + p_{1(n-1)}\}$$
$$y_2(k) = \max\{x_{22}(k) + p_{22}, x_{2,4}(k) + p_{34}, \cdots$$
$$\cdots, x_{2n}(k) + p_{1n}\}$$
or in the Max-Plus Algebra as
$$\begin{cases} y_1 = x_{11} \otimes p_{11} \oplus x_{13} \otimes p_{13} \oplus \cdots \\ \qquad \cdots \oplus x_{1(n-1)}(k) \otimes p_{1(n-1)} \\ y_2 = x_{22} \otimes p_{22} \oplus x_{24} \otimes p_{24} \oplus \cdots \\ \qquad \cdots \oplus x_{2n}(k) \otimes p_{2n} \end{cases} \qquad (9)$$

and from Equation (8) and (9) the modified model can be written as follows:
$$X(k+1) = Y(k) \oplus R(k) \oplus A_0 \otimes X(k+1)$$
$$Y(k+1) = A_2 \otimes X(k+1) \qquad (10)$$
With $y(0) = \begin{pmatrix} y_1(0) \\ y_2(0) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$

and
$$R(k) = r_{ij}(k) = \begin{cases} 0 & \text{for} \quad k = 0 \\ \bigotimes_{i=1}^{k} \lambda_i \bigotimes_{m,n \in P_j} p_{m,n} & \text{for } k = 1, 2, \cdots \end{cases}$$

Equation (10) can be rewritten as difference equation of 1st-order as follows:
$$X(k+1) = A_0 \otimes X(k+1) \oplus A_1 \otimes X(k) \oplus R(k) \qquad (11)$$

The solution can be found by transforming Equation (11) into difference equation 1st-order as Equation (5), i.e. :
$$\widetilde{X}(k+1) = \widetilde{A} \otimes \widetilde{X}(k) \oplus R(k) \qquad (12)$$
with $\widetilde{A} = A_0^* \otimes A_1$ and $A_0^* = \bigoplus_{i=0}^{n-1} A_0^{\otimes i}$

Example 2 Let be given flow shop production system that contains two jobs J_1 and J_2 scheduled in two machines M_1 and M_2 with time processing is given by Table 1.

By using Equation (11) and transforming this one into Equation (12), we get the modified model as follows:
$$\begin{pmatrix} x_{11}(k+1) \\ x_{13}(k+1) \\ x_{22}(k+1) \\ x_{24}(k+1) \end{pmatrix} = \underbrace{\begin{pmatrix} 2 & 3 & \varepsilon & \varepsilon \\ 2 & 3 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 5 & 6 \\ \varepsilon & \varepsilon & 5 & 6 \end{pmatrix}}_{A_1} \otimes \begin{pmatrix} x_{11}(k) \\ x_{13}(k) \\ x_{22}(k) \\ x_{24}(k) \end{pmatrix} \oplus$$

$$\underbrace{\begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 2 & \varepsilon & \varepsilon & \varepsilon \\ 2 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 3 & 5 & \varepsilon \end{pmatrix}}_{A_0} \otimes \begin{pmatrix} x_{11}(k+1) \\ x_{13}(k+1) \\ x_{22}(k+1) \\ x_{24}(k+1) \end{pmatrix} \oplus \underbrace{\begin{pmatrix} r_{11}(k) \\ r_{13}(k) \\ \varepsilon \\ \varepsilon \end{pmatrix}}_{R(k)}$$

$$X(k+1) = A_0 \otimes X(k+1) \oplus A_1 \otimes X(k) \oplus R(k)$$

And it is transformed as
$$\widetilde{X}(k+1) = \widetilde{A} \otimes \widetilde{X}(k) \oplus R(k)$$
$$\begin{pmatrix} \widetilde{x}_{11}(k+1) \\ \widetilde{x}_{13}(k+1) \\ \widetilde{x}_{22}(k+1) \\ \widetilde{x}_{24}(k+1) \end{pmatrix} = \underbrace{\begin{pmatrix} 5 & 3 & \varepsilon & \varepsilon \\ 5 & 3 & \varepsilon & \varepsilon \\ 13 & 9 & 11 & 6 \\ 13 & 9 & 11 & 6 \end{pmatrix}}_{\widetilde{A}} \otimes \begin{pmatrix} \widetilde{x}_{11}(k) \\ \widetilde{x}_{13}(k) \\ \widetilde{x}_{22}(k) \\ \widetilde{x}_{24}(k) \end{pmatrix} \oplus \begin{pmatrix} r_{11}(k) \\ r_{13}(k) \\ \varepsilon \\ \varepsilon \end{pmatrix}$$

So, we get

$$\begin{array}{ccc} x(1) & x(2) & x(3) \\ \begin{pmatrix} 0 \\ 2 \\ 2 \\ 7 \end{pmatrix}, & \begin{pmatrix} 11 \\ 13 \\ 13 \\ 18 \end{pmatrix}, & \begin{pmatrix} 22 \\ 24 \\ 24 \\ 29 \end{pmatrix}, \end{array} \qquad (13)$$

or one can be rewritten in the following form

$$x(k+1) = \begin{pmatrix} x_{11}(1) \otimes k\lambda \\ x_{22}(1) \otimes k\lambda \\ x_{13}(1) \otimes k\lambda \\ x_{24}(1) \otimes k\lambda \end{pmatrix} = x(1) \otimes k\lambda \otimes \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad (14)$$
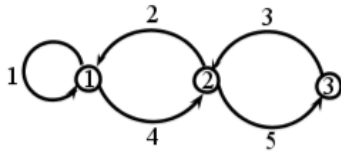
The result can be seen by using Gantt chart (Fig. 2).

Fig.1:Graph G(A)

TABLE 1.
TIME PROCESSING JOB J_1 AND J_2 IN M_1 AND M_2

	J_1	J_2
M_1	2	3
M_2	5	6

If the initial time of ready resources at 07.00 is written as 00 in the Table 2 and unit time processing in minute, then from scheduling of Equation (12) it can identify processing J_1 in M_1 (denoted by *node* x_1) starting at 07.00. Processing job J_2 in M_1 (x_2) and J_1 in M_2 (x_3) at 07.02 is written as 02. And processing J_2 in M_2 (x_4) at 07.07 is written as 07. In the same way, by using 07.00 as a reference time, we can know starting time of the next period processing that is given by Table 2.

From Fig. 2, it is shown that the system periodic with period equals 11 unit time and makespan 13 unit time, this result exactly equal to makespan as it is calculated in [13], i.e. firstly we make job matrix as follow:

$$M(J_x) = \begin{pmatrix} p_{x,1} & p_{x,1} p_{x,2} \\ \varepsilon & p_{x,2} \end{pmatrix}$$

with $M(J_{[i]})$ represents job matrix and $p_{[i],k}$ is time processing, $1 \le i \le n$ and $k = 1, 2, \dots$ So we get

$$M(J_1) = \begin{pmatrix} 2 & 2 \otimes 5 \\ \varepsilon & 5 \end{pmatrix} = \begin{pmatrix} 2 & 7 \\ \varepsilon & 5 \end{pmatrix}$$

$$M(J_2) = \begin{pmatrix} 3 & 3 \otimes 6 \\ \varepsilon & 6 \end{pmatrix} = \begin{pmatrix} 3 & 9 \\ \varepsilon & 6 \end{pmatrix}$$

$$M(\sigma) = \overset{|\sigma|}{\underset{i=1}{\otimes}} M(\sigma(i))$$

$$= \overset{2}{\underset{i=1}{\otimes}} M(\sigma(i)) = M(J_1) \otimes M(J_2)$$

$$= \begin{pmatrix} p_1(1) \otimes p_1(2) & p_1(1) \otimes p_{1,2}(2) \oplus p_{1,2}(1) \otimes p_2(2) \\ \varepsilon & p_2(1) \otimes p_2(2) \end{pmatrix}$$

$$= \begin{pmatrix} 2 \otimes 3 & (2 \otimes 9) \oplus (7 \otimes 6) \\ \varepsilon & 5 \otimes 6 \end{pmatrix}$$

$$= \begin{pmatrix} 5 & 13 \\ \varepsilon & 11 \end{pmatrix}$$

Therefore, makespan can be calculated from equation in below:

$$C(\sigma) = \begin{pmatrix} 0 & \varepsilon \end{pmatrix} \otimes M(\sigma)$$

$$= \begin{pmatrix} 0 & \varepsilon \end{pmatrix} \otimes \begin{pmatrix} 5 & 13 \\ \varepsilon & 11 \end{pmatrix} = \begin{pmatrix} 5 & 13 \end{pmatrix}$$

It is shown that completion time in M_1 equals 5 and completion time in M_2 equals 13. So, makespan of this case equals 13. With the same ordering job, this makespan less than initial makespan of Example in [10] that equals 17.

Therefore, Equation (12) can be used to obtain regular production model with minimum makespan and scheduling of starting process can be obtained by using Equation (14).
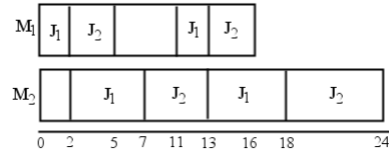

Fig. 2. Gantt chart production flow of flow shop 2 jobs 2 machines of Example 2 (modified model)

TABLE 2
SCHEDULING OF STARTING PROCESS
OF EXAMPLE 2 FOR 3 CYCLES PRODUCTION

	1		2		3	
	J_1	J_2	J_1	J_2	J_1	J_2
M_1	0	2	11	13	22	24
M_2	2	7	13	18	24	29

IV. CONCLUSION

In this section describe some notes for discussing especially flow shop scheduling. The modified model given by:

$$X(k+1) = Y(k) \oplus R(k) \oplus A_0 \otimes X(k+1)$$

$$Y(k+1) = A_2 \otimes X(k+1)$$

with
$$y(0) = \begin{pmatrix} y_1(0) \\ y_2(0) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

and

$$R(k) = r_{ij}(k) = \begin{cases} 0 & \text{for} \quad k = 0 \\ \overset{k}{\underset{i=1}{\otimes}} \lambda_i \underset{m,n \in P_j}{\bigotimes} p_{m,n} & \text{for } k = 1,2,\cdots \end{cases}$$

This model can be rewritten as a difference equation of 1st-order with $A_0 \ne \varepsilon$:

$$X(k+1) = A_0 \otimes X(k+1) \oplus A_1 \otimes X(k) \oplus R(k).$$

Next, we transform the model and get a new model that is given by:

$$\widetilde{X}(k+1) = \widetilde{A} \otimes \widetilde{X}(k) \oplus R(k) \text{ with } \widetilde{A} = A_0^* \otimes A_1 \text{ and } A_0^* = \bigoplus_{i=0}^{n-1} A_0^{\otimes i}$$

The scheduling of this model is regular and it has the minimal makespan. This result gives an idea to make model of flow shop production by using max-plus algebra. For the future work, the discussing can be continued for flow shop production with set-up time and more than two machines. The case of more than two machines the problem is NP-Hard [14].

REFERENCES

[1] F. Baccelli, G. Cohen, G.J. olsder, and J.-P. Quadrat, 1992, Synchronization and linearity: *Analgebra for Discrete Event Systems,* Wiley.

[2] B. Heidergott, G.J. olsder, and J. van der Woude, 2006, Max Plus at Work, *Modelling and Analysis of Synchronized Systems: A Course on Max-Plus Algebra and Its Applications*, Princeton University Press.

[3] C. Cassandras and S. Lafortune, 1999, *Introduction to Discrete Event Systems*, Kluwer Academic Publisher.

[4] Subiono, 2000, *On Classes of Min-max-plus Systems and Their Application*, PhD Thesis, Delft University of Technology, the Netherlands.

[5] Subiono and J. van der Woude, 2000, *Power Algorithms for* *(max,+)-and Bipartite (min,max,+)-Systems*, Discrete *Event* Dynamic Systems: Theory and Applications, 10(4):369-389.

[6] Subiono, 2000, *Operator Linier dalam Aljabar Max Plus dan Terapannya*, Proceeding Seminar Nasional Matematika: Peran Matematika Memasuki Milenium III, ITS, Surabaya.