Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 17 (2014) 71 – 75

www.elsevier.com/locate/procedia

Variety Management in Manufacturing. Proceedings of the 47th CIRP Conference on Manufacturing Systems

Max-Plus Modeling of Manufacturing Flow Lines

Seleim, A., ElMaraghy H.*

Intelligent Manufacturing Systems Centre, Univerity of Windsor, 401 Sunset Ave, Windsor, ON N9B 3P4, Canada

* Corresponding author. *E-mail address:* hae@uwindsor.ca

Abstract

Max-plus algebra can be used to model manufacturing flow lines using linear state-space-like equations which can be used in analysis and control. This paper presents a method for easy and quick generation of the max-plus equations for manufacturing flow lines of any size or structure. The generated equations can model flow lines with infinite as well as finite buffer sizes.

A flow line to be modeled is initially assumed to have infinite buffers for all stations. The line model equations are then generated as a combination of serial and merging stations after identifying the different stages using an adjacency matrix for the flow line. In the generated equations, the dynamics of the system are captured in two matrices that are function of the processing times of the different stations in the line. After generating these equations, extra terms are added to account for the finite buffers where for each buffer size, a matrix is added multiplied by the vector of system parameters delayed by the buffer size plus one.

The method is intuitive and easy to understand and code in software and thus can facilitate quick analysis of different configurations of manufacturing flow lines and assessing what if scenarios. This can also allow quick on-line reconfiguration of controllers for frequently reconfiguring flow lines.

Selection and peer-review under responsibility of the International Scientific Committee of "The 47th CIRP Conference on Manufacturing Systems" in the person of the Conference Chair Professor Hoda ElMaraghy"

Keywords: *Flow lines; Max-plus alegbra; Parametric modeling*

1. Introduction

Manufacturing systems fall under the category of Discrete Event Dynamic Systems (DEDS). For these systems, modelling tools include automata, petri-nets, markov-chains, queuing networks, simulation, and max-plus algebra [1]. Among these models, max-plus algebra is the only tool that can model the system using linear algebraic equations analogous to conventional state-space linear equations [2]. Using these equations, real time control as well as parametric system analysis becomes possible.

The use of Max-plus in modelling discrete event systems is fairly new starting in 1984 and since then it has been used in many applications in manufacturing systems including: manufacturing systems modelling [3, 4], performance evaluation [5, 6], performance optimization [7], and control [8, 9].

A corner stone in modeling, performance evaluation, performance optimization or control of manufacturing systems using max-plus algebra is obtaining the equations that describe the system. For relatively small and simple systems, these equations can be derived easily by hand, however; for large and complex systems, obtaining these equations is difficult, tedious and time consuming. In this paper, a method for easy and quick generation of the max-plus system equations for flow lines is presented. Flow lines studied in this paper are assumed to have deterministic processing times and reliable machines. The first assumption is realistic for automated systems as well as semi-automated systems with palletized material handling where the process time variation is much less than the processing time and thus can be neglected. The second

Selection and peer-review under responsibility of the International Scientific Committee of "The 47th CIRP Conference on Manufacturing Systems" in the person of the Conference Chair Professor Hoda ElMaraghy"

doi:10.1016/j.procir.2014.01.133

assumption is also realistic when studying the normal short-term system operation with the objective of understanding and optimizing the system behavior as opposed to studying long-term operation with the objective of planning system capacity where machines breakdown would have an effect.

A review of related research is presented in section 2. Section 3 presents a review of the basics of max-plus algebra; section 4 presents the method for generating the max-plus equations and section 5 presents the discussion and conclusions.

2. Related research

Modeling simple manufacturing systems using max-plus algebraic equations is easy and intuitive. The necessary conditions for each station to start operating on a job can be extracted from the description of the system and then written as a combination of addition and maximization operations. These equations can then be put together in a state-space matrix form, where the system parameters are the starting (or finishing) times of operating of the stations, and the system matrices are formed of the processing times of these stations. However, as the systems grow in size and/or have a complicated structure, generating the model equations becomes less intuitive, tedious and time consuming. In addition, modeling finite buffers with max-plus equations is not straight-forward or easy. Several papers have been published focusing on facilitating the modeling of manufacturing systems using max-plus algebra. Doustmohammadi and Kamen [10] presented a procedure for direct generation of event-time max-plus equations for generalized flow shop manufacturing systems. The procedure generates the equations directly only for serial flow lines with one station in each stage. In more complicated cases, the equations are generated for each machine separately, interconnection matrices which describe the flow of jobs through the line are derived and then the final equations are generated using matrix manipulations and several recursions. In addition, the procedure is limited to flow shops with infinite buffers. In [11], Goto et al. proposed a representation form for manufacturing systems that can account for finite buffers by adding relations between future starting times of jobs on a station and the past starting times for the same station and the following one. Imaev and Judd [12] used block diagrams which can be interconnected to form a manufacturing system model. This approach also assumes infinite buffer sizes.

In summary, the literature is lacking a tool that can easily generate max-plus equations for flow lines that are complex and contain finite buffers.

3. Basics of max-plus algebra

Max-plus algebra is an algebraic structure in which the two allowable operations are “maximization” and “addition”. In this section an introduction to the basic concepts and tools of the max-plus algebra will be presented.

Max-plus algebra is defined over $\mathcal{R}_{max} \to \{ \mathcal{R} \cup -\infty \}$ where $\mathcal{R}$ is the set of real numbers. The two main algebraic operations are maximization, denoted by the symbol $\oplus$, and addition, denoted by the symbol $\otimes$ where:

$$\mathrm{a} \oplus \mathrm{b} = \max\ (\mathrm{a}, \mathrm{b}) \quad \forall\, \mathrm{a}, \mathrm{b}\ \in \mathcal{R}_{\max}$$
$$\mathrm{a} \otimes \mathrm{b} = \mathrm{a} + \mathrm{b} \quad \forall\, \mathrm{a}, \mathrm{b}\ \in \mathcal{R}_{\max}$$

The null element of the operation $\oplus$ is ε which is equal to $-\infty$, and the null element for the operation $\otimes$ is e which is equal to 0. This can by demonstrated by:

$$\mathrm{a} \oplus \varepsilon = \max\ (\mathrm{a}, -\infty) = \mathrm{a} \quad \forall\ \mathrm{a}\ \in \mathcal{R}_{\max}$$
$$a \otimes e = a + 0 = a \quad \forall\ a\ \in \mathcal{R}_{max}$$

Similar to traditional algebra, both $\oplus$ and $\otimes$ are associative and commutative, and multiplication is left and right distributive over addition:

$$a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)\ \forall\, a, b, c\ \in \mathcal{R}_{max}$$
$$(a \oplus b) \otimes c = (a \otimes c) \oplus (b \otimes c)\ \forall\, a, b, c\ \in \mathcal{R}_{max}$$

Max-Plus algebra can be extended over matrices similar to conventional algebra. If $\boldsymbol{A}$ and $\boldsymbol{B}$ are two matrices with equal dimension then:

$$\boldsymbol{A} \oplus \boldsymbol{B} = \boldsymbol{C},$$

where $\boldsymbol{C}_{ij} = \boldsymbol{A}_{ij} \oplus \boldsymbol{B}_{ij}$. If the number of columns of $\boldsymbol{A}$ is equal to the number of rows of $\boldsymbol{B}$ equal to n, then:

$$\boldsymbol{A} \otimes \boldsymbol{B} = \boldsymbol{C},$$

where

$$\boldsymbol{C}_{ij} = \oplus_{k=1}^{n} (\boldsymbol{A}_{ik} \otimes \boldsymbol{B}_{ki}) \ ,$$

where $\oplus_{k=1}^{n} \boldsymbol{C}$ is maximization of all the elements of $\boldsymbol{C}$ over $k = 1$ to n.

Through the rest of the paper, the $\otimes$ operator will be omitted whenever its use is obvious, thus $a \otimes b \oplus c \otimes d$ will be written as $ab \oplus cd$.

An equation is the general form:

$$\boldsymbol{X} = \boldsymbol{A}\,\boldsymbol{X} \oplus \boldsymbol{B}\,\boldsymbol{U} \qquad (1)$$

where $\boldsymbol{X}$ is an $n \times 1$ vector of variables, $\boldsymbol{U}$ is an $m \times 1$ vector of inputs, $\boldsymbol{A}$ is an $n \times n$ square matrix and $\boldsymbol{B}$ is an $n \times m$ matrix, has a solution [14]:

$$\boldsymbol{X} = \boldsymbol{A}^{*}\,\boldsymbol{B}\,\boldsymbol{U} \qquad (2)$$

where $\boldsymbol{A}^{*}$ is defined as:

$$\boldsymbol{A}^{*} = e \oplus \boldsymbol{A} \oplus \boldsymbol{A}^{2} \oplus \ldots \oplus \boldsymbol{A}^{\infty}.$$

A complete detailed description and analysis of the max-plus algebra can be found in [14] and [15].

4. Flow lines modeling

Modelling will start with a flow line with *n* serial stations, then *n* different lines merging (assembling) in one line, then a general flow line with multiple serial lines with multiple merging. Modelling stations with finite buffers will then be presented afterwards in section *4.4*.

4.1. Modeling ‘n’ serial stations

The most common structure of a flow line is a serial structure with *n* processing stations, one input of raw material $\boldsymbol{U}$, and one output of finished products Y as shown in fig. 1. Let U_k, Y_k, and $X_{i,k}$ be the time at which the raw material is made available to the line, the time at which the finished product leaves the line and the starting time of processing on the i^{th} station for the k^{th} job respectively.

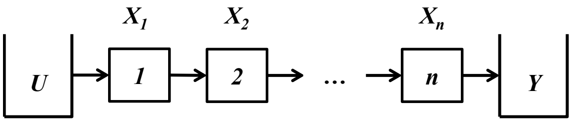


Fig. 1 Flow line with n serial stations

For station 1 to start processing the k^{th} job, the following conditions must be fulfilled: 1) raw material for the k^{th} job is made available, and 2) the station should have finished processing the k-1^{th} job. If t_1 is the processing time for station 1, then these conditions are translated into the following equation:

$$X_{1,k} = max(\, t_1 + X_{1,k-1}, U_k) \ , \tag{3}$$

which, in the max-plus algebra, is presented as:

$$X_{1,k} = t_1 X_{1,k-1} \oplus U_k \ . \tag{4}$$

Similarly, for any station i the conditions for starting processing the k^{th} job are: 1) the k^{th} job has finished processing on the i-1^{th} station, and 2) the i^{th} station should have finished processing the k-1^{th} job. These are expressed in max-plus algebra as:

$$X_{i,k} = t_i X_{i,k-1} \oplus t_{i-1} X_{i-1,k} \ , \tag{5}$$

Combining equations (4) and (5) in matrix form yields:

$$\boldsymbol{X}_k = \boldsymbol{A}\, \boldsymbol{X}_k \oplus \boldsymbol{B}\, \boldsymbol{X}_{k-1} \oplus \boldsymbol{D}\, \boldsymbol{U}_k \tag{6}$$

where,

$$\boldsymbol{X}_k = \begin{bmatrix} X_{1,k} \\ X_{2,k} \\ \vdots \\ X_{n,k} \end{bmatrix}, \ \boldsymbol{A} = \begin{bmatrix} \varepsilon & \varepsilon & \cdots & \varepsilon \\ t_1 & \varepsilon & \cdots & \varepsilon \\ & \ddots & & \vdots \\ \varepsilon & \varepsilon & t_{n-1} & \varepsilon \end{bmatrix},$$

$$\boldsymbol{B} = \begin{bmatrix} t_1 & \varepsilon & & \varepsilon \\ \varepsilon & t_2 & & \varepsilon \\ \vdots & & \ddots & \vdots \\ \varepsilon & \varepsilon & \cdots & t_n \end{bmatrix}, \text{ and } \boldsymbol{D} = \begin{bmatrix} e \\ \varepsilon \\ \vdots \\ \varepsilon \end{bmatrix}.$$

Following equation (2), equation (6) can be written as:

$$\boldsymbol{X}_k = \widehat{\boldsymbol{A}}\, \boldsymbol{X}_{k-1} \oplus \widehat{\boldsymbol{B}}\, \boldsymbol{U}_k \tag{7}$$

where:

$$\widehat{\boldsymbol{A}} = \boldsymbol{A}^* \otimes \boldsymbol{B} = \begin{bmatrix} t_1 & \varepsilon & \cdots & \varepsilon \\ t_1^2 & t_2 & & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ t_1^2 t_2 .. t_{n-1} & t_2^2 t_3 .. t_{n-1} & \cdots \ t_{n-1}^2 & t_n \end{bmatrix},$$

$$\text{and } \widehat{\boldsymbol{B}} = \boldsymbol{A}^* \otimes \boldsymbol{D} = \begin{bmatrix} e \\ t_1 \\ \vdots \\ t_1 t_2 .. t_{n-1} \end{bmatrix}.$$

From equation (7) it can be deduced that for any station $\boldsymbol{i}$, the starting time for the k^{th} job is equal to:

$$X_{i,k} = t_i X_{i,k-1} \oplus t_{i-1}^2 X_{i-1,k-1} \oplus t_{i-2}^2 t_{i-1} X_{i-2,k-1} \oplus \ldots \oplus t_1^2 t_2 \ldots t_{i-1} X_{1,k-1} \oplus t_1 t_2 \ldots t_{i-1} U_k \tag{8}$$

Equations (7) and (8) can be used to directly generate the max-plus equations for serial lines with any number of stages.

The output equation for the line can be written as:

$$\boldsymbol{Y}_k = \boldsymbol{C}\, \boldsymbol{X}_k \tag{9}$$

where, $\boldsymbol{C} = [\varepsilon \ \ \ldots \ \ \varepsilon \ \ t_n]$. Equation (9) can be used to represent the output equation for any line throughout the paper as the finishing time for the k^{th} job will always be the starting time of operation on the last station plus the processing time on that station.

4.2. Modeling 'n' merging lines

Merging lines are common in assembly flow lines. A merging station requires input from more than one station or line and delivers one output to the next station. Figure 2 shows n stations, each with its own input of raw material, merging into one station.

If t_i is the processing time for station i, and $U_{i,k}$ is the time at which raw material is made available for the 1_i^{th} station, then equation (4) holds for any station 1_i and the conditions for station 2 to start processing are: 1) the k^{th} job has finished processing on stations 1_i ($i = 1 \rightarrow n$), and 2) station 2 should have finished processing the k-1^{th} job.

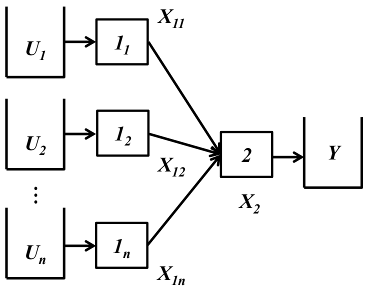


Fig. 2 Flow line with n merging lines

Accordingly, the max-plus equations for the system in figure 2 can be presented as:

$$\boldsymbol{X}_k = \boldsymbol{A}\, \boldsymbol{X}_k \oplus \boldsymbol{B}\, \boldsymbol{X}_{k-1} \oplus \boldsymbol{D}\, \boldsymbol{U}_k \tag{10}$$

where,

$$\boldsymbol{X}_k = \begin{bmatrix} X_{11,k} \\ X_{12,k} \\ \vdots \\ X_{1n,k} \\ X_{2,k} \end{bmatrix}, \boldsymbol{U}_k = \begin{bmatrix} U_{1,k} \\ U_{2,k} \\ \vdots \\ U_{n,k} \end{bmatrix}, \ \boldsymbol{A} = \begin{bmatrix} \varepsilon & \varepsilon & & \cdots & \varepsilon \\ \vdots & \vdots & & \cdots & \vdots \\ & & \ddots & & \\ \varepsilon & \varepsilon & & \varepsilon & \varepsilon \\ t_1 & t_2 & \cdots & t_{1n} & \varepsilon \end{bmatrix},$$

$$\boldsymbol{B} = \begin{bmatrix} t_{11} & \varepsilon & \cdots & & \varepsilon \\ \varepsilon & t_{12} & \varepsilon & \cdots & \varepsilon \\ \vdots & \varepsilon & \ddots & & \vdots \\ & \vdots & & t_{1n} & \varepsilon \\ \varepsilon & \varepsilon & \cdots & \varepsilon & t_2 \end{bmatrix},$$

$$\text{and } \boldsymbol{D} = \begin{bmatrix} e & \varepsilon & & \cdots & \varepsilon \\ \varepsilon & e & \varepsilon & \cdots & \varepsilon \\ \vdots & \varepsilon & \ddots & & \vdots \\ \varepsilon & \vdots & & e & \varepsilon \\ \varepsilon & \varepsilon & \cdots & \varepsilon & \varepsilon \end{bmatrix}.$$

Again following equation (2), equation (10) becomes:

$$\boldsymbol{X}_k = \widehat{\boldsymbol{A}}\,\boldsymbol{X}_{k-1} \oplus \widehat{\boldsymbol{B}}\,\boldsymbol{U}_k \quad (11)$$

where:

$$\widehat{\boldsymbol{A}} = \begin{bmatrix} t_{11} & \varepsilon & & \cdots & \varepsilon \\ \varepsilon & t_{12} & & & \varepsilon \\ \vdots & & \ddots & & \vdots \\ \varepsilon & \varepsilon & & t_{1n} & \varepsilon \\ t_{11}^2 & t_{12}^2 & \cdots & t_{1n}^2 & t_2 \end{bmatrix},$$

$$\text{and } \widehat{\boldsymbol{B}} = \begin{bmatrix} e & \varepsilon & & \cdots & \varepsilon \\ \varepsilon & e & & & \vdots \\ \vdots & & \ddots & & \\ \varepsilon & \varepsilon & & e & \varepsilon \\ t_{11} & t_{12} & & \cdots & t_{1n} \end{bmatrix}.$$

From equation (11) it can be deduced that for any station 1_i, the starting time for the k^{th} time is equal to:

$$X_{1i,k} = t_{1i} X_{i,k-1} \oplus U_i \quad (12)$$

and for station *2*, the starting time for the k^{th} time is equal to:

$$X_{2,k} = t_{11}^2 X_{11,k-1} \oplus t_{12}^2 X_{12,k-1} \oplus \ldots \oplus t_2 X_{2,k-1} \oplus t_{11} U_{1,k} \oplus t_{12} U_{2,k} \oplus \ldots \oplus t_{1n} U_{n,k} \quad (13)$$

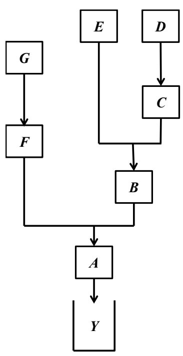
Again, equations (11), (12), and (13) can be used to directly generate the max-plus equations for merging lines of any number.

4.3. Modeling lines with serial and merging stations

Typical flow lines include both serial and merging lines. In the last two sub-sections it was shown that modeling serial and merging lines is easy and intuitive for each type by itself, however, when combined, modeling becomes less intuitive and as the system grows it becomes tedious and difficult.

In this sub-section, an algorithm is presented for the automatic generation of matrices $\widehat{\boldsymbol{A}}$ and $\widehat{\boldsymbol{B}}$ for any flow line with serial and merging stations. With matrices $\widehat{\boldsymbol{A}}$ and $\widehat{\boldsymbol{B}}$ generated, max-plus equations similar to equations (7) and (11) would be available for system analysis and control.

Step 1: Encode the flow line into an adjacency matrix while assuming the line to be a uni-directional graph. Figure 3 shows a flow line and its corresponding adjacency matrix.



	Y	A	B	C	D	E	F	G
Y	0	1	0	0	0	0	0	0
A	1	0	1	0	0	0	1	0
B	0	1	0	1	0	1	0	0
C	0	0	1	0	1	0	0	0
D	0	0	0	1	0	0	0	0
E	0	0	1	0	0	0	0	0
F	0	1	0	0	0	0	0	1
G	0	0	0	0	0	0	1	0

Fig. 3 A general flow line and its corresponding adjacency matrix

Step 2: Re-arrange the rows and columns of the matrix to cluster merging stations and identify the different stages in the line as shown in figure 4.

	Y	A	B	F	E	C	G	D
Y	0	1	0	0	0	0	0	0
A	1	0	1	1	0	0	0	0
B	0	1	0	0	1	1	0	0
F	0	1	0	0	0	0	1	0
E	0	0	1	0	0	0	0	0
C	0	0	1	0	0	0	0	1
G	0	0	0	1	0	0	0	0
D	0	0	0	0	0	1	0	0

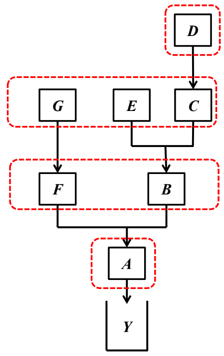


Fig. 4 Adjacency matrix and its corresponding flow line diagram after re-arranging the rows and columns of the matrix

Step 3: Generate the $\widehat{\boldsymbol{A}}$ and $\widehat{\boldsymbol{B}}$ matrices for the identified clusters using equations (8), (12), and (13) where stages are treated as serial stations and clusters of stations in the same stage are treated as merging ones. Assuming the starting times of stations in the given example for the k^{th} time is given by: $\boldsymbol{X} = [D \quad G \quad C \quad E \quad F \quad B \quad A]^T$, the $\widehat{\boldsymbol{A}}$ and $\widehat{\boldsymbol{B}}$ matrices would be:

$$\widehat{\boldsymbol{A}} = \begin{bmatrix} t_D & \varepsilon & & \cdots & & & \varepsilon \\ \varepsilon & t_G & \varepsilon & & & & \\ t_D^2 & \varepsilon & t_C & \varepsilon & & & \\ \varepsilon & \varepsilon & \varepsilon & t_E & \varepsilon & & \vdots \\ \varepsilon & t_G^2 & \varepsilon & \varepsilon & t_F & \varepsilon & \\ t_D^2 t_C & \varepsilon & t_C^2 & t_E^2 & \varepsilon & t_B & \varepsilon \\ t_D^2 t_C t_B & t_G^2 t_B & t_C^2 t_B & t_E^2 t_B & t_F^2 & t_B^2 & t_A \end{bmatrix}$$

$$\text{and } \widehat{\boldsymbol{B}} = \begin{bmatrix} e & \varepsilon & \varepsilon \\ t_D & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon \\ \varepsilon & \varepsilon & e \\ t_D t_C & t_E & \varepsilon \\ \varepsilon & \varepsilon & t_G \\ t_D t_C t_B & t_E t_B & t_G t_F \end{bmatrix}.$$

4.4. Modeling finite buffers

To model finite buffers; assume a general station *i* followed by a buffer with a finite buffer size *b*. For the k^{th} job to start on station *i* an extra condition is added by the buffer, which is for station *i+1* to have started processing the job number *k-b-1*. Assuming station *i* mentioned above is in a general flow line, then the equations for the line will be exactly according to the above sections with the addition of one term as follows:

$$\boldsymbol{X}_k = \widehat{\boldsymbol{A}}\,\boldsymbol{X}_{k-1} \oplus \widehat{\boldsymbol{B}}\,U_k \oplus \widehat{\boldsymbol{A}}_i\,\boldsymbol{X}_{k-b-1} \quad (14)$$

where:

$\widehat{\boldsymbol{A}}_i = \boldsymbol{A}^* \otimes \boldsymbol{A}_i$, and

$$\boldsymbol{A_i} = \begin{bmatrix} \varepsilon & \dots & \varepsilon & \varepsilon & \dots & \varepsilon \\ \vdots & & \vdots & \vdots & & \vdots \\ & & & \varepsilon & & \\ \varepsilon & \dots & \varepsilon & e & \varepsilon & \\ \vdots & & \vdots & \varepsilon & \vdots & \vdots \\ & & & \vdots & & \\ \varepsilon & \dots & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix},$$

where $\boldsymbol{A_i}$ is a null matrix with only one e located at the i^{th} row and the $i+1^{\text{th}}$ column.

To demonstrate, assume a line with four serial machines and three buffers as in figure 5. Assume the size of buffers 1 and 3 is b_1 and that of buffer 2 is b_2, then the equations for the line can be directly generated as:

$$\boldsymbol{X}_k = \widehat{\boldsymbol{A}}\,\boldsymbol{X}_{k-1} \oplus \widehat{\boldsymbol{B}}\,U_k \oplus \widehat{\boldsymbol{A}}_{\boldsymbol{1}}\,\boldsymbol{X}_{k-b1-1} \oplus \widehat{\boldsymbol{A}}_{\boldsymbol{2}}\,\boldsymbol{X}_{k-b2-1} \quad (15)$$

where $\widehat{\boldsymbol{A}}$ and $\widehat{\boldsymbol{B}}$ are the same as in equation (7),

$$\mathbf{A}_1 = \begin{bmatrix} \varepsilon & e & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & e \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}, \mathbf{A}_2 = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix},$$

$$\widehat{\boldsymbol{A}}_{\boldsymbol{1}} = \begin{bmatrix} \varepsilon & e & \varepsilon & \varepsilon \\ \varepsilon & t_1 & \varepsilon & \varepsilon \\ \varepsilon & t_1 t_2 & \varepsilon & e \\ \varepsilon & t_1 t_2 t_3 & \varepsilon & t_3 \end{bmatrix}, \text{ and } \widehat{\boldsymbol{A}}_{\boldsymbol{2}} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon \\ \varepsilon & \varepsilon & t_2 & \varepsilon \\ \varepsilon & \varepsilon & t_2 t_3 & \varepsilon \end{bmatrix}.$$

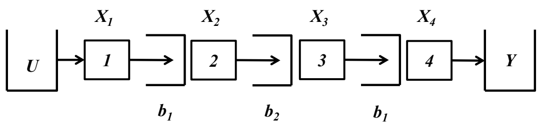


Fig. 5 Flow line with 4 serial stations and 3 finite buffers.

5. Discussion and Conclusions

A method was developed for quick and easy generation of the max-plus equations for flow lines of any size and structure, while taking into consideration finite buffer sizes. The method is based on the observation that a flow line can be decomposed into different ‘features’ each of which uniquely affects the final equations. These features can be added sequentially to form the final system equations. The correctness of all generated equations was verified by comparing the results with discrete event simulation models equivalent to each of the examples presented in the paper. The discrete event simulation software used is FlexSim [16].

Max-plus algebra can be used in modelling, performance evaluation and optimization as well as control of manufacturing systems. It offers the advantage of presenting the system in a parametric form and thus enables changing the value of the system parameters to evaluate different scenarios using the same set of equations. With the help of the method presented in this paper, this advantage is extended to different system configurations, where the equations for every configuration can be easily and quickly generated. The method also facilitates the use of max-plus model predictive control [8] and just in time control [9] for reconfigurable systems where quick-online generation of the system equations is required with every system reconfiguration.

The developed method for generating the Max Plus algebra system equations is easy to understand and implement in computer software, and thus can be used in analysis and control of flow lines.

References

[1] Cassandras CG, Lafortune S. Introduction to discrete event systems: Springer; 2007.

[2] Ho YC. Introduction to special issue on dynamics of discrete event systems. Proceedings of the IEEE 1989;77:3-6.

[3] Imaev A, Judd RP. Hierarchial modeling of manufacturing systems using max-plus algebra. American Control Conference, 2008: IEEE; 2008. p. 471-6.

[4] Ren S, Xu D, Wang F, Tan M. Timed event graph-based cyclic reconfigurable flow shop modelling and optimization. International Journal of Production Research 2007;45:143-56.

[5] Cohen G, Dubois D, Quadrat J, Viot M. A linear-system-theoretic view of discrete-event processes and its use for performance evaluation in manufacturing. Automatic Control, IEEE Transactions on 1985;30:210-20.

[6] Reddy Y, Janardhana GR, Kun KS. Modular modelling and performance evaluation of manufacturing systems using Max–Plus Algebra. International Journal of Industrial and Systems Engineering 2009;4:577-96.

[7] Di Febbraro A, Minciardi R, Parodi G, Prati A, Profumo M, Sacone S. Performance optimization in manufacturing systems by use of max-plus algebraic techniques. Systems, Man, and Cybernetics, 1994'Humans, Information and Technology', 1994 IEEE International Conference on: IEEE; 1994. p. 1995-2001.

[8] De Schutter B, Van Den Boom T. Model predictive control for max-plus-linear discrete event systems. Automatica 2001;37:1049-56.

[9] Houssin L, Lahaye S, Boimond J-L. Just in time control of constrained (max,+)-linear systems. Discrete Event Dynamic Systems 2007;17:159-78.

[10] Doustmohammadi A, Kamen EW. Direct generation of event-timing equations for generalized flow shop systems. Photonics East'95: International Society for Optics and Photonics; 1995. p. 50-62.

[11] Goto H, Shoji K, Nagai H, Onuma M. On the Max-Plus Linear representation for event-varying systems with buffer and order constraints. Industrial Engineering and Engineering Management, 2007 IEEE International Conference on: IEEE; 2007. p. 1823-7.

[12] Imaev A, Judd RP. Block diagram-based modeling of manufacturing systems using max-plus algebra. American Control Conference, 2009 ACC'09: IEEE; 2009. p. 4711-6.

[13] Park K, Morrison JR. Performance evaluation of deterministic flow lines: Redundant modules and application to semiconductor manufacturing equipment. Automation Science and Engineering (CASE), 2010 IEEE Conference on: IEEE; 2010. p. 45-50.

[14] Baccelli FL, Cohen G, Olsder GJ, Quadrat JP. Synchronization and linearity: Wiley New York; 1992.

[15] Heidergott B, Olsder GJ, van der Woude JW. Max Plus at work: modeling and analysis of synchronized systems: a course on Max-Plus algebra and its applications: Princeton University Press; 2006.

[16] FlexSim. FlexSim 3D Simulation Software.