Engineering Applications of Artificial Intelligence 25 (2012) 720–727

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Modeling and control of high-throughput screening systems in a max-plus algebraic setting☆

T. Brunsch [a,b,*], J. Raisch [a,c]

[a] Fachgebiet Regelungssysteme, Technische Universität Berlin, 10587 Berlin, Germany
[b] Laboratoire d'Ingénierie des Systèmes Automatisés, Université d'Angers, Angers, France
[c] Fachgruppe System- und Regelungstheorie, Max-Planck-Institut für Dynamik komplexer technischer Systeme, 39106 Magdeburg, Germany

ARTICLE INFO

Available online 4 December 2010

Keywords:
Cyclic systems
Discrete-event systems
Max-plus algebra
High-throughput screening
Scheduling

ABSTRACT

In this paper, we present a max-plus algebraic modeling and control approach for cyclically operated high-throughput screening plants. In previous work an algorithm has been developed to determine the globally optimal solution of the cyclic scheduling problem. The obtained optimal schedule is modeled in a max-plus algebraic framework. The max-plus algebraic model can then be used to generate appropriate control actions to handle unexpected deviations from the predetermined cyclic operation during runtime.

1. Introduction

Until the early 1990s the search for new pharmaceutical ingredients was performed manually. This was an extremely time-consuming procedure lasting for months or even years. Through advances in robotics and high-speed computer technology, it was possible to develop systems that are able to automatically screen thousands of substances in a very short time. The procedure of automatically analyzing biochemical compounds is called *high-throughput screening* (HTS). Nowadays HTS systems play an important role in the pharmaceutical industries, but they are also relevant to other fields of biology and chemistry.

A batch in HTS subsumes all worksteps that are necessary to analyze one set of substances. The set of substances is aggregated on one microplate. Additional microplates may be included in the batch to convey reagents or waste material. The plates are automatically moved between the resources of the HTS system, which include readers, incubators, and pipettors. To be able to compare many different batches of an experiment, each batch has to follow an identical pattern within the system, in terms of timing as well as in terms of ordering of resources. Thus, the system has to be operated cyclically. The aim of maximizing the throughput of the system results in a cyclic scheduling problem. An overview on cyclic scheduling can be found in, e.g., Hanen (1994) and Hanen and Munier (1995).

A method to determine the globally optimal schedules for cyclic systems, such as HTS systems, has been introduced by Mayer and Raisch (2004). This approach is based on *discrete-event systems* modeling, i.e., the system is characterized by the occurrence of discrete changes or events. More specifically, the model is given as a time window precedence network. Using standard graph reduction methods, the complexity of this network can then be reduced. The procedure ensures that the globally optimal solution of the scheduling system is not cut off. Another important step in the proposed method is the transformation of the resulting mixed integer non-linear program (MINLP) into a mixed integer linear program (MILP). Although these steps decrease the complexity of the system significantly, the scheduling problem is still too complex to be performed on-line. Therefore, the algorithm is carried out off-line before the execution of the HTS systems starts, i.e., it determines a static schedule. Static schedules, however, do not perform well when deviations from the predetermined cyclic scheme occur during runtime (Murray and Anderson, 1996).

To handle such deviations, we propose a supervisory control scheme using a *max-plus algebraic* model of the HTS system. The model is based on the specific operation the user wants to run as well as on the globally optimal cyclic schedule determined off-line. In case of a deviation from the cyclic scheme, the supervisor generates possible actions to be taken, i.e., the controller updates the schedule of the HTS plant and thus ensures continuous operation.

This paper is structured as follows. Section 2 summarizes the necessary concepts from graph theory and max-plus algebra. The different constraints for high-throughput screening systems are explained in Section 3. It is described how the constraints are merged into a max-plus algebraic model of the optimal HTS operation. In Section 4, a max-plus algebraic control scheme introduced for cyclic

☆This work has been partially supported by the European Community's Seventh Framework Programme under project DISC (Grant Agreement n. INFSO-ICT-224498).

* Corresponding author at: Fachgebiet Regelungssysteme, Technische Universität Berlin, 10587 Berlin, Germany.
E-mail address: brunsch@control.tu-berlin.de (T. Brunsch).

0952-1976/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.engappai.2010.11.006

systems by Li et al. (2007) is adapted for HTS systems. Conclusions and suggestions for future work are given in Section 5.

2. Graph theory and max-plus algebra

2.1. Graph theory

A directed graph (or digraph) is a pair $(\mathcal{V},\mathcal{E})$, where $\mathcal{V}$ is the set of nodes or vertices, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is a set of ordered pairs of nodes, called edges or arcs. A weighted graph is a digraph with a real number (the weight) $w_{ji} \in \mathbb{R}$ assigned to each arc $(v_i,v_j) \in \mathcal{E}$. It can be represented by a matrix $W \in \mathbb{R}_{max}^{n\times n}$, with $\mathbb{R}_{max} = \mathbb{R} \cup \{-\infty\}$ and n being the total number of nodes in the graph. The entries of the matrix W represent the weights of arcs. If no arc exists from node v_i to node v_j a weight of $-\infty$ is assigned to w_{ji}. $(\mathcal{V},\mathcal{E})$, together with the weight function $w : \mathcal{E} \to \mathbb{R}$, is then called the precedence graph of W. If the weights $w_{ji} \in \mathbb{R}_{max}$ represent times, the corresponding weighted digraph will also be referred to as a time window precedence network. Then, nodes represent events and arcs represent minimum time offsets between the occurrence of events.

2.2. Max-plus algebra

Max-plus algebra (e.g., Baccelli et al., 2001; Heidergott et al., 2006) is a powerful tool for the analysis and simulation of a certain class of discrete-event systems and provides a compact representation of such systems. It consists of two operations, $\oplus$ and $\otimes$ on the set $\mathbb{R}_{max} = \mathbb{R} \cup \{-\infty\}$. The operations are defined by: $\forall a,b \in \mathbb{R}_{max}$:

$$a \oplus b = \max(a,b),$$

$$a \otimes b = a+b.$$

The operation $\oplus$ is called *addition* of the max-plus algebra, the operation $\otimes$ is called *multiplication* of the max-plus algebra. The neutral element of max-plus addition is $-\infty$, also denoted as ε. The neutral element of multiplication is 0, also denoted as e.

For matrices $A,B \in \mathbb{R}_{max}^{n\times m}$ max-plus addition is defined by

$$[A \oplus B]_{ji} = [A]_{ji} \oplus [B]_{ji}.$$

The matrix product $A \otimes B$ for matrices $A \in \mathbb{R}_{max}^{n\times l}$ and $B \in \mathbb{R}_{max}^{l\times m}$ is defined by

$$[A \otimes B]_{ji} = \bigoplus_{k=1}^{l} ([A]_{jk} \otimes [B]_{ki}) = \max_{k=1,\dots,l}\{[A]_{jk} + [B]_{ki}\}.$$

Similar to conventional algebra, some standard properties, such as associativity and commutativity for $\oplus$ and $\otimes$, and distributivity of $\otimes$ over $\oplus$, hold for the max-plus algebra.

The earliest time instants for the occurrence of events in a timed precedence graph are determined by linear equations in the max-plus algebra. In particular, if we distinguish external (input and output) and internal events,

$$x = A_0 \otimes x \oplus B \otimes u,$$

$$y = C \otimes x,$$

where the vectors x, u and y contain the earliest time instants for the occurrence of the internal, the input and output events, respectively. Note that only the instants of the occurrence of input events, i.e., only the elements in u, can be delayed directly by a controller. The elements of matrix A_0 represent the minimum time offsets between the internal events. If the corresponding graph does not contain any circuits, matrix A_0 is said to be acyclic. In this case the matrix $A_0^* = I \oplus A_0 \oplus A_0^2 \oplus \cdots$ can be determined as $A_0^* = I \oplus A_0 \oplus A_0^2 \oplus \cdots \oplus A_0^{n-1}$, where I is the identity matrix with respect to max-plus algebra. For acyclic system matrices A_0, the implicit representation of the system can be rewritten in an explicit form:

$$x = A_0^* \otimes B \otimes u,$$

$$y = C \otimes x.$$

For cyclically repeated systems, the max-plus model has to be extended such that dependencies of events belonging to different cycles can be included. For systems that are causal with respect to the cycle index, an event in cycle k can only depend on events in the same cycle or in previous cycles. Thus, the recurrence relation for such systems can formally be written as

$$x(k) = \bigoplus_{q\in\mathbb{N}_0} (A_q \otimes x(k-q)) \oplus B \otimes u(k),$$

$$y(k) = C \otimes x(k),$$

with $k \in \mathbb{Z}$. This implicit recurrence relation can be rewritten in explicit form if the matrix A_0 is acyclic:

$$x(k) = \bigoplus_{q} (A_0^* A_q \otimes x(k-q)) \oplus A_0^* B \otimes u(k),$$

$$y(k) = C \otimes x(k),$$

with $k \in \mathbb{Z}$ and $q \in \mathbb{N}$.

2.3. Min-plus algebra

As mentioned in the previous section, max-plus algebra can be used to determine the earliest possible time instants for the occurrences of events in a timed precedence graph. However, it may not always be desirable that an event occurs as early as possible. From a scheduling point of view it is often desired that events occur just in time, i.e., the occurrence of some events shall be delayed as much as possible without interfering with the throughput of the system. The so-called latest necessary event times can be determined with min-plus algebra. Similar to max-plus algebra, min-plus algebra consists of two operations, $\oplus'$ and $\otimes'$ defined on the set $\mathbb{R}_{min} = \mathbb{R} \cup \{+\infty\}$, $\forall a,b \in \mathbb{R}_{min}$:

$$a \oplus' b = \min(a,b),$$

$$a \otimes' b = a+b.$$

The operations are called addition and multiplication of the min-plus algebra. The neutral element of min-plus addition is $\varepsilon' = +\infty$ and the neutral element of multiplication is $e' = 0$.

The standard properties of max-plus algebra, e.g., associativity and commutativity, also hold for min-plus algebra. The matrix operations for min-plus algebra can be directly derived form max-plus algebra by using $\oplus'$, $\otimes'$, ε', and e' instead of $\oplus$, $\otimes$, ε, and e.

3. Max-plus model of HTS systems

An HTS plant is assumed to consist of m resources. According to the operation the user wants to run, the sequence of activities for a single batch is given. It consists of μ activities and each activity i is assigned to one of the resources, denoted by $J_i \in \{1,\dots,m\}$, where it is executed. During the execution of activity i the respective resource J_i is said to be occupied. Different activities of a batch may overlap in time. Thus, a microplate may occupy two resources at the same time, e.g., during the transfer from one resource to another one. However, we assume all resources to have capacity one, i.e., no activity can allocate a resource while this resource is occupied by another activity.

One possibility to model temporal dependencies between events within a batch is through a time window precedence network. To do this, three different events have to be considered: start events o_i denoting the start of activity i, release events r_i

referring to the end of activity i, and transfer events which model the transfer of a batch between two resources. A transfer event always occurs simultaneously with a corresponding transfer event associated with another resource. In the corresponding graph, events are represented by nodes. Weighted arcs represent the temporal interdependencies and the weight denotes the minimum time which has to elapse between the events connected by the arc.

In general, dependencies between activities belonging to a single batch are called conjunctive constraints, while dependencies between activities (also within different batches) on each single resource are referred to as disjunctive constraints. Note that the disjunctive constraints also ensure that no activity is started as long as the corresponding resource is occupied by another activity.

3.1. Conjunctive constraints

As mentioned before, the single batch time scheme is defined by the operation the user wants to run. It includes specifications on the sequence of activities of a single batch as well as information on the minimal time needed for or between activities.

To illustrate the modeling of HTS systems, we will introduce a simple example. The considered process of a single batch consists of four activities executed on a total of three resources. The specific operation defined by the user is given as a time window precedence network consisting of start events, release events and transfer events (Fig. 1). For graphical illustration the system can also be represented as a Gantt chart (Fig. 2), where it can easily be seen that resource 1 is revisited by each batch. In particular, the activities $i=1$ and 4 of each batch are executed on resource 1. Time window precedence networks for real HTS applications are usually quite large. To simplify exposition, the number of nodes will be reduced using standard graph reduction methods. The reduced graph for the example from Fig. 1, denoted by $\mathcal{G}_c$, is called conjunctive graph. As can be seen in Fig. 3, the reduced graph consists only of the start and release events of every activity, i.e., $x=[o_1,r_1,o_2,r_2,o_3,r_3,o_4,r_4]^T$. This reduction is only used for illustrative reasons. For real systems, it is essential to also consider transfer events. However, the approach for non-reduced systems is identical to the approach for the reduced system. Thus, in this paper a reduced graph will be used, while keeping in mind that transfer events should be considered for real world systems. The matrix with the corresponding minimal time offsets is denoted by A_c. For our example,

$$A_c = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 9 & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 6 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 12 & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 9 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 16 & \varepsilon & 6 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 10 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 13 & \varepsilon \end{pmatrix}.$$

In max-plus algebra, the conjunctive constraints can then be conveniently represented by

$$x(k) = A_c \otimes x(k), \tag{1}$$

where the vector $x(k)$ contains the earliest possible times for the k-th occurrence of each activity's start and release events. By definition, the conjunctive graph does not contain any circuits and thus the matrix A_c is acyclic.

3.2. Disjunctive constraints

The disjunctive constraints describe the sequences of activities on each resource. They can be modeled as disjunctive graphs $\mathcal{G}_{d_\kappa}$, $\kappa=1,\ldots,m$, where m represents the total number of resources in the system. Note that disjunctive graphs are sometimes also called selection graphs (Hanen, 1994). Disjunctive constraints may contain dependencies between activities which do not belong to the same batch. To denote dependencies between events in different batches, marked arcs are added to the conventional notation of precedence graphs. An arc marked with a single "/" is said to be of first order. A first

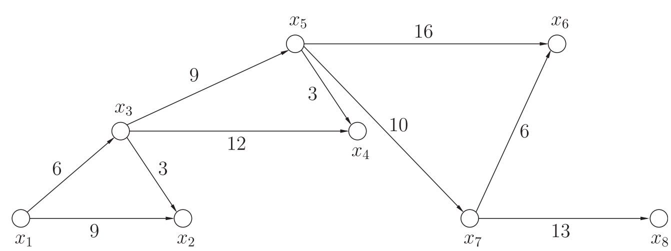

Fig. 3. Conjunctive graph $\mathcal{G}_c$.

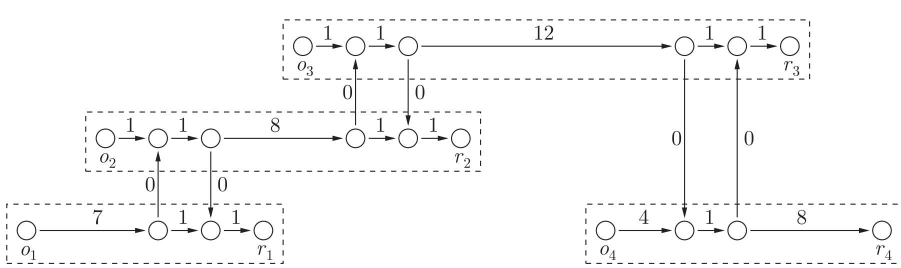

Fig. 1. Time window precedence network of single batch time scheme.

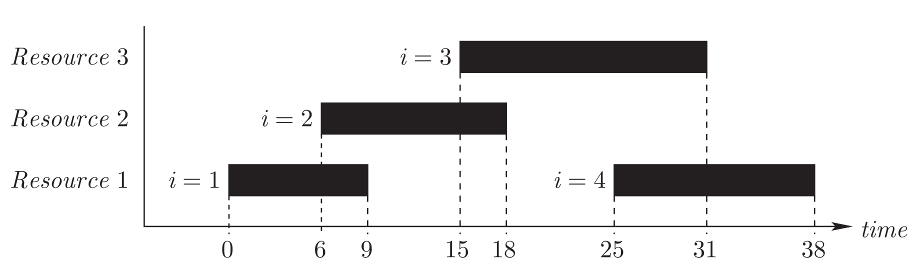

Fig. 2. Gantt chart of a single batch example.

order arc connecting event i with event j denotes the temporal dependencies between the two events belonging to subsequent batches. More precisely, a first order arc with weight w_{ji} is to be interpreted as $x_j(k) = x_i(k-1) + w_{ji}$, i.e., the earliest time instant of event j occurring in batch k is w_{ji} time units after event i of batch $k-1$ has occurred. Higher order arcs may also be necessary to represent the sequence of activities. Additionally, since some activities of a batch may be executed prior to some activities of a previous batch, arcs of negative order may be necessary. Such schedules are called overtaking (Seo and Lee, 2002) or interleaving (Murray and Anderson, 1996). The symbols for marked arcs are given in Table 1. Formally, any arc of order q, connecting node i to j, can be described by $(j,i)^{(q)}$, $q \in \mathbb{Z}$. The algebraic expression of this arc results in $x_j(k) = x_i(k-q) + w_{ji}$ with $q,k \in \mathbb{Z}$.

In general, the disjunctive graphs of systems containing only single capacity resources, i.e., resources that can handle one activity at the same time, are composed of one circuit of order 1 (Hanen and Munier, 1995). The order of a circuit is the sum of orders of the corresponding arcs.

Using marked arcs, it is possible to model the sequence and timing of activities on each resource (Geyer, 2004). The sequence and timing are determined by the globally optimal solution, obtained, i.e., with the off-line algorithm developed by Mayer and Raisch (2004). Fig. 4 shows the optimal sequence on the first resource for our example. It can be seen that the first activity of batch ρ is followed by the fourth activity of batch $\rho-1$ and this activity is then followed by the first activity of batch $\rho+1$. Thus the disjunctive graph for resource one consists of an arc of order $q=-1$, an arc of order $q=2$, and two zero order arcs.

The resulting disjunctive graph $\mathcal{G}_{d_1}$ is shown in Fig. 5. With this graph, it is easily possible to determine the dependencies between the corresponding events in the max-plus framework. E.g., for the events in Fig. 5 the dependencies are

$$x_1(k) = 0 \otimes x_8(k-2),$$

$$x_2(k) = 9 \otimes x_1(k),$$

$$x_7(k) = 0 \otimes x_2(k+1),$$

$$x_8(k) = 13 \otimes x_7(k).$$

Similarly, the disjunctive graphs and dependencies for resource two and three can be determined. Since only one activity of a batch is executed on each of these resources their corresponding disjunctive graphs contain only one first order arc and one arc of zeroth order between the start and release event of the activity (Fig. 6). After

Table 1
Notation of marked arcs.

Marking	Order of arc	Algebraic expression
None	0	$x_j(k) = x_i(k) + w_{ji}$
/	1	$x_j(k) = x_i(k-1) + w_{ji}$
//	2	$x_j(k) = x_i(k-2) + w_{ji}$
⋮	⋮	⋮
∘	−1	$x_j(k) = x_i(k+1) + w_{ji}$
∘∘	−2	$x_j(k) = x_i(k+2) + w_{ji}$
⋮	⋮	⋮

determining the disjunctive graphs $\mathcal{G}_{d_\kappa}, \kappa = 1, \ldots, m$, their max-plus algebraic representation can be written as

$$x(k) = \bigoplus_{q \in \mathbb{Z}} (A_{d_\kappa(q)} \otimes x(k-q)), \quad k \in \mathbb{Z}, \tag{2}$$

where x is the vector of the start and release events of all activities, and q denotes the order of arcs. Thus, the matrix $A_{d_\kappa(q)}$ encodes all arcs of q-th order within the disjunctive graph of resource κ.

3.3. Complete model

The overall HTS system operated in a predetermined time-optimal cyclic way can then be described by merging the conjunctive and all disjunctive graphs into one extended precedence graph $\mathcal{G}_{sys}$ (also called folded graph, Hanen and Munier, 1995). The extended precedence graph for our HTS example is shown in Fig. 7. In max-plus algebra, this can then be written as

$$x(k) = \underbrace{\left(A_c \oplus \left(\bigoplus_{\kappa=1}^{m} A_{d_\kappa(0)}\right)\right)}_{A_{(0)}} x(k) \oplus \left(\bigoplus_{q \in \mathbb{Z}\backslash 0} \underbrace{\left(\bigoplus_{\kappa=1}^{m} A_{d_\kappa(q)}\right)}_{A_{(q)}} x(k-q)\right), \tag{3}$$

with $k \in \mathbb{Z}$. On-line control in our scenario is restricted to delaying certain events. In the max-plus model (3) this is reflected by adding the term $B \otimes u(k)$ to the right hand side to give the overall model

$$x(k) = \left(\bigoplus_{q \in \mathbb{Z}} A_{(q)} x(k-q)\right) \oplus B \otimes u(k). \tag{4}$$

In particular, we assume that we can delay the start event of every single activity. Thus, $u(k) \in \mathbb{R}^{\mu}_{max}$, and in our example the matrix B is a 8×4 max-plus matrix containing only ε- and e-elements:

$$B = \begin{pmatrix} e & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & e \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}.$$

In general, Eq. (4) contains disjunctive dependencies of negative orders, i.e., terms with $q < 0$. Then, (4) is acausal with respect to the cycle index k (though, of course, not with respect to absolute time). To facilitate further analysis, (4) can be reformulated as a causal (in terms of k) system specification, i.e., without negative order terms. This can be done using the γ−transform (Baccelli et al., 2001).

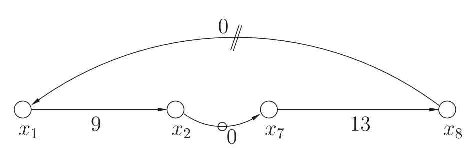

Fig. 5. Disjunctive graph $\mathcal{G}_{d_1}$.

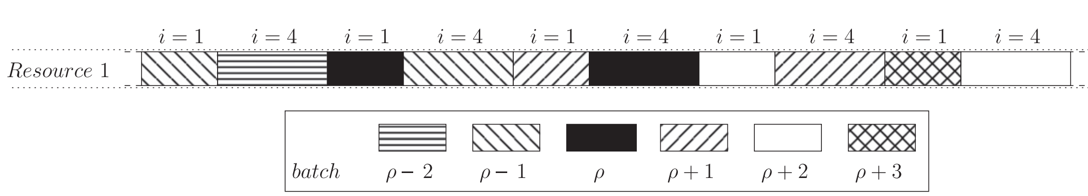

Fig. 4. Gantt chart of activities executed on resource 1.

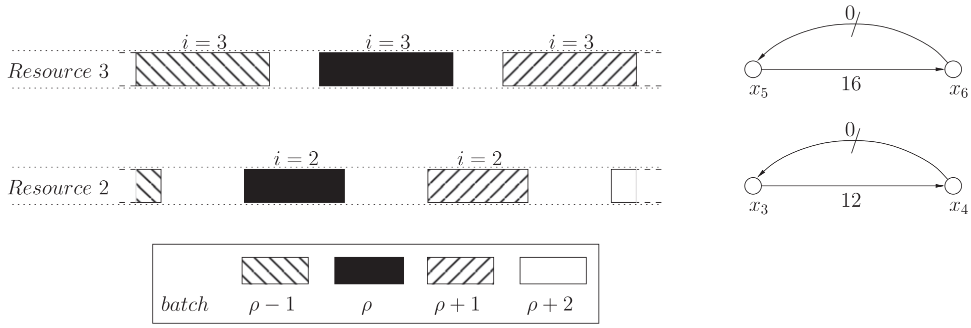

Fig. 6. Gantt charts of resources 2 and 3 on the left hand side and the corresponding disjunctive graphs on the right hand side.

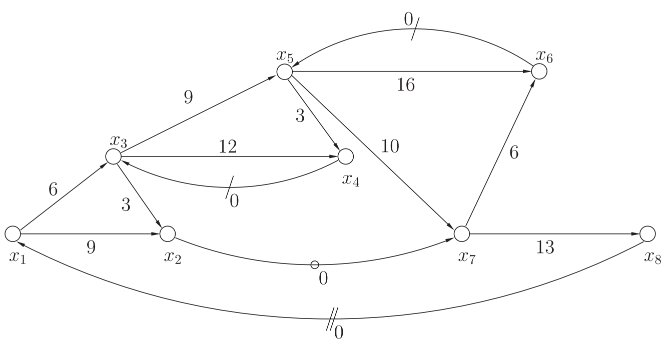

Fig. 7. Extended precedence graph $\mathcal{G}_{sys}$.

Formally, the $\gamma-$transform of x is defined as

$$X(\gamma) = \bigoplus_{k \in \mathbb{Z}} x(k) \otimes \gamma^k. \tag{5}$$

The definition of powers in max-plus algebra is extended to negative powers of γ by

$$\gamma^a \otimes \gamma^b = \gamma^{a+b} \quad \text{and} \quad \gamma^a \otimes \gamma^{-a} = \gamma^0 = e \quad \text{for } a,b \in \mathbb{Z},$$

where the exponents are added using conventional algebra.

Applying (5) to (4) provides

$$X(\gamma) = \mathcal{A}(\gamma) \otimes X(\gamma) \oplus B \otimes U(\gamma), \tag{6}$$

$$\text{with} \quad \mathcal{A}(\gamma) = \bigoplus_{q \in \mathbb{Z}} A_{(q)} \gamma^q,$$

$$\text{and} \quad U(\gamma) = \bigoplus_{k \in \mathbb{Z}} u(k) \gamma^k.$$

The elements of the matrix $\mathcal{A}(\gamma)$ can therefore be determined as max-plus polynomials

$$[\mathcal{A}(\gamma)]_{ji} = \bigoplus_{q \in \mathbb{Z}} a_{ji}^{(q)} \gamma^q, \tag{7}$$

where $a_{ji}^{(q)} = [A_{(q)}]_{ji}$.

Since HTS models, by definition, do not contain multiple arcs between two nodes this equation can also be written as

$$[\mathcal{A}(\gamma)]_{ji} = \begin{cases} a_{ji}^{(q)} \gamma^q & \text{if } \exists\ (j,i)^{(q)} \text{ for some } q, \\ \varepsilon & \text{else}. \end{cases} \tag{8}$$

After transforming the system into the $\gamma-$domain, another transform has to be applied to remove the dependencies of negative orders. Such dependencies are represented by $\gamma-$entries with negative powers in $\mathcal{A}(\gamma)$. We introduce a transformation matrix $\mathcal{T}_x(\gamma)$ and a transformed vector $\tilde{X}(\gamma) = \mathcal{T}_x(\gamma) \otimes X(\gamma)$, i.e, $\mathcal{T}_x(\gamma)^{-1} \otimes \tilde{X}(\gamma) = X(\gamma)$. Furthermore, we introduce an additional transformation matrix $\mathcal{T}_u(\gamma)$ and a transformed input vector $\tilde{U}(\gamma) = \mathcal{T}_u(\gamma) \otimes U(\gamma)$. Substituting this into (6) results in

$$\tilde{X}(\gamma) = \underbrace{\mathcal{T}_x(\gamma)\mathcal{A}(\gamma)\mathcal{T}_x(\gamma)^{-1}}_{\tilde{\mathcal{A}}(\gamma)} \otimes \tilde{X}(\gamma) \oplus \underbrace{\mathcal{T}_x(\gamma) B\ \mathcal{T}_u^{-1}(\gamma)}_{\tilde{\mathcal{B}}(\gamma)} \otimes \tilde{U}(\gamma). \tag{9}$$

The goal is to find a set of transformation matrices $\mathcal{T}_x$ and $\mathcal{T}_u$ such that $\tilde{\mathcal{A}}(\gamma)$ as well as $\tilde{\mathcal{B}}(\gamma)$ are devoid of entries with negative powers.

Note that inversion of matrices in max-plus algebra is not always possible unless the matrices are diagonal. Therefore, we choose diagonal transformation matrices $\mathcal{T}_x$ and $\mathcal{T}_u$, where the diagonal elements are powers of γ, i.e.,

$$\mathcal{T}_x = \begin{pmatrix} \gamma^{\pi_1} & \varepsilon & \cdots & \cdots & \varepsilon \\ \varepsilon & \gamma^{\pi_2} & \varepsilon & \cdots & \varepsilon \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \varepsilon \\ \varepsilon & \cdots & \cdots & \varepsilon & \gamma^{\pi_n} \end{pmatrix},$$

$$\mathcal{T}_u = \begin{pmatrix} \gamma^{\xi_1} & \varepsilon & \cdots & \cdots & \varepsilon \\ \varepsilon & \gamma^{\xi_2} & \varepsilon & \cdots & \varepsilon \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \varepsilon \\ \varepsilon & \cdots & \cdots & \varepsilon & \gamma^{\xi_\mu} \end{pmatrix},$$

with $\pi_i \in \mathbb{N}_0$, $i = 1, \ldots, n$, and $\xi_i \in \mathbb{N}_0$, $i = 1, \ldots, \mu$.

Obviously,

$$[\mathcal{T}_x(\gamma)^{-1}]_{ji} = \begin{cases} \gamma^{-\pi_i} & \text{for } j = i, \\ \varepsilon & \text{else}. \end{cases} \tag{10}$$

Therefore,

$$[\tilde{\mathcal{A}}]_{ji} = \gamma^{\pi_j} [\mathcal{A}]_{ji} \gamma^{-\pi_i}. \tag{11}$$

Substituting (8) into (11) results in

$$[\tilde{\mathcal{A}}]_{ji} = \gamma^{\pi_j} a_{ji}^{(q)} \gamma^q \gamma^{-\pi_i}. \tag{12}$$

Thus, to ensure that the matrix $\tilde{\mathcal{A}}$ is devoid of elements with negative powers, we have to choose the non-negative integers π_i, $i = 1, \ldots, n$, such that

$$\pi_j - \pi_i \geq -q \tag{13}$$

for every element $[\mathcal{A}]_{ji} \neq \varepsilon$.

Similarly, the elements of matrix $\tilde{\mathcal{B}}(\gamma) = \mathcal{T}_x(\gamma) B \mathcal{T}_u^{-1}(\gamma)$ can be determined as

$$[\tilde{\mathcal{B}}]_{ji} = \gamma^{\pi_j} [B]_{ji} \gamma^{-\xi_i} = \gamma^{\pi_j} b_{ji} \gamma^{-\xi_i}.$$

Of course, we have to ensure that no negative order elements are generated in $\tilde{\mathcal{B}}$, i.e., we have to choose non-negative exponents ξ_i, $i = 1, \ldots, \mu$, such that

$$\pi_j - \xi_i \geq 0, \tag{14}$$

for all elements $b_{ji} \neq \varepsilon$.

It can be argued, that conditions (13) and (14) can be satisfied simultaneously if the extended precedence graph $\mathcal{G}_{sys}$ does not contain non-positive order circuits (Geyer, 2004). This is always true for implementable specifications.

This procedure does not yield unique transformation matrices $\mathcal{T}_x$ and $\mathcal{T}_u$. Therefore,

$$\tilde{\mathcal{A}}(\gamma) = \bigoplus_{q \in \mathbb{N}_0} \tilde{A}_{(q)} \gamma^q \tag{15}$$

and

$$\tilde{\mathcal{B}}(\gamma) = \bigoplus_{q \in \mathbb{N}_0} \tilde{B}_{(q)} \gamma^q, \tag{16}$$

are also non-unique. However, any set of transformation matrices obtained that way provides the desired result.

In a last step, it is necessary to apply the inverse γ−transformation to (9) to obtain an implicit recurrence relation for a time-optimal schedule of the HTS system

$$\tilde{x}(k) = \bigoplus_{q \in \mathbb{N}_0} (\tilde{A}_{(q)} \otimes \tilde{x}(k-q) \oplus \tilde{B}_{(q)} \otimes \tilde{u}(k-q)), \tag{17}$$

with $k \in \mathbb{Z}$. Obviously, we have

$$\tilde{x}_i(k) = x_i(k-\pi_i),$$

$$\tilde{u}_i(k) = u_i(k-\xi_i). \tag{18}$$

Therefore, the vector $\tilde{x}$ contains time instants of events which do not necessarily belong to the same batch. Hence, the transformation $\mathcal{T}_x$ corresponds to a relabeling of events within the single batch time scheme. Eq. (17) can be rewritten in explicit form as $\tilde{A}_{(0)}$ is acyclic:

$$\tilde{x}(k) = \bigoplus_{q \in \mathbb{N}} (\tilde{A}^*_{(0)} \tilde{A}_{(q)} \otimes \tilde{x}(k-q) \oplus \tilde{A}^*_{(0)} \tilde{B}_{(q)} \otimes \tilde{u}(k-q)), \tag{19}$$

where $\tilde{A}^*_{(0)} = I \oplus \tilde{A}_{(0)} \oplus \tilde{A}^2_{(0)} \oplus \cdots \oplus \tilde{A}^{n-1}_{(0)}$, and $k \in \mathbb{Z}$.

Returning to our example, there are 14 arcs in the graph $\mathcal{G}_{sys}$ illustrated in Fig. 7, i.e., there are 14 non-ε elements in the corresponding matrix $\mathcal{A}(\gamma)$:

$$\mathcal{A}(\gamma) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma^2 \\ 9 & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 6 & \varepsilon & \varepsilon & \gamma & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 12 & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 9 & \varepsilon & \varepsilon & \gamma & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 16 & \varepsilon & 6 & \varepsilon \\ \varepsilon & \gamma^{-1} & \varepsilon & \varepsilon & 10 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 13 & \varepsilon \end{pmatrix}.$$

Thus, a total of 14 constraints have to hold for a valid transformation matrix $\mathcal{T}_x$:

$$\begin{array}{llll} \pi_1-\pi_8 \geq -2, & \pi_2-\pi_1 \geq 0, & \pi_2-\pi_3 \geq 0, & \pi_3-\pi_1 \geq 0, \\ \pi_3-\pi_4 \geq -1, & \pi_4-\pi_3 \geq 0, & \pi_4-\pi_5 \geq 0, & \pi_5-\pi_3 \geq 0, \\ \pi_5-\pi_6 \geq -1, & \pi_6-\pi_5 \geq 0, & \pi_6-\pi_7 \geq 0, & \pi_7-\pi_2 \geq 1, \\ \pi_7-\pi_5 \geq 0, & \pi_8-\pi_7 \geq 0. & & \end{array}$$

One possible solution meeting all constraints is $\pi_1 = \pi_2 = \pi_3 = \pi_4 = \pi_5 = 0$, $\pi_6 = \pi_7 = \pi_8 = 1$. The resulting transformation matrix has the form:

$$\mathcal{T}_x(\gamma) = \begin{pmatrix} \gamma^0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \gamma^0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \gamma^0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \gamma^0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma^0 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma^1 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma^1 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma^1 \end{pmatrix}.$$

In a similar fashion one can determine a possible transformation matrix $\mathcal{T}_u$. One solution $\mathcal{T}_u(\gamma) = diag\{\gamma^{\xi_i}\}$, with $i = 1,\ldots,4$ and $\xi_1 = \xi_2 = \xi_3 = 0, \xi_4 = 1$.

Applying these transformations results in the following system without negative order dependencies:

$$\tilde{X}(\gamma) = \underbrace{\begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma \\ 9 & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 6 & \varepsilon & \varepsilon & \gamma & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 12 & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 9 & \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 16\gamma & \varepsilon & 6 & \varepsilon \\ \varepsilon & 0 & \varepsilon & \varepsilon & 10\gamma & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 13 & \varepsilon \end{pmatrix}}_{\tilde{\mathcal{A}}(\gamma)} \otimes \tilde{X}(\gamma)$$

$$\oplus \underbrace{\begin{pmatrix} 0 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 0 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}}_{\tilde{\mathcal{B}}(\gamma)} \otimes \tilde{U}(\gamma).$$

The corresponding timed precedence network is shown in Fig. 8. Since it does not contain arcs of order higher than 1, the resulting recurrence relation for the example HTS system can be written as

$$\tilde{x}(k) = \tilde{A}_{(0)} \tilde{x}(k) \oplus \tilde{A}_{(1)} \tilde{x}(k-1) \oplus \tilde{B}_{(0)} \tilde{u}(k),$$

which can then be modified to give an explicit equation:

$$\tilde{x}(k) = \underbrace{\tilde{A}^*_{(0)} \tilde{A}_{(1)}}_{\tilde{A}} \tilde{x}(k-1) \oplus \underbrace{\tilde{A}^*_{(0)} \tilde{B}_{(0)}}_{\tilde{B}} \tilde{u}(k), \tag{20}$$

with $\tilde{A}^*_{(0)} = I \oplus \tilde{A}_{(0)} \oplus \tilde{A}^2_{(0)} \oplus \cdots \oplus \tilde{A}^{n-1}_{(0)}$.

In general, higher order arcs will also be present in $\tilde{G}_{sys}$. Then by extending the state vector, the system can again be written in the form (20).

In this explicit form, it is now possible to analyze the system using max-plus algebra. For example, it is possible to determine the eigenvalue λ of $\tilde{A}$ and its corresponding eigenvector v (Baccelli et al., 2001; Heidergott et al., 2006). In max-plus algebra, the eigenvalue is also called the asymptotic growth rate, as it represents the minimal time interval between the occurrence of the same events in subsequent batches. Thus, the eigenvalue is the minimal cycle time T. For the max-plus representations of HTS systems, there is one unique eigenvalue, however, in general, a max-plus system may have more than one eigenvalue.

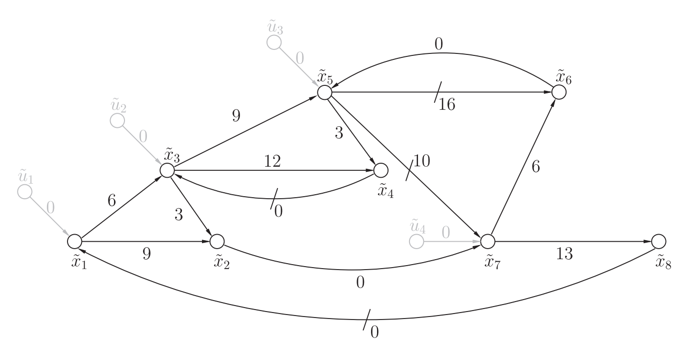

Fig. 8. Timed precedence network after transformation.

Note that the max-plus model is a result of an off-line optimization procedure, which determines the sequence and timing of activities on each resource. The max-plus model is a compact representation of the earliest possible instants of time for the occurrence of events in different batches. In the next section, it will be shown that the described max-plus model can be conveniently used for on-line adjustments (i.e., feedback control) in case unforeseen disturbances make the predetermined optimal policy impossible.

4. Control of HTS systems

For the control of high-throughput screening systems we propose a model-based feedback scheme. The (possibly disturbed) state of the HTS plant is measured. This state is then used to generate a vector of earliest possible event times (EPETs). Note that in this work we only consider disturbances which correspond to finite-time delays. Usually permanent breakdowns of single resources cannot be compensated by other resources. The "loss" of one or several resources will in general result in a stop of the complete operation of the system. However, if the breakdown is only temporary and the problem can be resolved in a reasonable time, such a breakdown may be considered as a simple delay. Furthermore, to be able to model unforeseen delays it is necessary to expand the max-plus model (20) by a disturbance d:

$$\tilde{x}(k) = \tilde{A}\tilde{x}(k-1) \oplus \tilde{B}\tilde{u}(k) \oplus Sd(k). \tag{21}$$

In the following, we adopt the control schemes proposed in Li et al. (2007) for a class of cyclically repeated discrete-event systems and in Lhommeau et al. (2002) for disturbance decoupling to max-plus models of HTS systems. In this context, the state is fed back in an on-line fashion, i.e., it is updated with a high sampling frequency. The input to the feedback controller is therefore $\tilde{x}(k,t_j)$. The controller then determines the input signal $\tilde{u} = \tilde{u}(k,t_j) \in \mathbb{R}^{\mu}_{max}$. Inserting such an update into Eq. (20) will result in a max-plus system which will then depend on the current update time t_j, i.e.,

$$\tilde{x}(k,t_j) = \tilde{A}\tilde{x}(k-1) \oplus \tilde{B}\tilde{u}(k,t_j) \oplus Sd(k). \tag{22}$$

In this way the earliest possible event time instances can be determined. However, as mentioned before it may not be desirable to start every event as early as possible. Using min-plus algebra, it is possible to calculate the latest necessary event time instances by

$$\overline{x}(k,t_j) = -(\tilde{A}^*_{(0)})^T \otimes' \overline{x}_R(k,t_j), \tag{23}$$

where

$$[\overline{x}_R]_i(k,t_j) = \begin{cases} \tilde{x}_i(k,t_j) & \text{if } \tilde{x}_i \text{ is a release event,} \\ \varepsilon' & \text{otherwise.} \end{cases} \tag{24}$$

Transposing matrix $\tilde{A}^*_{(0)}$ in (23) changes the direction of every arc in the timed precedence network. In vector $\overline{x}_R$ only the time instances of release events are considered and multiplying this vector in min-plus algebra with matrix $-(\tilde{A}^*_{(0)})^T$ basically means that the system is simulated backwards, i.e., the release times are set and the latest necessary start times of all other events are determined. As we fix the time instances of all release events to the earliest possible values, the throughput of the system is not changed.

The elements of the updated input $\tilde{u}(k,t_j)$ are then determined the following way:

- For a start event that has already occurred in cycle k, the corresponding element in $\tilde{u}(k,t_j)$ is exactly the time instant of its occurrence. For a start event which is next to occur on a resource, the time of its occurrence is estimated by

  $$\tilde{u}_i(k,t_j) = \left[\tilde{B}^T_{(0)} \otimes (\tilde{x}(k,t_j) \oplus \overline{x}(k,t_j))\right]_i,$$

  where $\tilde{x}(k,t_j)$ and $\overline{x}(k,t_j)$ are determined by (22) and (23), respectively.
- For all other start events, the corresponding entry in $\tilde{u}(k,t_j)$ will be

  $$\tilde{u}_i(k,t_j) = \varepsilon,$$

  i.e., these start events will not pose any restriction for the future evolution of the system.

The overall structure of the controlled HTS system is shown in Fig. 9. Using this input, the max-plus model can be used to handle unexpected deviations from the predetermined cyclic schedule of HTS systems. If a delay occurs at time t_d during cycle k, i.e., some activity in cycle k may take longer than expected, it will first appear in the vector $\tilde{x}(k,t_d)$. This vector is comprised the time instants of the occurrences of events in the k-th cycle, and it is used to determine the new input $\tilde{u}(k,t_d)$ of the system. If the delay affects any future events in cycle k, the scheduled time instants of their occurrence are rescheduled. Furthermore, possible delays of events in the k-th cycle may be carried over into the next cycles. This effect is handled through the first part of (22), where possible deviations of previous cycles are included through the state vector $\tilde{x}(k-1)$. The newly determined (updated) information on the time instants of future events can then be used to control the plant.

Generally speaking, after the occurrence of a delay the max-plus controller postpones every future event by the minimal amount of time needed to make the schedule feasible. Doing so every constraint included in the max-plus model is enforced. In the terms of scheduling theory this is also referred to as a minimal *right shift rescheduling* (Vieira et al., 2003). Consequently, the controller ensures continuous operation of the HTS plant and guarantees the earliest possible finishing of subsequent batches.

4.1. Simulation

For the simulation we consider the complete system shown in Fig. 1, rather than the reduced one. The Gantt charts of the controlled system and the uncontrolled system is illustrated in Fig. 10. In the uncontrolled system, every activity starts as soon as the corresponding resource is available. Using the controller, however, the activities start *just-in-time* to achieve the desired feedback. Especially for the activity which is executed on *Resource 2*, it can clearly be seen that it starts later with the controller, while the throughput is the same. So even for the undisturbed case the use of a controller improves the operation of the system by reducing the length of allocation intervals of resources.

Furthermore, the controller improves the behavior in case of unexpected deviations from the cyclic scheme. In the following we consider a delay of known duration in the post-processing time of activity 2 on *Resource*2. For the uncontrolled case, every activity is extended by the time of the disturbance (see upper part of Fig. 11). For the controlled case, we also have a right-shift of some events (comp. lower part of Fig. 11). The result is the fastest recovery of the predetermined cyclic regime. During the transition phase start events occur just in time, i.e., at the latest necessary time instants. Also, in this specific case, only one batch is affected by the delay for the controlled system, while two batches are affected in the

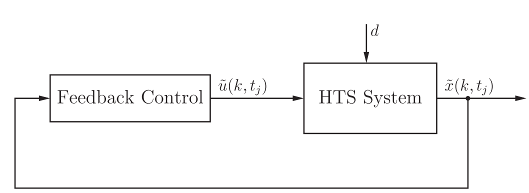

Fig. 9. Control structure.

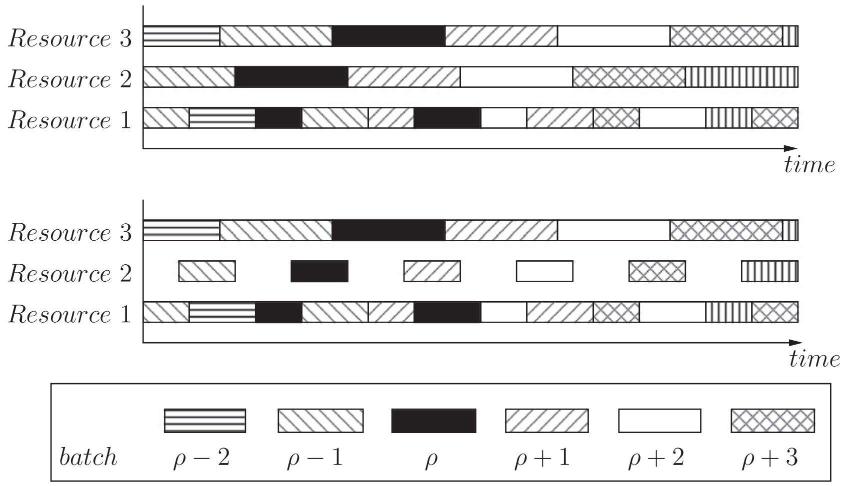

Fig. 10. Gantt charts of the uncontrolled system (upper part) and the controlled system (lower part).

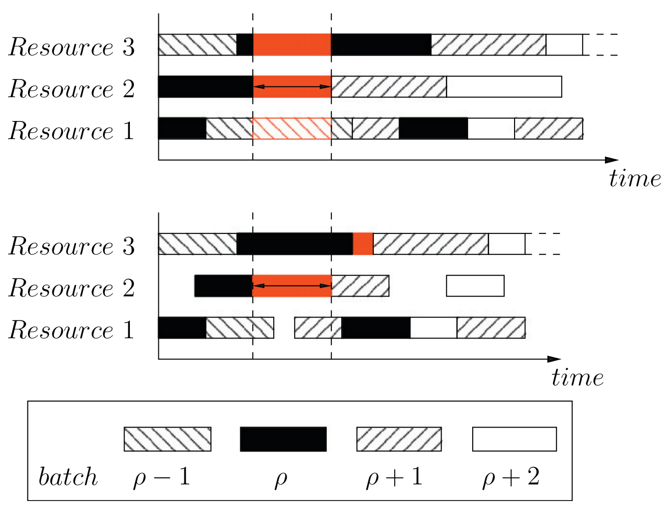

Fig. 11. Gantt charts of the uncontrolled system (upper part) and the controlled system (lower part) in case of a disturbance.

uncontrolled system. This is an advantage, because data obtained from batches which were disturbed cannot be considered for further analysis. Thus, for the uncontrolled system the sets of data of batches ρ and $\rho-1$ need to be removed, while only the set of data of batch ρ is removed in the controlled system.

5. Conclusion and future work

This contribution proposes a max-plus algebraic model with negative order arcs for high-throughput screening systems operated in a predetermined time-optimal cyclic mode. It is shown how a system with negative order arcs can be transformed into a system devoid of such arcs. Furthermore, the control scheme introduced in Li et al. (2007) is adapted and it is shown how, on the basis of the max-plus model, deviations of the predetermined cyclic schedule, such as delay of events can be handled.

First results show that the proposed model can easily be extended to HTS systems with multi-capacity resources, i.e., resources that can accommodate more than one activity at a time (Brunsch and Raisch, 2009). Given a HTS system with multi-capacity resources, further deviations from the predetermined cyclic scheme, such as partial break down of multi-capacity resources, can be considered.

Another possible modification is the way the latest necessary event times are determined. Instead of a min-plus algebraic approach, residuation theory (Blyth and Janowitz, 1972) seems to be promising for the calculation of a *just-in-time* schedule. It has already been applied to standard manufacturing systems (e.g., Cottenceau et al., 2001; Lhommeau et al., 2002) and with some modifications these approaches could be used for high-throughput-screening systems.

References

Baccelli, F., Cohen, G., Olsder, G.J., Quadrat, J.-P., 2001. Synchronization and Linearity—An Algebra for Discrete Event Systems, web Edition. Wiley.
Blyth, T., Janowitz, M., 1972. Residuation Theory. Pergamon Press, Oxford.
Brunsch, T., Raisch, J., 2009. Max-plus algebraic modeling and control of high-throughput screening systems with multi-capacity resources. In: Preprints of the 3rd IFAC Conference on Analysis and Design of Hybrid Systems (ADHS'09), Zaragoza, Spain, September, pp. 132–137.
Cottenceau, B., Hardouin, L., Boimond, J.-L., Ferrier, J.-L., 2001. Model reference control for timed event graphs in dioids. Automatica 37 (9), 1451–1458.
Geyer, F., 2004. Analyse und Optimierung zyklischer ereignisdiskreter Systeme mit Reihenfolgealternativen. Diploma Thesis, Otto-von-Guericke-Universität Magdeburg.
Hanen, C., 1994. Study of a NP-hard cyclic scheduling problem: the recurrent job-shop. European Journal of Operational Research 72, 82–101.
Hanen, C., Munier, A., 1995. Cyclic scheduling on parallel processors: an overview. In: Scheduling Theory and its Applications. John Wiley & Sons, pp. 193–226.
Heidergott, B., Olsder, G.J., vander Woude, J., 2006. Max Plus at Work, first ed. Princeton University Press, Princeton, NJ, USA.
Lhommeau, M., Hardouin, L., Cottenceau, B., 2002. Disturbance decoupling of timed event graphs by output feedback controller. In: WODES'2002, Workshop on Discrete Event Systems, Zaragoza, Spain, October.
Li, D., Mayer, E., Raisch, J., 2007. A new hierarchical control scheme for a class of cyclically repeated discrete-event systems. In: Informatics in Control, Automation and Robotics II. Springer, Netherlands, pp. 227–233.
Mayer, E., Raisch, J., 2004. Time-optimal scheduling for high throughput screening processes using cyclic discrete event models. Mathematics and Computers in Simulation 66, 181–191.
Murray, C., Anderson, C., 1996. Scheduling software for high-throughput screening. Laboratory Robotics and Automation 8 (5), 295–305.
Seo, J.-W., Lee, T.-E., 2002. Steady-state analysis and scheduling of cyclic job shops with overtaking. International Journal of Flexible Manufacturing Systems 14 (4), 291–318.
Vieira, G.E., Herrmann, J.W., Lin, E., 2003. Rescheduling manufacturing systems: a framework of strategies, policies, and methods. Journal of Scheduling 6 (1), 39–62.