Thèse de Doctorat en cotutelle

Thomas Brunsch

Mémoire présenté en vue de l'obtention
du grade de Docteur de l'Université d'Angers
Sous le label de l'Université Nantes Angers Le Mans

Spécialité: Automatique et Génie Informatique
Laboratoire: Laboratoire d'Ingénierie des Systèmes Automatisés EA 4094

Soutenue le 30 janvier 2014

École doctorale: ED STIM
Thèse N°1285

Modeling and Control of Complex Systems in a Dioid Framework

Jury

Rapporteurs:	Jean-Jacques Lesage	Professeur, ENS Cachan
	Thomas Moor	Professeur, Universität Erlangen-Nürnberg
Examinateurs:	Isabel Demongodin	Professeur, Université Aix-Marseille
	Oliver Brock	Professeur, Technische Universität Berlin
Directeurs de thèse:	Laurent Hardouin	Professeur, Université d'Angers
	Jörg Raisch	Professeur, Technische Universität Berlin

Laboratoire d'Ingénierie des Systèmes Automatisés, Université d'Angers
62 Avenue Notre-Dame du Lac, 49000 Angers, France

Fachgebiet Regelungssysteme, Fakultät IV - Elektrotechnik und Informatik
Technische Universität Berlin, Einsteinufer 17, 10587 Berlin, Germany

Modeling and Control of Complex Systems in a Dioid Framework

Angefertigt von
Dipl.-Ing. Thomas Brunsch
geb. 24.06.1982 in Halberstadt

Eingereicht bei der
Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
- Dr.-Ing. -

Promotionsausschuss:

Vorsitzender:	Prof. Dr. Oliver Brock	Technische Universität Berlin
stellv. Vorsitzende:	Prof. Isabel Demongodin	Université Aix-Marseille
Gutachter:	Prof. Dr.-Ing. Jörg Raisch	Technische Universität Berlin
Gutachter:	Prof. Laurent Hardouin	Université d'Angers
Gutachter:	Prof. Dr.-Ing. Thomas Moor	Universität Erlangen-Nürnberg
Gutachter:	Prof. Jean-Jacques Lesage	ENS Cachan

Tag der wissenschaftliche Aussprache: 30. Januar 2014

PhD Thesis

Modeling and Control of Complex Systems in a Dioid Framework

THOMAS BRUNSCH

Thomas Brunsch: *Modeling and Control of Complex Systems in a Dioid Framework*
PhD Thesis, © November 2013

Supervisors:
Prof. Dr.-Ing. Jörg Raisch
Prof. Laurent Hardouin

Locations:
Fachgebiet Regelungssysteme, Fakultät IV - Elektrotechnik und Informatik, Technische Universität Berlin, Einsteinufer 17, 10587 Berlin, Germany

Laboratoire d'Ingénierie des Systèmes Automatisés, Université d'Angers, 62 Avenue Notre-Dame du Lac, 49000 Angers, France

Abstract

Many different kinds of manufacturing systems can be modeled by timed event graphs (TEG), a sub-class of Petri nets. The main advantage of timed event graphs is their linear representation in specific mathematical structures named dioids or idempotent semirings. For linear systems in dioids, in turn, exists an established control theory, which can be used to determine feedback and feedforward controllers. However, if the considered system shall be operated with a re-entrant workflow, resulting in a nested schedule, i.e., a resource may be occupied by the same part more than once and in between these resource allocations another part is processed on this very resource, it is not possible to determine a TEG modeling the system's behavior. Furthermore, in standard Petri nets, and consequently in standard timed event graphs, timing information is always considered to be the minimal time of a (sub-) process. Nevertheless, often manufacturing systems are operated with time windows, i.e., a minimal time is necessary to complete a (sub-) process but, at the same time, a maximal time is given, at which the (sub-) process needs to be finished. Such time windows cannot easily be included in timed event graphs. Similarly, it is rather straightforward to include maximum capacities of (sub-) processes or resources, but not possible to include minimum capacities in timed event graphs. While the issue of time windows has been addressed in various publications, an extension of timed event graphs to model systems with nested schedules or minimum capacities has not been studied.

In this work, we propose an approach to model manufacturing systems with nested schedules. This approach is based on an extension for timed event graphs which, in turn, results in non-causal dioid representations with respect to the standard definition of causality for linear systems in dioids. Consequently, the causality issue for systems with nested schedules is addressed. Eventually, the control theory developed for systems in a dioid framework can be applied to determine suitable controllers. In the second part of this thesis, timed event graphs with constraints are investigated. The constraints include time windows, i.e., minimal and maximal time bounds for some (sub-) processes as well as minimum and maximum capacities. Using results from residuation theory, an algorithm to determine linear systems for extended timed event graphs is developed. Finally, a method is introduced to compute suitable controllers for timed event graphs with the mentioned additional constraints. Using a real world example from high-throughput screening, the applicability of our approach is demonstrated.

Zusammenfassung

Viele Systeme in der Produktions- und Fertigungsindustrie können mit Hilfe von Synchronisationsgraphen, einer Klasse von Petri Netzen, modelliert werden. Vorteil solcher Synchronisationsgraphen ist die Möglichkeit, sie als lineares System in Dioidalgebren (auch idempotente Halbringe genannt) abzubilden. Für solche linearen Systeme in Dioidalgebren existiert wiederum eine etablierte Theorie zur Bestimmung von Steuerungen und Regelungen. Es ist allerdings nicht möglich Systeme mit einem verschachtelten Ablaufplan mit Hilfe von Synchronisationsgraphen zu modellieren. Ein verschachtelter Ablaufplan zeichnet sich dadurch aus, dass mehrere Teilprozesse eines Produkts ein und dieselbe Ressource belegen und diese Ressource in der Zwischenzeit Produktionsschritte anderer Teile durchführt. Desweiteren beziehen sich Zeitinformationen in Petri Netzen immer auf die minimale Zeiten, die Marken in einer Stelle bleiben müssen. Häufig werden in Produktionssystemen jedoch Zeitfenster angegeben, wo neben der minimalen Zeit auch eine maximale Zeit festgelegt wird, zu der der Produktionsschritt abgeschlossen sein muss. Solche Zeitfenster können nicht in (standard) Synchronisationsgraphen modelliert werden. Ähnlich verhält es sich mit Kapazitäten von Teilprozessen. Während maximale Kapazitäten relativ einfach durch Marken in Synchronisationsgraphen modelliert werden können, ist es nicht möglich eine Mindestanzahl von Marken in einer Stelle zu garantieren. Eine Erweiterung von Synchronisationsgraphen im Hinblick auf Zeitfenster ist bereits in mehreren Publikationen untersucht worden. Es gibt jedoch noch keine Studien darüber, wie man verschachtelte Ablaufpläne oder Mindestkapazitäten in Fertigungsanlagen bei der Modellierung berücksichtigen kann.

In dieser Arbeit stellen wir einen Ansatz vor, mit dem man Prozesse mit verschachtelten Ablaufplänen durch eine Erweiterung von Synchronisationsgraphen modellieren kann. Das Modell eines solchen Prozesses resultiert jedoch in einer akausalen Systembeschreibung in Dioidalgebren. Da die modellierten Systeme jedoch kausal sind, wird der Kausalitätsbegriff näher untersucht und erweitert. Mit Hilfe der Regelungstheorie für lineare Systeme in Dioiden ist es dann möglich, geeignete Regler zu entwerfen. Im zweiten Teil dieser Arbeit untersuchen wir Synchronisationsgraphen mit zusätzlichen Nebenbedingungen wie maximalen Bearbeitungszeiten und Mindestkapazitäten. Durch Nutzung der Residuentheorie ist es möglich, ein lineares Dioidsystem für Synchronisationsgraphen mit Nebenbedingungen zu bestimmen. Schließlich wird ein Algorithmus zum Entwerfen geeigneter Regler vorgestellt. Die Anwendbarkeit unserer Ergebnisse wird anhand realer Systeme aus dem Bereich des Hochdurchsatz-Screenings demonstriert.

Résumé

De nombreux systèmes, notamment manufacturiers, peuvent être modélisés par des graphes d'événements temporisés, une classe particulière de réseaux de Petri. Un avantage majeur de cette approche est l'existence d'une représentation linéaire dans certains dioïdes (ou semi-anneaux idempotents). Cette caractéristique a permis le développement d'une théorie du contrôle dédiée aux graphes d'événements temporisés. Il n'est cependant pas possible de représenter certains modes de fonctionnement, dits imbriqués, par des graphes d'événemts temporisés. Par imbriqué, nous entendons qu'une ressource effectue plusieurs tâches sur un même produit, mais, qu'entre certaines de ces tâches, elle peut être affectée à la réalisation de tâches sur d'autres produits. De plus, alors qu'il est facile de spécifier dans les graphes d'événements temporisés un temps minimal pour une tâche, il n'est pas possible de définir un temps maximal. Pourtant, dans de nombreuses applications, la durée d'une tâche doit être comprise dans un intervalle. Ce type de spécifications ne peut donc pas être inclus dans les graphes d'événements temporisés classiques. Il en va de même pour le nombre de marques dans une place : le nombre maximal de marques dans une place peut être modélisé, mais pas le nombre minimal de marques. Contrairement au problème relatif au temps maximal pour une tâche, le problème relatif au mode de fonctionnement imbriqué et au nombre minimal de marques dans une place n'a pas encore été traité.

Dans ce mémoire, la modélisation de systèmes avec un mode de fonctionnement imbriqué est abordée. Notre approche repose sur la présence dans le modèle d'éléments non causaux selon la définition classique de la causalité pour les systèmes linéaires dans les dioïdes. De fait, le problème de la causalité est traité pour les modes de fonctionnement imbriqués. L'automatique dédiée aux graphes d'événements temporisés est étendue aux systèmes avec un mode de fonctionnement imbriqué. Dans la deuxième partie de ce mémoire, la modélisation de graphes d'événements temporisés avec contraintes (temps mininal et maximal et nombre minimal et maximal de marques pour chaque place) est abordée. En utilisant la théorie de la résiduation, une représentation linéaire est obtenue et une méthode est présentée pour étendre certaines commandes aux graphes d'événements temporisés avec contraintes. Pour finir, l'intérêt de ce travail est illustré au moyen d'un exemple industriellement pertinent : un système de test à haut débit de produits pharmacologiques provenant de notre partenaire industriel.

Publications

Some ideas and figures have appeared previously in the following publications:

[BHMR12] T. Brunsch, L. Hardouin, C. A. Maia, and J. Raisch. Duality and interval analysis over idempotent semirings. *Linear Algebra and its Applications*, 437(10):2436–2454, November 2012.

[BHR10] T. Brunsch, L. Hardouin, and J. Raisch. Control of cyclically operated high-throughput screening systems. In *Preprints of the 10th International Workshop on Discrete Event Systems (WODES'10)*, pages 177–182, Berlin, Germany, September 2010.

[BHR11] T. Brunsch, L. Hardouin, and J. Raisch. Modeling and control of nested manufacturing processes using dioid models. In *Preprints of the 3rd International Workshop on Dependable Control of Discrete Systems (DCDS11)*, pages 78–83, Saarbrücken, Germany, June 2011.

[BHR13] T. Brunsch, L. Hardouin, and J. Raisch. Modeling manufacturing systems in a dioid framework. In J. Campos, C. Seatzu, and X. Xie, editors, *Formal Methods in Manufacturing*, Series on Industrial Information Technology, chapter 2, pages 29–74. CRC Press/Taylor and Francis, 2013. Accepted for publication.

[BR09a] T. Brunsch and J. Raisch. Max-plus algebraic modeling and control of high-throughput screening systems. In *Preprints of the 2nd IFAC Workshop on Dependable Control of Discrete Systems*, pages 103–108, Bari, Italy, June 2009.

[BR09b] T. Brunsch and J. Raisch. 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)*, pages 132–137, Zaragoza, Spain, September 2009.

[BR12] T. Brunsch and J. Raisch. Modeling and control of high-throughput screening systems in a max-plus algebraic setting. *Engineering Applications of Artificial Intelligence*, 25(4):720–727, June 2012.

[BRH12] T. Brunsch, J. Raisch, and L. Hardouin. Modeling and control of high-throughput screening systems. *Control Engineering Practice*, 20(1):14–23, January 2012.

[BRHB13] T. Brunsch, J. Raisch, L. Hardouin, and O. Boutin. Discrete-event systems in a dioid framework: Modeling and analysis. In Carla Seatzu, Manuel Silva, and Jan H. van Schuppen, editors, *Control of Discrete-Event Systems*, volume 433 of *Lecture Notes in Control and Information Sciences*, chapter 21, pages 431–450. Springer Berlin / Heidelberg, 2013.

[HBC+13] L. Hardouin, O. Boutin, B. Cottenceau, T. Brunsch, and J. Raisch. Discrete-event systems in a dioid framework: Control theory. In Carla Seatzu, Manuel Silva, and Jan H. van Schuppen, editors, *Control of Discrete-Event Systems*, volume 433 of *Lecture Notes in Control and Information Sciences*, chapter 22, pages 451–469. Springer Berlin / Heidelberg, 2013.

Experience is what you get
when you didn't get what you wanted.

— Randy Pausch

Acknowledgments

At this point I would like to say *thank you* to all the people, who have contributed to this thesis, be it by discussion, inspiration, or different kinds of support.

In particular, I would like to thank my supervisor, Prof. Dr.-Ing. Jörg Raisch, for giving me the opportunity to work in such an interesting field of research, for many discussions of my work, for his faith in me and my work, and for introducing me to so many nice people.

In the same manner, I am greatly indebted to my French supervisor, Prof. Laurent Hardouin, for agreeing to supervise me in the first place, for supporting me in so many different situations, for handling all the administrational stuff at Université d'Angers, and for caring way beyond the professional aspects of my work.

A big *thank you* deserve the members of the Control Systems Group at TU Berlin and the Systems and Control Theory Group at the Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, for letting me be part of the group, for many discussions and support during my research. In particular, I would like to thank Xavier, Behrang, Darina, Thomas, and of course our "coffee club".

On the French side I'd like to say *merci beaucoup* to the group of the Laboratoire d'Ingénierie des Systèmes Automatisés for welcoming me every so often in their group, for always trying to make me feel home, and for not minding to speak English. A *grand merci* go to Euriell, Guillaume, Julien, Jean-Luc, and Agnès for making every visit in Angers very special.

I would also like to express my deep gratitude to Janine Holzmann and Ulrike Locherer, for smoothing the way towards my PhD degree with respect to many administrational issues.

A special *thank you* goes to Thomas Haehnel from CyBio AG, for giving lots of inside information on High-Throughput Screening systems, for providing lots of data (in my preferred formatting), for showing interest in my work, and for many discussion on how to implement my ideas on real HTS systems.

Barbara Schenk deserves an extra *thank you* for her limitless patience, love and support.

Last but not least, I would like to thank my family and close friends for their love and support, not only during the course of my PhD, but throughout my whole life.

Berlin, November, 2013

T.B.

Contents

List of Figures

1 Introduction

A dynamical system is called *discrete-event system* (DES) if its behavior can be completely described by discrete-valued signals, or sequences of discrete events. If timing information is explicitly included, it is a timed discrete-event system. Otherwise, it is referred to as untimed, or logical, DES.

Within the last decades many different modeling approaches for DES have been developed. Among the best studied approaches are *Finite-State Automata* (e.g., [6, 32]), *Petri nets* (e.g., [49, 55]), and *Markov Chains* (e.g., [5, 23]). A broad overview of different modeling approaches for discrete-event systems can be found in, e.g., [11]. Depending on the specific process or system to be modeled an approach may be better suitable than another, but which approach is eventually applied also strongly depends on the personal biases of the user.

According to specific features of the system, many extensions for the standard approaches have been introduced. For example, colored Petri nets have been introduced to model and validate concurrent and distributed systems [34]. Also timed Petri nets can be seen as an extension of the original (logical) Petri nets.

In some cases, however, it is beneficial to restrict oneself to a sub-class of a specific modeling approach. If the characteristic properties of such a sub-class are sufficient to handle the systems of interest, the restriction to this sub-class may simplify the modeling approach and increase the applicability. One specific sub-class of Petri nets is, for example, the class of so-called state machines. State machines cannot model synchronization effects, but are perfectly suitable to model conflicts or decisions [54]. The counterpart to state machines are the so-called marked graphs or event graphs. Marked graphs constitute a sub-class of Petri nets that cannot model

conflicts but are a strong tool to model synchronization phenomena [16]. Many transportation networks, for example, can be modeled with event graphs. In such networks, different transportation devices are synchronized but due to the fixed timetable no decisions have to be made. Also manufacturing processes are often subject to synchronization, e.g., the welding of two parts cannot start before both parts are available. Of course, usually time cannot be regarded as negligible with respect to the modeling of transportation networks, manufacturing systems, or other systems. Therefore, event graphs have been extended to *timed event graphs (TEG)*, which explicitly contain timing information for the different events [12]. Of course, timed event graphs can also be seen as a sub-class of timed Petri nets. A fundamental advantage of TEG compared to general Petri nets is that TEG have a linear representation in specific mathematical structures, namely, in *idempotent semirings* or *dioids* [2, 21]. Consequently, the dynamical behavior of a TEG can be described by a linear system in a dioid framework, which is very similar to linear systems well-known from classical systems theory. Furthermore, a fundamental control theory for such linear systems in dioids is available and can be used to determine feedforward and feedback controllers. Among the control approaches studied for linear systems in dioids are optimal open loop control [2, 48], optimal input filtering [20, 26], state and output feedback control [17, 19, 20], disturbance decoupling and robust control in case of uncertainties [39, 40, 41], and model predictive control strategies [58, 61]. By applying a suitable controller, the corresponding system can be "steered" towards a desired behavior. In manufacturing industries, for example, this desired behavior is often a just-in-time operation, which minimizes the necessary stock while guaranteeing a certain throughput.

However, the time included in timed event graphs corresponds to the minimal time a token has to spend in a certain place before it can contribute to the firing of the output transition of this place. This is oftentimes not sufficient to precisely model the behavior of the desired system. Especially in chemical industries, an upper time bound, which, with respect to TEG, corresponds to a maximal time a token may stay in a place, is essential to obtain an accurate model of the underlying system. This issue has been addressed in [51, 52, 53], where a control law for systems with sojourn time constraints is obtained by using residuated and dual residuated mappings. Similar problems have recently been addressed in [1, 33, 42] where different approaches have been developed to handle TEG with additional temporal constraints.

Furthermore, due to the definition of timed event graphs, the number of tokens in a place is non-negative at any time. This, however, means that a system operating with a re-entrant workflow, i.e., a workpiece may be processed more than once at the same resource, cannot be represented by a linear system in a dioid setting, if the resource performing two (or more) processing steps on one part is occupied by another part in between these two processing steps. Similarly, a system cannot be modeled by a TEG or, equivalently, by a linear system in dioids, if a sub-process shall be executed (at least) a certain number of times more often than another sub-process of the system. With respect to timed event graphs, such a behavior can be modeled by a TEG which allows an integer number of tokens in a place, i.e., a negative number of tokens may be present in some places. Negative tokens have been previously introduced in [50] for the application to automated reasoning. The basic idea in [50] is that each place (of

a Petri net) represents a proposition and a positive (resp. negative) token in a place means that the corresponding proposition (resp. the negative of the corresponding proposition) has been deduced. Consequently, this notion of a negative token does not correspond to "our idea" of a negative number of tokens. In [37, 38] the authors introduce negative places and negative tokens to model time window constraints, i.e., negative tokens in negative places have a negative holding time and basically model the upper time bound between the firing of two consecutive transitions. Similarly, maximal times have been modeled in [43, 44, 45], where an algorithm to determine globally optimal schedules for cyclic systems with non-blocking specification and time window constraints has been introduced. Nonetheless, negative tokens in [37, 38] are only used to model upper bounds of time windows, while our work proposes to use a negative number of tokens to model the least number of tokens which have to be present in a specific place of a TEG.

To model systems with minimal and maximal timing information as well as minimal and maximal numbers of tokens in a place, an extension of timed event graphs is proposed in this thesis. This extension includes the possibility to define time windows, in which the token has to leave the place, as well as bounds on the number of tokens in places. It is shown how such an extended timed event graph can be converted into a linear system in a dioid framework and the definitions on causality, rationality, and realizability are adapted accordingly. Finally, an algorithm to determine suitable feedforward and feedback controllers for the extended systems is introduced and the described approaches are applied to real world examples of so-called High-Throughput Screening (HTS) systems.

This thesis is structured as follows:

Chapter 2 provides a broad overview of the algebraic preliminaries used in this thesis. This includes order relations and ordered sets as well as idempotent semirings, and results from residuation theory. Furthermore, the dioid $\mathcal{M}_{\text{in}}^{\text{ax}}[\![\gamma, \delta]\!]$, an idempotent semiring of formal power series, which is used throughout this thesis, is defined.

Chapter 3 gives a brief introduction to timed event graphs and how a dioid model of such timed event graphs can be obtained. Using a simple example, the necessity of an extension of timed event graphs is motivated and the notion of negative numbers of tokens is presented. The concepts of causality, realizability, and rationality are re-defined with respect to integer numbers of tokens.

Chapter 4 shortly summarizes different control approaches for linear systems in a dioid framework. Since the standard definition of the dioid $\mathcal{M}_{\text{in}}^{\text{ax}}[\![\gamma, \delta]\!]$ includes integer exponents the mentioned approaches are identical for timed event graphs with and without negative numbers of tokens.

Chapter 5 describes how additional constraints, namely an upper bound on the time and a minimal number of tokens (with respect to timed event graphs), can be included in a dioid model. It is shown how a system matrix containing the information on all imposed constraints can be derived and how suitable pre-filters and feedback controllers can be

determined. Furthermore, several special cases for timed event graphs with constraints are discussed.

Chapter 6 introduces high-throughput screening systems, which are mainly used in the pharmaceutical industries. As HTS systems are often operated with nested schedules and (bio-)chemical reactions play a major role, HTS is one example, where the described constraints and extensions are frequently necessary. The applicability of our modeling and control approach is demonstrated on a real world HTS system.

Chapter 7 draws conclusions of the introduced modeling and control methods for systems with the aforementioned additional constraints and extensions.

2
Mathematical Background

This section provides the algebraic preliminaries used in this thesis and has partially been published in [8]. For a more exhaustive presentation of the mathematical matters please refer to [2].

2.1 Order relation and ordered sets

Definition 2.1 (Equivalence relation). A binary relation $\sim$ on a set $\mathcal{C}$ is an *equivalence relation* if and only if it is reflexive, symmetric and transitive, i.e., $\forall a, b, c \in \mathcal{C}$:

- reflexivity: $a \sim a$
- symmetry: if $a \sim b$ then $b \sim a$
- transitivity: if $a \sim b$ and $b \sim c$ then $a \sim c$

Definition 2.2 (Order relation). An *order relation* $\preceq$ on a set $\mathcal{C}$ is a binary relation for which the following properties hold $\forall a, b, c \in \mathcal{C}$:

- reflexivity: $a \preceq a$
- anti-symmetry: $(a \preceq b \text{ and } b \preceq a) \Rightarrow a = b$
- transitivity: $(a \preceq b \text{ and } b \preceq c) \Rightarrow a \preceq c$

Definition 2.3 (Ordered set). A set $\mathcal{C}$ endowed with an order relation $\preceq$ is said to be an *ordered set* and denoted $(\mathcal{C}, \preceq)$. If any pair of elements of $\mathcal{C}$ can be compared with respect to $\preceq$, i.e., $\forall a, b \in \mathcal{C}$ one can either write $a \preceq b$ or $b \preceq a$, the ordered set $(\mathcal{C}, \preceq)$ is said to be a *totally ordered set.* If, however, a pair of elements of $\mathcal{C}$ exists, for which one can neither write $a \preceq b$

nor $b \preceq a$, i.e., $a \npreceq b$ and $b \npreceq a$, the order relation is said to be partial on $\mathcal{C}$ and $(\mathcal{C}, \preceq)$ is said to be a *partially ordered set.*

Example 2.4 (Ordered sets). Classical examples of totally ordered sets are real and integer numbers with respect to the classical $\leqslant$, i.e., $(\mathbb{R}, \leqslant)$ and $(\mathbb{Z}, \leqslant)$. If, however, a set of vectors, e.g., $\mathbb{Z}^{2\times 1}$, is considered, the resulting $(\mathbb{Z}^{2\times 1}, \leqslant)$ is only partially ordered (if the order relation is applied element wise). This can easily be demonstrated, if one assumes two vectors $v_1 = [a, b]^T$ and $v_2 = [b, a]^T$ with $a, b \in \mathbb{Z}$ and $a \neq b$. Clearly, it is not possible to relate these to vectors with respect to the order relation $\leqslant$, i.e., one can neither write $v_1 \leqslant v_2$ nor $v_2 \leqslant v_1$.

Definition 2.5 (Bounds on ordered sets). Given an ordered set $(\mathcal{C}, \preceq)$ and a (non-empty) subset $\mathcal{S} \subseteq \mathcal{C}$, an element $a \in \mathcal{C}$ is called *lower bound* of $\mathcal{S}$ if $\forall b \in \mathcal{S} : a \preceq b$. Similarly, if an element $c \in \mathcal{C}$ exists, such that $\forall b \in \mathcal{S} : b \preceq c$, c is called *upper bound* of $\mathcal{S}$. Given that a subset $\mathcal{S}$ has a lower bound, its *greatest lower bound (glb)* is denoted $\bigwedge \mathcal{S}$. Likewise, if an upper bound exists, its *least upper bound (lub)* of a subset $\mathcal{S}$ is denoted $\bigvee \mathcal{S}$.

Definition 2.6 (Lattices). An ordered set $(\mathcal{C}, \preceq)$ is called a *sup-semi-lattice*, if $\forall a, b \in \mathcal{C}$ there exists $a \vee b$. It is a *complete sup-semi-lattice*, if every subset $\mathcal{S} \subseteq \mathcal{C}$ admits a least upper bound, i.e., $\bigvee \mathcal{S}$ exists $\forall \mathcal{S} \subseteq \mathcal{C}$. Analogously, an ordered set $(\mathcal{C}, \preceq)$ is called an *inf-semi-lattice*, if $\forall a, b \in \mathcal{C}$ there exists $a \wedge b$, and it is called a *complete inf-semi-lattice*, if every subset $\mathcal{S} \subseteq \mathcal{C}$ has a greatest lower bound, i.e., $\bigwedge \mathcal{S}$ exists $\forall \mathcal{S} \subseteq \mathcal{C}$. An ordered set $(\mathcal{C}, \preceq)$ is called a *lattice*, if it is both a sup-semi-lattice and an inf-semi-lattice and denoted $(\mathcal{C}, \vee, \wedge)$. It is called a *complete lattice* if it is a complete sup-semi-lattice and a complete inf-semi-lattice. A complete lattice is also called a bounded lattice and the bounds are denoted $\top$ (top element) and $\bot$ (bottom element). If an ordered set $(\mathcal{C}, \preceq)$ forms a lattice, the following properties hold $\forall a, b \in \mathcal{C}$:

$$a \preceq b \Leftrightarrow a \vee b = b \Leftrightarrow a \wedge b = a.$$

Furthermore, in a lattice $(\mathcal{C}, \vee, \wedge)$ the operations $\vee$ and $\wedge$ are associative, commutative, idempotent, and the absorption property, i.e., $a \vee (a \wedge b) = a$ and $a \wedge (a \vee b) = a$, $\forall a, b \in \mathcal{C}$, holds. A lattice is said to be distributive if $\wedge$ distributes over $\vee$. In general the following inequalities hold for all lattices $(\mathcal{C}, \vee, \wedge)$:

$$\begin{aligned} a \vee (b \wedge c) &\preceq (a \vee b) \wedge (a \vee c) \\ a \wedge (b \vee c) &\succeq (a \wedge b) \vee (a \wedge c) \qquad \forall a, b, c \in \mathcal{C}. \end{aligned}$$

Example 2.7 (Lattices). The (totally) ordered set of integers $(\mathbb{Z}, \leqslant)$ forms a lattice $(\mathbb{Z}, \vee, \wedge)$ but not a complete lattice. If, however, one "adds" $+\infty$ and $-\infty$ to the set, $(\mathbb{Z} \cup \{-\infty, +\infty\}, \leqslant)$ forms a complete lattice.

2.1.1 Mappings in ordered sets

Definition 2.8 (Isotone mapping). A mapping Π from an ordered set $(\mathcal{C}, \preceq)$ to an ordered set $(\mathcal{D}, \preceq)$, i.e., $\Pi : \mathcal{C} \to \mathcal{D}$ is said to be order preserving or *isotone* if the following property holds

$$a \preceq b \Rightarrow \Pi(a) \preceq \Pi(b), \quad \forall a, b \in \mathcal{C}.$$

Definition 2.9 (Antitone mapping). Similarly, a mapping $\Pi : \mathcal{C} \to \mathcal{D}$ is *antitone* if the mapping "inverts" the order, i.e., the following property holds

$$a \preceq b \Rightarrow \Pi(a) \succeq \Pi(b), \quad \forall a, b \in \mathcal{C}.$$

Definition 2.10 (Monotone mapping). In general, a mapping Π is said to be monotone if it is either isotone or antitone.

Definition 2.11 (Semi-continuous mapping). A mapping Π from a complete lattice $(\mathcal{C}, \vee, \wedge)$ to a complete lattice $(\mathcal{D}, \vee, \wedge)$ is *lower semi-continuous (l.s.c.)* if for any $\mathcal{S} \subseteq \mathcal{C}$ one can write

$$\Pi\left(\bigvee_{a \in \mathcal{S}} a\right) = \bigvee_{a \in \mathcal{S}} \Pi(a).$$

Analogously, a mapping from a complete lattice to a complete lattice, $\Pi : \mathcal{C} \to \mathcal{D}$, is said to be *upper semi-continuous (u.s.c.)* if one can write

$$\Pi\left(\bigwedge_{a \in \mathcal{S}} a\right) = \bigwedge_{a \in \mathcal{S}} \Pi(a), \quad \forall \mathcal{S} \subseteq \mathcal{C}.$$

Definition 2.12 (Continuous mapping). A mapping Π is said to be continuous if it is lower semi-continuous as well as upper semi-continuous.

Furthermore, given two complete lattices $(\mathcal{C}, \vee, \wedge)$ and $(\mathcal{D}, \vee, \wedge)$, it can easily be shown that a l.s.c. mapping $\Pi : \mathcal{C} \to \mathcal{D}$ is isotone by considering arbitrary $a, b \in \mathcal{C}$. If $a \preceq b$, then $a \vee b = b$ and since Π is l.s.c., $\Pi(a \vee b) = \Pi(a) \vee \Pi(b) = \Pi(b)$ and therefore $\Pi(a) \preceq \Pi(b)$. Additionally, it can be shown that an u.s.c. mapping $\Pi : \mathcal{C} \to \mathcal{D}$ is also isotone. Considering two arbitrary elements $a, b \in \mathcal{C}$ with $a \preceq b$ one can write $a \wedge b = a$ and since Π is u.s.c. it is clear that $\Pi(a \wedge b) = \Pi(a) \wedge \Pi(b) = \Pi(a)$ and thus $\Pi(a) \preceq \Pi(b)$.

Remark 2.13. If a mapping $\Pi : \mathcal{C} \to \mathcal{D}$ is isotone, but neither lower nor upper semi-continuous, the following inequalities hold $\forall a, b \in \mathcal{C}$:

$$\Pi(a \vee b) \succeq \Pi(a) \vee \Pi(b)$$
$$\Pi(a \wedge b) \preceq \Pi(a) \wedge \Pi(b).$$

2.2 Idempotent semirings

Definition 2.14 (Monoid). A *monoid*, $(M, \cdot, e)$, is a set M endowed with an internal law $\cdot$, which is associative and with an identity element e. If the internal law $\cdot$ is commutative, $(M, \cdot, e)$ is said to be a commutative monoid. If the internal law $\cdot$ is idempotent, i.e., $a \cdot a = a \; \forall a \in M$, the monoid is said to be idempotent.

Definition 2.15 (Dioid). An *idempotent semiring* (also called *dioid*) is a set $\mathcal{D}$, endowed with two internal operations denoted $\oplus$ (addition) and $\otimes$ (multiplication) such that $(\mathcal{D}, \oplus, \varepsilon)$ constitutes an idempotent commutative monoid and $(\mathcal{D}, \otimes, e)$ constitutes a monoid. Additionally, multiplication is left- and right-distributive with respect to addition, i.e., $a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$ and $(a \oplus b) \otimes c = (a \otimes c) \oplus (b \otimes c) \;\; \forall a, b, c \in \mathcal{D}$, and ε is absorbing with respect to $\otimes$, i.e., $a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon \; \forall a \in \mathcal{D}$. The idempotent semiring is denoted $(\mathcal{D}, \oplus, \otimes)$ and the neutral elements of addition and multiplication are often referred to as zero and unit element, respectively. If multiplication is commutative, the corresponding idempotent semiring is said to be commutative. Furthermore, if all elements of a dioid (except ε) have a multiplicative inverse, the idempotent semiring forms an *idempotent semifield*.

Consequently, the following properties hold $\forall a, b, c \in \mathcal{D}$
- addition:
  - associativity: $(a \oplus b) \oplus c = a \oplus (b \oplus c)$
  - commutativity: $a \oplus b = b \oplus a$
  - idempotency: $a \oplus a = a$
  - neutral element: $a \oplus \varepsilon = a$
- multiplication
  - associativity: $(a \otimes b) \otimes c = a \otimes (b \otimes c)$
  - neutral element: $a \otimes e = e \otimes a = a$

Remark 2.16. As in standard algebra, the multiplication sign $\otimes$ is often omitted when unambiguous.

Example 2.17 (Max-plus algebra). One of the most widely known idempotent semirings is the so-called $(\max, +)$-*algebra* denoted $(\mathbb{Z}_{\max}, \oplus, \otimes)$. It is defined on the set $\mathbb{Z}_{\max} = \mathbb{Z} \cup \{-\infty\}$. In $(\max, +)$-algebra, addition is defined as the standard maximum operation and multiplication is the standard addition, i.e., $a \oplus b := \max(a, b)$ and $a \otimes b := a + b$, respectively. The zero element of $(\max, +)$-algebra is $\varepsilon = -\infty$ and the unit element is $e = 0$. Instead of writing $(\mathbb{Z}_{\max}, \oplus, \otimes)$ this idempotent semiring is often denoted $\mathbb{Z}_{\max}$. Note that $(\max, +)$-algebra may also be defined on the set of real numbers, i.e., $\mathbb{R}_{\max} = \mathbb{R} \cup \{-\infty\}$.

Example 2.18 (Min-plus algebra). Similar to the $(\max, +)$-algebra is the so-called $(\min, +)$-algebra. It is defined on the set $\mathbb{Z}_{\min} = \mathbb{Z} \cup \{+\infty\}$ and denoted $(\mathbb{Z}_{\min}, \oplus, \otimes)$ (or simply $\mathbb{Z}_{\min}$). Min-plus addition is defined as the standard minimum operation and multiplication in $(\min, +)$-algebra is the standard addition, i.e., $a \oplus b := \min(a, b)$ and $a \otimes b := a + b \; \forall a, b \in$

$\mathbb{Z}_{\min}$. The zero and unit element of $(\min, +)$-algebra are $\varepsilon = +\infty$ and $e = 0$, respectively. Analogical to $(\max, +)$-algebra also $(\min, +)$-algebra may be defined on real numbers instead of integers, i.e., $\mathbb{R}_{\min} = \mathbb{R} \cup \{+\infty\}$.

Definition 2.19 (Sub-semiring). Given an idempotent semiring $(\mathcal{D}, \oplus, \otimes)$ and a subset $\mathcal{S} \subseteq \mathcal{D}$, then $(\mathcal{S}, \oplus, \otimes)$ is called a *sub-semiring* of $(\mathcal{D}, \oplus, \otimes)$, if:

- the zero and unit elements of $(\mathcal{D}, \oplus, \otimes)$ are included in $\mathcal{S}$, i.e., $\varepsilon \in \mathcal{S}$ and $e \in \mathcal{S}$,
- $\mathcal{S}$ is closed for $\oplus$ and $\otimes$, i.e., $\forall a, b \in \mathcal{S}$, $a \oplus b \in \mathcal{S}$ and $a \otimes b \in \mathcal{S}$.

Remark 2.20. According to this definition, it is clear that $\mathbb{Z}_{\max}$ is a sub-semiring of $\mathbb{R}_{\max}$ and $\mathbb{Z}_{\min}$ is a sub-semiring of $\mathbb{R}_{\min}$.

Definition 2.21 (Addition and multiplication of matrices with entries in dioids). As in classical algebra, the operations of idempotent semirings can be extended to matrices of compatible dimensions with entries in dioids. Given an idempotent semiring $(\mathcal{D}, \oplus, \otimes)$, two $n \times p$ matrices A, B, and one $p \times m$ matrix C with entries in $\mathcal{D}$, i.e., $[A]_{ij} \in \mathcal{D}, [B]_{ij} \in \mathcal{D}, [C]_{jk} \in \mathcal{D}, \forall i, j, k$, addition and multiplication are defined as follows

$$A \oplus B: \quad [A \oplus B]_{ij} = [A]_{ij} \oplus [B]_{ij} \qquad \forall i = 1, \ldots, n; \forall j = 1, \ldots, p;$$

$$A \otimes C: \quad [A \otimes C]_{ij} = \bigoplus_{k=1}^{p} [A]_{ik} \otimes [C]_{kj} \qquad \forall i = 1, \ldots, n; \forall j = 1, \ldots, m.$$

Definition 2.22 (Dioid of matrices). An idempotent semiring of square matrices is denoted $(\mathcal{D}^{n\times n}, \oplus, \otimes)$ and addition and multiplication are defined for $A, B \in \mathcal{D}^{n\times n}$:

$$A \oplus B: \quad [A \oplus B]_{ij} = [A]_{ij} \oplus [B]_{ij} \qquad \forall i, j = 1, \ldots, n;$$

$$A \otimes B: \quad [A \otimes B]_{ij} = \bigoplus_{k=1}^{n} [A]_{ik} \otimes [B]_{kj} \qquad \forall i, j = 1, \ldots, n.$$

The unit element of $(\mathcal{D}^{n\times n}, \oplus, \otimes)$, also called the identity matrix, is denoted I with $[I]_{ii} = e$ and $[I]_{ij} = \varepsilon$, if $i \neq j$. The zero matrix of the dioid is denoted $\mathcal{E}$ with $[\mathcal{E}]_{ij} = \varepsilon, \forall i, j = 1, \ldots, n$.

To include non-square matrices of different dimensions in an idempotent semiring of matrices, one has to consider the idempotent semiring of square matrices with the dimension $\max(n, p, m) \times \max(n, p, m)$ and set all elements of the corresponding "additional" rows and columns of matrices to ε.

2.2.1 Natural order in idempotent semirings

Due to the idempotent character of addition in idempotent semirings, they are naturally equipped with an order:

$$a \oplus b = b \Leftrightarrow a \preceq b \qquad \forall a, b \in \mathcal{D}.$$

It can be easily checked that $\preceq$ is indeed reflexive, anti-symmetric, and transitive.

Definition 2.23 (Complete dioids). An idempotent semiring is said to be a *complete*, if it is closed for infinite sums and if $\otimes$ distributes over infinite sums, i.e., $\forall a \in \mathcal{D}$ and $\forall \mathcal{S} \subseteq \mathcal{D}$:

$$a \otimes \left(\bigoplus_{b \in \mathcal{S}} b \right) = \bigoplus_{b \in \mathcal{S}} (a \otimes b) \quad \text{and} \quad \left(\bigoplus_{b \in \mathcal{S}} b \right) \otimes a = \bigoplus_{b \in \mathcal{S}} (b \otimes a) .$$

Thus, a complete dioid admits a greatest element, the so-called *top element* $\top$, which corresponds to the sum of all elements in $\mathcal{D}$, i.e., $\top = \bigoplus_{x \in \mathcal{D}} x$ and $\top \in \mathcal{D}$.

Remark 2.24. With respect to lattice theory, a dioid constitutes an ordered set $(\mathcal{D}, \preceq)$ with the structure of a sup-semi-lattice and with $a \oplus b$ being the least upper bound of a and b. A complete dioid has the structure of a complete sup-semi-lattice with $\top = \bigvee \mathcal{D} = \bigoplus_{x \in \mathcal{D}} x$. Moreover, since every dioid always admits $\varepsilon \in \mathcal{D}$ as the bottom element, i.e., $\varepsilon = \bigwedge \mathcal{D}$, a complete dioid has the structure of a complete lattice $(\mathcal{D}, \oplus, \wedge)$ with

$$a \oplus b = b \Leftrightarrow a \preceq b \Leftrightarrow a \wedge b = a.$$

Example 2.25 (Max-plus algebra). The natural order in $(\max, +)$-algebra $(\mathbb{Z}_{\max}, \oplus, \otimes)$ coincides with the order relation defined in classical algebra, e.g., $3 \oplus 4 = 4 \Leftrightarrow 3 \preceq 4$. As defined above $(\mathbb{Z}_{\max}, \oplus, \otimes)$ does not constitute a complete dioid since $\top = +\infty \notin \mathbb{Z}_{\max}$. If, however, one defines $(\max, +)$-algebra on the set $\overline{\mathbb{Z}}_{\max} = \mathbb{Z}_{\max} \cup \{+\infty\} = \mathbb{Z} \cup \{-\infty, +\infty\}$, it becomes a complete dioid and, therefore, $(\overline{\mathbb{Z}}_{\max}, \oplus, \wedge)$ is a complete lattice.

Example 2.26 (Min-plus algebra). The natural order in $(\min, +)$-algebra $(\mathbb{Z}_{\min}, \oplus, \otimes)$ corresponds to the "reverse" of the order relation in classical algebra, e.g., $3 \oplus 4 = 3 \Leftrightarrow 4 \preceq 3$. Similar to $(\max, +)$-algebra, $(\min, +)$-algebra does not constitute a complete dioid, when defined on the set $\mathbb{Z}_{\min}$. If, however, it is defined on the set $\overline{\mathbb{Z}}_{\min} = \mathbb{Z}_{\min} \cup \{-\infty\} = \mathbb{Z} \cup \{-\infty, +\infty\}$, it becomes a complete dioid (with $\top = -\infty$) and, therefore, $(\overline{\mathbb{Z}}_{\min}, \oplus, \wedge)$ is also a complete lattice.

Definition 2.27 (Rational closure [2, 15]). Given a complete dioid $(\mathcal{D}, \oplus, \otimes)$ and a subset $\mathcal{S} \subseteq \mathcal{D}$, with $\mathcal{S}$ containing e and ε. The rational closure of $\mathcal{S}$, denoted $(\mathcal{S}^*, \oplus, \otimes)$, is the smallest sub-semiring of $(\mathcal{D}, \oplus, \otimes)$ containing all elements of $\mathcal{S}$ and all finite sums, products, and star operations over its elements. The subset $\mathcal{S}$ is rationally closed if $\mathcal{S} = \mathcal{S}^*$.

2.2.2 Mappings in idempotent semirings

As idempotent semirings constitute ordered sets, the properties of monotonicity introduced in Sec. 2.1.1, also apply for mappings in dioids. For complete dioids (which form complete lattices) also the concept of continuity can be applied, i.e., a mapping Π from a complete dioid $(\mathcal{D}, \oplus, \otimes)$ to a complete dioid $(\mathcal{C}, \oplus, \otimes)$ is lower semi-continuous if $\forall \mathcal{S} \subseteq \mathcal{D}$:

$$\Pi \left(\bigoplus_{a \in \mathcal{S}} a \right) = \bigoplus_{a \in \mathcal{S}} \Pi(a),$$

and upper semi-continuous if $\forall \mathcal{S} \subseteq \mathcal{D}$:

$$\Pi\left(\bigwedge_{a\in\mathcal{S}} a\right) = \bigwedge_{a\in\mathcal{S}} \Pi(a).$$

Definition 2.28 (Homomorphism). A mapping $\Pi : \mathcal{D} \to \mathcal{C}$ defined on idempotent semirings is a *homomorphism* if $\forall a, b \in \mathcal{D}$

$$\begin{aligned} \Pi(a \oplus b) &= \Pi(a) \oplus \Pi(b) \quad \text{and} \quad \Pi(\varepsilon) = \varepsilon \\ \Pi(a \otimes b) &= \Pi(a) \otimes \Pi(b) \quad \text{and} \quad \Pi(e) = e, \end{aligned}$$

where the same symbols are used for the zero and unit elements of $\mathcal{D}$ and $\mathcal{C}$.

Definition 2.29 (Isomorphism). If the inverse of a homomorphism Π is defined and is itself a homomorphism the mapping Π is called an *isomorphism*.

Definition 2.30 (Image of a mapping). The *image* of a mapping $\Pi : \mathcal{D} \to \mathcal{C}$ is denoted $\mathrm{Im}\Pi$ (sometimes also $\Pi(\mathcal{D})$) is defined by

$$\mathrm{Im}\Pi = \{b \in \mathcal{C} \,|\, b = \Pi(a),\ a \in \mathcal{D}\}.$$

Definition 2.31 (Kernel of a mapping). The *kernel* of a mapping $\Pi : \mathcal{D} \to \mathcal{C}$ is denoted $\ker \Pi$ and defined by

$$\ker \Pi = \{(a, b) \in \mathcal{D} \times \mathcal{D} : \Pi(a) = \Pi(b)\}.$$

Remark 2.32. Please note that, while the definition on the image of a mapping in dioids (Def. 2.30) is equivalent to the corresponding definition in conventional algebra, the kernel of a mapping in dioids is not identical to the kernel of a mapping in conventional algebra.

Definition 2.33 (Identity mapping). A mapping $\Pi : \mathcal{D} \to \mathcal{D}$ is called *identity mapping* and denoted $\mathrm{Id}_{\mathcal{D}}$, if it assigns to each element $a \in \mathcal{D}$ the element a, i.e., $\Pi(a) = a,\ \forall a \in \mathcal{D}$.

Definition 2.34 (Closure mapping). A mapping from an idempotent semiring into the same idempotent semiring, i.e., $\Pi : \mathcal{D} \to \mathcal{D}$ is called a *closure mapping*, if it is

- extensive, i.e., $\Pi \geq \mathrm{Id}_{\mathcal{D}}$, i.e., $\Pi(a) \geq a \quad \forall a \in \mathcal{D}$
- idempotent, i.e., $\Pi \circ \Pi = \Pi$, i.e., $\Pi(\Pi(a)) = \Pi(a) \quad \forall a \in \mathcal{D}$
- isotone, i.e., $a \preceq b \Rightarrow \Pi(a) \preceq \Pi(b),\ \forall a, b \in \mathcal{D}$.

2.2.3 Residuation theory

The product law $\otimes$ in idempotent semirings does not necessarily admit an inverse. However, the so-called *residuation theory* [3, 4] provides a *pseudo inversion* of mappings defined over ordered sets. Since dioids are defined on (partially) ordered sets, it is possible to use residuation theory to determine the greatest solution (with respect to the natural order of the dioid) of inequality $\Pi(a) \preceq b$, if it exists.

Definition 2.35 (Residuated mapping). An isotone mapping $\Pi : \mathcal{D} \to \mathcal{C}$ with $(\mathcal{D}, \preceq)$ and $(C, \preceq)$ being ordered sets, is said to be *residuated*, if the inequality $\Pi(a) \preceq b$ has a greatest solution in $\mathcal{D}$ for all $b \in \mathcal{C}$.

Theorem 2.36 ([4]). For an isotone mapping $\Pi : \mathcal{D} \to \mathcal{C}$ from one ordered set to another ordered set, the following statements are equivalent:

(i) Π is residuated.

(ii) There exists an isotone mapping $\Pi^\sharp : \mathcal{C} \to \mathcal{D}$ such that

$$\Pi \circ \Pi^\sharp \preceq \mathrm{Id}_{\mathcal{C}}$$
$$\Pi^\sharp \circ \Pi \succeq \mathrm{Id}_{\mathcal{D}}.$$

Mapping $\Pi^\sharp$ is said to be the residual of Π.

Theorem 2.37 ([2]). For a residuated mapping $\Pi : \mathcal{D} \to \mathcal{C}$ the following equalities hold:

$$\Pi \circ \Pi^\sharp \circ \Pi = \Pi$$
$$\Pi^\sharp \circ \Pi \circ \Pi^\sharp = \Pi^\sharp$$

An illustration of these properties is given in Fig. 2.1.

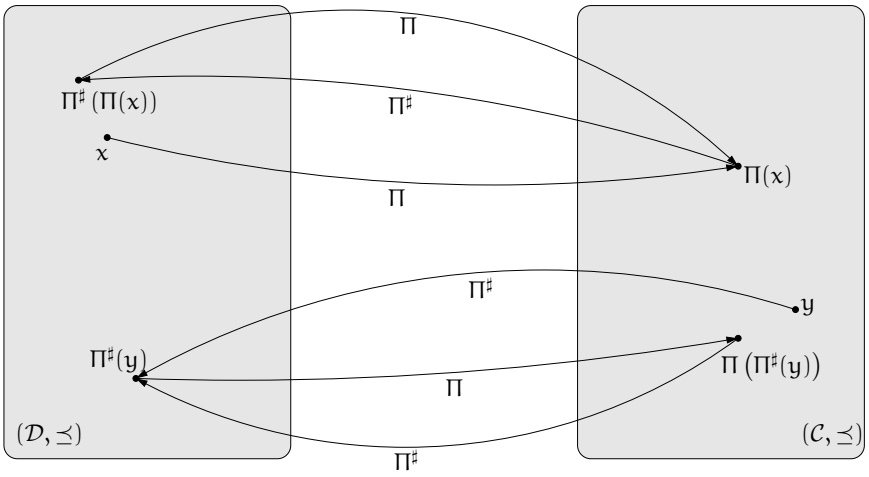


Figure 2.1: Properties of residuated mapping $\Pi : \mathcal{D} \to \mathcal{C}$ and the corresponding mapping $\Pi^\sharp : \mathcal{C} \to \mathcal{D}$.

Definition 2.38 (Canonical injection). An isotone mapping $\Pi_{\mathcal{S}} : \mathcal{S} \to \mathcal{D}$, with $\mathcal{S}$ being a sub-semiring of $\mathcal{D}$ and both $\mathcal{S}$ and $\mathcal{D}$ being complete dioids, is defined by:

$$\Pi_{\mathcal{S}}(a) = a \qquad \forall a \in \mathcal{S}.$$

Theorem 2.39 (Projection [24]). According to the definition of a residuated mapping (Def. 2.35), the canonical injection $\Pi_{\mathcal{S}}$ is a residuated mapping. The residual $\Pi_{\mathcal{S}}^{\sharp}$ is a *projector* from the dioid $\mathcal{D}$ to its sub-dioid $\mathcal{S}$ and denoted $\mathrm{Pr}_{\mathcal{S}}$. The following statements hold for $\mathrm{Pr}_{\mathcal{S}}$:

(i) $\mathrm{Pr}_{\mathcal{S}} \circ \mathrm{Pr}_{\mathcal{S}} = \mathrm{Pr}_{\mathcal{S}}$

(ii) $\mathrm{Pr}_{\mathcal{S}} \preceq \mathrm{Id}_{\mathcal{D}}$

(iii) $a \in \mathcal{S} \Leftrightarrow \mathrm{Pr}_{\mathcal{S}}(a) = a$.

Definition 2.40 (Projection on the image of a mapping [14]). Given a residuated mapping $f : \mathcal{D} \to \mathcal{C}$, the mapping $P_f = f \circ f^{\sharp}$ is a projector, i.e., $P_f \circ P_f = P_f$, and $P_f(c)$ with $c \in \mathcal{C}$ is the greatest element in $\mathrm{Im} f$ less than or equal to c.

It can be shown that two very elementary mappings in a complete dioid $(\mathcal{D}, \otimes, \oplus)$, namely, the left and right product with a constant, i.e.,

$$\begin{aligned} L_a &: \mathcal{D} \to \mathcal{D} \\ &\quad x \mapsto a \otimes x \quad \text{(left product with } a\text{)} \\ R_a &: \mathcal{D} \to \mathcal{D} \\ &\quad x \mapsto x \otimes a \quad \text{(right product with } a\text{)} \end{aligned}$$

are residuated mappings. The corresponding residual mappings are denoted:

$$\begin{aligned} L_a^{\sharp}(x) &= a \backslash\!\!\diamond x \quad \text{(left division by } a\text{)}, \\ R_a^{\sharp}(x) &= x \diamond\!\!/ a \quad \text{(right division by } a\text{)}. \end{aligned}$$

Consequently, $a \otimes x \preceq b$ has a greatest solution denoted $L_a^{\sharp}(b) = a \backslash\!\!\diamond b = \bigoplus \{x | ax \preceq b\}$. Analogously, the greatest solution of inequality $x \otimes a \preceq b$ is $R_a^{\sharp}(b) = b \diamond\!\!/ a = \bigoplus \{x | xa \preceq b\}$. These solutions are called left and right residuals, respectively. In case the dividend or the divisor are ε or $\top$ the following conventions apply for all $a \in \mathcal{D}$:

$$\begin{aligned} \varepsilon \backslash\!\!\diamond a &= a \diamond\!\!/ \varepsilon = \top \\ \top \backslash\!\!\diamond a &= a \diamond\!\!/ \top = \begin{cases} \top & \text{if } a = \top \\ \varepsilon & \text{else.} \end{cases} \end{aligned}$$

Remark 2.41. Depending on the literature different notations may be used to refer to the residual mappings of the left and right product. In [2], for example, the following notation is used

$$\begin{aligned} L_a^{\sharp} &: x \mapsto \frac{x}{a} = a \backslash\!\!\diamond x \\ R_a^{\sharp} &: x \mapsto \frac{x}{a} = x \diamond\!\!/ a. \end{aligned}$$

Of course, residuation theory is not restricted to mappings in ordered set of scalars, but can be extended to the matrix case. Given the matrices $A \in \mathcal{D}^{m\times n}$, $B \in \mathcal{D}^{m\times p}$, and $C \in \mathcal{D}^{n\times p}$, the greatest solution of inequality $A \otimes X \preceq B$ is given by $D = A \backslash\!\!\diamond B$ and inequality $X \otimes C \preceq B$ admits $E = B \diamond\!\!/ C$ as its greatest solution. The entries of D and E are determined as follows:

$$[D]_{ij} = \bigwedge_{k=1}^{m} ([A]_{ki} \backslash\!\!\diamond [B]_{kj})$$
$$[E]_{ij} = \bigwedge_{k=1}^{p} ([B]_{ik} \diamond\!\!/ [C]_{jk}) \,.$$

Example 2.42 (Max-plus algebra). Considering the relation $A \otimes X \preceq B$ with

$$A = \begin{pmatrix} 1 & 5 \\ \varepsilon & 4 \\ 2 & 7 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 7 \\ 5 \\ 9 \end{pmatrix}$$

being matrices with entries in $\overline{\mathbb{Z}}_{\max}$. As introduced above the product in $(\max, +)$-algebra corresponds to the classical addition. Consequently the residuation in $\overline{\mathbb{Z}}_{\max}$ corresponds to the classical subtraction, i.e., $2 \otimes x \preceq 6$ admits the solution set $\mathcal{X} = \{x | x \preceq 2 \backslash\!\!\diamond 6\}$ with $2 \backslash\!\!\diamond 6 = 6 - 2 = 4$ being the greatest solution of this set. Applying the rules of residuation in $(\max, +)$-algebra to $A \otimes X \preceq B$ results in:

$$A \backslash\!\!\diamond B = \begin{pmatrix} 1 \backslash\!\!\diamond 7 \wedge \varepsilon \backslash\!\!\diamond 5 \wedge 2 \backslash\!\!\diamond 9 \\ 5 \backslash\!\!\diamond 7 \wedge 4 \backslash\!\!\diamond 5 \wedge 7 \backslash\!\!\diamond 9 \end{pmatrix} = \begin{pmatrix} 6 \\ 1 \end{pmatrix}.$$

Matrix $A \backslash\!\!\diamond B = (6 \;\; 1)^T$ is the greatest solution for X which ensures $A \otimes X \preceq B$, i.e.,

$$A \otimes (A \backslash\!\!\diamond B) = \begin{pmatrix} 1 & 5 \\ \varepsilon & 4 \\ 2 & 7 \end{pmatrix} \otimes \begin{pmatrix} 6 \\ 1 \end{pmatrix} = \begin{pmatrix} 7 \\ 5 \\ 8 \end{pmatrix} \preceq \begin{pmatrix} 7 \\ 5 \\ 9 \end{pmatrix} = B.$$

Remark 2.43. Note that in max-plus algebra, residuation theory achieves equality in the scalar case, while this is in general not true for the matrix case. In general, one can say, that the equation $A \otimes X = B$ admits a solution if matrix $B \in \mathcal{D}^{m\times p}$ is in the image of $A \in \mathcal{D}^{m\times n}$, i.e., there exists a matrix $L \in \mathcal{D}^{n\times p}$ such that $B = A \otimes L$.

Below, some properties of the left and right "division" are given. To get a more exhaustive

list of properties and the corresponding proofs the reader is invited to consult [2, 20, 24].

$$a\,(a\backslash x) \preceq x \qquad (x/a)\,a \preceq x \tag{2.1}$$
$$a\backslash(ax) \succeq x \qquad (xa)/a \succeq x \tag{2.2}$$
$$a\,(a\backslash(ax)) = ax \qquad ((xa)/a)\,a = xa \tag{2.3}$$
$$a\backslash(a\,(a\backslash x)) = a\backslash x \qquad ((x/a)\,a)/a = x/a \tag{2.4}$$
$$a\backslash(x \wedge y) = a\backslash x \wedge a\backslash y \qquad (x \wedge y)/a = x/a \wedge y/a \tag{2.5}$$
$$a\backslash(x \oplus y) \succeq (a\backslash x) \oplus (a\backslash y) \qquad (x \oplus y)/a \succeq (x/a) \oplus (y/a) \tag{2.6}$$
$$(a \wedge b)\backslash x \succeq (a\backslash x) \oplus (b\backslash x) \qquad x/(a \wedge b) \succeq (x/a) \oplus (x/b) \tag{2.7}$$
$$(a \oplus b)\backslash x = a\backslash x \wedge b\backslash x \qquad x/(a \oplus b) = x/a \wedge x/b \tag{2.8}$$
$$(ab)\backslash x = b\backslash(a\backslash x) \qquad x/(ba) = (x/a)/b \tag{2.9}$$
$$(a\backslash x)\,b \preceq a\backslash(xb) \qquad b\,(x/a) \preceq (bx)/a \tag{2.10}$$
$$b\,(a\backslash x) \preceq (a/b)\backslash x \qquad (x/a)\,b \preceq x/(b\backslash a) \tag{2.11}$$

2.2.4 Fixed point equations

Theorem 2.44 (Fixpoint theorem). Every order preserving mapping of a complete lattice into itself has a fixpoint [59]. Furthermore, the set of fixpoints of an order preserving mapping of a complete lattice $(\mathcal{C}, \preceq)$ into itself, forms a complete lattice with respect to the ordering of $(\mathcal{C}, \preceq)$.

This theorem is essentially due to Knaster [35] and Tarski [60]. Formally, for an isotone mapping $\Pi : \mathcal{C} \to \mathcal{C}$ with $(\mathcal{C}, \vee, \wedge)$ being a complete lattice and $\mathcal{Y} = \{x \in \mathcal{C} | \Pi(x) = x\}$ being the set of fixed points of Π, one can write [36]

1. $\bigwedge\limits_{y \in \mathcal{Y}} y \in \mathcal{Y}$, and $\bigwedge\limits_{y \in \mathcal{Y}} y = \bigwedge \{x \in \mathcal{C} | \Pi(x) \preceq x\}$.
2. $\bigvee\limits_{y \in \mathcal{Y}} y \in \mathcal{Y}$, and $\bigvee\limits_{y \in \mathcal{Y}} y = \bigvee \{x \in \mathcal{C} | x \preceq \Pi(x)\}$.

Theorem 2.45 (Smallest fixed point). Let $(\mathcal{D}, \oplus, \otimes)$ be a complete dioid, $\Pi : \mathcal{D} \to \mathcal{D}$ be a lower semi-continuous mapping in this dioid, and $\mathcal{Y} = \{x \in \mathcal{D} | \Pi(x) = x\}$ the set of fixed points of Π. The smallest fixed point of Π is [2]

$$\bigwedge_{y \in \mathcal{Y}} y = \Pi^* \left(\bigwedge_{x \in \mathcal{D}} x \right) = \Pi^*(\varepsilon)$$

$$\text{with} \quad \Pi^* = \bigoplus_{i=0}^{+\infty} \Pi^i, \quad \Pi^{i+1} = \Pi \circ \Pi^i, \text{ and } \quad \Pi^0 = \mathrm{Id}_{\mathcal{D}}.$$

Theorem 2.46 (Greatest fixed point). Let $(\mathcal{D}, \oplus, \otimes)$ be a complete dioid, $\Pi : \mathcal{D} \to \mathcal{D}$ be an upper semi-continuous mapping in this dioid, and $\mathcal{Y} = \{x \in \mathcal{D} | \Pi(x) = x\}$ the set of fixed points of Π. The greatest fixed point of Π is [2]

$$\bigoplus_{y \in \mathcal{Y}} y = \Pi_* \left(\bigoplus_{x \in \mathcal{D}} x \right) = \Pi_*(\top)$$

$$\text{with} \quad \Pi_* = \bigwedge_{i=0}^{+\infty} \Pi^i, \quad \Pi^{i+1} = \Pi \circ \Pi^i, \text{ and } \quad \Pi^0 = \mathrm{Id}_{\mathcal{D}}.$$

Definition 2.47 (Kleene star). The Kleene star is a mapping denoted *. In a complete dioid $(\mathcal{D}, \oplus, \otimes)$ it is defined $\forall a \in \mathcal{D}$ by:

$$a^* = \bigoplus_{i=0}^{\infty} a^i \quad \text{with} \quad a^{i+1} = a \otimes a^i \quad \text{and} \quad a^0 = e.$$

Remark 2.48. Of course, the Kleene star can also be applied to (square) matrices in the corresponding complete dioid. In [2] the algorithm for a matrix with four blocks, i.e., $A \in \mathcal{D}^{n \times n}$ with

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix},$$

where a_{11} and a_{22} are square matrices of dimension n_1 and n_2, respectively, with $n_1 + n_2 = n$, is given by

$$A^* = \begin{pmatrix} a_{11}^* \oplus a_{11}^* a_{12} (a_{21} a_{11}^* a_{12} \oplus a_{22})^* a_{21} a_{11}^* & a_{11}^* a_{12} (a_{21} a_{11}^* a_{12} \oplus a_{22})^* \\ (a_{21} a_{11}^* a_{12} \oplus a_{22})^* a_{21} a_{11}^* & (a_{21} a_{11}^* a_{12} \oplus a_{22})^* \end{pmatrix}.$$

Example 2.49 (Solution of $x = ax \oplus b$). According to theorem 2.45, the least fixed point of the lower semi-continuous mapping $\Pi : x \mapsto ax \oplus b$ in a complete dioid $(\mathcal{D}, \oplus, \otimes)$ is $x = a^* b$. This can be shown by computing Π^*, i.e.,

$$\begin{aligned}
\Pi^0(x) &= x \\
\Pi^1(x) &= ax \oplus b \\
\Pi^2(x) &= a(ax \oplus b) \oplus b = a^2 x \oplus ab \oplus b \\
\Pi^3(x) &= a^2(ax \oplus b) \oplus ab \oplus b = a^3 x \oplus a^2 b \oplus ab \oplus b \\
&\vdots \\
\Pi^k(x) &= a^{k-1}(ax \oplus b) \oplus a^{k-2} b \oplus \ldots \oplus ab \oplus b
\end{aligned}$$

and according to the definition of Π^*:

$$\begin{aligned}\Pi^*(x) &= \bigoplus_{i=0}^{+\infty} \Pi^i(x) \\ &= x \oplus ax \oplus a^2x \oplus a^3x \oplus \ldots \oplus b \oplus ab \oplus a^2b \oplus a^3b \oplus \ldots \\ &= \left(e \oplus a \oplus a^2 \oplus a^3 \oplus \ldots\right) x \oplus \left(e \oplus a \oplus a^2 \oplus a^3 \oplus \ldots\right) b \\ &= a^*x \oplus a^*b.\end{aligned}$$

Finally, the least fixed point of $\Pi : x \mapsto ax \oplus b$ is equal to $\Pi^*(\varepsilon)$, i.e.,

$$\Pi^*(\varepsilon) = a^*\varepsilon \oplus a^*b = a^*b.$$

This is also the least solution of inequality $x \geq ax \oplus b$.

Remark 2.50. According to Def. 2.47 the mapping $L_{a^*} : x \mapsto a^*x$ in a complete dioid $(\mathcal{D}, \oplus, \otimes)$ is a closure mapping, i.e., the following equivalence holds

$$x = a^*x \Leftrightarrow x \in \mathrm{Im}L_{a^*}.$$

The Kleene star in a complete dioid $(\mathcal{D}, \oplus, \otimes)$ has the following properties $\forall a, b \in \mathcal{D}$:

$$\begin{aligned}(a^*)^* &= a^* \\ a(ba)^* &= (ab)^* a \\ (a \oplus b)^* &= (a^*b)^* a^* = b^* (ab^*)^* = (a \oplus b)^* a^* = b^* (a \oplus b)^* \\ a^*a^* &= a^* \\ (ab^*)^* &= e \oplus a(a \oplus b)^*.\end{aligned}$$

For the proofs and a more extensive list of properties of the Kleene star, the reader is invited to consult [20, 24]. Additionally, some nice properties of the Kleene star in combination with the residual mappings of the left and right product can be derived [20, 24]:

$$a = a^* \Leftrightarrow a = a \backslash a \qquad a = a^* \Leftrightarrow a = a / a \tag{2.12}$$

$$a^* \backslash x = a^* \backslash (a^* \backslash x) \qquad x / a^* = (x / a^*) / a^* \tag{2.13}$$

$$a^* x = a^* \backslash (a^* x) \qquad x a^* = (x a^*) / a^* \tag{2.14}$$

$$a^* \backslash x = a^* (a^* \backslash x) \qquad x / a^* = (x / a^*) a^*. \tag{2.15}$$

Moreover,

$$a \backslash a = (a \backslash a)^* \qquad a / a = (a / a)^*.$$

Furthermore, in [13] and [19] it has been shown that this property also holds for matrices $A \in \mathcal{D}^{p \times n}$ and $A \backslash A \in \mathcal{D}^{n \times n}$, i.e.,

$$A \backslash A = (A \backslash A)^*. \tag{2.16}$$

Theorem 2.51 (Greatest solution of $x^* \preceq a^*$ [39]). Let $(\mathcal{D}, \oplus, \otimes)$ be a complete dioid and $a, x \in \mathcal{D}$. The greatest solution of $x^* \preceq a^*$ is $x = a^*$.

Proof. By considering the definition of the Kleene star the following equivalence has to hold

$$x^* = e \oplus x \oplus x^2 \oplus \ldots \preceq a^* \Leftrightarrow \begin{cases} e \preceq a^* \\ x \preceq a^* \\ x^2 \preceq a^* \\ \vdots \end{cases}$$

The first inequality of the right hand side holds due to the definition of a^*. The second inequality obviously holds as well for $x = a^*$. According to the properties of the Kleene star in complete dioids, $(a^*)^2 = a^* a^* = a^*$ which indicates that a^* is also the solution for the following inequalities. Consequently, $x = a^*$ satisfies all inequalities on the right hand side and since $x \preceq a^*$, $x = a^*$ is indeed the greatest solution of $x^* \preceq a^*$. □

Lemma 2.52 ([2]). Given a matrix $A \in \mathcal{D}^{n \times n}$ and a matrix $x \in \mathcal{D}^{n \times p}$, the following equivalences hold

$$x \preceq A \backslash x \Leftrightarrow x \succeq Ax \Leftrightarrow x = A^* x \Leftrightarrow x = A^* \backslash x,$$

where the order relations $\preceq$ and $\succeq$ are applied element wise.

Lemma 2.53 ([53]). For two matrices $A, B \in \mathcal{D}^{n \times n}$ with $(\mathcal{D}^{n \times n}, \oplus, \otimes)$ being a complete dioid, the following statements are equivalent

$$A^* \succeq B^* \Leftrightarrow A^* B^* = B^* A^* = B^* \backslash A^* = A^* / B^* = A^*.$$

Remark 2.54. Given two closure mappings $L_{A^*} : x \mapsto A^* x$ and $L_{B^*} : x \mapsto B^* x$, such that $L_{A^*} \succeq L_{B^*}$, i.e., $A^* \succeq B^*$, the following equivalence holds:

$$L_{A^*} \succeq L_{B^*} \Leftrightarrow L_{A^*} \circ L_{B^*} = L_{B^*} \circ L_{A^*} = L_{A^*} \Leftrightarrow \mathrm{Im} L_{A^*} \subseteq \mathrm{Im} L_{B^*}.$$

2.2.5 Dual residuation

In Sec. 2.2.3 it has been shown how residuation theory can be applied to determine the greatest solution of inequalities like $\Pi(a) \preceq b$. However, it is of course also possible to determine the least solution of inequalities such as $\Pi(a) \succeq b$. The definitions for the so-called dual residuation are analogous to the definitions for the previously introduced residuation.

Definition 2.55 (Dually residuated mapping). An isotone mapping $\Pi : \mathcal{D} \to C$ with $(\mathcal{D}, \preceq)$ and $(\mathcal{C}, \preceq)$ being ordered sets, is said to be *dually residuated*, if inequality $\Pi(a) \succeq b$ has a least solution in $\mathcal{D}$ for all $b \in \mathcal{C}$.

Theorem 2.56 ([2]). For an isotone mapping $\Pi : \mathcal{D} \to \mathcal{C}$ from one complete dioid to another complete dioid, the following statements are equivalent:

(i) Π is dually residuated

(ii) $\Pi(\top) = \top$ and Π is upper semi-continuous

(iii) There exists an isotone lower semi-continuous mapping $\Pi^\flat : \mathcal{C} \to \mathcal{D}$, such that:

$$\Pi \circ \Pi^\flat \geq \mathrm{Id}_{\mathcal{C}}$$
$$\Pi^\flat \circ \Pi \preceq \mathrm{Id}_{\mathcal{D}}.$$

Mapping $\Pi^\flat$ is said to be the dual residual of Π.

Theorem 2.57 ([53]). For a dually residuated mapping $\Pi : \mathcal{D} \to \mathcal{C}$ the following equalities hold:

$$\Pi \circ \Pi^\flat \circ \Pi = \Pi$$
$$\Pi^\flat \circ \Pi \circ \Pi^\flat = \Pi^\flat.$$

Dual multiplication

In this section we are interested in determining a "pseudo inverse" of a particular operation denoted $\odot$, which is the so-called *dual multiplication*. This operation is not included in the standard definition of idempotent semirings.

Definition 2.58 (Dual multiplication). For two matrices $A \in \mathcal{D}^{p\times n}$ and $B \in \mathcal{D}^{n\times q}$ in a complete dioid the dual multiplication $A \odot B$ is defined by

$$[A \odot B]_{ij} = \bigwedge_{k=1}^{n} \Big([A]_{ik} \odot [B]_{kj} \Big) \qquad \forall i = 1, \ldots, p;\ \forall j = 1, \ldots, q$$

with the following convention in the scalar case:

$$a \odot b = a \otimes b \qquad \forall a, b \in \mathcal{D}\backslash\top$$
$$x \odot \top = \top \odot x = \top \quad \forall x \in \mathcal{D}.$$

Furthermore this dual product is assumed to distribute with respect to $\wedge$ of infinitely many elements.

In particular, this implies that $\varepsilon \odot \top = \top$, while $\varepsilon \otimes \top = \varepsilon$.

Definition 2.59 (Dual Kleene star). The dual Kleene star is a mapping denoted $_*$. In a complete dioid $(\mathcal{D}, \oplus, \otimes)$ it is defined for $A \in \mathcal{D}^{n\times n}$ as:

$$A_* = \bigwedge_{k=0}^{\infty} A^{\odot k},$$

where $A^{\odot 0} = I^{\odot}$ and $A^{\odot k} = A \odot A^{\odot(k-1)}$, with $I^{\odot}$ being the identity of the dual multiplication, i.e.,

$$\left[I^{\odot}\right]_{ij} = \begin{cases} e & \text{if } i = j \\ \top & \text{else.} \end{cases}$$

Definition 2.60 (Dual closure mapping). In a complete dioid $(\mathcal{D}, \oplus, \otimes)$ an isotone mapping $h : \mathcal{D} \to \mathcal{D}$ is said to be a dual closure mapping, if $h \circ h = h \preceq Id_{\mathcal{D}}$.

Remark 2.61. According to the previous definitions the mapping $\Lambda_{A_*} : x \mapsto A_* \odot x$ in a complete dioid is a dual closure mapping, i.e., the following equivalence holds

$$x = A_* \odot x \Leftrightarrow x \in Im\Lambda_{A_*}.$$

Definition 2.62. An element $a \in \mathcal{D}$ with $(\mathcal{D}, \oplus, \otimes)$ being a complete dioid, admits a left inverse (respectively a right inverse), if there exists an element b (respectively c), such that $b \otimes a = e$ ($a \otimes c = e$, respectively).

Lemma 2.63 ([2]). Given a scalar $a \in \mathcal{D}$, with $(\mathcal{D}, \oplus, \otimes)$ being a complete dioid, admitting a left inverse b and a right inverse c, the following statements hold:

- $b = c$ and both are denoted a^{-1}
- $\forall x, y \in \mathcal{D}$, $a(x \wedge y) = ax \wedge ay$.

Lemma 2.64 ([53]). Given a matrix $A \in \mathcal{D}^{p \times n}$ and the set $\mathcal{X}$ of elements in $\mathcal{D}^{n \times q}$. If every entry of A admits an inverse, the mapping $\Gamma_A : x \mapsto A \odot x$ is upper semi-continuous, i.e.,

$$\Gamma_A\left(\bigwedge_{x \in \mathcal{X}} x\right) = \bigwedge_{x \in \mathcal{X}} \Gamma_A(x).$$

Of particular interest is the dual left product, i.e., given the scalars $a, x \in \mathcal{D}$ and a admits an inverse denoted a^{-1}, then mapping $\Gamma_a : x \mapsto a \odot x$ is dually residuated and the dual residual is denoted:

$$\Gamma_a^{\flat} : x \mapsto a \backslash\!\!\bullet x$$

with

$$a \backslash\!\!\bullet x = a^{-1} \otimes x$$

under the conventions that

$$\top \backslash\!\!\bullet x = \varepsilon,\ \varepsilon \backslash\!\!\bullet x = \top, \text{ and } \varepsilon \backslash\!\!\bullet \varepsilon = \varepsilon.$$

This can be easily extended to the matrix case, i.e., given two matrices $A \in \mathcal{D}^{n\times n}$ and $X \in \mathcal{D}^{n\times n}$ and every entry in A admits an inverse, then mapping $\Gamma_A : X \mapsto A \odot X$ is dually residuated and the dual residual is denoted:

$$\Gamma_A^\flat : X \mapsto A \backslash\!\bullet X$$

with

$$[A \backslash\!\bullet X]_{ij} = \bigoplus_{k=1}^{n} \left([A]_{ki} \backslash\!\bullet [X]_{kj} \right) = \bigoplus_{k=1}^{n} \left([A]_{ki}^{-1} \otimes [X]_{kj} \right)$$

under the conventions that

$$\top \backslash\!\bullet x = \varepsilon,\ \varepsilon \backslash\!\bullet x = \top, \text{ and } \varepsilon \backslash\!\bullet \varepsilon = \varepsilon.$$

Remark 2.65. An important thing to realize is that

$$a \geq b \Rightarrow a \backslash\!\bullet x \preceq b \backslash\!\bullet x \quad \forall, a, b, x \in \mathcal{D}.$$

Furthermore, given that b admits an inverse and due to the associativity of multiplication in dioids, the following statement is true

$$b \backslash\!\bullet (a \otimes c) = (b \backslash\!\bullet a) \otimes c.$$

This can easily be shown by rewriting the equation, i.e.,

$$b \backslash\!\bullet (a \otimes c) = b^{-1} \otimes (a \otimes c) = \left(b^{-1} \otimes a \right) \otimes c.$$

Remark 2.66. Of course residuation theory can also be applied to determine the dual residual of the mapping $\Lambda_a : x \mapsto x \odot a$, with $a, x \in \mathcal{D}$. It is denoted

$$\Lambda_a^\flat : x \mapsto x \bullet\!\!/ a$$

with

$$x \bullet\!\!/ a = x \otimes a^{-1}$$

with the conventions that

$$x \bullet\!\!/ \top = \varepsilon,\ x \bullet\!\!/ \varepsilon = \top, \text{and } \varepsilon \bullet\!\!/ \varepsilon = \varepsilon.$$

In the matrix case, the dual residual of the mapping $\Lambda_A : X \mapsto X \odot A$, with $A, X \in \mathcal{D}^{n\times n}$, is denoted

$$\Lambda_A^\flat : X \mapsto X \bullet\!\!/ A$$

with

$$[X \not\bullet A]_{ij} = \bigoplus_{k=1}^{n} \left([X]_{ik} \not\bullet [A]_{jk} \right) = \bigoplus_{k=1}^{n} \left([X]_{ik} \otimes [A]_{jk}^{-1} \right)$$

with the conventions that

$$x \not\bullet \top = \varepsilon,\ x \not\bullet \varepsilon = \top, \text{and } \varepsilon \not\bullet \varepsilon = \varepsilon.$$

The dual left and right "division" have the following properties [53]:

$$a \backslash\!\bullet (x \oplus y) = a \backslash\!\bullet x \oplus a \backslash\!\bullet y \qquad (x \oplus y) \not\bullet a = x \not\bullet a \oplus y \not\bullet a \tag{2.17}$$

$$a \backslash\!\bullet (x \wedge y) \preceq a \backslash\!\bullet x \wedge a \backslash\!\bullet y \qquad (x \wedge y) \not\bullet a \preceq x \not\bullet a \wedge y \not\bullet a \tag{2.18}$$

$$(x \oplus y) \backslash\!\bullet a \preceq x \backslash\!\bullet a \wedge y \backslash\!\bullet a \qquad a \not\bullet (x \oplus y) \preceq a \not\bullet x \wedge a \not\bullet y \tag{2.19}$$

$$a \odot (a \backslash\!\bullet x) \succeq x \qquad (x \not\bullet a) \odot a \succeq x \tag{2.20}$$

$$a \backslash\!\bullet (a \odot x) \preceq x \qquad (x \odot a) \not\bullet a \preceq x \tag{2.21}$$

$$a \odot (a \backslash\!\bullet (a \odot x)) = a \odot x \qquad ((x \odot a) \not\bullet a) \odot a = x \odot a \tag{2.22}$$

$$a \backslash\!\bullet (a \odot (a \backslash\!\bullet x)) = a \backslash\!\bullet x \qquad ((x \not\bullet a) \odot a) \not\bullet a = x \not\bullet a \tag{2.23}$$

$$(a \odot b) \backslash\!\bullet x = b \backslash\!\bullet (a \backslash\!\bullet x) \qquad x \not\bullet (b \odot a) = (x \not\bullet a) \not\bullet b \tag{2.24}$$

$$(a \backslash\!\bullet x) \not\bullet b = a \backslash\!\bullet (x \not\bullet b) \qquad b \backslash\!\bullet (x \not\bullet a) = (b \backslash\!\bullet x) \not\bullet a \tag{2.25}$$

Lemma 2.67 ([7]). Similar to Lem. 2.52 it is possible to show that the following equivalences hold for two matrices $A \in \mathcal{D}^{n \times n}$ and $x \in \mathcal{D}^{n \times p}$

$$x \preceq A \odot x \Leftrightarrow x \succeq A \backslash\!\bullet x \Leftrightarrow x = A_* \backslash\!\bullet x \Leftrightarrow x = A_* \odot x,$$

where the order relations $\preceq$ and $\succeq$ are applied element wise.

Proof. (1) $\Rightarrow$ (2) According to Def. 2.55 mapping $\Lambda_A^\flat$ is order preserving, hence $x \preceq A \odot x \Rightarrow A \backslash\!\bullet x \preceq A \backslash\!\bullet (A \odot x)$, furthermore the same definition implies $A \backslash\!\bullet x \preceq A \backslash\!\bullet (A \odot x) \preceq x$. Hence $x \preceq A \odot x \Rightarrow A \backslash\!\bullet x \preceq x$.

(2) $\Rightarrow$ (3) According to Eq. 2.24 $A \backslash\!\bullet (A \backslash\!\bullet x) = A^{\odot 2} \backslash\!\bullet x$, furthermore mapping $\Lambda_A^\flat$ is order preserving, then

$$x \succeq A \backslash\!\bullet x \Rightarrow A \backslash\!\bullet x \succeq A \backslash\!\bullet (A \backslash\!\bullet x) = A^{\odot 2} \backslash\!\bullet x,$$

hence

$$x \succeq A \backslash\!\bullet x \succeq A^{\odot 2} \backslash\!\bullet x \succeq \ldots \Rightarrow x \succeq (I^{\odot} \backslash\!\bullet x) \oplus (A \backslash\!\bullet x) \oplus (A^{\odot 2} \backslash\!\bullet x) \oplus \ldots$$

Furthermore, according to [2] and to Def. 2.59,

$$(I^{\odot} \backslash\!\bullet x) \oplus (A \backslash\!\bullet x) \oplus (A^{\odot 2} \backslash\!\bullet x) \oplus \ldots = (I^{\odot} \wedge A \wedge A^{\odot 2} \wedge \ldots) \backslash\!\bullet x = A_* \backslash\!\bullet x,$$

then, $x \geq (A \backslash\!\bullet x) \Rightarrow x \geq A_* \backslash\!\bullet x$. On the other hand $A_* \preceq I^{\odot}$ then $A_* \backslash\!\bullet x \geq x$, hence $x \geq (A \backslash\!\bullet x) \Rightarrow x = A_* \backslash\!\bullet x$.

(3) $\Rightarrow$ (4) From Theorem 2.56 the following inequality holds: $A_* \odot (A_* \backslash\!\bullet x) \geq x$, hence,

$$x = A_* \backslash\!\bullet x \Rightarrow A_* \odot x = A_* \odot (A_* \backslash\!\bullet x) \geq x,$$

but the definition of a dual closure mapping (Def. 2.60) yields $A_* \odot x \preceq x$, hence

$$x = A_* \backslash\!\bullet x \Rightarrow A_* \odot x = x.$$

(4) $\Rightarrow$ (1) According to Def. 2.58 and Def. 2.59,

$$A_* \odot x = (I^{\odot} \wedge A \wedge A^{\odot 2} \wedge \ldots) \odot x = (x \wedge A \odot x \wedge A^{\odot 2} \odot x \wedge \ldots),$$

hence $x = A_* \odot x \Rightarrow x \preceq A \odot x$.

□

Lemma 2.68 ([7]). Given three matrices $A \in \mathcal{D}^{n \times p}$, $X \in \mathcal{D}^{p \times q}$, and $B \in \mathcal{D}^{n \times r}$. If every entry of B admits an inverse the following property holds

$$B \backslash\!\bullet (A \otimes X) = (B \backslash\!\bullet A) \otimes X. \tag{2.26}$$

Proof.

$$\begin{aligned}
[B \backslash\!\bullet (A \otimes X)]_{ij} &= \bigoplus_{l=1}^{n} [B]_{li} \backslash\!\bullet [A \otimes X]_{lj} \\
&= \bigoplus_{l=1}^{n} [B]_{li} \backslash\!\bullet \left(\bigoplus_{k=1}^{p} ([A]_{lk} \otimes [X]_{kj}) \right) \\
&= \bigoplus_{l=1}^{n} \bigoplus_{k=1}^{p} [B]_{li} \backslash\!\bullet ([A]_{lk} \otimes [X]_{kj}) \quad \left(\Gamma_B^b \text{ is lower semi-continuous} \right) \\
&= \bigoplus_{l=1}^{n} \bigoplus_{k=1}^{p} [B]_{li}^{-1} \otimes ([A]_{lk} \otimes [X]_{kj}) \quad ([B]_{li} \text{ admits an inverse}) \\
&= \bigoplus_{l=1}^{n} \bigoplus_{k=1}^{p} \left([B]_{li}^{-1} \otimes [A]_{lk} \right) \otimes [X]_{kj} \quad (\otimes \text{ is associative}) \\
&= \bigoplus_{l=1}^{n} \bigoplus_{k=1}^{p} ([B]_{li} \backslash\!\bullet [A]_{lk}) \otimes [X]_{kj} \\
&= \bigoplus_{k=1}^{p} [B \backslash\!\bullet A]_{ik} \otimes [X]_{kj} = [(B \backslash\!\bullet A) \otimes X]_{ij}.
\end{aligned}$$

□

2.2.6 Idempotent semirings of formal power series

Definition 2.69 (Formal power series). A formal power series in p (commutative) variables, denoted z_1 to z_p, with coefficients in a semiring $\mathcal{D}$, is a mapping s defined from $\mathbb{Z}^p$ into $\mathcal{D}$: $\forall k = (k_1, \ldots, k_p) \in \mathbb{Z}^p$, $s(k)$ represents the coefficient of $z_1^{k_1} \ldots z_p^{k_p}$ and $(k_1, \ldots, k_p)$ are the exponents. Another equivalent representation is

$$s(z_1, \ldots, z_p) = \bigoplus_{k \in \mathbb{Z}^p} s(k) z_1^{k_1} \ldots z_p^{k_p}$$

Definition 2.70 (Support, degree, and valuation of a formal power series). The support of a formal power series is defined as

$$\mathrm{supp}(s) = \{(k_1, \ldots, k_p) \in \mathbb{Z}^p | s(k_1, \ldots, k_p) \neq \varepsilon\}.$$

The *degree* $\deg(s)$ (respectively *valuation* $\mathrm{val}(s)$) is the least upper bound (respectively greatest lower bound) of $\mathrm{supp}(s)$ in the complete lattice $(\overline{\mathbb{Z}}, \vee, \wedge)$, where $\overline{\mathbb{Z}} = \mathbb{Z} \cup \{-\infty, +\infty\}$. A series with a finite support is called a polynomial and a monomial if there is only one element in the support.

Definition 2.71 (Idempotent semiring of series). The set of formal power series with coefficients in an idempotent semiring $\mathcal{D}$ endowed with the following sum and Cauchy product

$$\begin{aligned} s \oplus s' : &\quad (s \oplus s')(k) = s(k) \oplus s'(k) \\ s \otimes s' : &\quad (s \otimes s')(k) = \bigoplus_{i+j=k} s(i) \otimes s'(j), \end{aligned}$$

is an idempotent semiring denoted $\mathcal{D}[\![z_1, \ldots, z_p]\!]$. If $\mathcal{D}$ is complete, $\mathcal{D}[\![z_1, \ldots, z_p]\!]$ is complete. The greatest lower bound of two series is given by

$$s \wedge s' : \quad (s \wedge s')(k) = s(k) \wedge s'(k).$$

Definition 2.72 (γ-transform). The γ-transform of a signal $\tilde{s}(k)$ is defined by

$$s(\gamma) = \bigoplus_{k \in \mathbb{Z}} \tilde{s}(k) \otimes \gamma^k.$$

Remark 2.73. The γ-transform is analogous to the z-transform in classical systems theory, which allows to describe a discrete signal by a formal power series.

Remark 2.74. Since $s'(\gamma) = s(\gamma) \otimes \gamma = \bigoplus_{k \in \mathbb{Z}} \tilde{s}(k) \otimes \gamma^{k+1} = \bigoplus_{k \in \mathbb{Z}} \tilde{s}'(k) \otimes \gamma^k$, it is clear that $\tilde{s}'(k) = \tilde{s}(k-1)$ and therefore, γ can be seen as a backward shift operator.

Definition 2.75 (Idempotent semiring $\overline{\mathbb{Z}}_{\max}[\![\gamma]\!]$). The set of formal power series in γ with exponents in $\mathbb{Z}$ and coefficients in $\overline{\mathbb{Z}}_{\max}$ is an idempotent semiring and is denoted $\overline{\mathbb{Z}}_{\max}[\![\gamma]\!]$.

The zero element is the series $\varepsilon(\gamma) = \bigoplus_{k\in\mathbb{Z}} \varepsilon\gamma^k$, where $\varepsilon = -\infty$, the zero element of $(\overline{\mathbb{Z}}_{\max}, \oplus, \otimes)$. The unit element is the formal series $e(\gamma) = e\gamma^0$, where $e = 0$ is the unit element of $(\overline{\mathbb{Z}}_{\max}, \oplus, \otimes)$. The sum and product in $\overline{\mathbb{Z}}_{\max}[\![\gamma]\!]$ are defined by

$$s_1(\gamma) \oplus s_2(\gamma) = \bigoplus_{k\in\mathbb{Z}} (\tilde{s}_1(k) \oplus \tilde{s}_2(k))\,\gamma^k$$
$$s_1(\gamma) \otimes s_2(\gamma) = \bigoplus_{k\in\mathbb{Z}} \bigoplus_{k_1+k_2=k} (\tilde{s}_1(k_1) \otimes \tilde{s}_2(k_2))\,\gamma^k.$$

Example 2.76. Given two formal power series in $\overline{\mathbb{Z}}_{\max}[\![\gamma]\!]$, e.g., the monomial $s_1(\gamma) = 3\gamma^2$ and the polynomial $s_2(\gamma) = 0\gamma^1 \oplus 2\gamma^2$, their sum and product are

$$s_1(\gamma) \oplus s_2(\gamma) = 3\gamma^2 \oplus 0\gamma^1 \oplus 2\gamma^2 = 0\gamma^1 \oplus (3 \oplus 2)\,\gamma^2 = 0\gamma^1 \oplus 3\gamma^2$$
$$s_1(\gamma) \otimes s_2(\gamma) = 3\gamma^2 \otimes \left(0\gamma^1 \oplus 2\gamma^2\right) = (3 \otimes 0)\,\gamma^3 \oplus (3 \otimes 2)\,\gamma^4 = 3\gamma^3 \oplus 5\gamma^4.$$

Remark 2.77. In general, we will only write the terms of a power series which have a non-zero coefficient, e.g., $s_1(\gamma) = \ldots \oplus \varepsilon\gamma^0 \oplus \varepsilon\gamma^1 \oplus 3\gamma^2 \oplus \varepsilon\gamma^3 \oplus \varepsilon\gamma^4 \oplus \ldots = 3\gamma^2$.

Definition 2.78 (δ-transform). Analogously to the γ-transform, the δ-transform of a signal $\hat{s}(t)$ is defined by

$$s(\delta) = \bigoplus_{t\in\mathbb{Z}} \hat{s}(t) \otimes \delta^t.$$

Definition 2.79 (Idempotent semiring $\overline{\mathbb{Z}}_{\min}[\![\delta]\!]$). The set of formal power series in δ with exponents in $\mathbb{Z}$ and coefficients in $\overline{\mathbb{Z}}_{\min}$ has a dioid structure and is denoted $\overline{\mathbb{Z}}_{\min}[\![\delta]\!]$. The zero and unit element are $\varepsilon(\delta) = \bigoplus_{t\in\mathbb{Z}} \varepsilon\delta^t$, with $\varepsilon = +\infty$, and $e(\delta) = e\delta^0$, with $e = 0$, respectively. Addition and multiplication in $\overline{\mathbb{Z}}_{\min}[\![\delta]\!]$ are defined by

$$s_1(\delta) \oplus s_2(\delta) = \bigoplus_{t\in\mathbb{Z}} (\hat{s}_1(t) \oplus \hat{s}_2(t))\,\delta^t$$
$$s_1(\delta) \otimes s_2(\delta) = \bigoplus_{t\in\mathbb{Z}} \bigoplus_{t_1+t_2=t} (\hat{s}_1(t_1) \otimes \hat{s}_2(t_2))\,\delta^t.$$

Example 2.80. Given two formal power series in $\overline{\mathbb{Z}}_{\min}[\![\delta]\!]$, e.g., the monomial $s_1(\delta) = 2\delta^3$ and the polynomial $s_2(\delta) = 1\delta^0 \oplus 2\delta^2$, their sum and product are

$$s_1(\delta) \oplus s_2(\delta) = 2\delta^3 \oplus 1\delta^0 \oplus 2\delta^2 = 1\delta^0 \oplus 2\delta^2 \oplus 2\delta^3$$
$$s_1(\delta) \otimes s_2(\delta) = 2\delta^3 \otimes \left(1\delta^0 \oplus 2\delta^2\right) = (2 \otimes 1)\,\delta^3 \oplus (2 \otimes 2)\,\delta^5 = 3\delta^3 \oplus 4\delta^5.$$

Definition 2.81 (Idempotent semiring $\mathbb{B}[\![\gamma, \delta]\!]$). The dioid of formal power series in two commutative variables γ and δ with Boolean coefficients, i.e., $\mathbb{B} = \{\varepsilon, e\}$, and exponents in $\mathbb{Z}$ is denoted $\mathbb{B}[\![\gamma, \delta]\!]$. A series $s \in \mathbb{B}[\![\gamma, \delta]\!]$ is represented by

$$s(\gamma, \delta) = \bigoplus_{k,t\in\mathbb{Z}} \bar{s}(k, t)\gamma^k\delta^t,$$

with $s(k,t) \in \mathbb{B}$. $\mathbb{B}[\![\gamma,\delta]\!]$ is a complete and commutative dioid. The zero and unit element are $\varepsilon(\gamma,\delta) = \bigoplus_{k,t\in\mathbb{Z}} \varepsilon\gamma^k\delta^t$ and $e(\gamma,\delta) = \gamma^0\delta^0$, respectively.

Example 2.82. One possible series in $\mathbb{B}[\![\gamma,\delta]\!]$ is $s(\gamma,\delta) = \gamma^1\delta^0 \oplus \gamma^2\delta^1 \oplus \gamma^3\delta^3 \oplus \gamma^4\delta^3$. Please remember that according to remark 2.77, we only write the elements of the power series whose coefficients are equal to e. Then, a series can graphically be represented in the $\mathbb{Z}^2$-plane, with the exponents of γ on the horizontal axis and the exponents of δ on the vertical axis, by drawing a black dot for all elements with non-zero coefficient. The corresponding graphical representation of the series $s_1(\gamma,\delta)$ is given in Fig. 2.2.

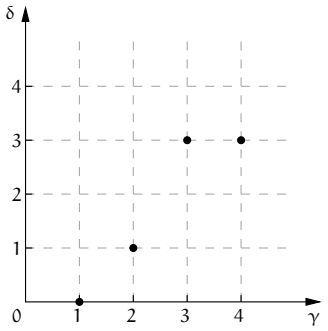


Figure 2.2: Graphical representation of the series $s(\gamma,\delta) = \gamma^1\delta^0 \oplus \gamma^2\delta^1 \oplus \gamma^3\delta^3 \oplus \gamma^4\delta^3 \in \mathbb{B}[\![\gamma,\delta]\!]$.

Quotient dioids

Definition 2.83 (Congruence). In a dioid $(\mathcal{D},\oplus,\otimes)$, a *congruence relation* is an equivalence relation denoted $\equiv$, which satisfies $\forall a,b,c \in \mathcal{D}$:

$$a \equiv b \Rightarrow \begin{cases} a \oplus c \equiv b \oplus c \\ a \otimes c \equiv b \otimes c. \end{cases}$$

Definition 2.84 (Equivalence class). Given a dioid $(\mathcal{D},\oplus,\otimes)$ equipped with an equivalence relation $\equiv$. The *equivalence class* represented by an element $a \in \mathcal{D}$ is denoted $[a]_\equiv$ and is defined as

$$[a]_\equiv = \{x \in \mathcal{D} | x \equiv a\}\,.$$

Thus, an equivalence class $[a]_\equiv$ is the set of all elements which are equivalent to a with respect to the equivalence relation $\equiv$.

Lemma 2.85 (Quotient dioid). The quotient of a dioid $(\mathcal{D},\oplus,\otimes)$ with respect to a congruence relation $\equiv$ is itself a dioid. It is called *quotient dioid* and is denoted $\mathcal{D}_{/\equiv}$. For addition and

multiplication the following properties hold [2]

$$[a]_{\equiv} \oplus [b]_{\equiv} = [a \oplus b]_{\equiv}$$
$$[a]_{\equiv} \otimes [b]_{\equiv} = [a \otimes b]_{\equiv}.$$

Definition 2.86 (Idempotent semiring $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$). The quotient dioid of $\mathbb{B}[\![\gamma,\delta]\!]$ with respect to the congruence relation in $\mathbb{B}[\![\gamma,\delta]\!]$:

$$a \equiv b \Leftrightarrow \gamma^* \left(\delta^{-1}\right)^* a = \gamma^* \left(\delta^{-1}\right)^* b,$$

is denoted $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$, i.e., $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!] = \mathbb{B}[\![\gamma,\delta]\!]_{/\gamma^*(\delta^{-1})^*}$, where * refers to the Kleene star. $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ constitutes a complete dioid and the zero and unit elements are $\varepsilon(\gamma,\delta) = \bigoplus_{k,t\in\mathbb{Z}} \varepsilon\gamma^k\delta^t$ and $e(\gamma,\delta) = \gamma^0\delta^0$, respectively.

Remark 2.87. In the following, the dioid $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ will be used to describe how often an event can occur within a specific time. For example, the monomial $\gamma^k\delta^t$ is to be interpreted as: "The $(k+1)$-st occurrence of the event is at time t at the earliest."

Graphically, a monomial $\gamma^k\delta^t \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ cannot be represented as a point in the $\mathbb{Z}^2$-plane (as it was the case for $\gamma^k\delta^t \in \mathbb{B}[\![\gamma,\delta]\!]$). This is due to the fact that in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ $\gamma^k\delta^t \equiv \gamma^k\delta^t \otimes \gamma^* \left(\delta^{-1}\right)^*$. Rewriting the right hand side of this equivalence results in

$$\begin{aligned}
\gamma^k\delta^t &\equiv \gamma^k\delta^t \underbrace{\left(\bigoplus_{i=0}^{\infty}\gamma^i\right)}_{\gamma^*} \underbrace{\left(\bigoplus_{j=0}^{\infty}\left(\delta^{-1}\right)^j\right)}_{(\delta^{-1})^*} \\
&= \gamma^k\delta^t \left(\bigoplus_{i=0}^{\infty}\gamma^i\right)\left(\bigoplus_{j=0}^{\infty}\delta^{-j}\right) \\
&= \bigoplus_{i=0}^{\infty}\bigoplus_{j=0}^{\infty}\gamma^{k+i}\delta^{t-j}.
\end{aligned}$$

Consequently, the monomial $\gamma^k\delta^t \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ represents the set $\{\gamma^n\delta^m \in \mathbb{B}[\![\gamma,\delta]\!] \,|\, n \geqslant k, m \leqslant t\}$. Graphically speaking, every monomial $\gamma^k\delta^t \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ represents all points in the $\mathbb{Z}^2$-plane that are "south-east" of the point (k,t). Consequently, a polynomial in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is graphically represented as the union of south-east cones of the single monomials composing the polynomial.

Example 2.88. A possible series in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is $s(\gamma,\delta) = \gamma^1\delta^0 \oplus \gamma^2\delta^3 \oplus \gamma^4\delta^4$. Its corresponding graphical representation is given in Fig. 2.3.

Remark 2.89. The graphical representation allows for a straightforward visualization of the partial order $\preceq$ in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$. Namely, $s_1 \preceq s_2$, if the graphical representation of s_1 is contained in the respective representation of s_2. Consequently, the zero element of $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$

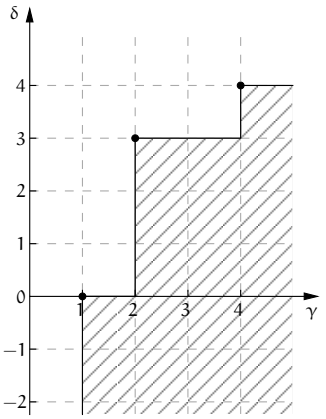


Figure 2.3: Graphical representation of the series $s(\gamma,\delta) = \gamma^1\delta^0 \oplus \gamma^2\delta^3 \oplus \gamma^4\delta^4 \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$.

needs to be the "bottom right" element and the top element the "top left" element. Therefore (and for simplicity reasons), these elements are often denoted $\varepsilon = \gamma^{+\infty}\delta^{-\infty}$ and $\top = \gamma^{-\infty}\delta^{+\infty}$, respectively.

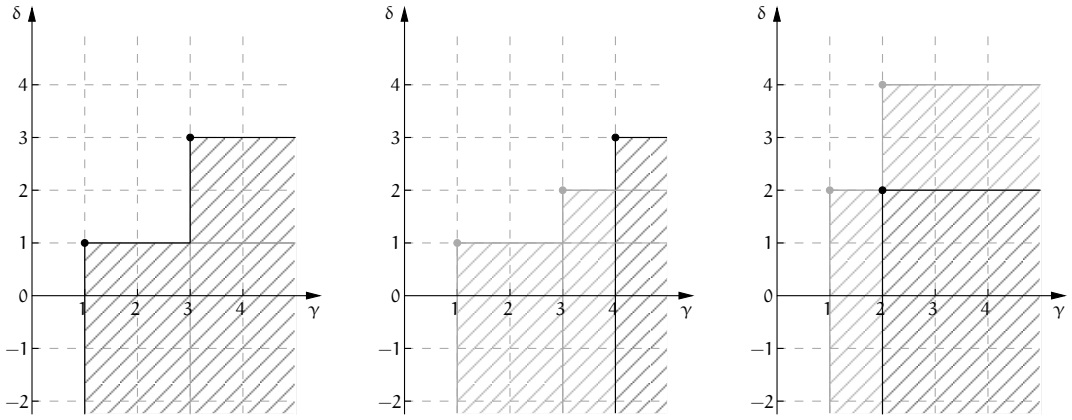
Remark 2.90 (Minimal representation). As mentioned before, two series s_1 and s_2 in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ belong to an equivalence class if $s_1 \otimes \gamma^* \left(\delta^{-1}\right)^* = s_2 \otimes \gamma^* \left(\delta^{-1}\right)^*$. Graphically speaking this means all series of an equivalence class "cover" the same area in the $\mathbb{Z}^2$-plane. For example, the series

$$\begin{aligned} s_1 &= \gamma^1\delta^0 \oplus \gamma^2\delta^3 \oplus \gamma^4\delta^4 \\ s_2 &= \gamma^1\delta^0 \oplus \gamma^2\delta^3 \oplus \gamma^3\delta^2 \oplus \gamma^4\delta^4 \\ s_3 &= \gamma^1\delta^0 \oplus \gamma^2\delta^3 \oplus \gamma^3\delta^2 \oplus \gamma^4\delta^4 \oplus \gamma^6\delta^4 \end{aligned}$$

are all equivalent with respect to the congruence relation $\gamma^* \left(\delta^{-1}\right)^*$. However, series s_1 is the so-called *minimal representation*, as its support consists of a minimal number of elements. In the following the minimal representation is (always) used to denote an equivalence class. Note that every equivalence class has one unique minimal representation.

For monomials in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ the following rules apply for addition, multiplication and the greatest lower bound:

$$\begin{aligned} \gamma^k\delta^t \oplus \gamma^l\delta^t &= \gamma^{\min(k,l)}\delta^t \\ \gamma^k\delta^t \oplus \gamma^k\delta^\tau &= \gamma^k\delta^{\max(t,\tau)} \\ \gamma^k\delta^t \otimes \gamma^l\delta^\tau &= \gamma^{(k+l)}\delta^{(t+\tau)} \\ \gamma^k\delta^t \wedge \gamma^l\delta^\tau &= \gamma^{\max(k,l)}\delta^{\min(t,\tau)} \end{aligned}$$



(a) $\gamma^1\delta^1 \oplus \gamma^3\delta^3$ (b) $\gamma^1\delta^1 \otimes \gamma^3\delta^2 = \gamma^4\delta^3$ (c) $\gamma^1\delta^2 \wedge \gamma^2\delta^4 = \gamma^2\delta^2$

Figure 2.4: Graphical representation of operations in $\mathcal{M}_{\text{in}}^{\text{ax}}[\![\gamma, \delta]\!]$.

Graphically, for monomials in $\mathcal{M}_{\text{in}}^{\text{ax}}[\![\gamma, \delta]\!]$

- addition: $\gamma^k\delta^t \oplus \gamma^l\delta^\tau$ refers to the union of south-east cones of (k, t) and (l, τ)
- multiplication: $\gamma^k\delta^t \otimes \gamma^l\delta^\tau$ refers to the south-east cone of $(k + l, t + \tau)$
- greatest lower bound: $\gamma^k\delta^t \wedge \gamma^l\delta^\tau$ refers to the intersection of the two south-east cones of (k, t) and (l, τ), i.e., the south-east cone of $(\max(k, l), \min(t, \tau))$.

The graphical representation of these operations is given in Fig. 2.4.

3
Model of Discrete-Event Systems

As mentioned before, there are many different modeling approaches for discrete-event systems. Among them are *finite state automata (FSA)*, *Markov chains*, and *queueing systems* (see [11] for an overview). In this thesis *Petri nets* are used for the modeling of discrete-event systems. This modeling approach was first developed in the early 1960s by Carl Adam Petri in his PhD thesis [56]. A standard Petri net consists of a *Petri net graph* (or *Petri net structure*) and its *initial marking*. The first part of this chapter is mainly based on [8, 9] and [30].

Definition 3.1 (Petri net graph [11]). A Petri net graph is a directed bipartite graph

$$N = (P, T, A, w)$$

where

$P = \{p_1, \ldots, p_n\}$ is the finite set of places
$T = \{t_1, \ldots, t_m\}$ is the finite set of transitions
$A \subseteq (P \times T) \cup (T \times P)$ is the set of directed arcs from places to transitions and from transitions to places
$w : (P \times T) \cup (T \times P) \to \mathbb{N}_0$ is a weight function.

In general, events are associated with transitions and conditions for events to occur are associated with places. In the sequel, the following notation is used for Petri net graphs:

$$I(t_j) = \{p_i \in P | (p_i, t_j) \in A\}$$

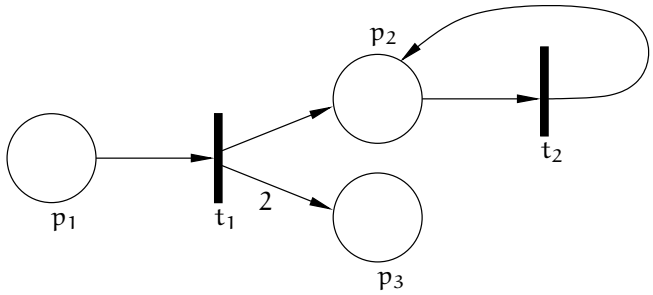


Figure 3.1: Simple Petri net graph for Example 3.2.

is the set of all input places for transition t_j, i.e., the set of places with arcs to t_j. Similarly,

$$O(t_j) = \{p_i \in P | (t_j, p_i) \in A\}$$

denotes the set of all output places for transition t_j, i.e., the set of all places with incoming arcs from t_j. Analogously, the sets of input and output transitions for place p_i are denoted

$$I(p_i) = \{t_j \in T | (t_j, p_i) \in A\}$$
$$O(p_i) = \{t_j \in T | (p_i, t_j) \in A\}.$$

Obviously, $p_i \in I(t_j)$ if and only if $t_j \in O(p_i)$, and $t_j \in I(p_i)$ if and only if $p_i \in O(t_j)$. Graphically, places are represented by circles, transitions by bars, arcs by arrows, and weights by numbers at the corresponding arrows. Usually, weights are only displayed explicitly if they are different from one. Furthermore, if there is no arc from place p_i to transition t_j, i.e., $p_i \notin I(t_j)$ and $t_j \notin O(p_i)$, the corresponding weight $w(p_i, t_j) = 0$. Similarly, if there is no arc from transition t_j to place p_i, i.e., $t_j \notin I(p_i)$ and $p_i \notin O(t_j)$, the weight $w(t_j, p_i) = 0$.

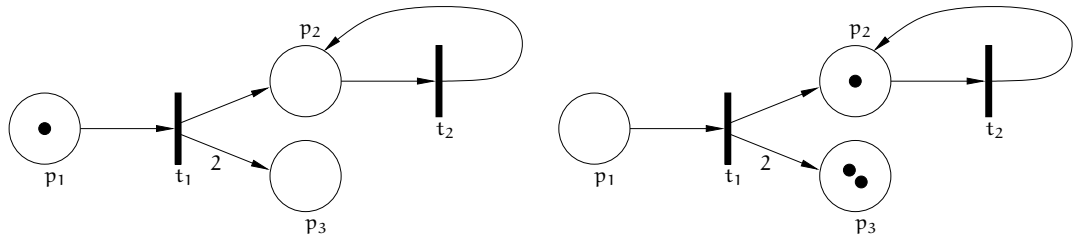
Example 3.2 (Petri net graph). A (rather simple) Petri net graph is defined by

$$P = \{p_1, p_2, p_3\}, \quad T = \{t_1, t_2\}, \quad A = \{(p_1, t_1), (t_1, p_2), (t_1, p_3), (p_2, t_2), (t_2, p_2)\}$$
$$w(p_1, t_1) = 1, \quad w(t_1, p_2) = 1, \quad w(t_1, p_3) = 2, \quad w(p_2, t_2) = 1, \quad w(t_2, p_2) = 1.$$

The corresponding graphical representation is given in Fig. 3.1. Please note that, as in this example, a transition t_j may be an input and an output transition of the same place p_i, e.g., $t_2 \in I(p_2)$ and $t_2 \in O(p_2)$.

Definition 3.3 (Petri net system). A Petri net system (or Petri net) is a pair (N, m^0), where $N = (P, T, A, w)$ is a Petri net graph and $m^0 \in \mathbb{N}_0^n$ with $n = |P|$ is a vector of initial markings.

In graphical representations the vector of initial markings is indicated by black dots, also called *tokens* in the corresponding places, i.e., place p_i contains m_i^0 tokens. A Petri net can



(a) Petri net at the initial state.

(b) Petri net after firing transition t_1.

Figure 3.2: Petri net before and after the firing of transition t_1 (Example 3.4).

then be interpreted as a dynamical system with a state signal $m : \mathbb{N}_0 \to \mathbb{N}_0^n$ and an initial state $m(0) = m^0$ and $m(k)$ is the state after the k-th firing of a transition. Its dynamics is governed by the so called *firing rules*:

(i) In state $m(k)$, a transition t_j can occur (or "fire") if and only if all of its input places contain at least as many tokens as the weight of the arc connecting the input place with the transition t_j, i.e.,

$$m_i(k) \geqslant w(p_i, t_j), \forall p_i \in I(t_j).$$

(ii) When a transition t_j fires, the number of tokens in all its input places is decreased by the weight of the connecting arc and the number of tokens in all its output places is increased by the weight of the arc connecting transition t_j with the corresponding place, i.e., the state signal changes according to

$$m_i(k+1) = m_i(k) - w(p_i, t_j) + w(t_j, p_i), i = 1, \ldots, n,$$

where $m_i(k)$ and $m_i(k+1)$ represent the numbers of tokens in place p_i before and after the firing of transition t_j, respectively.

Example 3.4 (Petri net). Given the Petri net graph introduced in Example 3.2 is endowed with the initial marking $m(0) = m^0 = [1, 0, 0]^T$. The corresponding Petri net is given in Subfig. 3.2(a). At this (initial) state only transition t_1 is enabled to fire. Firing this transition changes the state to $m(1) = [0, 1, 2]^T$, which is shown in Subfig. 3.2(b). At this state, transition t_2 is enabled to fire. Firing transition t_2, however, does not affect the overall marking, i.e., $m(2) = m(1) = [0, 1, 2]^T$.

It should also be noted that a transition enabled to fire might not actually do so. In fact, it is well possible that, in a certain state, several transitions are enabled simultaneously, and that firing one of them will disable the other ones. The corresponding transitions are said to be in conflict. In this thesis, however, the focus is on non-conflicting Petri nets. More precisely, only so called *event graphs* (or *synchronization graphs*) are considered.

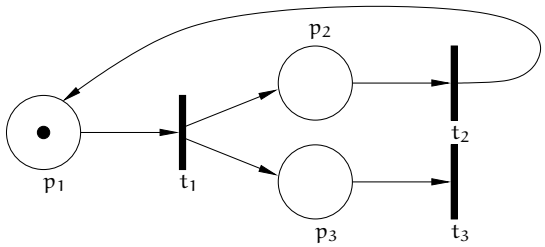


Figure 3.3: Simple event graph (Example. 3.6).

Definition 3.5 (Event graph). A Petri net (N, m^0) is called an event graph, if each place has exactly one input transition and one output transition, i.e.,

$$|I(p_i)| = |O(p_i)| = 1, \forall p_i \in P,$$

and if all arcs have weight 1. As mentioned above, an event graph cannot model conflicts (or decisions), but it does model synchronization effects.

Example 3.6 (Event graph). Obviously, the Petri net introduced in Example 3.4 is not an event graph, because place p_1 does not have an input transition, place p_2 on the other hand has two input transitions, and the arc connecting transition t_1 with p_3 has a weight $w(t_1, p_3) = 2$. An example of a simple event graph is given in Fig. 3.3.

3.1 Timed event graphs

Standard Petri nets (and also event graphs) only model the possible ordering of firings of transitions, but not the actual firing times. However, in many applications the specific firing times or the earliest possible firing times are of particular interest. Therefore, standard logical event graphs have been equipped with timing information. Two different approaches have been developed to include timing information, i.e., time can either be associated with transitions (representing transition delays) or with places (representing holding times). Equipping an event graph with either transition delays or holding times provides a so called *timed event graph (TEG)*. If every transition in a timed event graph, associated with a transition delay, has at least one input place, the transition delays can always be converted into holding times (by simply shifting each transition delay to all input places of the corresponding transition). However, in general, it is not possible to convert every TEG with holding times into a TEG with transition delays. Therefore, we will only consider timed event graphs with holding times.

In a TEG with holding times, a token entering a place p_i has to spend v_i time units before it can contribute to the firing of the output transition of p_i. The graphical representation of a part of a timed event graph with holding times is given in Fig. 3.4. The earliest time instant when place p_i receives its k^{th} token is denoted $\pi_i(k)$, and the resulting earliest time instant

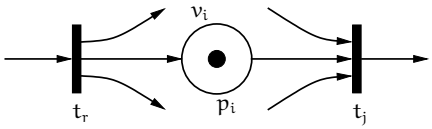


Figure 3.4: Part of a general timed event graph with holding times.

that the output transition t_j can fire for the k^{th} time is denoted $\tau_j(k)$ and can be determined by

$$\tau_j(k) = \max_{p_i \in I(t_j)} (\pi_i(k) + \nu_i), \tag{3.1}$$

i.e., a transition can fire for the k^{th} time as soon as all its input transitions received their k^{th} token and the corresponding holding times have elapsed. Similarly, the (earliest) time instants when place p_i receives its $(k + m_i^0)^{th}$ token can be determined by the earliest k^{th} firing time of the respective input transition t_r, i.e.,

$$\pi_i(k + m_i^0) = \tau_r(k), \quad t_r \in I(p_i). \tag{3.2}$$

Since every place in a TEG has exactly one input transition, it is possible to replace π_i in Eq. 3.1 with Eq. 3.2. Therefore, recursive equations for the (earliest) firing times of transitions in TEG can be obtained.

Remark 3.7 (Earliest firing rule). In the following it is assumed that transitions in timed event graphs always fire as soon as they are enabled.

Example 3.8 (Simple transportation network). Obviously, the earliest firing instants of transitions describe the dynamical behavior of a TEG. To illustrate this, a small example is introduced (borrowed from [10]).

Imagine a train network consisting of two stations and three lines, one inner loop and two outer loops. The inner loop has two rail tracks (one for each direction) and the outer loops have one rail track each. At the stations passengers are able to change lines. The basic structure of this train network is outlined in Fig. 3.5.

Initially, it is assumed that the train company operates a total of four trains, i.e., one train on each track. A train needs 3 time units to travel from station 1 to station 2 and 5 time units for the reverse travel. For the outer loop of station 1 (resp. station 2) the train needs 2 time units (resp. 3 time units). The aim is to implement a passenger-friendly timetable, where trains wait for each other at the stations to allow passengers to change from the inner to an outer loop or vice versa, i.e., the departure times of trains in each station shall be synchronized. As mentioned before, timed event graphs are suitable to model synchronization phenomena, and indeed it is rather straight forward to model the described train network as a TEG with holding

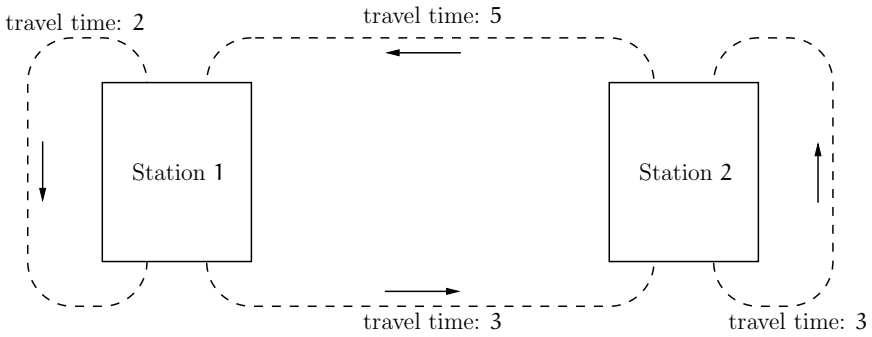


Figure 3.5: Simple transportation network taken from [10].

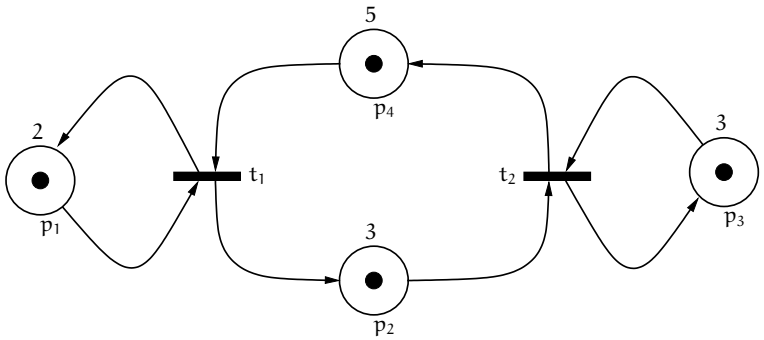


Figure 3.6: Timed event graph of the transportation network.

times (see Fig. 3.6). The tokens in places p_1 to p_4 represent trains on each of the four tracks and the transitions t_1 and t_2 represent the departure of trains from station 1 and station 2, respectively. The holding times associated to the places are equivalent to the corresponding travel times on the tracks. The earliest possible departure times for trains, i.e., the earliest firing instants for transitions t_1 and t_2 can be determined according to Eq. 3.1:

$$\tau_1(k) = \max(\pi_1(k) + 2, \pi_4(k) + 5)$$
$$\tau_2(k) = \max(\pi_2(k) + 3, \pi_3(k) + 3)$$

and the earliest time instants of tokens entering places p_1 to p_4 are

$$\pi_1(k + m_1^0) = \pi_1(k+1) = \tau_1(k)$$
$$\pi_2(k + m_1^0) = \pi_2(k+1) = \tau_1(k)$$
$$\pi_3(k + m_1^0) = \pi_3(k+1) = \tau_2(k)$$
$$\pi_4(k + m_1^0) = \pi_4(k+1) = \tau_2(k).$$

Therefore, the recursive equations for the firing instants of transitions t_1 and t_2 can be determined by

$$\begin{aligned}\tau_1(k+1) &= \max\left(\tau_1(k)+2, \tau_2(k)+5\right)\\ \tau_2(k+1) &= \max\left(\tau_1(k)+3, \tau_2(k)+3\right).\end{aligned}$$

Now, given initial departure times, i.e., the departure times for the first trains, $\tau_1(1) = \tau_2(1) = 0$, the following timetable can be achieved:

$$\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}5\\3\end{pmatrix}, \begin{pmatrix}8\\8\end{pmatrix}, \begin{pmatrix}13\\11\end{pmatrix}, \begin{pmatrix}16\\16\end{pmatrix}, \dots$$

Thus, on average trains are leaving station 1 and 2 every four time units. The obtained timetable is so-called 2-periodic. The train company, however, would most probably prefer a 1-periodic timetable, i.e., a train departs every 4 time units. This can be achieved by considering different initial departure times. If, for example, the departure times for the first trains is changed to $\tau_1(1) = 1$ and $\tau_2(1) = 0$. The resulting new timetable is

$$\begin{pmatrix}1\\0\end{pmatrix}, \begin{pmatrix}5\\4\end{pmatrix}, \begin{pmatrix}9\\8\end{pmatrix}, \begin{pmatrix}13\\12\end{pmatrix}, \begin{pmatrix}17\\16\end{pmatrix}, \dots$$

Clearly, a timed event graph is a suitable tool to model systems, in which synchronization of specific events plays a major role and decisions are not an issue.

3.2 Dioid model of timed event graphs

Taking a look at the recursive equations for the transition firing times of Example 3.8, it is easy to recognize that addition and the maximum operation are necessary to determine the desired timetable. Due to the max-operation, these equations are non-linear in conventional algebra, however, recalling Example 2.17 it is possible to rewrite such equations in $(\max, +)$-algebra, i.e., in the idempotent semiring $(\overline{\mathbb{Z}}_{\max}, \oplus, \otimes)$. The recursive equations for τ_1 and τ_2 are:

$$\begin{aligned}\tau_1(k+1) &= 2 \otimes \tau_1(k) \oplus 5 \otimes \tau_2(k)\\ \tau_2(k+1) &= 3 \otimes \tau_1(k) \oplus 3 \otimes \tau_2(k)\end{aligned}$$

which can be rewritten in matrix-vector form

$$\begin{aligned}x(k+1) &= \begin{pmatrix}2 & 5\\ 3 & 3\end{pmatrix} \otimes x(k)\\ x(k+1) &= A \otimes x(k)\end{aligned} \tag{3.3}$$

with $x(k) = (\tau_1(k)\ \tau_2(k))^T$. Obviously, Equation 3.3 is linear. It is also called "max-plus linear system". Within the framework of $(\max, +)$-algebra it is then possible to determine initial firing times for the transitions such that the system evolves in a 1-periodic manner. In conventional algebra this means that the following equation shall be satisfied

$$\tau_i(k+1) = \lambda + \tau_i(k), \qquad \begin{matrix} k = 1, 2, \dots \\ i = 1, 2, \dots, n. \end{matrix}$$

Rewriting this requirement in $(\max, +)$-algebra provides

$$x_i(k+1) = \lambda x_i(k), \qquad \begin{matrix} k = 1, 2, \dots \\ i = 1, 2, \dots, n \end{matrix}$$

or, equivalently

$$x(k+1) = \lambda x(k), \quad k = 1, 2, \dots$$

with $x(k) \in \overline{\mathbb{Z}}_{\max}^n$. This yields the max-plus eigenproblem. If, for a given matrix $A \in \overline{Z}_{max}^{n \times n}$, there exists $\xi \in \overline{Z}_{\max}^n$ and a scalar λ such that

$$A\xi = \lambda\xi,$$

λ is called eigenvalue and ξ eigenvector of matrix A. Consequently, choosing the initial firing times as an eigenvector, i.e., $x(1) = \xi$, results in

$$\begin{aligned} x(2) &= Ax(1) = \lambda x(1) \\ x(3) &= Ax(2) = A^2x(1) = \lambda^2 x(1) \\ &\vdots \\ x(k) &= Ax(k-1) = A^2x(k-2) = \dots = A^{(k-1)}x(1) = \lambda^{(k-1)}x(1), \end{aligned}$$

which is the desired 1-periodic behavior with a period length of λ. There are several algorithms to solve the max-plus eigenproblem, e.g., Howard's algorithm and the power algorithm. However, the eigenproblem in $(\max, +)$-algebra or other idempotent semirings is not part of this thesis. The interested reader may consult one of the many publications on this issue, e.g., [31].

Example 3.9 (Simple transportation network (continued)). Recalling the transportation network introduced in Example 3.8, the max-plus system matrix is

$$A = \begin{pmatrix} 2 & 5 \\ 3 & 3 \end{pmatrix}.$$

It turns out that $\xi = (1\ 0)^T$ is an eigenvector of matrix A, i.e., it satisfies $A \otimes \xi = \lambda \otimes \xi$. Not surprisingly, this confirms the observation of possible timetables. When the initial departure times are set to $x(1) = (1\ 0)^T$, the timetable has a 1-periodic behavior with a departure interval of $\lambda = 4$.

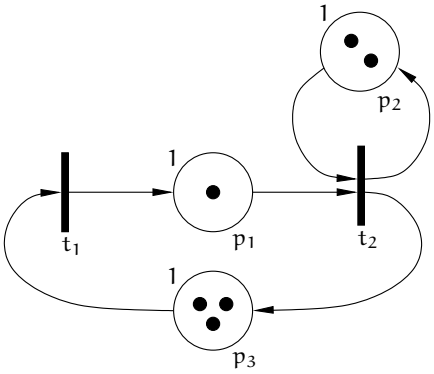


Figure 3.7: Simple timed event graph (Example 3.11).

Remark 3.10 (Eigenproblem in $(\max, +)$-algebra). A matrix $A \in \overline{\mathbb{Z}}_{\max}^{n \times n}$ is called *reducible* if there exists a permutation matrix P, i.e., a square matrix that has exactly one 1-element in each row and column and zeros elsewhere, such that $\tilde{A} = PAP^T$ is upper block-triangular. Otherwise, A is called *irreducible*. While a reducible matrix may have more than one eigenvalue, irreducible matrices have a unique eigenvalue but may possess several linearly independent eigenvectors.

In general it is possible to convert any timed event graph (as defined in this thesis) into a linear system in $(\max, +)$-algebra. In such a max-plus linear system the variable $x_i(k)$ refers to the earliest possible time instant that event x_i occurs for the k^{th} time, i.e., transition t_i fires for the k^{th} time. Therefore, $x(k)$ is also called a dater function, as it determines a specific (earliest possible) time (or date) for the occurrence of every event. Another possible way to describe timed event graphs is through so-called counter functions denoted $x(t)$. These counter functions determine the number of events that have occurred up to time t, i.e., $x_i(t)$ refers to the number of occurrences of event x_i up to time t. Note that in this case time is assumed to be discrete. Converting a timed event graph into a linear system of counter functions implies modeling the timed event graph as a min-plus linear system.

Example 3.11 (Timed event graphs and $(\min, +)$-algebra). Given the timed event graph shown in Fig. 3.7. The numbers of tokens in place p_i at time t is denoted by $\pi_i(t)$ and can be determined by

$$\begin{aligned} \pi_1(t) &= \tau_1(t) + m_1^0 = \tau_1(t) + 1 \\ \pi_2(t) &= \tau_2(t) + m_2^0 = \tau_2(t) + 2 \\ \pi_3(t) &= \tau_2(t) + m_3^0 = \tau_2(t) + 3 \end{aligned} \tag{3.4}$$

with $\tau_i(t)$ being the number of firings of transition t_i up to time t and m^0 being the vector of initial markings, i.e., the number of tokens in every place at time $t = 0$. Similar to Example 3.8,

it is also possible to determine the number of firings of every transition up to time t:

$$\begin{aligned}\tau_1(t) &= \pi_3(t-1)\\ \tau_2(t) &= \min\left(\pi_1(t-1), \pi_2(t-1)\right).\end{aligned} \tag{3.5}$$

Note that the number of firings at time t depends on the number of tokens that have been present in the places at time $t-1$. This is due to the fact that the holding time for every place is equal to one and therefore, only tokens that have entered a place and have remained (at least) one time unit in the corresponding place can contribute to the firing of a transition. Finally, replacing the π_i in Eq. 3.5 with Eq. 3.4, one obtains the recursive equations for the maximal number of firings of transition t_i up to time t:

$$\begin{aligned}\tau_1(t+1) &= \tau_2(t) + 3\\ \tau_2(t+1) &= \min\left(\tau_1(t)+1, \tau_2(t)+2\right).\end{aligned}$$

These equations can be rewritten as a min-plus linear system, i.e., in the idempotent semiring $\left(\overline{\mathbb{Z}}_{\min}, \oplus, \otimes\right)$:

$$\begin{aligned}x(t+1) &= \begin{pmatrix} \varepsilon & 3 \\ 1 & 2 \end{pmatrix} \otimes x(t)\\ x(t+1) &= A \otimes x(t),\end{aligned}$$

with $x(t) = (\tau_1(t) \;\; \tau_2(t))^{\mathsf{T}}$.

Consequently, it is possible to model the dynamic behavior of a timed event graph as a max-plus linear system as well as a min-plus linear system. As a matter of fact, it is also possible to achieve linear models of TEG in other dioids, e.g., $\overline{\mathbb{Z}}_{\max}[\![\gamma]\!]$ and $\overline{\mathbb{Z}}_{\min}[\![\delta]\!]$. Which dioid is used to model the dynamic behavior of a specific TEG depends on the system to be modeled itself but also on the biases of the user who wants to describe the system. The first issue becomes clear, when looking at a slightly bigger example.

Example 3.12 (Manufacturing systems and dioids). Taking a look at the TEG given in Fig. 3.8, which could possibly model a simple manufacturing system. The linear dynamical system in $(\max, +)$-algebra, with $x_i(k), u_j(k), y(k)$ being the earliest time instants that the transitions x_i, u_j, and y fire for the k^{th} time, is

$$\begin{aligned}x_1(k) &= 2u_1(k) \oplus 1x_2(k-1)\\ x_2(k) &= 1x_1(k)\\ x_3(k) &= 3u_2(k-1) \oplus ex_4(k-2)\\ x_4(k) &= 4x_3(k)\\ y(k) &= 2x_2(k-1) \oplus 1x_4(k)\end{aligned}$$

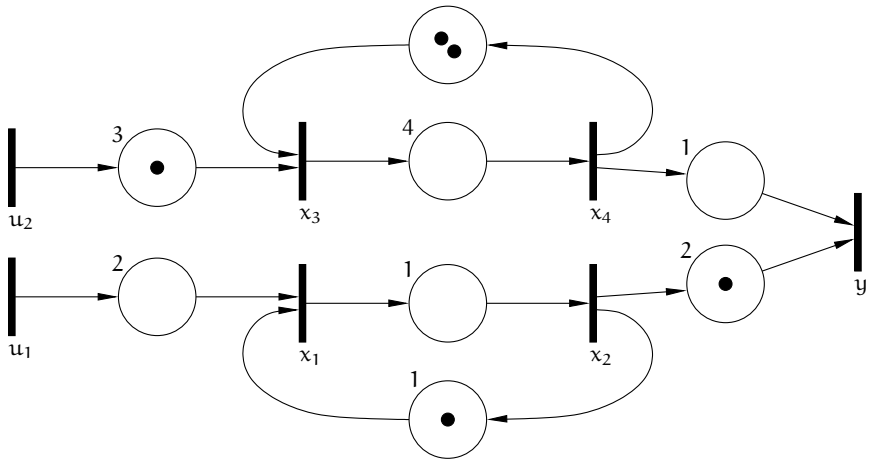


Figure 3.8: TEG of a simple manufacturing system (Example 3.12).

Rewriting the linear system in matrix-vector form, one obtains

$$\begin{aligned} x(k) &= A_0 x(k) \oplus A_1 x(k-1) \oplus A_2 x(k-2) \oplus B_0 u(k) \oplus B_1 u(k-1) \\ y(k) &= C_0 x(k) \oplus C_1 x(k-1) \end{aligned} \tag{3.6}$$

with

$$A_0 = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 4 & \varepsilon \end{pmatrix}, \quad A_1 = \begin{pmatrix} \varepsilon & 1 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \quad A_2 = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & e \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}$$

$$B_0 = \begin{pmatrix} 2 & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix}, B_1 = \begin{pmatrix} \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & 3 \\ \varepsilon & \varepsilon \end{pmatrix}, C_0 = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & 1 \end{pmatrix}, C_1 = \begin{pmatrix} \varepsilon & 2 & \varepsilon & \varepsilon \end{pmatrix}.$$

Clearly, Eq. 3.6 is an implicit equation, however, according to Example 2.49, the least fixed point can be used to solve this equation, i.e., to rewrite it in explicit form. Formally, one can write:

$$\begin{aligned} x(k) &= A_0 x(k) \oplus A_1 x(k-1) \oplus A_2 x(k-2) \oplus B_0 u(k) \oplus B_1 u(k-1) \\ &= A_0^* \left(A_1 x(k-1) \oplus A_2 x(k-2) \oplus B_0 u(k) \oplus B_1 u(k-1) \right) \\ &= A_0^* A_1 x(k-1) \oplus A_0^* A_2 x(k-2) \oplus A_0^* B_0 u(k) \oplus A_0^* B_1 u(k-1) \end{aligned}$$

with

$$
\begin{aligned}
A_0^* &= \bigoplus_{i=0}^{\infty} A_0^i \\
&= \begin{pmatrix} e & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & e \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 4 & \varepsilon \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix} \oplus \dots \\
&= \begin{pmatrix} e & \varepsilon & \varepsilon & \varepsilon \\ 1 & e & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon \\ \varepsilon & \varepsilon & 4 & e \end{pmatrix}.
\end{aligned}
$$

Consequently,

$$
\begin{aligned}
x(k) = &\begin{pmatrix} \varepsilon & 1 & \varepsilon & \varepsilon \\ \varepsilon & 2 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix} x(k-1) \oplus \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & e \\ \varepsilon & \varepsilon & \varepsilon & 4 \end{pmatrix} x(k-2) \oplus \dots \\
&\begin{pmatrix} 2 & \varepsilon \\ 3 & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix} u(k) \oplus \begin{pmatrix} \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & 3 \\ \varepsilon & 7 \end{pmatrix} u(k-1).
\end{aligned}
$$

Then, we can transform this second order system to a first order system by suitably defining a state space, e.g., by defining a state vector $\tilde{x}(k) = \left(x(k)^T \; x(k-1)^T \; u(k)^T\right)^T$. The resulting first order system is

$$
\begin{aligned}
\tilde{x}(k) &= \tilde{A}\tilde{x}(k-1) \oplus \tilde{B}u(k) \\
y(k) &= \tilde{C}\tilde{x}(k)
\end{aligned}
$$

where

$$\tilde{A} = \begin{pmatrix} \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & 3 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 4 & \varepsilon & 7 \\ e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \quad \tilde{B} = \begin{pmatrix} 2 & \varepsilon \\ 3 & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ e & \varepsilon \\ \varepsilon & e \end{pmatrix},$$

$$\tilde{C} = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & 1 & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}.$$

Obviously, modeling a TEG as given in Fig. 3.8 in $(\max, +)$-algebra leads to a sparse matrix and a rather large state space dimension. However, other idempotent semirings such as the dioid $\overline{\mathbb{Z}}_{\max}[\![\gamma]\!]$ are well capable to describe such a TEG with a smaller state space. Another advantage of using the smallest possible state space is that the physical meaning of each transition of the TEG is preserved in the resulting dioid model. The system equations in the dioid $\overline{\mathbb{Z}}_{\max}[\![\gamma]\!]$ can be achieved by applying the γ-transform (see Def. 2.72) to Eq. 3.6. Formally, we get

$$\begin{aligned} \overline{x}(\gamma) &= A_0\overline{x}(\gamma) \oplus A_1\gamma\overline{x}(\gamma) \oplus A_2\gamma^2\overline{x}(\gamma) \oplus B_0\overline{u}(\gamma) \oplus B_1\gamma\overline{u}(\gamma) \\ \overline{y}(\gamma) &= C_0\overline{x}(\gamma) \oplus C_1\gamma\overline{x}(\gamma) \end{aligned}$$

or equivalently

$$\begin{aligned} \overline{x}(\gamma) &= \underbrace{\left(A_0 \oplus \gamma A_1 \oplus \gamma^2 A_2\right)}_{\overline{A}(\gamma)}\overline{x}(\gamma) \oplus \underbrace{(B_0 \oplus \gamma B_1)}_{\overline{B}(\gamma)}\overline{u}(\gamma) \\ \overline{y}(\gamma) &= \underbrace{(C_0 \oplus \gamma C_1)}_{\overline{C}(\gamma)}\overline{x}(\gamma) \end{aligned}$$

with

$$\overline{A}(\gamma) = \begin{pmatrix} \varepsilon & 1\gamma & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \gamma^2 \\ \varepsilon & \varepsilon & 4 & \varepsilon \end{pmatrix}, \overline{B}(\gamma) = \begin{pmatrix} 2 & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & 3\gamma \\ \varepsilon & \varepsilon \end{pmatrix}, \overline{C}(\gamma) = \begin{pmatrix} \varepsilon & 2\gamma & \varepsilon & 1 \end{pmatrix}.$$

In the remainder of this thesis timed event graphs will be described in the idempotent semiring $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$, which is an efficient way to model any TEG. Therefore, from this point on, all dioid operations and system descriptions are meant to be in $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$ unless otherwise indicated.

Example 3.13 (Manufacturing systems in $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$). The example of the manufacturing system given as a TEG in Fig. 3.8 can be written as a linear system in $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$. Formally, we get

$$\begin{aligned} x(\gamma,\delta) &= A(\gamma,\delta)x(\gamma,\delta) \oplus B(\gamma,\delta)u(\gamma,\delta) \\ y(\gamma,\delta) &= C(\gamma,\delta)x(\gamma,\delta) \end{aligned}$$

with

$$A(\gamma,\delta) = \begin{pmatrix} \varepsilon & \gamma\delta & \varepsilon & \varepsilon \\ \delta & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \gamma^2 \\ \varepsilon & \varepsilon & \delta^4 & \varepsilon \end{pmatrix}, \; B(\gamma,\delta) = \begin{pmatrix} \delta^2 & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \gamma\delta^3 \\ \varepsilon & \varepsilon \end{pmatrix},$$
$$C(\gamma,\delta) = \begin{pmatrix} \varepsilon & \gamma\delta^2 & \varepsilon & \delta \end{pmatrix}.$$

3.2.1 Causality, Periodicity, Realizability, Rationality

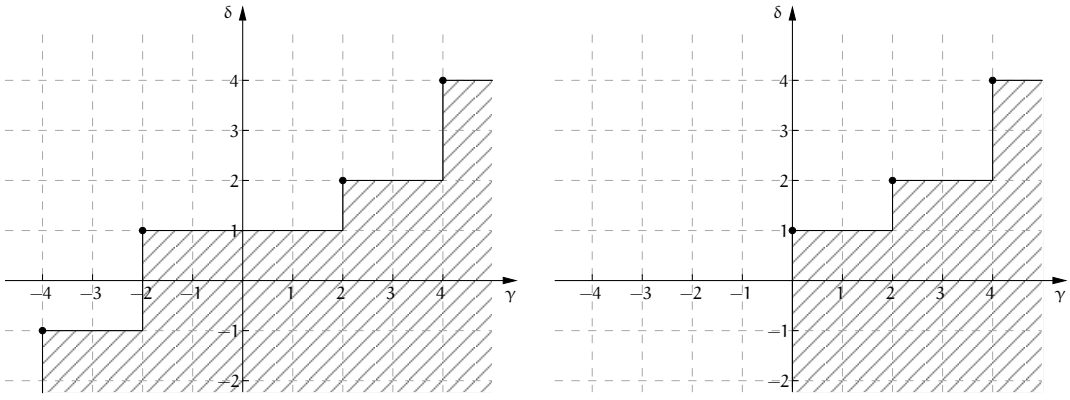
As mentioned in Chapter 2 the elements in the dioid $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$ are formal power series. When it comes to modeling real system, the question on causality, periodicity, realizability, and rationality of periodic power series in $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$ has to be addressed.

Definition 3.14 (Causality of a series in $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!][2]$). A series $s \in \mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$ is causal if $s = \varepsilon$ or if both $\mathrm{val}_\gamma(s) \geqslant 0$ and $s \geq \gamma^{\mathrm{val}_\gamma(s)}\delta^0$, where $\mathrm{val}_\gamma(s)$ refers to the valuation in γ of series s.

Consequently, the exponents of all monomials composing a causal series s are greater or equal to zero and all series composed of monomials with exponents greater or equal to zero are causal. The set of causal elements of $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$ has a complete semiring structure and is denoted $\mathcal{M}_{in}^{max\ddagger}[\![\gamma,\delta]\!]$. Obviously, $\mathcal{M}_{in}^{max\ddagger}[\![\gamma,\delta]\!]$ is a complete sub-dioid of $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$.

Remark 3.15 (Causality of a matrix in $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$). A matrix A with entries in $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$ is causal, if all its entries are causal.

Remark 3.16 (Causal projection [17]). The canonical injection $\Pi_{\mathcal{M}_{in}^{max\ddagger}[\![\gamma,\delta]\!]} : \mathcal{M}_{in}^{max\ddagger}[\![\gamma,\delta]\!] \to \mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$ is residuated and its residual is denoted $\mathrm{Pr}_{caus}^{\ddagger} : \mathcal{M}_{in}^{max}[\![\gamma,\delta]\!] \to \mathcal{M}_{in}^{max\ddagger}[\![\gamma,\delta]\!]$. Formally, the series $\mathrm{Pr}_{caus}^{\ddagger}(s)$ is the greatest causal series less or equal to series $s \in \mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$.



(a) $s = \gamma^{-4}\delta^{-1} \oplus \gamma^{-2}\delta^{1} \oplus \gamma^{2}\delta^{2} \oplus \gamma^{4}\delta^{4}$ (b) $\mathrm{Pr}^{\ddagger}_{\mathrm{caus}}(s) = \gamma^{0}\delta^{1} \oplus \gamma^{2}\delta^{2} \oplus \gamma^{4}\delta^{4}$

Figure 3.9: Causal projection of a (non-causal) series $s \in \mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$.

Example 3.17 (Causal projection of a series). Given a non-causal series $s = \gamma^{-4}\delta^{-1} \oplus \gamma^{-2}\delta^{1} \oplus \gamma^{2}\delta^{2} \oplus \gamma^{4}\delta^{4} \in \mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$. Its causal projection $s^{\ddagger} = \mathrm{Pr}^{\ddagger}_{\mathrm{caus}}(s) = \gamma^{0}\delta^{1} \oplus \gamma^{2}\delta^{2} \oplus \gamma^{4}\delta^{4} \in \mathcal{M}^{\mathrm{ax}\ddagger}_{\mathrm{in}}[\![\gamma,\delta]\!]$. Graphically, the causal projection of a series is the series that covers the same area in the first quadrant but is devoid of any points on the left hand side of the Cartesian system. In Fig. 3.9 the series s and its causal projection $\mathrm{Pr}^{\ddagger}_{\mathrm{caus}}(s)$ is given.

Definition 3.18 (Periodic series in $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$ [15]). Often systems modeled in $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$ have a periodic behavior, which can be represented by a periodic series. A series $s \in \mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$ is said to be periodic if it can be written as $s = p \oplus q \otimes r^{*}$, where p is a polynomial referring to a transient phase, e.g., start-up of the system, q is a polynomial representing the periodical behavior, i.e., the pattern that will be repeated periodically, and $r = \gamma^{\nu}\delta^{\tau}$ is a monomial describing the periodicity. Then, the ratio ν/τ is the throughput, i.e., once the periodic regime is reached an event occurs ν times every τ time units. Accordingly, the inverse ratio τ/ν is the asymptotic slope of the series.

Remark 3.19 (Periodic matrices in $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$). A matrix $A \in \mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$ is said to be periodic, if all its entries are periodic series.

Example 3.20 (Periodic series in $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$). Considering the series $s \in \mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$

$$s = e \oplus \gamma\delta \oplus \gamma^{2}\delta^{4} \oplus \gamma^{4}\delta^{5} \oplus \gamma^{5}\delta^{6} \oplus \gamma^{6}\delta^{7} \oplus \gamma^{8}\delta^{9} \oplus \gamma^{9}\delta^{10} \oplus \gamma^{11}\delta^{12} \oplus \gamma^{12}\delta^{13} \oplus \dots$$

This series is a periodic series and can be written

$$s = \underbrace{e \oplus \gamma\delta \oplus \gamma^{2}\delta^{4} \oplus \gamma^{4}\delta^{5}}_{p} \oplus \underbrace{\left(\gamma^{5}\delta^{6} \oplus \gamma^{6}\delta^{7}\right)}_{q} \underbrace{\left(\gamma^{3}\delta^{3}\right)^{*}}_{r^{*}}.$$

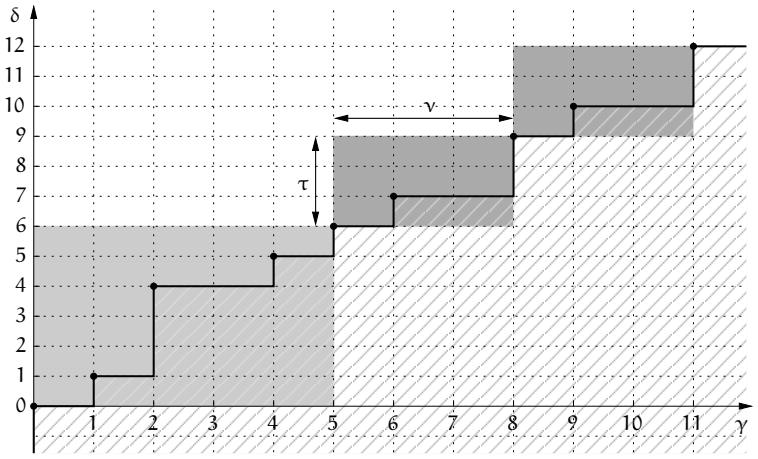


Figure 3.10: Graphical representation of a periodic series in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$.

The graphical representation of this series is given in Fig. 3.10.

Definition 3.21 (Realizability of a series in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ [15]). A series $s \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is said to be realizable if there exist two square $(n \times n)$ matrices A_1 and A_2 with Boolean entries and two $(n \times 1)$ respectively $(1 \times n)$ matrices B and C with Boolean entries such that $s = C(\gamma A_1 \oplus \delta A_2)^* B$.

Remark 3.22. This definition implies that a series $s \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is realizable if there exists a timed event graph with single input and single output which has a transfer relation depicted by this series.

Example 3.23 (Realizable series in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$). The series $s = e \oplus (\gamma\delta^2 \oplus \gamma^2\delta^4)(\gamma^2\delta^5)^*$ is realizable according to Def. 3.21. A possible timed event graph with single input and single output whose transfer relation is depicted by this series is given in Fig. 3.11. This TEG can be "expanded" to the TEG given in Fig. 3.12. The single input single output transfer function of

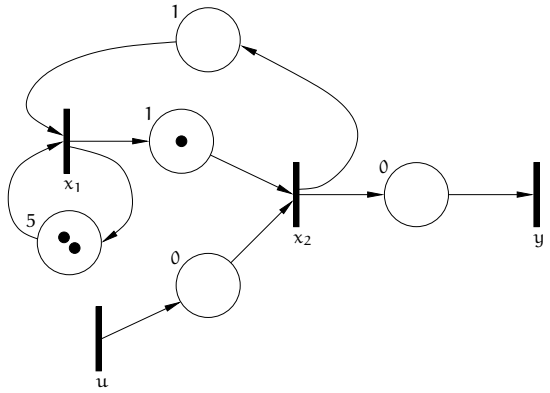


Figure 3.11: Timed event graph with transfer relation $s = e \oplus (\gamma\delta^2 \oplus \gamma^2\delta^4)(\gamma^2\delta^5)^*$.

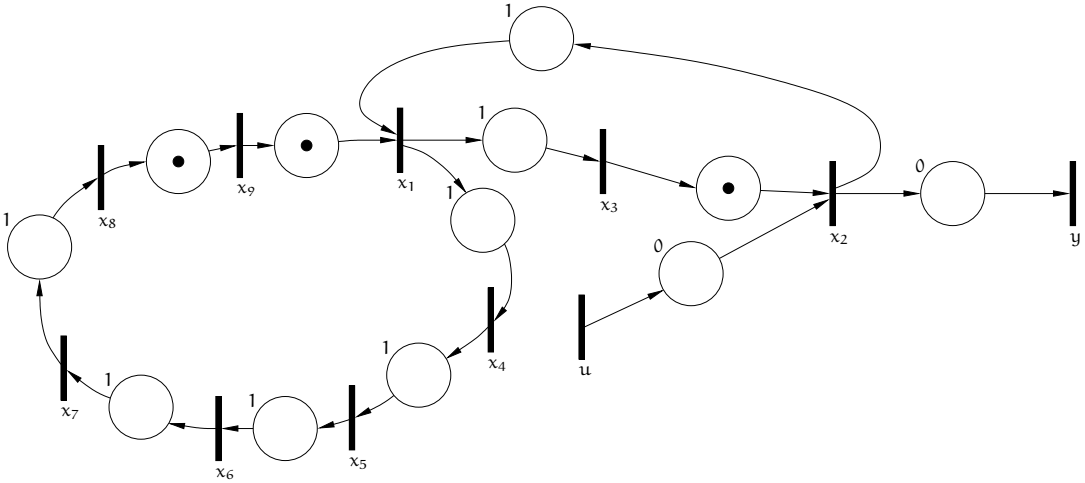


Figure 3.12: Expanded timed event graph with transfer relation $s = e \oplus (\gamma\delta^2 \oplus \gamma^2\delta^4)(\gamma^2\delta^5)^*$.

this expanded TEG can then be written in the form $s = C(\gamma A_1 \oplus \delta A_2)^* B$ with

$$\gamma A_1 \oplus \delta A_2 = \begin{pmatrix} \varepsilon & \delta & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma \\ \varepsilon & \varepsilon & \gamma & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \delta & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma & \varepsilon \end{pmatrix}$$

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

$$C = \begin{pmatrix} \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}$$

and indeed $s = C(\gamma A_1 \oplus \delta A_2)^* B = e \oplus (\gamma\delta^2 \oplus \gamma^2\delta^4)(\gamma^2\delta^5)^*$.

Remark 3.24 (Realizability of a matrix in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$). A matrix A with entries in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is said to be realizable if all its entries are realizable.

Definition 3.25 (Rationality of a series in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ [15]). A series $s \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is said to be rational if it belongs to the rational closure of the subset $\mathcal{T} = \{\varepsilon, e, \gamma, \delta\}$, i.e., it can be written with a finite number of operations $\{\oplus, \otimes, *\}$ on elements of the set $\mathcal{T}$.

Remark 3.26 (Rationality of a matrix in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$). A matrix $A \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is said to be rational if all its entries are rational.

Theorem 3.27 (Causality, Periodicity, Realizability, Rationality). Given a series $s \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$, the following statements are equivalent [2, 15]:

- s is causal and periodic
- s is realizable
- s is rational.

3.3 "Extended" timed event graphs

As mentioned before, timed event graphs are a suitable tool to model discrete event systems characterized by synchronization and delay phenomena. Among the systems that can be modeled by TEG are traffic systems, computer communication systems, production lines, and flows in networks.

In general, many different properties of real systems can be modeled using TEG. Apart from the TEG's basic structure, one can use the holding times to model the minimal time that has to elapse between the firing of two transitions, i.e., between the occurrence of two events. Furthermore, tokens play a major role in modeling different characteristics of the system. If tokens represent an entity of the real system (as they did in the transportation system of Example 3.8), they can be used to model the number of entities, e.g., trains in the railway network, and their corresponding position within the network. However, tokens cannot only be used to model entities moving through the system, but also other characteristics such as the capacity of a resource in a production line, or similarly, the maximal number of items in a part of the system.

Nonetheless, despite the vast variety of properties that can be modeled by (standard) TEG, there are some specific features of real systems, that cannot be included in a (standard) TEG. In automated manufacturing systems, for example, the processing of a part on a resource may have to be performed within a time interval also called time window. Thus, there exists not only a minimal time that a token has to stay in a place but also an upper bound for the time, by which the token has to be removed from the place by its output transition. Similarly, while it is possible to model the maximal number of tokens in a part of the TEG, it is not possible to model a minimal number of tokens that have to be present in this part of the TEG. This means that a transition fires at least a certain number of times more often than another transition.

While the issue of timed event graphs with time window constraints has been handled in several publications, e.g., [37, 38, 53], the latter issue concerning a minimum number of tokens between two transitions has, to our knowledge, not yet been addressed.

3.3.1 Motivating example

Example 3.28 (Nested schedules in manufacturing systems). Given a simple manufacturing system which consists of a single resource. However, each part is processed twice on this resource. The processing times for the two processing steps, also called activities, are 2 time units and 1 time unit, respectively. In between these two activities the part is moved to a buffer of infinite size and has to rest there for 3 time units. Now the company wants to produce these parts in an efficient way. A rather naive approach would be to start producing one part after the other. The corresponding TEG of such a system is given in Fig. 3.13. In this figure, transition u is the input of raw material, transition y refers to the finishing of a part and transitions x_1 and x_2 (resp. x_3 and x_4) model the start and finish events of activity one (respectively activity two). The activities are indicated by the dashed boxes and the buffer is represented by the place in between the two activities. The places containing a token model the capacity of the resource – the resource can handle one activity at a time, i.e., activity act_1 may not start before act_1 of the previous part has been finished and act_2 of the previous part has been finished. Similarly, act_2 may only start if the same activity of the previous part has been finished. Modeling the

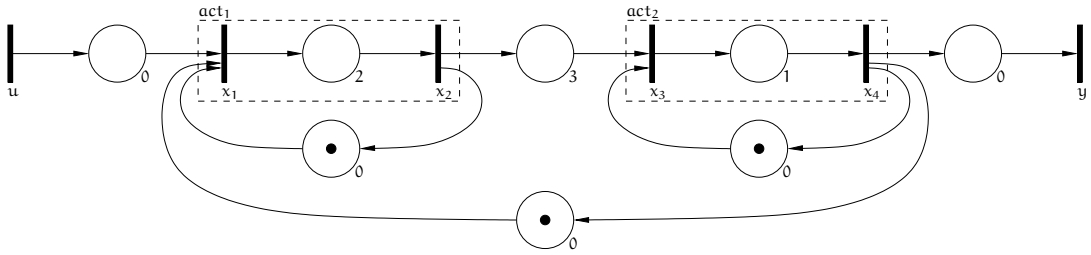


Figure 3.13: TEG of a simple manufacturing system.

dynamic behavior of the TEG as a linear system in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ results in

$$x = \underbrace{\begin{pmatrix} \varepsilon & \gamma & \varepsilon & \gamma \\ \delta^2 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \delta^3 & \varepsilon & \gamma \\ \varepsilon & \varepsilon & \delta & \varepsilon \end{pmatrix}}_{A} x \oplus \underbrace{\begin{pmatrix} e \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix}}_{B} u$$

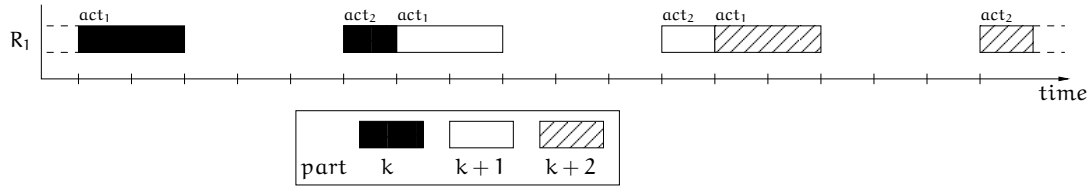
$$y = \underbrace{\begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & e \end{pmatrix}}_{C} x.$$

The smallest solution of the implicit equation is

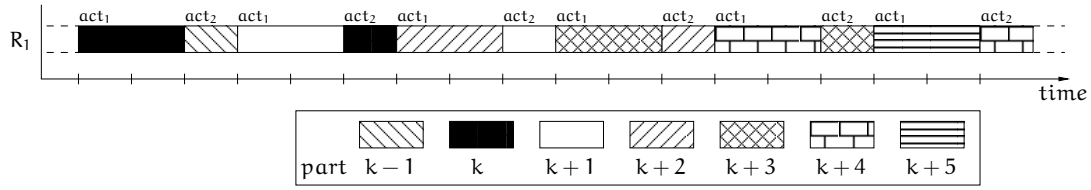
$$\begin{aligned} x &= A^* B u \\ &= \begin{pmatrix} (\gamma\delta^6)^* & \gamma\delta^4(\gamma\delta^6)^* & \gamma\delta(\gamma\delta^6)^* & \gamma(\gamma\delta^6)^* \\ \delta^2(\gamma\delta^6)^* & (\gamma\delta^6)^* & \gamma\delta^3(\gamma\delta^6)^* & \gamma\delta^2(\gamma\delta^6)^* \\ \delta^5(\gamma\delta^6)^* & \delta^3(\gamma\delta^6)^* & (\gamma\delta^6)^* & \gamma\delta^5(\gamma\delta^6)^* \\ \delta^6(\gamma\delta^6)^* & \delta^4(\gamma\delta^6)^* & \delta(\gamma\delta^6)^* & (\gamma\delta^6)^* \end{pmatrix} \begin{pmatrix} e \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix} u \\ &= \begin{pmatrix} (\gamma\delta^6)^* \\ \delta^2(\gamma\delta^6)^* \\ \delta^5(\gamma\delta^6)^* \\ \delta^6(\gamma\delta^6)^* \end{pmatrix} u \end{aligned}$$

And the input-output behavior of the system is

$$\begin{aligned} y &= Cx = CA^* B u \\ &= \underbrace{\delta^6\left(\gamma\delta^6\right)^*}_{H} u, \end{aligned}$$



(a) Gantt chart of a simple schedule.



(b) Gantt chart of a nested schedule.

Figure 3.14: Gantt chart of possible schedules of the manufacturing system.

with H being the input-output relation. Looking at the input-output relation one can easily see, that six time units after the input has fired, the first part is finished. Furthermore, given that enough raw material is available, i.e., u fires such that it does not slow the system down, the throughput of the system is $\left(\gamma\delta^6\right)^*$, i.e, one part will be finished every six time units. Often the operation of a manufacturing system is visualized by a so called Gantt chart. The Gantt chart of this example is given in Fig. 3.14(a). Using a Gantt chart makes it quite easy to get a feeling about the efficiency of a system's operation. Looking at the Gantt chart of the example it is obvious, that the capacity utilization of resource R_1 is rather low. In fact, the capacity utilization is 50 per cent, i.e., only 50 per cent of the time the resource is busy (and consequently, 50 per cent of the time the resource is idle). To increase the efficiency of the manufacturing system, the user may want to try to reduce the idle time by using a different schedule. For example, the idle time between the execution of act_1 and act_2 of part k may be used to execute act_1 of the next part, i.e., part $k+1$, and consequently act_2 of part k will be executed between act_1 and act_2 of part $k+1$. A schedule where at least one activity of part $k+1$ is "squeezed" in between activities of part k is said to be a nested schedule. Clearly, the resulting Gantt chart of the manufacturing system shown in Fig. 3.14(b) is a nested schedule, as two activities (one of part $k-1$ and one of part $k+1$) are executed in between the activities of part k. It can easily be seen that the capacity utilization is increased to 100 per cent, i.e., the proposed nested schedule is an optimal schedule, as it is impossible to find a schedule with a greater throughput. The question is how the proposed nested schedule can be modeled as a TEG or a linear system in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$. To answer this question, one has to find the dependencies for the firing of transitions in the system. First of all, it is clear, that the (minimal) timing information for the production of a single part remains unchanged, i.e., in

$\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ one gets:

$$
\begin{aligned}
x_2 &\geq \delta^2 x_1 && \text{processing time of } \mathtt{act}_1 \\
x_3 &\geq \delta^3 x_2 && \text{resting time between } \mathtt{act}_1 \text{ and } \mathtt{act}_2 \\
x_4 &\geq \delta x_3 && \text{processing time of } \mathtt{act}_2.
\end{aligned}
$$

Furthermore, as the capacity of the resource does not change, an activity for part k can still only start if the same activity for part $k-1$ has been finished, i.e., one gets

$$
\begin{aligned}
x_1 &\geq \gamma x_2 \\
x_3 &\geq \gamma x_4.
\end{aligned}
$$

Thus, to this point nothing has changed with respect to the dependencies of the simple schedule. What changes are the dependencies of different activities executed on different parts to be processed. Looking at the Gantt chart of the nested schedule, one can easily determine, that $\mathtt{act}_2$ of part $k-1$ can start as soon as $\mathtt{act}_1$ of part k has been finished. Similarly, $\mathtt{act}_1$ of part $k+1$ cannot start until $\mathtt{act}_2$ of part $k-1$ has been finished. Formally, this means

$$
\begin{aligned}
x_3 &\geq \gamma^{-1} x_2 \\
x_1 &\geq \gamma^2 x_4.
\end{aligned}
$$

Furthermore, the input and output dependencies do not change, i.e.

$$
\begin{aligned}
x_1 &\geq u \\
y &= x_4
\end{aligned}
$$

Combining all dependencies results in

$$
\begin{aligned}
x_1 &\geq \gamma x_2 \oplus \gamma^2 x_4 \oplus u \\
x_2 &\geq \delta^2 x_1 \\
x_3 &\geq \delta^3 x_2 \oplus \gamma x_4 \oplus \gamma^{-1} x_2 = \left(\delta^3 \oplus \gamma^{-1}\right) x_2 \oplus \gamma x_4 \\
x_4 &\geq \delta^1 x_3
\end{aligned}
$$

or, equivalently, in vector-matrix form

$$
x \geq \underbrace{\begin{pmatrix} \varepsilon & \gamma & \varepsilon & \gamma^2 \\ \delta^2 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \delta^3 \oplus \gamma^{-1} & \varepsilon & \gamma \\ \varepsilon & \varepsilon & \delta^1 & \varepsilon \end{pmatrix}}_{A} x \oplus \begin{pmatrix} e \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix} u \tag{3.7}
$$

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

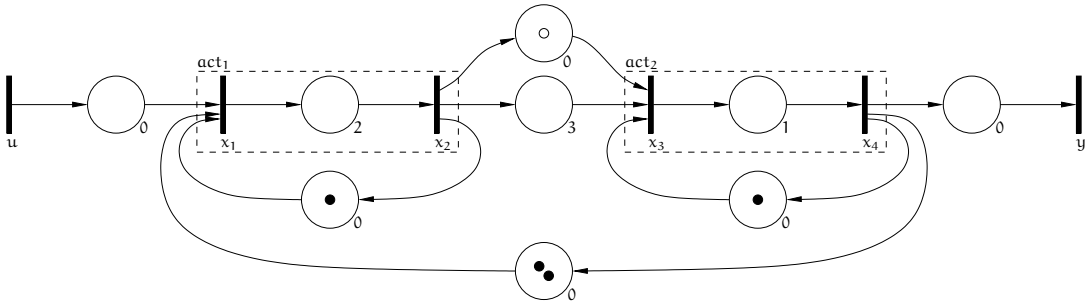


Figure 3.15: "Extended" timed event graph of the nested schedule.

Looking at the system matrix one realizes that element $A_{32} = \delta^3 \oplus \gamma^{-1}$ is different from all other entries. First of all, its a polynomial and not a monomial, but that just indicates that there should be two different places between transition x_3 and transition x_2 of the corresponding timed event graph. The second thing that is unusual is the negative exponent in γ. With respect to timed event graphs, this basically means that there is a *negative number of tokens* in a place between transitions x_2 and x_3. In the standard definition of TEG, however, negative numbers of tokens are not allowed as they would represent a non-causal behavior of the system. The corresponding "extended" timed event graph, i.e., with a negative number of tokens represented by the corresponding number of white bullets is given in Fig. 3.15. Ignoring the non-causality with respect to the number of tokens for now, it is possible to obtain the smallest solution of Eq. 3.7 using the star algorithm, i.e.,

$$
\begin{aligned}
x &= A^* B u \\
&= \begin{pmatrix} (\gamma\delta^3)^* & \gamma\delta\,(\gamma\delta^3)^* & \gamma^2\delta\,(\gamma\delta^3)^* & \gamma^2\,(\gamma\delta^3)^* \\ \delta^2\,(\gamma\delta^3)^* & (\gamma\delta^3)^* & \gamma^2\delta^3\,(\gamma\delta^3)^* & \gamma^2\delta^2\,(\gamma\delta^3)^* \\ \gamma^{-1}\delta^2\,(\gamma\delta^3)^* & \gamma^{-1}\,(\gamma\delta^3)^* & (\gamma\delta^3)^* & \gamma\delta^2\,(\gamma\delta^3)^* \\ \gamma^{-1}\delta^3\,(\gamma\delta^3)^* & \gamma^{-1}\delta\,(\gamma\delta^3)^* & \delta\,(\gamma\delta^3)^* & (\gamma\delta^3)^* \end{pmatrix} \begin{pmatrix} e \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix} u \\
&= \begin{pmatrix} (\gamma\delta^3)^* \\ \delta^2\,(\gamma\delta^3)^* \\ \gamma^{-1}\delta^2\,(\gamma\delta^3)^* \\ \gamma^{-1}\delta^3\,(\gamma\delta^3)^* \end{pmatrix} u \\
y &= Cx = CA^* B u \\
&= \gamma^{-1}\delta^3 \left(\gamma\delta^3\right)^* u.
\end{aligned}
$$

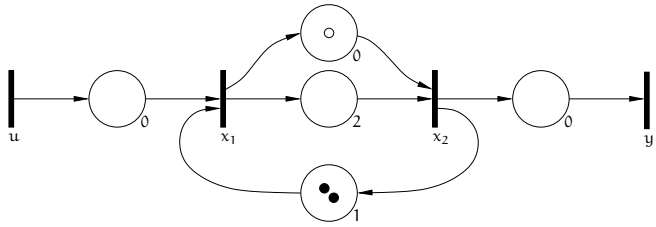


Figure 3.16: Simple TEG with negative token.

Clearly, the input output behavior of this system is non-causal. Applying the causal projection $\mathrm{Pr}^{\ddagger}_{caus}$ results in

$$\begin{aligned} y^{\ddagger} &= \mathrm{Pr}^{\ddagger}_{caus}(y) \\ &= \delta^{6}\left(\gamma\delta^{3}\right)^{*} u \end{aligned}$$

Comparing this result with the result obtained for the simple schedule, it becomes clear that the first part is finished after 6 time units in both schedules, i.e., the processing time for a single part cannot be reduced. However, while for the simple schedule one part is finished every 6 time units, the nested schedule allows to finish one part every 3 time units (after the first part has been finished). Thus, operating the system with the nested schedule is evidently more efficient than with the simple schedule. Obviously, these behaviors are also represented in the Gantt charts of the two different schedules (see Fig. 3.14).

3.3.2 Negative number of tokens

As mentioned before, negative numbers of tokens are not allowed in standard Petri nets or timed event graphs as the state signal, i.e., the number of tokens, $m : \mathbb{N}_0 \to \mathbb{N}_0^n$ (see Def. 3.3). In general, a negative number of tokens represents a non-causal behavior. Given that tokens represent physical entities moving through the system, e.g., trains in a transportation network, this definitely makes sense. In this case the number of -1 tokens would represent one train less than no train, which obviously is not possible. Looking, however, at the motivating example, where tokens do not solely represent physical parts moving through the system, a negative number of tokens may not be "as non-causal" as in the train example. Clearly, the nested schedule given in Fig. 3.14(b) is causal with respect to time. To further investigate the issue of negative numbers of tokens another small example is introduced.

Example 3.29 (Simple TEG with a negative number of tokens). Taking a look at a timed event graph with a negative number of tokens given in Fig. 3.16. The corresponding dependencies for

$x \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ are

$$x_1 \geq \gamma^2\delta x_2 \oplus u$$
$$x_2 \geq \left(\gamma^{-1} \oplus \delta^2\right) x_1.$$

With respect to the TEG, this means that x_1 should (after an initial phase) always fire once more often than x_2. Despite the fact that this is modeled with -1 token, this clearly represents a possible behavior of the system. In $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ the complete system description is

$$\begin{aligned} x &\geq Ax \oplus Bu \\ x &\geq \begin{pmatrix} \varepsilon & \gamma^2\delta \\ \gamma^{-1} \oplus \delta^2 & \varepsilon \end{pmatrix} x \oplus \begin{pmatrix} e \\ \varepsilon \end{pmatrix} u \\ y &= Cx = \begin{pmatrix} \varepsilon & e \end{pmatrix} x \end{aligned} \tag{3.8}$$

According to the TEG and given an input $u = e \oplus \gamma\delta^4 \oplus \gamma^2\delta^5 \oplus \gamma^3\delta^6 \oplus (\gamma^4\delta^7)(\gamma\delta^3)^*$, the resulting system state satisfies:

$$x \geq \begin{pmatrix} e \oplus \gamma\delta^4 \oplus \gamma^2\delta^5 \oplus \gamma^3\delta^7 \oplus \gamma^4\delta^8 \oplus (\gamma^5\delta^{10})(\gamma\delta^3)^* \\ \delta^4 \oplus \gamma\delta^6 \oplus \gamma^2\delta^7 \oplus \gamma^3\delta^9 \oplus (\gamma^4\delta^{10})(\gamma\delta^3)^* \end{pmatrix} \tag{3.9}$$

Important to note is that the first possible firing of x_2 cannot be earlier than 2 time units after the first and 0 time units after the second firing of x_1. Consequently, the first firing of x_2 is at time 4.

The interesting question is, how can we determine this evolution in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$? First of all, looking at inequality 3.8 it is clear that matrix A is non-causal w.r.t. Def. 3.14. Applying the causal projection $\mathrm{Pr}_{caus}^{\ddagger}$ the following (causal) system is obtained

$$\hat{x} \geq \underbrace{\begin{pmatrix} \varepsilon & \gamma^2\delta \\ \delta^2 & \varepsilon \end{pmatrix}}_{\mathrm{Pr}_{caus}^{\ddagger}(A)} \hat{x} \oplus \begin{pmatrix} e \\ \varepsilon \end{pmatrix} u,$$

which basically means that the place with -1 tokens is removed. Then, applying the Kleene star we get the least solution

$$\begin{aligned} \hat{x} &= \left(\mathrm{Pr}_{caus}^{\ddagger}(A)\right)^* Bu \\ &= \begin{pmatrix} (\gamma^2\delta^3)^* & \gamma^2\delta(\gamma^2\delta^3)^* \\ \delta^2(\gamma^2\delta^3)^* & (\gamma^2\delta^3)^* \end{pmatrix} \begin{pmatrix} e \\ \varepsilon \end{pmatrix} u \\ &= \begin{pmatrix} (\gamma^2\delta^3)^* \\ \delta^2(\gamma^2\delta^3)^* \end{pmatrix} u. \end{aligned}$$

Inserting the given input u the system state can be determined

$$\hat{x} = \begin{pmatrix} e \oplus \gamma\delta^4 \oplus \gamma^2\delta^5 \oplus \gamma^3\delta^7 \oplus \gamma^4\delta^8 \oplus (\gamma^5\delta^{10})(\gamma\delta^3)^* \\ \delta^2 \oplus \gamma\delta^6 \oplus \gamma^2\delta^7 \oplus \gamma^3\delta^9 \oplus \gamma^4\delta^{10} \oplus (\gamma^5\delta^{12})(\gamma\delta^3)^* \end{pmatrix}.$$

Obviously, $\hat{x} \prec x$, where x represents the right hand side of inequality 3.9, i.e., the obtained system state is not equivalent with the desired state. Especially, when looking at the first firing of x_2 one notices that it fires earlier than expected. This is not surprising as the causal projection removed the place with the -1 token which means that the first firing of x_2 is not dependent on the second firing of x_1.

Another option would be to first compute A^* and then apply the causal projection as A^* is also not causal w.r.t. Def. 3.14. This results in

$$\begin{aligned}
\tilde{x} &= \mathrm{Pr}^{\ddagger}_{\mathrm{caus}}(A^*)Bu \\
&= \mathrm{Pr}^{\ddagger}_{\mathrm{caus}}\left(\begin{pmatrix} (e \oplus \gamma\delta)(\gamma^2\delta^3)^* & (\gamma^2\delta \oplus \gamma^3\delta^2)(\gamma^2\delta^3)^* \\ (\gamma^{-1} \oplus \delta^2)(\gamma^2\delta^3)^* & (e \oplus \gamma\delta)(\gamma^2\delta^3)^* \end{pmatrix}\right)\begin{pmatrix} e \\ \varepsilon \end{pmatrix} u \\
&= \begin{pmatrix} (e \oplus \gamma\delta)(\gamma^2\delta^3)^* & (\gamma^2\delta \oplus \gamma^3\delta^2)(\gamma^2\delta^3)^* \\ (\delta^2 \oplus \gamma\delta^3)(\gamma^2\delta^3)^* & (e \oplus \gamma\delta)(\gamma^2\delta^3)^* \end{pmatrix}\begin{pmatrix} e \\ \varepsilon \end{pmatrix} u \\
&= \begin{pmatrix} (e \oplus \gamma\delta)(\gamma^2\delta^3)^* \\ (\delta^2 \oplus \gamma\delta^3)(\gamma^2\delta^3)^* \end{pmatrix} u.
\end{aligned}$$

Inserting the given input u the system state can be determined

$$\tilde{x} = \begin{pmatrix} e \oplus \gamma\delta^4 \oplus \gamma^2\delta^5 \oplus \gamma^3\delta^7 \oplus \gamma^4\delta^8 \oplus (\gamma^5\delta^{10})(\gamma\delta^3)^* \\ \delta^2 \oplus \gamma\delta^6 \oplus \gamma^2\delta^7 \oplus \gamma^3\delta^9 \oplus \gamma^4\delta^{10} \oplus (\gamma^5\delta^{12})(\gamma\delta^3)^* \end{pmatrix}.$$

Comparing the $\tilde{x}$ with the desired state x we get

$$\tilde{x} = \hat{x} \prec x.$$

Thus, also this approach does not cover the correct system behavior. The same result can be obtained, if the causal projection of A^*B is considered, i.e., $\tilde{x} = \mathrm{Pr}^{\ddagger}_{\mathrm{caus}}(A^*B)u$.

If we, however, do not apply the causal projection, i.e., use the apparently non-causal system matrix A, we get

$$\begin{aligned}
\check{x} &= A^*Bu \\
&= \begin{pmatrix} (e \oplus \gamma\delta)(\gamma^2\delta^3)^* & (\gamma^2\delta \oplus \gamma^3\delta^2)(\gamma^2\delta^3)^* \\ (\gamma^{-1} \oplus \delta^2)(\gamma^2\delta^3)^* & (e \oplus \gamma\delta)(\gamma^2\delta^3)^* \end{pmatrix}\begin{pmatrix} e \\ \varepsilon \end{pmatrix} u
\end{aligned}$$

and applying the given input u results in

$$\check{x} = \begin{pmatrix} e \oplus \gamma\delta^4 \oplus \gamma^2\delta^5 \oplus \gamma^3\delta^7 \oplus \gamma^4\delta^8 \oplus (\gamma^5\delta^{10})(\gamma\delta^3)^* \\ \gamma^{-1} \oplus \delta^4 \oplus \gamma\delta^6 \oplus \gamma^2\delta^7 \oplus \gamma^3\delta^9 \oplus (\gamma^4\delta^{10})(\gamma\delta^3)^* \end{pmatrix},$$

which, obviously, is not causal. Applying the causal projection we get

$$\mathrm{Pr}^{\ddagger}_{\mathrm{caus}}(\check{x}) = \begin{pmatrix} e \oplus \gamma\delta^4 \oplus \gamma^2\delta^5 \oplus \gamma^3\delta^7 \oplus \gamma^4\delta^8 \oplus (\gamma^5\delta^{10})(\gamma\delta^3)^* \\ \delta^4 \oplus \gamma\delta^6 \oplus \gamma^2\delta^7 \oplus \gamma^3\delta^9 \oplus (\gamma^4\delta^{10})(\gamma\delta^3)^* \end{pmatrix}.$$

Clearly, the obtained result $\mathrm{Pr}^{\ddagger}_{\mathrm{caus}}(\check{x})$ is equivalent to the desired output given in Eq. 3.9.

Summarizing, we have seen that an obviously possible behavior of a system may have a non-causal representation w.r.t. Def. 3.14 in $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$. Applying the causal projection on this system representation results in a causal linear system in $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$, but does not model the desired system behavior. However, using the non-causal system representation, it is possible to determine a system state which is also non-causal and applying the causal projection to this non-causal system behavior results in the desired system behavior.

Consequently, as the obviously causal behavior of the system can only be represented by a non-causal linear system in $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$, the definition of causality has to be reconsidered. Basically, the causality of a series in $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$ has to be confirmed, if the series represents the dynamical behavior of a specific event or in terms of TEG the firing of a transition. In this case negative exponents in γ or δ are non-causal since it would mean that there are negative events or negative time. If, however, a series describes the relation between two different events, i.e., the firing instants of two different transitions in TEG, a negative exponent in γ may be causal, as this simply means that one of these transitions has to fire a certain number of times more often than another transition. Negative exponents in δ, on the other hand, are non-causal even if the corresponding series models the relation of occurrences of different events. Such a negative exponent in δ means that one of these events occurs a certain time before the occurrence of the other event, which indicates that according to the firing instant of the second transition the past firing instant of the first transition may have to be adjusted.

Regarding to this, a new definition of causality for series and matrices of series representing the transfer relation between two events has to be introduced. In the following we will call such a series *transfer function.*

3.3.3 Causality, Periodicity, Realizability, Rationality

Definition 3.30 (Causality of a transfer function in $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$). A transfer function $s \in \mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$ is causal if $s = \varepsilon$ or if $s \succeq \gamma^{\mathrm{val}_\gamma(s)}$, i.e., the exponents of δ in all monomials composing the transfer function s are greater or equal to zero. The set of causal transfer elements of $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$ has a complete semiring structure and is denoted $\mathcal{M}^{\mathrm{ax}\overline{+}}_{\mathrm{in}}[\![\gamma,\delta]\!]$. Obviously, $\mathcal{M}^{\mathrm{ax}\overline{+}}_{\mathrm{in}}[\![\gamma,\delta]\!]$ is a complete sub-dioid of $\mathcal{M}^{\mathrm{ax}}_{\mathrm{in}}[\![\gamma,\delta]\!]$.

Remark 3.31 (Transfer matrices in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$). A matrix A with entries in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is called *transfer matrix* if it describes the relation between two vectors, e.g., $y = Ax$.

Definition 3.32 (Causality of a transfer matrix in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$). A transfer matrix A with entries in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is causal, if all its entries are causal w.r.t. Def. 3.30.

Definition 3.33 (Causal projection of transfer function). The canonical injection $\Pi_{\mathcal{M}_{in}^{ax\mp}[\![\gamma,\delta]\!]} : \mathcal{M}_{in}^{ax\mp}[\![\gamma,\delta]\!] \to \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is residuated and its residual is denoted $\mathrm{Pr}_{caus}^{\mp} : \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!] \to \mathcal{M}_{in}^{ax\mp}[\![\gamma,\delta]\!]$. Formally, the series $\mathrm{Pr}_{caus}^{\mp}(s)$ is the greatest causal transfer function less or equal to series $s \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$.

Definition 3.34 (Rationality of a transfer function in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$). A transfer function $s \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is rational if it belongs to the rational closure of the subset $\mathcal{T}^{\mp} = \{\varepsilon, e, \gamma^{-1}, \gamma, \delta\}$.

Remark 3.35 (Rationality of a transfer matrix in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$). A transfer matrix A with entries in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is rational if all its entries are rational transfer functions w.r.t. Def. 3.34.

Definition 3.36 (Realizability of a transfer function in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$). A transfer function $s \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is said to be realizable if there exists three square $(n \times n)$ matrices A_1, A_2, and A_3 with Boolean entries and two $(n \times 1)$ respectively $(1 \times n)$ matrices B and C with Boolean entries such that $s = C(\gamma A_1 \oplus \delta A_2 \oplus \gamma^{-1} A_3)^* B$.

Remark 3.37. According to this definition, a transfer function $s \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is realizable, if there exists a timed event graph (possibly with a negative number of tokens) with single input and single output, whose transfer relation can be depicted by the series s.

Remark 3.38 (Realizability of a transfer matrix in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$). A transfer matrix A with entries in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is said to be realizable if all its entries are realizable transfer functions w.r.t. Def. 3.36.

Definition 3.39 (Periodicity of a transfer function in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$). A transfer function $s \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is said to be periodic if it can be written as $s = p \oplus q \otimes r^*$, where p is a polynomial referring to a transient phase, q is a polynomial representing the periodic behavior, and r is a monomial describing the periodicity.

Remark 3.40 (Periodicity of a transfer matrix in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$). A transfer matrix A with entries in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is said to be periodic if all its entries are periodic transfer functions.

Remark 3.41. The definition of periodic transfer functions coincides with the definition of periodic series as given in Def. 3.18.

Theorem 3.42 (Causality, Periodicity, Realizability, Rationality). Given a transfer function $s \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$, the following statements are equivalent:

(i) s is realizable

(ii) s is rational

(iii) s is causal and periodic.

Proof. The proof of this theorem is strongly based on the proof of Theorem 3.27 given in [2, 15].

The implication (i) ⇒ (ii) is straightforward, i.e., if $s = C(\gamma A_1 \oplus \delta A_2 \oplus \gamma^{-1} A_3)^* B$ with all entries in A_1, A_2, A_3, B, and C are either ε or e, then s is in the rational closure of $\mathcal{T}^{\mp} = \{\varepsilon, e, \gamma^{-1}, \gamma, \delta\}$.

The converse implication (ii) ⇒ (i) follows from the fact that the rational closure of the set $\mathcal{T}^{\mp}$ coincides with the set of elements ξ which can be written as

$$\xi = C_\xi A_\xi^* B_\xi$$

where A_ξ is a $n_\xi \times n_\xi$ matrix with entries in $\mathcal{T}^{\mp}$, and C_ξ and B_ξ are $1 \times n_\xi$ and $n_\xi \times 1$ matrices with entries in $\mathcal{C}$ and $\mathcal{B}$, with $\mathcal{C}$ and $\mathcal{B}$ being subsets of $\mathcal{T}^{\mp}$ which contain ε and e, and n_ξ is an arbitrary but finite integer number [2, 15]. Setting $\mathcal{B} = \mathcal{C} = \{\varepsilon, e\}$, it remains to split up matrix A_ξ into $\gamma A_1 \oplus \delta A_2 \oplus \gamma^{-1} A_3$, which is straightforward.

The implication (iii) ⇒ (ii) is also quite obvious. If s can be written as $s = p \oplus q \otimes r^*$ with p and q being polynomials in $\mathcal{M}_{in}^{max^{\mp}}[\![\gamma, \delta]\!]$ and r being a monomial in $\mathcal{M}_{in}^{max^{\mp}}[\![\gamma, \delta]\!]$, then it is evident that s is an element of the rational closure of $\mathcal{T}^{\mp}$.

The implication (ii) ⇒ (iii) follows from the fact that, since $\mathcal{M}_{in}^{max}[\![\gamma, \delta]\!]$ is a commutative dioid (in the scalar case), a rational series s can be written as [2, 15]

$$s = \bigoplus_{i \in I} \gamma^{\alpha_i} \delta^{\beta_i} \left(\bigoplus_{j \in J_i} \gamma^{\nu_j} \delta^{\tau_j} \right)^* , \tag{3.10}$$

where I and $J_i, i \in I$, are finite sets, α_i and ν_j are integers and β_i and τ_j are non-negative integers. Since $s \in \mathcal{M}_{in}^{max}[\![\gamma, \delta]\!]$ and $\mathcal{M}_{in}^{max}[\![\gamma, \delta]\!]$ is a complete commutative dioid, (3.10) can be written as

$$s = \bigoplus_{i \in I} \gamma^{\alpha_i} \delta^{\beta_i} \bigotimes_{j \in J_i} (\gamma^{\nu_j} \delta^{\tau_j})^* . \tag{3.11}$$

To complete the proof, it is necessary to show that (3.11) can be written as $s = p \oplus q \otimes r^*$, which basically means that we have to show that a single monomial $(\gamma^{\nu_j} \delta^{\tau_j})^*$, with τ_j/ν_j representing the maximal asymptotic slope of the series, "absorbs" all other similar terms in sums and products. For positive integers ν_j and τ_j, this has been shown in [15].

For terms with negative integers ν_j, two different cases have to be considered.

(I) If there exists a term $\gamma^{\nu_j} \delta^{\tau_j}$ with $\nu_j < 0$ and $\tau_j > 0$, the corresponding Kleene star is $(\gamma^{\nu_j} \delta^{\tau_j})^* = \top$, and $\top$ dominates all other terms on which the star operation is applied, i.e., $\top \otimes a^* = a^* \otimes \top = \top, \forall a \in \mathcal{M}_{in}^{max^{\mp}}[\![\gamma, \delta]\!]$.

(II) If such a term (with $\nu_j < 0$ and $\tau_j > 0$) does not exist but there exists at least one term with $\nu_j < 0$ and $\tau_j = 0$, other terms with positive exponents in γ have to be considered. More precisely, if there exists $j \in J_i$ with $\nu_j < 0$ and $\tau_j = 0$ and $k \neq j$, $k \in J_i$ with $\nu_k \geqslant 0$ and $\tau_k \geqslant 0$, two further cases have to be considered:

(II.i) If there exists $j \in J_i$ with $\nu_j < 0$ and $\tau_j = 0$ and a $k \neq j$, $k \in J_i$ with $\nu_k \geqslant 0$ and $\tau_k = 0$, i.e., $(\gamma^{\nu_j}\delta^{\tau_j})^* \otimes (\gamma^{\nu_k}\delta^{\tau_k})^* = (\gamma^{\nu_j}\delta^0)^* \otimes (\gamma^{\nu_k}\delta^0)^* = (\gamma^{\nu_j})^* \otimes (\gamma^{\nu_k})^*$, then, since $\gamma^{\nu_j} \succeq \gamma^{\nu_k}$ and according to Lem. 2.53, $(\gamma^{\nu_j})^* \otimes (\gamma^{\nu_k})^* = (\gamma^{\nu_j})^*$. Consequently, for this case, $(\gamma^{\nu_j})^* = \gamma^{-\infty}$ "absorbs" all other terms.

(II.ii) If there exists $j \in J_i$ with $\nu_j < 0$ and $\tau_j = 0$ and a $k \neq j$, $k \in J_i$ with $\nu_k \geqslant 0$ and $\tau_k > 0$, then $(\gamma^{\nu_j})^* \otimes (\gamma^{\nu_k}\delta^{\tau_k})^* = \top$. This can easily be shown by reconsidering the definition of the Kleene star (see Def. 2.47), i.e., since $\nu_j < 0$, $(\gamma^{\nu_j})^* = \gamma^{-\infty}$ and

$$\begin{aligned}(\gamma^{\nu_j})^* \otimes (\gamma^{\nu_k}\delta^{\tau_k})^* &= \gamma^{-\infty} \otimes (\gamma^{\nu_k}\delta^{\tau_k})^* \\ &= \gamma^{-\infty} \otimes (e \oplus \gamma^{\nu_k}\delta^{\tau_k} \oplus \gamma^{2\nu_k}\delta^{2\tau_k} \oplus \dots) \\ &= \gamma^{-\infty} \oplus \gamma^{-\infty}\delta^{\tau_k} \oplus \gamma^{-\infty}\delta^{2\tau_k} \oplus \dots \\ &= \gamma^{-\infty}\delta^{\infty} = \top.\end{aligned}$$

Thus, also in this case the dominating term is $\top$.

Consequently, if there exists at least one term with a negative exponent in γ, i.e., $\nu_j < 0$, the dominating term is either $\top$ or (in the special case that all exponents of δ are equal to 0) $\gamma^{-\infty}$. Otherwise, the term which represents the maximal asymptotic slope of the series "absorbs" all other terms as shown in [15]. □

One additional issue, which has to be considered when dealing with TEG with negative numbers of tokens, is the issue of blocking. Clearly, if a transition shall fire a certain number of times more often than another transition but is itself dependent on the firing of this other transition, there are some cases where the system may be blocked. Note that, in this work, a system is said to be blocked if there are constraints between the firings of transitions such that these transitions wait for each other to fire and therefore never do so. Further note that a system may also "run" into a blocking situation after the corresponding transitions have fired for a certain number of times. To illustrate the issue of blocking, a few small examples are considered in the following.

Example 3.43. Given a simple TEG with negative and positive numbers of tokens as displayed in Fig. 3.17. Counting the number of tokens in the loop we get the *loop token weight* which in

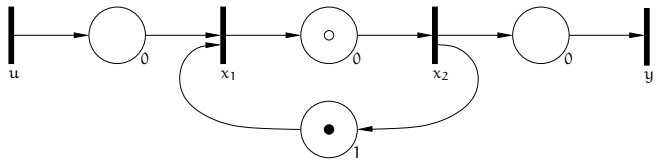


Figure 3.17: TEG with negative and positive numbers of tokens and a loop token weight of 0.

this example is $-1 + 1 = 0$. The corresponding system in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is

$$x \succeq Ax \oplus Bu$$

with least solution

$$x = A^*Bu$$

with

$$\begin{aligned} A &= \begin{pmatrix} \varepsilon & \gamma\delta \\ \gamma^{-1} & \varepsilon \end{pmatrix} \\ A^* &= \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \gamma\delta \\ \gamma^{-1} & \varepsilon \end{pmatrix} \oplus \begin{pmatrix} \delta & \varepsilon \\ \varepsilon & \delta \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \gamma\delta^2 \\ \gamma^{-1}\delta & \varepsilon \end{pmatrix} \oplus \dots \\ &\quad \dots \oplus \begin{pmatrix} \delta^2 & \varepsilon \\ \varepsilon & \delta^2 \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \gamma\delta^3 \\ \gamma^{-1}\delta^2 & \varepsilon \end{pmatrix} \oplus \begin{pmatrix} \delta^3 & \varepsilon \\ \varepsilon & \delta^3 \end{pmatrix} \oplus \dots \\ &= \begin{pmatrix} \delta^\infty & \gamma\delta^\infty \\ \gamma^{-1}\delta^\infty & \delta^\infty \end{pmatrix} \end{aligned}$$

and

$$\begin{aligned} A^*B &= \begin{pmatrix} \delta^\infty & \gamma\delta^\infty \\ \gamma^{-1}\delta^\infty & \delta^\infty \end{pmatrix} \begin{pmatrix} e \\ \varepsilon \end{pmatrix} \\ &= \begin{pmatrix} \delta^\infty \\ \gamma^{-1}\delta^\infty \end{pmatrix} \end{aligned}$$

Multiplying any reasonable input signal $u \succ \varepsilon$ to this transfer relation results in some x where all entries x_i are of the form $\gamma^{m_i}\delta^\infty$, with m_i being an integer number. With respect to timed event graphs this corresponds to a blocking behavior as the earliest possible occurrence of the $(m_i + 1)$-st firing of transition x_i is at time $t = \infty$, i.e., at the "end of time".

Example 3.44. Given a very similar TEG with a loop token weight of 0 but without any delays within the loop (see Fig. 3.18). The corresponding system in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is

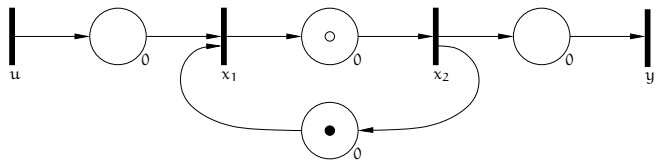


Figure 3.18: TEG with a loop token weight of 0 and without delays.

$$x \geq Ax \oplus Bu$$

with least solution

$$x = A^* B u$$

with

$$\begin{aligned} A &= \begin{pmatrix} \varepsilon & \gamma \\ \gamma^{-1} & \varepsilon \end{pmatrix} \\ A^* &= \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \gamma \\ \gamma^{-1} & \varepsilon \end{pmatrix} \oplus \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \gamma \\ \gamma^{-1} & \varepsilon \end{pmatrix} \oplus \dots \\ &= \begin{pmatrix} e & \gamma \\ \gamma^{-1} & e \end{pmatrix} \end{aligned}$$

and

$$\begin{aligned} A^* B &= \begin{pmatrix} e & \gamma \\ \gamma^{-1} & e \end{pmatrix} \begin{pmatrix} e \\ \varepsilon \end{pmatrix} \\ &= \begin{pmatrix} e \\ \gamma^{-1} \end{pmatrix} \end{aligned}$$

Thus, for any reasonable input signal $u \succ \varepsilon$ the system state will result in $x_1 = u$ and $x_2 = \gamma^{-1} u = \gamma^{-1} x_1$. Consequently, independent of the input signal the time instant of the $(k+1)$-st firing of transition x_1 coincides with the time instant of the k-th firing of transition x_2. With respect to the system in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ the system is not blocked.

Remark 3.45 (Special case loop token weight of 0). Looking at Ex. 3.44 we have shown that, with respect to the system in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$, blocking is not an issue. The only constraint is that the time instants of the $(k+1)$-st and the k-th firings of transitions x_1 and x_2 coincide, i.e., $x_1 = \gamma x_2$. Looking at the corresponding TEG (see Fig. 3.18), blocking clearly is an issue. In TEG a transition is enabled as soon as all its firing conditions are fulfilled. In Ex. 3.44 transition x_2 is enabled to fire for the k-th time as soon as transition x_1 has fired for the $(k+1)$-st time, transition x_1, however, is enabled to fire for the $(k+1)$-st time as soon as transition x_2 has fired for the k-th time. Consequently, the firings of these transitions depend on each other and since it is not possible to fulfill either firing condition prior to the firing of the corresponding transition, the TEG is blocked. In $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$, on the other hand, it is not necessary to fulfill the firing condition prior to the firing but it is sufficient to fulfill the firing condition at the very moment of the firing and, therefore, the transitions can fire as long as their firing instants coincide.

The TEG in Ex. 3.43 (see Fig. 3.17) also has a loop token weight of 0, however, within this loop there are delays and, consequently, the corresponding firing instants of the transitions can not coincide. Therefore, the TEG given in Fig. 3.17 is blocked.

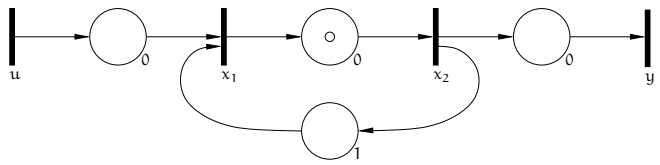


Figure 3.19: TEG with a negative number of tokens and a loop token weight of -1.

Example 3.46. Given a TEG with only a negative number of tokens (see Fig. 3.19). The loop token weight in this example is -1 and the corresponding system in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is

$$x \geq Ax \oplus Bu$$

with least solution

$$x = A^*Bu$$

with

$$\begin{aligned}
A &= \begin{pmatrix} \varepsilon & \delta \\ \gamma^{-1} & \varepsilon \end{pmatrix} \\
A^* &= \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \delta \\ \gamma^{-1} & \varepsilon \end{pmatrix} \oplus \begin{pmatrix} \gamma^{-1}\delta & \varepsilon \\ \varepsilon & \gamma^{-1}\delta \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \gamma^{-1}\delta^2 \\ \gamma^{-2}\delta & \varepsilon \end{pmatrix} \oplus \dots \\
&\quad \dots \oplus \begin{pmatrix} \gamma^{-2}\delta^2 & \varepsilon \\ \varepsilon & \gamma^{-2}\delta^2 \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \gamma^{-2}\delta^3 \\ \gamma^{-3}\delta^2 & \varepsilon \end{pmatrix} \oplus \begin{pmatrix} \gamma^{-3}\delta^3 & \varepsilon \\ \varepsilon & \gamma^{-3}\delta^3 \end{pmatrix} \oplus \dots \\
&= \begin{pmatrix} \gamma^{-\infty}\delta^{\infty} & \gamma^{-\infty}\delta^{\infty} \\ \gamma^{-\infty}\delta^{\infty} & \gamma^{-\infty}\delta^{\infty} \end{pmatrix} = \begin{pmatrix} \top & \top \\ \top & \top \end{pmatrix}
\end{aligned}$$

and

$$\begin{aligned}
A^*B &= \begin{pmatrix} \top & \top \\ \top & \top \end{pmatrix} \begin{pmatrix} e \\ \varepsilon \end{pmatrix} \\
&= \begin{pmatrix} \top \\ \top \end{pmatrix}
\end{aligned}$$

Multiplying any reasonable $u > \varepsilon$ to this transfer relation results in a state x with $x_i = \top = \gamma^{-\infty}\delta^{\infty}$. This, however, means that the $-\infty$ occurrence of transition x_i occurs at the earliest at time $t = \infty$, i.e., the "beginning of the evolution" occurs at the earliest at the "end of time". Thus, the corresponding timed event graph is blocked.

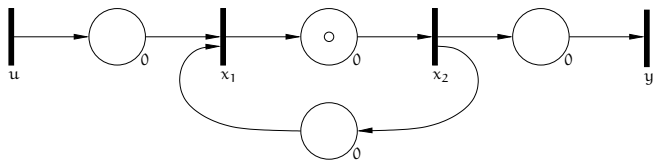


Figure 3.20: TEG with a negative number of tokens, a loop token weight of -1 and without delays.

Example 3.47. In another example a TEG is considered which has a loop token weight of -1 but without any delays. The considered TEG is shown in Fig. 3.20). The corresponding system in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is

$$x \geq Ax \oplus Bu$$

with least solution

$$= A^*Bu$$

with

$$A = \begin{pmatrix} \varepsilon & e \\ \gamma^{-1} & \varepsilon \end{pmatrix}$$

$$A^* = \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & e \\ \gamma^{-1} & \varepsilon \end{pmatrix} \oplus \begin{pmatrix} \gamma^{-1} & \varepsilon \\ \varepsilon & \gamma^{-1} \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \gamma^{-1} \\ \gamma^{-2} & \varepsilon \end{pmatrix} \oplus \dots$$
$$\dots \oplus \begin{pmatrix} \gamma^{-2} & \varepsilon \\ \varepsilon & \gamma^{-2} \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \gamma^{-2} \\ \gamma^{-3} & \varepsilon \end{pmatrix} \oplus \begin{pmatrix} \gamma^{-3} & \varepsilon \\ \varepsilon & \gamma^{-3} \end{pmatrix} \oplus \dots$$
$$= \begin{pmatrix} \gamma^{-\infty} & \gamma^{-\infty} \\ \gamma^{-\infty} & \gamma^{-\infty} \end{pmatrix}$$

and

$$A^*B = \begin{pmatrix} \gamma^{-\infty} & \gamma^{-\infty} \\ \gamma^{-\infty} & \gamma^{-\infty} \end{pmatrix} \begin{pmatrix} e \\ \varepsilon \end{pmatrix}$$
$$= \begin{pmatrix} \gamma^{-\infty} \\ \gamma^{-\infty} \end{pmatrix}$$

Multiplying any reasonable $u > \varepsilon$ to this transfer relation results in a state x with $x_i = \gamma^{-\infty}\delta^{t_i}$, with t_i being an integer number. This indicates that this system is also blocked.

Looking at the examples it becomes clear that to avoid blocking of the system the loop token weight has to be larger than 0. Systems with a loop token weight smaller than 0 are blocking and systems with a loop token weight equivalent to 0 are blocking if the corresponding loop contains any delays.

A system with more than one loop is non-blocking if every loop has either a loop token weight larger than 0 or a loop token weight of 0 but without any delays.

Once a causal model has been determined the question is, how a controller can be computed to manipulate the system such that a predefined goal will be achieved. In the following chapter some results from control theory in the dioid setting will be recalled.

4

Control of Systems in a Dioid Setting

Within the last two decades an extensive theory for control of systems in a dioid setting has been established. Among the developed approaches are feedforward control, state feedback control, output feedback control, but also an approach for model predictive control and feedback control based on the results of an observer. In this chapter, the basics of control theory for systems in a dioid setting are recalled.

In general, it is assumed that a linear model in a dioid framework is given. The considered model is of the form

$$\begin{aligned} x &\geq Ax \oplus Bu \\ y &= Cx \end{aligned} \tag{4.1}$$

with $A \in \mathcal{D}^{n\times n}$, $B \in \mathcal{D}^{n\times p}$, $C \in \mathcal{D}^{q\times n}$, the state vector $x \in \mathcal{D}^{n}$, the output vector $y \in \mathcal{D}^{q}$, and the input vector $u \in \mathcal{D}^{p}$. According to Theorem 2.45 the smallest solution for this system is

$$\begin{aligned} x &= A^{*}Bu \\ y &= CA^{*}Bu \end{aligned} \tag{4.2}$$

where $H = CA^{*}B$ is the input-output relation of the system.

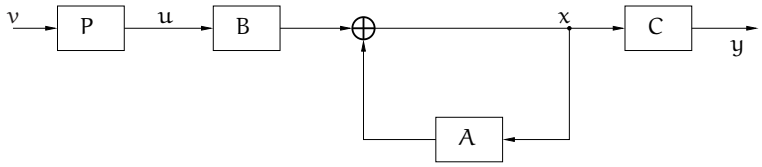


Figure 4.1: System structure with pre-filter.

4.1 Feedforward control

4.1.1 Optimal open-loop control

In this approach, the desired output z of a given linear system in a dioid setting is given a priori and the goal is to find the optimal input u_{opt} to achieve this output. The input is supposed to be optimal in terms of a *just-in-time* criterion, i.e., every input shall occur as late as possible while ensuring that the corresponding output satisfies the constraints given by z. First results for this approach have appeared in [15] and some extensions have been proposed in [47, 48].

Formally, the approach consists of computing the greatest u of the system given in Eq. 4.2 such that $y \preceq z$. Recalling that the mapping L_A is residuated (see Def. 2.35) the following equivalence holds

$$y = CA^*Bu \preceq z \Leftrightarrow u \preceq (CA^*B) \backslash z$$

and consequently, the greatest control u_{opt} achieving that $y_{opt} \preceq z$ is

$$u_{opt} = (CA^*B) \backslash z.$$

4.1.2 Optimal input filtering

In some cases, however, it is not possible or not reasonable to determine a desired output, but rather a reference model (which is given as a linear system in a dioid setting). The considered linear model of the system to be controlled shall then be as large as possible but not larger than the reference model G_{ref}. This can be achieved by adding an open loop pre-filter (in the literature often called pre-compensator) to the input of the system (4.2). The structure of the system with pre-filter is given in Fig. 4.1.

Formally this means for any input v and $u = Pv$

$$CA^*BPv \preceq G_{ref}v$$

which is equivalent to

$$CA^*BP \preceq G_{ref}.$$

Thanks to residuation theory the admissible pre-filters can be determined by

$$P \preceq (CA^*B) \backslash\!\!\diamond G_{ref},$$

and the greatest pre-filter is

$$P_{opt} = (CA^*B) \backslash\!\!\diamond G_{ref}. \tag{4.3}$$

Remark 4.1 (Causality of the optimal pre-filter). The optimal pre-filter determined by $P_{opt} = (CA^*B) \backslash\!\!\diamond G_{ref}$ may be non-causal. In this case the causal projection $Pr^{\top}_{caus}$ can be applied since the pre-filter matrix describes the transfer between v and u. Thus, the optimal causal pre-filter is

$$P^{\top}_{opt} = Pr^{\top}_{caus}(P_{opt}).$$

Remark 4.2 (Neutral pre-filter). Often a manufacturing system shall operate at its highest possible throughput while the internal stocks are minimized. This can be achieved by applying a so-called neutral pre-filter, which does not reduce the performance of the overall system. In such a case the reference model to be considered is $G_{ref} = CA^*B$, which represents the fastest behavior of the underlying system. Consequently, the optimal (neutral) pre-filter is

$$P_{opt} = (CA^*B) \backslash\!\!\diamond (CA^*B)$$

and the resulting system operates as fast as possible while all inputs occur as late as possible.

4.2 Feedback control

Up to now, all mentioned control approaches are feedforward strategies, i.e., changes in the system during runtime are not considered. However, as such unforeseen deviations may (more or less frequently) occur, a closed loop, i.e., feedback, control approach is necessary.

4.2.1 State feedback control with optimal pre-filter

One possible feedback control approach is state feedback control with an optimal pre-filter. In this approach all the states are assumed to be measured or estimated thanks to an observer (see e.g., [28]). For the first case, the resulting structure of the system to be controlled is given in Fig. 4.2. Then, the linear system (4.1) changes to

$$\begin{aligned} x &\geq Ax \oplus Bu \\ &= Ax \oplus B(Fx \oplus Pv) \\ &= (A \oplus BF)x \oplus BPv \\ y &= Cx \end{aligned}$$

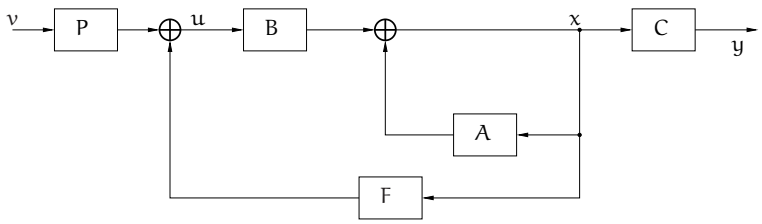


Figure 4.2: System structure with state feedback controller and pre-filter.

and the smallest solution is

$$x = (A \oplus BF)^* BPv$$
$$y = C(A \oplus BF)^* BPv.$$

The aim is to find an optimal (i.e., maximal) pre-filter P and an optimal feedback controller F, with respect to P, such that the controlled system is for any input v as large as possible but less or equal to a given reference model G_{ref} for the same input v. Formally,

$$C(A \oplus BF)^* BPv \preceq G_{ref} v \quad \forall v$$

which is equivalent to

$$C(A \oplus BF)^* BP \preceq G_{ref}. \tag{4.4}$$

According to the properties of the Kleene star

$$(a \oplus b)^* = (a^* b)^* a^*$$

and

$$(ab)^* a = a(ba)^*$$

the left hand side of Ineq. 4.4 can be written

$$C(A^* BF)^* A^* BP \preceq G_{ref} \tag{4.5}$$
$$CA^* B(FA^* B)^* P \preceq G_{ref}. \tag{4.6}$$

According to the definition of the Kleene star

$$(FA^* B)^* = I \oplus FA^* B \oplus (FA^* B)^2 \oplus \ldots$$

which implies that, independent of the controller F, the pre-filter has to meet the following constraint

$$CA^* BP \preceq G_{ref},$$

which is equivalent to

$$P \preceq (CA^*B)\backslash\!\!\diamond G_{ref} = P_{opt}. \tag{4.7}$$

Clearly, the resulting optimal pre-filter P_{opt} is equivalent to the pre-filter given in Eq. 4.3. Considering this optimal pre-filter in Eq. 4.6, we get

$$CA^*B(FA^*B)^*P_{opt} \preceq G_{ref}$$

which is according to residuation theory equivalent to

$$(FA^*B)^*P_{opt} \preceq (CA^*B)\backslash\!\!\diamond G_{ref} = P_{opt}$$

and

$$(FA^*B)^* \preceq (CA^*B)\backslash\!\!\diamond G_{ref} \not\diamond P_{opt} = P_{opt} \not\diamond P_{opt}. \tag{4.8}$$

According to Eq. 2.16 the right hand side of this inequality can be written

$$(CA^*B)\backslash\!\!\diamond G_{ref} \not\diamond P_{opt} = P_{opt} \not\diamond P_{opt} = (P_{opt} \not\diamond P_{opt})^* = ((CA^*B)\backslash\!\!\diamond G_{ref} \not\diamond P_{opt})^*.$$

Then, (4.8) can be rewritten (see Theorem 2.51)

$$FA^*B \preceq ((CA^*B)\backslash\!\!\diamond G_{ref} \not\diamond P_{opt})^* = (CA^*B)\backslash\!\!\diamond G_{ref} \not\diamond P_{opt}.$$

Thus, using residuation theory we get

$$F \preceq (CA^*B)\backslash\!\!\diamond G_{ref} \not\diamond P_{opt} \not\diamond (A^*B)$$

which is equivalent to

$$F \preceq (CA^*B)\backslash\!\!\diamond G_{ref} \not\diamond (A^*BP_{opt})$$

and the optimal (greatest) controller F_{opt} achieves equality, i.e.,

$$F_{opt} = (CA^*B)\backslash\!\!\diamond G_{ref} \not\diamond (A^*BP_{opt}).$$

However, in order to be realizable, the optimal pre-filter and the optimal feedback controller need to be causal. The optimal causal pre-filter and feedback controller can be obtained by applying the causal projection $Pr^{\mp}_{caus}$:

$$P^{\mp}_{opt} = Pr^{\mp}_{caus}(P_{opt}) \tag{4.9}$$

and

$$\begin{aligned} F^{\mp}_{opt} &= Pr^{\mp}_{caus}\left((CA^*B)\backslash\!\!\diamond G_{ref} \not\diamond (A^*BP_{opt})\right) \\ &= Pr^{\mp}_{caus}(F_{opt}). \end{aligned} \tag{4.10}$$

Example 4.3 (State feedback of a simple manufacturing system). Reconsidering the simple manufacturing system of Ex. 3.12. Its TEG representation is given in Fig. 3.8. The corresponding linear system in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is

$$\begin{aligned} x &= Ax \oplus Bu \\ &= \begin{pmatrix} \varepsilon & \gamma\delta & \varepsilon & \varepsilon \\ \delta & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \gamma^2 \\ \varepsilon & \varepsilon & \delta^4 & \varepsilon \end{pmatrix} x \oplus \begin{pmatrix} \delta^2 & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \gamma\delta^3 \\ \varepsilon & \varepsilon \end{pmatrix} u \\ y &= Cx \\ &= \begin{pmatrix} \varepsilon & \gamma\delta^2 & \varepsilon & \delta \end{pmatrix} x. \end{aligned}$$

The aim is to find the optimal pre-filter and state feedback controller such that the system operates at its highest possible throughput while minimizing the internal stock. Thus, the reference model is set to the fastest possible model, which is the uncontrolled model, i.e.,

$$\begin{aligned} G_{ref} &= CA^*B \\ &= \begin{pmatrix} \gamma\delta^5 (\gamma\delta^2)^* & \gamma\delta^8 (\gamma^2\delta^4)^* \end{pmatrix}. \end{aligned}$$

According to Eq. 4.7 the optimal pre-filter can be determined by

$$\begin{aligned} P_{opt} &= (CA^*B) \backslash\!\!\backslash G_{ref} \\ &= \begin{pmatrix} (\gamma\delta^2)^* & \delta (\gamma\delta^2)^* \\ \delta^{-3} (\gamma\delta^2)^* & (\gamma^2\delta^4)^* \end{pmatrix}. \end{aligned}$$

Then the state feedback controller F_{opt} is

$$\begin{aligned} F_{opt} &= (CA^*B) \backslash\!\!\backslash G_{ref} /\!\!\phi (A^*BP_{opt}) \\ &= \begin{pmatrix} \delta^{-2}(\gamma\delta^2)^* & \delta^{-3}(\gamma\delta^2)^* & \gamma^{-1}\delta^{-2} (\gamma\delta^2)^* & \gamma^{-1}\delta^{-6} (\gamma\delta^2)^* \\ \delta^{-5} (\gamma\delta^2)^* & \delta^{-6} (\gamma\delta^2)^* & \gamma^{-1}\delta^{-3} (\gamma^2\delta^4)^* & \gamma^{-1}\delta^{-7} (\gamma^2\delta^4)^* \end{pmatrix}. \end{aligned}$$

Obviously, the obtained pre-filter P_{opt} as well as the state feedback controller F_{opt} are non-causal transfer matrices. Applying the causal projection $Pr_{caus}^{\mp}$ the greatest causal pre-filter and feedback controller can be determined

$$\begin{aligned} P_{opt}^{\mp} &= Pr_{caus}^{\mp}(P_{opt}) \\ &= \begin{pmatrix} (\gamma\delta^2)^* & \delta (\gamma\delta^2)^* \\ \gamma^2\delta^1 (\gamma\delta^2)^* & (\gamma^2\delta^4)^* \end{pmatrix} \end{aligned}$$

and

$$\begin{aligned} F^{\mp}_{opt} &= Pr^{\mp}_{caus}(F_{opt}) \\ &= \begin{pmatrix} \gamma\left(\gamma\delta^2\right)^* & \gamma^2\delta\left(\gamma\delta^2\right)^* & \left(\gamma\delta^2\right)^* & \gamma^2\left(\gamma\delta^2\right)^* \\ \gamma^3\delta\left(\gamma\delta^2\right)^* & \gamma^3\left(\gamma\delta^2\right)^* & \gamma\delta\left(\gamma^2\delta^4\right)^* & \gamma^3\delta\left(\gamma^2\delta^4\right)^* \end{pmatrix}. \end{aligned}$$

Given an input, e.g.,

$$\nu = \begin{pmatrix} \left(\gamma\delta^3\right)^* \\ e \end{pmatrix}$$

the uncontrolled (meaning $F = \mathcal{E}, P = I$) and controlled ($F = F^{\mp}_{opt}, P = P^{\mp}_{opt}$) state x_u and x_c as well as the uncontrolled and controlled output y_u and y_c can be compared

$$x_u = A^*B\nu = \begin{pmatrix} \delta^2\left(\gamma\delta^3\right)^* \\ \delta^3\left(\gamma\delta^3\right)^* \\ \gamma\delta^3\left(\gamma^2\delta^4\right)^* \\ \gamma\delta^7\left(\gamma^2\delta^4\right)^* \end{pmatrix}$$

$$x_c = (A \oplus BF^{\mp}_{opt})^*BP^{\mp}_{opt}\nu = \begin{pmatrix} \delta^3 \oplus \left(\gamma\delta^5\right)\left(\gamma\delta^3\right)^* \\ \delta^4 \oplus \left(\gamma\delta^6\right)\left(\gamma\delta^3\right)^* \\ \gamma\delta^3 \oplus \gamma^3\delta^7 \oplus \gamma^5\delta^{11} \oplus \left(\gamma^6\delta^{13}\right)\left(\gamma\delta^3\right)^* \\ \gamma\delta^7 \oplus \gamma^3\delta^{11} \oplus \gamma^5\delta^{15} \oplus \left(\gamma^6\delta^{17}\right)\left(\gamma\delta^3\right)^* \end{pmatrix}$$

$$y_u = Cx_u = \gamma\delta^8 \oplus \gamma^3\delta^{12} \oplus \left(\gamma^4\delta^{14}\right)\left(\gamma\delta^3\right)^*$$

$$y_c = Cx_c = \gamma\delta^8 \oplus \gamma^3\delta^{12} \oplus \left(\gamma^4\delta^{14}\right)\left(\gamma\delta^3\right)^*$$

Clearly, the output of the controlled system is equivalent to the output of the uncontrolled system. Thus, the performance is not reduced due to the controller. What has changed, however, is the evolution of the internal state. In the uncontrolled case the slow input ν_1 (one occurrence every three time units) has an effect on the dynamics of x_1 and x_2, while x_3 and x_4 occur at a faster rate (two occurrences every four time units). In terms of TEG this means that x_3 and x_4 fire more frequently than x_1 and x_2 and consequently, there will be an accumulation of tokens in the system in the place between the transitions x_4 and y, which represents an increasing internal stock. In the controlled case, however, the information of the slow input is also passed to x_3 and x_4, which after a short transient phase have the same firing rate as x_1 and x_2. Thus, there will be no accumulation of tokens. As, by definition, the controlled input $u = P^{\mp}_{opt}\nu \oplus F^{\mp}_{opt}x$ is the greatest input such that the output is not reduced; this implies that the internal stock is minimized.

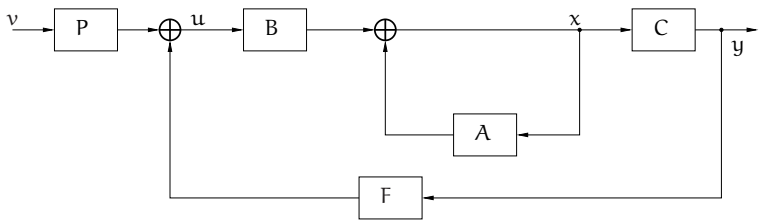


Figure 4.3: System structure with output feedback controller and pre-filter.

4.2.2 Output feedback control with optimal pre-filter

In some cases, e.g., when the internal state of the system is unknown, an output feedback control strategy with an optimal pre-filter may be considered. The corresponding system structure is given in Fig. 4.3.

The linear system describing the output controlled behavior is

$$\begin{aligned} x &\geq Ax \oplus Bu \\ &= Ax \oplus B\,(Fy \oplus Pv) \\ y &= Cx. \end{aligned}$$

Replacing y in the equation by Cx, one can write

$$\begin{aligned} x &\geq Ax \oplus B\,(FCx \oplus Pv) \\ &= (A \oplus BFC)\,x \oplus BPv, \end{aligned}$$

which has least solution

$$x = (A \oplus BFC)^* BPv,$$

and corresponding output

$$y = C\,(A \oplus BFC)^* BPv.$$

As in state feedback control, the aim is to find for any input v an optimal pre-filter P and an optimal output feedback controller F such that the controlled system is as large as possible but smaller or equal to a given reference model G_{ref}. Formally, we want to determine the maximal F and P such that

$$C\,(A \oplus BFC)^* BPv \preceq G_{ref} v \quad \forall v,$$

which is equivalent to

$$C\,(A \oplus BFC)^* BP \preceq G_{ref}.$$

Similar to the procedure for a state feedback controller with optimal pre-filter, using properties of the Kleene star, this equation can be rewritten as

$$C\,(A^*BFC)^*\,A^*BP \preceq G_{ref}$$
$$CA^*B\,(FCA^*B)^*\,P \preceq G_{ref}.$$

By definition

$$(FCA^*B)^* = I \oplus FCA^*B \oplus (FCA^*B)^2 \oplus \dots$$

which implies that, independent of the controller F, the pre-filter has to be chosen such that

$$CA^*BP \preceq G_{ref}.$$

Consequently, the optimal pre-filter, independent of the feedback controller F, is

$$P_{opt} = (CA^*B) \backslash\!\!\diamond G_{ref}.$$

Furthermore, similar to the determination of a state feedback controller, it can be shown that the optimal output feedback controller can be computed by

$$F_{opt} = (CA^*B) \backslash\!\!\diamond G_{ref} \not\diamond (CA^*BP_{opt}). \tag{4.11}$$

Example 4.4 (Output feedback of a simple manufacturing system). Reconsidering, once again, the simple manufacturing system of Ex. 4.3 with $G_{ref} = CA^*B$. The optimal (causal) pre-filter $P^{\mp}_{opt}$ is equivalent to the pre-filter determined for system with state feedback control. The optimal output controller is

$$\begin{aligned} F_{opt} &= (CA^*B) \backslash\!\!\diamond G_{ref} \not\diamond (CA^*BP_{opt}) \\ &= \begin{pmatrix} \gamma^{-1}\delta^{-7}\left(\gamma\delta^2\right)^* \\ \gamma^{-1}\delta^{-8}\left(\gamma^2\delta^4\right)^* \end{pmatrix} \end{aligned}$$

and the optimal causal output controller is

$$\begin{aligned} F^{\mp}_{opt} &= \mathrm{Pr}^{\mp}_{caus}(F_{opt}) \\ &= \begin{pmatrix} \gamma^3\delta^1\left(\gamma\delta^2\right)^* \\ \gamma^3\left(\gamma^2\delta^4\right)^* \end{pmatrix}. \end{aligned}$$

For the given input $\nu = \begin{pmatrix} \left(\gamma\delta^3\right)^* & e \end{pmatrix}^{\mathsf{T}}$ the state evolution with output feedback is equivalent to the state evolution with state feedback.

As a matter of fact, for a given input ν and the reference model $G_{ref} = CA^*B$, the system with optimal feedback control – both state and output feedback – will behave equivalently to the system solely equipped with the open loop optimal pre-filter. However, unforeseen disturbances, which have not yet been considered, can only be taken into account if feedback control strategies are applied. Such unforeseen disturbances can be modeled as uncontrollable inputs [40].

4.3 Optimal control with disturbances

4.3.1 State feedback control with optimal pre-filter and disturbances

The considered disturbances act on the state of the system, such that they can delay the occurrence of certain events. The structure of such a disturbed system with state feedback controller and pre-filter is given in Fig. 4.4. The corresponding linear system in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ is

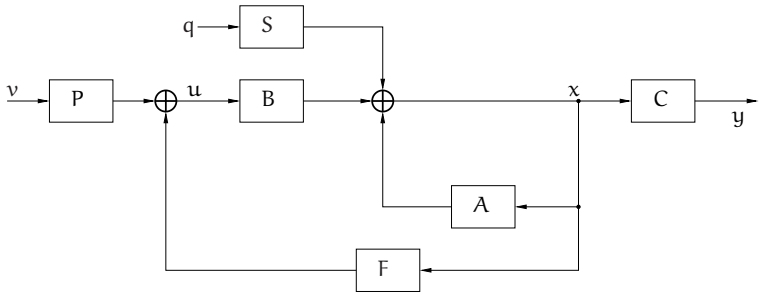


Figure 4.4: System structure with state feedback, pre-filter and disturbances.

$$\begin{aligned} x &\geq Ax \oplus Sq \oplus Bu \\ &= Ax \oplus Sq \oplus B(Fx \oplus Pv) \\ &= (A \oplus BF)x \oplus BPv \oplus Sq \\ y &= Cx \end{aligned}$$

where q is a disturbance signal and S is the disturbance matrix. This system can be rewritten

$$x \geq (A \oplus BF)x \oplus \underbrace{\begin{pmatrix} B & S \end{pmatrix}}_{\tilde{B}} \underbrace{\begin{pmatrix} P & \varepsilon \\ \varepsilon & I \end{pmatrix}}_{\tilde{P}} \underbrace{\begin{pmatrix} v \\ q \end{pmatrix}}_{\tilde{v}}$$

with least solution

$$x = (A \oplus BF)^* \tilde{B}\tilde{P}\tilde{v}$$

and output

$$y = C(A \oplus BF)^* \tilde{B}\tilde{P}\tilde{v}.$$

Assuming that a reference model G_{ref} is given and that the controlled system shall be as large as possible but not larger than the reference model, the following problem has to be solved: Find maximal F and $\tilde{P}$ such that

$$C(A \oplus BF)^* \tilde{B}\tilde{P} \preceq \tilde{G}_{ref}$$

with

$$\tilde{G}_{ref} = \begin{pmatrix} G_{ref} & CA^*S \end{pmatrix}.$$

The structure of $\tilde{G}_{ref}$ is due to the uncontrollability of q. Independent of the feedback controller F the pre-filter $\tilde{P}$ has to meet the following constraint

$$CA^*\tilde{B}\tilde{P} \preceq \tilde{G}_{ref}$$

and, thus, for admissible pre-filters

$$\tilde{P} \preceq (CA^*\tilde{B}) \backslash\!\!\backslash \tilde{G}_{ref}$$

has to hold. Additionally, it has to be ensured that the pre-filter does not modify q. Formally, the optimal pre-filter can be determined by

$$\tilde{P}_{opt} = \tilde{P} \wedge \begin{pmatrix} (CA^*B) \backslash\!\!\backslash G_{ref} & \varepsilon \\ \varepsilon & I \end{pmatrix}$$

with

$$\tilde{P} = (CA^*\tilde{B}) \backslash\!\!\backslash \tilde{G}_{ref}.$$

Taking a closer look at the equation for the optimal pre-filter and using a generalization of the left residual of block matrices, one obtains

$$\begin{aligned} \tilde{P}_{opt} &= (CA^*\tilde{B}) \backslash\!\!\backslash \tilde{G}_{ref} \wedge \begin{pmatrix} (CA^*B) \backslash\!\!\backslash G_{ref} & \varepsilon \\ \varepsilon & I \end{pmatrix} \\ &= \begin{pmatrix} CA^*B & CA^*S \end{pmatrix} \backslash\!\!\backslash \begin{pmatrix} G_{ref} & CA^*S \end{pmatrix} \wedge \begin{pmatrix} (CA^*B) \backslash\!\!\backslash G_{ref} & \varepsilon \\ \varepsilon & I \end{pmatrix} \\ &= \begin{pmatrix} (CA^*B) \backslash\!\!\backslash G_{ref} & (CA^*B) \backslash\!\!\backslash (CA^*S) \\ (CA^*S) \backslash\!\!\backslash G_{ref} & (CA^*S) \backslash\!\!\backslash (CA^*S) \end{pmatrix} \wedge \begin{pmatrix} (CA^*B) \backslash\!\!\backslash G_{ref} & \varepsilon \\ \varepsilon & I \end{pmatrix} \end{aligned}$$

and with ε being the greatest lower bound of a dioid and with $a \backslash\!\!\backslash a \succeq e$ the optimal pre-filter results in

$$\tilde{P}_{opt} = \begin{pmatrix} (CA^*B) \backslash\!\!\backslash G_{ref} & \varepsilon \\ \varepsilon & I \end{pmatrix}.$$

Using this optimal pre-filter, the greatest controller F can be determined by

$$\begin{aligned} &C(A \oplus BF)^*\tilde{B}\tilde{P}_{opt} \preceq \tilde{G}_{ref}, \text{ i.e.,} \\ &C(A^*BF)^*A^*\tilde{B}\tilde{P}_{opt} \preceq \tilde{G}_{ref} \end{aligned}$$

with

$$B = \tilde{B} \begin{pmatrix} e \\ \varepsilon \end{pmatrix}$$

this is equivalent to

$$C \left(A^* \tilde{B} \begin{pmatrix} e \\ \varepsilon \end{pmatrix} F \right)^* A^* \tilde{B} \tilde{P}_{opt} \preceq \tilde{G}_{ref},$$

or using the equality $(ab)^* a = a(ba)^*$,

$$\begin{aligned} CA^* \tilde{B} \left(\begin{pmatrix} e \\ \varepsilon \end{pmatrix} FA^* \tilde{B} \right)^* \tilde{P}_{opt} &\preceq \tilde{G}_{ref} \\ \left(\begin{pmatrix} e \\ \varepsilon \end{pmatrix} FA^* \tilde{B} \right)^* &\preceq \left(CA^* \tilde{B} \right) \backslash \tilde{G}_{ref} / \tilde{P}_{opt}. \end{aligned} \tag{4.12}$$

Assuming that the right hand side of the last inequality is a Kleene star, i.e., $(CA^* \tilde{B}) \backslash \tilde{G}_{ref} / \tilde{P}_{opt} = \left((CA^* \tilde{B}) \backslash \tilde{G}_{ref} / \tilde{P}_{opt} \right)^*$, one can write

$$\begin{pmatrix} e \\ \varepsilon \end{pmatrix} FA^* \tilde{B} \preceq \left(CA^* \tilde{B} \right) \backslash \tilde{G}_{ref} / \tilde{P}_{opt}$$

and, consequently, the following inequality has to hold for a controller F

$$\begin{aligned} F &\preceq \begin{pmatrix} e \\ \varepsilon \end{pmatrix} \backslash \left(CA^* \tilde{B} \right) \backslash \tilde{G}_{ref} / \tilde{P}_{opt} / \left(A^* \tilde{B} \right) \\ &\preceq \left(CA^* \tilde{B} \begin{pmatrix} e \\ \varepsilon \end{pmatrix} \right) \backslash \tilde{G}_{ref} / \left(A^* \tilde{B} \tilde{P}_{opt} \right) \\ &\preceq (CA^* B) \backslash \tilde{G}_{ref} / \left(A^* \tilde{B} \tilde{P}_{opt} \right). \end{aligned}$$

The greatest controller F_{opt} achieves equality

$$F_{opt} = (CA^* B) \backslash \tilde{G}_{ref} / \left(A^* \tilde{B} \tilde{P}_{opt} \right). \tag{4.13}$$

One open issue at this point is the question whether the right hand side of (4.12) is a Kleene star. In the following theorem we show that for a specific reference model G_{ref} the term $(CA^* \tilde{B}) \backslash \tilde{G}_{ref} / \tilde{P}_{opt}$ is always a star.

Theorem 4.5. For a linear system in a complete dioid defined by

$$\begin{aligned} x &= (A \oplus BF)x \oplus \underbrace{\begin{pmatrix} B & S \end{pmatrix}}_{\tilde{B}} \underbrace{\begin{pmatrix} P & \varepsilon \\ \varepsilon & I \end{pmatrix}}_{\tilde{P}} \underbrace{\begin{pmatrix} v \\ q \end{pmatrix}}_{\tilde{v}} \\ &= (A \oplus BF)^* \tilde{B} \tilde{P} \tilde{v} \\ y &= C(A \oplus BF)^* \tilde{B} \tilde{P} \tilde{v} \end{aligned}$$

and a given reference model

$$G_{ref} = CA^*B$$

and

$$\tilde{G}_{ref} = \begin{pmatrix} G_{ref} & CA^*S \end{pmatrix} = \begin{pmatrix} CA^*B & CA^*S \end{pmatrix}$$

such that

$$C(A \oplus BF)^* \tilde{B} \tilde{P}_{opt} \preceq \tilde{G}_{ref}$$

with

$$\tilde{P}_{opt} = \begin{pmatrix} (CA^*B) \backslash\!\!\backslash G_{ref} & \varepsilon \\ \varepsilon & I \end{pmatrix} = \begin{pmatrix} (CA^*B) \backslash\!\!\backslash (CA^*B) & \varepsilon \\ \varepsilon & I \end{pmatrix} \tag{4.14}$$

the following equivalence holds

$$(CA^*\tilde{B}) \backslash\!\!\backslash \tilde{G}_{ref} /\!\!/ \tilde{P}_{opt} = \left((CA^*\tilde{B}) \backslash\!\!\backslash \tilde{G}_{ref} /\!\!/ \tilde{P}_{opt} \right)^*.$$

Proof. Taking a closer look at the optimal pre-filter given in (4.14), it can be observed that $\tilde{P}_{opt}$ is a Kleene star, i.e., $\tilde{P}_{opt} = \tilde{P}_{opt}^*$. Furthermore, the term $(CA^*\tilde{B}) \backslash\!\!\backslash \tilde{G}_{ref}$ can be written as

$$(CA^*\tilde{B}) \backslash\!\!\backslash \tilde{G}_{ref} = \begin{pmatrix} CA^*B & CA^*S \end{pmatrix} \backslash\!\!\backslash \begin{pmatrix} G_{ref} & CA^*S \end{pmatrix}$$

and with $G_{ref} = CA^*B$ this is equivalent to

$$(CA^*\tilde{B}) \backslash\!\!\backslash \tilde{G}_{ref} = \begin{pmatrix} CA^*B & CA^*S \end{pmatrix} \backslash\!\!\backslash \begin{pmatrix} CA^*B & CA^*S \end{pmatrix},$$

which is obviously a Kleene star as well. Consequently one can write

$$\underbrace{(CA^*\tilde{B}) \backslash\!\!\backslash \tilde{G}_{ref}}_{\Phi} /\!\!/ \underbrace{\tilde{P}_{opt}}_{\Psi} = \Phi^* /\!\!/ \Psi^*.$$

Moreover, comparing Φ^*

$$\begin{aligned}\Phi^* = \Phi &= \begin{pmatrix} CA^*B & CA^*S \end{pmatrix} \backslash \begin{pmatrix} CA^*B & CA^*S \end{pmatrix} \\ &= \begin{pmatrix} (CA^*B)\backslash(CA^*B) & (CA^*B)\backslash(CA^*S) \\ (CA^*S)\backslash(CA^*B) & (CA^*S)\backslash(CA^*S) \end{pmatrix}\end{aligned}$$

with Ψ^*

$$\Psi^* = \Psi = \begin{pmatrix} (CA^*B)\backslash(CA^*B) & \varepsilon \\ \varepsilon & I \end{pmatrix}$$

it is obvious that

$$\Phi^* \geq \Psi^*,$$

since $\Phi_{22} = (CA^*S)\backslash(CA^*S)$ is a star, i.e., $(CA^*S)\backslash(CA^*S) = ((CA^*S)\backslash(CA^*S))^*$ and by definition of the Kleene star $((CA^*S)\backslash(CA^*S))^* \geq I$. Then, according to Lem. 2.53 one can write

$$\Phi^* \geq \Psi^* \Leftrightarrow \Phi^* \not\phi \Psi^* = \Phi^*,$$

and, therefore,

$$\begin{aligned}(CA^*\tilde{B})\backslash\tilde{G}_{ref} \not\phi \tilde{P}_{opt} &= \Phi^* \not\phi \Psi^* = \Phi^* \\ &= (CA^*\tilde{B})\backslash\tilde{G}_{ref} \\ &= \begin{pmatrix} CA^*B & CA^*S \end{pmatrix} \backslash \begin{pmatrix} CA^*B & CA^*S \end{pmatrix}\end{aligned}$$

which obviously is a Kleene star. □

4.3.2 Output feedback control with optimal pre-filter and disturbances

Of course, also output feedback control strategies can be applied when disturbances are explicitly modeled. The structure of such a disturbed system with output feedback controller and pre-filter is given in Fig. 4.5. The corresponding linear system describing the behavior of the controlled system is

$$\begin{aligned}x &\geq Ax \oplus Sq \oplus Bu \\ &= Ax \oplus Sq \oplus B(FCx \oplus Pv) \\ &= (A \oplus BFC)x \oplus BPv \oplus Sq \\ y &= Cx\end{aligned}$$

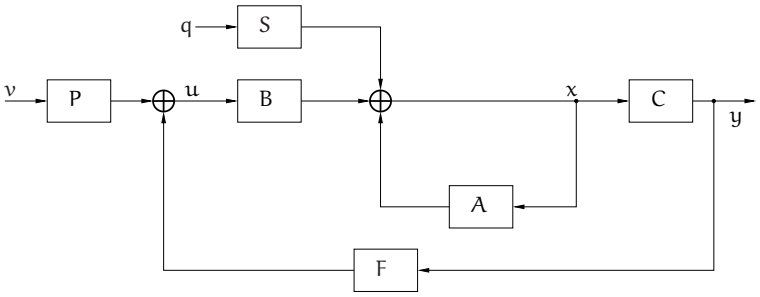


Figure 4.5: System structure with output feedback, pre-filter and disturbances.

Similar to the state feedback control approach we can write

$$
\begin{aligned}
x &\geq (A \oplus BFC)x \oplus \underbrace{\begin{pmatrix} B & S \end{pmatrix}}_{\tilde{B}} \underbrace{\begin{pmatrix} P & \varepsilon \\ \varepsilon & I \end{pmatrix}}_{\tilde{P}} \underbrace{\begin{pmatrix} v \\ q \end{pmatrix}}_{\tilde{v}} \\
&= (A \oplus BFC)^* \tilde{B}\tilde{P}\tilde{v} \\
y &= C(A \oplus BFC)^* \tilde{B}\tilde{P}\tilde{v}.
\end{aligned}
$$

The aim is to find controllers F and P such that the controlled system is as large as possible but not larger than a given reference model G_{ref}. Formally, we want to find the largest F and P, such that,

$$
\begin{aligned}
&C(A \oplus BFC)^* \tilde{B}\tilde{P} \preceq \tilde{G}_{ref}, \text{ i.e.,} \\
&C(A^*BFC)^* A^* \tilde{B}\tilde{P} \preceq \tilde{G}_{ref}
\end{aligned}
$$

with

$$
\tilde{G}_{ref} = \begin{pmatrix} G_{ref} & CA^*S \end{pmatrix}.
$$

Clearly, independent of the choice of the state feedback controller F, the pre-filter $\tilde{P}$ has to fulfill the following constraint

$$
CA^*\tilde{B}\tilde{P} \preceq \tilde{G}_{ref}
$$

and applying residuation theory one obtains

$$
\tilde{P} \preceq (CA^*\tilde{B}) \backslash\!\!\circ \tilde{G}_{ref}
$$

which is equivalent to the pre-filter in the state feedback control approach. The optimal pre-filter can be determined by

$$\begin{aligned}\tilde{P}_{opt} &= (CA^*\tilde{B})\backslash\!\!\backslash\tilde{G}_{ref} \wedge \begin{pmatrix} (CA^*B)\backslash\!\!\backslash G_{ref} & \varepsilon \\ \varepsilon & I \end{pmatrix} \\ &= \begin{pmatrix} (CA^*B)\backslash\!\!\backslash G_{ref} & \varepsilon \\ \varepsilon & I \end{pmatrix}.\end{aligned}$$

Very similar to the determination of the state feedback controller the optimal output feedback controller with the determined optimal pre-filter can be obtained

$$\begin{aligned} C(A^*BFC)^*A^*\tilde{B}\tilde{P}_{opt} &\preceq \tilde{G}_{ref} \\ C\left(A^*\tilde{B}\begin{pmatrix} e \\ \varepsilon \end{pmatrix}FC\right)^* A^*\tilde{B}\tilde{P}_{opt} &\preceq \tilde{G}_{ref} \\ CA^*\tilde{B}\left(\begin{pmatrix} e \\ \varepsilon \end{pmatrix}FCA^*\tilde{B}\right)^* \tilde{P}_{opt} &\preceq \tilde{G}_{ref} \\ \left(\begin{pmatrix} e \\ \varepsilon \end{pmatrix}FCA^*\tilde{B}\right)^* &\preceq (CA^*\tilde{B})\backslash\!\!\backslash\tilde{G}_{ref}/\!\!/\tilde{P}_{opt}. \end{aligned}$$

If the right hand side of the last inequality is a Kleene star, which is, for example, the case if $G_{ref} = CA^*B$ (see Thm. 4.5 for details), we can write

$$\begin{pmatrix} e \\ \varepsilon \end{pmatrix}FCA^*\tilde{B} \preceq (CA^*\tilde{B})\backslash\!\!\backslash\tilde{G}_{ref}/\!\!/\tilde{P}_{opt}$$

and applying residuation theory results in

$$\begin{aligned} F &\preceq \begin{pmatrix} e \\ \varepsilon \end{pmatrix}\backslash\!\!\backslash(CA^*\tilde{B})\backslash\!\!\backslash\tilde{G}_{ref}/\!\!/\tilde{P}_{opt}/\!\!/(CA^*\tilde{B}) \\ &\preceq (CA^*B)\backslash\!\!\backslash\tilde{G}_{ref}/\!\!/(CA^*\tilde{B}\tilde{P}_{opt}). \end{aligned}$$

Consequently, the optimal output feedback controller is

$$F_{opt} = (CA^*B)\backslash\!\!\backslash\tilde{G}_{ref}/\!\!/(CA^*\tilde{B}\tilde{P}_{opt}).$$

5
Control of extended TEG with Constraints

As shown in the previous chapters, timed event graphs and their linear representation in dioids are suitable tools for the modeling and control of discrete event systems subject to delay and synchronization phenomena. However, despite the vast variety of systems that can be modeled by (standard) TEG, some specific features of many real systems cannot be included. For example, the processing of a part on a resource may have to be performed within a fixed time interval, also called time window. Thus, with respect to TEG, there exists not only a minimal time a token has to spend in a place but also an upper bound for the time, by which the token has to be removed from the place by its output transition. Similarly, while it is easily possible to model the maximal number of tokens within a certain part of a TEG, e.g., the capacity of a resource determines the maximal number of parts being processed in this resource simultaneously, it is not possible to model the minimal number of tokens that have to be present in a certain part of a TEG. This means that a transition fires a certain number of times more often than another transition.

While the issue of timed event graphs with time window constraints has been handled in several publications, e.g., [37, 38, 53], the latter issue concerning the minimal number of tokens in a place is closely related to the notion of negative numbers of tokens in extended TEG as introduced in Section 3.3, and has, to our knowledge, not yet been addressed. In the following, one possible way of including time window constraints as well as constraints on the minimal number of tokens in TEG and their corresponding linear representation in dioids will be discussed. The presented approach is based on the work by Iteb Ouerghi [53] and has partially been published in [7, 8].

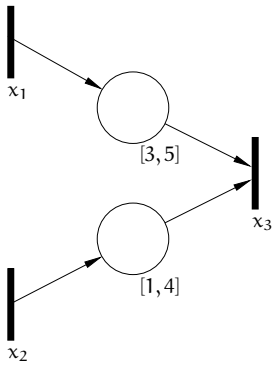


Figure 5.1: Part of a timed event graph with time window constraints.

Example 5.1 (Manufacturing system with time window constraints). Considering a (part of a) manufacturing system which combines two parts A and B to a final product C. Before the production of C can start, parts A and B have to spend a minimum of 3 (resp. 1) time units but not longer than 5 (resp. 4) time units. The corresponding (part of the) TEG is given in Fig. 5.1. In this TEG transition x_1 (resp. x_2) represents the provision of part A (resp. part B) and x_3 models the start of combining A and B to produce C. Between the firings of x_1 and x_3 there is a time window with a lower bound of 3 time units and an upper bound of 5 time units. Additionally, there is a time window between the firings of x_2 and x_3 with a lower and upper bound of 1, respectively 4, time units. The lower bounds of these time windows represent the holding times as defined for standard TEG. Consequently, these dependencies can easily be modeled in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$. For our example we get

$$\begin{aligned} x_3 &\geq \delta^3 x_1 \\ x_3 &\geq \delta x_2 \end{aligned}$$

which is equivalent to

$$x_3 \geq \delta^3 x_1 \oplus \delta x_2. \tag{5.1}$$

The upper bounds of the time windows can be modeled in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ in a similar manner. Basically, x_3 has to fire at the latest 5 time units after the firing of x_1 and no later than 4 time units after x_2 has fired. In $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ this can be written

$$\begin{aligned} x_3 &\leq \delta^5 x_1 \\ x_3 &\leq \delta^4 x_2 \end{aligned} \tag{5.2}$$

which is equivalent to

$$x_3 \leq \delta^5 x_1 \wedge \delta^4 x_2. \tag{5.3}$$

Equations (5.1) and (5.3) can be written in matrix-vector form with $x = (x_1\ x_2\ x_3)^T$

$$x \geq \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \delta^3 & \delta & \varepsilon \end{pmatrix} \otimes x \tag{5.4}$$

$$x \leq \begin{pmatrix} \top & \top & \top \\ \top & \top & \top \\ \delta^5 & \delta^4 & \top \end{pmatrix} \odot x. \tag{5.5}$$

It is important to note that the dependencies of the upper bounds of the time window have to be written with the dual multiplication and that the zero element of this operation is $\top$.

Thus, there are two different kinds of constraints and without further ado it is not possible to merge them to obtain a linear system representation in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ (or any other dioid).

Remark 5.2. Of course the constraints (5.2) could also be handled in the form

$$x_1 \geq \delta^{-5} x_3$$
$$x_2 \geq \delta^{-4} x_3.$$

This, however, would result in non-causal system descriptions with respect to $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ and, therefore, such constraints are handled as described in the previous example.

Example 5.3 (Manufacturing systems with minimal and maximal number of parts in process). Given a (part of a) manufacturing system, in which a process p_1 provides raw parts for two subsequent processes p_2 and p_3. For structural reasons, there are, however, constraints on the minimal and maximal parts in process. More precisely, after some initial period (e.g., after a start-up process) there should be at least 1 raw part but due to capacity reasons not more than 2 parts available for p_2. Similarly, at least 2 but at most 4 raw parts should be available for p_3 once the system has been started-up. The corresponding TEG is given in Fig. 5.2. Transition x_1 in this figure represents the provision of raw parts by process p_1, transitions x_2 and x_3 represent the start of processes p_2 and p_3, respectively. The minimal and maximal numbers of allowed tokens in the place between x_1 and x_2 (resp. x_3) are indicated by the tokens below the places. In terms of TEG, this means that at any time (after some start-up period) x_1 has to fire at least once more often than x_2 and at least twice more often than x_3, but at the same time x_1 shall fire at most twice more often than x_2 and at most four times more often than x_3. With respect to $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ this can be written

$$x_1 \geq \gamma x_2$$
$$x_1 \geq \gamma^2 x_3,$$

which is equivalent to

$$x_1 \geq \gamma x_2 \oplus \gamma^2 x_3 \tag{5.6}$$

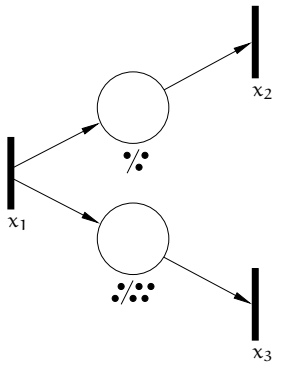


Figure 5.2: Part of a timed event graph with minimal and maximal numbers of tokens.

and

$$\begin{aligned} x_1 &\preceq \gamma^2 x_2 \\ x_1 &\preceq \gamma^4 x_3, \end{aligned} \tag{5.7}$$

which can be written

$$x_1 \preceq \gamma^2 x_2 \wedge \gamma^4 x_3. \tag{5.8}$$

Just as for Ex. 5.1 the dependencies depicted in equations (5.6) and (5.8) can be written in matrix vector form with $x = (x_1\ x_2\ x_3)^T$

$$x \geq \begin{pmatrix} \varepsilon & \gamma & \gamma^2 \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{pmatrix} \otimes x \tag{5.9}$$

$$x \preceq \begin{pmatrix} \top & \gamma^2 & \gamma^4 \\ \top & \top & \top \\ \top & \top & \top \end{pmatrix} \odot x. \tag{5.10}$$

Clearly, equations (5.9) and (5.10) are very similar to equations (5.4) and (5.5) of Ex. 5.1.

Remark 5.4. Similar to the upper bounds of time windows the constraints on minimal number of parts (given in (5.7)) could also be handled by

$$\begin{aligned} x_2 &\succeq \gamma^{-2} x_1 \\ x_3 &\succeq \gamma^{-4} x_1, \end{aligned}$$

which, according to Def. 3.30, results in a causal system description. However, since upper bounds of time windows are handled in the form (5.5), constraints on the minimal number tokens can be modeled either by a negative number of tokens or in form of (5.10).

Looking at the last two examples the question on how to find a linear system in $\mathcal{M}_{\text{in}}^{\text{ax}}[\![\gamma, \delta]\!]$, which includes lower and upper bounds on time or number of tokens, arises.

5.1 Modeling extended TEG with constraints

Formally, the internal behavior of the considered systems with constraints is determined by two types of constraints, i.e.,

$$x \geq \underline{A} \otimes x \tag{5.11}$$

$$x \preceq \overline{A} \odot x \tag{5.12}$$

where $\underline{A}$ represents the standard timed event graph including the holding times, i.e., the minimal time durations a token has to spend in each place, and the maximally allowed tokens in places, e.g., when modeling the capacity of a resource. The elements in matrix $\overline{A}$ represent additional constraints on the upper bound of possible time windows and the minimal number of tokens which has to be present in the places. Using the Kleene star (resp. the dual Kleene star) a solution of (5.11) and (5.12) must, according to Lem. 2.52 and Lem. 2.67, fulfill

$$x = \underline{A}^* \otimes x \tag{5.13}$$

$$x = \overline{A}_* \odot x. \tag{5.14}$$

Consequently, the aim is to find an x which respects both constraints, which is equivalent to require that x is in the image of $L_{\underline{A}^*}$ and in the image of $\Lambda_{\overline{A}_*}$ (see Rem. 2.50 and Rem. 2.61). Formally,

$$\underline{A}^* \otimes x = x = \overline{A}_* \odot x \Leftrightarrow x \in \mathrm{Im} L_{\underline{A}^*} \cap \mathrm{Im} \Lambda_{\overline{A}_*}. \tag{5.15}$$

According to Lemma 2.67, Eq. 5.14 is equivalent to

$$x = \overline{A}_* \backslash\!\bullet x.$$

Inserting (5.13) in this equation results in

$$x = \overline{A}_* \backslash\!\bullet (\underline{A}^* \otimes x)$$

which according to Lem. 2.68 is equivalent to

$$x = \left(\overline{A}_* \backslash\!\bullet \underline{A}^*\right) \otimes x.$$

Theorem 5.5 ([7]). Given two matrices $\overline{A}, \underline{A} \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]^{n\times n}$ and a vector $x \in \mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]^{n}$. If $\forall x$, the equality $\overline{A}_* \backslash\!\bullet (\underline{A}^* \otimes x) = \left(\overline{A}_* \backslash\!\bullet \underline{A}^*\right) \otimes x$ holds, then the mapping

$$P : x \mapsto \left(\overline{A}_* \backslash\!\bullet \underline{A}^*\right)^* \backslash\!\circ x,$$

is a projector in $\mathrm{Im}L_{\underline{A}^*} \cap \mathrm{Im}\Lambda_{\overline{A}_*}$ and $P(x) = y$ is the greatest element in $\mathrm{Im}L_{\underline{A}^*} \cap \mathrm{Im}\Lambda_{\overline{A}_*}$ less or equal to x. Formally

$$P(x) = \bigvee \left\{ y \middle| \, y \preceq x \text{ and } y \in \mathrm{Im}L_{\underline{A}^*} \cap \mathrm{Im}\Lambda_{\overline{A}_*} \right\}.$$

Proof. First of all, according to Def. 2.40 it is clear that P is a projector on the image of $L_{(\overline{A}_* \backslash\!\bullet \underline{A}^*)^*}$, and $P(x) \preceq x$. Then, by definition (Def. 2.59) $\overline{A}_* \preceq I^{\odot}$ and consequently,

$$\overline{A}_* \backslash\!\bullet \underline{A}^* \succeq I^{\odot} \backslash\!\bullet \underline{A}^* = \underline{A}^*$$

and

$$(\overline{A}_* \backslash\!\bullet \underline{A}^*)^* \succeq (\underline{A}^*)^* = \underline{A}^*,$$

which according to Rem. 2.54 implies that $\mathrm{Im}L_{(\overline{A}_* \backslash\!\bullet \underline{A}^*)^*} \subseteq \mathrm{Im}L_{\underline{A}^*}$, hence $P(x) \in \mathrm{Im}L_{\underline{A}^*}$. Thus, we have shown that $P(x)$ is a projector in the image of $L_{\underline{A}^*}$. Now, we have to show that it is also a projector in the image of $L_{\overline{A}_*}$. Since $P(x) \in \mathrm{Im}L_{(\overline{A}_* \backslash\!\bullet \underline{A}^*)^*}$ one can write

$$P(x) = (\overline{A}_* \backslash\!\bullet \underline{A}^*)^* P(x),$$

which according to Lem. 2.52 is equivalent to

$$P(x) \succeq (\overline{A}_* \backslash\!\bullet \underline{A}^*) \otimes P(x).$$

Due to the assumption that

$$(\overline{A}_* \backslash\!\bullet \underline{A}^*) \otimes P(x) = \overline{A}_* \backslash\!\bullet (\underline{A}^* \otimes P(x))$$

and the already obtained result that $P(x) \in \mathrm{Im}L_{\underline{A}^*}$, which implies that $\underline{A}^* P(x) = P(x)$, one can write

$$\begin{aligned} P(x) &\succeq (\overline{A}_* \backslash\!\bullet \underline{A}^*) \otimes P(x) \\ &\succeq \overline{A}_* \backslash\!\bullet (\underline{A}^* \otimes P(x)) \\ &\succeq \overline{A}_* \backslash\!\bullet P(x). \end{aligned}$$

Furthermore, as mentioned before $\overline{A}_* \preceq I^{\odot}$, which implies that

$$\overline{A}_* \backslash\!\bullet P(x) \succeq I^{\odot} \backslash\!\bullet P(x) = P(x).$$

Consequently, equality holds, i.e.,

$$P(x) = \overline{A}_* \backslash\!\bullet P(x),$$

which according to Lem. 2.67 implies

$$P(x) = \overline{A}_* \backslash\!\bullet P(x) = \overline{A}_* \odot P(x).$$

This also implies (considering Rem. 2.61) that $P(x) \in \mathrm{Im}\Lambda_{\overline{A}_*}$. Thus, it has been shown that

$$P(x) \in \mathrm{Im}L_{\underline{A}^*} \cap \mathrm{Im}\Lambda_{\overline{A}_*}.$$

Last but not least, it has to be shown that $P(x)$ is the greatest element in $\mathrm{Im}L_{\underline{A}^*} \cap \mathrm{Im}\Lambda_{\overline{A}_*}$ less or equal to x. We know that $P(x) = y \preceq x$ and $y \in \mathrm{Im}L_{\underline{A}^*} \cap \mathrm{Im}\Lambda_{\overline{A}_*}$. Consequently the following equalities hold

$$\begin{aligned} y &= \underline{A}^* \otimes y \\ &= \overline{A}_* \odot y \\ &= \overline{A}_* \backslash\!\bullet y \\ &= \overline{A}_* \backslash\!\bullet (\underline{A}^* \otimes y) \\ &= \left(\overline{A}_* \backslash\!\bullet \underline{A}^*\right) \otimes y. \end{aligned}$$

Lem. 2.52 implies that

$$y = \left(\overline{A}_* \backslash\!\bullet \underline{A}^*\right) \otimes y \Rightarrow y \preceq \left(\overline{A}_* \backslash\!\bullet \underline{A}^*\right) \backslash\!\circ y,$$

which is equivalent to (see Lem. 2.52)

$$y = (\overline{A}_* \backslash\!\bullet \underline{A}^*)^* \backslash\!\circ y.$$

Since mapping $L^\sharp_{(\overline{A}_* \backslash\!\bullet \underline{A}^*)^*}$ is an isotone mapping, the following implication holds for $z \in \mathrm{Im}L_{\underline{A}^*} \cap \mathrm{Im}\Lambda_{\overline{A}_*}$

$$z \preceq x \Rightarrow (\overline{A}_* \backslash\!\bullet \underline{A}^*)^* \backslash\!\circ z \preceq (\overline{A}_* \backslash\!\bullet \underline{A}^*)^* \backslash\!\circ x$$

which also means that if $z \preceq x$ then z is also less or equal to $P(x)$, i.e., $z = (\overline{A}_* \backslash\!\bullet \underline{A}^*)^* \backslash\!\circ z \preceq P(x) = y$. Hence $P(x) = y$ is the greatest element in $\mathrm{Im}L_{\underline{A}^*} \cap \mathrm{Im}\Lambda_{\overline{A}_*}$ less or equal to x. □

Remark 5.6. Theorem 5.5 and the corresponding proof show that the mapping $P : x \mapsto (\overline{A}_* \backslash\!\bullet \underline{A}^*)^* \backslash\!\circ x$ is a projector in $\mathrm{Im}L_{\underline{A}^*} \cap \mathrm{Im}\Lambda_{\overline{A}_*}$ and $P(x)$ is the greatest element in $\mathrm{Im}L_{\underline{A}^*} \cap \mathrm{Im}\Lambda_{\overline{A}_*}$ less or equal to x. According to Lem. 2.52 mapping P is equivalent to

$$\tilde{P} : x \mapsto \underbrace{(\overline{A}_* \backslash\!\bullet \underline{A}^*)^*}_{\overline{A}^*} \otimes x.$$

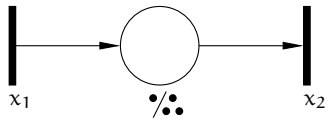


Figure 5.3: Simple TEG with constraints on the minimal and maximal number of tokens.

For a vector x, being a solution of the equation $x = \overline{\underline{A}}^* \otimes x$, it is guaranteed that the constraints

$$x \geq \underline{A} \otimes x$$
$$x \leq \overline{A} \odot x$$

are always met. Thus, matrix $\overline{\underline{A}}^*$ includes time window constraints as well as constraints on the minimal (and maximal) number of tokens.

Example 5.7. Given a part of a TEG with minimal and maximal numbers of tokens shown in Fig. 5.3. The constraints indicate that a minimum of one but never more than three tokens shall be present in the place between transition x_1 and x_2. In $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ this can be modeled by the constraints

$$x_1 \geq \gamma^3 x_2 \tag{5.16}$$
$$x_1 \leq \gamma x_2. \tag{5.17}$$

Using the introduced approach the matrices $\overline{A}_*$, $\underline{A}^*$, and $\overline{\underline{A}}^*$ can be determined and result in

$$\overline{A}_* = \begin{pmatrix} e & \gamma \\ \top & e \end{pmatrix}, \quad \underline{A}^* = \begin{pmatrix} e & \gamma^3 \\ \varepsilon & e \end{pmatrix}, \text{ and } \quad \overline{\underline{A}}^* = \begin{pmatrix} e & \gamma^3 \\ \gamma^{-1} & e \end{pmatrix}.$$

However, it is also possible to rewrite constraint (5.17), i.e.,

$$x_2 \geq \gamma^{-1} x_1.$$

Thus, the behavior of the system can also be modeled in terms of negative tokens. The corresponding representation as a timed event graph is given in Fig. 5.4. and the resulting matrices A and A^* are

$$A = \begin{pmatrix} \varepsilon & \gamma^3 \\ \gamma^{-1} & \varepsilon \end{pmatrix} \quad \text{and} \quad A^* = \begin{pmatrix} e & \gamma^3 \\ \gamma^{-1} & e \end{pmatrix}.$$

Clearly, the matrices $\overline{\underline{A}}^*$ and A^* are identical, which indicates the close relationship of negative tokens and constraints on the minimal number of tokens. Consequently, one can either model the behavior of a system with a negative number of tokens or add constraints on the corresponding states of a system.

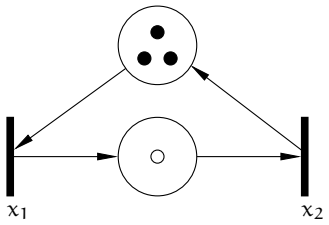


Figure 5.4: TEG with negative token representing the behavior of the TEG given in Fig. 5.3.

5.2 Control of extended TEG with constraints

The aim of control in this section is to find a state feedback controller and a pre-filter such that the controlled system is the greatest system possible which is less or equal to a given reference model G_{ref} for any input v and at the same time the controlled system shall respect all imposed constraints including time window constraints and constraints on the number of tokens. Determining the greatest possible system which is less or equal to a given reference model by adding a pre-filter and a state feedback controller corresponds to the standard approach of state feedback control in a dioid setting (see Sec. 4.2). The controlled state of the system then evolves according to the equation

$$x = (\underline{A} \oplus BF)^* BPv,$$

where $\underline{A}$ is the system matrix of the standard timed event graph. Furthermore, it has to be guaranteed that the controlled system respects all imposed additional constraints for all inputs v. Thus, the state vector x has to be a solution of the equation $x = \overline{\underline{A}}^* x$ (see Rem. 5.6), where $\overline{\underline{A}}^*$ is the system matrix which contains all constraints. Formally, this can be written in two dependencies

$$C(\underline{A} \oplus BF)^* BP \preceq G_{ref} \tag{5.18}$$

$$(\underline{A} \oplus BF)^* BP = \overline{\underline{A}}^* (\underline{A} \oplus BF)^* BP. \tag{5.19}$$

(5.18) basically means that, with respect to the output, the controlled system should be at least as fast as the reference model for any input. A feedback controller and pre-filter fulfilling Eq. 5.19 guarantee that all additional constraints modeled in matrix $\overline{A}$ are met, or formally, that the controlled system is in the image of matrix $\overline{\underline{A}}^*$ (see Rem. 2.50), i.e.,

$$\mathrm{Im}(\underline{A} \oplus BF)^* BP \subseteq \mathrm{Im}\overline{\underline{A}}^*.$$

Using residuation theory, (5.18) can be rewritten

$$\underbrace{(\underline{A} \oplus BF)^* BP}_{H_{xv}} \preceq C \backslash G_{ref},$$

where H_{xv} is the transfer relation between the input of the pre-filter and the corresponding system state. According to Eq. 5.19 every column of H_{xv} has to be in the image of matrix $\overline{\underline{A}}^*$. Consequently, the aim is to find the greatest $H_{xv} \preceq C \backslash\!\!\diamond G_{ref}$ such that $\mathrm{Im} H_{xv} \subseteq \mathrm{Im} \overline{\underline{A}}^*$, which is equivalent to

$$\begin{aligned} \hat{H}_{xv} &= \overline{\underline{A}}^* \backslash\!\!\diamond C \backslash\!\!\diamond G_{ref} \\ &= (C \overline{\underline{A}}^*) \backslash\!\!\diamond G_{ref}. \end{aligned}$$

Therefore, the objectives (5.18) and (5.19) can be reformulated

$$(\underline{A} \oplus BF)^* BP \preceq \hat{H}_{xv} \tag{5.20}$$

$$(\underline{A} \oplus BF)^* BP = \overline{\underline{A}}^* (\underline{A} \oplus BF)^* BP. \tag{5.21}$$

As in Sec. 4.2, inequality 5.20 can be written as

$$\underline{A}^* B (F \underline{A}^* B)^* BP \preceq \hat{H}_{xv}.$$

Therefore, any pre-filter has to satisfy

$$\underline{A}^* BP \preceq \hat{H}_{xv}.$$

Using residuation theory this can be rewritten

$$P \preceq (\underline{A}^* B) \backslash\!\!\diamond \hat{H}_{xv}.$$

Furthermore, it has to be ensured that the pre-filter P fulfills objective (5.21), this implies that, independent of F,

$$\underline{A}^* BP = \overline{\underline{A}}^* \underline{A}^* BP \tag{5.22}$$

Since, by definition (Def. 2.59), $\overline{A}_* \preceq I^{\odot}$, it can easily be verified that

$$\overline{A}_* \backslash\!\!\bullet \underline{A}^* \succeq I^{\odot} \backslash\!\!\bullet \underline{A}^* = \underline{A}^*$$

and consequently,

$$\underbrace{(\overline{A}_* \backslash\!\!\bullet \underline{A}^*)^*}_{\overline{\underline{A}}^*} \succeq (\underline{A}^*)^* = \underline{A}^*.$$

Furthermore, according to Lem. 2.53, (5.22) is equivalent to

$$\underline{A}^* BP = \overline{\underline{A}}^* BP$$

which is identical to require that

$$\overline{\underline{A}}^{*}B P \geq \underline{A}^{*}BP \tag{5.23}$$

AND

$$\overline{\underline{A}}^{*}B P \leq \underline{A}^{*}BP. \tag{5.24}$$

Property (5.23) can easily be verified as $\overline{\underline{A}}^{*} \geq \underline{A}^{*}$. For property (5.24) residuation theory can be applied, which results in

$$P \preceq (\overline{\underline{A}}^{*}B) \backslash\!\!\backslash (\underline{A}^{*}BP) \Leftrightarrow P = P \wedge (\overline{\underline{A}}^{*}B) \backslash\!\!\backslash (\underline{A}^{*}BP).$$

Furthermore, this pre-filter has to be causal in order to be realizable (see Def. 3.30). This means that P has to satisfy the following equality

$$P = Pr^{\mp}_{caus}(P) \Leftrightarrow \begin{array}{l} P \geq Pr^{\mp}_{caus}(P) \\ P \preceq Pr^{\mp}_{caus}(P) \end{array}$$

According to Def. 3.30, $Pr^{\mp}_{caus}(P) \preceq P$ and, consequently, we are looking for a pre-filter P such that,

$$P \preceq Pr^{\mp}_{caus}(P) \Leftrightarrow P = P \wedge Pr^{\mp}_{caus}(P).$$

To summarize, we are looking for a pre-filter such that

$$P = P \wedge (\overline{\underline{A}}^{*}B) \backslash\!\!\backslash (\underline{A}^{*}BP) \wedge Pr^{\mp}_{caus}(P) \wedge (\underline{A}^{*}B) \backslash\!\!\backslash \hat{H}_{xv},$$

and the optimal pre-filter can then be determined by

$$P_{opt} = \bigoplus \left\{ P \mid P = P \wedge (\overline{\underline{A}}^{*}B) \backslash\!\!\backslash (\underline{A}^{*}BP) \wedge Pr^{\mp}_{caus}(P) \wedge (\underline{A}^{*}B) \backslash\!\!\backslash \hat{H}_{xv} \right\}. \tag{5.25}$$

Clearly, the mapping $\Pi(P) = P \wedge (\overline{\underline{A}}^{*}B) \backslash\!\!\backslash (\underline{A}^{*}BP) \wedge Pr^{\mp}_{caus}(P) \wedge (\underline{A}^{*}B) \backslash\!\!\backslash \hat{H}_{xv}$ is isotone and non-increasing. Then, if a solution exists, the following fixed point algorithm will converge to the greatest solution (see e.g., [22, 27]):

$$\begin{array}{rl} & P^{(0)} = (\underline{A}^{*}B) \backslash\!\!\backslash \hat{H}_{xv} \\ \texttt{do} & P^{(i+1)} = \Pi(P^{(i)}) \\ \texttt{while} & P^{(i+1)} \neq P^{(i)}. \end{array}$$

Taking a closer look at the starting point

$$\begin{aligned} P^{(0)} &= (\underline{A}^{*}B) \backslash\!\!\backslash \hat{H}_{xv} \\ &= (\underline{A}^{*}B) \backslash\!\!\backslash (C\overline{\underline{A}}^{*}) \backslash\!\!\backslash G_{ref} \\ &= (C\overline{\underline{A}}^{*}\underline{A}^{*}B) \backslash\!\!\backslash G_{ref} \end{aligned}$$

and since $\overline{\underline{A}}^{*} \geq \underline{A}^{*}$ this is equivalent to

$$P^{(0)} = (C\overline{\underline{A}}^{*}B) \backslash\!\!\backslash G_{ref}.$$

Remark 5.8. Obviously, if $P^{(0)} = (C\overline{\underline{A}}^* B)\backslash\!\!\backslash G_{ref}$ is causal and $(C\overline{\underline{A}}^*)\backslash\!\!\backslash G_{ref} \in \operatorname{Im} \underline{A}^* B$ then it is a solution. Indeed in this case $\operatorname{Pr}^{\mp}_{caus}(P^{(0)}) = P^{(0)}$ and $(C\overline{\underline{A}}^*)\backslash\!\!\backslash G_{ref} \in \operatorname{Im} \underline{A}^* B$ then

$$\begin{aligned}(C\overline{\underline{A}}^*)\backslash\!\!\backslash G_{ref} &= \underline{A}^* B \left((\underline{A}^* B)\backslash\!\!\backslash (C\overline{\underline{A}}^*)\backslash\!\!\backslash G_{ref} \right) \\ &= \underline{A}^* B \left((C\overline{\underline{A}}^* B)\backslash\!\!\backslash G_{ref} \right)\end{aligned}$$

hence,

$$\begin{aligned}(\overline{\underline{A}}^* B)\backslash\!\!\backslash (\underline{A}^* B P^{(0)}) &= (\overline{\underline{A}}^* B)\backslash\!\!\backslash \left(\underline{A}^* B \left((C\overline{\underline{A}}^* B)\backslash\!\!\backslash G_{ref} \right) \right) \\ &= (\overline{\underline{A}}^* B)\backslash\!\!\backslash \left((C\overline{\underline{A}}^*)\backslash\!\!\backslash G_{ref} \right) \\ &= (C\overline{\underline{A}}^* B)\backslash\!\!\backslash G_{ref} = P^{(0)}\end{aligned}$$

which leads to

$$P_{opt} = P^{(0)}.$$

The obtained optimal pre-filter P_{opt} fulfills the necessary requirements

$$\underline{A}^* B P_{opt} \preceq \hat{H}_{xv} \tag{5.26}$$

$$\underline{A}^* B P_{opt} = \overline{\underline{A}}^* B P_{opt}. \tag{5.27}$$

Once an optimal pre-filter has been determined, the aim is to find a feedback controller F that preserves the properties of the pre-filtered system and respects all additional constraints modeled in matrix $\overline{A}$. Formally, the feedback controller F must meet the following requirements

$$\underline{A}^* B P_{opt} \preceq (\underline{A} \oplus BF)^* B P_{opt} \preceq \underline{A}^* B P_{opt} \tag{5.28}$$

$$(\underline{A} \oplus BF)^* B P_{opt} = \overline{\underline{A}}^* (\underline{A} \oplus BF)^* B P_{opt}. \tag{5.29}$$

Clearly, by definition of "$\oplus$" the left hand side of (5.28) can easily be verified, i.e., $\underline{A}^* B P_{opt} \preceq (\underline{A} \oplus BF)^* B P_{opt}$, $\forall F$. Furthermore, it can easily be recognized that $F = \mathcal{E}$ (the zero matrix) is a (trivial) solution, i.e., $\underline{A}^* B P_{opt} \preceq (\underline{A} \oplus B\mathcal{E})^* B P_{opt} = \underline{A}^* B P_{opt} \preceq \underline{A}^* B P_{opt}$ and (5.29) reduces to (5.27).

To determine feedback controllers $F \succeq \mathcal{E}$, residuation theory is applied to the right hand side of (5.28)

$$\begin{aligned}(\underline{A} \oplus BF)^* B P_{opt} &\preceq \underline{A}^* B P_{opt} \\ (\underline{A}^* BF)^* \underline{A}^* B P_{opt} &\preceq \underline{A}^* B P_{opt} \\ (\underline{A}^* BF)^* &\preceq (\underline{A}^* B P_{opt}) \not\backslash (\underline{A}^* B P_{opt}).\end{aligned}$$

Obviously, the right hand side of the last inequality is a Kleene star. Therefore, the controller F can be determined by

$$F \preceq (\underline{A}^* B) \backslash\!\!\backslash (\underline{A}^* B P_{opt}) /\!\!/ (\underline{A}^* B P_{opt})$$

and the greatest solution of this inequality achieves equality, i.e.,

$$F_{max} = (\underline{A}^* B) \backslash\!\!\backslash (\underline{A}^* B P_{opt}) /\!\!/ (\underline{A}^* B P_{opt}).$$

However, F_{max} may not be causal. The greatest causal feedback controller is given by

$$F_{opt} = Pr^{\mp}_{caus}(F_{max}) = Pr^{\mp}_{caus}\left((\underline{A}^* B) \backslash\!\!\backslash (\underline{A}^* B P_{opt}) /\!\!/ (\underline{A}^* B P_{opt})\right). \tag{5.30}$$

Furthermore, it is known that $\mathcal{E} \preceq F_{opt}$ and, by definition of the causal projection $Pr^{\mp}_{caus}$, $F_{opt} \preceq F_{max}$. Then

$$\underline{A}^* B P_{opt} \preceq (\underline{A} \oplus B\mathcal{E})^* B P_{opt} \preceq (\underline{A} \oplus B F_{opt})^* B P_{opt} \preceq (\underline{A} \oplus B F_{max})^* B P_{opt} \preceq \underline{A}^* B P_{opt}$$

and consequently, the greatest causal feedback controller F_{opt} achieves equality in (5.28), i.e., F_{opt} is such that

$$(\underline{A}^* \oplus B F_{opt})^* B P_{opt} = \underline{A}^* B P_{opt},$$

which, according to Eq. 5.27, is equivalent to

$$\begin{aligned} (\underline{A}^* \oplus B F_{opt})^* B P_{opt} &= \overline{\underline{A}}^* B P_{opt} \\ &\Downarrow \\ \overline{\underline{A}}^* (\underline{A} \oplus B F_{opt})^* B P_{opt} &= \overline{\underline{A}}^* \overline{\underline{A}}^* B P_{opt} = \overline{\underline{A}}^* B P_{opt} \end{aligned}$$

and which then ensures the requirement of (5.29). Thus, the optimal controller F_{opt} determined by (5.30) meets the requirements (5.28) and (5.29) and, therefore, it can be guaranteed that the initial constraints (5.18) and (5.19) are fulfilled.

Example 5.9 (System with Constraints). Consider a system represented by the extended TEG given in Fig. 5.5. This system contains time window constraints as well as constraints on the minimal number of tokens. Namely, the place between x_2 and x_5 shall contain at least one token but never more than two. Furthermore, a token entering the place between transition x_4 and x_5 has to stay at least one time unit in the place but has to be removed at the latest after four time units. The corresponding dependencies in $\mathcal{M}^{ax}_{in}[\![\gamma, \delta]\!]$ can be written

$$\begin{aligned} x_2 &\geq \gamma^2 x_5 \\ x_5 &\geq \delta x_4 \end{aligned}$$

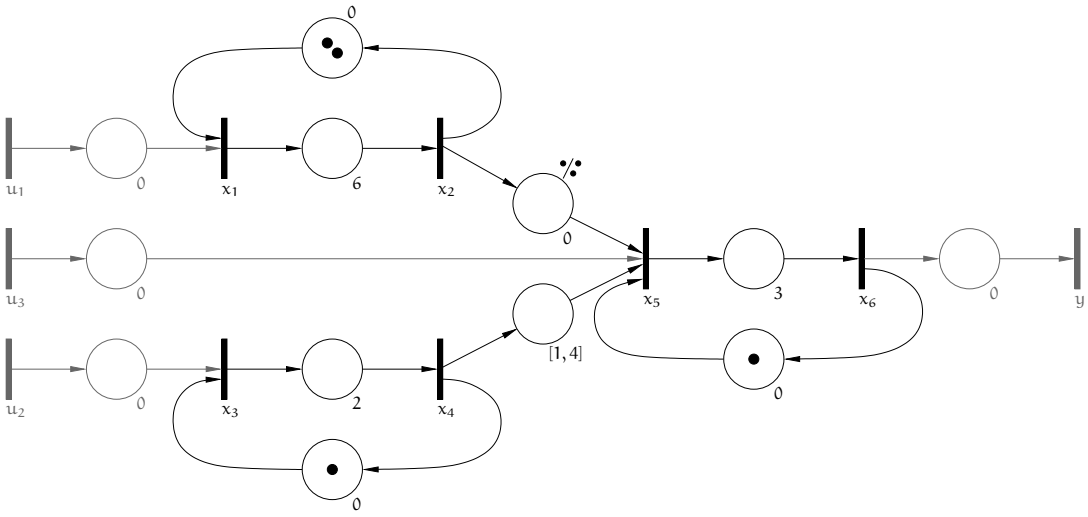


Figure 5.5: Extended TEG with time window constraints and constraints on the minimal number of tokens.

and

$$x_2 \preceq \gamma x_5$$
$$x_5 \preceq \delta^4 x_4.$$

The system matrices $\underline{A}$ and $\overline{A}$ in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ are

$$\underline{A} = \begin{pmatrix} \varepsilon & \gamma^2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^6 & \varepsilon & \varepsilon & \varepsilon & \gamma^2 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \gamma & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \delta^2 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \delta & \varepsilon & \gamma \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta^3 & \varepsilon \end{pmatrix}, \quad \overline{A} = \begin{pmatrix} \top & \top & \top & \top & \top & \top \\ \top & \top & \top & \top & \gamma & \top \\ \top & \top & \top & \top & \top & \top \\ \top & \top & \top & \top & \top & \top \\ \top & \top & \top & \delta^4 & \top & \top \\ \top & \top & \top & \top & \top & \top \end{pmatrix}$$

and consequently, the matrix $\overline{\underline{A}}^*$ capturing all constraints is

$$\overline{\underline{A}}^* = \left(\overline{A}_* \backslash \underline{A}^*\right)^* = \begin{pmatrix} (\gamma^2\delta^6)^* & \gamma^2(\gamma^2\delta^6)^* & \gamma^4\delta^3(\gamma\delta^3)^* & \cdots \\ \delta^6(\gamma^2\delta^6)^* & (\gamma^2\delta^6)^* & \gamma^2\delta^3(\gamma\delta^3)^* & \\ \delta^2(\gamma\delta^3)^* & \delta^{-4}(\gamma\delta^3)^* & e \oplus \gamma\delta^2 \oplus \gamma^2\delta^4 \oplus \gamma^3\delta^6 \oplus (\gamma^4\delta^8)(\gamma\delta^3)^* & \\ \gamma^{-1}\delta^2(\gamma\delta^3)^* & \gamma^{-1}\delta^{-4}(\gamma\delta^3)^* & \delta^2 \oplus \gamma\delta^4 \oplus \gamma^2\delta^6 \oplus (\gamma^3\delta^8)(\gamma\delta^3)^* & \\ \gamma^{-1}\delta^6(\gamma\delta^3)^* & \gamma^{-1}(\gamma\delta^3)^* & \delta^3(\gamma\delta^3)^* & \\ \gamma^{-1}\delta^9(\gamma\delta^3)^* & \gamma^{-1}\delta^3(\gamma\delta^3)^* & \delta^6(\gamma\delta^3)^* & \end{pmatrix}$$

$$\begin{pmatrix} \cdots & \gamma^4\delta(\gamma\delta^3)^* & \gamma^4(\gamma\delta^3)^* & \gamma^5(\gamma\delta^3)^* \\ & \gamma^2\delta(\gamma\delta^3)^* & \gamma^2(\gamma\delta^3)^* & \gamma^3(\gamma\delta^3)^* \\ & \gamma \oplus \gamma^2\delta^2 \oplus \gamma^3\delta^4 \oplus (\gamma^4\delta^6)(\gamma\delta^3)^* & \gamma\delta^{-4}(\gamma\delta^3)^* & \gamma^2\delta^{-4}(\gamma\delta^3)^* \\ & e \oplus \gamma\delta^2 \oplus \gamma^2\delta^4 \oplus (\gamma^3\delta^6)(\gamma\delta^3)^* & \delta^{-4}(\gamma\delta^3)^* & \gamma\delta^{-4}(\gamma\delta^3)^* \\ & \delta(\gamma\delta^3)^* & (\gamma\delta^3)^* & \gamma(\gamma\delta^3)^* \\ & \delta^4(\gamma\delta^3)^* & \delta^3(\gamma\delta^3)^* & (\gamma\delta^3)^* \end{pmatrix}.$$

Furthermore, according to the TEG shown in Fig. 5.5 the matrices B and C are

$$B = \begin{pmatrix} e & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e \\ \varepsilon & \varepsilon & \varepsilon \end{pmatrix} \quad \text{and} \quad C = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e \end{pmatrix}.$$

Aim is to find a pre-filter P and a feedback controller F such that the constraints (5.18) and (5.19) with the given reference model

$$G_{\text{ref}} = \begin{pmatrix} \delta^{12}(\gamma\delta^4)^* & \delta^6(\gamma\delta^4)^* & \delta^3(\gamma\delta^4)^* \end{pmatrix}.$$

are fulfilled. Note that this example (and all the following ones) were computed with the C++ library *MinmaxGD* [18], which has been developed by Prof. Hardouin's group at Université d'Angers, France. For this thesis, the original library has been equipped with additional algorithms for the dual Kleene star, the dual left residuation, and for the causal projection of transfer series $\text{Pr}^{\mp}_{\text{caus}}$. Using this software and according to the described approach, a suitable pre-filter is determined by a fixed point algorithm. In this case the fixed point algorithm converges in one step, i.e., $P_{\text{opt}} = P^{(1)} = \Pi(P^{(0)}) = P^{(0)} \wedge (\overline{\underline{A}}^* B) \backslash (\underline{A}^* B P^{(0)}) \wedge \text{Pr}^{\mp}_{\text{caus}}(P^{(0)}) \wedge (\underline{A}^* B) \backslash \hat{H}_{xv}$, and results in

$$P_{\text{opt}} = \begin{pmatrix} \gamma\delta^3(\gamma\delta^4)^* & \gamma^2\delta(\gamma\delta^4)^* & \gamma^3\delta^2(\gamma\delta^4)^* \\ \delta^6(\gamma\delta^4)^* & (\gamma\delta^4)^* & \gamma\delta(\gamma\delta^4)^* \\ \delta^9(\gamma\delta^4)^* & \delta^3(\gamma\delta^4)^* & (\gamma\delta^4)^* \end{pmatrix}.$$

In a second step the optimal feedback controller can be determined and results in

$$
\begin{aligned}
F_{opt} &= \mathrm{Pr}^{\top}_{caus}\left((\underline{A}^{*}B)\backslash\!\!\backslash(\underline{A}^{*}BP_{opt})/\!\!/(\underline{A}^{*}BP_{opt})\right) \\
&= \begin{pmatrix}
(\gamma\delta^{4})^{*} & \gamma^{2}\delta^{2}(\gamma\delta^{4})^{*} & \gamma^{2}\delta(\gamma\delta^{4})^{*} & \\
\gamma^{-1}\delta^{3}(\gamma\delta^{4})^{*} & \delta(\gamma\delta^{4})^{*} & (\gamma\delta^{4})^{*} & \cdots \\
\gamma^{-1}\delta^{6}(\gamma\delta^{4})^{*} & \gamma^{-1}(\gamma\delta^{4})^{*} & \delta^{3}(\gamma\delta^{4})^{*} &
\end{pmatrix} \\
&\qquad \begin{pmatrix}
\gamma^{3}\delta^{3}(\gamma\delta^{4})^{*} & \gamma^{3}\delta^{2}(\gamma\delta^{4})^{*} & \gamma^{4}\delta^{3}(\gamma\delta^{4})^{*} \\
\cdots & \gamma\delta^{2}(\gamma\delta^{4})^{*} & \gamma\delta(\gamma\delta^{4})^{*} & \gamma^{2}\delta^{2}(\gamma\delta^{4})^{*} \\
& \delta(\gamma\delta^{4})^{*} & (\gamma\delta^{4})^{*} & \gamma\delta(\gamma\delta^{4})^{*}
\end{pmatrix}.
\end{aligned}
$$

The optimal pre-filter and feedback controller fulfill constraints (5.18) and (5.19). More precisely, in this specific example, we get equality also for the first constraint, i.e.,

$$
C(\underline{A} \oplus BF_{opt})^{*}BP_{opt} = \begin{pmatrix} \delta^{12}(\gamma\delta^{4})^{*} & \delta^{6}(\gamma\delta^{4})^{*} & \delta^{3}(\gamma\delta^{4})^{*} \end{pmatrix} = G_{ref}.
$$

5.3 Unfeasible constraints

An important issue not mentioned so far is related to unfeasible constraints, e.g., when the upper bound of a time window constraint is smaller than the lower bound of this time window or the time window is not feasible with respect to overall TEG. Formally, the first case means that there are two constraints

$$
\begin{aligned}
x_j &\geq \delta^{\underline{t}} x_i \\
x_j &\preceq \delta^{\overline{t}} x_i
\end{aligned}
$$

with $\underline{t} > \overline{t}$. The corresponding TEG is given in Fig. 5.6. The resulting system model of this TEG with unfeasible time window constraints is $x = \overline{\underline{A}}^{*} x$, with

$$
\begin{aligned}
\overline{\underline{A}}^{*} &= \left(\overline{A}_{*} \backslash\!\!\bullet \underline{A}^{*}\right)^{*} \\
&= \left(\begin{pmatrix} e & \top \\ \delta^{\overline{t}} & e \end{pmatrix} \backslash\!\!\bullet \begin{pmatrix} e & \varepsilon \\ \delta^{\underline{t}} & e \end{pmatrix}\right)^{*}
\end{aligned}
$$

and $x = (x_i \;\; x_j)^{\mathsf{T}}$. Since all elements of $\overline{A}_{*}$ are either ε, $\top$, or have a multiplicative inverse, matrix $(\overline{A}_{*} \backslash\!\!\bullet \underline{A}^{*})$ can be computed by

$$
\begin{aligned}
(\overline{A}_{*} \backslash\!\!\bullet \underline{A}^{*}) &= \left(\begin{pmatrix} e & \top \\ \delta^{\overline{t}} & e \end{pmatrix} \backslash\!\!\bullet \begin{pmatrix} e & \varepsilon \\ \delta^{\underline{t}} & e \end{pmatrix}\right) \\
&= \begin{pmatrix} \delta^{(\underline{t}-\overline{t})} & \delta^{-\overline{t}} \\ \delta^{\underline{t}} & e \end{pmatrix},
\end{aligned}
$$

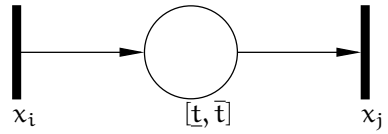


Figure 5.6: Simple TEG with unfeasible time window constraints, i.e., $\underline{t} > \overline{t}$.

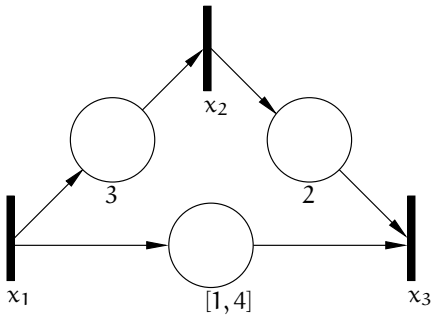


Figure 5.7: TEG with unfeasible time window constraints.

and since $\underline{t} - \overline{t} > 0$ matrix $\overline{\underline{A}}^*$ results in

$$\overline{\underline{A}}^* = (\overline{A}_* \backslash \underline{A}^*)^* = \begin{pmatrix} \delta^\infty & \delta^\infty \\ \delta^\infty & \delta^\infty \end{pmatrix}.$$

This result means, the only way to guarantee that the time window constraints are not violated is to never start the system, i.e., all events fire for the first time at the earliest at time $t = \infty$.

In the following an example with time window constraints is introduced. The time window itself is feasible, i.e., the lower bound of the time window is less (or equal) to its upper bound. However, with respect to the overall TEG the time window is not feasible.

Example 5.10 (TEG with unfeasible time window constraints). Consider the TEG given in Fig. 5.7. Clearly, the time window itself models a possible behavior, however, even if transition x_2 fires as soon as it is enabled, i.e., 3 time units after the firing of x_1, transition x_3 can fire at the earliest 2 time units after the firing of x_2 and consequently 5 time units after the firing of x_1, which makes it impossible to meet the upper bound of the time window between x_1 and x_3. The matrices $\overline{A}$ and $\underline{A}$ describing the behavior of the TEG in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ and their

corresponding Kleene stars are

$$\overline{A} = \begin{pmatrix} \top & \top & \top \\ \top & \top & \top \\ \delta^4 & \top & \top \end{pmatrix}, \quad \overline{A}_* = \begin{pmatrix} e & \top & \top \\ \top & e & \top \\ \delta^4 & \top & e \end{pmatrix}$$

$$\underline{A} = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \delta^3 & \varepsilon & \varepsilon \\ \delta & \delta^2 & \varepsilon \end{pmatrix}, \quad \underline{A}^* = \begin{pmatrix} e & \varepsilon & \varepsilon \\ \delta^3 & e & \varepsilon \\ \delta^5 & \delta^2 & e \end{pmatrix}.$$

As all entries in $\overline{A}_*$ are monomials, the matrix $(\overline{A}_* \backslash\!\bullet \underline{A}^*)$ can be determined

$$(\overline{A}_* \backslash\!\bullet \underline{A}^*) = \begin{pmatrix} \delta & \delta^{-2} & \delta^{-4} \\ \delta^3 & e & \varepsilon \\ \delta^5 & \delta^2 & e \end{pmatrix},$$

and applying the Kleene star one obtains

$$\overline{\underline{A}}^* = (\overline{A}_* \backslash\!\bullet \underline{A}^*)^* = \begin{pmatrix} \delta^\infty & \delta^\infty & \delta^\infty \\ \delta^\infty & \delta^\infty & \delta^\infty \\ \delta^\infty & \delta^\infty & \delta^\infty \end{pmatrix}.$$

Once again, the result indicates that all constraints can only be fulfilled if the system never starts at all.

Hence if the user, modeling a system with time window constraints, makes a mistake and asks for unfeasible constraints with respect to a time window, the resulting linear model in $\mathcal{M}_{\text{in}}^{\text{ax}}[\![\gamma, \delta]\!]$ will "tell" the user to check his or her constraints once more.

A similar effect can be observed if unfeasible constraints with respect to the number of tokens are examined. Consider, for example, the constraints

$$x_i \geq \gamma^{\underline{k}} \delta^\tau x_j$$
$$x_i \preceq \gamma^{\overline{k}} x_j,$$

i.e., at any time t transition x_i may fire at most $\underline{k}$ times more often than x_j has fired at time $t - \tau$ and x_i shall fire at least $\overline{k}$ times more often than x_j at time t. The corresponding matrix $(\overline{A}_* \backslash\!\bullet \underline{A}^*)$ is

$$(\overline{A}_* \backslash\!\bullet \underline{A}^*) = \begin{pmatrix} e & \gamma^{\underline{k}} \delta^\tau \\ \gamma^{-\overline{k}} & \gamma^{(\underline{k}-\overline{k})} \delta^\tau \end{pmatrix},$$

and with $\underline{k} - \overline{k} < 0$ and $\tau > 0$ the corresponding Kleene star results in

$$\overline{\underline{A}}^* = (\overline{A}_* \backslash\!\bullet \underline{A}^*)^* = \begin{pmatrix} \top & \top \\ \top & \top \end{pmatrix}.$$

Not surprisingly, this result shows that the only way to guarantee fulfillment of all constraints is to block the system, i.e., to never start it at all.

Of course, a TEG may contain time window constraints as well as constraints on the number of tokens. However, it is not possible to have a time window constraint as well as a constraint on the minimal and maximal number of tokens for a simple TEG comprised to two transitions as shown in Fig. 5.6. Consider, for example, the time window constraints

$$x_j \geq \delta^{\underline{t}} x_i \tag{5.31}$$

$$x_j \preceq \delta^{\overline{t}} x_i, \tag{5.32}$$

with $\underline{t} \leqslant \overline{t}$ and the constraints on the number of firings, e.g.,

$$x_i \geq \gamma^{\underline{k}} \delta^{\tau} x_j \tag{5.33}$$

$$x_i \preceq \gamma^{\overline{k}} x_j, \tag{5.34}$$

with $\underline{k} \geqslant \overline{k}$ and $\tau > 0$. Clearly, constraints (5.31) and (5.32) by themselves as well as constraints (5.33) and (5.34) by themselves are feasible. The resulting matrices $\overline{A}$ and $\underline{A}$ and their corresponding (dual) Kleene star are

$$\underline{A} = \begin{pmatrix} \varepsilon & \delta^{\underline{t}} \\ \gamma^{\underline{k}} \delta^{\tau} & \varepsilon \end{pmatrix} \qquad \underline{A}^* = \begin{pmatrix} \left(\gamma^{\underline{k}} \delta^{(\underline{t}+\tau)}\right)^* & \delta^{\underline{t}} \left(\gamma^{\underline{k}} \delta^{(\underline{t}+\tau)}\right)^* \\ \gamma^{\underline{k}} \delta^{\tau} \left(\gamma^{\underline{k}} \delta^{(\underline{t}+\tau)}\right)^* & \left(\gamma^{\underline{k}} \delta^{(\underline{t}+\tau)}\right)^* \end{pmatrix}$$

$$\overline{A} = \begin{pmatrix} \top & \delta^{\overline{t}} \\ \gamma^{\overline{k}} & \top \end{pmatrix} \qquad \overline{A}_* = \begin{pmatrix} \gamma^{\infty} \delta^{0} & \gamma^{\infty} \delta^{\overline{t}} \\ \gamma^{\infty} \delta^{0} & \gamma^{\infty} \delta^{0} \end{pmatrix}.$$

Then, matrix $(\overline{A}_* \backslash\!\bullet \underline{A}^*)$ results in

$$(\overline{A}_* \backslash\!\bullet \underline{A}^*) = \begin{pmatrix} \top & \top \\ \top & \top \end{pmatrix} = (\overline{A}_* \backslash\!\bullet \underline{A}^*)^* = \overline{\underline{A}}^*.$$

Consequently, even though the constraints by themselves are feasible, it is not possible to combine these constraints as the resulting matrix $\overline{\underline{A}}^*$ indicates that the constraints can only be met if the system is prevented from firing at all. Taking a closer look at matrix $\overline{A}$, one can see that there is a loop, i.e., transition x_i is connected with x_j, which is again connected with x_i. By computing the dual Kleene star of matrix $\overline{A}$, i.e., $\overline{A}_* = \bigwedge_{k=0}^{\infty} A^{\odot k}$, this loop is repeated infinitely often. Therefore, the exponent of γ in this loop increases for increasing k and since $\gamma^{k_1} \delta^{t_1} \wedge \gamma^{k_2} \delta^{t_2} = \gamma^{\max(k_1,k_2)} \delta^{\min(t_1,t_2)}$, we obtain the dual Kleene star of $\overline{A}$ as given above.

6 High-Throughput Screening Systems

6.1 Introduction to HTS systems

Among the vast variety of real systems which can be modeled and controlled with the approaches mentioned and introduced in this thesis are the so called *high-throughput screening systems*. High-throughput screening (HTS) has become an important technology to rapidly test thousands of biochemical substances [25, 57]. In pharmaceutical industries, for example, HTS is often used for a first screening in the process of drug discovery. This first screening helps to reduce the almost unlimited number of possible combinations of active ingredients to a reasonable number of compounds, on which further screening methods are applied.

In general, high-throughput screening plants are fully automated systems containing a fixed set of devices performing, e.g., liquid handling, storage, reading, and incubation processes. An example of a HTS plant, developed and produced by CyBio AG, Jena, is given in Fig. 6.1.

All operations necessary to analyze one set of substances are combined in a so-called *batch*. The testing vessel, i.e., the carrier of the substances to be tested, is called a *microplate*. Such a microplate features a grid of up to 3456 wells. The number of wells is always a multiple of 96, which is the number of wells on the microplates of the first HTS systems [46]. A batch may incorporate several microplates conveying reagents or waste material. A screening definition in HTS is called *assay*. It consists of a limited or unlimited number of batches and the single batch time scheme, i.e., the definition and sequence of worksteps (also called activities) that have to be performed on a single batch and their timing. During the operation of a HTS system several batches may be processed at the same time. Furthermore, the system usually

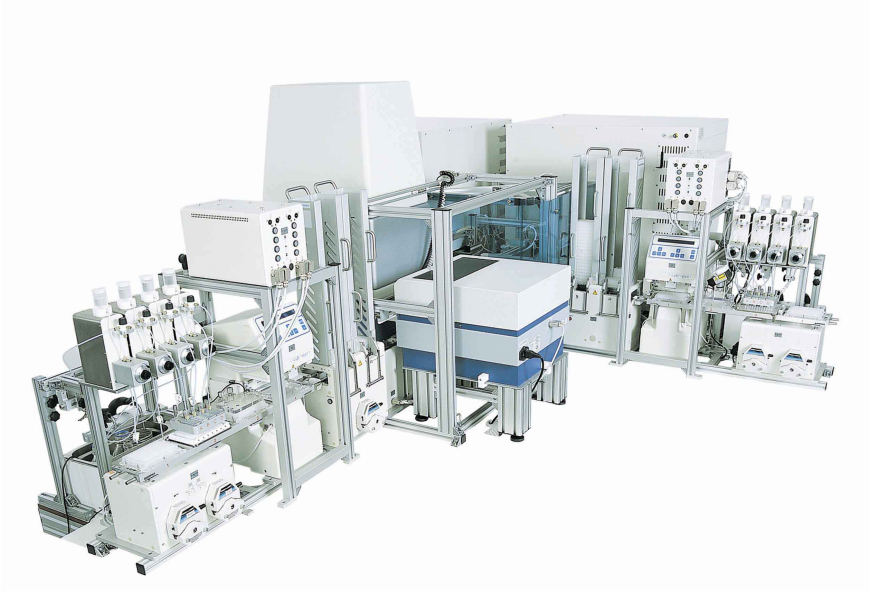


Figure 6.1: High-throughput screening plant (CyBio AG, Jena).

works with a re-entrant work flow, i.e., several activities of a single batch may be performed on the same resource, and a single batch may occupy two (or more) resources simultaneously. The definition of a single batch contains the specific (logical) sequence of activities to be performed and the minimal and maximal processing times. For comparison reasons the single batch time scheme needs to be identical for all batches to be analyzed.

6.2 Model of HTS processes

For a more comprehensible presentation of HTS processes in general as well as their modeling in terms of timed event graphs, a small HTS example is considered. A single batch in this example consists of four activities, which are executed on three different resources. A graph model the single batch time scheme is given in the upper part of Fig. 6.2. In such a graph model, nodes represent events, arcs represent the connection between events, and the numbers connected to arcs are the corresponding minimal durations between the events. In our example, every node labeled o_i or r_i represents the start event or release event, respectively, of activity i. Furthermore, double arcs each attached with 0 correspond to the synchronization of two events belonging to different activities. The minimal durations associated with the arcs connecting node a and b are marked with a $*$ and one of these durations is a negative number. This indicates that there is a time window for the corresponding events. In our example, at least 10 time units have to pass between the occurrence of event a and the occurrence of

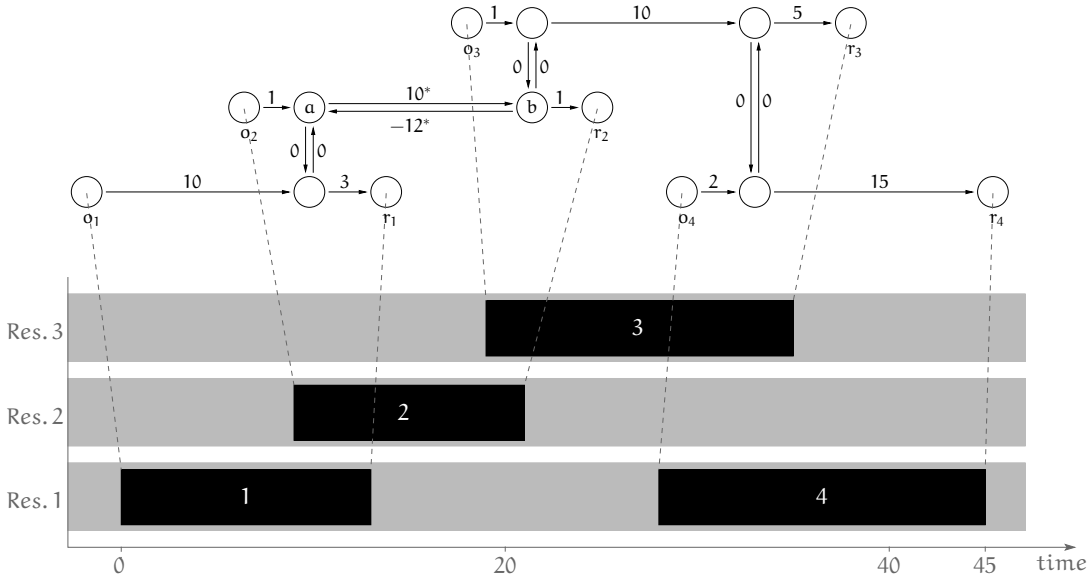


Figure 6.2: Graph model and Gantt chart for a single batch.

event b, but event b has to occur at the latest 12 time units after the occurrence of event a. In HTS systems time windows are often given for the incubation, i.e., the process in which the compounds undergo possible biochemical reactions. Fig. 6.2 also shows a Gantt chart that corresponds to the given graph model. In this Gantt chart it can be seen that the three different activities are executed on three resources. Furthermore, all durations considered in the Gantt chart are minimal, i.e., all activities are as short as possible and, consequently, it represents the fastest processing scheme of a single batch.

Looking at Fig. 6.2, it becomes clear that the (discrete-event) model of a single batch is subject to synchronization phenomena but devoid of any conflicts or decisions. Furthermore, minimal and maximal duration times play an important role. Consequently, it is possible to convert the graph model into an (extended) timed event graph with time window constraints. The corresponding TEG of our example is given in Fig. 6.3. In this TEG, transitions x_1 and x_3 correspond to the nodes o_1 and r_1 of the graph model in Fig. 6.2. Similarly, transitions x_4 and x_7, x_8 and x_{11}, and x_{12} and x_{14} correspond to nodes o_2 and r_2, o_3 and r_3, and o_4 and r_4, respectively. The minimal and maximal duration time between nodes a and b in the graph model is represented by the time window constraint between transition x_5 and x_6. Consequently, the TEG in Fig. 6.3 represents the behavior given by the graph model in Fig. 6.3. As described in the previous section, extended timed event graphs can easily be converted into linear models in idempotent semirings, e.g., $\mathcal{M}_{\text{in}}^{\text{ax}}[\![\gamma, \delta]\!]$. Consequently, it is possible to determine a linear model of a single batch in $\mathcal{M}_{\text{in}}^{\text{ax}}[\![\gamma, \delta]\!]$.

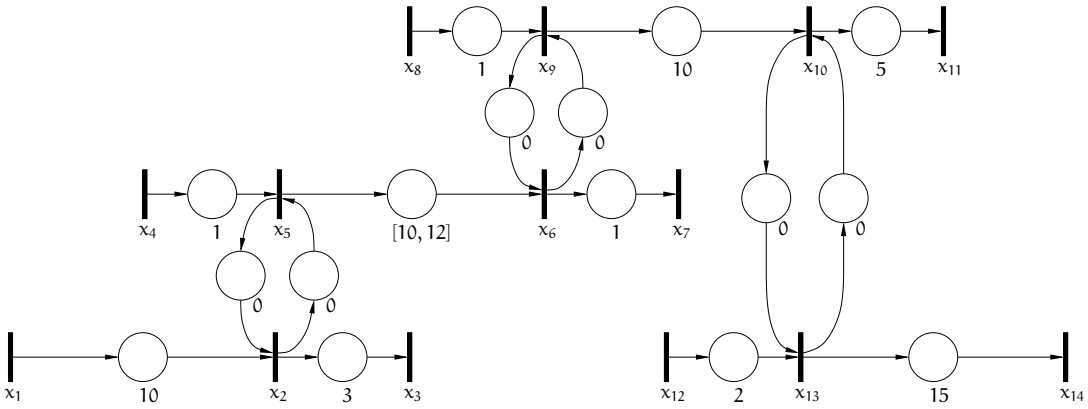


Figure 6.3: TEG of the single batch.

However, a single batch does only define the processing steps for one set of compounds. To be able to compare different compound sets, many batches have to be processed. Since the operation of HTS plants is rather costly, a scheduling problem to find the optimal schedule with the highest possible throughput has to be solved. Due to the fact that a single batch may be processed more than once on the same resource, the resulting optimal schedule may be nested, which increases the complexity of the scheduling problem. One possible scheduling algorithm to determine globally optimal schedules for HTS processes has been developed in [43]. In this approach, the graph model in Fig. 6.2 and the resource allocation of every activity is formulated as a mixed integer non-linear optimization problem (MINLP). Applying a certain transformation, this MINLP can be rewritten as a mixed integer linear optimization problem (MILP). To solve large MILPs is computational expensive and time consuming. Therefore, the optimal schedule is calculated prior to the beginning of the HTS run. Consequently, the resulting schedule is a static one, which is not meant to be altered during the course of a screening run.

For our example, the globally optimal schedule is shown as a Gantt chart in Fig. 6.4. In the globally optimal schedule with the highest possible throughput, the time necessary to process one single batch has increased compared to the optimal single batch given in the lower part of Fig. 6.2. However, by delaying and extending some of the activities it is possible to "squeeze" activities of batches in between activities of other batches executed on the same resource, i.e., the activities on some resources may be nested.

In general, the globally optimal schedule with the highest possible throughput can be described by the exact timing of one single batch and the corresponding cycle time Θ, i.e., every Θ time units a new batch has to be started. Consequently, this also means that, once the cyclic regime has been reached, i.e., after a start-up phase, one batch will be finished every Θ time units. As it can be easily observed in Fig. 6.4, the optimal schedule of our example has a cycle

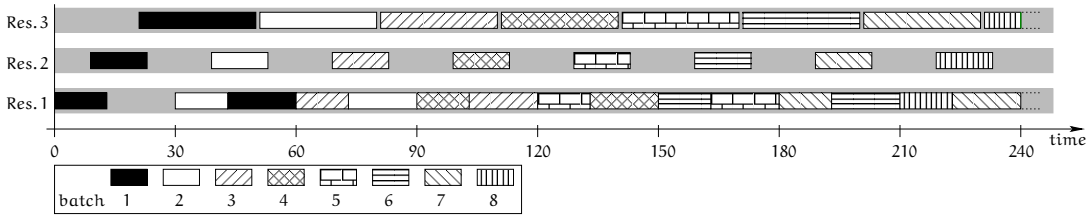


Figure 6.4: Gantt chart of the optimal schedule including the start-up phase.

time of 30 time units and the exact timing of the first batch is

$$\tau = \begin{pmatrix} 0 & 10 & 13 & 9 & 10 & 22 & 23 & 21 & 22 & 45 & 50 & 43 & 45 & 60 \end{pmatrix}^T \tag{6.1}$$

where the i^{th} element of vector τ corresponds to the time of the occurrence of event x_i. Using this vector it is easy to determine that 12 time units pass between the occurrence of events x_5 and x_6 and, therefore, the constraint of the time window is fulfilled.

At this point it is necessary to clarify that our approach does not provide a solution to the original mixed integer linear optimization problem. Our approach rather uses the solution of the MILP to determine an optimal feedback strategy, which can respond to possible disturbances during runtime. More precisely, to determine an optimal feedback strategy, it is necessary to have the information on the optimal sequence of activities for every resource. Furthermore, the solution of the MILP also provides information on the exact timing of a single batch and the corresponding optimal cycle time. This information, however, is not necessary for our approach. Basically, the timing information needed is already included in the TEG of the single batch (see Fig. 6.3). Even though the TEG contains the minimal durations and time windows instead of the exact timing of a single batch in the optimal schedule, the combination of this TEG with the sequence of the activities executed on every resource in the optimal schedule is sufficient to specify an optimal schedule. Using our example, we will describe our approach in more detail. In Fig. 6.5 a section of the globally optimal schedule of our example is given. The numbers in this figure indicate the different activities, e.g., for batch k activities 1, 2, 3, and 4 are displayed. Note that the optimal schedule of our example is strictly 1-cyclic, i.e., every event of every activity occurs exactly once every $\Theta = 30$ time units. Taking a look at the sequence of activities executed on every resource, it is clear that activity 3 of batch k, executed on $\mathrm{Res.}\,3$, is preceded by activity 3 of batch $k-1$ and is followed by activity 3 of batch $k+1$. The same behavior can be observed for activity 2, which is executed on $\mathrm{Res.}\,2$. Such a "sequence" of activities, i.e., an activity is followed by the same activity of the succeeding batch, is the standard behavior if the corresponding resource processes a single activity for every batch. Interesting is the sequence of activities executed on $\mathrm{Res.}\,1$. Taking a look at Fig. 6.5 one can observe, that activity 1 of batch k is preceded by activity 4 of batch $k-2$ and succeeded by activity 4 of batch $k-1$. Consequently, we can also say that activity 4 of batch

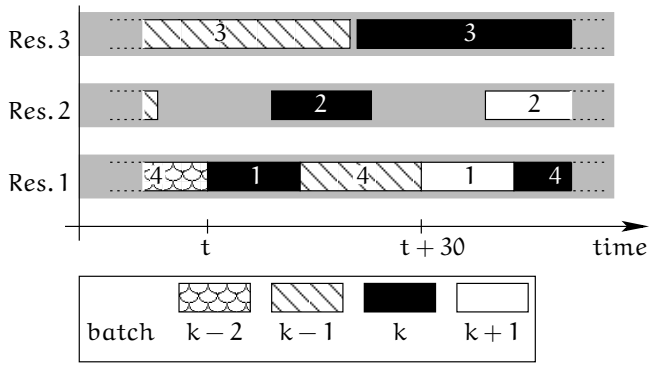


Figure 6.5: Section of the Gantt chart describing the optimal schedule.

$k-1$ is succeeded by activity 1 of batch $k+1$. Summarizing we get the following constraints for the sequence of activities on Res. 1.

(i) Activity 1 of batch k may start after activity 4 of batch $k-2$ has been finished.

(ii) Once activity 1 of batch k has been finished, activity 4 of batch $k-1$ may start.

Considering the start and finish events of every activity given in the TEG in Fig. 6.3, these constraints can be rewritten as constraints in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$.

(i) $x_1 \geq \gamma^2 x_{14}$

(ii) $x_{12} \preceq \gamma x_3$

Similarly, the constraints (in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$) for the sequences of activities executed on Res. 2 and Res. 3 are

$$x_4 \geq \gamma x_7 \qquad \text{(Res. 2)}$$
$$x_8 \geq \gamma x_{11}. \qquad \text{(Res. 3)}$$

Finally, the information of the TEG of a single batch depicted in Fig. 6.3 and the constraints

on the sequence of activities can be merged to a model in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$. For our example we get

$$\underline{A} = \begin{pmatrix}
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma^2 \\
\delta^{10} & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \delta^3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & e & \varepsilon & \delta^1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta^{10} & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta^1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & \delta^1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta^{10} & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta^5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & \delta^2 & \varepsilon & \varepsilon \\
\varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta^{15} & \varepsilon
\end{pmatrix},$$

$$\overline{A} = \begin{pmatrix}
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \gamma & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \delta^{12} & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top \\
\top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top & \top
\end{pmatrix}.$$

Then, matrix $\overline{\underline{A}}^{*} = (\overline{A}_{*} \backslash\!\!\bullet\, \underline{A}^{*})^{*}$ can be determined.

$$
\overline{\mathrm{A}}^{*} = \begin{pmatrix}
(e)\,(\gamma\delta^{30})^{*} & (\gamma\delta^{20})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{17})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{21})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{20})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{8})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{21})\,(\gamma\delta^{30})^{*} \\
(\delta^{10})\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*} & \delta^{-4} \oplus (\gamma\delta^{27})\,(\gamma\delta^{30})^{*} & (\delta)\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*} & (\delta^{-12})\,(\gamma\delta^{30})^{*} & (\gamma\delta)\,(\gamma\delta^{30})^{*} \\
(\delta^{13})\,(\gamma\delta^{30})^{*} & (\delta^{3})\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*} & (\delta^{4})\,(\gamma\delta^{30})^{*} & (\delta^{3})\,(\gamma\delta^{30})^{*} & (\delta^{-9})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{4})\,(\gamma\delta^{30})^{*} \\
(\gamma\delta^{22})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{12})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{9})\,(\gamma\delta^{30})^{*} & e \oplus (\gamma\delta^{13})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{12})\,(\gamma\delta^{30})^{*} & \gamma\delta \oplus (\gamma^{2}\delta^{30})\,(\gamma\delta^{30})^{*} & \gamma \oplus (\gamma^{2}\delta^{13})\,(\gamma\delta^{30})^{*} \\
(\delta^{10})\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*} & \delta^{-4} \oplus (\gamma\delta^{27})\,(\gamma\delta^{30})^{*} & (\delta)\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*} & (\delta^{-12})\,(\gamma\delta^{30})^{*} & (\gamma\delta)\,(\gamma\delta^{30})^{*} \\
(\delta^{21})\,(\gamma\delta^{30})^{*} & (\delta^{11})\,(\gamma\delta^{30})^{*} & (\delta^{8})\,(\gamma\delta^{30})^{*} & (\delta^{12})\,(\gamma\delta^{30})^{*} & (\delta^{11})\,(\gamma\delta^{30})^{*} & e \oplus (\gamma\delta^{29})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{12})\,(\gamma\delta^{30})^{*} \\
(\delta^{22})\,(\gamma\delta^{30})^{*} & (\delta^{12})\,(\gamma\delta^{30})^{*} & (\delta^{9})\,(\gamma\delta^{30})^{*} & (\delta^{13})\,(\gamma\delta^{30})^{*} & (\delta^{12})\,(\gamma\delta^{30})^{*} & \delta \oplus (\gamma\delta^{30})\,(\gamma\delta^{30})^{*} & e \oplus (\gamma\delta^{13})\,(\gamma\delta^{30})^{*} \\
(\delta^{20})\,(\gamma\delta^{30})^{*} & (\delta^{10})\,(\gamma\delta^{30})^{*} & (\delta^{7})\,(\gamma\delta^{30})^{*} & (\delta^{11})\,(\gamma\delta^{30})^{*} & (\delta^{10})\,(\gamma\delta^{30})^{*} & (\delta^{-2})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{11})\,(\gamma\delta^{30})^{*} \\
(\delta^{21})\,(\gamma\delta^{30})^{*} & (\delta^{11})\,(\gamma\delta^{30})^{*} & (\delta^{8})\,(\gamma\delta^{30})^{*} & (\delta^{12})\,(\gamma\delta^{30})^{*} & (\delta^{11})\,(\gamma\delta^{30})^{*} & e \oplus (\gamma\delta^{29})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{12})\,(\gamma\delta^{30})^{*} \\
(\gamma^{-1}\delta^{15})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{5})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{2})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{6})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{5})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{-7})\,(\gamma\delta^{30})^{*} & (\delta^{6})\,(\gamma\delta^{30})^{*} \\
(\gamma^{-1}\delta^{20})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{10})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{7})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{11})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{10})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{-2})\,(\gamma\delta^{30})^{*} & (\delta^{11})\,(\gamma\delta^{30})^{*} \\
(\gamma^{-1}\delta^{13})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{3})\,(\gamma\delta^{30})^{*} & (\gamma^{-1})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{4})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{3})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{-9})\,(\gamma\delta^{30})^{*} & (\delta^{4})\,(\gamma\delta^{30})^{*} \\
(\gamma^{-1}\delta^{15})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{5})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{2})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{6})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{5})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{-7})\,(\gamma\delta^{30})^{*} & (\delta^{6})\,(\gamma\delta^{30})^{*} \\
(\gamma^{-1}\delta^{30})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{20})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{17})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{21})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{20})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{8})\,(\gamma\delta^{30})^{*} & (\delta^{21})\,(\gamma\delta^{30})^{*}
\end{pmatrix} \cdots
$$

$$
\cdots \begin{pmatrix}
(\gamma\delta^{9})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{8})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{15})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{9})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{17})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{15})\,(\gamma\delta^{30})^{*} & (\gamma^{2})\,(\gamma\delta^{30})^{*} \\
(\delta^{-11})\,(\gamma\delta^{30})^{*} & (\delta^{-12})\,(\gamma\delta^{30})^{*} & \gamma\delta^{-6} \oplus (\gamma^{2}\delta^{25})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{-11})\,(\gamma\delta^{30})^{*} & \gamma\delta^{-4} \oplus (\gamma^{2}\delta^{27})\,(\gamma\delta^{30})^{*} & \gamma\delta^{-6} \oplus (\gamma^{2}\delta^{25})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{10})\,(\gamma\delta^{30})^{*} \\
(\delta^{-8})\,(\gamma\delta^{30})^{*} & (\delta^{-9})\,(\gamma\delta^{30})^{*} & \gamma\delta^{-3} \oplus (\gamma^{2}\delta^{28})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{-8})\,(\gamma\delta^{30})^{*} & \gamma\delta^{-1} \oplus (\gamma^{2}\delta^{30})\,(\gamma\delta^{30})^{*} & \gamma\delta^{-3} \oplus (\gamma^{2}\delta^{28})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{13})\,(\gamma\delta^{30})^{*} \\
\gamma\delta^{2} \oplus (\gamma^{2}\delta^{31})\,(\gamma\delta^{30})^{*} & \gamma\delta \oplus (\gamma^{2}\delta^{30})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{7})\,(\gamma\delta^{30})^{*} & \gamma^{2}\delta^{2} \oplus (\gamma^{3}\delta^{31})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{9})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{7})\,(\gamma\delta^{30})^{*} & (\gamma^{3}\delta^{22})\,(\gamma\delta^{30})^{*} \\
(\delta^{-11})\,(\gamma\delta^{30})^{*} & (\delta^{-12})\,(\gamma\delta^{30})^{*} & \gamma\delta^{-6} \oplus (\gamma^{2}\delta^{25})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{-11})\,(\gamma\delta^{30})^{*} & \gamma\delta^{-4} \oplus (\gamma^{2}\delta^{27})\,(\gamma\delta^{30})^{*} & \gamma\delta^{-6} \oplus (\gamma^{2}\delta^{25})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{10})\,(\gamma\delta^{30})^{*} \\
\delta \oplus (\gamma\delta^{30})\,(\gamma\delta^{30})^{*} & e \oplus (\gamma\delta^{29})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{6})\,(\gamma\delta^{30})^{*} & \gamma\delta \oplus (\gamma^{2}\delta^{30})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{8})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{6})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{21})\,(\gamma\delta^{30})^{*} \\
\delta^{2} \oplus (\gamma\delta^{31})\,(\gamma\delta^{30})^{*} & \delta \oplus (\gamma\delta^{30})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{7})\,(\gamma\delta^{30})^{*} & \gamma\delta^{2} \oplus (\gamma^{2}\delta^{31})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{9})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{7})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{22})\,(\gamma\delta^{30})^{*} \\
e \oplus (\gamma\delta^{29})\,(\gamma\delta^{30})^{*} & (\delta^{-2})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{5})\,(\gamma\delta^{30})^{*} & \gamma \oplus (\gamma^{2}\delta^{29})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{7})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{5})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{20})\,(\gamma\delta^{30})^{*} \\
\delta \oplus (\gamma\delta^{30})\,(\gamma\delta^{30})^{*} & e \oplus (\gamma\delta^{29})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{6})\,(\gamma\delta^{30})^{*} & \gamma\delta \oplus (\gamma^{2}\delta^{30})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{8})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{6})\,(\gamma\delta^{30})^{*} & (\gamma^{2}\delta^{21})\,(\gamma\delta^{30})^{*} \\
(\gamma^{-1}\delta^{-6})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{-7})\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*} & (\delta^{-6})\,(\gamma\delta^{30})^{*} & (\delta^{2})\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*} & (\gamma\delta^{15})\,(\gamma\delta^{30})^{*} \\
(\gamma^{-1}\delta^{-1})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{-2})\,(\gamma\delta^{30})^{*} & (\delta^{5})\,(\gamma\delta^{30})^{*} & e \oplus (\gamma\delta^{29})\,(\gamma\delta^{30})^{*} & (\delta^{7})\,(\gamma\delta^{30})^{*} & (\delta^{5})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{20})\,(\gamma\delta^{30})^{*} \\
(\gamma^{-1}\delta^{-8})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{-9})\,(\gamma\delta^{30})^{*} & \delta^{-3} \oplus (\gamma\delta^{28})\,(\gamma\delta^{30})^{*} & (\delta^{-8})\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*} & \delta^{-3} \oplus (\gamma\delta^{28})\,(\gamma\delta^{30})^{*} & (\gamma\delta^{13})\,(\gamma\delta^{30})^{*} \\
(\gamma^{-1}\delta^{-6})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{-7})\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*} & (\delta^{-6})\,(\gamma\delta^{30})^{*} & (\delta^{2})\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*} & (\gamma\delta^{15})\,(\gamma\delta^{30})^{*} \\
(\gamma^{-1}\delta^{9})\,(\gamma\delta^{30})^{*} & (\gamma^{-1}\delta^{8})\,(\gamma\delta^{30})^{*} & (\delta^{15})\,(\gamma\delta^{30})^{*} & (\delta^{9})\,(\gamma\delta^{30})^{*} & (\delta^{17})\,(\gamma\delta^{30})^{*} & (\delta^{15})\,(\gamma\delta^{30})^{*} & (e)\,(\gamma\delta^{30})^{*}
\end{pmatrix}.
$$

Given an input and an output matrix

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

$$C = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e \end{pmatrix}$$

the behavior of the system in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ can be computed. Note that the start event of every activity is an input, i.e., the start of every activity can be delayed, and the output is the last event of a single batch. Since matrix $\underline{\overline{A}}^*$ is the system matrix which contains all constraints on the system, the corresponding fastest output y_{max} of the system, i.e., the output when the system operates with maximal throughput, can then be determined by

$$y_{max} = C\underline{\overline{A}}^* B u_{min},$$

where u_{min} corresponds to an instantaneous firing of all inputs, i.e., $u_{min} = e$. Once y_{max} has been determined, the input of the just-in-time behavior can be computed by

$$u_{jit} = (C\underline{\overline{A}}^* B) \backslash\!\!\backslash y_{max},$$

and the just-in-time behavior with respect to the system state x is

$$x_{jit} = \underline{\overline{A}}^* B u_{jit}.$$

For our example we get

$$\begin{aligned} y_{max} &= \gamma^{-1}\delta^{30}(\gamma\delta^{30})^* \\ u_{jit} &= \begin{pmatrix} (\gamma\delta^{30})^* & \delta^{9}(\gamma\delta^{30})^* & \delta^{21}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{13}(\gamma\delta^{30})^* \end{pmatrix}^T \\ x_{jit} &= \begin{pmatrix} e & \delta^{10} & \delta^{13} & \delta^{9} & \delta^{10} & \delta^{22} & \delta^{23} & \delta^{21} & \delta^{22} & \dots \\ & \dots & \gamma^{-1}\delta^{15} & \gamma^{-1}\delta^{20} & \gamma^{-1}\delta^{13} & \gamma^{-1}\delta^{15} & \gamma^{-1}\delta^{30} \end{pmatrix}^T (\gamma\delta^{30})^*, \end{aligned}$$

and the corresponding causal projections are

$$\begin{aligned} y_{max}^{\ddagger} &= \delta^{60}(\gamma\delta^{30})^* \\ u_{jit}^{\ddagger} &= Pr_{caus}^{\ddagger}(u_{jit}) \\ &= \begin{pmatrix} (\gamma\delta^{30})^* & \delta^{9}(\gamma\delta^{30})^* & \delta^{21}(\gamma\delta^{30})^* & \delta^{43}(\gamma\delta^{30})^* \end{pmatrix}^T \\ x_{jit}^{\ddagger} &= \begin{pmatrix} e & \delta^{10} & \delta^{13} & \delta^{9} & \delta^{10} & \delta^{22} & \delta^{23} & \delta^{21} & \delta^{22} & \dots \\ & \dots & \delta^{45} & \delta^{50} & \delta^{43} & \delta^{45} & \delta^{60} \end{pmatrix}^T (\gamma\delta^{30})^*. \end{aligned}$$

Clearly, the determined just-in-time behavior x_{jit} is equivalent to the information on the optimal schedule given by the optimization algorithm. More precisely, the element $(\gamma\delta^{30})^*$ in x_{jit} describes the throughput and, therefore, the cycle time of the system, i.e., every event occurs once every 30 time units, and the "vector" in x_{jit} represents the timing of the occurrence of any event in the first batch given by τ in (6.1). Consequently, our model in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ describes the optimal operation of the underlying HTS system.

6.3 Control of HTS systems

Once the globally optimal schedule of a specific HTS process has been determined the resulting optimal operation can be modeled in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$. This model includes information on the single batch as well as on the optimal sequence of activities of the schedule with the highest possible throughput. As described in Sec. 5.2 it is possible to determine an optimal feedback controller in combination with an optimal pre-filter. The pre-filter and feedback controller are able to react in case of unforeseen disturbances and to alter the timing of future events during the course of a screening run.

Formally, the pre-filter and feedback controller are determined as described in 5.2, i.e., we search for a pre-filter and controller such that the dependencies

$$\begin{aligned} C(\underline{A} \oplus BF)^* BP &\preceq G_{ref} \\ (\underline{A} \oplus BF)^* BP &= \overline{\underline{A}}^* (\underline{A} \oplus BF)^* BP \end{aligned}$$

are fulfilled for a given reference model G_{ref}. Consequently, the control problem of an HTS operation is a control problem of extended timed event graphs. The optimal pre-filter P_{opt} can be determined by a fixed point algorithm and the optimal state feedback controller is

$$F_{opt} = Pr_{caus}^{\top}(F_{max}) = Pr_{caus}^{\top}((\underline{A}^* B) \wr (\underline{A}^* B P_{opt}) \not\wr (\underline{A}^* B P_{opt}))\,.$$

Given the reference model $G_{ref} = C\overline{\underline{A}}^* B$, i.e., the system shall operate with the highest possible throughput, the optimal pre-filter can be determined as

$$P_{opt} = \begin{pmatrix} (\gamma\delta^{30})^* & \gamma\delta^{21}(\gamma\delta^{30})^* & \gamma\delta^{9}(\gamma\delta^{30})^* & \gamma^2\delta^{17}(\gamma\delta^{30})^* \\ \delta^{9}(\gamma\delta^{30})^* & (\gamma\delta^{30})^* & \gamma\delta^{18}(\gamma\delta^{30})^* & \gamma^2\delta^{26}(\gamma\delta^{30})^* \\ \delta^{21}(\gamma\delta^{30})^* & \delta^{12}(\gamma\delta^{30})^* & \gamma\delta^{30}(\gamma\delta^{30})^* & \gamma^2\delta^{38}(\gamma\delta^{30})^* \\ \gamma^{-1}\delta^{13}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{4}(\gamma\delta^{30})^* & \delta^{22}(\gamma\delta^{30})^* & \gamma\delta^{30}(\gamma\delta^{30})^* \end{pmatrix}.$$

The corresponding maximal feedback controller F_{max} and its causal projection F_{opt} are

$$F_{max} = \begin{pmatrix} (\gamma\delta^{30})^* & \delta^{9}(\gamma\delta^{30})^* & \delta^{21}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{13}(\gamma\delta^{30})^* \\ \gamma\delta^{20}(\gamma\delta^{30})^* & \delta^{-1}(\gamma\delta^{30})^* & \delta^{11}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{3}(\gamma\delta^{30})^* \\ \gamma\delta^{17}(\gamma\delta^{30})^* & \delta^{-4}(\gamma\delta^{30})^* & \delta^{8}(\gamma\delta^{30})^* & \gamma^{-1}(\gamma\delta^{30})^* \\ \gamma\delta^{21}(\gamma\delta^{30})^* & (\gamma\delta^{30})^* & \delta^{12}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{4}(\gamma\delta^{30})^* \\ \gamma\delta^{20}(\gamma\delta^{30})^* & \delta^{-1}(\gamma\delta^{30})^* & \delta^{11}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{3}(\gamma\delta^{30})^* \\ \gamma\delta^{8}(\gamma\delta^{30})^* & \delta^{-13}(\gamma\delta^{30})^* & \delta^{-1}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{-9}(\gamma\delta^{30})^* \\ \gamma\delta^{7}(\gamma\delta^{30})^* & \delta^{-14}(\gamma\delta^{30})^* & \delta^{-2}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{-10}(\gamma\delta^{30})^* \\ \gamma\delta^{9}(\gamma\delta^{30})^* & \delta^{-12}(\gamma\delta^{30})^* & (\gamma\delta^{30})^* & \gamma^{-1}\delta^{-8}(\gamma\delta^{30})^* \\ \gamma\delta^{8}(\gamma\delta^{30})^* & \delta^{-13}(\gamma\delta^{30})^* & \delta^{-1}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{-9}(\gamma\delta^{30})^* \\ \gamma^2\delta^{15}(\gamma\delta^{30})^* & \gamma\delta^{-6}(\gamma\delta^{30})^* & \gamma\delta^{6}(\gamma\delta^{30})^* & \delta^{-2}(\gamma\delta^{30})^* \\ \gamma^2\delta^{10}(\gamma\delta^{30})^* & \gamma\delta^{-11}(\gamma\delta^{30})^* & \gamma\delta(\gamma\delta^{30})^* & \delta^{-7}(\gamma\delta^{30})^* \\ \gamma^2\delta^{17}(\gamma\delta^{30})^* & \gamma\delta^{-4}(\gamma\delta^{30})^* & \gamma\delta^{8}(\gamma\delta^{30})^* & (\gamma\delta^{30})^* \\ \gamma^2\delta^{15}(\gamma\delta^{30})^* & \gamma\delta^{-6}(\gamma\delta^{30})^* & \gamma\delta^{6}(\gamma\delta^{30})^* & \delta^{-2}(\gamma\delta^{30})^* \\ \gamma^2(\gamma\delta^{30})^* & \gamma\delta^{-21}(\gamma\delta^{30})^* & \gamma\delta^{-9}(\gamma\delta^{30})^* & \delta^{-17}(\gamma\delta^{30})^* \end{pmatrix}^{T}$$

and

$$F_{opt} = \begin{pmatrix} (\gamma\delta^{30})^* & \delta^{9}(\gamma\delta^{30})^* & \delta^{21}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{13}(\gamma\delta^{30})^* \\ \gamma\delta^{20}(\gamma\delta^{30})^* & \gamma\delta^{29}(\gamma\delta^{30})^* & \delta^{11}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{3}(\gamma\delta^{30})^* \\ \gamma\delta^{17}(\gamma\delta^{30})^* & \gamma\delta^{26}(\gamma\delta^{30})^* & \delta^{8}(\gamma\delta^{30})^* & \gamma^{-1}(\gamma\delta^{30})^* \\ \gamma\delta^{21}(\gamma\delta^{30})^* & (\gamma\delta^{30})^* & \delta^{12}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{4}(\gamma\delta^{30})^* \\ \gamma\delta^{20}(\gamma\delta^{30})^* & \gamma\delta^{29}(\gamma\delta^{30})^* & \delta^{11}(\gamma\delta^{30})^* & \gamma^{-1}\delta^{3}(\gamma\delta^{30})^* \\ \gamma\delta^{8}(\gamma\delta^{30})^* & \gamma\delta^{17}(\gamma\delta^{30})^* & \gamma\delta^{29}(\gamma\delta^{30})^* & \delta^{21}(\gamma\delta^{30})^* \\ \gamma\delta^{7}(\gamma\delta^{30})^* & \gamma\delta^{16}(\gamma\delta^{30})^* & \gamma\delta^{28}(\gamma\delta^{30})^* & \delta^{20}(\gamma\delta^{30})^* \\ \gamma\delta^{9}(\gamma\delta^{30})^* & \gamma\delta^{18}(\gamma\delta^{30})^* & (\gamma\delta^{30})^* & \delta^{22}(\gamma\delta^{30})^* \\ \gamma\delta^{8}(\gamma\delta^{30})^* & \gamma\delta^{17}(\gamma\delta^{30})^* & \gamma\delta^{29}(\gamma\delta^{30})^* & \delta^{21}(\gamma\delta^{30})^* \\ \gamma^2\delta^{15}(\gamma\delta^{30})^* & \gamma^2\delta^{24}(\gamma\delta^{30})^* & \gamma\delta^{6}(\gamma\delta^{30})^* & \gamma\delta^{28}(\gamma\delta^{30})^* \\ \gamma^2\delta^{10}(\gamma\delta^{30})^* & \gamma^2\delta^{19}(\gamma\delta^{30})^* & \gamma\delta(\gamma\delta^{30})^* & \gamma\delta^{23}(\gamma\delta^{30})^* \\ \gamma^2\delta^{17}(\gamma\delta^{30})^* & \gamma^2\delta^{26}(\gamma\delta^{30})^* & \gamma\delta^{8}(\gamma\delta^{30})^* & (\gamma\delta^{30})^* \\ \gamma^2\delta^{15}(\gamma\delta^{30})^* & \gamma^2\delta^{24}(\gamma\delta^{30})^* & \gamma\delta^{6}(\gamma\delta^{30})^* & \gamma\delta^{28}(\gamma\delta^{30})^* \\ \gamma^2(\gamma\delta^{30})^* & \gamma^2\delta^{9}(\gamma\delta^{30})^* & \gamma^2\delta^{21}(\gamma\delta^{30})^* & \gamma\delta^{13}(\gamma\delta^{30})^* \end{pmatrix}^{T}.$$

By design, the controlled system operates at the highest possible throughput. If an unforeseen disturbance, with respect to the optimal throughput regime, occurs, the feedback controller delays all future events as much as possible without reducing the throughput more than

it is already reduced by the disturbance itself. Therefore, the controller guarantees that all future batches are started just in time after a possible disturbance, which may reduce the number of batches affected by the disturbance. More precisely, if a disturbance occurs, the time scheme of some batches may differ from the pre-defined optimal single batch time scheme. Consequently, these batches cannot be considered for the screening result as every single batch has to follow the exact same time scheme. By starting single batches just in time (after a disturbance), it is possible to return to the pre-defined time scheme more quickly than in the uncontrolled case, and thus, it is possible to reduce the number of "waste batches". Formally, in case of an unforeseen disturbance, the following statements hold:

$$\begin{array}{rcl} \text{time to return to the optimal throughput regime with control} & \leqslant & \text{time to return to the optimal throughput regime without control} \\ \text{number of waste batches with control} & \leqslant & \text{number of waste batches without control} \end{array}$$

6.4 Example of a real HTS operation

The example of an HTS operation introduced in the previous section is relatively small compared to real HTS operations. Nonetheless, it is hardly to display the 14×14 matrix $\overline{A}^*$, as every entry of this matrix is a formal power series in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$. Real world HTS systems differ in their size depending on the specific screening problem. One (not necessarily large) single batch of a real world HTS operation, provided by CyBio AG [1], a company developing and producing HTS plants, consists of 22 activities which are executed on 7 different resources. The definition of a single batch contains 140 events, i.e., the corresponding system matrix $\overline{A}^* \in \mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ is of dimension 140×140. The graph model of this single batch is given in Fig. 6.6 and the Gantt chart of the associated globally optimal schedule is given in Fig. 6.7. The cycle time of the optimal schedule is 324 time units, i.e., every 324 time units a new batch enters the system, while the processing time of a single batch is 754 time units. Clearly, it is not possible to display the corresponding matrices $\underline{A}$, and $\overline{\underline{A}}^*$ here. The only elements of matrix $\overline{A}$ different from $\top$ are

$$[\underline{A}]_{5,136} = \gamma^2 \qquad [\underline{A}]_{33,130} = \gamma^2 \qquad [\underline{A}]_{97,118} = \gamma$$
$$[\underline{A}]_{81,76} = \delta^{17} \qquad [\underline{A}]_{59,54} = \delta^9.$$

Thus, there are three pairs of transitions with a minimal number of tokens in between and two time windows. Given a reference model, e.g., $G_{ref} = C\overline{\underline{A}}^* B$ (of dimension 1×22), the optimal pre-filter P_{opt} and state feedback controller F_{opt} have dimensions 22×22 and 22×140, respectively. Consequently, also P_{opt} and F_{opt} are too large to be displayed. In the

1. http://www.cybio-ag.com

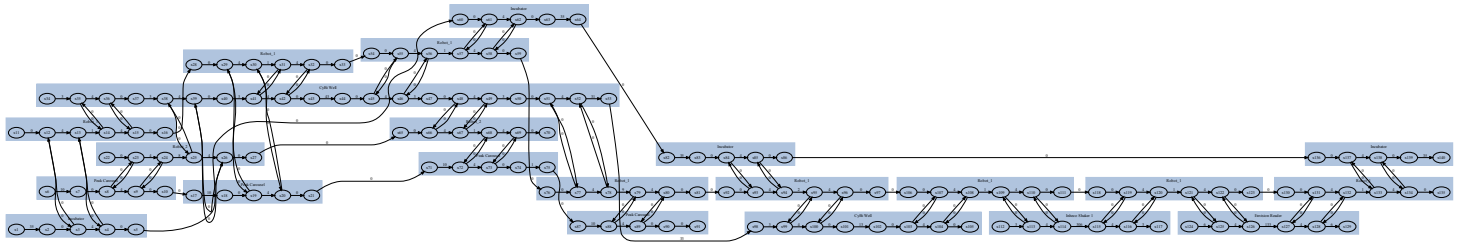


Figure 6.6: Graph model of a single batch of a real world HTS operation (provided by CyBio AG).

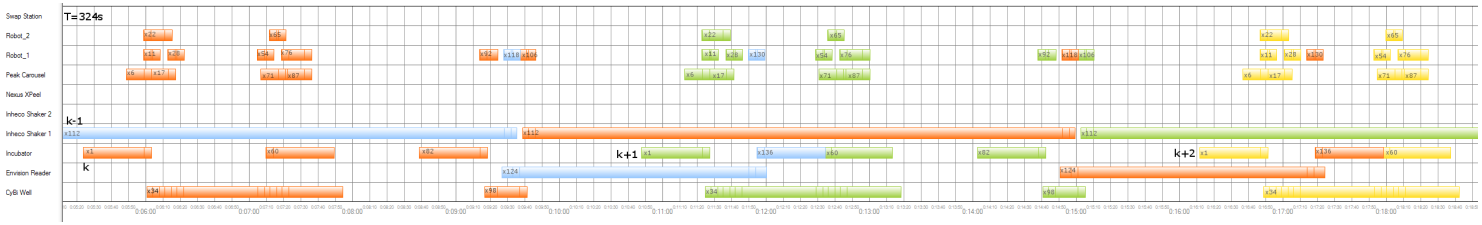


Figure 6.7: Gantt chart of the real world HTS operation with the optimal schedule (provided by CyBio AG). Activities belonging to the same single batch are indicated by the same color.

disturbance-free case the optimal (pre-filtered and feedback controlled) input $u^{\ddagger}_{\mathrm{opt}}$ is

$$u^{\ddagger}_{\mathrm{opt}} = \Big(\; e \quad \delta^{106} \quad \delta^{195} \quad \delta^{715} \quad \delta^{35} \quad \delta^{49} \quad \delta^{101} \quad \delta^{115} \quad \delta^{230} \quad \delta^{253} \quad \delta^{568} \quad \delta^{710} \quad \dots$$
$$\dots \quad \delta^{35} \quad \delta^{108} \quad \delta^{37} \quad \delta^{233} \quad \delta^{25} \quad \delta^{39} \quad \delta^{103} \quad \delta^{118} \quad \delta^{255} \quad \delta^{567} \; \Big) \, (\gamma\delta^{324})^{*}.$$

Clearly, the controlled system is not slowed down, i.e., the throughput of the system is still one batch every 324 time units. Furthermore, the input is delayed as much as possible while maintaining this throughput. If an unexpected disturbance occurs during run-time, the feedback controller F_{opt} will take this disturbance into account and, if necessary, change the input $u^{\ddagger}_{\mathrm{opt}}$.

According to CyBio AG, a "normal" HTS example usually involves up to 250 events. However, recently, "new" HTS techniques were developed, resulting in single batches including more than 1000 events. Consequently, the corresponding system description increases significantly. Of course, our approach is also valid for systems of this size. Nonetheless, computing an optimal pre-filter and feedback controller for systems of that size may be quite time consuming and computationally complex. Especially, the computation of the Kleene stars of $\overline{A}$ and $\underline{A}^{*}$ and the dual Kleene star of matrix $\underline{A}$ is time consuming. The computation of the optimal pre-filter P_{opt} and feedback controller F_{opt} of the real world HTS operation given in Fig. 6.7 was performed using the C++ library MinmaxGD [18] equipped with additional algorithms for the dual Kleene star, the dual left residuation, and for the causal projection for transfer series, i.e., $\mathrm{Pr}^{\mp}_{\mathrm{caus}}$. The computer used was running the Ubuntu 10.04 (lucid) operating system on a Intel® Core™ 2 Duo E8400 3.00 GHz processor. The machine was equipped with 2 GB of RAM and the overall computation time for the pre-filter and feedback controller took about 60 minutes.

The issue of computation time and computational complexity is not crucial as these computations can be performed off-line, i.e., prior to the screening run. Nonetheless, it may be necessary to implement highly efficient algorithms for the computation of the Kleene star to be able to cope with systems of more than 1000 events in reasonable time. These algorithms could, for example, take the sparse character of the matrices $\overline{A}$ and $\underline{A}$ into account.

7 Conclusion

This work proposes an extension to linear systems and control theory in dioid algebras. Using the established (standard) systems theory in dioids, it is possible to model a large class of (discrete-event) systems, which is characterized by synchronization phenomena but devoid of any decisions. Belonging to this class are, for example, systems from manufacturing industries, communication technologies, and traffic networks. In general, the considered systems which have a linear representation in dioids can be represented as timed event graphs, a class of Petri nets. In such timed event graphs the timing information is usually considered to be the minimal time, i.e., the least time necessary to execute a specific (part of a) process or operation.

Once a dioid model of a system of interest has been derived, a well-established control theory is available to compute various strategies, i.e., controllers, to achieve a desired system behavior. Among the developed strategies are approaches for optimal feedforward control, i.e., optimal input filtering [20, 26], optimal feedforward control based on a desired reference output [2, 48], and optimal feedback control, i.e., state and output feedback [17, 19, 20], disturbance decoupling and robust control in case of uncertainties [39, 40, 41], and model predictive control [58, 61]. Recently, also concepts for observers for partially observable (linear) systems in dioids have been investigated [28, 29]. It is important to mention that control in this framework is restricted to delaying the system, as the timing information given in the model is the fastest possible behavior of the uncontrolled system. Consequently, possible controllers cannot speed up the system but only delay specific events. This, however is sufficient to design controllers which guarantee a just-in-time behavior, i.e., everything is started as late as possible without delaying a pre-defined desired output. In case of feedback strategies, the controlled system is "robust" to

unforeseen disturbances, i.e., if an unforeseen disturbance delays the desired output, everything is delayed as much as possible to achieve the disturbed output.

As mentioned before, the timing information in timed event graphs and consequently also in their linear representation in dioids is usually the least time to perform the corresponding modeled task. For many systems these minimal times are sufficient to determine a corresponding model of their behavior. Sometimes, however, it is crucial to include not only minimal times of processes or between processes but also maximal times to correctly represent specifications. This is, for example, the case, for many systems from chemical or bio-chemical industries, where reactions may not exceed a maximal reaction time.

Furthermore, while it is easily possible to model the maximal number of entities within a (part of a) process, it is not straightforward to model processes which require a minimal number of parts to be present during operation. Considering, for example, a manufacturing system with a minimal stock of one part between process A and B, this would mean that before B can start processing the k-th part, process A has to be finished with the $(k+1)$-st part. Such a behavior is very similar to the operation of interleaving manufacturing systems, in which some processing steps on the next part may be executed prior to the finishing of a previous part. In general, in such interleaving manufacturing systems the time necessary to finish one part is greater than the cycle time of the system, i.e., the time interval in which parts enter the system. However, there is a significant difference in "allowing" to start processing the next part prior to releasing the previous part and the *requirement* to start (and finish) processing steps prior to the start of other processing steps of a previous part.

To model such a requirement in terms of timed event graphs, the notion of negative numbers of tokens has been introduced in this thesis. A negative number of tokens in a place basically represents a minimal number of additional firings that the place's input transition has to perform compared to its output transition. However, by definition only non-negative numbers of tokens in places are permitted in (standard) timed event graphs. Therefore, we have introduced extended timed event graphs to resolve this issue. Furthermore, negative numbers of tokens in (extended) timed event graphs correspond to negative powers in γ with respect to the idempotent semiring $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$. Mathematically this is not an issue as the powers in $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$ are by definition integer numbers. However, the idea of causality, realizability, and rationality of a series in $\mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$ as defined in [2] is based on the definition of standard timed event graphs. More precisely, a series $s \in \mathcal{M}_{in}^{max}[\![\gamma,\delta]\!]$ is realizable if it corresponds to the transfer relation of a (standard) timed event graph [15, 30] and according to Theorem 3.27 a realizable series is also causal and periodic. Consequently, we have extended the notion of causality and realizability for series corresponding to the transfer relations of extended timed event graphs.

Using negative numbers of tokens and their corresponding representation in dioids it is possible to model the minimal number of entities within a process. Analogously, it might be possible to model the maximal time a token may stay in a place by a negative time. However, while it is relatively straightforward to argue that a negative number of tokens between two transitions in timed event graphs may well represent a causal behavior, it is hard to do the same for negative times.

To incorporate maximal times a different approach has been investigated. This approach is mainly based on the dual operations, namely the dual multiplication and its corresponding dual residuation, of idempotent semirings. Using these dual operations it is possible to model two different kinds of inequalities, i.e., $x \geq \underline{A} \otimes x$ and $x \leq \overline{A} \odot x$. With respect to timed event graphs, the first inequality refers to holding times, i.e., minimal times tokens have to spend in places, and positive tokens, while the latter inequality includes maximal times tokens have to spend in places. Additionally, it is possible to include constraints on the minimal number of tokens in places in the second inequality. By applying the dual product, these negative numbers of tokens, however, are modeled by positive exponents in γ, e.g., $a \geq \gamma^{-1} \otimes b$ corresponds to $b \leq \gamma^{+1} \odot a$.

In other words, the inequality using the standard multiplication $\otimes$ corresponds to the part of the system that can be modeled with standard timed event graphs and the inequality containing the dual multiplication $\odot$ refers to the part of the system which can only be modeled by extended timed event graphs. This includes lower bounds on the number of tokens in places and upper bounds for the time tokens may spend in places.

We have shown how matrices $\underline{A}$ and $\overline{A}$ can be used to determine a general system matrix $\overline{\underline{A}}$ of extended timed event graphs. If the constraints modeled in $\underline{A}$ and $\overline{A}$ are not feasible, the corresponding system matrix $\overline{\underline{A}}$ will indicate such conflicts of constraints. Once a "feasible" system matrix has been obtained, it can be used to design controllers such that the controlled system meets all constraints imposed. If, however, a unforeseen disturbance occurs during runtime, the constraints on the maximal time between the firing of two consecutive transitions may not be guaranteed. As mentioned before, a controller in a dioid framework may only delay the occurrence of events, i.e., the firing of transitions. If, for example, a transition is delayed by a disturbance, the controller cannot force the transition to fire, even though a possible constraint on the maximal time between this and another transition may be violated. In such a case the controlled system returns to the desired system behavior as soon as the disturbance is resolved.

The introduced approach has been tested on so-called high-throughput screening (HTS) systems. Such systems are used in the pharmaceutical industries to detect unknown combinations of active bio-chemical ingredients, which may serve as a basis for new drugs. Due to the fact that in HTS systems bio-chemical reactions take place it is not surprising that maximal times for some of these reactions may be an issue. Furthermore, the operation of HTS systems is very expensive. Therefore, HTS companies have a great interest in operating their systems with the highest possible throughput. As a microplate, i.e., the physical entity of a HTS system, may be processed several times on the same resource, the optimal schedule providing the highest possible throughput may be nested. Nested schedules are characterized by the phenomenon that some processing steps of a previous entity are executed after processing steps of a current entity. Such a behavior can be modeled with a negative number tokens. Consequently, high-throughput screening systems are suitable to test our modeling and control approach. Through an established cooperation with CyBio AG, a company developing and manufacturing HTS plants, real HTS examples were available for testing. The applicability of our approach has been shown on such a real HTS example.

Bibliography

[1] A. M. Atto, C. Martinez, and S. Amari. Control of discrete event systems with respect to strict duration: Supervision of an idustrial manufacturing plant. *Computers & Industrial Engineering*, 61(4):1149–1159, 2011. doi: 10.1016/j.cie.2011.07.004. (Cited on page 2.)

[2] François Baccelli, Guy Cohen, Geert Jan Olsder, and Jean-Pierre Quadrat. *Synchronization and Linearity – An Algebra for Discrete Event Systems*. Wiley, web edition, 2001. (Cited on pages 2, 5, 10, 12, 13, 15, 16, 18, 19, 20, 22, 27, 44, 48, 59, 117, and 118.)

[3] T. S. Blyth. *Lattices and ordered algebraic structures*. Springer Verlag, 2005. (Cited on page 11.)

[4] T. S. Blyth and M. F. Janowitz. *Residuation Theory*. Pergamon press, 1972. (Cited on pages 11 and 12.)

[5] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi. *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*. John Wiley & Sons, 2nd edition, 2006. (Cited on page 1.)

[6] T. L. Booth. *Sequential Machines and Automata Theory*. John Wiley, New York, 1967. (Cited on page 1.)

[7] T. Brunsch, L. Hardouin, C. A. Maia, and J. Raisch. Duality and interval analysis over idempotent semirings. *Linear Algebra and its Applications*, 437(10):2436–2454, November 2012. doi: 10.1016/j.laa.2012.06.025. (Cited on pages 22, 23, 83, and 88.)

[8] T. Brunsch, L. Hardouin, and J. Raisch. Modeling manufacturing systems in a dioid framework. In J. Campos, C. Seatzu, and X. Xie, editors, *Formal Methods in Manufacturing*, Series on Industrial Information Technology, chapter 2, pages 29–74. CRC Press/Taylor and Francis, 2013. Accepted for publication. (Cited on pages 5, 31, and 83.)

[9] T. Brunsch, J. Raisch, L. Hardouin, and O. Boutin. Discrete-event systems in a dioid framework: Modeling and analysis. In Carla Seatzu, Manuel Silva, and Jan H. van Schuppen, editors, *Control of Discrete-Event Systems*, volume 433 of *Lecture Notes in Control and*

Information Sciences, chapter 21, pages 431–450. Springer Berlin / Heidelberg, 2013. doi: 10.1007/978-1-4471-4276-8_21. (Cited on page 31.)

[10] C. G. Cassandras, S. Lafortune, and G. J. Olsder. Introduction to the modelling, control and optimization of discrete event systems. In A. Isidori, editor, *Trends in Control – A European Perspective*, pages 217–291. Springer, 1995. (Cited on pages 35 and 36.)

[11] Christos G. Cassandras and Stéphane Lafortune. *Introduction to Discrete Event Systems*. Springer, 2nd edition, 2008. (Cited on pages 1 and 31.)

[12] P. Chrétienne. *Les Réseaux de Pétri Temporisés*. Thèse d'état, Université de Paris VI, Paris, France, 1983. (Cited on page 2.)

[13] G. Cohen, S. Gaubert, R. Nikoukhah, and J.-P. Quadrat. Second order theory of min-linear systems and its application to discrete event systems. In *Proceedings of the 30th CDC*, 1991. (Cited on page 17.)

[14] G. Cohen, S. Gaubert, and J.-P. Quadrat. Linear projectors in the max-plus algebra. In *Proceedings of the IEEE-Mediterranean Conference*, Cyprus, July 1997. (Cited on page 13.)

[15] Guy Cohen, Pierre Moller, Jean-Pierre Quadrat, and Michel Viot. Algebraic tools for the performance evaluation of discrete event systems. *IEEE Proceedings*, 77:39–58, 1989. (Cited on pages 10, 45, 46, 48, 59, 60, 68, and 118.)

[16] F. Commoner, A. W. Holt, S. Even, and A. Pnueli. Marked directed graphs. *Journal of Computer and System Sciences*, 5(5):511–523, October 1971. (Cited on page 2.)

[17] B. Cottenceau, L. Hardouin, J.-L. Boimond, and J.-L. Ferrier. Synthesis of greatest linear feedback for timed event graphs in dioid. *IEEE Transactions on Automatic Control*, 44(6): 1258–1262, 1999. (Cited on pages 2, 44, and 117.)

[18] B. Cottenceau, L. Hardouin, M. Lhommeau, and J.-L. Boimond. Data processing tool for calculation in dioid. In *Proc. 5th International Workshop on Discrete Event Systems*, Ghent, Belgium, 2000. (Cited on pages 97 and 116.)

[19] B. Cottenceau, L. Hardouin, J.-L. Boimond, and J.-L. Ferrier. Model reference control for timed event graphs in dioids. *Automatica*, 37(9):1451–1458, September 2001. doi: 10.1016/S0005-1098(01)00073-5. (Cited on pages 2, 17, and 117.)

[20] Bertrand Cottenceau. *Contribution à la commande des systèmes à événements discrets : synthèse de correcteurs pour les graphs d'événements temporisés dans les dioïdes*. PhD thesis, LISA - Université d'Angers, 1999. (Cited on pages 2, 15, 17, and 117.)

[21] R. A. Cuninghame-Green. *Minimax Algebra*. Springer, 1979. (Cited on page 2.)

[22] R. A. Cuninghame-Green and P. Butkovic. The equation $a \otimes x = b \otimes y$ over (max,+). *Theoretical Computer Science*, 293(1):3–12, 2003. doi: 10.1016/S0304-3975(02)00228-1. (Cited on page 93.)

[23] David Freedman. *Markov Chains*. Springer New York, 1983. (Cited on page 1.)

[24] Stéphane Gaubert. *Théorie des systèmes linéaires dans les dioïdes*. PhD thesis, INRIA - Ecole des Mines de Paris, 1992. (Cited on pages 13, 15, and 17.)

[25] David Harding, Martyn Banks, Simon Fogarty, and Alastair Binnie. Development of an automated high-throughput screening systems: a case history. *Drug Discovery Today*, 2 (9):385–390, September 1997. (Cited on page 103.)

[26] L. Hardouin, E. Menguy, J.-L. Boimond, and J.-L. Ferrier. SISO discrete event systems control in dioids algebra. *Special Issue of Journal Européen des Systèmes Automatisés*, 31 (3):433–452, 1997. (Cited on pages 2 and 117.)

[27] L. Hardouin, B. Cottenceau, M. Lhommeau, and E. Le Corronc. Interval systems over idempotent semirings. *Linear Algebra and its Applications*, 432(5–7):855–862, 2009. doi: 10.1016/j.laa.2009.03.039. (Cited on page 93.)

[28] L. Hardouin, C. A. Maia, B. Cottenceau, and M. Lhommeau. Observer design for (max,+) linear systems. *IEEE Transactions on Automatic Control*, 55(2):538–543, February 2010. (Cited on pages 69 and 117.)

[29] L. Hardouin, C. A. Maia, B. Cottenceau, and R. Santos-Mendes. Max-plus linear observer: Application to manufacturing systems. In *Preprints of the 10th International Workshop on Discrete Event Systems (WODES'10)*, Berlin, Germany, August 2010. (Cited on page 117.)

[30] L. Hardouin, O. Boutin, B. Cottenceau, T. Brunsch, and J. Raisch. Discrete-event systems in a dioid framework: Control theory. In Carla Seatzu, Manuel Silva, and Jan H. van Schuppen, editors, *Control of Discrete-Event Systems*, volume 433 of *Lecture Notes in Control and Information Sciences*, chapter 22, pages 451–469. Springer Berlin / Heidelberg, 2013. doi: 10.1007/978-1-4471-4276-8_22. (Cited on pages 31 and 118.)

[31] Bernd Heidergott, Geert Jan Olsder, and Jacob van der Woude. *Max Plus at Work*. Princeton University Press, Princeton, NJ, USA, 1st edition, 2006. (Cited on page 38.)

[32] J. E. Hopcroft, R. Motwani, and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Prentice Hall, 3rd edition, 2006. (Cited on page 1.)

[33] L. Houssin, S. Lahaye, and J.-L. Boimond. Just in time control of constrained (max,+)-linear systems. *Discrete Event Dynamic Systems*, 17(2):159–178, 2007. doi: 10.1007/s10626-006-0009-5. (Cited on page 2.)

[34] K. Jensen and L. M. Kristensen. *Coloured Petri Nets – Modelling and Validation of Concurrent Systems*. Springer-Verlag Berlin Heidelberg, 2009. doi: 10.1007/b95112. (Cited on page 1.)

[35] Bronisław Knaster. Une théorème sur les fonctions d'ensembles. *Annales Soc. Polonaise Math.*, 6:133–134, 1927. (Cited on page 15.)

[36] R. Kumar and V. Garg. *Modelling and control of logical discrete event systems*. Kluwer Academic Publishers, 1995. (Cited on page 15.)

[37] Tae-Eog Lee and Seong-Ho Park. Steady state analysis of a timed event graph with time window constraints. In *Proc. of the 2005 IEEE International Conference on Automation Science and Engineering*, pages 404–409, Edmonton, Canada, 2005. (Cited on pages 3, 49, and 83.)

[38] Tae-Eog Lee and Seong-Ho Park. An extended event graph with negative places and tokens for time window constraints. *IEEE Transactions on Automation Science and Engineering*, 2(4):319–332, October 2005. (Cited on pages 3, 49, and 83.)

[39] M. Lhommeau. *Etude de systèmes à événements discrets dans l'algèbre (max,+): 1. Synthèse de correcteurs robustes dans un dioïde d'intervalles. 2. Synthèse de correcteurs en présence de pertubations*. PhD thesis, LISA – Université d'Angers, 2003. (Cited on pages 2, 18, and 117.)

[40] M. Lhommeau, L. Hardouin, and B. Cottenceau. Disturbance decoupling of timed event graphs by output feedback controller. In *Workshop on Discrete Event Systems (WODES'2002)*, Zaragoza, Spain, October 2002. (Cited on pages 2, 75, and 117.)

[41] M. Lhommeau, L. Hardouin, B. Cottenceau, and L. Jaulin. Interval analysis and dioid: Application to robust controller design for timed event graphs. *Automatica*, 40:1923–1930, 2004. (Cited on pages 2 and 117.)

[42] C. A. Maia, C. R. Andrade, and L. Hardouin. On the control of max-plus linear system subject to state restriction. *Automatica*, 47(5):988–992, 2011. doi: 10.1016/j.automatica.2011.01.047. (Cited on page 2.)

[43] E. Mayer. *Globally optimal schedules for cyclic systems with nonblocking specification and time window constraints*. PhD thesis, Technische Universität Berlin, 2007. (Cited on pages 3 and 106.)

[44] E. Mayer and J. Raisch. Time-optimal scheduling for high throughput screening processes using cyclic discrete event models. *MATCOM - Mathematics and Computers in Simulation*, 66(2–3):181–191, June 2004. doi: 10.1016/j.matcom.2003.11.004. (Cited on page 3.)

[45] E. Mayer, U.-U. Haus, J. Raisch, and R. Weismantel. Throughput-optimal sequences for cyclically operated plants. *Discrete Event Dynamic Systems*, 18(3):355–383, 2008. (Cited on page 3.)

[46] L. M. Mayr and P. Fuerst. The future of high-throughput screening. *Journal of Biomolecular Screening*, 13(6):443–448, 2008. (Cited on page 103.)

[47] E. Menguy, J.-L. Boimond, and L. Hardouin. Optimal control of discrete event systems in case of updated reference input. In *Proc. IFAC Conference on System Structure and Control*, Nantes, France, 1998. (Cited on page 68.)

[48] E. Menguy, J.-L. Boimond, L. Hardouin, and J.-L. Ferrier. Just in time control of timed event graphs: update of referecence input, presence of uncontrollable input. *IEEE Transactions on Automatic Control*, 45(11):2155–2159, 2000. (Cited on pages 2, 68, and 117.)

[49] T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77 (4):541–580, 1989. (Cited on page 1.)

[50] T. Murata and H. Yamaguchi. A petri net with negative tokens and its application to automated reasoning. In *Proceedings of the 33rd Midwest Symposium on Circuits and Systems*, August 1990. doi: 10.1109/MWSCAS.1990.140832. (Cited on page 2.)

[51] I. Ouerghi and L. Hardouin. A precompensator synthesis for p-temporal event graphs. *Positive Systems, LNCIS*, 341:391–398, 2006. (Cited on page 2.)

[52] I. Ouerghi and L. Hardouin. Control synthesis for p-temporal event graphs. In *International Workshop on Discrete Event Systems (WODES 2006)*, Ann Arbor, USA, 2006. doi: 10.1109/WODES.2006.1678435. (Cited on page 2.)

[53] Iteb Ouerghi. *Etude de systèmes (max,+)-linéaires soumis à des contraintes, application à la commande des graphes d'événements P-temporal.* PhD thesis, LISA - Université d'Angers, 2006. (Cited on pages 2, 18, 19, 20, 22, 49, and 83.)

[54] J. L. Peterson. Petri nets. *ACM Computing Surveys (CSUR)*, 9(3):223–252, September 1977. doi: 10.1145/356698.356702. (Cited on page 1.)

[55] J. L. Peterson. *Petri Net Theory and the Modeling of Systems.* Prentice Hall, 1981. (Cited on page 1.)

[56] Carl Adam Petri. *Kommunikation mit Automaten.* PhD thesis, Technische Universität Darmstadt, 1962. (Cited on page 31.)

[57] Mark V. Rogers. High-throughput screening. *Drug Discovery Today*, 2(11):503–504, November 1997. (Cited on page 103.)

[58] B. De Schutter and T. van den Boom. Model predictive control for max-plus-linear discrete event systems. *Automatica*, 37(7):1049–1056, 2001. (Cited on pages 2 and 117.)

[59] Gábor Szász. *Introduction to lattice theory*. Academic Press New York and London, 1963. (Cited on page 15.)

[60] Alfred Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific J. Math.*, 5:285–310, 1955. (Cited on page 15.)

[61] T. van den Boom and B. De Schutter. Properties of MPC for max-plus-linear systems. *European Journal of Control*, 8(5):453–462, 2002. (Cited on pages 2 and 117.)

Declaration

I declare that this thesis has been composed by myself, that the work contained herein is my own except where explicitly stated otherwise, and that this work has not been submitted for any other degree or professional qualification except as specified.

Berlin, November 2013

Thomas Brunsch