

Synchronization and Linearity

An Algebra for Discrete Event Systems

Francois Louis Baccelli

*Institut National de Recherche en Informatique et Automatique
(INRIA), Sophia Antipolis, France*

Guy Cohen

Ecole des Mines de Paris, Fontainebleau, France

Geert Jan Olsder

Delft University of Technology, The Netherlands

Jean-Pierre Quadrat

*Institut National de Recherche en Informatique et Automatique
(INRIA), Rocquencourt, France*

JOHN WILEY & SONS

Chichester · New York · Brisbane · Toronto · Singapore

Copyright © 1992 by John Wiley & Sons Ltd.
Baffins Lane, Chichester
West Sussex PO19 1UD, England

All rights reserved.

No part of this book may be reproduced by any means,
or transmitted, or translated into a machine language
without the written permission of the publisher.

Other Wiley Editorial Offices

John Wiley & Sons, Inc., 605 Third Avenue,
New York, NY 10158-0012, USA

Jacaranda Wiley Ltd, G.P.O. Box 859, Brisbane,
Queensland 4001, Australia

John Wiley & Sons (Canada) Ltd, 22 Worcester Road,
Rexdale, Ontario M9W 1L1, Canada

John Wiley & Sons (SEA) Pte Ltd, 37 Jalan Pemimpin #05-04,
Block B, Union Industrial Building, Singapore 2057

Library of Congress Cataloging-in-Publication Data

Synchronization and linearity : an algebra for discrete event systems
/ Francois Louis Baccelli . . . [et al.].

p. cm.

Includes bibliographical references and index.

ISBN 0 471 93609 X

1. System analysis. 2. Discrete-time systems. 3. Algebra.

I. Baccelli, F. (François), 1954

T57.6.S953 1992

003—dc20

92-542

CIP

British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

ISBN 0 471 93609 X

Produced from camera-ready copy supplied by the authors
Printed and bound in Great Britain by Biddles Ltd, Guildford, Surrey

*To
our
families*



Contents

Preface	xv
I Discrete Event Systems and Petri Nets	1
1 Introduction and Motivation	3
1.1 Preliminary Remarks and Some Notation	3
1.2 Miscellaneous Examples	8
1.2.1 Planning	9
1.2.2 Communication	13
1.2.3 Production	15
1.2.4 Queuing System with Finite Capacity	18
1.2.5 Parallel Computation	19
1.2.6 Traffic	22
1.2.7 Continuous System Subject to Flow Bounds and Mixing	25
1.3 Issues and Problems in Performance Evaluation	27
1.4 Notes	33
2 Graph Theory and Petri Nets	35
2.1 Introduction	35
2.2 Directed Graphs	35
2.3 Graphs and Matrices	38
2.3.1 Composition of Matrices and Graphs	41
2.3.2 Maximum Cycle Mean	46
2.3.3 The Cayley-Hamilton Theorem	49
2.4 Petri Nets	54
2.4.1 Definition	54
2.4.2 Subclasses and Properties of Petri Nets	59
2.5 Timed Event Graphs	63
2.5.1 Simple Examples	64
2.5.2 The Basic Autonomous Equation	69
2.5.3 Constructiveness of the Evolution Equations	78
2.5.4 Standard Autonomous Equations	82
2.5.5 The Nonautonomous Case	84
2.5.6 Construction of the Marking	87
2.5.7 Stochastic Event Graphs	88

2.6	Modeling Issues	89
2.6.1	Multigraphs	89
2.6.2	Places with Finite Capacity	90
2.6.3	Synthesis of Event Graphs from Interacting Resources	90
2.7	Notes	98

II Algebra 101

3 Max-Plus Algebra 103

3.1	Introduction	103
3.1.1	Definitions	103
3.1.2	Notation	105
3.1.3	The min Operation in the Max-Plus Algebra	105
3.2	Matrices in $\mathbb{R}_{\max}$	106
3.2.1	Linear and Affine Scalar Functions	107
3.2.2	Structures	108
3.2.3	Systems of Linear Equations in $(\mathbb{R}_{\max})^n$	110
3.2.4	Spectral Theory of Matrices	113
3.2.5	Application to Event Graphs	116
3.3	Scalar Functions in $\mathbb{R}_{\max}$	118
3.3.1	Polynomial Functions $\mathcal{P}(\mathbb{R}_{\max})$	118
3.3.2	Rational Functions	126
3.3.3	Algebraic Equations	129
3.4	Symmetrization of the Max-Plus Algebra	131
3.4.1	The Algebraic Structure $\mathbb{S}$	131
3.4.2	Linear Balances	133
3.5	Linear Systems in $\mathbb{S}$	135
3.5.1	Determinant	136
3.5.2	Solving Systems of Linear Balances by the Cramer Rule	137
3.6	Polynomials with Coefficients in $\mathbb{S}$	140
3.6.1	Some Polynomial Functions	140
3.6.2	Factorization of Polynomial Functions	142
3.7	Asymptotic Behavior of A^k	145
3.7.1	Critical Graph of a Matrix A	145
3.7.2	Eigenspace Associated with the Maximum Eigenvalue	147
3.7.3	Spectral Projector	149
3.7.4	Convergence of A^k with k	150
3.7.5	Cyclic Matrices	153
3.8	Notes	154

4 Dioids 155

4.1	Introduction	155
4.2	Basic Definitions and Examples	156
4.2.1	Axiomatics	156
4.2.2	Some Examples	157
4.2.3	Subdioids	158
4.2.4	Homomorphisms, Isomorphisms and Congruences	159
4.3	Lattice Properties of Dioids	160
4.3.1	Basic Notions in Lattice Theory	160
4.3.2	Order Structure of Dioids	163
4.3.3	Complete Dioids, Archimedian Dioids	164
4.3.4	Lower Bound	166
4.3.5	Distributive Dioids	167
4.4	Isotone Mappings and Residuation	169
4.4.1	Isotony and Continuity of Mappings	169
4.4.2	Elements of Residuation Theory	174
4.4.3	Closure Mappings	180
4.4.4	Residuation of Addition and Multiplication	181
4.5	Fixed-Point Equations, Closure of Mappings and Best Approximation	187
4.5.1	General Fixed-Point Equations	187
4.5.2	The Case $\Pi(x) = a \dot{x} x \wedge b$	191
4.5.3	The Case $\Pi(x) = ax \oplus b$	192
4.5.4	Some Problems of Best Approximation	194
4.6	Matrix Dioids	197
4.6.1	From 'Scalars' to Matrices	197
4.6.2	Residuation of Matrices and Invertibility	198
4.7	Dioids of Polynomials and Power Series	200
4.7.1	Definitions and Properties of Formal Polynomials and Power Series	200
4.7.2	Subtraction and Division of Power Series	204
4.7.3	Polynomial Matrices	204
4.8	Rational Closure and Rational Representations	206
4.8.1	Rational Closure and Rational Calculus	206
4.8.2	Rational Representations	208
4.8.3	Yet Other Rational Representations	210
4.8.4	Rational Representations in Commutative Dioids	212
4.9	Notes	214
4.9.1	Dioids and Related Structures	214
4.9.2	Related Results	215

III Deterministic System Theory 217

5 Two-Dimensional Domain Description of Event Graphs 219

5.1	Introduction	219
5.2	A Comparison Between Counter and Dater Descriptions	221
5.3	Daters and their Embedding in Nonmonotonic Functions	225
5.3.1	A Dioid of Nondecreasing Mappings	225
5.3.2	γ -Transforms of Daters and Representation by Power Series in γ	228
5.4	Moving to the Two-Dimensional Description	234
5.4.1	The $\mathbb{Z}_{\max}$ Algebra through Another Shift Operator	234
5.4.2	The $\mathcal{M}_{\text{in}}^{\text{ax}}[\gamma, \delta]$ Algebra	236
5.4.3	Algebra of Information about Events	242
5.4.4	$\mathcal{M}_{\text{in}}^{\text{ax}}[\gamma, \delta]$ Equations for Event Graphs	243
5.5	Counters	249
5.5.1	A First Derivation of Counters	249
5.5.2	Counters Derived from Daters	250
5.5.3	Alternative Definition of Counters	252
5.5.4	Dynamic Equations of Counters	252
5.6	Backward Equations	254
5.6.1	$\mathcal{M}_{\text{in}}^{\text{ax}}[\gamma, \delta]$ Backward Equations	254
5.6.2	Backward Equations for Daters	255
5.7	Rationality, Realizability and Periodicity	258
5.7.1	Preliminaries	258
5.7.2	Definitions	259
5.7.3	Main Theorem	260
5.7.4	On the Coding of Rational Elements	261
5.7.5	Realizations by γ - and δ -Transforms	264
5.8	Frequency Response of Event Graphs	265
5.8.1	Numerical Functions Associated with Elements of $\mathbb{B}[\gamma, \delta]$	266
5.8.2	Specialization to $\mathcal{M}_{\text{in}}^{\text{ax}}[\gamma, \delta]$	268
5.8.3	Eigenfunctions of Rational Transfer Functions	269
5.9	Notes	273

6 Max-Plus Linear System Theory 275

6.1	Introduction	275
6.2	System Algebra	275
6.2.1	Definitions	275
6.2.2	Some Elementary Systems	277
6.3	Impulse Responses of Linear Systems	280
6.3.1	The Algebra of Impulse Responses	280
6.3.2	Shift-Invariant Systems	282
6.3.3	Systems with Nondecreasing Impulse Response	283
6.4	Transfer Functions	284

6.4.1	Evaluation Homomorphism	284
6.4.2	Closed Concave Impulse Responses and Inputs	286
6.4.3	Closed Convex Inputs	288
6.5	Rational Systems	290
6.5.1	Polynomial, Rational and Algebraic Systems	290
6.5.2	Examples of Polynomial Systems	292
6.5.3	Characterization of Rational Systems	292
6.5.4	Minimal Representation and Realization	296
6.6	Correlations and Feedback Stabilization	298
6.6.1	Sojourn Time and Correlations	299
6.6.2	Stability and Stabilization	302
6.6.3	Loop Shaping	305
6.7	Notes	306

IV Stochastic Systems 307

7 Ergodic Theory of Event Graphs 309

7.1	Introduction	309
7.2	A Simple Example in $\mathbb{R}_{\max}$	310
7.2.1	The Event Graph	310
7.2.2	Statistical Assumptions	311
7.2.3	Statement of the Eigenvalue Problem	313
7.2.4	Relation with Event Graph	317
7.2.5	Uniqueness and Coupling	319
7.2.6	First-Order and Second-Order Theorems	321
7.3	First-Order Theorems	323
7.3.1	Notation and Statistical Assumptions	323
7.3.2	Examples in $\mathbb{R}_{\max}$	324
7.3.3	Maximal Lyapunov Exponent in $\mathbb{R}_{\max}$	325
7.3.4	The Strongly Connected Case	327
7.3.5	General Graph	328
7.3.6	First-Order Theorems in Other Dioids	333
7.4	Second-Order Theorems; Nonautonomous Case	333
7.4.1	Notation and Assumptions	333
7.4.2	Ratio Equation in a General Dioid	334
7.4.3	Stationary Solution of the Ratio Equation	335
7.4.4	Specialization to $\mathbb{R}_{\max}$	337
7.4.5	Multiplicative Ergodic Theorems in $\mathbb{R}_{\max}$	351
7.5	Second-Order Theorems; Autonomous Case	353
7.5.1	Ratio Equation	353
7.5.2	Backward Process	354
7.5.3	From Stationary Ratios to Random Eigenpairs	356
7.5.4	Finiteness and Coupling in $\mathbb{R}_{\max}$; Positive Case	357

7.5.5	Finiteness and Coupling in $\mathbb{R}_{\max}$; Strongly Connected Case	366
7.5.6	Finiteness and Coupling in $\mathbb{R}_{\max}$; General Case	368
7.5.7	Multiplicative Ergodic Theorems in $\mathbb{R}_{\max}$	370
7.6	Stationary Marking of Stochastic Event Graphs	370
7.7	Appendix on Ergodic Theorems	373
7.8	Notes	375
8	Computational Issues in Stochastic Event Graphs	377
8.1	Introduction	377
8.2	Monotonicity Properties	378
8.2.1	Notation for Stochastic Ordering	378
8.2.2	Monotonicity Table for Stochastic Event Graphs	378
8.2.3	Properties of Daters	379
8.2.4	Properties of Counters	383
8.2.5	Properties of Cycle Times	388
8.2.6	Comparison of Ratios	390
8.3	Event Graphs and Branching Processes	393
8.3.1	Statistical Assumptions	393
8.3.2	Statistical Properties	393
8.3.3	Simple Bounds on Cycle Times	394
8.3.4	General Case	399
8.4	Markovian Analysis	404
8.4.1	Markov Property	404
8.4.2	Discrete Distributions	405
8.4.3	Continuous Distribution Functions	413
8.5	Appendix	417
8.5.1	Stochastic Comparison	417
8.5.2	Markov Chains	419
8.6	Notes	419

V Postface **421**

9 Related Topics and Open Ends **423**

9.1	Introduction	423
9.2	About Realization Theory	423
9.2.1	The Exponential as a Tool; Another View on Cayley-Hamilton	423
9.2.2	Rational Transfer Functions and ARMA Models	426
9.2.3	Realization Theory	427
9.2.4	More on Minimal Realizations	429
9.3	Control of Discrete Event Systems	431
9.4	Brownian and Diffusion Decision Processes	433
9.4.1	Inf-Convolutions of Quadratic Forms	434

9.4.2	Dynamic Programming	434
9.4.3	Fenchel and Cramer Transforms	436
9.4.4	Law of Large Numbers in Dynamic Programming	437
9.4.5	Central Limit Theorem in Dynamic Programming	438
9.4.6	The Brownian Decision Process	438
9.4.7	Diffusion Decision Process	440
9.5	Evolution Equations of General Timed Petri Nets	443
9.5.1	FIFO Timed Petri Nets	443
9.5.2	Evolution Equations	444
9.5.3	Evolution Equations for Switching	449
9.5.4	Integration of the Recursive Equations	450
9.6	Min-Max Systems	452
9.6.1	General Timed Petri Nets and Descriptor Systems	453
9.6.2	Existence of Periodic Behavior	455
9.6.3	Numerical Procedures for the Eigenvalue	457
9.6.4	Stochastic Min-Max Systems	460
9.7	About Cycle Times in General Petri Nets	461
9.8	Notes	464

Bibliography **465**

Notation **475**

Index **479**

Preface

The mathematical theory developed in this book finds its initial motivation in the modeling and the analysis of the time behavior of a class of dynamic systems now often referred to as 'discrete event (dynamic) systems' (DEDS). This class essentially contains man-made systems that consist of a finite number of resources (processors or memories, communication channels, machines) shared by several users (jobs, packets, manufactured objects) which all contribute to the achievement of some common goal (a parallel computation, the end-to-end transmission of a set of packets, the assembly of a product in an automated manufacturing line). The coordination of the user access to these resources requires complex control mechanisms which usually make it impossible to describe the dynamic behavior of such systems in terms of differential equations, as in physical phenomena. The dynamics of such systems can in fact be described using the two (Petri net like) paradigms of 'synchronization' and 'concurrency'. Synchronization requires the availability of several resources or users at the same time, whereas concurrency appears for instance when, at a certain time, some user must choose among several resources. The following example only contains the synchronization aspect which is the main topic of this book.

Consider a railway station. A departing train must wait for certain incoming trains so as to allow passengers to change, which reflects the synchronization feature. Consider a network of such stations where the traveling times between stations are known. The variables of interest are the arrival and departure times, assuming that trains leave as soon as possible. The departure time of a train is related to the maximum of the arrival times of the trains conditioning this departure. Hence the max operation is the basic operator through which variables interact. The arrival time at a station is the sum of the departure time from the previous station and the traveling time. There is no concurrency since it has tacitly been assumed that each train has been assigned a fixed route.

The thesis developed here is that there exists an algebra in which DEDS that do not involve concurrency can naturally be modeled as linear systems. A linear model is a set of equations in which variables can be added together and in which variables can also be multiplied by coefficients which are a part of the data of the model. The train example showed that the max is the essential operation that captures the synchronization phenomenon by operating on arrival times to compute departure times. Therefore the basic idea is to treat the max as the 'addition' of the algebra (hence this max will be written $\oplus$ to suggest 'addition'). The same example indicates that we also need conventional addition to transform variables from one end of an arc of the network to the other end (the addition of the traveling time, the data, to the departure time). This is why $+$ will be treated as multiplication in this

algebra (and it will be denoted $\otimes$). The operations $\oplus$ and $\otimes$ will play their own roles and in other examples they are not necessarily confined to operate as max and +, respectively.

The basic mathematical feature of $\oplus$ is that it is idempotent: $x \oplus x = x$. In practice, it may be the max or the min of numbers, depending on the nature of the variables which are handled (either the times at which events occur, or the numbers of events during given intervals). But the main feature is again idempotency of addition. The role of $\otimes$ is generally played by conventional addition, but the important thing is that it behaves well with respect to addition (e.g. that it distributes with respect to $\oplus$). The algebraic structure outlined is known under the name of 'dioid', among other names. It has connections with standard linear algebra with which it shares many combinatorial properties (associativity and commutativity of addition, etc.), but also with lattice-ordered semigroup theory for, speaking of an idempotent addition is equivalent to speaking of the 'least upper bound' in lattices.

Conventional system theory studies networks of integrators or 'adders' connected in series, parallel and feedback. Similarly, queuing theory or Petri net theory build up complex systems from elementary objects (namely, queues, or transitions and places). The theory proposed here studies complex systems which are made up of elementary systems interacting through a basic operation, called synchronization, located at the nodes of a network.

The mathematical contributions of the book can be viewed as the first steps toward the development of a theory of *linear* systems on dioids. Both deterministic and stochastic systems are considered. Classical concepts of system theory such as state space' recursive equations, input-output (transfer) functions, feedback loops, etc. are introduced. Overall, this theory offers a unifying framework for systems in which the basic 'engine' of dynamics is synchronization, when these systems are considered from the point of view of performance evaluation. In other words, dioid algebra appears to be the right tool to handle synchronization in a *linear* manner, whereas this phenomenon seems to be very nonlinear, or even nonsmooth, 'through the glasses' of conventional algebraic tools. Moreover, this theory may be a good starting point to encompass other basic features of discrete event systems such as concurrency, but at the price of considering systems which are *nonlinear* even in this new framework. Some perspectives are opened in this respect in the last chapter.

Although the initial motivation was essentially found in the study of discrete event systems, it turns out that this theory may be appropriate for other purposes too. This happens frequently with mathematical theories which often go beyond their initial scope, as long as other objects can be found with the same basic features. In this particular case the common feature may be expressed by saying that the input-output relation has the form of an inf- (or a sup-) convolution. In the same way, the scope of conventional system theory is the study of input-output relations which are convolutions. In Chapter 1 it is suggested that this theory is also relevant for some systems which either are continuous or do not involve synchronization. Systems which mix fluids in certain proportions and which involve flow constraints all in the former category. Recursive 'optimization processes', of which dynamic

programming is the most immediate example, fall in the latter category. All these systems involve max (or min) and + as the basic operations. Another situation where dioid algebra naturally shows up is the asymptotic behavior of exponential functions. In mathematical terms, the conventional operations + and $\times$ over positive numbers, say, are transformed into max and +, respectively, by the mapping: $x \mapsto \lim_{s \rightarrow +\infty} \exp(sx)$. This is relevant, for example, in the theory of large deviations, and, coming back to conventional system theory, when outlining Bode diagrams by their asymptotes.

There are numerous concurrent approaches for constructing a mathematical framework for discrete event systems. An important dichotomy arises depending on whether the framework is intended to assess the logical behavior of the system or its temporal behavior. Among the first class, we would quote theoretical computer science languages like CSP or CCS and recent system-theoretic extensions of automata theory [114]. The algebraic approach that is proposed here is clearly of the latter type, which makes it comparable with such formalisms as timed (or stochastic) Petri nets [1], generalized semi-Markov processes [63] and in a sense queuing network theory. Another approach, that emphasizes computational aspects, is known as Perturbation Analysis [70].

A natural question of interest concerns the scope of the methodology that we develop here. Most DEDS involve concurrency at an early stage of their design. However, it is often necessary to handle this concurrency by choosing certain priority rules (by specifying routing and/or scheduling, etc.), in order to completely specify their behavior. The theory developed in this book may then be used to evaluate the consequences of these choices in terms of performance. If the delimitation of the class of queuing systems that admit a max-plus representation is not an easy task within the framework of queuing theory, the problem becomes almost transparent within the setting of Petri networks developed in Chapter 2: stochastic event graphs coincide with the class of discrete event systems that have a representation as a max-plus linear system in a random medium (i.e. the matrices of the linear system are random); any topological violation of the event graph structure, be it a competition like in multiserver queues, or a superimposition like in certain Jackson networks, results in a min-type nonlinearity (see Chapter 9). Although it is beyond the scope of the book to review the list of queuing systems that are stochastic event graphs, several examples of such systems are provided ranging from manufacturing models (e.g. assembly/disassembly queues, also called fork-join queues, jobshop and flowshop models, production lines, etc.) to communication and computer science models (communication blocking, wave front arrays, etc.)

Another important issue is that of the design gains offered by this approach. The most important structural results are probably those pertaining to the existence of periodic and stationary regimes. Within the deterministic setting, we would quote the interpretation of the pair (cycle time, periodic regime) in terms of eigenpairs together with the polynomial algorithms that can be used to compute them. Moreover, because bottlenecks of the systems are explicitly revealed (through the notion of critical circuits), this approach provides an efficient way not only to evaluate the

formance but also to assess certain design choices made at earlier stages. In the stochastic case, this approach first yields new characterizations of throughput or delay times as Lyapunov exponents associated with the matrices of the underlying linear system, whereas the steady-state regime receives a natural characterization in terms of ‘stochastic eigenvalues’ in max-plus algebra, very much in the flavor of Oseledec’s multiplicative ergodic theorems. Thanks to this, queueing theory and related Petri nets find some sort of (linear) garden where several known results concerning small dimensional systems can be derived from a few classical theorems (or more precisely from the max-plus counterpart of classical theorems).

The theory of DEDS came into existence only at the beginning of the 1980s, though it is fair to say that max-plus algebra is older, see [49], [130], [67]. The field of DEDS is in full development and this book presents in a coherent fashion the results obtained so far by this algebraic approach. The book can be used as a textbook, but it also presents the current state of the theory. Short historical notes and other remarks are given in the note sections at the end of most chapters. The book should be of interest to (applied) mathematicians, operations researchers, electrical engineers, computer scientists, probabilists, statisticians, management scientists and generally to those with a professional interest in parallel and distributed processing, manufacturing, etc. An undergraduate degree in mathematics should be sufficient to follow the flow of thought (though some parts go beyond this level). Introductory courses in algebra, probability theory and linear system theory form an ideal background. For algebra, [61] for instance provides suitable background material; for probability theory this role is for instance played by [20], and for linear system theory it is [72] or the more recent [122].

The heart of the book consists of four main parts, each of which consists of two chapters. Part I (Chapters 1 and 2) provides a natural motivation for DEDS, it is devoted to a general introduction and relationships with graph theory and Petri nets. Part II (Chapters 3 and 4) is devoted to the underlying algebras. Once the reader has gone through this part, he will also appreciate the more abstract approach presented in Parts III and IV. Part III (Chapters 5 and 6) deals with deterministic system theory, where the systems are mostly DEDS, but continuous max-plus linear systems are also discussed in Chapter 6. Part IV (Chapters 7 and 8) deals with stochastic DEDS. Many interplays of comparable results between the deterministic and the stochastic framework are shown. There is a fifth part, consisting of one chapter (Chapter 9), which deals with related areas and some open problems. The notation introduced in Parts I and II is used throughout the other parts.

The idea of writing this book took form during the summer of 1989, during which the third author (GJO) spent a mini-sabbatical at the second author’s (GC’s) institute. The other two authors (FB and JPQ) joined in the fall of 1989. During the process of writing, correcting, cutting, pasting, etc., the authors met frequently, mostly in Fontainebleau, the latter being situated close to the center of gravity of the authors’ own home towns. We acknowledge the working conditions and support of our home institutions that made this project possible. The Systems and Control Theory Network in the Netherlands is acknowledged for providing some

financial support for the necessary travels. Mr. J. Schonewille of Delft University of Technology is acknowledged for preparing many of the figures using Adobe Illustrator. Mr. G. Ouanounou of INRIA-Rocquencourt deserves also many thanks for his help in producing the final manuscript using the high-resolution equipment of this Institute. The contents of the book have been improved by remarks of P. Bougerol of the University of Paris VI, and of A. Jean-Marie and Z. Liu of INRIA-Sophia Antipolis who were all involved in the proofreading of some parts of the manuscript. The authors are grateful to them. The second (GC) and fourth (JPQ) authors wish to acknowledge the permanent interaction with the other past or present members of the so-called Max Plus working group at INRIA-Rocquencourt. Among them, M. Viot and S. Gaubert deserve special mention. Moreover, S. Gaubert helped us to check some examples included in this book, thanks to his handy computer software MAX manipulating the $\mathcal{M}_{\text{lin}}^{\text{ax}}[\gamma, \delta]$ algebra. Finally, the publisher, in the person of Ms. Helen Ramsey, is also to be thanked, specifically because of her tolerant view with respect to deadlines.

We would like to stress that the material covered in this book has been and is still in fast evolution. Owing to our different backgrounds, it became clear to us that many different cultures within mathematics exist with regard to style, notation, etc. We did our best to come up with one, uniform style throughout the book. Chances are, however, that, when the reader notices a higher density of Theorems, Definitions, etc., GC and/or JPQ were the primary authors of the corresponding parts¹. As a last remark, the third author can always be consulted on the problem of coping with three French co-authors.

François Baccelli, Sophia Antipolis
Guy Cohen, Fontainebleau
Geert Jan Olsder, Delft
Jean-Pierre Quadrat, Rocquencourt

June 1992

¹GC: I do not agree. FB is more prone to that than any of us!

PART I

Discrete Event Systems and Petri Nets

CHAPTER 1

Introduction and Motivation

1.1 PRELIMINARY REMARKS AND SOME NOTATION

Probably the most well-known equation in the theory of difference equations is

$$x(t+1) = Ax(t) \quad , \quad t = 0, 1, 2, \dots \quad (1.1)$$

The vector $x \in \mathbb{R}^n$ represents the 'state' of an underlying model and this state evolves in time according to this equation; $x(t)$ denotes the state at time t . The symbol A represents a given $n \times n$ matrix. If an initial condition

$$x(0) = x_0 \quad (1.2)$$

is given, then the whole future evolution of (1.1) is determined.

Implicit in the text above is that (1.1) is a vector equation. Written out in scalar equations it becomes

$$x_i(t+1) = \sum_{j=1}^n A_{ij}x_j(t) \quad , \quad i = 1, \dots, n \quad ; \quad t = 0, 1, \dots \quad (1.3)$$

The symbol x_i denotes the i -th component of the vector x ; the elements A_{ij} are the entries of the square matrix A . If $A_{ij}, i, j = 1, \dots, n$, and $x_j(t), j = 1, \dots, n$, are given, then $x_j(t+1), j = 1, \dots, n$, can be calculated according to (1.3).

The only operations used in (1.3) are multiplication ($A_{ij} \times x_j(t)$) and addition (the $\sum$ symbol). Most of this book can be considered as a study of formulæ of the form (1.1), in which the operations are changed. Suppose that the two operations in (1.3) are changed in the following way: addition becomes maximization and multiplication becomes addition. Then (1.3) becomes

$$\begin{aligned} x_i(k+1) &= \max(A_{i1} + x_1(k), A_{i2} + x_2(k), \dots, A_{in} + x_n(k)) \\ &= \max_j(A_{ij} + x_j(k)) \quad , \quad i = 1, \dots, n \end{aligned} \quad (1.4)$$

If the initial condition (1.2) also holds for (1.4), then the time evolution of (1.4) is completely determined again. Of course the time evolutions of (1.3) and (1.4) will be different in general. Equation (1.4), as it stands, is a nonlinear difference equation. As an example take $n = 2$, such that A is a 2×2 matrix. Suppose

$$A = \begin{pmatrix} 3 & 7 \\ 2 & 4 \end{pmatrix} \quad (1.5)$$

and that the initial condition is

$$x_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \quad (1.6)$$

the time evolution of (1.1) becomes

$$x(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad x(1) = \begin{pmatrix} 3 \\ 2 \end{pmatrix}, \quad x(2) = \begin{pmatrix} 23 \\ 14 \end{pmatrix}, \quad x(3) = \begin{pmatrix} 167 \\ 102 \end{pmatrix}, \dots$$

the time evolution of (1.4) becomes

$$x(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad x(1) = \begin{pmatrix} 7 \\ 4 \end{pmatrix}, \quad x(2) = \begin{pmatrix} 11 \\ 9 \end{pmatrix}, \quad x(3) = \begin{pmatrix} 16 \\ 13 \end{pmatrix}, \dots \quad (1.7)$$

We are used to thinking of the argument t in $x(t)$ as time; at time t the state is $x(t)$. With respect to (1.4) we will introduce a different meaning for this argument. In order to emphasize this different meaning, the argument t has been replaced by k . For this new meaning we need to think of a network, which consists of a number of nodes and some arcs connecting these nodes. The network corresponding to (1.7) has n nodes; one for each component x_i . Entry A_{ij} corresponds to the arc from node j to node i . In terms of graph theory such a network is called a directed graph ('directed' because the individual arcs between the nodes are one-way arrows). Therefore the arcs corresponding to A_{ij} and A_{ji} , if both exist, are considered to be different.

The nodes in the network can perform certain activities; each node has its own activity. Such activities take a finite time, called the activity time, to be performed. These activity times may be different for different nodes. It is assumed that an activity at a certain node can only start when all preceding ('directly upstream') activities have finished their activities and sent the results of these activities along the output channel for node j and simultaneously as an input channel for node i . Those that this node i starts its activity as soon as all preceding nodes have sent their results (the rather neutral word 'results' is used, it could equally have been 'pages, ingredients or products, etc.') to node i , then (1.4) describes when the activities take place. The interpretation of the quantities used is:

$x_i(k)$ is the earliest epoch at which node i becomes active for the k -th time;

A_{ij} is the sum of the activity time of node j and the traveling time from node j to node i (the rather neutral expression 'traveling time' is used instead of, for instance, 'transportation time' or 'communication time').

Fact that we write A_{ij} rather than A_{ji} for a quantity connected to the arc from node j to node i has to do with matrix equations which will be written in the classical way with column vectors, as will be seen later on. For the example given above, the network has two nodes and four arcs, as given in Figure 1.1. The interpretation of the number 3 in this figure is that if node 1 has started an activity, the next

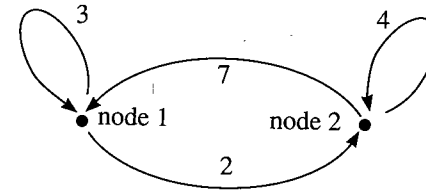


Figure 1.1: Network corresponding to Equation (1.5)

activity cannot start within the next 3 time units. Similarly, the time between two subsequent activities of node 2 is at least 4 time units. Node 1 sends its results to node 2 and once an activity starts in node 1, it takes 2 time units before the result of this activity reaches node 2. Similarly it takes 7 time units after the initiation of an activity of node 2 for the result of that activity to reach node 1. Suppose that an activity refers to some production. The production time of node 1 could for instance be 1 time unit; after that, node 1 needs 2 time units for recovery (lubrication say) and the traveling time of the result (the final product) from node 1 to node 2 is 1 time unit. Thus the number $A_{11} = 3$ is made up of a production time 1 and a recovery time 2 and the number $A_{21} = 2$ is made up of the same production time 1 and a traveling time 1. Similarly, if the production time at node 2 is 4, then this node does not need any time for recovery (because $A_{22} = 4$), and the traveling time from node 2 to node 1 is 3 (because $A_{12} = 7 = 4 + 3$).

If we now look at the sequence (1.7) again, the interpretation of the vectors $x(k)$ is different from the initial one. The argument k is no longer a time instant but a counter which states how many times the various nodes have been active. At time 14, node 1 has been active twice (more precisely, node 1 has started two activities, respectively at times 7 and 11). At the same time 14, node 2 has been active three times (it started activities at times 4, 9 and 13). The counting of the activities is such that it coincides with the argument of the x vector. The initial condition is henceforth considered to be the zeroth activity.

In Figure 1.1 there was an arc from any node to any other node. In many networks referring to more practical situations, this will not be the case. If there is no arc from node j to node i , then node i does not need any result from node j . Therefore node j does not have a direct influence on the behavior of node i . In such a situation it is useful to consider the entry A_{ij} to be equal to $-\infty$. In (1.4) the term $-\infty + x_j(k)$ does not influence $x_i(k+1)$ as long as $x_j(k)$ is finite. The number $-\infty$ will occur frequently in what follows and will be indicated by ε .

For reasons which will become clear later on, Equation (1.4) will be written as

$$x_i(k+1) = \bigoplus_j A_{ij} \otimes x_j(k), \quad i = 1, \dots, n,$$

or in vector notation,

$$x(k+1) = A \otimes x(k). \quad (1.8)$$

The symbol $\bigoplus_j c(j)$ refers to the maximum of the elements $c(j)$ with respect to

appropriate j , and $\otimes$ (pronounced 'o-times') refers to addition. Later on the symbol $\oplus$ (pronounced 'o-plus') will also be used; $a \oplus b$ refers to the maximum of scalars a and b . If the initial condition for (1.8) is $x(0) = x_0$, then

$$\begin{aligned} x(1) &= A \otimes x_0, \\ x(2) &= A \otimes x(1) = A \otimes (A \otimes x_0) = (A \otimes A) \otimes x_0 = A^2 \otimes x_0. \end{aligned}$$

It will be shown in Chapter 3 that indeed $A \otimes (A \otimes x_0) = (A \otimes A) \otimes x_0$. For the example given above it is easy to check this by hand. Instead of $A \otimes A$ we simply calculate A^2 . We obtain

$$x(3) = A \otimes x(2) = A \otimes (A^2 \otimes x_0) = (A \otimes A^2) \otimes x_0 = A^3 \otimes x_0,$$

in general

$$x(k) = \underbrace{(A \otimes A \otimes \cdots \otimes A)}_{k \text{ times}} \otimes x_0 = A^k \otimes x_0.$$

matrices $A^2, A^3, \dots$, can be calculated directly. Let us consider the A -matrix (1.5) again, then

$$A^2 = \begin{pmatrix} \max(3+3, 7+2) & \max(3+7, 7+4) \\ \max(2+3, 4+2) & \max(2+7, 4+4) \end{pmatrix} = \begin{pmatrix} 9 & 11 \\ 6 & 9 \end{pmatrix}.$$

in general

$$(A^2)_{ij} = \bigoplus_l A_{il} \otimes A_{lj} = \max_l (A_{il} + A_{lj}). \quad (1.9)$$

An extension of (1.8) is

$$\begin{cases} x(k+1) = (A \otimes x(k)) \oplus (B \otimes u(k)), \\ y(k) = C \otimes x(k). \end{cases} \quad (1.10)$$

symbol $\oplus$ in this formula refers to componentwise maximization. The m -vector u is called the input to the system; the p -vector y is the output of the system. The components of u refer to nodes which have no predecessors. Similarly, the components of y refer to nodes with no successors. The components of x now refer to internal nodes, i.e. to nodes which have both successors and predecessors. The matrices $B = \{B_{ij}\}$ and $C = \{C_{ij}\}$ have sizes $n \times m$ and $p \times n$, respectively. The usual way of writing (1.10) would be

$$\begin{aligned} x_i(k+1) &= \max(A_{i1} + x_1(k), \dots, A_{in} + x_n(k), \\ &\quad B_{i1} + u_1(k), \dots, B_{im} + u_m(k)), \quad i = 1, \dots, n; \\ y_i(k) &= \max(C_{i1} + x_1(k), \dots, C_{in} + x_n(k)), \quad i = 1, \dots, p. \end{aligned}$$

Equation (1.10) is written as

$$\begin{cases} x(k+1) = A \otimes x(k) \oplus B \otimes u(k), \\ y(k) = C \otimes x(k), \end{cases} \quad (1.11)$$

where it is understood that multiplication has priority over addition. Usually, however, (1.10) is written as

$$\begin{cases} x(k+1) = Ax(k) \oplus Bu(k), \\ y(k) = Cx(k). \end{cases} \quad (1.12)$$

If it is clear where the ' $\otimes$ '-symbols are used, they are sometimes omitted, as shown in (1.12). This practice is exactly the same one as with respect to the more common multiplication ' $\times$ ' or ' $\cdot$ ' symbol in conventional algebra. In the same vein, in conventional algebra $1 \times x$ is the same as $1x$, which is usually written as x . Within the context of the $\otimes$ and $\oplus$ symbols, $0 \otimes x$ is exactly the same as x . The symbol ε is the neutral element with respect to maximization; its numerical value equals $-\infty$. Similarly, the symbol e denotes the neutral element with respect to addition; it assumes the numerical value 0. Also note that $1 \otimes x$ is different from x .

If one wants to think in terms of a network again, then $u(k)$ is a vector indicating when certain resources become available for the k -th time. Subsequently it takes B_{ij} time units before the j -th resource reaches node i of the network. The vector $y(k)$ refers to the epoch at which the final products of the network are delivered to the outside world.

Take for example

$$\begin{cases} x(k+1) = \begin{pmatrix} 3 & 7 \\ 2 & 4 \end{pmatrix} x(k) \oplus \begin{pmatrix} \varepsilon \\ 1 \end{pmatrix} u(k), \\ y(k) = \begin{pmatrix} 3 & \varepsilon \end{pmatrix} x(k). \end{cases} \quad (1.13)$$

The corresponding network is shown in Figure 1.2. Because $B_{11} = \varepsilon (= -\infty)$, the

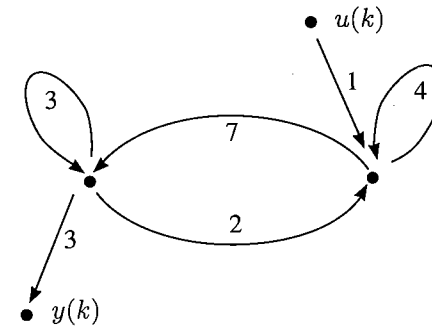


Figure 1.2: Network with input and output

input $u(k)$ only goes to node 2. If one were to replace B by $\begin{pmatrix} 2 & 1 \end{pmatrix}'$ for instance, where the prime denotes transposition, then each input would 'spread' itself over

two nodes. In this example from epoch $u(k)$ on, it takes 2 time units for the input to reach node 1 and 1 time unit to reach node 2. In many practical situations the input will enter the network through one node. That is why, as in this example, the one B -entry per column is different from ε . Similar remarks can be made with respect to the output. Suppose that we have (1.6) as an initial condition and that

$$u(0) = 1, \quad u(1) = 7, \quad u(2) = 13, \quad u(3) = 19, \dots,$$

it easily follows that

$$x(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad x(1) = \begin{pmatrix} 7 \\ 4 \end{pmatrix}, \quad x(2) = \begin{pmatrix} 11 \\ 9 \end{pmatrix}, \quad x(3) = \begin{pmatrix} 16 \\ 14 \end{pmatrix}, \dots,$$

$$y(0) = 4, \quad y(1) = 10, \quad y(2) = 14, \quad y(3) = 19, \dots$$

We started this section with the difference equation (1.1), which is a first-order vector difference equation. It is well known that a higher order linear scalar difference equation

$$z(k+1) = a_1 z(k) + a_2 z(k-1) + \dots + a_n z(k-n+1) \quad (1.14)$$

can be written in the form of Equation (1.1). If we introduce the vector $(z(k), z(k-1), \dots, z(k-n+1))'$, then (1.14) can be written as

$$\begin{pmatrix} z(k+1) \\ z(k) \\ \vdots \\ z(k-n+2) \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & \dots & \dots & a_n \\ 1 & 0 & \dots & \dots & 0 \\ 0 & & & & \\ \vdots & & & & \\ 0 & \dots & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} z(k) \\ z(k-1) \\ \vdots \\ z(k-n+1) \end{pmatrix}. \quad (1.15)$$

Equation (1.14) has exactly the form of (1.1). If we change the operations in (1.14) the standard way, addition becomes maximization and multiplication becomes min; then the numerical evaluation of (1.14) becomes

$$z(k+1) = \max(a_1 + z(k), a_2 + z(k-1), \dots, a_n + z(k-n+1)). \quad (1.16)$$

Equation (1.15) can also be written as a first-order linear vector difference equation. In fact this equation is almost Equation (1.15), which must now be evaluated with the operations maximization and addition. The only difference is that the 1's and 0's in (1.15) must be replaced by ε 's and ε 's, respectively.

MISCELLANEOUS EXAMPLES

In this section, seven examples from different application areas are presented, with a special emphasis on the modeling process. The examples can be read independently. It is shown that all problems formulated lead to equations of the kind (1.8), (1.10), and (1.11). The related ones. Solutions to the problems which are formulated are not given in this

section. To solve these problems, the theory must first be developed and that will be done in the next chapters. Although some of the examples deal with equations with the look of (1.8), the operations used will again be different. The mathematical expressions are the same for many applications. The underlying algebra, however, differs. The emphasis of this book is on these algebras and their relationships.

1.2.1 Planning

Planning is one of the traditional fields in which the max-operation plays a crucial role. In fact, many problems in planning areas are more naturally formulated with the min-operation than with the max-operation. However, one can easily switch from minimization to maximization and vice versa. Two applications will be considered in this subsection; the first one is the shortest path problem, the second one is a scheduling problem. Solutions to such problems have been known for some time, but here the emphasis is on the notation introduced in §1.1 and on some analysis pertaining to this notation.

1.2.1.1 Shortest Path

Consider a network of n cities; these cities are the nodes in a network. Between some cities there are road connections; the distance between city j and city i is indicated by A_{ij} . A road corresponds to an arc in the network. If there is no road from j to i , then we set $A_{ij} = \varepsilon$. In this example $\varepsilon = +\infty$; nonexisting roads get assigned a value $+\infty$ rather than $-\infty$. The reason is that we will deal with minimization rather than maximization. Owing to the possibility of one-way traffic, it is allowed that $A_{ij} \neq A_{ji}$. Matrix A is defined as $A = (A_{ij})$.

The entry A_{ij} denotes the distance between j and i if only one link is allowed. Sometimes it may be more advantageous to go from j to i via k . This will be the case if $A_{ik} + A_{kj} < A_{ij}$. The shortest distance from j to i using exactly two links is

$$\min_{k=1, \dots, n} (A_{ik} + A_{kj}). \quad (1.17)$$

When we use the shorthand symbol $\oplus$ for the minimum operation, then (1.17) becomes

$$\bigoplus_k A_{ik} \otimes A_{kj}.$$

Note that $\oplus$ has been used for both the maximum and the minimum operation. It should be clear from the context which is meant. The symbol $\oplus$ will be used similarly. The reason for not distinguishing between these two operations is that $(\mathbb{R} \cup \{-\infty\}, \max, +)$ and $(\mathbb{R} \cup \{+\infty\}, \min, +)$ are isomorphic algebraic structures. Chapters 3 and 4 will deal with such structures. It is only when the operations max and min appear in the same formula that this convention would lead to ambiguity. This situation will occur in Chapter 9 and different symbols for the two operations will be used there. Expression (1.17) is the ij -th entry of A^2 :

$$(A^2)_{ij} = \bigoplus_k A_{ik} \otimes A_{kj}.$$

te that the expression A^2 can have different meanings also. In (1.9) the max-
operation was used whereas the min-operation is used here.

If one is interested in the shortest path from j to i using one or two links, then
length of the shortest path becomes

$$(A \oplus A^2)_{ij} .$$

ve continue, and if one, two or three links are allowed, then the length of the
test path from j to i becomes

$$(A \oplus A^2 \oplus A^3)_{ij} ,$$

re $A^3 = A^2 \otimes A$, and so on for more than three links. We want to find the
test path whereby any number of links is allowed. It is easily seen that a road
nection consisting of more than $n - 1$ links can never be optimal. If it were
mal, the traveler would visit one city at least twice. The road from this city
itself forms a part of the total road connection and is called a circuit. Since it
actly) assumed that all distances are nonnegative, this circuit adds to the total
ance and can hence be disregarded. The conclusion is that the length of the
test path from j to i is given by

$$(A \oplus A^2 \oplus \dots \oplus A^{n-1})_{ij} .$$

ivalently one can use the following infinite series for the shortest path (the terms
 $k \geq n$ do not contribute to the sum):

$$A^+ \stackrel{\text{def}}{=} A \oplus A^2 \oplus \dots \oplus A^n \oplus A^{n+1} \oplus \dots . \quad (1.18)$$

matrix A^+ , sometimes referred to as the *shortest path matrix*, also shows up in
cheduling problem that we define below shortly.

Note that $(A^+)_{ii}$ refers to a path which first leaves node i and then comes back
. If one wants to include the possibility of staying at a node, then the shortest
matrix should be defined as $e \oplus A^+$, where e denotes the identity matrix of the
size as A . An identity matrix in this set-up has zeros on the diagonal and the
entries have the value $+\infty$. In general, e is a unit matrix of appropriate size.
The shortest path problem can also be formulated according to a difference
tion of the form (1.8). To that end, consider an $n \times n$ matrix X : the ij -th entry
refers to a connection from city j to city i , $X_{ij}(k)$ is the minimum length with
ct to all roads from j to i with k links. Then it is not difficult to see that this
or satisfies the equation

$$X_{ij}(k) = \min_{l=1, \dots, n} (X_{il}(k-1) + A_{lj}) , \quad i, j = 1, \dots, n . \quad (1.19)$$

ally this equation can be written as

$$X(k) = X(k-1)A = X(k-1) \otimes A ,$$

but it cannot be seen from this equation that the operations to be used are mini-
mization and addition. The principle of dynamic programming can be recognized
in (1.19). The following formula gives exactly the same results as (1.19):

$$X_{ij}(k) = \min_{l=1, \dots, n} (A_{il} + X_{lj}(k-1)) , \quad i, j = 1, \dots, n .$$

The difference between this formula and (1.19) is that one uses the principle of
forward dynamic programming and the other one uses backward dynamic program-
ming.

1.2.1.2 Scheduling

Consider a project which consists of various tasks. Some of these tasks cannot
be started before some others have been finished. The dependence of these tasks
can be given in a directed graph in which each node coincides with a task (or,
equivalently, with an activity). As an example, consider the graph of Figure 1.3.
There are six nodes, numbered 1, ..., 6. Node 1 represents the initial activity and

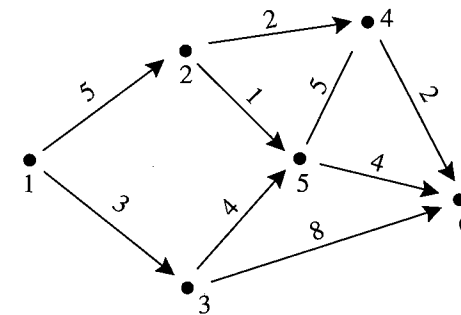


Figure 1.3: Ordering of activities in a project

node 6 represents the final activity. It is assumed that the activities, except the final
one, take a certain time to be performed. In addition, there may be traveling times.
The fact that the final activity has a zero cost is not a restriction. If it were to
have a nonzero cost, a fictitious node 7 could be added to node 6. Node 7 would
represent the final activity. The arcs between the nodes in Figure 1.3 denote the
precedence constraints. For instance, node 4 cannot start before nodes 2 and 5 have
finished their activities. The number A_{ij} associated with the arc from node j to
node i denotes the minimum time that should elapse between the beginning of an
activity at node j and the beginning of an activity at node i .

By means of the principle of dynamic programming it is not difficult to calculate
the *critical path* in the graph. Critical here refers to 'slowest'. The total duration of
the overall project cannot be smaller than the summation of all numbers A_{ij} along
the critical path.

Another way of finding the time at which the activity at node i can start at the
earliest, which will be denoted x_i , is the following. Suppose activity 1 can start at

epoch u at the earliest. This quantity u is an input variable which must be given from the outside. Hence $x_1 = u$. For the other x_i 's we can write

$$x_i = \max_{j=1,\dots,6} (A_{ij} + x_j) .$$

If there is no arc from node i to j , then A_{ij} gets assigned the value $\varepsilon (= -\infty)$. If $x = (x_1, \dots, x_6)'$ and $A = (A_{ij})$, then we can compactly write

$$x = Ax \oplus Bu , \quad (1.20)$$

where

$$A = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 2 & \varepsilon & \varepsilon & 5 & \varepsilon \\ \varepsilon & 1 & 4 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 8 & 2 & 4 & \varepsilon \end{pmatrix}, \quad B = \begin{pmatrix} e \\ \varepsilon \\ \varepsilon \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix} .$$

Note that e in B equals 0 in this context. Here we recognize the form of (1.11), although in (1.20) time does not play a role; x in the left-hand side equals the x in the right-hand side. Hence (1.20) is an implicit equation for the vector x . Let us see what we obtain by repeated substitution of the complete right-hand side of (1.20) into x of this same right-hand side. After one substitution:

$$\begin{aligned} x &= A^2x \oplus ABu \oplus Bu \\ &= A^2x \oplus (A \oplus e)Bu , \end{aligned}$$

and after n substitutions:

$$x = A^n x \oplus (A^{n-1} \oplus A^{n-2} \oplus \dots \oplus A \oplus e)Bu .$$

In the formulæ above, e refers to the identity matrix; zeros on the diagonal and ε 's elsewhere. The symbol e will be used as the identity element for all spaces that will be encountered in this book. Similarly, ε will be used to denote the zero element of any space to be encountered.

Since the entries of A^n denote the weights of paths with length n in the corresponding graph and A does not have paths of length greater than 4, we get $A^n = -\infty$ for $n \geq 5$. Therefore the solution x in the current example becomes

$$x = (A^4 \oplus A^3 \oplus A^2 \oplus A \oplus e)Bu , \quad (1.21)$$

for which we can write

$$x = (e \oplus A^+)Bu ,$$

where A^+ was defined in (1.18).

In (1.21), we made use of the series

$$A^* \stackrel{\text{def}}{=} e \oplus A \oplus \dots \oplus A^n \oplus A^{n+1} \oplus \dots , \quad (1.22)$$

although it was concluded that $A^k, k > n$, does not contribute to the sum. With the conventional matrix calculus in mind one might be tempted to write for (1.22):

$$(e \oplus A \oplus A^2 \oplus \dots) = (e \ominus A)^{-1} . \quad (1.23)$$

Of course, we have not defined the inverse of a matrix within the current setting and so (1.23) is an empty statement. It is also strange to have a 'minus' sign $\ominus$ in (1.23) and it is not known how to interpret this sign in the context of the max-operation at the left-hand side of the equation. It should be the reverse operation of $\oplus$. If we dare to continue along these shaky lines, one could write the solution of (1.20) as

$$(e \ominus A)x = Bu \Rightarrow x = (e \ominus A)^{-1}Bu .$$

Quite often one can guide one's intuition by considering formal expressions of the kind (1.23). One tries to find formal analogies in the notation using conventional analysis. In Chapter 3 it will be shown that an inverse as in (1.23) does not exist in general and therefore we get 'stuck' with the series expansion.

There is a dual way to analyze the critical path of Figure 1.3. Instead of starting at the initial node 1, one could start at the final node 6 and then work backward in time. This latter approach is useful when a target time for the completion of the project has been set. The question then is: what is the latest moment at which each node has to start its activity in such a way that the target time can still be fulfilled? If we call the starting times x_i again, then it is not difficult to see that

$$x_i = \min \left[\min_j \left((\hat{A})_{ij} + x_j \right), (\hat{B})_i + u \right], \quad i = 1, \dots, 6 ,$$

where

$$\hat{A} = \begin{pmatrix} \varepsilon & 5 & 3 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 2 & 1 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 4 & 8 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 2 \\ \varepsilon & \varepsilon & \varepsilon & 5 & \varepsilon & 4 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \quad \hat{B} = \begin{pmatrix} \varepsilon \\ \varepsilon \\ \varepsilon \\ \varepsilon \\ \varepsilon \\ e \end{pmatrix} .$$

It is easily seen that $\hat{A}$ is equal to the transpose of A in (1.20); x_6 has been chosen as the completion time of the project. In matrix form, we can write

$$x = (\hat{A} \odot x) \wedge (\hat{B} \odot u) ,$$

where $\odot$ is a new matrix multiplication using min as addition of scalars and $+$ as multiplication, whereas $\wedge$ is the min of vectors, componentwise. This topic of target times will be addressed in §5.6.

1.2.2 Communication

This subsection focuses on the Viterbi algorithm. It can conveniently be described by a formula of the form (1.1). The operations to be used this time are maximization and multiplication.

The stochastic process of interest in this section, $\nu(k)$, $k \geq 0$, is a time homogeneous Markov chain with state space $\{1, 2, \dots, n\}$, defined on some probability space $(\Omega, \mathbb{F}, \mathbb{P})$. The Markov property means that

$$\mathbb{P}[\nu(k+1) = i_{k+1} \mid \nu(0) = i_0, \dots, \nu(k) = i_k] = \mathbb{P}[\nu(k+1) = i_{k+1} \mid \nu(k) = i_k],$$

where $\mathbb{P}[\mathcal{A} \mid \mathcal{B}]$ denotes the conditional probability of the event $\mathcal{A}$ given the event $\mathcal{B}$ and $\mathcal{A}$ and $\mathcal{B}$ are in $\mathbb{F}$. Let M_{ij} denote the transition probability¹ from state j to i . The initial distribution of the Markov chain will be denoted p .

The process $\nu = (\nu(0), \dots, \nu(K))$ is assumed to be observed with some noise. This means that there exists a sequence of $\{1, 2, \dots, n\}$ -valued random variables $z(k)$, $k = 0, \dots, K$, called the *observation*, and such that $N_{i_k j_k} \stackrel{\text{def}}{=} \mathbb{P}[z(k) = i_k \mid \nu(k) = j_k]$ does not depend on k and such that the joint law of (ν, z) , where $z = (z(0), \dots, z(K))$, is given by the relation

$$\mathbb{P}[\nu = j, z = i] = \left(\prod_{k=1}^K N_{i_k j_k} M_{j_k j_{k-1}} \right) N_{i_0 j_0} p_{j_0}, \quad (1.24)$$

where $i = (i_0, \dots, i_K)$ and $j = (j_0, \dots, j_K)$.

Given such a sequence z of observations, the question to be answered is to find the sequence j for which the probability $\mathbb{P}[\nu = j \mid z]$ is maximal. This problem is a highly simplified version of a text recognition problem. A machine reads handwritten text, symbol after symbol, but makes mistakes (the observation errors). The underlying model of the text is such that after having read a symbol, the probability of the occurrence of the next one is known. More precisely, the sequence of symbols is assumed to be produced by a Markov chain.

We want to compute the quantity

$$x_{j_K}(K) = \max_{j_0, \dots, j_{K-1}} \mathbb{P}[\nu = j, z = i]. \quad (1.25)$$

This quantity is also a function of i , but this dependence will not be made explicit. The argument that achieves the maximum in the right-hand side of (1.25) is the most likely text up to the $(K-1)$ -st symbol for the observation i ; similarly, the argument j_K which maximizes $x_{j_K}(K)$ is the most likely K -th symbol given that the first K observations are i . From (1.24), we obtain

$$\begin{aligned} x_{j_K}(K) &= \max_{j_0, \dots, j_{K-1}} \left(\prod_{k=1}^K (N_{i_k j_k} M_{j_k j_{k-1}}) N_{i_0 j_0} p_{j_0} \right) \\ &= \max_{j_{K-1}} ((N_{i_K j_K} M_{j_K j_{K-1}}) x_{j_{K-1}}(K-1)), \end{aligned}$$

with initial condition $x_{j_0}(0) = N_{i_0 j_0} p_{j_0}$. The reader will recognize the above algorithm as a simple version of (forward) dynamic programming. If $N_{i_k j_k} M_{j_k j_{k-1}}$ is denoted $A_{j_k j_{k-1}}$, then the general formula is

$$x_m(k) = \max_{\ell=1, \dots, n} (A_{m\ell} x_\ell(k-1)), \quad m = 1, \dots, n. \quad (1.26)$$

¹Classically, in Markov chains, M_{ij} would rather be denoted M_{ji} .

Table 1.1: Processing times

	P_1	P_2	P_3
M_1		1	5
M_2	3	2	3
M_3	4	3	

This formula is similar to (1.1) if addition is replaced by maximization and multiplication remains multiplication.

The Viterbi algorithm maximizes $\mathbb{P}[\nu, z]$ as given in (1.24). If we take the logarithm of (1.24), and multiply the result by -1 , (1.26) becomes

$$-\ln(x_m(k)) = \min_{\ell=1, \dots, n} [-\ln(A_{m\ell}) - \ln(x_\ell(k-1))], \quad m = 1, \dots, n.$$

The form of this equation exactly matches (1.19). Thus the Viterbi algorithm is identical to an algorithm which determines the shortest path in a network. Actually, it is this latter algorithm—minimizing $-\ln(\mathbb{P}[\nu \mid z])$ —which is quite often referred to as the Viterbi algorithm, rather than the one expressed by (1.26).

1.2.3 Production

Consider a manufacturing system consisting of three machines. It is supposed to produce three kinds of parts according to a certain product mix. The routes to be followed by each part and each machine are depicted in Figure 1.4 in which $M_i, i = 1, 2, 3$, are the machines and $P_i, i = 1, 2, 3$, are the parts. Processing

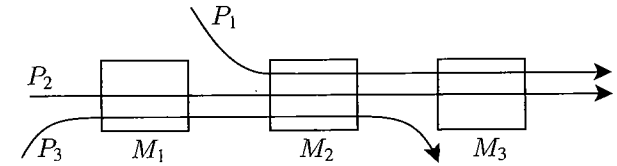


Figure 1.4: Routing of parts along machines

times are given in Table 1.1. Note that this manufacturing system has a flow-shop structure, i.e. all parts follow the same sequence on the machines (although they may skip some) and every machine is visited at most once by each part. We assume that there are no set-up times on machines when they switch from one part type to another. Parts are carried on a limited number of pallets (or, equivalently, product carriers). For reasons of simplicity it is assumed that

1. only one pallet is available for each part type;

2. the final product mix is balanced in the sense that it can be obtained by means of a periodic input of parts, here chosen to be P_1, P_2, P_3 ;
3. there are no set-up times or traveling times;
4. the sequencing of part types on the machines is known and it is (P_2, P_3) on M_1 , (P_1, P_2, P_3) on M_2 and (P_1, P_2) on M_3 .

The last point mentioned is not for reasons of simplicity. If any machine were to start working on the part which arrived first instead of waiting for the appropriate part, the modeling would be different. Manufacturing systems in which machines start working on the first arriving part (if it has finished its current activity) will be dealt with in Chapter 9. We can draw a graph in which each node corresponds to a combination of a machine and a part. Since M_1 works on 2 parts, M_2 on 3 and M_3 on 2, this graph has seven nodes. The arcs between the nodes express the precedence constraints between operations due to the sequencing of operations on the machines. To each node i in Figure 1.5 corresponds a number x_i which denotes the earliest epoch at which the node can start its activity. In order to be able to calculate these quantities, the epochs at which the machines and parts (together called the resources) are available must be given. This is done by means of a six-dimensional input vector u (six since there are six resources: three machines and three parts). There is an output vector also; the components of the six-dimensional vector y denote the epochs at which the parts are ready and the machines have finished their jobs (for one cycle). The model becomes

$$x = Ax \oplus Bu ; \quad (1.27)$$

$$y = Cx , \quad (1.28)$$

in which the matrices are

$$A = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 5 & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon & 4 & \varepsilon \end{pmatrix}; \quad B = \begin{pmatrix} e & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e \\ \varepsilon & e & \varepsilon & e & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix};$$

$$C = \begin{pmatrix} \varepsilon & 5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 4 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon \end{pmatrix}.$$

Equation (1.27) is an implicit equation in x which can be solved as we did in the subsection on Planning;

$$x = A^*Bu .$$

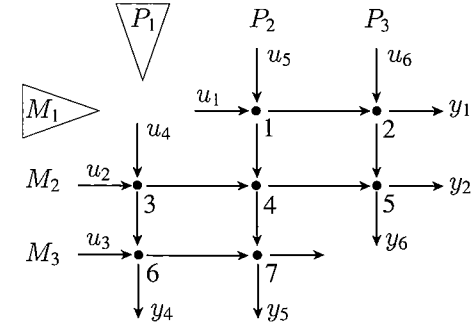


Figure 1.5: The ordering of activities in the flexible manufacturing system

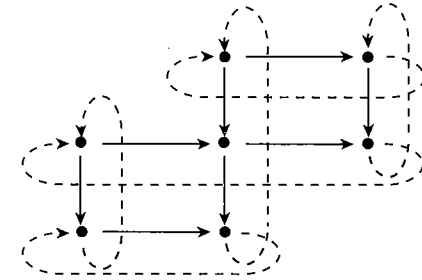


Figure 1.6: Production system with feedback arcs

Now we add feedback arcs to Figure 1.5 as illustrated in Figure 1.6. In this graph the feedback arcs are indicated by dotted lines. The meaning of these feedback arcs is the following. After a machine has finished a sequence of products, it starts with the next sequence. If the pallet on which product P_i was mounted is at the end, the finished product is removed and the empty pallet immediately goes back to the starting point to pick up a new part P_i . If it is assumed that the feedback arcs have zero cost, then $u(k) = y(k-1)$, where $u(k)$ is the k -th input cycle and $y(k)$ the k -th output. Thus we can write

$$\begin{aligned} y(k) &= Cx(k) = CA^*Bu(k) \\ &= CA^*By(k-1) . \end{aligned} \quad (1.29)$$

The transition matrix from $y(k-1)$ to $y(k)$ can be calculated (it can be done by hand, but a simple computer program does the job also):

$$M \stackrel{\text{def}}{=} CA^*B = \begin{pmatrix} 6 & \varepsilon & \varepsilon & \varepsilon & 6 & 5 \\ 9 & 8 & \varepsilon & 8 & 9 & 8 \\ 6 & 10 & 7 & 10 & 6 & \varepsilon \\ \varepsilon & 7 & 4 & 7 & \varepsilon & \varepsilon \\ 6 & 10 & 7 & 10 & 6 & \varepsilon \\ 9 & 8 & \varepsilon & 8 & 9 & 8 \end{pmatrix}.$$

This matrix M determines the speed with which the manufacturing system can work. We will return to this issue in §1.3

1.2.4 Queuing System with Finite Capacity

Let us consider four servers, $S_i, i = 1, \dots, 4$, in series (see Figure 1.7). Each



Figure 1.7: Queuing system with four servers

customer is to be served by S_1, S_2, S_3 and S_4 , and specifically in this order. It takes $\tau_i(k)$ time units for S_i to serve customer k ($k = 1, 2, \dots$). Customer k arrives at epoch $u(k)$ into the buffer associated with S_1 . If this buffer is empty and S_1 is idle, then this customer is served directly by S_1 . Between the servers there are no buffers. The consequence is that if $S_i, i = 1, 2, 3$, has finished serving customer k , but S_{i+1} is still busy serving customer $k-1$, then S_i cannot start serving the new customer $k+1$. He must wait. To complete the description of the queuing system, it is assumed that the traveling times between the servers are zero. Let $x_i(k)$ denote the beginning of the service of customer k by server S_i . Before S_i can start serving customer $k+1$, the following three conditions must be fulfilled:

- S_i must have finished serving customer k ;
- S_{i+1} must be idle (for $i = 4$ this condition is an empty one);
- S_{i-1} must have finished serving customer $k+1$ (for $i = 1$ this condition is an empty one and must be related to the arrival of customer $k+1$ in the queuing system).

It is not difficult to see that the vector x , consisting of the four x -components, satisfies

$$\begin{aligned}
 x(k+1) &= \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \tau_1(k+1) & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \tau_2(k+1) & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \tau_3(k+1) & \varepsilon \end{pmatrix} x(k+1) \\
 &\oplus \begin{pmatrix} \tau_1(k) & e & \varepsilon & \varepsilon \\ \varepsilon & \tau_2(k) & e & \varepsilon \\ \varepsilon & \varepsilon & \tau_3(k) & e \\ \varepsilon & \varepsilon & \varepsilon & \tau_4(k) \end{pmatrix} x(k) \oplus \begin{pmatrix} e \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix} u(k+1). \quad (1.30)
 \end{aligned}$$

We will not discuss issues related to initial conditions here. For those questions, the reader is referred to Chapters 2 and 7. Equation (1.30), which we write formally as

$$x(k+1) = A_2(k+1, k+1)x(k+1) \oplus A_1(k+1, k)x(k) \oplus Bu(k+1),$$

is an implicit equation in $x(k+1)$ which can be solved again, as done before. The result is

$$x(k+1) = (A_2(k+1, k+1))^* (A_1(k+1, k)x(k) \oplus Bu(k+1)),$$

where $(A_2(k+1, k+1))^*$ equals

$$\begin{pmatrix} e & \varepsilon & \varepsilon & \varepsilon \\ \tau_1(k+1) & e & \varepsilon & \varepsilon \\ \tau_1(k+1)\tau_2(k+1) & \tau_2(k+1) & e & \varepsilon \\ \tau_1(k+1)\tau_2(k+1)\tau_3(k+1) & \tau_2(k+1)\tau_3(k+1) & \tau_3(k+1) & e \end{pmatrix}.$$

The customers who arrive in the queuing system and cannot directly be served by S_1 , wait in the buffer associated with S_1 . If one is interested in the buffer contents, i.e. the number of waiting customers, at a certain moment, one should use a counter (of customers) at the entry of the buffer and one at the exit of the buffer. The difference of the two counters yields the buffer contents, but this operation is nonlinear in the max-plus algebra framework. In §1.2.6 we will return to the ‘counter’-description of discrete event systems. The counters just mentioned are nondecreasing with time, whereas the buffer contents itself is fluctuating as a function of time.

The design of buffer sizes is a basic problem in manufacturing systems. If the buffer contents tends to go to ∞ , one speaks of an unstable system. Of course, an unstable system is an example of a badly designed system. In the current example, buffering between the servers was not allowed. Finite buffers can also be modeled within the max-plus algebra context as shown in the next subsection and more generally in §2.6.2. Another useful parameter is the utilization factor of a server. It is defined by the ‘busy time’ divided by the total time elapsed.

Note that we did not make any assumptions on the service time $\tau_i(k)$. If one is faced with unpredictable breakdowns (and subsequently a repair time) of the servers, then the service times might be modeled stochastically. For a deterministic and invariant (‘customer invariant’) system, the serving times do not, by definition, depend on the particular customer.

1.2.5 Parallel Computation

The application of this subsection belongs to the field of VLSI array processors (VLSI stands for ‘Very Large Scale Integration’). The theory of discrete events provides a method for analyzing the performances of so-called systolic and wavefront array processors. In both processors, the individual processing nodes are connected together in a nearest neighbor fashion to form a regular lattice. In the application to be described, all individual processing nodes perform the same basic operation.

The difference between systolic and wavefront array processors is the following. In a systolic array processor, the individual processors, i.e. the nodes of the network, operate synchronously and the only clock required is a simple global clock. The wavefront array processor does not operate synchronously, although the required processing function and network configuration are exactly the same as for the systolic

processor. The operation of each individual processor in the wavefront case is controlled locally. It depends on the necessary input data available and on the output of the previous cycle having been delivered to the appropriate (i.e. directly downstream) nodes. For this reason a wavefront array processor is also called a data-driven net.

We will consider a network in which the execution times of the nodes (the individual processors) depend on the input data. In the case of a simple multiplication, the difference in execution time is a consequence of whether at least one of the operands is a zero or a one. We assume that if one of the operands is a zero or a one, the multiplication becomes trivial and, more importantly, faster. Data driven networks are at least as fast as systolic networks since in the latter case the period of the synchronization clock must be large enough to include the slowest local cycle or largest execution time.

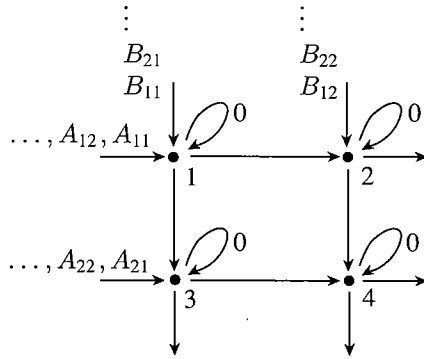


Figure 1.8: The network which multiplies two matrices

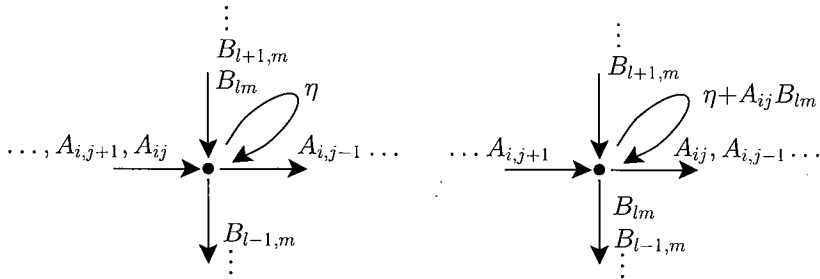


Figure 1.9: Input and output of individual node, before and after one activity

Consider the network shown in Figure 1.8. In this network four nodes are connected. Each of these nodes has an input/output behavior as given in Figure 1.9. The purpose of this network is to multiply two matrices A and B ; A has size $2 \times n$

and B has size $n \times 2$. The number n is large ($\gg 2$), but otherwise arbitrary. The entries of the rows of A are fed into the network as indicated in Figure 1.8; they enter the two nodes on the left. Similarly, the entries of B enter the two top nodes. Each node will start its activity as soon as each of the input channels contains a data item, provided it has finished its previous activity and sent the output of that previous activity downstream. Note that in the initial situation—see Figure 1.8—not all channels contain data. Only node 1 can start its activity immediately. Also note that initially all loops contain the number zero.

The activities will stop when all entries of A and B have been processed. Then each of the four nodes contains an entry of the product AB . Thus for instance, node 2 contains $(AB)_{12}$. It is assumed that each node has a local memory at each of its input channels which functions as a buffer and in which at most one data item can be stored if necessary. Such a necessity arises when the upstream node has finished an activity and wants to send the result downstream while the downstream node is still busy with an activity. If the buffer is empty, the output of the upstream node can temporarily be stored in this buffer and the upstream node can start a new activity (provided the input channels contain data). If the upstream node cannot send out its data because one or more of the downstream buffers are full, then it cannot start a new activity (it is 'blocked').

Since it is assumed that each node starts its activities as soon as possible, the network of nodes can be referred to as a wavefront array processor. The execution time of a node is either τ_1 or τ_2 units of time. It is τ_1 if at least one of the input items, from the left and from above (A_{ij} and B_{ij}), is a zero or a one. Then the product to be performed becomes a trivial one. The execution time is τ_2 if neither input contains a zero or a one.

It is assumed that the entry A_{ij} of A equals zero or one with probability p , $0 \leq p \leq 1$, and that A_{ij} is neither zero nor one with probability $1 - p$. The entries of B are assumed to be neither zero nor one (or, if such a number would occur, it will not be detected and exploited).

If $x_i(k)$ is the epoch at which node i becomes active for the k -th time, then it follows from the description above that

$$\left. \begin{aligned} x_1(k+1) &= \alpha_1(k)x_1(k) \oplus x_2(k-1) \oplus x_3(k-1), \\ x_2(k+1) &= \alpha_1(k)x_2(k) \oplus \alpha_1(k+1)x_1(k+1) \oplus x_4(k-1), \\ x_3(k+1) &= \alpha_2(k)x_3(k) \oplus \alpha_1(k+1)x_1(k+1) \oplus x_4(k-1), \\ x_4(k+1) &= \alpha_2(k)x_4(k) \oplus \alpha_1(k+1)x_2(k+1) \\ &\quad \oplus \alpha_2(k+1)x_3(k+1). \end{aligned} \right\} \quad (1.31)$$

In these equations, the coefficients $\alpha_i(k)$ are either τ_1 (if the entry is either a zero or a one) or τ_2 (otherwise); $\alpha_i(k+1)$ have the same meaning with respect to the next entry. Systems of this type will be considered in Chapter 7.

There is a correlation among the coefficients of (1.31) during different time steps. By replacing $x_2(k)$ and $x_3(k)$ by $x_2(k+1)$ and $x_3(k+1)$, respectively, $x_4(k)$ by

$x_4(k+2)$ and $\alpha_2(k)$ by $\alpha_2(k+1)$, we obtain

$$\left. \begin{aligned} x_1(k+1) &= \alpha_1(k)x_1(k) \oplus x_2(k) \oplus x_3(k) , \\ x_2(k+1) &= \alpha_1(k-1)x_2(k) \oplus \alpha_1(k)x_1(k) \oplus x_4(k) , \\ x_3(k+1) &= \alpha_2(k)x_3(k) \oplus \alpha_1(k)x_1(k) \oplus x_4(k) , \\ x_4(k+1) &= \alpha_2(k-1)x_4(k) \oplus \alpha_1(k-1)x_2(k) \oplus \alpha_2(k)x_3(k) . \end{aligned} \right\} \quad (1.32)$$

The correlation between some of the α_i -coefficients still exists. The standard procedure to avoid problems connected to this correlation is to augment the state vector x . Two new state variables are introduced: $x_5(k+1) = \alpha_1(k)$ and $x_6(k+1) = \alpha_2(k)$. Equation (1.32) can now be written as

$$\left. \begin{aligned} x_1(k+1) &= \alpha_1(k)x_1(k) \oplus x_2(k) \oplus x_3(k) , \\ x_2(k+1) &= x_5(k)x_2(k) \oplus \alpha_1(k)x_1(k) \oplus x_4(k) , \\ x_3(k+1) &= \alpha_2(k)x_3(k) \oplus \alpha_1(k)x_1(k) \oplus x_4(k) , \\ x_4(k+1) &= x_6(k)x_4(k) \oplus x_5(k)x_2(k) \oplus \alpha_2(k)x_3(k) , \\ x_5(k+1) &= \alpha_1(k) , \\ x_6(k+1) &= \alpha_2(k) . \end{aligned} \right\} \quad (1.33)$$

The correlation in time of the coefficients α_i has disappeared at the expense of a larger state vector. Also note that Equation (1.33) has terms $x_j(k) \otimes x_l(k)$, which cause the equation to become nonlinear (actually, bilinear). For our purposes of calculating the performance of the array processor, this does not constitute basic difficulties, as will be seen in Chapter 8. Equation (1.32) is non-Markovian and linear, whereas (1.33) is Markovian and nonlinear.

1.2.6 Traffic

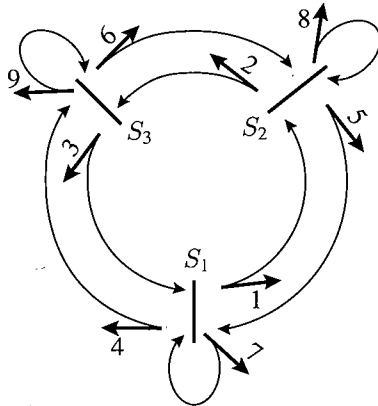


Figure 1.10: The railway system, routing no. 1

In a metropolitan area there are three railway stations, S_1 , S_2 , and S_3 , which are connected by a railway system as indicated in Figure 1.10. The railway system

consists of two inner circles, along which the trains run in opposite direction, and of three outer circles. The trains on these outer circles deliver and pick up passengers at local stations. These local stations have not been indicated in Figure 1.10 since they do not play any role in the problem to be formulated.

There are nine railway tracks. Track $S_i S_j$ denotes the direct railway connection from station S_i to station S_j ; track $S_i S_i$ denotes the outer circle connected to station S_i . Initially, a train runs along each of these nine tracks. At each station the trains must wait for the trains arriving from the other directions (except for the trains coming from the direction the current train is heading for) in order to allow for transfers. Another assumption to be satisfied is that trains on the same circle cannot bypass one another. If $x_i(k)$ denotes the k -th departure time of the train in direction i (see Figure 1.10) then these departure times are described by $x(k+1) = A_1 \otimes x(k)$, where A_1 is the 9×9 matrix

$$A_1 = \begin{pmatrix} e & \varepsilon & s_{31} & \varepsilon & \varepsilon & \varepsilon & s_{11} & \varepsilon & \varepsilon \\ s_{12} & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & s_{22} & \varepsilon \\ \varepsilon & s_{23} & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & s_{33} \\ \varepsilon & \varepsilon & \varepsilon & e & s_{21} & \varepsilon & s_{11} & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & e & s_{32} & \varepsilon & s_{22} & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & s_{13} & \varepsilon & e & \varepsilon & \varepsilon & s_{33} \\ \varepsilon & \varepsilon & s_{31} & \varepsilon & s_{21} & \varepsilon & s_{11} & \varepsilon & \varepsilon \\ s_{12} & \varepsilon & \varepsilon & \varepsilon & \varepsilon & s_{32} & \varepsilon & s_{22} & \varepsilon \\ \varepsilon & s_{23} & \varepsilon & s_{13} & \varepsilon & \varepsilon & \varepsilon & \varepsilon & s_{33} \end{pmatrix}.$$

An entry s_{ij} refers to the traveling time on track $S_i S_j$. These quantities include transfer times at the stations. The diagonal entries e prevent trains from bypassing one another on the same track at a station. The routing of the trains was according to Figure 1.10; trains on the two inner circles stay on these inner circles and keep the same direction; trains on the outer circles remain there. Other routings of the trains are possible; two such different routings are given in Figures 1.11 and 1.12. If $x_i(k)$ denotes the k -th departure time from the same station as given in Figure 1.10, then the departure times are described again by a model of the form $x(k+1) = A \otimes x(k)$. The A -matrix corresponding to Figure 1.11 is indicated by A_2 and the A -matrix corresponding to Figure 1.12 by A_3 . If we define matrices F_i of size 3×3 in the following way:

$$F_1 = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \quad F_2 = \begin{pmatrix} \varepsilon & \varepsilon & s_{31} \\ s_{12} & \varepsilon & \varepsilon \\ \varepsilon & s_{23} & \varepsilon \end{pmatrix},$$

$$F_3 = \begin{pmatrix} s_{11} & \varepsilon & \varepsilon \\ \varepsilon & s_{22} & \varepsilon \\ \varepsilon & \varepsilon & s_{33} \end{pmatrix}, \quad F_4 = \begin{pmatrix} \varepsilon & s_{21} & \varepsilon \\ \varepsilon & \varepsilon & s_{32} \\ s_{13} & \varepsilon & \varepsilon \end{pmatrix},$$

then the matrices A_1 , A_2 and A_3 can be compactly written as

$$A_1 = \begin{pmatrix} e \oplus F_2 & F_1 & F_3 \\ F_1 & e \oplus F_4 & F_3 \\ F_2 & F_4 & F_3 \end{pmatrix}, \quad A_2 = \begin{pmatrix} e \oplus F_2 & F_4 & F_3 \\ F_2 & e \oplus F_4 & F_3 \\ F_2 & F_4 & F_3 \end{pmatrix},$$

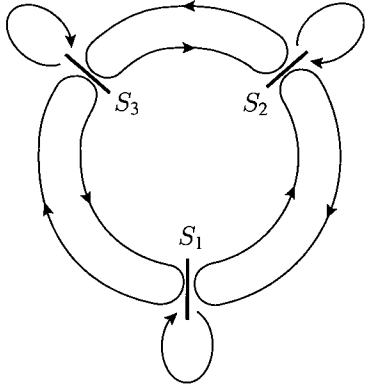


Figure 1.11: Routing no. 2

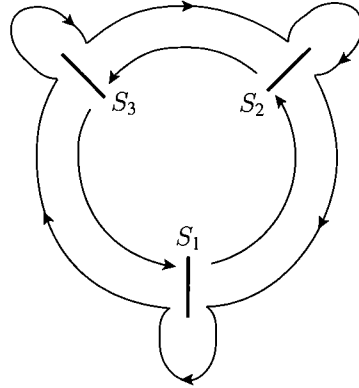


Figure 1.12: Routing no. 3

$$A_3 = \begin{pmatrix} e \oplus F_2 & F_1 & F_3 \\ F_1 & e & F_3 \\ F_2 & F_4 & e \end{pmatrix}.$$

It is seen that for different routing schemes, different models are obtained. In general, depending on the numerical values of s_{ij} , the time evolutions of these three models will be different. One needs a criterion for deciding which of the routing schemes is preferable.

Depending on the application, it may be more realistic to introduce stochastic s_{ij} -quantities. Suppose for instance that there is a swing bridge in the railway track from S_3 to S_1 (only in this track; there is no bridge in the track from S_1 to S_3). Each time a train runs from S_3 to S_1 there is a probability p , $0 \leq p \leq 1$, that the train will be delayed, i.e. s_{31} becomes larger. Thus the matrices A_i become k -dependent; $A_i(k)$. The system has become stochastic. In this situation one may also have a preference for one of the three routings, or for another one.

The last part of the discussion of this example will be devoted to the deterministic model of routing no. 1 again. However, to avoid some mathematical subtleties which are not essential at this stage, we assume that there is at least a difference of τ time units between two subsequent departures of trains in the directions 1 to 6 ($\tau > e$). Consequently, the equations are now $x(k+1) = \hat{A}_1 \otimes x(k)$, where $\hat{A}_1$ equals the previous A_1 except for the first six diagonal entries that are replaced by τ (instead of e earlier), and the last three diagonal entries s_{ii} that are replaced by $s_{ii} \oplus \tau$ respectively.

We then introduce a quantity $\chi_i(t)$ which is related to $x_i(k)$. The argument t of $\chi_i(t)$ refers to the actual clock time and $\chi_i(t)$ itself refers to the number of trains departures in the direction i which have occurred up to (and including) time t . The quantity χ_i can henceforth only assume the values $0, 1, 2, \dots$. At an arbitrary time t , the number of trains which left in the direction 1 can exceed at most by one:

- the same number of trains τ units of time earlier;

- the number of trains that left in the direction 3 s_{31} time units earlier (recall that initially a train was already traveling on each track);
- the number of trains that left in the direction 7 s_{11} time units earlier.

Therefore, we have

$$\chi_1(t) = \min (\chi_1(t - \tau) + 1, \chi_3(t - s_{31}) + 1, \chi_7(t - s_{11}) + 1) .$$

For χ_2 one similarly obtains

$$\chi_2(t) = \min (\chi_1(t - s_{12}) + 1, \chi_2(t - \tau) + 1, \chi_8(t - s_2) + 1) ,$$

etc. If all quantities s_{ij} and τ are equal to 1, then one can compactly write

$$\chi(t) = \bar{A} \otimes \chi(t - 1) ,$$

where $\chi(t) = (\chi_1(t), \dots, \chi_9(t))'$, and where the matrix $\bar{A}$ is derived from A_1 by replacing all entries equal to e by $+\infty$ and the other entries by 1. This equation must be read in the min-plus algebra setting. In case we want something more general than s_{ij} and τ equal to 1, we will consider the situation when all these quantities are integer multiples of a positive constant g_c ; then g_c can be interpreted as the time unit along which the evolution will be expressed. One then obtains

$$\chi(t) = \bar{A}_1 \otimes \chi(t - 1) \oplus \bar{A}_2 \otimes \chi(t - 2) \oplus \dots \oplus \bar{A}_l \otimes \chi(t - l) \quad (1.34)$$

for some finite l . The latter equation (in the min-plus algebra) and $x(k+1) = \hat{A}_1 \otimes x(k)$ (in the max-plus algebra) describe the same system. Equation (1.34) is referred to as the *counter description* and the other one as the *dater description*. The word ‘dater’ must be understood as ‘timer’, but since the word ‘time’ and its declinations are already used in various ways in this book, we will stick to the word ‘dater’. The awareness of these two different descriptions for the same problem has far-reaching consequences as will be shown in Chapter 5.

The reader should contemplate that the stochastic problem (in which some of the s_{ij} are random) is more difficult to handle in the counter description, since the delays in (1.34) become stochastic (see §8.2.4).

1.2.7 Continuous System Subject to Flow Bounds and Mixing

So far, the examples have been related to the realm of discrete event systems, and dynamic equations have been obtained under the form of recursive equations involving max (or min) and $+$ -operations. We close this section by showing that a continuous-time system may naturally lead to essentially the same type of equations whenever some flow limitation and mixing phenomena are involved. Had we adopted the point of view of conventional system theory, the model of such a system in terms of differential equations would have exhibited very complex nonlinearities. With the following approach, we will see that this system is ‘linear’ in a precise mathematical sense (see Chapter 6). An analogous discrete event model will be discussed in Example 2.48.

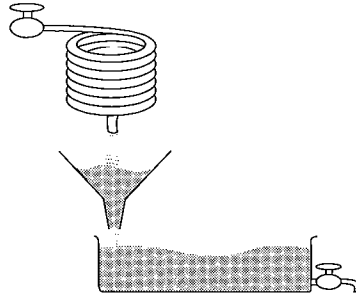


Figure 1.13: A continuous system

In Figure 1.13, a fluid is poured through a long pipe into a first reservoir (empty at time $t = 0$). The input $u(t)$ denotes the *cumulated* flow at the inlet of the pipe up to time t (hence $u(t)$ is a nondecreasing time function and $u(t) = 0$ for $t \leq 0$). It is assumed that it takes a delay of 2 units of time (say, 2 seconds) for the fluid to travel through the pipe. From the first reservoir, the fluid drops into a second reservoir through an aperture which limits the instantaneous flow to a maximum value of, say, 0.5 liter per second. The volume of fluid at time t in this second reservoir is denoted $y(t)$, and it is assumed that $y(0) = 3$ liters.

Let us establish dynamic equations for such a system relating the output y to the input u . Because of the flow limitation into the second reservoir, we have:

$$\forall t, \quad \forall s \geq 0, \quad y(t+s) \leq y(t) + 0.5s. \quad (1.35)$$

On the other hand, since there is a delay of 2 seconds caused by the pipe, $y(t)$ should be compared with $u(t-2)$, and because there is a stock of 3 liters in the second reservoir at $t = 0$, we have:

$$\forall t, \quad y(t) \leq u(t-2) + 3. \quad (1.36)$$

It follows that

$$\begin{aligned} \forall t, \quad \forall s \geq 0, \quad y(t) &\leq y(t-s) + 0.5s \\ &\leq u(t-2-s) + 3 + 0.5s, \end{aligned}$$

hence,

$$\begin{aligned} \forall t, \quad y(t) &\leq \inf_{s \geq 0} [u(t-2-s) + 3 + 0.5s] \\ &= \inf_{\tau \geq 2} [u(t-\tau) + 3 + 0.5(\tau-2)]. \end{aligned} \quad (1.37)$$

Let

$$h(t) \stackrel{\text{def}}{=} \begin{cases} 3 & \text{if } t \leq 2; \\ 3 + 0.5(t-2) & \text{otherwise.} \end{cases} \quad (1.38)$$

and consider

$$\forall t, \quad \bar{y}(t) \stackrel{\text{def}}{=} \inf_{\tau \in \mathbb{R}} [u(t-\tau) + h(\tau)]. \quad (1.39)$$

Indeed, in (1.39), the range of τ may be limited to $\tau \geq 2$ since, for $\tau < 2$, $h(\tau)$ remains equal to 3 whereas $u(t-\tau) \geq u(t-2)$ (remember that $u(\cdot)$ is nondecreasing). Therefore, comparing (1.39) with (1.37), it is clear that $y(t) \leq \bar{y}(t), \forall t$.

Moreover, choosing $\tau = 2$ at the right-hand side of (1.39), we see that $\bar{y}$ satisfies (1.36). In addition, since for all s and all $\vartheta \geq 0$, $h(s+\vartheta) \leq h(s) + 0.5\vartheta$, then

$$\begin{aligned} \forall t, \quad \forall \vartheta \geq 0, \quad \bar{y}(t+\vartheta) &= \inf_{\tau \in \mathbb{R}} [u(t+\vartheta-\tau) + h(\tau)] \\ &= \inf_{s \in \mathbb{R}} [u(t-s) + h(s+\vartheta)] \\ &\leq \inf_{s \in \mathbb{R}} [u(t-s) + h(s)] + 0.5\vartheta \\ &= \bar{y}(t) + 0.5\vartheta. \end{aligned}$$

Thus, $\bar{y}$ satisfies (1.35).

Finally, we have proved that $\bar{y}$ is the maximum solution of (1.35)–(1.36). It can also be checked that (1.39) yields $\bar{y}(t) = 3, \forall t \leq 2$. Therefore, $\bar{y}$ is the solution which will be physically realized if we assume that, subject to (1.37)–(1.39), the fluid flows as fast as possible. This output trajectory is related to the input history u by an ‘inf-convolution’ (see Equation (1.39)). In order to make this inf-convolution more visible, the inf-operator should be viewed as an ‘integration’ (which is nothing else than the $\oplus$ -operator ranging over the real numbers). If moreover $+$ in (1.39) is replaced by $\otimes$ one obtains the appearance of the conventional convolution. The same kind of input-output relationship (indeed, a ‘sup-convolution’ in that context) can be obtained from the recursive equations (1.12) by developing the recursion from any initial condition.

As a final remark, observe that if we have two systems similar to the one shown in Figure 1.13, one producing a red fluid and the other producing a white fluid, and if we want to produce a pink fluid by mixing them in equal proportions, then the new output is related to the two inputs by essentially the same type of equations. More specifically, let $y_r(t)$ and $y_w(t)$ be the quantities of red and white fluids that have been produced in the two downstream reservoirs up to time t (including the initial reserves). Suppose that the two taps at their outlets are opened so that the same (very large) outflow of red and white liquids can be obtained unless one of the two reservoirs is empty, in which case the two taps are closed immediately. Then, $\min(y_r(t), y_w(t))$ is directly related to the quantity $y_p(t)$ of pink fluid produced up to time t . Therefore, this mixing operation does not introduce new mathematical operators.

1.3 ISSUES AND PROBLEMS IN PERFORMANCE EVALUATION

In the previous sections we dealt with equations of the form $x(k+1) = Ax(k)$, or more generally $x(k+1) = A(k)x(k) \oplus B(k)u(k)$. In the applications three different interpretations in terms of the operations were given: maximization and

addition, minimization and addition and lastly, maximization and multiplication. In this section only the first interpretation will be considered (we will say that the system under consideration is in the max-plus algebra framework). Before that a brief introduction to the solution of the conventional equation (1.1) is needed.

Assume that the initial vector (1.2) equals an eigenvector of A ; the corresponding eigenvalue is denoted by λ . The solution of (1.1) can be written as

$$x(t) = \lambda^t x_0, \quad t = 0, 1, \dots \quad (1.40)$$

More generally, if the initial vector can be written as a linear combination of the set of linearly independent eigenvectors,

$$x_0 = \sum_j c_j v_j, \quad (1.41)$$

where v_j is the j -th eigenvector with corresponding eigenvalue λ_j , the c_j are coefficients, then

$$x(t) = \sum_j c_j \lambda_j^t v_j.$$

If the matrix A is diagonalizable, then the set of linearly independent eigenvectors spans $\mathbb{R}^n$, and any initial condition x_0 can be expressed as in (1.41). If A is not diagonalizable, then one must work with generalized eigenvectors and the formula which expresses $x(t)$ in terms of eigenvalues and x_0 is slightly more complicated. This complication does not occur in the max-plus algebra context and therefore will not be dealt with explicitly.

In Chapter 3 it will be shown that under quite general conditions an eigenvalue (λ) and corresponding eigenvector (v) also exist in the max-plus algebra context for a square matrix (A). The definition is

$$A \otimes v = \lambda \otimes v.$$

To exclude degenerate cases, it is assumed that not all components of v are identical to ε . For example,

$$\begin{pmatrix} 3 & 7 \\ 2 & 4 \end{pmatrix} \begin{pmatrix} 2.5 \\ e \end{pmatrix} = 4.5 \begin{pmatrix} 2.5 \\ e \end{pmatrix}.$$

Thus it is seen that the matrix A of (1.5) has an eigenvalue 4.5. Equation (1.40) is also valid in the current setting. If x_0 is an eigenvector of A , with corresponding eigenvalue λ , then the solution of the difference equation (1.8) can be written as

$$x(k) = \lambda^k x_0 \quad (= \lambda^k \otimes x_0), \quad k = 0, 1, \dots \quad (1.42)$$

The numerical evaluation of λ^k in this formula equals $k\lambda$ in conventional analysis. The eigenvalue λ can be interpreted as the cycle time (defined as the inverse of the throughput) of the underlying system; each node of the corresponding network becomes active every λ units of time, since it follows straightforwardly from (1.42).

Also, the relative order in which the nodes become active for the k -th time, as expressed by the components $x_i(k)$, is exactly the same as the relative order in which the nodes become active for the $(k+1)$ -st time. More precisely, Equation (1.42) yields

$$x_l(k+1) - x_j(k+1) = x_l(k) - x_j(k), \quad j, l = 1, \dots, n.$$

Thus the solution (1.42) exhibits a kind of periodicity. Procedures exist for the calculation of eigenvalues and eigenvectors; an efficient one is the procedure known as Karp's algorithm for which the reader is referred to Chapter 2. More discussion about related issues can be found in Chapter 3. Under suitable conditions the eigenvalue turns out to be unique (which differs from the situation in conventional analysis). It can be shown for instance that A of (1.5) has only one eigenvalue. Similarly, the matrix M of §1.2.3 also has a unique eigenvalue:

$$\begin{pmatrix} 6 & \varepsilon & \varepsilon & \varepsilon & 6 & 5 \\ 9 & 8 & \varepsilon & 8 & 9 & 8 \\ 6 & 10 & 7 & 10 & 6 & \varepsilon \\ \varepsilon & 7 & 4 & 7 & \varepsilon & \varepsilon \\ 6 & 10 & 7 & 10 & 6 & \varepsilon \\ 9 & 8 & \varepsilon & 8 & 9 & 8 \end{pmatrix} \begin{pmatrix} e \\ 3 \\ 3.5 \\ 0.5 \\ 3.5 \\ 3 \end{pmatrix} = 9.5 \begin{pmatrix} e \\ 3 \\ 3.5 \\ 0.5 \\ 3.5 \\ 3 \end{pmatrix}.$$

It follows that the eigenvalue equals 9.5, which means in more practical terms that the manufacturing system 'delivers' an item (a product or a machine) at all of its output channels every 9.5 units of time. The eigenvector of this example is also unique, apart from adding the same constant to all components. If v is an eigenvector, then cv , where c is a scalar, also is an eigenvector, since it follows directly from the definition of eigenvalue. It is possible that several eigenvectors can be associated with the only eigenvalue of a matrix, i.e. eigenvectors may not be identical up to an additional constant.

Suppose that we deal with the system characterized by the matrix of (1.5); then it is known from earlier that the 'cycle time' is $9/2$ units of time. The throughput is defined as the inverse of the cycle time and equals $2/9$. If we had the choice of reducing one arbitrary entry of A by 2, which entry should we choose such that the cycle time becomes as small as possible? To put it differently, if a piece of equipment were available which reduces the traveling time at any connection by 2, where should this piece of equipment be placed? By trial and error it is found that either A_{12} or A_{21} should be reduced by 2; in both cases the new cycle time becomes 4. If one reduces A_{11} or A_{22} by this amount instead of A_{12} or A_{21} , then the cycle time remains $9/2$. The consequences of the four potential ways of reduction are expressed by

$$\begin{pmatrix} 1 & 7 \\ 2 & 4 \end{pmatrix} \begin{pmatrix} 2.5 \\ e \end{pmatrix} = 4.5 \begin{pmatrix} 2.5 \\ e \end{pmatrix}; \quad \begin{pmatrix} 3 & 5 \\ 2 & 4 \end{pmatrix} \begin{pmatrix} 1 \\ e \end{pmatrix} = 4 \begin{pmatrix} 1 \\ e \end{pmatrix};$$

$$\begin{pmatrix} 3 & 7 \\ 0 & 4 \end{pmatrix} \begin{pmatrix} 3 \\ e \end{pmatrix} = 4 \begin{pmatrix} 3 \\ e \end{pmatrix}; \quad \begin{pmatrix} 3 & 7 \\ 2 & 2 \end{pmatrix} \begin{pmatrix} 2.5 \\ e \end{pmatrix} = 4.5 \begin{pmatrix} 2.5 \\ e \end{pmatrix}.$$

To answer the question of which 'transportation line' to speed up for more general networks, application of the trial and error method as used above would become very

laborious. Fortunately more elegant and more efficient methods exist. For those one needs the notion of a critical circuit, which is elaborated upon in Chapters 2 and 3. Without defining such a circuit in this section formally let us mention that, in Figure 1.1, this critical circuit consists of the arcs determined by A_{12} and A_{21} . Note that $(A_{12} + A_{21})/2 = \lambda = 9/2$, and this equality is not a coincidence.

Stochastic extensions are possible. Towards that end, consider

$$x(k+1) = A(k)x(k) ,$$

where the matrix A now depends on k in a stochastic way. Assume that $x \in \mathbb{R}^2$ and that for each k the matrix A is one of the following two matrices:

$$\begin{pmatrix} 3 & 7 \\ 2 & 4 \end{pmatrix}, \quad \begin{pmatrix} 3 & 5 \\ 2 & 4 \end{pmatrix}.$$

Both matrices occur with probability $1/2$ and there is no correlation in time. A suitable definition of cycle time turns out to be

$$\lim_{k \rightarrow \infty} \mathbb{E} [x_i(k+1) - x_i(k)] ,$$

where $\mathbb{E}$ denotes mathematical expectation. Application of the theory presented in Chapters 7 and 8 shows that this cycle time is independent of i and is equal to $13/3$.

Conventional linear systems with inputs and outputs are of the form (1.10), although (1.10) itself has the max-plus algebra interpretation. This equation is a representation of a linear system in the time domain. Its representation in the z -domain equals

$$Y(z) = C(zI - A)^{-1}BU(z) ,$$

where $Y(z)$ and $U(z)$ are defined by

$$Y(z) = \sum_{i=0}^{\infty} y(i)z^{-i} , \quad U(z) = \sum_{i=0}^{\infty} u(i)z^{-i} ,$$

where it is tacitly assumed that the system was at rest for $t \leq 0$. The matrix $H(z) \stackrel{\text{def}}{=} C(zI - A)^{-1}B$ is called the transfer matrix of the system. Here I refers to the identity matrix in conventional algebra. The notion of transfer matrix is especially useful when subsystems are combined to build larger systems, by means of parallel, series and feedback connections.

In the max-plus algebra context, the z -transform also exists (see [72]), but here we will rather refer to the γ -transform where γ operates as z^{-1} . For instance, the γ -transform of u is defined as

$$U(\gamma) = \bigoplus_{i=0}^{\infty} u(i) \otimes \gamma^i ,$$

and $Y(\gamma)$ and $X(\gamma)$ are defined likewise. Multiplication of (1.12) by γ^k yields

$$\left. \begin{aligned} \gamma^{-1}x(k+1)\gamma^{k+1} &= A \otimes x(k)\gamma^k \oplus B \otimes u(k)\gamma^k , \\ y(k)\gamma^k &= C \otimes x(k)\gamma^k . \end{aligned} \right\} \quad (1.43)$$

If these equations are summed with respect to $k = 0, 1, \dots$, then we obtain

$$\left. \begin{aligned} \gamma^{-1}X(\gamma) &= A \otimes X(\gamma) \oplus B \otimes U(\gamma) \oplus \gamma^{-1}x_0 , \\ Y(\gamma) &= C \otimes X(\gamma) . \end{aligned} \right\} \quad (1.44)$$

The first of these equations can be solved by first multiplying (max-plus algebra), equivalently adding (conventional) the left- and right-hand sides by γ and then repeatedly substituting the right-hand side for $X(\gamma)$ within this right-hand side. This results in

$$X(\gamma) = (\gamma A)^*(\gamma B U(\gamma) \oplus x_0) .$$

Thus we obtain $Y(\gamma) = H(\gamma)U(\gamma)$, provided that $x_0 = \varepsilon$, and where the transfer matrix $H(\gamma)$ is defined by

$$H(\gamma) = C \otimes (\gamma A)^* \otimes \gamma \otimes B = \gamma C B \oplus \gamma^2 C A B \oplus \gamma^3 C A^2 B \oplus \dots \quad (1.45)$$

The transfer matrix is defined by means of an infinite series and the convergence depends on the value of γ . If the series is convergent for $\gamma = \gamma'$, then it is also convergent for all γ 's which are smaller than γ' . If the series does not converge, it still has a meaning as a formal series.

Exactly as in conventional system theory, the product of two transfer matrices (in which it is tacitly assumed that the sizes of these matrices are such that the multiplication is possible), is a new transfer matrix which refers to a system which consists of the original systems connected in series. In the same way, the sum of two transfer matrices refers to two systems put in parallel. This section will be concluded by an example of such a parallel connection.

We are given two systems. The first one is given in (1.13), and is characterized by the 1×1 transfer matrix

$$H_1 = \varepsilon \gamma \oplus 11\gamma^2 \oplus 14\gamma^3 \oplus 20\gamma^4 \oplus 24\gamma^5 \oplus 29\gamma^6 \oplus \dots$$

It is easily shown that this series converges for $\gamma \leq -4.5$; the number 4.5 corresponds to the eigenvalue of A . The second system is given by

$$x(k+1) = \begin{pmatrix} \varepsilon & \varepsilon & 4 \\ 1 & 1 & \varepsilon \\ \varepsilon & 6 & 3 \end{pmatrix} x(k) \oplus \begin{pmatrix} \varepsilon \\ 2 \\ \varepsilon \end{pmatrix} u(k) ,$$

$$y(k) = (1 \quad 1 \quad 4) x(k) ,$$

and its transfer matrix is

$$H_2 = 4\gamma \oplus 12\gamma^2 \oplus 15\gamma^3 \oplus 18\gamma^4 \oplus 23\gamma^5 \oplus 26\gamma^6 \oplus \dots$$

The transfer matrix of the two systems put in parallel has size 1×1 again (one can talk about a transfer function) and is obtained as

$$H_{\text{par}} = H_1 \oplus H_2 = 4\gamma \oplus 12\gamma^2 \oplus 15\gamma^3 \oplus 20\gamma^4 \oplus 24\gamma^5 \oplus 29\gamma^6 \oplus \dots \quad (1.46)$$

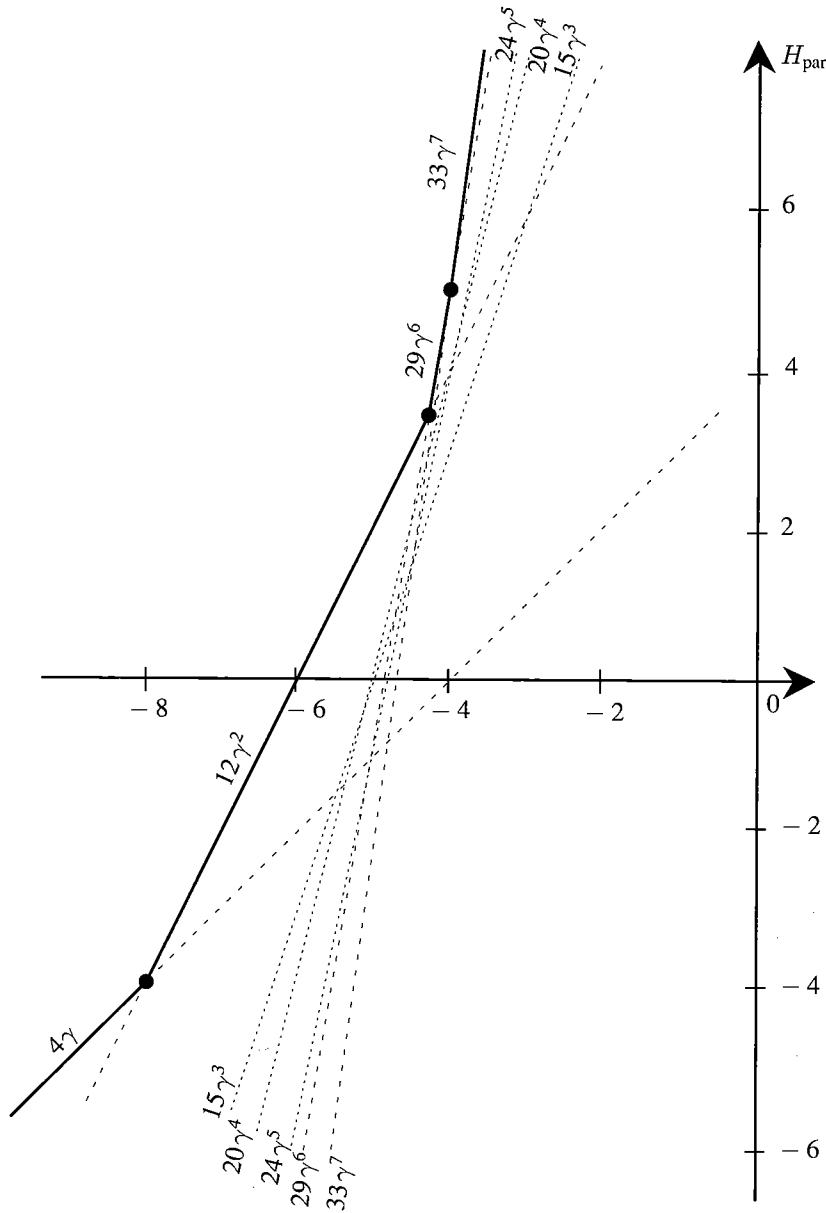


Figure 1.14: The transfer function H_{par} as a function of γ

A transfer function can easily be visualized. If $H(\gamma)$ is a scalar function, i.e. the system has one input and one output, then it is a continuous and piecewise linear function. As an example, the transfer function of the parallel connection considered above is pictured in Figure 1.14.

Above it was shown how to derive the transfer matrix of a system if the representation of the system in the 'time domain' is given. This time domain representation is characterized by the matrices A , B and C . Now one could pose the opposite question: How can we obtain a time domain representation, or equivalently, how can we find A , B and C if the transfer matrix is given? A partial answer to this question is given in [103]. For the example above, one would like to obtain a time domain representation of the two systems put in parallel starting from (1.46). This avenue will not be pursued now. Instead, one can always obtain such a representation by connecting the underlying networks of the two original systems in the appropriate ('parallel') way and then derive the state equations directly. In this way one gets for the above example,

$$x(k+1) = \begin{pmatrix} 3 & 7 & \varepsilon & \varepsilon & \varepsilon \\ 2 & 4 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon & 4 \\ \varepsilon & \varepsilon & 1 & 1 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 6 & 3 \end{pmatrix} x(k) \oplus \begin{pmatrix} \varepsilon \\ 1 \\ \varepsilon \\ 2 \\ e \end{pmatrix} u(k),$$

$$y(k) = (3 \quad \varepsilon \quad 1 \quad 1 \quad 4) x(k).$$

1.4 NOTES

A few times, reference has been made to (linear) system theory. Classic texts are [81], [32] or [72]. A few examples could be phrased in terms of dynamic programming. There are many elementary texts which explain the theory of dynamic programming. A more advanced text is [18]. Systems in the context of max-plus algebra were probably first described in [49], though most theory in this book is algebraically rather than system oriented. It was in [39] where the relation between system theory and max algebra was clearly shown. The shortest path problem is a standard example in dynamic programming texts. The Viterbi algorithm was found in [59]. The example on production was presented in [38], the examples of parallel computation and traffic can be found in [97]. Other simple examples can be found in [101]. A max-plus modeling of the Dutch intercity railway net is given in [27]. An application to chemical batch processing is given in [107]. A good introductory text to design methods of processors for parallel computation is [79]. The relation between eigenvalue and cycle time was developed in [39]. Stochastic extensions were given in [104]. The connection between transfer matrices and state equations in the max-plus algebra context was investigated in [103]; see also §9.2.3.

CHAPTER 2

Graph Theory and Petri Nets

2.1 INTRODUCTION

An overview of various results in the theories of graphs and Petri nets will be given. Several properties of DEDS can be expressed in terms of graph-theoretic notions such as strong connectedness and critical circuits.

A rich relationship exists between graphs and matrices, about which many books have been written. Here, we will emphasize some of these relationships between directed graphs and matrices, together with their consequences if such matrices and graphs are composed to build larger ones. The way to construct such compositions is by means of parallel and series connections.

Petri nets, which describe DEDS pictorially, can be viewed as bipartite graphs. An essential feature of Petri nets, not present in conventional graphs, is that they are dynamic systems. Tokens are used to reflect this dynamic behavior.

There is an equivalence between DEDS without concurrency and a subclass of Petri nets called 'event graphs'. For any timed event graph, we will show how to obtain a mathematical model in terms of recurrence equations. In the proper algebraic framework, these equations are linear and the model offers a strong analogy with conventional linear dynamic systems.

In the last part of this chapter, starting from the point of view of resources involved in DEDS, we propose a methodology to go from the specifications of a concrete system to its modeling by event graphs.

2.2 DIRECTED GRAPHS

A *directed graph* G is defined as a pair $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of elements called *nodes* and where $\mathcal{E}$ is a set the elements of which are ordered (not necessarily different) pairs of nodes, called *arcs*. The possibility of several arcs between two nodes exists (one then speaks about a multigraph); in this chapter, however, we almost exclusively deal with directed graphs in which there is at most one (i.e. zero or one) arc between any two nodes. One distinguishes graphs and directed graphs. The difference between the two is that in a graph the elements of $\mathcal{E}$ are not ordered while they are in a directed graph. Instead of nodes and arcs, one also speaks about vertices and edges, respectively. The origin of the symbols $\mathcal{V}$ and $\mathcal{E}$ in the definition of a (directed) graph is due to the first letters of the latter two names. Instead of directed graph one often uses the shorter word 'digraph', or even 'graph'

if it is clear from the context that digraph is meant. In this chapter we will almost exclusively deal with digraphs (hence also called graphs).

Denote the number of nodes by n , and number the individual nodes $1, 2, \dots, n$. If $(i, j) \in \mathcal{E}$, then i is called the initial node or the origin of the arc (i, j) , and j the final node or the destination of the arc (i, j) . Graphically, the nodes are represented by points, and the arc (i, j) is represented by an 'arrow' from i to j .

We now give a list of concepts of graph theory which will be used later on.

Predecessor, successor. If in a graph $(i, j) \in \mathcal{E}$, then i is called a predecessor of j and j is called a successor of i . The set of all predecessors of j is indicated by $\pi(j)$ and the set of all successors of i is indicated by $\sigma(i)$. A predecessor is also called an *upstream node* and a successor is also called a *downstream node*.

Source, sink. If $\pi(i) = \emptyset$, then node i is called a source; if $\sigma(i) = \emptyset$ then i is called a sink. Depending on the application, a source, respectively sink, is also called an *input(-node)*, respectively an *output(-node)* of the graph.

Path, circuit, loop, length. A *path* ρ is a sequence of nodes $(i_1, i_2, \dots, i_p)$, $p > 1$, such that $i_j \in \pi(i_{j+1})$, $j = 1, \dots, p-1$. Node i_1 is the initial node and i_p is the final one of this path. Equivalently, one also says that a path is a sequence of arcs which connects a sequence of nodes. An *elementary path* is a path in which no node appears more than once. When the initial and the final nodes coincide, one speaks of a *circuit*. A circuit $(i_1, i_2, \dots, i_p = i_1)$ is an *elementary circuit* if the path $(i_1, i_2, \dots, i_{p-1})$ is elementary. A *loop* is a circuit (i, i) , that is, a circuit composed of a single node which is initial and final. This definition assumes that $i \in \pi(i)$, that is, there does exist an arc from i to i . The *length* of a path or a circuit is equal to the sum of the lengths of the arcs of which it is composed, the lengths of the arcs being 1 unless otherwise specified. With this convention, the length of a loop is 1. The length of path ρ is denoted $|\rho|_1$. The subscript '1' here refers to the word 'length' (later on, another subscript 'w' will appear for a different concept). The set of all paths and circuits in a graph is denoted R . A digraph is said to be *acyclic* if R contains no circuits.

Descendant, ascendant. The set of descendants $\sigma^+(i)$ of node i consists of all nodes j such that a path exists from i to j . Similarly the set of ascendants $\pi^+(i)$ of node i is the set of all nodes j such that a path exists from j to i . One has, e.g., $\pi^+(i) = \pi(i) \cup \pi(\pi(i)) \cup \dots$. The mapping $i \mapsto \pi^*(i) = \{i\} \cup \pi^+(i)$ is the transitive closure of π ; the mapping $i \mapsto \sigma^*(i) = \{i\} \cup \sigma^+(i)$ is the transitive closure of σ .

Subgraph. Given a graph $G = (\mathcal{V}, \mathcal{E})$, a graph $G' = (\mathcal{V}', \mathcal{E}')$ is said to be a subgraph of G if $\mathcal{V}' \subset \mathcal{V}$ and if $\mathcal{E}'$ consists of the set of arcs of G which have their origins and destinations in $\mathcal{V}'$.

Chain, connected graph. A graph is called connected if for all pairs of nodes i and j there exists a chain joining i and j . A chain is a sequence of nodes

$(i_1, i_2, \dots, i_p)$ such that between each pair of successive nodes either the arc (i_j, i_{j+1}) or the arc (i_{j+1}, i_j) exists. If one disregards the directions of the arcs in the definition of a path, one obtains a chain.

Strongly connected graph. A graph is called strongly connected if for any two different nodes i and j there exists a path from i to j . Equivalently, $i \in \sigma^*(j)$ for all $i, j \in \mathcal{V}$, with $i \neq j$. Note that, according to this definition, an isolated node, with or without a loop, is a strongly connected graph.

Bipartite graph. If the set of nodes $\mathcal{V}$ of a graph G can be partitioned into two disjoint subsets $\mathcal{V}_1$ and $\mathcal{V}_2$ such that every arc of G connects an element of $\mathcal{V}_1$ with one of $\mathcal{V}_2$ or the other way around, then G is called bipartite.

In §2.3, it will be useful to introduce the notion of an 'empty circuit' the length of which is equal to 0 by definition. An empty circuit contains no arcs. The circuit (i) is an empty circuit which should not be confused with the loop (i, i) of length 1 (the latter makes sense only if there exists an arc from node i to itself). Empty circuits are *not* included in the set R of paths.

To exemplify the various concepts introduced, consider the graph presented in Figure 2.1. It is a digraph since the arcs are indeed directed. The graph has seven nodes. Node 3 is a predecessor of node 6; $3 \in \pi(6)$. Similarly, $6 \in \sigma(3)$. The sequence of nodes 1, 3, 6, 4, 3, 2 is a nonelementary path. The arc $(1, 1)$ is a loop and the sequence of nodes 3, 6, 4, 3 is an elementary circuit of length 3. The sequence of nodes 2, 3, 6 is a chain. It should be clear that the graph of Figure 2.1 is connected.

Definition 2.1 (Equivalence relation $\mathcal{R}_\ell$) Let $i, j \in \mathcal{V}$ be two nodes of a graph. We say that $i \mathcal{R}_\ell j$, if either $i = j$ or there exist paths from i to j and from j to i .

Then $\mathcal{V}$ is split up into equivalence classes $\mathcal{V}_1, \dots, \mathcal{V}_q$, with respect to the relation $\mathcal{R}_\ell$. Note that if node i belongs to $\mathcal{V}_\ell$, then $\mathcal{V}_\ell = \sigma^*(i) \cap \pi^*(i)$. To each equivalence class $\mathcal{V}_\ell$ corresponds a subgraph $G_\ell = (\mathcal{V}_\ell, \mathcal{E}_\ell)$, where $\mathcal{E}_\ell$ is the restriction of $\mathcal{E}$ to $\mathcal{V}_\ell$, which is strongly connected.

Definition 2.2 (Maximal strongly connected subgraphs—m.s.c.s.) The subgraphs $G_i = (\mathcal{V}_i, \mathcal{E}_i)$ corresponding to the equivalence classes determined by $\mathcal{R}_\ell$ are the maximal strongly connected subgraphs of G .

Notation 2.3

- The subset of nodes of the m.s.c.s. containing node i (and possibly reduced to i) is denoted $[i]$.
- The subset of nodes $\bigcup_{j \in \pi^*(i)} [j]$ is denoted $[\leq i]$.
- The symbol $< i$ represents the subset of nodes $[\leq i] \setminus [i]$. ■

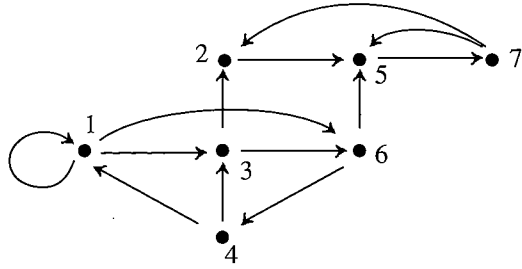


Figure 2.1: A digraph

The graph of Figure 2.1 has two m.s.c.s.'s, namely the subgraphs consisting of the nodes 1, 3, 6, 4 and 2, 5, 7, respectively. If one 'lumps' the nodes of each m.s.c.s. into a single node, one obtains the so-called *reduced graph*.

Definition 2.4 (Reduced graph) The reduced graph of G is the graph with nodes $\overline{\mathcal{V}} \stackrel{\text{def}}{=} \{1, \dots, q\}$ (one node per m.s.c.s.), and with arcs $\overline{\mathcal{E}}$, where $(i, j) \in \overline{\mathcal{E}}$ if $(k, l) \in \mathcal{E}$ for some node k of $\mathcal{V}_i$ and some node l of $\mathcal{V}_j$.

Figure 2.2 shows the reduced graph of the graph in Figure 2.1.

Notation 2.5 A node i of the reduced graph corresponds to a collection of nodes of the original graph. Let x be a vector the entries of which are associated with nodes of the original graph. If we want to refer to the subvector associated with the nodes of m.s.c.s. i , we will use the notation $x_{(i)}$. Similarly, for a matrix A , $A_{(i)(j)}$ is the block extracted from A by keeping the rows associated with the nodes of m.s.c.s. i and the columns associated with the nodes of m.s.c.s. j . If node l of the original graph belongs to m.s.c.s. i , the notation $x_{[l]}$ is equivalent to $x_{(i)}$. Similarly, $x_{(<i)}$, respectively $x_{(\leq i)}$, is equivalent to $x_{(<l)}$, respectively $x_{(\leq l)}$. ■

Lemma 2.6 The reduced graph is acyclic.

Proof If there is a path from $k \in \mathcal{V}_i$ to $l \in \mathcal{V}_j$, then there is no path from any node of $\mathcal{V}_j$ to any node of $\mathcal{V}_i$ (otherwise, k and l would be in the same m.s.c.s.). ■

Denote the existence of a path from one subgraph G_i to another one G_j by the binary relation $\mathcal{R}'$; $G_i \mathcal{R}' G_j$. Then these subgraphs $G_1, \dots, G_q$, together with the relation $\mathcal{R}'$ form a partially ordered set, see Chapter 4 and also [85].

2.3 GRAPHS AND MATRICES

In this section we consider matrices with entries belonging to an abstract alphabet C in which some algebraic operations will be defined in §2.3.1. Some relationships between these matrices and 'weighted graphs' will be introduced. Consider a graph

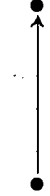


Figure 2.2: The reduced graph

$G = (\mathcal{V}, \mathcal{E})$ and associate an element $A_{ij} \in C$ with each arc $(j, i) \in \mathcal{E}$: then G is called a *weighted graph*. The quantity A_{ij} is called the *weight* of arc (j, i) . Note that the second subscript of A_{ij} refers to the initial (and not the final) node. The reason is that, in the algebraic context, we will work with column vectors (and not with row vectors) later on. In addition, we will also consider compositions of matrices and the resulting consequences for the corresponding graphs.

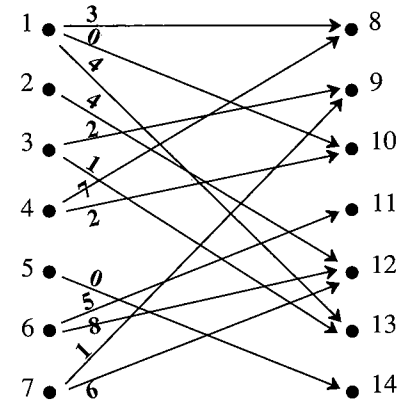
The alphabet C contains a special symbol ε the properties of which will be given in §2.3.1.

Definition 2.7 (Transition graph) If an $n \times m$ matrix $A = (A_{ij})$ with entries in C is given, the transition graph of A is a weighted, bipartite graph with $n + m$ nodes, labeled $1, \dots, m, m + 1, \dots, m + n$, such that each row of A corresponds to one of the nodes $1, \dots, m$; each column of A corresponds to one of the nodes $m + 1, \dots, m + n$. An arc from j to $m + i$, $1 \leq i \leq m$, $1 \leq j \leq n$, is introduced with weight A_{ij} if $A_{ij} \neq \varepsilon$.

As an example, consider the matrix

$$A = \begin{pmatrix} 3 & \varepsilon & \varepsilon & 7 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon & 1 \\ \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 5 & \varepsilon \\ \varepsilon & 4 & \varepsilon & \varepsilon & \varepsilon & 8 & 6 \\ 4 & \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}. \quad (2.1)$$

Its transition graph is depicted in Figure 2.3.

Figure 2.3: The transition graph of A

Definition 2.8 (Precedence graph) The precedence graph of a square $n \times n$ matrix A with entries in C is a weighted digraph with n nodes and an arc (j, i) if $A_{ij} \neq \varepsilon$.

in which case the weight of this arc receives the numerical value of A_{ij} . The precedence graph is denoted $G(A)$.

It is not difficult to see that any weighted digraph $G = (\mathcal{V}, \mathcal{E})$ is the precedence graph of an appropriately defined square matrix. The weight A_{ij} of the arc from node j to node i defines the ij -th entry of a matrix A . If an arc does not exist, the corresponding entry of A is set to ε . The matrix A thus defined has G as its precedence graph.

The transition graph of a square $n \times n$ matrix, which has $2n$ nodes, can be transformed into a precedence graph of n nodes. Towards that end, one combines the nodes i and $n+i$ of the transition graph into one single node for the precedence graph, $i = 1, \dots, n$. As an example, Figure 2.4 gives the precedence graph of the

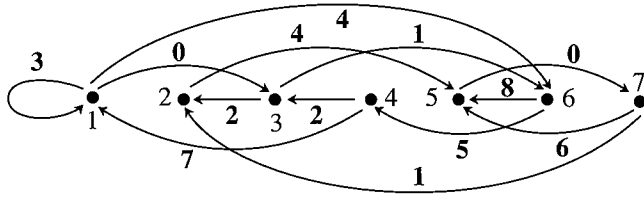


Figure 2.4: The precedence graph of A

matrix A defined in (2.1). One directly recognizes the relation of this graph with the transition graph in Figure 2.3. The latter graph has been 'folded' so as to obtain the precedence graph.

It may be convenient to consider that entries A_{ij} equal to ε define *dummy arcs* (which are not drawn) in the associated precedence or transition graph. Later, a path including a dummy arc will be called a *dummy path*. Dummy paths are not included in the set R of paths (which were taken into account in Definition 2.1; therefore these dummy paths are not involved in the definition of m.s.c.s.'s). The interest of the notion of dummy arcs is that these arcs may be considered as being of the same length as arcs associated with entries $A_{ij} \neq \varepsilon$ (generally this length is 1); hence arcs of the same length can be associated with all entries of a matrix.

Notation 2.9 The number of m.s.c.s.'s of $G(A)$ is denoted N_A . ■

For later reference, the following definitions are given.

Definition 2.10 (Incidence matrix) The incidence matrix $F = (F_{ij})$ of a graph $G = (\mathcal{V}, \mathcal{E})$ is a matrix the number of columns of which equals the number of arcs and the number of rows of which equals the number of nodes of the graph. The entries of F can take the values 0, 1 or -1 . If $l = (i, j) \in \mathcal{E}$, $i \neq j$, then $F_{il} = 1, F_{jl} = -1$ and the other entries of column l are 0. If $l = (i, i) \in \mathcal{E}$, then $F_{il} = 0$.

Definition 2.11 (Adjacency matrix) The adjacency matrix $G = (G_{ij})$ of a graph $G = (\mathcal{V}, \mathcal{E})$ is a matrix the numbers of rows and columns of which are equal to

the number of nodes of the graph. The entry G_{ij} is equal to 1 if $j \in \pi(i)$ and to 0 otherwise.

Note that if $G = G(A)$, then $G_{ij} = 1$ if and only if $A_{ij} \neq \varepsilon$ (G describes the 'support' of A).

2.3.1 Composition of Matrices and Graphs

We now study two kinds of compositions of matrices, and the relation between the transition graph of these compositions and the original transition graphs. These compositions are, respectively, the *parallel composition*, denoted $\oplus$, and the *series composition*, denoted $\otimes$. These compositions will be defined by means of the corresponding composition operations of elements in the alphabet C , for which the same symbols $\oplus$ and $\otimes$ will be used. The operation $\oplus$ is usually referred to as 'addition' or 'sum', and the operation $\otimes$ as 'multiplication' or 'product'. The alphabet C includes two special elements ε and e with specific properties to be defined in the following set of axioms:

Associativity of addition:

$$\forall a, b, c \in C, \quad (a \oplus b) \oplus c = a \oplus (b \oplus c).$$

Commutativity of addition:

$$\forall a, b \in C, \quad a \oplus b = b \oplus a.$$

Associativity of multiplication:

$$\forall a, b, c \in C, \quad (a \otimes b) \otimes c = a \otimes (b \otimes c).$$

Right and left distributivity of multiplication over addition:

$$\forall a, b, c \in C, \quad (a \oplus b) \otimes c = (a \otimes c) \oplus (b \otimes c),$$

$$\forall a, b, c \in C, \quad c \otimes (a \oplus b) = (c \otimes a) \oplus (c \otimes b).$$

Existence of a zero element:

$$\exists \varepsilon \in C : \quad \forall a \in C, \quad a \oplus \varepsilon = a.$$

Absorbing zero element:

$$\forall a \in C, \quad a \otimes \varepsilon = \varepsilon.$$

Existence of an identity element:

$$\exists e \in C : \quad \forall a \in C, \quad a \otimes e = e \otimes a = a.$$

In Chapter 3 other related axioms will be discussed in detail. There the notion of a *semifield* will be introduced and its relation to axioms of this type will be made clear.

The *parallel composition* $\oplus$ of matrices is defined for matrices of the same size by the following rule: if $A = (A_{ij})$ and $B = (B_{ij})$ have the same size, then

$$(A \oplus B)_{ij} = A_{ij} \oplus B_{ij} .$$

A transition graph can of course be associated with the matrix $C = A \oplus B$. This transition graph has the same set of nodes as the transition graph of A (and therefore of B) and there exists a (nondummy) arc from node j to node i if and only if at least one of the transition graphs of A and B has a (nondummy) arc from j to i . This is a consequence of the axiom $A_{ij} \oplus \varepsilon = A_{ij}$. In general, this arc receives the weight $A_{ij} \oplus B_{ij}$. It may be viewed as the arc resulting from the merging of two parallel arcs (if both A_{ij} and B_{ij} are different from ε). The symbol ε is called the *zero element* of the operation $\oplus$. Two other axioms are that $\oplus$ is associative and commutative and they have obvious consequences for the parallel composition of transition graphs.

The *series composition* $\otimes$ of matrices A and B is defined only when the number of columns of A equals the number of rows of B (say, A is $m \times n$ and B is $n \times p$) by the following rule:

$$(A \otimes B)_{ij} = \bigoplus_{k=1}^n A_{ik} \otimes B_{kj} . \quad (2.2)$$

A transition graph can be associated with the matrix $C = A \otimes B$ of size $m \times p$. With the help of Figure 2.5, we explain how this graph is obtained. First, the

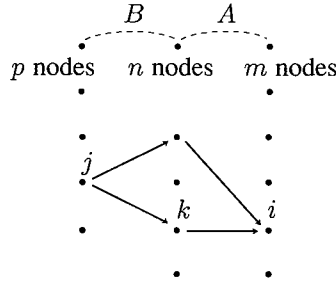


Figure 2.5: The series composition of two transition matrices

transition graphs of matrices B and A are concatenated. The graph so obtained is not a transition graph since it has n intermediate nodes in addition of its p input nodes and its m output nodes. These intermediate nodes are removed, and an arc from node j to node i exists in the transition graph of C if and only if there exists at least one (nondummy) path from node j to node i in the concatenated graph shown in Figure 2.5. This arc receives the weight indicated by Equation (2.2). In order to interpret this formula, let us first define the weight of a path $\rho = (i_1, \dots, i_p)$ in a

weighted graph as the product $A_{i_p, i_{p-1}} \otimes \dots \otimes A_{i_2, i_1}$ of weights of arcs composing this path (observe the order). This weight is denoted $|\rho|_w$, where the subscript w refers to the word 'weight'. Note that a dummy path always has a weight ε thanks to the absorbing property of ε in products. Then, each term of the sum in (2.2) can be interpreted as the weight of some parallel path of length 2 from node j to node i , characterized by the intermediate node k it passes through (see Figure 2.5). The previous rule pertaining to the weights of parallel compositions of arcs is thus extended to parallel paths. Since ε is the zero element for addition, dummy paths do not contribute to (nonzero) weights in parallel compositions.

For the series composition of matrices, we have $(A \otimes B) \otimes C = A \otimes (B \otimes C)$. This associativity property of matrix multiplication is a direct consequence of the axioms given above, namely the associativity of $\otimes$ and the right and left distributivity of $\otimes$ over $\oplus$. It is easily seen that these right and left distributivities also hold for matrices of appropriate sizes.

The notion $(A^p)_{ji}$, $p = 1, 2, \dots$, where A is square, is defined by $(A^p)_{ji} = (A \otimes A^{p-1})_{ji}$. Observe that this involves the enumeration and the sums of weights of all paths of length p with initial node i and final node j . This definition makes sense for both the transition graph of A , concatenated p times, and the precedence graph of A . We get

$$(A^p)_{ji} = \bigoplus_{\{\rho \mid |\rho|=p; i_0=i; i_p=j\}} A_{i_p, i_{p-1}} \otimes A_{i_{p-1}, i_{p-2}} \otimes \dots \otimes A_{i_1, i_0} .$$

Because we removed the intermediate nodes in the graph of Figure 2.5 in order to obtain the transition graph of $C = A \otimes B$, and similarly when one considers the transition graph of a matrix C equal to A^p , the information that weights of the graph of C have been obtained as weights of paths of length larger than 1 (namely 2 and p , respectively) has been lost. In order to keep track of this information, one may introduce the notion of *length of a transition graph*. This length is an integer number associated with the transition graph and hence also with all its individual arcs, including dummy arcs, and finally with the matrix associated with this graph. These considerations explain why the lengths of arcs may be taken greater than 1 and why dummy arcs may also have a nonzero length.

We now consider the transition graph corresponding to matrices A^0 where A is an $n \times n$ matrix. In the same way as A^p , for $p \geq 1$, describes the weights of paths of length p , A^0 should describe the weights of paths of length 0, that is, empty circuits (i) corresponding to 'no transition at all'. Pictorially, the corresponding transition graph of such matrices has the special form depicted in Figure 2.6a in which input and output nodes are not distinguishable. This transition graph must not be confused either with that of the 'zero matrix' (with all entries equal to ε) or with that of the 'identity matrix' (with diagonal entries equal to e and all off-diagonal entries equal to ε). The transition graph of the zero matrix is depicted in Figure 2.6b: all its arcs are dummy with length 1. The transition graph of the identity matrix is depicted in Figure 2.6c: weights are indicated in the figure and the length is 1 for all dummy and nondummy arcs. The length associated with all entries of A^0 is 0, that is, the series composition of the transition graph of A^0 with any other bipartite graph do not modify the lengths of paths. In the same way, we would like the weights not

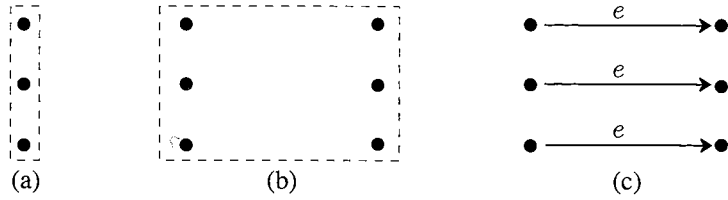


Figure 2.6: The transition graphs of A^0 , ε and e

to be modified in that operation: this requires that the entries of A^0 be the same as those of the identity matrix e . But again the respective lengths and transition graphs of A^0 and e are different (see Figure 2.6a–c). Also, dummy loops (i, i) of the transition graph of the zero matrix should not be confused with empty circuits (i) (see Figure 2.6a–b) even if it is difficult to distinguish them with the help of precedence, rather than transition, graphs.

In Chapter 1 we already met several examples of $\oplus$ and $\otimes$ operations which satisfy the axioms just stated. A particular example is that $\mathbb{C}$ equals $\mathbb{R}$, $e = 0$, $\varepsilon = -\infty$, $\oplus$ equals maximization and $\otimes$ equals addition. Note that the max-operation is idempotent, i.e. $a \oplus a = a$, but this property is not (yet) assumed as an axiom.

Remark 2.12 An example of operations which do *not* satisfy some of the axioms is one where $\oplus$ is addition and $\otimes$ is minimization. The axioms of distributivity are not satisfied. Indeed,

$$\min(5, 3 + 6) \neq \min(5, 3) + \min(5, 6) .$$

As a consequence, associativity of multiplication with respect to matrices does not hold. If for instance,

$$A = \begin{pmatrix} 1 & 3 \\ 2 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} 3 & 2 \\ 1 & 5 \end{pmatrix}, \quad C = \begin{pmatrix} 4 & 2 \\ 2 & 5 \end{pmatrix},$$

then

$$\begin{pmatrix} 4 & 6 \\ 5 & 6 \end{pmatrix} = (AB)C \neq A(BC) = \begin{pmatrix} 4 & 4 \\ 4 & 4 \end{pmatrix} .$$

A practical interpretation of a system with these operations is in terms of the calculation of the capacity of a network in which the arcs are pipes through which there is a continuous flow. The capacity of a pipe is assumed to be proportional to the diameter of this pipe. Then it is easily seen that the capacity of two pipes in parallel equals the sum of the two capacities. Similarly, the capacity of two pipes in series equals the minimum of their capacities. The reader should contemplate the physical consequences of the lack of associativity. ■

Definition 2.13 (Irreducibility) The (square) matrix A is called irreducible if no permutation matrix P exists such that the matrix $\tilde{A}$, defined by

$$\tilde{A} = P'AP ,$$

has an upper triangular block structure.

The reader should be aware of the fact that this definition is invariant with respect to the algebra used. Premultiplication of A by P' and postmultiplication by P simply refers to a renumbering of the nodes of the corresponding graph. Hence renumbering of the nodes of the same graph leads to different A -matrices. In an upper triangular block structure, diagonal blocks with non- ε entries are allowed. If one also wants the diagonal blocks to have ε -entries only, one should speak about a strictly upper triangular block structure.

Theorem 2.14 A necessary and sufficient condition for the square matrix A to be irreducible is that its precedence graph be strongly connected.

Proof Suppose that A is such that by an appropriate renumbering of the nodes, $\tilde{A}$ has an upper triangular block structure. Call the diagonal blocks $A_1^d, \dots, A_q^d$. If A_q^d has size $n_q \times n_q$, then there are no paths from any of the (renumbered) nodes $1, \dots, n - n_q$, to any of the nodes $n - n_q + 1, \dots, n$. Hence this graph is not strongly connected.

On the other hand, if the graph is not strongly connected, determine its m.s.c.s.'s $G_i, i = 1, \dots, q$. These subgraphs form a partially ordered set. Number the individual nodes of $\mathcal{V}$ in such a way that if $G_i \mathcal{R}' G_j$, then the nodes of $\mathcal{V}_i$ have lower indices than those of $\mathcal{V}_j$ ($\mathcal{R}'$ was defined in §2.2). With this numbering of the nodes, the corresponding matrix A will be upper block triangular. ■

Definition 2.15 (Aperiodicity) The irreducible square matrix A is aperiodic if there exists an integer N such that for all $n \geq N$ and for all i, j , $(A^n)_{ij} \neq \varepsilon$.

Theorem 2.16 An irreducible matrix A such that $A_{jj} \neq \varepsilon$ for all j is aperiodic.

Proof From Theorem 2.14, the irreducibility assumption implies that for all i, j , there exists n such that $(A^n)_{ij} \neq \varepsilon$. This together with the assumption $A_{jj} \neq \varepsilon$ in turn implies that $(A^m)_{ij} \neq \varepsilon$ for all $m \geq n$. The assertion of the theorem follows immediately from this since the number of nodes is finite. ■

Definition 2.17 A digraph is called a tree if there exists a single node such that there is a unique path from this node to any other node.

In order to determine whether A is irreducible, one can calculate A^+ (see (1.18)). Matrix A is irreducible if and only if all entries of A^+ are different from ε . This algorithm for determining whether A is irreducible can be simplified by considering only Boolean variables. Replace A by the adjacency matrix G of its precedence graph (see Definition 2.11), except that 0 and 1 are replaced by ε and e , respectively,

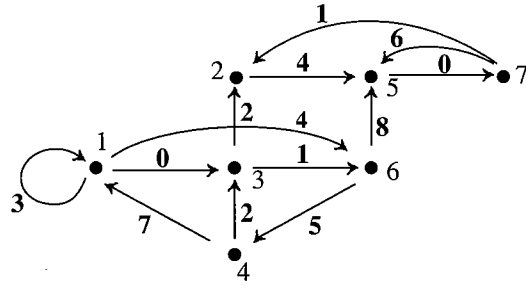


Figure 2.7: Weighted digraph consisting of two m.s.c.s.'s

in the present context. Then A is irreducible if and only if all entries of G^+ are identical to e . As an example, consider the matrix given in (2.1). The matrix G^+ becomes

$$\begin{pmatrix} e & \varepsilon & e & e & \varepsilon & e & \varepsilon \\ e & e & e & e & e & e & e \\ e & \varepsilon & e & e & \varepsilon & e & \varepsilon \\ e & \varepsilon & e & e & \varepsilon & e & \varepsilon \\ e & e & e & e & e & e & e \\ e & \varepsilon & e & e & \varepsilon & e & \varepsilon \\ e & e & e & e & e & e & e \end{pmatrix}. \quad (2.3)$$

Hence A is not irreducible. In fact it follows directly from (2.3) that the nodes 1, 3, 4 and 6 form a m.s.c.s., as do the nodes 2, 5 and 7. If one rearranges the nodes and arcs of Figure 2.4, one obtains Figure 2.7, in which the weights of the arcs have been indicated, and the two m.s.c.s.'s are clearly visible. Figure 2.7 is identical to Figure 2.1 apart from the fact that weights are given.

2.3.2 Maximum Cycle Mean

In this subsection the maximum cycle mean will be defined and some of its properties will be derived. The maximum cycle mean has a clear relation with eigenvalues of matrices within the context of the max algebra and with periodic regimes of systems described by linear equations, also within the same context. These relationships will be described in §3.

Let $G = (\mathcal{V}, \mathcal{E})$ be a weighted digraph with n nodes. The weights are real numbers here and are given by means of the $n \times n$ matrix A . As discussed before, the numerical value of A_{ij} equals the weight of the arc from node j to node i . If no such arc exists, then $A_{ij} = \varepsilon$. It is known from Chapter 1 that the entry (i, j) of $A^k = A \otimes \cdots \otimes A$, considered within the algebraic structure $\mathbb{R}_{\max} \stackrel{\text{def}}{=} (\mathbb{R} \cup \{-\infty\}, \max, +)$, denotes the maximum weight with respect to all paths of length k which go from node j to node i . If no such path exists, then $(A^k)_{ij} = \varepsilon$. Within this algebraic structure, ε gets assigned the numerical value $-\infty$ and $e = 0$. In this subsection we will confine ourselves to the algebraic structure $\mathbb{R}_{\max}$.

Definition 2.18 (Cycle mean) The mean weight of a path is defined as the sum of the weights of the individual arcs of this path, divided by the length of this path. If the path is denoted ρ , then the mean weight equals $|\rho|_w / |\rho|_l$. If such a path is a circuit one talks about the mean weight of the circuit, or simply the cycle mean.

We are interested in the maximum of these cycle means, where the maximum is taken over all circuits in the graph (empty circuits are not considered here). Consider an $n \times n$ matrix A with corresponding precedence graph $G = (\mathcal{V}, \mathcal{E})$. The maximum weight of all circuits of length j which pass through node i of G can be written as $(A^j)_{ii}$. The maximum of these maximum weights over all nodes is $\bigoplus_{i=1}^n (A^j)_{ii}$ which can be written $\text{trace}(A^j)$. The average weight is obtained by dividing this number by j in the conventional sense, but this can be written $(\text{trace}(A^j))^{1/j}$ in the max-plus algebra notation. Finally, we have to take the maximum with respect to the length j . It is not necessary to consider lengths larger than the number n of nodes since it is enough to limit ourselves to elementary circuits. It follows that a formula for the maximum cycle mean λ in the max-plus algebra notation is

$$\lambda = \bigoplus_{j=1}^n (\text{trace}(A^j))^{1/j}.$$

The following theorem provides an expression, due to R. Karp, for this maximum value. All circuits with a cycle mean equal to the maximum cycle mean are called critical circuits. Karp's theorem does not give the critical circuit(s).

Theorem 2.19 (Karp's theorem) Given is an $n \times n$ matrix A with corresponding precedence graph $G = (\mathcal{V}, \mathcal{E})$. The maximum cycle mean is given by

$$\lambda = \max_{i=1, \dots, n} \min_{k=0, \dots, n-1} \frac{(A^n)_{ij} - (A^k)_{ij}}{n - k}, \quad \forall j. \quad (2.4)$$

In this equation, A^n and A^k are to be evaluated in $\mathbb{R}_{\max}$; the other operations are conventional ones.

Proof Note that the index j in (2.4) is arbitrary (it will be shown in this proof that one can take any $j \in \{1, \dots, n\}$). The resulting value of λ is independent of j .

Without loss of generality, we may assume that G is strongly connected. If it were not, we would consider each of its m.s.c.s.'s—since G is assumed to be finite, there are only a finite number of such m.s.c.s.'s—and determine the maximum cycle mean of each of them and then take the maximum one.

We first assume that the maximum cycle mean is 0. Then it must be shown that

$$\max_{i=1, \dots, n} \min_{k=0, \dots, n-1} \frac{(A^n)_{ij} - (A^k)_{ij}}{n - k} = 0.$$

Since $\lambda = 0$ there exists a circuit of weight 0 and there exists no circuit with positive weight. Because there are no circuits (or loops) with positive weight, there

is a maximum weight of all paths from node j to node i which is equal to

$$\chi_{ij} \stackrel{\text{def}}{=} \max \sum_{l=1}^k A_{i_l, i_{l-1}} , \quad \text{subject to } i_0 = j , \quad i_k = i ,$$

where the maximum is taken with respect to all paths and all k . Since for $k \geq n$ the path would contain a circuit and since all circuits have nonpositive weight, we can restrict ourselves to $k < n$. Therefore we get

$$\chi_{ij} = \max_{k=0, \dots, n-1} (A^k)_{ij} .$$

Also, $(A^n)_{ij} \leq \chi_{ij}$, and hence

$$(A^n)_{ij} - \chi_{ij} = \min_{k=0, \dots, n-1} (A^n)_{ij} - (A^k)_{ij} \leq 0 .$$

Equivalently,

$$\min_{k=0, \dots, n-1} \frac{(A^n)_{ij} - (A^k)_{ij}}{n - k} \leq 0 . \quad (2.5)$$

Equality in (2.5) will only hold if $(A^n)_{ij} = \chi_{ij}$. It will be shown that indeed an index i exists such that this is true. Let ζ be a circuit of weight 0 and let l be a node of ζ . Let ρ_{lj} be a path from j to l with corresponding maximum weight $|\rho_{lj}|_w = \chi_{lj}$. Now this path is extended by appending to it a number of repetitions of ζ such that the total length of this extended path, denoted ρ_e , becomes greater than or equal to n . This is again a path of maximum weight from j to l . Now consider the path consisting of the first n arcs of ρ_e ; its initial node is j and denote its final node l' . Of course $l' \in \zeta$. Since any subpath of any path of maximum weight is of maximum weight itself, the path from j to l' is of maximum weight. Therefore $(A^n)_{lj} = \chi_{lj}$. Now choose $i = l'$ and we get

$$\max_{i=1, \dots, n} \left[\min_{k=0, \dots, n-1} \frac{(A^n)_{ij} - (A^k)_{ij}}{n - k} \right] = 0 .$$

This completes the part of the proof with $\lambda = 0$.

Now consider an arbitrary finite λ . A constant c is now subtracted from each weight A_{ij} . Then clearly λ will be reduced by c . Since $(A^k)_{ij}$ is reduced by kc , we get that

$$\frac{(A^n)_{ij} - (A^k)_{ij}}{n - k}$$

is reduced by c , for all i, j and k , and hence

$$\max_{i=1, \dots, n} \min_{k=0, \dots, n-1} \frac{(A^n)_{ij} - (A^k)_{ij}}{n - k}$$

is also reduced by c . Hence both sides of (2.4) are affected equally when all weights A_{ij} are reduced by the same amount. Now choose this amount such that λ becomes 0 and then we are back in the previous situation where $\lambda = 0$. ■

2.3.3 The Cayley-Hamilton Theorem

The Cayley-Hamilton theorem states that, in conventional algebra, a square matrix satisfies its own characteristic equation. In mathematical terms, let A be an $n \times n$ matrix and let

$$p_A(x) \stackrel{\text{def}}{=} \det(xI - A) = x^n + c_1 x^{n-1} + \dots + c_{n-1} x + c_n x^0 , \quad (2.6)$$

where I is the identity matrix, be its characteristic polynomial. The term x^0 in the polynomial equals 1. Then $p_A(A) = 0$, where 0 is the zero matrix. The coefficients $c_i, i = 1, \dots, n$, in (2.6) satisfy

$$c_k = (-1)^k \sum_{i_1 < i_2 < \dots < i_k} \det \begin{pmatrix} A_{i_1, i_1} & \dots & A_{i_1, i_k} \\ \vdots & & \vdots \\ A_{i_k, i_1} & \dots & A_{i_k, i_k} \end{pmatrix} . \quad (2.7)$$

The reason for studying the Cayley-Hamilton theorem is the following. In conventional system theory, this theorem is used for the manipulation of different system descriptions and for analyzing such properties as controllability (see [72]). In the context of discrete event systems, a utilization of this theorem will be shown in §9.2.2.

In this section it will be shown that a Cayley-Hamilton theorem also exists in an algebraic structure defined by a set C of elements supplied with two operations denoted $\oplus$ and $\otimes$ which obey some of the axioms given in §2.3.1, namely

- associativity of addition,
- commutativity of addition,
- associativity of multiplication,
- both right and left distributivity of multiplication over addition,
- existence of an identity element,

provided we also have

Commutativity of multiplication:

$$\forall a, b \in C , \quad a \otimes b = b \otimes a .$$

Note that the existence of a zero element (and its absorbing property) is not required in this subsection.

A *partial permutation* of $\{1, \dots, n\}$ is a bijection ς of a subset of $\{1, \dots, n\}$ onto itself. The domain of ς is denoted by $\text{dom}(\varsigma)$ and its cardinality is denoted $|\varsigma|$. A partial permutation ς for which $|\varsigma| = n$ is called a complete permutation. The completion $\widehat{\varsigma}$ of a partial permutation ς is defined by

$$\widehat{\varsigma}(i) = \begin{cases} \varsigma(i) & \text{if } i \in \text{dom}(\varsigma) , \\ i & \text{if } i \in \{1, \dots, n\} \setminus \text{dom}(\varsigma) . \end{cases}$$

The *signature* $*$ of a partial permutation ς , denoted $\text{sgn}^*(\varsigma)$, is defined by

$$\text{sgn}^*(\varsigma) = \text{sgn}(\hat{\varsigma})(-1)^{|\varsigma|},$$

where $\text{sgn}(\hat{\varsigma})$, sometimes also written as $(-1)^{\hat{\varsigma}}$, denotes the conventional signature of the permutation $\hat{\varsigma}$, see [61].

Every (partial) permutation has a unique representation as a set of disjoint circuits. For example, the permutation

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 4 & 6 & 3 & 5 & 1 & 2 \end{pmatrix}$$

has the circuit representation

$$\{(1, 4, 5), (3), (2, 6)\}.$$

With the graph-theoretic interpretation of permutations in mind, these disjoint circuits correspond to m.s.c.s.'s. The unique partial permutation of cardinality 0 has the empty set as its circuit representation. If ς is a partial permutation with cardinality k consisting of a single circuit, then $\text{sgn}^* = (-1)^{k-1}(-1)^k = -1$. It easily follows that for any partial permutation ς , $\text{sgn}^* = (-1)^r$, where r is the number of circuits appearing in the circuit representation of ς .

Given an $n \times n$ matrix $A = (A_{ij})$, the weight of ς is defined by

$$|\varsigma|_w = \bigotimes_{i \in \text{dom}(\varsigma)} A_{\varsigma(i), i}.$$

The weight of the partial permutation with cardinality 0 equals e , in accordance with the theory presented in §2.3.1. Let $1 \leq i, j \leq n$ and let T_{ji}^+ be the set of all pairs (ς, ρ) where ς is a partial permutation and where ρ is a path from i (the initial node) to j (the final node) in such a way that

$$|\varsigma|_1 + |\rho|_1 = n, \quad \text{sgn}^*(\varsigma) = 1.$$

The set T_{ji}^- is defined identically except for the fact that the condition $\text{sgn}^*(\varsigma) = 1$ is replaced by $\text{sgn}^*(\varsigma) = -1$.

Lemma 2.20 *For each pair (j, i) , with $1 \leq i, j \leq n$, there is a bijection $\eta_{ji} : T_{ji}^+ \rightarrow T_{ji}^-$ in such a way that $\eta_{ji}(\varsigma, \rho) = (\varsigma', \rho')$ implies $|\varsigma|_w \otimes |\rho|_w = |\varsigma'|_w \otimes |\rho'|_w$.*

Proof Each pair $(\varsigma, \rho) \in T_{ji}^+ \cup T_{ji}^-$ is represented by a directed graph with nodes $\{1, \dots, n\}$. The set of arcs consists of two classes of arcs:

$$\mathcal{E}_\varsigma = \{(i, \varsigma(i)) \mid i \in \text{dom}(\varsigma)\}, \quad \mathcal{E}_\rho = \{(l, k) \mid (l, k) \text{ is an arc of } \rho\}.$$

This graph will, in general, contain multiple arcs, since ρ may traverse the same arc more than once, or the same arc may appear in both $\mathcal{E}_\varsigma$ and $\mathcal{E}_\rho$. The expression

$|\varsigma|_w \otimes |\rho|_w$ is the series composition, with multiplicities taken into account, of all the A_{lk} for which (k, l) is an arc of the graph associated with (ς, ρ) .

Let ρ be the path connecting the nodes $i_0, i_1, \dots, i_q$, in this order. There is a smallest integer $v \geq 0$ such that either $i_u = i_v$ for some $u < v$, or $i_v \in \text{dom}(\varsigma)$. If such a v did not exist, then ρ must have $|\rho|_1 + 1$ distinct nodes (because there are no u and v with $i_v = i_u$). But then there are at least $|\varsigma|_1 + |\rho|_1 + 1 = n + 1$ distinct nodes (because ς and ρ do not have any node in common), which is a contradiction. Furthermore, it is easily seen that this smallest integer v cannot have both properties. Hence either $i_u = i_v$ for some $u < v$ or $i_v \in \text{dom}(\varsigma)$. An example of the first property is given in Figure 2.8 and an example of the second property is given in Figure 2.9. The dashed arcs refer to the set $\mathcal{E}_\varsigma$ and the solid arcs refer to the set $\mathcal{E}_\rho$. In Figure 2.8, v equals 3, and in Figure 2.9, v equals 1 (i.e. $i_1 = 2$).

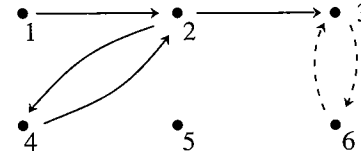


Figure 2.8: Example of a graph with the first property

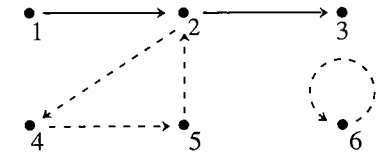


Figure 2.9: Example of a graph with the second property

Consider the situation with the first property such as pictured in Figure 2.8. We have $i_u = i_v$ for some u and v . The circuit passing through i_u is removed from ρ and adjoined as a new circuit to ς . The new path from i_0 to i_q is denoted ρ' and the new, longer, partial permutation is denoted ς' . The mapping η_{ji} is defined as $\eta_{ji}(\varsigma, \rho) = (\varsigma', \rho')$. Application of the mapping η_{31} to Figure 2.8 is given in Figure 2.10. Since the number of circuits of ς' and of ς differ by one, we have

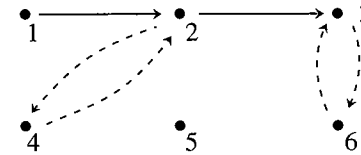


Figure 2.10: η_{31} applied to Figure 2.8

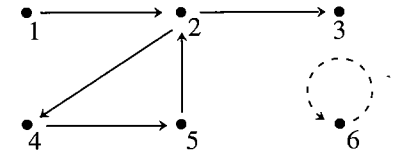


Figure 2.11: η_{31} applied to Figure 2.9

$\text{sgn}^*(\varsigma') = -\text{sgn}^*(\varsigma)$ and therefore η_{ji} maps T_{ji}^+ into T_{ji}^- and vice versa.

Consider next the situation with the second property such as pictured in Figure 2.9. The mapping $\eta_{ji}(\varsigma, \rho) = (\varsigma', \rho')$ is obtained by removing the circuit containing i_v from ς and adjoining it to ρ . Application of η_{31} to Figure 2.9 is given in Figure 2.11. Also in this situation, the numbers of circuits of ς' and of ς differ by one, and again we have $\text{sgn}^*(\varsigma') = -\text{sgn}^*(\varsigma)$ which results in the fact that η_{ji} maps T_{ji}^+ into T_{ji}^- and vice versa. In both situations $|\varsigma|_w \otimes |\rho|_w = |\varsigma'|_w \otimes |\rho'|_w$, since nothing has changed in the graph of (ς, ρ) . It is in the derivation of this equality

that the associativity and commutativity of multiplication, and the existence of an identity element, have been used. What remains to be shown is that the mapping η_{ji} is surjective. For this reason consider the iteration $\eta_{ji} \circ \eta_{ji}$. It easily follows that this mapping is the identity on $T_{ji}^+ \cup T_{ji}^-$, which is only possible if η_{ji} is surjective. ■

Definition 2.21 (Characteristic equation) *The characteristic equation is given by*

$$p_A^+(x) = p_A^-(x) , \quad (2.8)$$

where

$$p_A^+(x) = \bigoplus_{k=0}^n \left(\bigoplus_{|\zeta|=k, \text{sgn}^*(\zeta)=1} \left(\bigotimes_{i \in \text{dom}(\zeta)} A_{\zeta(i),i} \right) \right) x^{n-k} ,$$

and

$$p_A^-(x) = \bigoplus_{k=0}^n \left(\bigoplus_{|\zeta|=k, \text{sgn}^*(\zeta)=-1} \left(\bigotimes_{i \in \text{dom}(\zeta)} A_{\zeta(i),i} \right) \right) x^{n-k} .$$

It is easily verified that this characteristic equation, if considered the conventional algebra, coincides with the equation obtained by setting the characteristic polynomial (2.6) equal to zero. The crucial feature of (2.8) is that there are no terms with ‘negative’ coefficients (in contrast with the conventional characteristic equation, which can have negative coefficients). Since the inverse of $\oplus$ does not exist, the terms in (2.8) cannot freely be moved from one side to the other side of the equation.

Theorem 2.22 (The Cayley-Hamilton theorem) *The following identity holds true:*

$$p_A^+(A) = p_A^-(A) .$$

Proof For $k = 0, \dots, n$,

$$(A^{n-k})_{ji} = \bigoplus_{\substack{|\rho|=n-k \\ \text{initial node of } \rho \text{ equals } i \\ \text{final node of } \rho \text{ equals } j}} |\rho|_w .$$

It follows that

$$p_A^+(A)_{ji} = \bigoplus_{(\zeta, \rho) \in T_{ji}^+} |\zeta|_w \otimes |\rho|_w , \quad p_A^-(A)_{ji} = \bigoplus_{(\zeta, \rho) \in T_{ji}^-} |\zeta|_w \otimes |\rho|_w .$$

Owing to Lemma 2.20, these two sums are identical. It is in these two equalities that the associativity and commutativity of addition, the distributivity and the existence of an identity element have been used. ■

Let us give an example. For the 3×3 matrix $A = (A_{ij})$, the characteristic equation is $p_A^+(x) = p_A^-(x)$, where

$$\begin{aligned} p_A^+(x) &= x^3 \oplus (A_{11}A_{22} \oplus A_{11}A_{33} \oplus A_{22}A_{33})x \oplus A_{13}A_{22}A_{31} \\ &\quad \oplus A_{12}A_{21}A_{33} \oplus A_{11}A_{32}A_{23} , \\ p_A^-(x) &= (A_{11} \oplus A_{22} \oplus A_{33})x^2 \oplus (A_{12}A_{21} \oplus A_{13}A_{31} \oplus A_{23}A_{32})x \\ &\quad \oplus A_{11}A_{22}A_{33} \oplus A_{12}A_{23}A_{31} \oplus A_{21}A_{13}A_{32} , \end{aligned}$$

where, as usual, the $\otimes$ -symbols have been omitted. If we consider

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 1 & \varepsilon \\ e & 5 & 3 \end{pmatrix}$$

in the algebraic structure $\mathbb{R}_{\max}$, then the characteristic equation becomes

$$x^3 \oplus 4x \oplus 9 = 3x^2 \oplus 6x \oplus 12 ,$$

which can be simplified to

$$x^3 = 3x^2 \oplus 6x \oplus 12 ,$$

since the omitted terms are dominated by the corresponding terms at the other side of the equality. A simple calculation shows that if one substitutes A in the latter equation, one obtains an identity indeed:

$$\begin{pmatrix} 12 & 11 & 9 \\ 10 & 12 & 10 \\ 12 & 11 & 12 \end{pmatrix} = \begin{pmatrix} 9 & 11 & 9 \\ 8 & 9 & 10 \\ 12 & 11 & 9 \end{pmatrix} \oplus \begin{pmatrix} 7 & 8 & 9 \\ 10 & 7 & \varepsilon \\ 6 & 11 & 9 \end{pmatrix} \oplus \begin{pmatrix} 12 & \varepsilon & \varepsilon \\ \varepsilon & 12 & \varepsilon \\ \varepsilon & \varepsilon & 12 \end{pmatrix} .$$

This section will be concluded with some remarks on minimal polynomial equations. The Cayley-Hamilton theorem shows that there exists at least one polynomial equation satisfied by a given $n \times n$ matrix A . This polynomial equation is of degree n . For example, if A is the 3×3 identity matrix, then

$$p_A^+(x) = x^3 \oplus x , \quad p_A^-(x) = x^2 \oplus e ,$$

and A satisfies the equation $A^3 \oplus A = A^2 \oplus e$. There may exist equations of lower degree also satisfied by A . With the previous A , we also have $A = e$ and $x = e$ is a polynomial equation of degree 1 also satisfied by the identity matrix. A slightly less trivial example is obtained for

$$A = \begin{pmatrix} \varepsilon & 1 & \varepsilon \\ 1 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 1 \end{pmatrix} .$$

The characteristic equation is

$$x^3 \oplus 3 = 1x^2 \oplus 2x .$$

It is easily seen that A satisfies both $x^3 = 2x$ and $3 = 1x^2$. These equations have been obtained by a ‘partitioning’ of the characteristic equation; ‘adding’ these partitioned equations, one obtains the characteristic equation again. In this case, $3 = 1x^2$ is of degree 2. We may call a polynomial equation of least degree satisfied by a matrix, with the additional requirement that the coefficient of the highest power be equal to e , a *minimal polynomial equation* of this matrix. This is the counterpart of the notion of the minimal polynomial in conventional algebra and it is known that this minimal polynomial is a divisor of the characteristic polynomial [61]. In the present situation, it is not clear whether the minimal polynomial equation of a matrix is unique and how to extend the idea of division of polynomials to polynomial equations. In Chapter 3, a more detailed discussion on polynomials is given.

2.4 PETRI NETS

2.4.1 Definition

Petri nets are directed bipartite graphs. They are named after C.A. Petri, see [96]. The set of nodes $\mathcal{V}$ is partitioned into two disjoint subsets $\mathcal{P}$ and $\mathcal{Q}$. The elements of $\mathcal{P}$ are called places and those of $\mathcal{Q}$ are called transitions. Places will be denoted $p_i, i = 1, \dots, |\mathcal{P}|$, and transitions $q_j, j = 1, \dots, |\mathcal{Q}|$. The directed arcs go from a place to a transition or vice versa. Since a Petri net is bipartite, there are no arcs from place to place or from transition to transition. In the graphical representation of Petri nets, places are drawn as circles and transitions as bars (the orientation of these bars can be anything). An example of a Petri net is given in Figure 2.12.

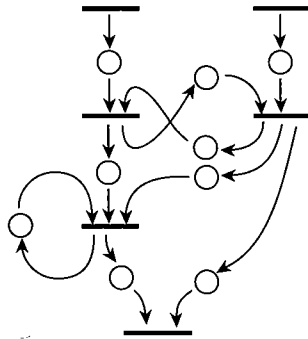


Figure 2.12: A Petri net with sources and sinks

In order to complete the formal definition of a Petri net, an initial marking must be introduced. The initial marking assigns a nonnegative integer μ_i to each place p_i . It is said that p_i is marked with μ_i initial tokens. Pictorially, μ_i dots (the tokens) are placed in the circle representing place p_i . The components μ_i form the vector μ , called the initial marking of the Petri net.

Definition 2.23 A Petri net is a pair (G, μ) , where $G = (\mathcal{E}, \mathcal{V})$ is a bipartite graph

with a finite number of nodes (the set $\mathcal{V}$) which are partitioned into the disjoint sets $\mathcal{P}$ and $\mathcal{Q}$; $\mathcal{E}$ consists of pairs of the form (p_i, q_j) and (q_j, p_i) , with $p_i \in \mathcal{P}$ and $q_j \in \mathcal{Q}$; the initial marking μ is a $|\mathcal{P}|$ -vector of nonnegative integers.

Notation 2.24 If $p_i \in \pi(q_j)$ (or equivalently $(p_i, q_j) \in \mathcal{E}$), then p_i is an upstream place for q_j . Downstream places are defined likewise. The following additional notation will also be used when we have to play with indices: if $p_i \in \pi(q_j)$, we write $i \in \pi^q(j), i = 1, \dots, |\mathcal{P}|, j = 1, \dots, |\mathcal{Q}|$; similarly, if $q_j \in \pi(p_i)$, we write $j \in \pi^p(i)$, with an analogous meaning for σ^p or σ^q . ■

Roughly speaking, places represent conditions and transitions represent events. A transition (i.e. an event) has a certain number of input and output places representing the pre-conditions and the post-conditions of the event, respectively. The presence of a token in a place is interpreted as the condition associated with that place being fulfilled. In another interpretation, μ_i tokens are put into a place to indicate that μ_i data items or resources are available. If a token represents data, then a typical example of a transition is a computation step for which these data are needed as an input. In Figure 2.13, the Petri net of the production example in §1.2.3 is given.

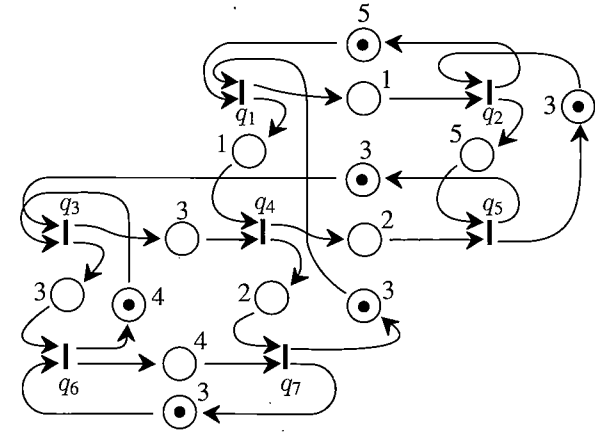


Figure 2.13: Petri net of the manufacturing system of §1.2.3

The tokens in this figure are located in such a way that machine M_1 starts working on product P_2 and M_2 on P_1 . Note that M_1 cannot work on P_3 in Figure 2.13.

Within the classical Petri-net setting, the marking of the Petri net is identified with the state. Changes occur according to the following rules:

- A transition is said to be *enabled* if each upstream place contains at least one token.
- A *firing* of an enabled transition removes one token from each of its upstream places and adds one token to each of its downstream places.

Remark 2.25 The enabling rule given above is not the most general one. Sometimes integer valued ‘weights’ are attached to arcs. A transition is enabled if the upstream place contains at least the number of tokens given by the weight of the connecting arc. Similarly, after the firing of a transition, a downstream place receives the number of tokens given by the weight of the connecting arc. Instead of talking about such ‘weights’, one sometimes talks about multi-arcs; the weight equals the number of arcs between a transition and a place or between a place and a transition. In terms of ‘modeling power’, see [96] and [108] for a definition, this generalization is not more powerful than the rules which will be used here. The word ‘weight’ of an arc will be used in a different sense later on. ■

For q_j to be enabled, we need that

$$\mu_i \geq 1, \quad \forall p_i \in \pi(q_j).$$

If the enabled transition q_j fires, then a new marking $\tilde{\mu}$ is obtained with

$$\tilde{\mu}_i = \begin{cases} \mu_i - 1 & \text{if } p_i \in \pi(q_j), \\ \mu_i + 1 & \text{if } p_i \in \sigma(q_j), \\ \mu_i & \text{otherwise.} \end{cases}$$

In case both $p_i \in \pi(q_j)$ and $p_i \in \sigma(q_j)$ for the same place p_i , then $\tilde{\mu}_i = \mu_i$.

In Figure 2.13, once M_1 has completed its work on P_2 and M_2 its work on P_1 , then Figure 2.14 is obtained. The next transitions that are now enabled are described

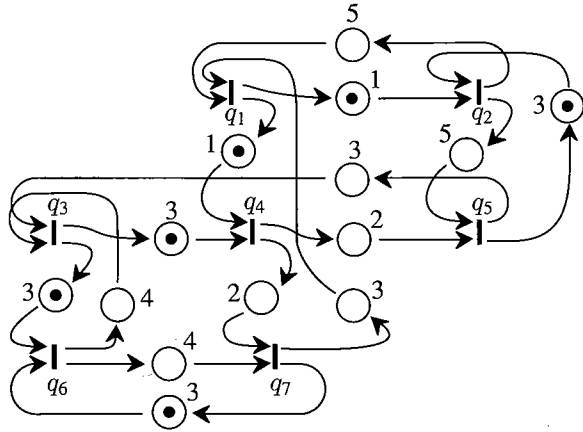


Figure 2.14: The tokens after the firing of q_1 and q_3

by the combinations (M_1, P_3) , (M_2, P_2) and (M_3, P_1) . Note that in general the total amount of tokens in the net is not left invariant by the firing of a transition, although this does not happen in Figure 2.13. If we have a ‘join’-type transition (Figure 2.15), which is called an *and-convergence*, or a ‘fork’-type transition (Figure 2.16), called an *and-divergence*, then clearly the number of tokens changes after a firing has taken

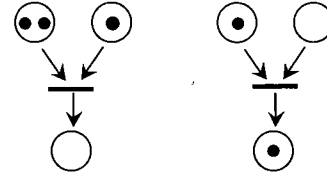


Figure 2.15: And-convergence before and after firing

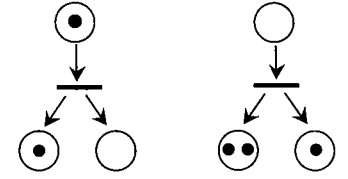


Figure 2.16: And-divergence before and after firing

place. In the same vein, an *or-convergence* refers to two or more arcs entering one place, and an *or-divergence* refers to two or more arcs originating from one place.

A transition without predecessor(s) is called a source transition or simply a source; it is enabled by the outside world. Similarly, a transition which does not have successor(s), is called a sink (or sink transition). Sink transitions deliver tokens to the outside world. In Figure 2.12 there are two transitions which are sources and there is one transition which is a sink. If there are no sources in the network, as in Figure 2.17, then we talk about an *autonomous network*. It is assumed that only transitions can be sources or sinks. This is no loss of generality, since one can always add a transition upstream or downstream of a place if necessary. A source transition is an input of the network, a sink transition is an output of the network.

The structure of a place p_i having two or more output transitions, as shown in Figure 2.18, is referred to as a *conflict*, since the transitions are competing for the

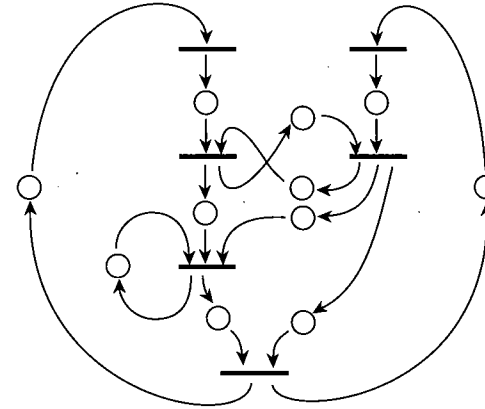


Figure 2.17: An autonomous Petri net

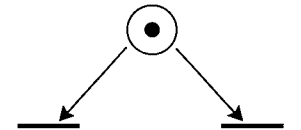


Figure 2.18: Part of a Petri net with a conflict

token in the place. The transitions concerned will be said to be *rivals*. There are no general rules as to which transition should fire first. One says that a Petri net with such an or-divergence exhibits nondeterminism. Depending on the application, one can also talk about a choice (between the transitions) or a decision.

The firing of an enabled transition will change the distribution of tokens. A sequence of firings will result in a sequence of markings. A marking $\bar{\mu}$ is said to be *reachable* from a marking μ if there exists a sequence of enabled firings that transforms μ into $\bar{\mu}$.

Definition 2.26 (Reachability tree) The reachability tree of a Petri net (G, μ) is a tree with nodes in $\mathbb{N}^{|P|}$ which is obtained as follows: the initial marking μ is a node of this tree; for each q enabled in μ , the marking $\bar{\mu}$ obtained by firing q is a new node of the reachability tree; arcs connect nodes which are reachable from one another in one step; this process is applied recursively from each such $\bar{\mu}$.

Definition 2.27 (Reachability graph) The reachability graph is obtained from the reachability tree by merging all nodes corresponding to the same marking into a single node.

Take as an example the Petri net depicted in Figure 2.19. The initial marking

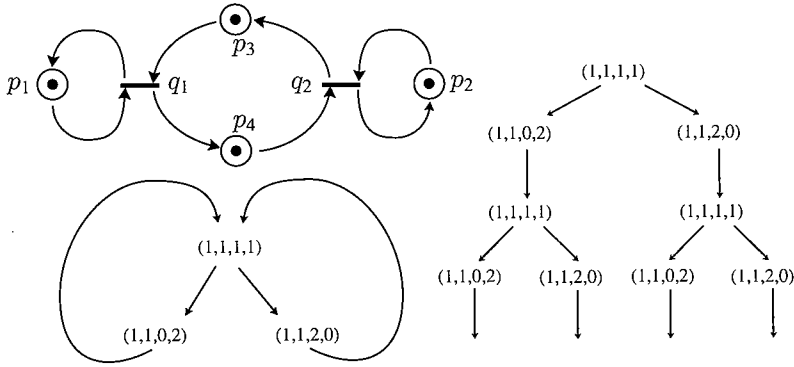


Figure 2.19: A Petri net with corresponding reachability graph and reachability tree

is $(1, 1, 1, 1)$. Both transitions are enabled. If q_1 fires first, the next marking is $(1, 1, 0, 2)$. If q_2 fires instead, the marking becomes $(1, 1, 2, 0)$. From $(1, 1, 0, 2)$, only the initial marking can be reached immediately by firing q_2 ; starting from $(1, 1, 2, 0)$, only q_1 can fire, which also leads to the initial marking. Thus it has been shown that there are three different markings in the reachability graph of $(1, 1, 1, 1)$.

Definition 2.28 For a Petri net with n transitions and m places, the incidence matrix $G = (G_{ij})$ is an $n \times m$ matrix of integers $-1, 0$ and $+1$. The entry G_{ij} is defined by

$$G_{ij} = G_{ij}^{\text{out}} - G_{ij}^{\text{in}} ,$$

where $G_{ij}^{\text{out}} = 1$ (0) if there is an (no) arc from q_i to p_j and $G_{ij}^{\text{in}} = 1$ (0) if there is an (no) arc from p_j to q_i . Matrices G^{out} and G^{in} are defined as $G^{\text{out}} = (G_{ij}^{\text{out}})$ and $G^{\text{in}} = (G_{ij}^{\text{in}})$, respectively.

Note that G does not uniquely define a Petri net since, if $G_{ii} = 0$, a path including exactly one place around the transition q_i is also possible. A circuit consisting of one transition and one place is called a *loop* in the context of Petri nets. If each place in the Petri net had only one upstream and one downstream transition, then the incidence matrix G would reduce to the well-known incidence matrix F introduced in Definition 2.10 by identifying each place p with the unique arc from $\pi(p)$ to $\sigma(p)$.

Transition q_j is enabled if and only if a marking μ is given such that

$$\mu \geq (G^{\text{in}})' e_j ,$$

where $e_j = (0, \dots, 0, 1, 0, \dots, 0)'$, with the 1 being the j -th component. If this transition fires, then the next marking $\tilde{\mu}$ is given by

$$\tilde{\mu} = \mu + G' e_j .$$

A destination marking $\bar{\mu}$ is reachable from μ if a firing sequence $e_{j_1}, \dots, e_{j_d}$ exists such that

$$\bar{\mu} = \mu + G' \sum_{l=1}^d e_{j_l} .$$

Hence a necessary condition for $\bar{\mu}$ to be reachable from μ is that an n -vector x of nonnegative integers exists such that

$$G' x = \bar{\mu} - \mu . \quad (2.9)$$

The existence of such a vector x is not a sufficient condition; for a counterexample see for instance [108]. The vector x does not reflect the order in which the firings take place. In the next subsection a necessary and sufficient condition for reachability will be given for a subclass of Petri nets. An integer solution x to (2.9), with its components not necessarily nonnegative, exists if and only if $\bar{\mu}' y = \mu' y$, for any y that satisfies $G y = 0$. The necessity of this statement easily follows if one takes the inner products of the left- and right-hand sides of (2.9) with respect to y . The sufficiency is easily shown if one assumes that x does not exist, i.e. $\text{rank}[G] < \text{rank}[G, \bar{\mu} - \mu]$ —the notation $[G, \bar{\mu} - \mu]$ refers to the matrix consisting of G and the extra column $\bar{\mu} - \mu$. Then a vector y exists with $y' G' = 0$ and $y' (\bar{\mu} - \mu) \neq 0$, which is a contradiction.

2.4.2 Subclasses and Properties of Petri Nets

In this subsection we introduce some subclasses of Petri nets and analyze their basic properties. Not all of these properties are used later on; this subsection is also meant to give some background information on distinct features of Petri nets. The emphasis will be on event graphs.

Definition 2.29 (Event graph) A Petri net is called an event graph if each place has exactly one upstream and one downstream transition.

Definition 2.30 (State machine) A Petri net is called a state machine if each transition has exactly one upstream and one downstream place.

Event graphs have neither or-divergences nor or-convergences. In event graphs each place together with its incoming and outgoing arcs can be interpreted as an arc itself, connecting the upstream and downstream transition, directly.

In the literature, event graphs are sometimes also referred to as *marked graphs* or as *decision free Petri nets*. Figure 2.20 shows both a state machine which is not

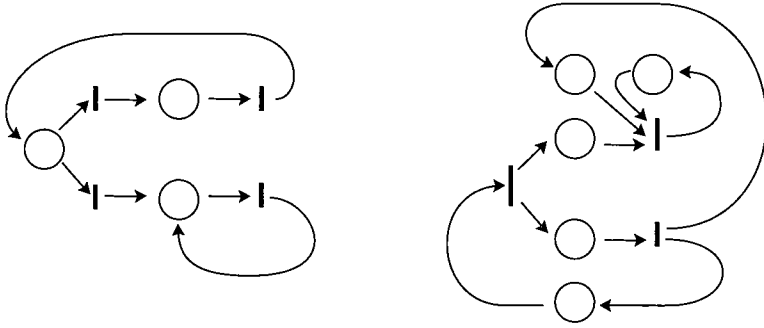


Figure 2.20: A state machine and an event graph

an event graph and an event graph which is not a state machine. An event graph does not allow and cannot model conflicts; a token in a place can be consumed by only one predetermined transition. In an event graph several places can precede a given transition. It is said that event graphs can model synchronization. State machines do not admit synchronization; however, they do allow competition. The number of tokens in an autonomous state machine never changes (Petri nets in which the number of tokens remains constant are called strictly conservative; a discussion follows later).

It can be shown [108] that state machines are equivalent to the finite state machines or automata in theoretical computer science. Each automaton can be rephrased as a state machine Petri net. This shows that Petri nets have more modeling power than automata.

Basic definitions and properties of Petri nets are now given.

Definition 2.31 (Bounded and safe nets) A Petri net, with initial marking μ , is said to be k -bounded if the number of tokens in each place does not exceed a finite number k for any marking reachable from μ . Instead of 1-bounded ($k = 1$) Petri nets, one speaks of safe Petri nets.

Concerning (practical) applications, it is important to know whether one deals with a bounded or safe Petri net, since one is then sure that there will be no overflows in the buffers or registers, no matter what the firing sequence will be.

Definition 2.32 (Live net) A Petri net is said to be live for the initial marking μ if for each marking ν reachable from μ and for each transition q , there exists a marking o which is reachable from ν and such that q is enabled on o . A Petri net which is not live is called *deadlocked*.

A Petri net is *deadlocked* if its reachability tree has a marking where a transition, or a set of transitions, can never fire whatever the firing sequences of the other transitions. For a live Petri net, whatever the finite initial sequence of firings, from that point onwards, any arbitrary transition can be fired an infinite number of times.

An example of a live net is a state machine the underlying graph of which is strongly connected and the initial marking of which has at least one token.

Definition 2.33 (Consistent net) A Petri net is called *consistent* (weakly consistent) if there exists a firing sequence, characterized by the x -vector with positive (non-negative) integers as components, such that $G'x = 0$, where G is the incidence matrix.

In a consistent Petri net we can choose a finite sequence of firings such that repeating this sequence results in a periodic behavior.

Definition 2.34 (Synchronous net) A consistent net is called *synchronous* if the only solutions x of $G'x = 0$ are of the form $x = k(1, 1, \dots, 1)'$.

Definition 2.35 (Strictly conservative net) A Petri net with initial marking μ is called *strictly conservative* if, for all reachable markings $\bar{\mu}$, we have $\sum_{p_i \in P} \bar{\mu}_i = \sum_{p_i \in P} \mu_i$.

Definition 2.36 (Conservative net) A Petri net with initial marking μ is called *conservative* if positive integers c_i exist such that, for all reachable markings $\bar{\mu}$, we have $\sum_{p_i \in P} c_i \bar{\mu}_i = \sum_{p_i \in P} c_i \mu_i$.

Theorem 2.37 The number of tokens in any circuit of an event graph is constant.

Proof If a transition is part of a circuit, then exactly one of its incoming arcs and one of its outgoing arcs belong to the circuit. The firing of the transition removes one token from the upstream place connected to the incoming arc (it may remove tokens from other places as well, but they do not belong to the circuit) and it adds one token to the downstream place connected to the outgoing arc (it may add tokens to other downstream places as well, but they do not belong to the circuit). ■

An event graph is not necessarily strictly conservative. Consider an and-convergence, where two circuits merge. The firing of this transition removes one token from each of the upstream places and adds one token to the only downstream place. At an and-divergence, where two circuits split up, the firing of the transition removes one token from the only upstream place and adds one token to each of the downstream places.

Theorem 2.38 An autonomous event graph is live if and only if every circuit contains at least one token with respect to the initial marking.

Proof

Only if part: If there are no tokens in a circuit of the initial marking of an event graph, then this circuit will remain free of tokens and thus all transitions along this circuit never fire.

If part: If a transition is never enabled by any firing sequence, then by backtracking token-free places, one can find a token-free circuit. Indeed, if in an event graph a transition never fires, there is at least one upstream transition that never fires also (this statement cannot be made for general Petri nets). This backtracking is only possible if each place has a transition as predecessor and each transition has at least one place as predecessor. This holds for autonomous event graphs. Thus the theorem has been proved. ■

Theorem 2.39 *For a connected event graph, with initial marking μ , a firing sequence can lead back to μ if and only if it fires every transition an equal number of times.*

Proof In a connected event graph all transitions are either and-divergences, and-convergences or they are simple, i.e. they have one upstream place as well as one downstream place. These categories may overlap one another. If an and-divergence is enabled and it fires, then the number of tokens in all downstream places is increased by one. In order to dispose of these extra tokens, the downstream transitions in each of these places must fire also (in fact, they must fire as many times as the originally enabled transition fired in order to keep the number of tokens of the places in between constant). If an and-convergence wants to fire, then the upstream transitions of its upstream places must fire first in order that the number of tokens of the places in between do not change. Lastly, if a transition is simple and it can fire, both the unique downstream transition and upstream transition must fire the same number of times in order that the number of tokens in the places in between do not change. The reasoning above only fails for loops. Since loops are connected to the event graph also and since a firing of a transition in a loop does not change the number of tokens in the place in the loop, these loops can be disregarded in the above reasoning. ■

This theorem states that the equation $G'x = 0$ has only one positive independent solution $x = (k, \dots, k)'$. An immediate consequence is that every connected event graph is synchronous.

Theorem 2.40 *Consider an autonomous live event graph. It is safe if and only if the total number of tokens in each circuit equals one.*

Proof The if part of the proof is straightforward. Now consider the only if part. Assume that the graph is safe and that the total number of tokens in a circuit ζ_k , indicated by $\mu(\zeta_k)$, is not necessarily one. Consider all circuits $(\zeta_1, \zeta_2, \dots, \zeta_m)$ passing through a place p_i and its upstream transition t_j . Bring as many tokens as possible to each of the upstream places of t_j and subsequently fire t_j as many times as possible. It can be seen that the maximum number of tokens that can be brought in p_i is bounded from above by $\min\{\mu(\zeta_1), \mu(\zeta_2), \dots, \mu(\zeta_m)\}$. In particular, if this minimum equals one, then this maximum number of tokens is less than or equal to one. Since the event graph is live, t_j can be enabled, and therefore this maximum equals one. ■

The following theorem is stated in [95].

Theorem 2.41 *In a live event graph $\bar{\mu}$ is reachable from μ if and only if $\bar{\mu}'y = \mu'y$, for any y that satisfies $G'y = 0$.*

This last theorem sharpens the result mentioned at the end of §2.4.1, where the condition $\bar{\mu}'y = \mu'y$ was only a necessary condition.

2.5 TIMED EVENT GRAPHS

The original theory of Petri nets deals with the ordering of events, and questions pertaining to when events take place are not addressed. However, for questions related to performance evaluation (how fast can a network produce?) it is necessary to introduce time. This can be done in two basic ways by associating durations with either transition firings or with the sojourn of tokens in places.

Durations associated with firing times can be used to represent production times in a manufacturing environment, where transitions represent machines, the length of a code in a computer science setting, etc. We adopt the following definition.

Definition 2.42 (Firing time) *The firing time of a transition is the time that elapses between the starting and the completion of the firing of the transition.*

We also adopt the additional convention that the tokens to be consumed by a transition remain in the preceding places during the firing time; they are called *reserved tokens*.

Durations associated with places can be used to represent transportation or communication time. When a transition produces a token into a place, this token cannot immediately contribute to the enabling of the downstream transitions; it must first spend some *holding time* in that place, which actually represents the time it takes to transport this token from the initial transition to the place.

Definition 2.43 (Holding time) *The holding time of a place is the time a token must spend in the place before contributing to the enabling of the downstream transitions.*

Observe that there is a basic asymmetry between both types of durations: firing times represent the actual time it takes to fire a transition, while holding times can be viewed as the minimal time tokens have to spend in places (indeed it is not because a specific token has completed its holding time in a place that it can immediately be consumed by some transition; it may be that no transition capable of consuming this token is enabled at this time). In practical situations, both types of durations may be present. However, as we shall see later on, if one deals with event graphs, one can disregard durations associated with transitions without loss of generality.

Roughly speaking, a Petri net is said to be timed if such durations are given as new data associated with the network. A basic dichotomy arises depending on whether these durations are constant or variable. Throughout the book, only the dependence on the index of the firing (the firing of transition q of index k is the k -th to be initiated), or on the index of the token (the token of p of index k is the k -th token of p to contribute enabling $\sigma(p)$) will be considered. The other possible dependences, like for instance the dependence on time, or on some possibly changing environment, will not be addressed.

The timing of a Petri net will be said to be variable if the firing times of a transition depend on the index of the firing or if the holding times of tokens in a place depend on the index of the token. The timing is constant otherwise.

In the constant case, the first, the k -th and $(k+1)$ -st firings of transition q take the same amount of time; this common firing time may however depend on q (see the examples below).

Remark 2.44 With our definitions, nothing prevents a transition from having several ongoing firings (indeed, a transition does not have to wait for the completion of an ongoing firing in order to initiate a new firing). If one wants to prevent such a phenomenon, one may add an extra place associated with this transition. This extra place should have the transition under consideration as unique predecessor and successor, and one token in the initial marking, as indicated in Figure 2.21. The addition of this loop models a mechanism that will be called a *recycling* of

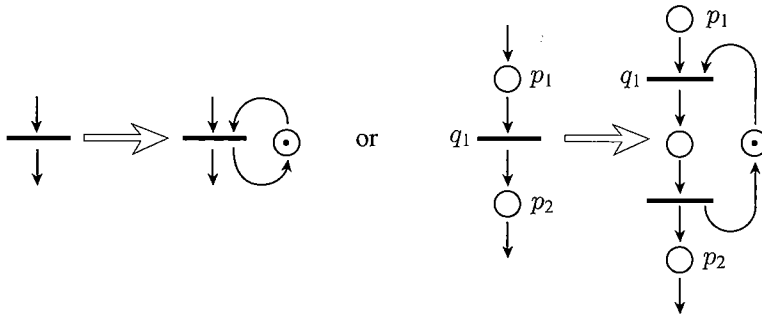


Figure 2.21: Recycling of a transition

the transition. Owing to this mechanism, the firings of the transition are properly serialized in the sense that its $(k+1)$ -st firing can only start after the completion of the k -th firing. ■

In the rest of this chapter, unless otherwise specified, $\oplus$ will be maximization and $\otimes$ addition, so that $\varepsilon = -\infty$ and $e = 0$. We will also use the symbol $\not\phi$ to denote subtraction.

2.5.1 Simple Examples

The global aim of the present section is to derive evolution equations for the variables $x_i(k)$, $i = 1, \dots, |Q|$, $k \geq 0$, not counting the sources and sinks, and where $x_i(k)$ is defined as the epoch at which transition q_i starts firing for the k -th time. Both constant and variable timings will be considered. For general Petri nets, such equations are difficult to derive. This problem will only be addressed in Chapter 9. For the moment, we shall confine ourselves to event graphs. We start with some

simple examples with constant timing before addressing the general case. The issue of the initial condition will not be addressed in these simple examples either (see §2.5.2.1). The general rule (to which we will return in the next subsection) is that transitions start firing as soon as they are enabled.

Example 2.45 (An autonomous timed event graph) The first example deals with the manufacturing system of §1.2.3, as depicted in Figure 1.5. The related Petri net, given in Figure 2.13, is the starting point for discussing the way the evolution equations are derived. The timing under consideration is limited to constant holding times on places, which are indicated in the figure. The firing times are all assumed to be 0. We have the following evolution equations:

$$\begin{aligned} x_1(k+1) &= 5x_2(k) \oplus 3x_7(k), \\ x_2(k+1) &= 1x_1(k+1) \oplus 3x_5(k), \\ x_3(k+1) &= 3x_5(k) \oplus 4x_6(k), \\ x_4(k+1) &= 1x_1(k+1) \oplus 3x_3(k+1), \\ x_5(k+1) &= 5x_2(k+1) \oplus 2x_4(k+1), \\ x_6(k+1) &= 3x_3(k+1) \oplus 3x_7(k), \\ x_7(k+1) &= 2x_4(k+1) \oplus 4x_6(k+1). \end{aligned}$$

In order to get this set of equations, one must first observe that owing to our assumption that holding times are constant, overtaking of tokens in places is not possible: the k -th token to enter a place will be the k -th token to leave that place (at least if the initial marking in that place is 0). If one uses this observation, the equation for x_6 (for instance) is obtained as follows: q_6 is enabled for the $(k+1)$ -st time at the latest of the two epochs when the $(k+1)$ -st token to enter the place between q_3 and q_6 completes its holding time there, and when the k -th token to enter the place between q_7 and q_6 completes its holding time. The difference between the arguments k and $k+1$ comes from the fact that the place between q_7 and q_6 has one token in the initial marking. If one now uses the definition of holding times, it is easily seen that the first of these two epochs is $x_3(k+1) + 3$, while the second one is $x_7(k) + 3$, which concludes the proof. There are clearly some further problems to be addressed regarding the initial condition, but let us forget them for the moment. In matrix form this equation can be written as

$$x(k+1) = A_0x(k+1) \oplus A_1x(k), \quad (2.10)$$

where

$$A_0 = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 5 & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 2 & \varepsilon & 4 & \varepsilon \end{pmatrix}, \quad A_1 = \begin{pmatrix} \varepsilon & 5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & 4 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}.$$

The equation is written in a more convenient way as

$$x(k+1) = Ax(k), \quad (2.11)$$

where $A = A_0^* A_1$ (see (1.22)), or, written out,

$$A = \begin{pmatrix} \varepsilon & 5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 \\ \varepsilon & 6 & \varepsilon & \varepsilon & 3 & \varepsilon & 4 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 3 & 4 & \varepsilon \\ \varepsilon & 6 & \varepsilon & \varepsilon & 6 & 7 & 4 \\ \varepsilon & 11 & \varepsilon & \varepsilon & 8 & 9 & 9 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 6 & 7 & 3 \\ \varepsilon & 8 & \varepsilon & \varepsilon & 10 & 11 & 7 \end{pmatrix}.$$

Remark 2.46 Both equations (2.11) and (1.29) describe the evolution of the firing times, the first equation with respect to the state x , the second one with respect to the output y . Using the notation of §1.2.3, one can check that

$$y(k+1) = Cx(k+1) = CAx(k)$$

and that

$$y(k+1) = My(k) = MCx(k),$$

where A is defined above, and where CA equals MC .

Conversely, it is easy to derive a Petri net from (2.11); such a net has 7 transitions (the dimension of the state vector) and 22 places (the number of entries in A which are not equal to ε). Each of these places has a token in the initial marking. The holding time associated with the place connecting transition j to transition i is given by the appropriate A_{ij} entry. Thus at least two different Petri nets exist which both yield the same set of evolution equations. In this sense these Petri nets are equivalent.

Example 2.47 (A nonautonomous timed event graph) The starting point for the next example is Figure 2.22, which coincides with Figure 2.12 with an initial marking added. The timing is again limited to places. The evolution equations are given by

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

where

$$\begin{aligned} A_0 &= \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ 3 & \varepsilon & \varepsilon \\ 3 & 4 & \varepsilon \end{pmatrix}, \quad A_1 = \begin{pmatrix} \varepsilon & 4 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \quad A_2 = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 2 \end{pmatrix}, \\ B_0 &= \begin{pmatrix} 1 & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix}, \quad B_1 = \begin{pmatrix} \varepsilon & \varepsilon \\ \varepsilon & 5 \\ \varepsilon & \varepsilon \end{pmatrix}, \end{aligned}$$

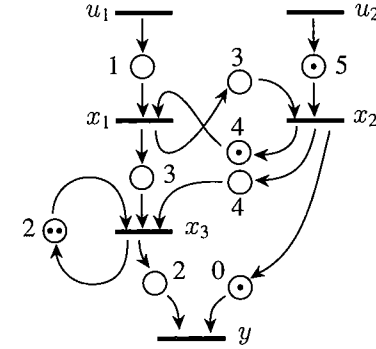


Figure 2.22: Petri net of Example 2.47

$$C_0 = \begin{pmatrix} \varepsilon & \varepsilon & 2 \end{pmatrix}, \quad C_1 = \begin{pmatrix} \varepsilon & e & \varepsilon \end{pmatrix}.$$

If one uses A_0^* , this system can be written as

$$\begin{aligned} x(k+1) &= \begin{pmatrix} \varepsilon & 4 & \varepsilon \\ \varepsilon & 7 & \varepsilon \\ \varepsilon & 11 & \varepsilon \end{pmatrix} x(k) \oplus \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 2 \end{pmatrix} x(k-1) \\ &\oplus \begin{pmatrix} 1 & \varepsilon \\ 4 & \varepsilon \\ 8 & \varepsilon \end{pmatrix} u(k+1) \oplus \begin{pmatrix} \varepsilon & \varepsilon \\ \varepsilon & 5 \\ \varepsilon & 9 \end{pmatrix} u(k), \\ y(k) &= \begin{pmatrix} \varepsilon & \varepsilon & 2 \end{pmatrix} x(k) \oplus \begin{pmatrix} \varepsilon & e & \varepsilon \end{pmatrix} x(k-1). \end{aligned}$$

This equation can be put into standard form by augmenting the state space: if one defines

$$\tilde{x}(k) = (x_1(k), x_1(k-1), x_2(k), x_2(k-1), x_3(k), x_3(k-1), u_1(k), u_2(k))',$$

the system can be written as

$$\begin{aligned} \tilde{x}(k+1) &= \tilde{A} \tilde{x}(k) \oplus \tilde{B} u(k+1), \\ y(k) &= \tilde{C} \tilde{x}(k), \end{aligned} \tag{2.12}$$

where

$$\tilde{A} = \begin{pmatrix} \varepsilon & \varepsilon & 4 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 7 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 5 \\ \varepsilon & \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 11 & \varepsilon & \varepsilon & 2 & \varepsilon & 9 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \quad \tilde{B} = \begin{pmatrix} 1 & \varepsilon \\ \varepsilon & \varepsilon \\ 4 & \varepsilon \\ \varepsilon & \varepsilon \\ 8 & \varepsilon \\ \varepsilon & \varepsilon \\ e & \varepsilon \\ \varepsilon & e \end{pmatrix},$$

and

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

Further simplifications are possible in this equation. Since x_1 is not observable (see [72]) and since it does not influence the dynamics of x_2 or x_3 either, it can be discarded from the state, as can u_1 . Thus a five-dimensional state vector suffices. Equation (2.12) is still not in the standard form since the argument of u is $k+1$ instead of k . If one insists on the ‘precise’ standard form, it can be shown that (2.12) and

$$\begin{aligned} \hat{x}(k+1) &= \tilde{A}\hat{x}(k) \oplus \tilde{A}\tilde{B}u(k), \\ y(k) &= \tilde{C}\hat{x}(k) \oplus \tilde{C}\tilde{B}u(k), \end{aligned} \quad (2.13)$$

are identical in the sense that their γ -transforms (see Chapter 1) are identical. This is left as an exercise to the reader. The latter equation does have the standard form, though there is a direct throughput term (the input $u(k)$ has a direct influence on the output $y(k)$). ■

Example 2.48 (Discrete analogue of the system of §1.2.7) The purpose of this example is to play again with the ‘counter’ description already alluded to in Chapter 1, and to show that discrete event systems may obey similar equations as some continuous systems, up to the problem of ‘quantization’. Figure 2.23 (left-hand side) represents a simple event graph: the three transitions are labelled u , x and y , holding

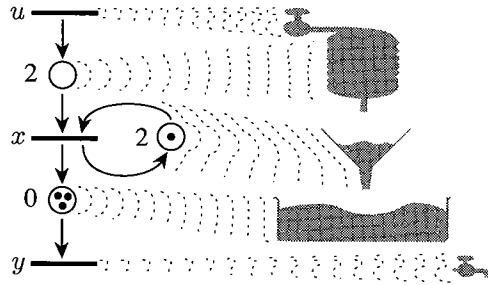


Figure 2.23: An event graph and its continuous analogue

times of places are indicated by numbers (firing times are all zero) and an initial marking is shown. With each transition, e.g. x , is associated a function of time t having the same name, e.g. $t \mapsto x(t)$, with the following meaning: $x(t)$ represents the number of firings of transition x up to time t , and it is assumed that $x(t) = e$ for $t < 0$. The following equations are obtained in the min-plus algebra:

$$x(t) = 1x(t-2) \oplus u(t-2); \quad y(t) = 3x(t),$$

where the delays in time originate from the holding times and where the coefficients are connected with the initial number of tokens in the places. By successive

substitutions, one obtains:

$$\begin{aligned} y(t) &= 3u(t-2) \oplus 4x(t-2) \\ &= 3u(t-2) \oplus 4u(t-4) \oplus 5x(t-4) \\ &\vdots \\ &= \bigoplus_{\substack{2 \leq \tau \leq t+2 \\ \tau \text{ even}}} h(\tau)u(t-\tau), \end{aligned}$$

where h is the function defined by (1.38). Observe that the min-summation can be limited to $t+2$ because $u(-1) = u(-2) = \dots = x(-1) = x(-2) = \dots = e$ and the coefficient of $x(t)$ is larger than that of $u(t)$. Indeed, for the same reason, the min-summation can be extended to $\tau < +\infty$, and also to $-\infty < \tau$ because $h(\tau)$ remains equal to 3 for values of τ below 2, whereas $u(t-\tau)$ is nonincreasing with τ . Finally, one obtains that

$$y(t) = \bigoplus_{\substack{-\infty < \tau < +\infty \\ \tau \text{ even}}} h(\tau)u(t-\tau),$$

which compares with (1.39), except that now τ ranges in $2\mathbb{Z}$ instead of $\mathbb{R}$.

The right-hand side of Figure 2.23 suggests the correspondence of continuous elements (extracted from Figure 1.13) with their discrete counterparts.

Recalling the mixing operation explained in the last paragraph of §1.2.7, we see that the discrete analogous operation consists here in ‘synchronizing’ two event graphs similar to that of Figure 2.23 by a join at their output transition y . ■

2.5.2 The Basic Autonomous Equation

The event graphs of this section are assumed to be autonomous. The nonautonomous case will be considered in §2.5.5. We now turn to the derivation of a set of evolution equations for event graphs with variable timing, under the general assumptions that

- the transitions start firing as soon as they are enabled;
- the tokens of a place start enabling the transitions downstream as soon as they have completed their holding times.

The problems that arise with variable timing are slightly more complex than in the preceding examples. The main reason for this lies in the fact that tokens can then overtake one another when traversing places or transitions. As we will see in Chapter 9, this precludes a simple ordering of events to hold, and event graphs with overtaking lose the nice linearity property emphasized in the preceding examples. The discussion will hence be limited to the case of event graphs with First In First Out (FIFO) places and transitions, where the ordering of events preserves linearity.

2.5.2.1 Initial Condition

We start with a first discussion of the *initial condition* which is here understood as a set of initial delays attached to the tokens of the initial marking in a way that generalizes what is often done in queuing theory. This topic will be revisited in §5.4.4.1 and §5.4.4.2, and further considered from a system-theoretic point of view in §5.4.4.3.

Assume that one starts looking at the system evolution at time $t = 0$, and that the piecewise constant function $N_i(t)$ describing the evolution of the number of tokens present in p_i , $i = 1, \dots, |\mathcal{P}|$, at time $t \in \mathbb{R}$, is right continuous. Let $N_i(0) = \mu_i$, where μ_i denotes the initial marking in place p_i .

The general idea behind the initial condition is as follows: the $N_i(0)$ ($= \mu_i$) tokens visible at time $t = 0$ in p_i are assumed to have entered p_i before time 0; at time 0, each token is completing its holding time or it is being consumed by the transition (namely it is a reserved token), or it is ready to be consumed. We can equivalently define the initial condition through the entrance times of the initial tokens, or through the vector of $\mathbb{R}$ -valued *lag times*, where

Definition 2.49 (Lag time) *The lag time of a token of the initial marking of p_i is the epoch when this token starts contributing to enabling $\sigma(p_i)$.*

However, these lag times should be compatible with the general rules that transitions fire as soon as they are enabled and that tokens start enabling the transition downstream as soon as they have completed their holding times. For instance

- if the lag time of an initial token exceeds its holding time, this token cannot have entered the place before time 0;
- if the lag times (which are possibly negative) are such that one of the transitions completes firing and consumes tokens of the initial marking before $t = 0$, these tokens cannot be part of the marking seen at time 0 since they must have left before time 0.

Definition 2.50 (Weakly compatible initial condition) *The initial condition of a timed event graph consists of an initial marking and a vector of lag times. This initial condition is weakly compatible if*

1. *the lag time of each initial token does not exceed its holding time;*
2. *the first epoch when a transition completes firing is nonnegative.*

2.5.2.2 FIFO Places and Transitions

A basic assumption that will be made throughout the chapter is that both places and transitions are First In First Out (FIFO) channels.

Definition 2.51 (FIFO place) *A place p_i is FIFO if the k -th token to enter this place is also the k -th which becomes available in this place.*

In view of the interpretation of holding times as communication or transportation times, this definition just means that the transportation or communication medium is overtake free. For instance, a place with constant holding times is FIFO.

Definition 2.52 (FIFO transition) *A transition q_j is FIFO if the k -th firing of q_j to start is also the k -th to complete.*

The interpretation is that tokens cannot overtake one another because of the firing mechanism, namely the tokens produced by the $(k+1)$ -st firing of q_j to be initiated cannot enter the places of $\sigma(q_j)$ earlier than those of the k -th firing. For instance, a transition with constant firing times is always FIFO. If a transition is recycled, its $(k+1)$ -st firing cannot start before the completion of the k -th one, so that a recycled transition is necessarily FIFO, regardless of the firing times.

Definition 2.53 (FIFO event graph) *An event graph is FIFO if all its places and transitions are FIFO.*

A typical example of a FIFO timed event graph is that of a system with constant holding times and recycled transitions with possibly variable firing times. An event graph with constant holding and firing times is always FIFO, even if its transitions are not recycled. Since the FIFO property is essential in order to establish the evolution equations of the present section, it is important to keep in mind that:

The classes of timed event graphs considered throughout the book are those with

1. constant firing and holding times;
2. constant holding times and variable firing times, provided all transitions are recycled.

2.5.2.3 Numbering of Events

The following way of numbering the tokens that traverse a place and the firings of a transition will be adopted.

By convention, the k -th token, $k \geq 1$, of place p_i is the k -th token to contribute enabling $\sigma(p_i)$ during the evolution of the event graph, including the tokens of the initial marking. The k -th firing, $k \geq 1$, of transition q_i is the k -th firing of q_i to be initiated, including the firings that consume initial tokens.

It may happen that two tokens in p_i contribute enabling $\sigma(p_i)$ at the same epoch, or that two firings of a transition are initiated at the same time (when the transition is not recycled). In this case, some ordering of these simultaneous events is chosen, keeping in mind that it should be compatible with the FIFO assumptions.

2.5.2.4 Dynamics

In what follows, the sequences of holding times $\alpha_i(k)$, $i = 1, \dots, |\mathcal{P}|$, $k \in \mathbb{Z}$, and of firing times $\beta_j(k)$, $j = 1, \dots, |\mathcal{Q}|$, $k \in \mathbb{Z}$, are assumed to be given nonnegative and finite real numbers. Initially, only the restriction of these sequences to $k \geq 1$ will be needed. However, we assume that these sequences can be continued to $k \leq 0$. Such a continuation is clear in the case of a constant timing, and we will see in due time how to define the continuation in more general circumstances (see §2.5.7).

We are now in a position to define the dynamics of the event graph more formally.

- The k -th token of place p_i incurs the holding time $\alpha_i(k)$.
- Once the k -th firing of transition q_j is enabled, the time for q_j to complete its k -th firing is the firing time $\beta_j(k)$. When this firing is completed, the reserved token is removed from each of the places of $\pi(q_j)$, and each place of $\sigma(q_j)$ receives one token.

We now state a few basic properties of the numbering in a FIFO event graph with a weakly compatible initial condition. For i such that $\mu_i \geq 1$, denote $w_i(1) \leq w_i(2) \leq \dots \leq w_i(\mu_i) \in \mathbb{R}$, the lag times of the initial tokens of place p_i ordered in a nondecreasing way.

Lemma 2.54 *If the initial condition is weakly compatible and if the timed event graph is FIFO, then for all k such that $1 \leq k \leq \mu_i$ and $\mu_i \geq 1$, the initial token with lag time $w_i(k)$ is also the k -th token of place p_i (that is the k -th token to enable $\sigma(p_i)$).*

Proof If this last property does not hold for some place p_i , then a token which does not belong to the initial marking of p_i , and which hence enters p_i after time 0 (the initial condition is weakly compatible), contributes to enabling $\sigma(p_i)$ before one of the tokens of the initial marking does. Since the tokens of the initial marking enter p_i before time 0 (the initial condition is weakly compatible), this contradicts the assumption that p_i is FIFO. ■

Lemma 2.55 *The firing of q_j that consumes the k -th token of p_i (for all $p_i \in \pi(q_j)$) is the k -th firing of q_j .*

Proof Owing to the numbering convention, the set of k -th tokens of $p_i \in \pi(q_j)$ enables q_j before the set of $(k+1)$ -st tokens. ■

Lemma 2.56 *The completion of the k -th firing of q_j , $k \geq 1$, produces the $(k+\mu_i)$ -th token of p_i , for all $p_i \in \sigma(q_j)$.*

Proof The FIFO assumptions on transitions imply that the completion of the k -th firing of q_j produces the k -th token to enter the places that follow p_i . The property follows immediately from the FIFO assumption on places and from Lemma 2.54. ■

2.5.2.5 Evolution Equations

Definition 2.57 (State variables, daters) *The state variable $x_j(k)$, $j = 1, \dots, |\mathcal{Q}|$, $k \geq 1$, of the event graph is the epoch when transition q_j starts firing for the k -th time, with the convention that for all q_i , $x_i(k) = \infty$ if q_i fires less than k times. These state variables will be called daters.*

These state variables are continued to negative values of k by the relation $x_j(k) = \varepsilon$, for all $k \leq 0$. Let

$$M = \max_{i=1, \dots, |\mathcal{P}|} \mu_i. \quad (2.14)$$

In what follows, we will adopt the convention that the $\oplus$ -sum over an empty set is ε . Define the $|\mathcal{Q}| \times |\mathcal{Q}|$ matrices $A(k, k)$, $A(k, k-1)$, $\dots$, $A(k, k-M)$, by

$$A_{jl}(k, k-m) \stackrel{\text{def}}{=} \left(\bigoplus_{\{i \in \pi^q(j) \mid \pi^p(i)=l, \mu_i=m\}} \alpha_i(k) \right) \otimes \beta_l(k-m), \quad (2.15)$$

and the $|\mathcal{Q}|$ -dimensional vector $v(k)$, $k = 1, \dots, M$, by

$$v_j(k) \stackrel{\text{def}}{=} \bigoplus_{\{i \in \pi^q(j) \mid \mu_i \geq k\}} w_i(k). \quad (2.16)$$

Theorem 2.58 *For a timed event graph with recycled transitions, the state vector $x(k) = (x_j(k))$ satisfies the evolution equations:*

$$x(k) = A(k, k)x(k) \oplus A(k, k-1)x(k-1) \oplus \dots \oplus A(k, k-M)x(k-M), \quad (2.17)$$

$k = M+1, M+2, \dots,$

with the initial conditions

$$x(k) = A(k, k)x(k) \oplus \dots \oplus A(k, k-M)x(k-M) \oplus v(k), \quad (2.18)$$

$k = 1, 2, \dots, M,$

where $x_j(k) \stackrel{\text{def}}{=} \varepsilon$ for all $k \leq 0$.

Proof We first prove that the variables $x_j(k)$, $j = 1, \dots, |\mathcal{Q}|$, satisfy the evolution equations:

$$x_j(k) = \bigoplus_{\{i \in \pi^q(j) \mid k > \mu_i\}} (\alpha_i(k) \otimes \beta_{\pi^p(i)}(k - \mu_i) \otimes x_{\pi^p(i)}(k - \mu_i)) \oplus \left(\bigoplus_{\{i \in \pi^q(j) \mid k \leq \mu_i\}} w_i(k) \right), \quad k = 1, 2, \dots \quad (2.19)$$

The k -th firing, $k \geq 1$, of transition q_j starts as soon as, for all $i \in \pi^q(j)$, the k -th token of p_i contributes to enabling q_j . In view of Lemmas 2.55 and 2.56, for

$k > \mu_i$, this k -th token is produced by the $(k - \mu_i)$ -th firing of transition $\pi(p_i)$, so that the epoch when this token contributes enabling $\sigma(p_i)$ is $\alpha_i(k) \otimes \beta_{\pi(p_i)}(k - \mu_i) \otimes x_{\pi(p_i)}(k - \mu_i)$. For $k \leq \mu_i$, this event takes place at time $w_i(k)$, in view of Lemma 2.54, which completes the proof of (2.19).

We now use associativity and commutativity of $\oplus$, together with our convention on $\oplus$ -sums over empty sets, to rewrite $x_j(k)$, $k > M$, as

$$\bigoplus_{m=0}^M \bigoplus_{l=1}^{|Q|} \bigoplus_{\{i \in \pi^q(j) | \pi^p(i)=l, \mu_i=m\}} \alpha_i(k) \otimes \beta_l(k-m) \otimes x_l(k-m).$$

The distributivity of $\otimes$ with respect to $\oplus$ implies in turn

$$x_j(k) = \bigoplus_{m=0}^M \bigoplus_{l=1}^{|Q|} \left(\bigoplus_{\{i \in \pi^q(j) | \pi^p(i)=l, \mu_i=m\}} \alpha_i(k) \right) \otimes \beta_l(k-m) \otimes x_l(k-m),$$

which completes the proof of (2.17), in view of the definition of A .

The proof of (2.18) follows the same lines (using the continuation of the functions $x_j(k)$ to ε for $k \leq 0$). ■

Remark 2.59 Owing to the dynamics, the first transition to complete its firing is necessarily within the set of transitions q_j having at least one token in the initial marking of p_i for all $p_i \in \pi(q_j)$. Since the set of tokens with the smallest lag times is the first to be consumed, the second weak compatibility condition in Definition 2.50 can be translated into the requirement that

$$\beta_j(1) \otimes v_j(1) \geq e, \quad (2.20)$$

for all j such that $\mu_i \geq 1 \forall p_i \in \pi(q_j)$, which can be seen as a first set of linear constraints on the lag times in view of (2.16). Similarly, the first weak compatibility relation is translated into the following additional set of linear constraints:

$$w_i(k) \leq \alpha_i(k), \quad i = 1, \dots, |\mathcal{P}|, \quad 1 \leq k \leq \mu_i. \quad (2.21)$$

For instance, if $w_i(k) = \alpha_i(k)$ for all $i = 1, \dots, |\mathcal{P}|$, $1 \leq k \leq \mu_i$, then the initial condition is weakly compatible, provided the condition $w_i(1) \leq w_i(2) \leq \dots \leq w_i(\mu_i)$ is satisfied. ■

Example 2.60 Consider the timed event graph of Figure 2.24. The firing times are assumed to be equal to e . The place connecting q_j to q_l is denoted p_{lj} . The holding times in this place are denoted $\alpha_{lj}(k)$, and the lag times of the initial tokens $w_{lj}(k)$. In order to have a FIFO event graph, places should be overtake free. This will always be true for p_{11} and p_{22} , regardless of the holding time sequences in these places (since there is always at most one token each of these places). A simple sufficient condition ensuring that the other places are overtake free is that the associated holding time sequences are non decreasing in k (for instance constant).

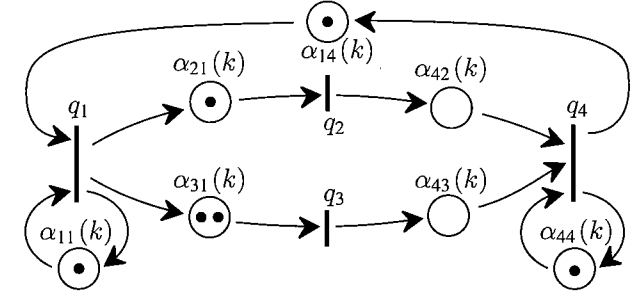


Figure 2.24: Event graph of Example 2.60

Under the assumption that the event graph is FIFO, the matrices and vectors involved in the evolution equations are

$$A(k, k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \alpha_{42}(k) & \alpha_{43}(k) & \varepsilon \end{pmatrix}, \quad A(k, k-1) = \begin{pmatrix} \alpha_{11}(k) & \varepsilon & \varepsilon & \alpha_{14}(k) \\ \alpha_{21}(k) & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \alpha_{44}(k) \end{pmatrix},$$

$$A(k, k-2) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \alpha_{31}(k) & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix},$$

and

$$v(1) = \begin{pmatrix} w_{11}(1) \oplus w_{14}(1) \\ w_{21}(1) \\ w_{31}(1) \\ w_{44}(1) \end{pmatrix}, \quad v(2) = \begin{pmatrix} \varepsilon \\ \varepsilon \\ w_{31}(2) \\ \varepsilon \end{pmatrix}.$$

The constraints (2.21) and (2.20) are translated into the bounds $w_{lj}(k) \leq \alpha_{lj}(k)$, and $w_{11}(1) \oplus w_{14}(1) \geq e$, $w_{21}(1) \geq e$, $w_{31}(1) \geq e$. ■

2.5.2.6 Simplifications

Firing Times The evolution equations (2.19) are unchanged if one sets all the firing times equal to e and if $\alpha_i(k)$ receives the value $\alpha_i(k) \otimes \beta_{\pi(p_i)}(k - \mu_i)$. Thus, one can always modify the holding times in order to get an 'equivalent' event graph with firing times of duration e , where equivalence means that the epochs when transitions fire are the same in both systems.

It may be assumed without loss of generality that the firing times are equal to $e = 0$.

Observe that under this assumption the state variable $x_j(k)$ is also the epoch when transition q_j completes its k -th firing. Graphically, this is exemplified in Figure 2.25. If a firing time were assigned to a transition within an event graph, then this time can always be assigned to (i.e. added to) the holding times of all the upstream places. Consider Figure 2.25a, in which the transition has a firing time of 5 time

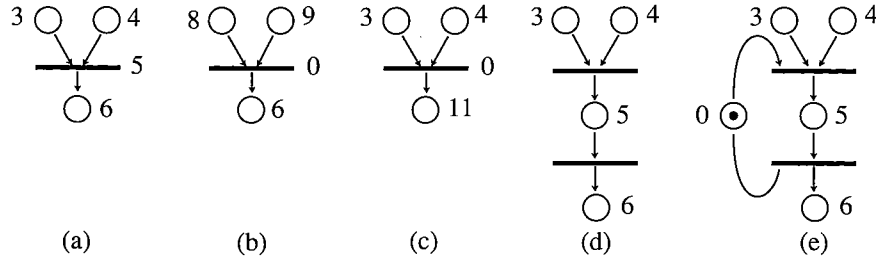


Figure 2.25: Firing times set to ε

units. The holding times of the places are 3, 4 and 6 as indicated in this figure. Figure 2.25b shows assignment of the firing time to the places. The firing of the transition in Figure 2.25b, which is instantaneous, corresponds to the completion of the firing of the transition in Figure 2.25a. Another, similar, solution is provided in Figure 2.25c, where the holding time has been assigned to all the downstream places. The firing of the transition in this figure corresponds to the initiation of the firing of the transition in Figure 2.25a.

A different solution is provided in Figure 2.25d. In this figure, both the beginning and completion of the firing are now explicitly represented. In Figure 2.25e, the holding time at the transitions is also 0, but in contrast to the previous solutions the transitions cannot fire twice (or more times) within 5 time units. The transitions cannot be engaged in (partly) parallel activities.

In what follows, we shall therefore often assume that the firing times are zero. The practical implication of this mathematical simplification is clear in the constant firing and holding time case. In the variable case, one should however always keep in mind that the only meaningful initial situation is that of variable firing times (on recycled transitions) and constant holding times.

Initial Condition The end of this subsection is devoted to a discussion of the initial condition. It is shown that Equation (2.18), defining the initial condition, can be further simplified whenever the lag times satisfy certain additional and natural constraints.

For all i such that $\mu_i > 0$, denote by $y_i(k)$, $k \leq 0$, the *entrance time* function associated with place p_i , defined by the relation

$$y_i(k - \mu_i) \stackrel{\text{def}}{=} \begin{cases} w_i(k) \not\propto \alpha_i(k) & \text{if } 1 \leq k \leq \mu_i ; \\ \varepsilon & \text{if } k > \mu_i , \end{cases} \quad (2.22)$$

where we recall that $\not\propto$ denotes conventional subtraction.

The initial condition is said to be *compatible* if it is weakly compatible and if for any pair of places p_i and p_j which follow the same transition, the entrance times $y_i(k)$ and $y_j(k)$ coincide provided $k \geq \min(\mu_i, \mu_j)$.

Definition 2.61 (Compatible initial condition) *The initial condition is compatible if it is weakly compatible and if there exist functions $z_j(k)$, $j = 1, \dots, |Q|$, $k \leq 0$, such that*

$$y_i(k) = z_{\pi^p(i)}(k) , \quad \forall i, k \text{ such that } -\mu_i + 1 \leq k \leq 0 . \quad (2.23)$$

This condition is quite natural, should the initial condition result from a past evolution of the event graph: for instance, the last tokens of the initial marking to enter two places p_i and p_j that follow the same transition q_j , have then been produced at the same time $z_j(0)$ by a firing of q_j .

Let

$$M_j \stackrel{\text{def}}{=} \max_{i \in \sigma^q(j)} (\mu_i) .$$

Observe that the function $z_j(k)$ is only defined through (2.23) for $-M_j < k \leq 0$, provided $M_j \geq 1$. For other values of k , or if $M_j = 0$, we take $z_j(k) = \varepsilon$.

Instead of the former continuation of $x(k)$ to $k \leq 0$ (which consisted in taking $x_j(k) = \varepsilon$ for $k \leq 0$), we now take

$$x_j(k) = z_j(k) , \quad \forall k \leq 0 , \quad j = 1, \dots, |Q| . \quad (2.24)$$

Corollary 2.62 *For a FIFO timed event graph with a compatible initial condition, the state vector $x(k) = (x_j(k))$ satisfies the evolution equations:*

$$x(k) = A(k, k)x(k) \oplus A(k, k-1)x(k-1) \oplus \dots \oplus A(k, k-M)x(k-M) , \quad k = 1, 2, \dots , \quad (2.25)$$

provided the continuation of $x(k)$ to negative values of k is the one defined by (2.24).

Proof By successively using (2.22) and (2.23), one gets

$$\begin{aligned} \bigoplus_{\{i \in \pi^q(j) | k \leq \mu_i\}} w_i(k) &= \bigoplus_{\{i \in \pi^q(j) | k \leq \mu_i\}} \alpha_i(k) \otimes y_i(k - \mu_i) \\ &= \bigoplus_{\{i \in \pi^q(j) | k \leq \mu_i\}} \alpha_i(k) \otimes z_{\pi^p(i)}(k - \mu_i) , \end{aligned}$$

for all $k = 1, 2, \dots$, so that one can rewrite (2.19) as

$$x_j(k) = \bigoplus_{\{i \in \pi^q(j)\}} (x_{\pi^p(i)}(k - \mu_i) \otimes \alpha_i(k)) , \quad k = 1, 2, \dots , \quad (2.26)$$

when using the continuation of x proposed in (2.24). Equation (2.25) follows immediately from (2.26). ■

Remark 2.63 A simple example of compatible initial condition is obtained when choosing $w_i(k) = \alpha_i(k)$ for all $1 \leq k \leq \mu_i$. Practically speaking, this means that all the initial tokens enter at time 0. This corresponds to the continuation

$$x_j(k) = \begin{cases} e & \text{if } -M_j < k \leq 0 ; \\ \varepsilon & \text{if } k \leq M_j . \end{cases} \quad (2.27)$$

Example 2.64 (Example 2.60 continued) If the initial condition is compatible, let

$$\begin{aligned} z_1(0) &\stackrel{\text{def}}{=} w_{11}(1) \not\prec \alpha_{11}(1) = w_{21}(1) \not\prec \alpha_{21}(1) = w_{31}(2) \not\prec \alpha_{31}(2) , \\ z_1(-1) &\stackrel{\text{def}}{=} w_{31}(1) \not\prec \alpha_{31}(1) , \\ z_4(0) &\stackrel{\text{def}}{=} w_{14}(1) \not\prec \alpha_{14}(1) = w_{44}(1) \not\prec \alpha_{44}(1) . \end{aligned}$$

Define

$$x(0) = \begin{pmatrix} z_1(0) \\ \varepsilon \\ \varepsilon \\ z_4(0) \end{pmatrix}, \quad x(-1) = \begin{pmatrix} z_1(-1) \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix}.$$

It is easily checked that

$$v(2) = A(2, 0)x(0)$$

and that

$$v(1) = A(1, 0)x(0) \oplus A(1, -1)x(-1) .$$

Thus

$$x(k) = A(k, k)x(k) \oplus A(k, k-1)x(k-1) \oplus A(k, k-2)x(k-2), \quad k = 1, 2, \dots$$

2.5.3 Constructiveness of the Evolution Equations

The first natural question concerning Equation (2.17) is: is it implicit or constructive? The main result of this section establishes that the evolution equations (2.17) are not implicit and that they allow one to recursively define the value of $x_j(k)$ for all $j = 1, \dots, |Q|$, and $k \geq 1$, provided the event graph under consideration is live.

Lemma 2.65 *The event graph is live if and only if there exists a permutation P of the coordinates for which the matrix $P'A(k, k)P$ is strictly lower triangular for all k .*

Proof If the matrix $P'A(k, k)P$ is strictly lower triangular for some permutation P , then there is no circuit with 0 initial marking, in view of the definition of $A(k, k)$ (see (2.15)). Conversely, if the event graph is live, the matrix $A(k, k)$ has no circuit, and

there exists a permutation of the coordinates that makes A strictly lower triangular. The proof is then concluded from Theorem 2.38. ■

Observe that the fact that P does not depend on k comes from the fact that the support of $A(k, k)$ does not depend on k (by 'the support' of matrix A we mean the matrix S with the same dimension as A defined by $S_{ij} = 1_{A_{ij} \neq \varepsilon}$).

If the matrix $P'A(k, k)P$ is strictly lower triangular, $A^n(k, k) = \varepsilon$ for $n \geq |Q|$, and the matrix

$$A^*(k, k) \stackrel{\text{def}}{=} e \oplus A(k, k) \oplus A^2(k, k) \oplus \dots$$

is finite. Let

$$\bar{A}(k, k-l) \stackrel{\text{def}}{=} A^*(k, k)A(k, k-l), \quad k \in \mathbb{Z}, \quad l = 1, \dots, M, \quad (2.28)$$

and

$$\bar{v}(k) \stackrel{\text{def}}{=} A^*(k, k)v(k), \quad k \in \mathbb{Z},$$

with $v_j(k) \stackrel{\text{def}}{=} \varepsilon$ for $k \leq 0$ or $k > M$.

Theorem 2.66 *If the event graph is live, the evolution equations (2.17) and (2.18) can be rewritten as*

$$\begin{aligned} x(k) &= \bar{A}(k, k-1)x(k-1) \oplus \dots \oplus \bar{A}(k, k-M)x(k-M) \oplus \bar{v}(k), \\ &\quad k = 1, 2, \dots, \end{aligned} \quad (2.29)$$

where $x_j(k) \stackrel{\text{def}}{=} \varepsilon$, for all $k \leq 0$.

Proof From (2.17) and (2.18), we obtain by induction on n that

$$\begin{aligned} x(k) &= A^{n+1}(k, k)x(k) \\ &\quad \oplus \left(\bigoplus_{m=0}^n A^m(k, k) \right) \left(\bigoplus_{l=1}^M A(k, k-l)x(k-l) \oplus v(k) \right), \quad k = 1, 2, \dots \end{aligned}$$

Equation (2.29) follows from the last relation by letting n go to ∞ . ■

Remark 2.67 If the initial condition is compatible and if one now takes the continuation of $x(k)$ for $k \leq 0$, as defined in (2.24), the same type of arguments shows that (2.25) becomes

$$x(k) = \bigoplus_{l=1}^M \bar{A}(k, k-l)x(k-l), \quad k = 1, 2, \dots$$

Corollary 2.68 *If the event graph is live and if the holding times and the lag times are all finite, so are the state variables $x_j(k)$, $j = 1, \dots, |Q|$, $k \geq 1$.*

Proof The proof is by induction based on (2.29). ■

Remark 2.69 The matrix $\bar{A}(k, k-l)$, $l \geq 1$, has a simple graph-theoretic interpretation. Let $S(j', j, l)$ be the set of paths in the graph G of the event graph, of length at least 2, with initial transition $q_{j'}$, with final transition q_j , and such that the first two transitions are connected by a place with initial marking equal to l , while the other transitions are connected by places with 0 initial marking. It is easily checked, using the results of §2.4, that $\bar{A}_{jj'}(k, k-l)$ is defined by the relation

$$\bar{A}_{jj'}(k, k-l) = \bigoplus_{\{p=(j_1, i_1, j_2, i_2, \dots, i_{h-1}, j_h) \in S(j', j, l)\}} \bigotimes_{n=1}^{h-1} \alpha_{i_n}(k), \quad (2.30)$$

with the usual convention if the set $S(j', j, l)$ is empty. The entry $\bar{A}_{jj'}(k, k-l)$ is hence simply the longest path in $S(j', j, l)$. ■

Example 2.70 (Example 2.60 continued) We have

$$A^*(k, k) = \begin{pmatrix} e & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon \\ \varepsilon & \alpha_{42}(k) & \alpha_{43}(k) & e \end{pmatrix},$$

so that

$$\bar{A}(k, k-1) = \begin{pmatrix} \alpha_{11}(k) & \varepsilon & \varepsilon & \alpha_{14}(k) \\ \alpha_{21}(k) & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \alpha_{42}(k)\alpha_{21}(k) & \varepsilon & \varepsilon & \alpha_{44}(k) \end{pmatrix},$$

$$\bar{A}(k, k-2) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \alpha_{31}(k) & \varepsilon & \varepsilon & \varepsilon \\ \alpha_{43}(k)\alpha_{31}(k) & \varepsilon & \varepsilon & \varepsilon \end{pmatrix},$$

and

$$\bar{v}(1) = \begin{pmatrix} w_{11}(1) \oplus w_{14}(1) \\ w_{21}(1) \\ w_{31}(1) \\ w_{21}(1)\alpha_{42}(1) \oplus w_{31}(1)\alpha_{43}(1) \oplus w_{44}(1) \end{pmatrix}, \quad \bar{v}(2) = \begin{pmatrix} \varepsilon \\ \varepsilon \\ w_{31}(2) \\ w_{31}(2)\alpha_{43}(2) \end{pmatrix}.$$

Remark 2.71 (Equivalent event graph with positive initial marking) With the evolution equation (2.29), one can associate a *derived* event graph with the same set of transitions as the initial event graph, and where the initial marking is such that $\mu_i > 0$ for all places p_i . This event graph is equivalent to the initial one in the sense that corresponding transitions fire at the same times. The derived event graph associated with the event graph of Example 2.60 is given in Figure 2.26, left-hand side. This derived event graph can be defined from the original one by the following transformation rules:

1. take the same set of transitions as in the original event graph;
2. for each path of $S(j', j, l)$, $l \geq 1$, in the original event graph, create a place connecting j' to j with l tokens and with the weight of the path as holding time. ■

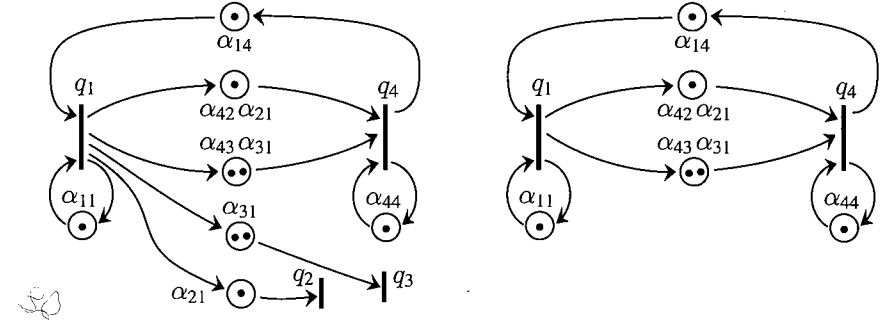


Figure 2.26: Illustration of Remarks 2.71 and 2.72

Remark 2.72 (Equivalent event graph with reduced state space) The dimension of the state space can be reduced using the following observation: the transitions followed by places all having a 0 initial marking, or equivalently the transitions q_j such that the entries of the j -th column of $A(k, k-l)$ are ε , are not present in the right-hand side of (2.29). Let Q' be the set of transitions followed by at least one place with a positive initial marking. One can take as reduced state variables $x_j(k)$, $j \in Q'$, $k \geq 1$. The remaining variables are obtained from them via (2.29). ■

Example 2.73 (Example 2.60 continued) Here we have $Q' = \{q_1, q_4\}$. With these new state variables, the evolution equations are reduced to

$$\begin{pmatrix} x_1(k) \\ x_4(k) \end{pmatrix} = \begin{pmatrix} \alpha_{11}(k) & \alpha_{14}(k) \\ \alpha_{42}(k)\alpha_{21}(k) & \alpha_{44}(k) \end{pmatrix} \begin{pmatrix} x_1(k-1) \\ x_4(k-1) \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \varepsilon \\ \alpha_{43}(k)\alpha_{31}(k) & \varepsilon \end{pmatrix} \begin{pmatrix} x_1(k-2) \\ x_4(k-2) \end{pmatrix} \oplus \begin{pmatrix} \bar{v}_1(k) \\ \bar{v}_4(k) \end{pmatrix}.$$

The event graph corresponding to these equations is depicted in Figure 2.26, right-hand side. It is obtained from the derived graph by deleting the transitions that do not belong to Q' .

The other state variables are obtained from the reduced state variables by the relation

$$\begin{pmatrix} x_2(k) \\ x_3(k) \end{pmatrix} = \begin{pmatrix} \alpha_{21}(k) & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix} \begin{pmatrix} x_1(k-1) \\ x_4(k-1) \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & \varepsilon \\ \alpha_{31}(k) & \varepsilon \end{pmatrix} \begin{pmatrix} x_1(k-2) \\ x_4(k-2) \end{pmatrix} \oplus \begin{pmatrix} \bar{v}_2(k) \\ \bar{v}_3(k) \end{pmatrix}.$$

The variables $(x_2(k), x_3(k))$ are output variables in the derived event graph. Equivalently, transitions q_2 and q_3 are sinks of the derived event graph. ■

2.5.4 Standard Autonomous Equations

The data of this section is a live event graph satisfying the evolution equations (2.29), with the reduction of the state space mentioned in Remark 2.72. We will assume that the transitions of Q' are numbered $1, \dots, |Q'|$, which introduces no loss of generality.

It may be desirable to replace the initial recurrence (2.29), which is of order M , by an equivalent recurrence of order 1. This is done by using the standard technique which consists in extending the state vector. As a new state vector, take the $(|Q'| \times M)$ -dimensional vector

$$\tilde{x}(k) \stackrel{\text{def}}{=} \begin{pmatrix} x(k) \\ x(k-1) \\ \vdots \\ x(k+1-M) \end{pmatrix}.$$

Let $\tilde{A}(k)$, $k \in \mathbb{Z}$, be the $(|Q'| \times M) \times (|Q'| \times M)$ matrix defined by the relation

$$\tilde{A}(k) = \begin{pmatrix} \bar{A}(k+1, k) & \bar{A}(k+1, k-1) & \dots & \dots & \bar{A}(k+1, k+1-M) \\ e & \varepsilon & \dots & \varepsilon & \varepsilon \\ \varepsilon & e & \ddots & \vdots & \varepsilon \\ \vdots & \ddots & e & \varepsilon & \varepsilon \\ \varepsilon & \dots & \varepsilon & e & \varepsilon \end{pmatrix},$$

where e and ε denote the $|Q'| \times |Q'|$ identity and zero matrices, respectively, and let $\tilde{v}(k)$ be the $(|Q'| \times M)$ -dimensional vector

$$\tilde{v}(k) \stackrel{\text{def}}{=} \begin{pmatrix} \bar{v}(k+1) \\ \varepsilon \\ \vdots \\ \varepsilon \end{pmatrix},$$

where ε represents here the $|Q'|$ -dimensional zero vector.

Adopting the convention that $x_j(k)$ and $v_j(k)$ are equal to ε for $k \leq 0$, it should be clear that Equation (2.31) in the following corollary is a mere rewriting of the evolution equations (2.29).

Corollary 2.74 *The extended state space vector $\tilde{x}(k)$ satisfies the $(M \times |Q'|)$ -dimensional recurrence relation of order 1*

$$\tilde{x}(k+1) = \tilde{A}(k)\tilde{x}(k) \oplus \tilde{v}(k), \quad k = 1, 2, \dots \quad (2.31)$$

Equation (2.31) will be referred to as the *standard form* of the evolution equations of an autonomous timed event graph satisfying (2.29).

Remark 2.75 In the particular case of a compatible initial condition, these equations read

$$\tilde{x}(k+1) = \tilde{A}(k)\tilde{x}(k), \quad k = 1, 2, \dots, \quad (2.32)$$

provided the continuation of $x_j(k)$ for $k \leq 0$, is that of Equation (2.24). Whenever the entrance times of the tokens of the initial marking are all equal to e (see Remark 2.63), it is easily checked from (2.27) that in this case

$$\tilde{x}_{l|Q'|+j}(0) = \begin{cases} e & \text{if } 0 \leq l < M_j; \\ \varepsilon & \text{for } l \geq M_j, \end{cases} \quad (2.33)$$

for $l = 0, \dots, M-1; j = 1, \dots, |Q'|$. ■

Example 2.76 (Example 2.60 continued) Here we have

$$\tilde{x}(k) = \begin{pmatrix} x_1(k) \\ x_4(k) \\ x_1(k-1) \\ x_4(k-1) \end{pmatrix}, \quad \tilde{v}(k) = \begin{pmatrix} \bar{v}_1(k+1) \\ \bar{v}_4(k+1) \\ \varepsilon \\ \varepsilon \end{pmatrix},$$

and

$$\tilde{A}(k) = \begin{pmatrix} \alpha_{11}(k+1) & \alpha_{14}(k+1) & \varepsilon & \varepsilon \\ \alpha_{42}(k+1) & \alpha_{44}(k+1) & \alpha_{43}(k+1) & \alpha_{31}(k+1) \\ e & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon \end{pmatrix}.$$

In the special case mentioned at the end of the preceding remark, we have $\tilde{x}(0) = (e, e, e, e)'$. ■

Remark 2.77 (Equivalent net with at most one token in the initial marking)

One can associate a timed event graph with the evolution equations (2.31). The interesting property of this event graph is that its initial marking is such that $M = 1$ (more precisely, each μ_i in this event graph is 1).

In view of Corollary 2.74, one can hence state that for any timed event graph, one can construct another 'equivalent' event graph with initial marking equal to 1 everywhere. The equivalence means here that one can find a bijective mapping from the set of transitions of the initial event graph to a subset of the transitions of the second one, such that two corresponding transitions fire at the same time. In particular, any observation of these transitions will be identical.

For instance, in the case of a compatible initial condition, this event graph can be obtained from the original one by first transforming it into an event graph with positive initial marking (as it was done in Remark 2.71), and by then applying the following transformation rules:

1. for each transition q_j of Q' in the original event graph, create M transitions $q_{jl}, l = 0, \dots, M-1$;
2. for each $q_{jl}, l = 0, \dots, M-2$, create a place that connects q_{jl} to $q_{j,l+1}$, with 0 holding times. Put one token in its initial marking, with initial lag time $z_j(-l)$;
3. for each place connecting $q_i \in Q'$ to q_j , and with $l+1$ initial tokens, $l \geq 0$, in the original system, create a place with one token with initial lag time $z_i(-l)$, and with the same holding times sequence as the original place. This new place has q_{il} as input transition, and q_{j0} as output transition.

For Example 2.60, the corresponding event graph is given in Figure 2.27. The behavior of q_{10} in this graph is the same as that of q_1 in Figure 2.24. The same property holds for q_{40} and q_4 respectively. ■

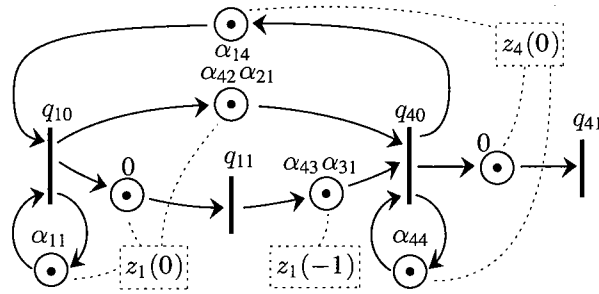


Figure 2.27: Illustration of Remark 2.77

2.5.5 The Nonautonomous Case

This subsection focuses on FIFO timed event graphs with external inputs. The firing times can be taken equal to 0, without loss of generality.

To the framework of the preceding sections, we add a new class of transitions called *input transitions*. This set of transitions will be denoted I .

Definition 2.78 (Input transition) An input transition consists of a source transition and of a nondecreasing real valued sequence, called the input sequence.

The input sequence associated with transition $q_j \in I$ will be denoted $u_j(k), k \geq 1$; the interpretation of this sequence is that $u_j(k), k \geq 1$, gives the epoch when q_j fires for the k -th time, due to some external trigger action. The input sequences are assumed to be given.

Definition 2.79 (Weakly compatible input sequence) The input sequence $u_j(k), k \in \mathbb{Z}$, is weakly compatible if $u_j(1) \geq 0$.

In what follows, all input sequences will be assumed to be weakly compatible. As in the autonomous case, the initial condition is said to be weakly compatible if in addition

- the entrance times of the tokens of the initial marking are nonpositive;
- the firings which consume tokens of the initial marking complete at nonnegative epochs (the assumption that the input sequences are weakly compatible may contribute ensuring that this holds).

The definition of compatibility is the same as in the autonomous case. For instance, if the lag times of a nonautonomous event graph are compatible, one can continue the input sequence $\{u_j(k)\}$ (with $j \in I$) to a nondecreasing sequence $\{u_j(k)\}_{k \in \mathbb{Z}}$, with $u_j(0) \leq 0$, such that for all $p_i \in \sigma(q_j)$ with $\mu_i \geq 1$,

$$w_i(k) = \alpha_i(k) \otimes u_j(k - \mu_i), \quad \forall k : 1 \leq k \leq \mu_i. \quad (2.34)$$

2.5.5.1 Basic Nonautonomous Equations

The derivation of the evolution equations is based on the same type of assumptions as in the autonomous case, namely the event graph is FIFO and the initial condition is weakly compatible.

Define the $|Q| \times |I|$ matrices $B(k, k), \dots, B(k, k-M)$ by

$$B_{jl}(k, k-m) \stackrel{\text{def}}{=} \bigoplus_{\{i \in \pi^q(j) | \pi^p(i)=l, \mu_i=m\}} \alpha_i(k), \quad (2.35)$$

and the $|I|$ -dimensional vector $u(k) = (u_1(k), \dots, u_{|I|}(k)), k = 1, 2, \dots$. Using the same arguments as in Theorem 2.58, we get the following result.

Theorem 2.80 Under the foregoing assumptions, the state vector $x(k) = (x_1(k), \dots, x_{|Q|}(k))'$ satisfies the evolution equations:

$$x(k) = A(k, k)x(k) \oplus \dots \oplus A(k, k-M)x(k-M) \oplus B(k, k)u(k) \oplus \dots \oplus B(k, k-M)u(k-M) \oplus v(k), \quad k = 1, 2, \dots, \quad (2.36)$$

where $x_j(k) \stackrel{\text{def}}{=} \varepsilon$ and $u_j(k) = \varepsilon$ for all $k \leq 0$; $v_j(k)$ is defined as in (2.16) for $1 \leq k \leq M$ and it is equal to ε otherwise.

If the initial lag times and the input sequences are both compatible, this equation can be simplified by using the same arguments as in Corollary 2.62, which leads to the equation

$$x(k) = A(k, k)x(k) \oplus \dots \oplus A(k, k-M)x(k-M) \oplus B(k, k)u(k) \oplus \dots \oplus B(k, k-M)u(k-M), \quad k = 1, 2, \dots, \quad (2.37)$$

where the continuations that are taken for $x(k)$ and $u(k), k \leq 0$, are now those defined in Corollary 2.62 and Equation 2.34, respectively.

in what follows, we will say that the nonautonomous event graph is live if the associated autonomous event graph (namely the one associated with the equation $x(k) = A(k, k)x(k) \oplus \dots \oplus A(k, k-M)x(k-M)$) is live. Let

$$\bar{B}(k, k-l) \stackrel{\text{def}}{=} A^*(k, k)B(k, k-l), \quad k \in \mathbb{Z}, \quad l = 0, \dots, M.$$

The following theorem is proved like Theorem 2.66.

Theorem 2.81 *If the event graph is live, the evolution equations (2.36) can be rewritten as*

$$\begin{aligned} x(k) &= \bar{A}(k, k-1)x(k-1) \oplus \dots \oplus \bar{A}(k, k-M)x(k-M) \\ &\quad \oplus \bar{B}(k, k)u(k) \oplus \dots \oplus \bar{B}(k, k-M)u(k-M) \oplus \bar{v}(k), \end{aligned} \quad (2.38)$$

$k = 1, 2, \dots,$

with the same simplification as in Corollary 2.62, provided the initial lag times and the input sequences are compatible.

The graph theoretic interpretation of $\bar{B}_{jj'}(k, k-l)$ is again the longest path in $S(j', j, l)$ (see Remark 2.69).

2.5.5.2 Standard Nonautonomous Equations

Define the $(M \times |I|)$ -dimensional vector

$$\tilde{u}(k) \stackrel{\text{def}}{=} \begin{pmatrix} u(k+1) \\ u(k) \\ \vdots \\ u(k+2-M) \end{pmatrix},$$

and the $(|I| \times M) \times (|Q'| \times M)$ matrix

$$\tilde{B}(k) = \begin{pmatrix} \bar{B}(k+1, k+1) & \bar{B}(k+1, k) & \dots & \bar{B}(k+1, k+2-M) \\ \varepsilon & \varepsilon & \dots & \varepsilon \\ \vdots & \vdots & \vdots & \vdots \\ \varepsilon & \varepsilon & \dots & \varepsilon \end{pmatrix}.$$

Corollary 2.82 (Standard nonautonomous equation) *The extended state space vector $\tilde{x}(k)$ satisfies the $(M \times |Q'|)$ -dimensional recurrence of order 1:*

$$\tilde{x}(k+1) = \tilde{A}(k)\tilde{x}(k) \oplus \tilde{B}(k)\tilde{u}(k) \oplus \tilde{v}(k), \quad k = 1, 2, \dots, \quad (2.39)$$

with the same simplification as in Corollary 2.62, provided the initial lag times and the input sequences are compatible.

Remark 2.83 Formally, the autonomous equation (2.17) can also be seen as that of a nonautonomous event graph with a set of input transitions $I = \{q'_1, q'_2, \dots\}$ of the same cardinality as the set Q' , with input sequence vector $v(k)$, and with $B(k, k) = e$, $B(k, k-l) = \varepsilon$ for $l = 1, \dots, M$ (the input transition q'_j is connected to the internal transition q_j by a single place with 0 initial marking).

However, the requirement that an input sequence should be nondecreasing contradicts our foregoing assumption on $v(k)$ (with our definitions, $v_j(k)$ eventually becomes ε for k large, as it can be seen from (2.16)). However, when using the fact that the sequences $x_j(k)$ are nondecreasing, it is easy to check that one can take the input sequence $u_j(k)$ defined by the function

$$u_j(k) \stackrel{\text{def}}{=} \begin{cases} v_j(k) & \text{if } 1 \leq k \leq M_j; \\ v_j(M_j) & \text{if } k \geq M_j, \end{cases}$$

instead of $v(k)$, without altering the values of $x(k)$.

This representation of an autonomous event graph as a nonautonomous one, where all initial lag times can be taken equal to ε , is exemplified on the event graph of Example 2.60 in Figure 2.28. ■

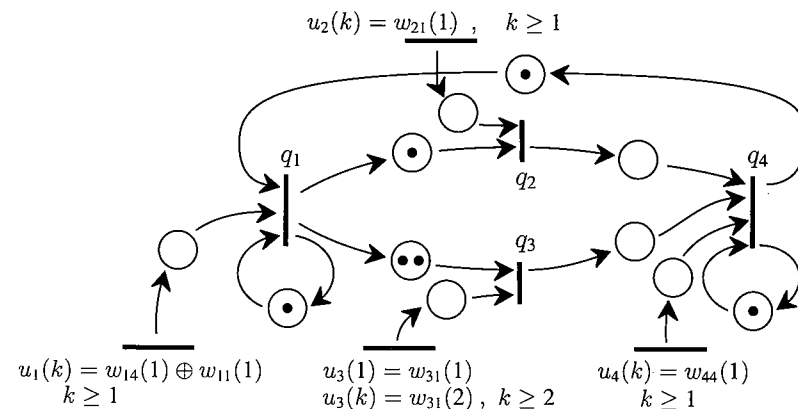


Figure 2.28: Illustration of Remark 2.83

2.5.6 Construction of the Marking

This subsection is concerned with the construction of the marking from the state variables. We will limit ourselves to the construction of the marking at certain epochs for which their expression is quite simple. However, the formulæ that are obtained are nonlinear in the max-plus algebra. We will therefore return to classical algebra, at least in this subsection. For the sake of simplicity, it is assumed that the initial condition is compatible. Pick some place p_i in $\mathcal{P}$, and let $q_j = \pi(p_i)$,

$q_l = \sigma(p_i)$. Let $N_i(t)$ be the number of tokens in place p_i at time $t, t \geq 0$, with the convention that this piecewise constant function is right-continuous.

Let

$$N_i^-(k) \stackrel{\text{def}}{=} N_i(x_{\pi^p(i)}(k)), \quad k \geq 1, \quad (2.40)$$

$$N_i^+(k) \stackrel{\text{def}}{=} N_i(x_{\sigma^p(i)}(k)), \quad k \geq 1. \quad (2.41)$$

Owing to our definitions, $N_i^-(k)$ is the number of tokens in p_i just after the k -th token entrance into p_i after $t = 0$, while $N_i^+(k)$ is the number of tokens p_i just after the departure of the k -th token to leave p_i after $t = 0$.

Lemma 2.84 *Under the foregoing assumptions,*

$$N_i^-(k) = \sum_{h=1}^{k+\mu_i} 1_{\{x_l(h) > x_j(k)\}}, \quad k = 1, 2, \dots, \quad (2.42)$$

$$N_i^+(k) = \sum_{h=k+1-\mu_i}^{\infty} 1_{\{x_l(h) \geq x_j(h)\}}, \quad k = 1, 2, \dots. \quad (2.43)$$

Proof The tokens present in p_i at time (just after) $x_j(k)$ are those that arrived into p_i no later than $x_j(k)$ and which are still in p_i at time $x_j(k)$. The tokens that arrived no later than $x_j(k)$ are those with index h with respect to this place, with $1 \leq h \leq k + \mu_i$. Among these tokens, those which satisfy the relation $x_l(h) > x_j(k)$ are still in p_i at time $x_j(k)$. Similarly, the only tokens that can be present in place p_i just after time $x_l(k)$ are those with index $h > k$ with respect to p_i . The token of index h with respect to p_i is the token produced by transition j at time $x_j(h - \mu_i)$ (where the continuation of $x_j(k)$ to $k \leq 0$ is the one defined in §2.5.2.6). Among these tokens, those which entered p_i no later than time $x_l(k)$ are in p_i at time $x_l(k)$. ■

2.5.7 Stochastic Event Graphs

Definition 2.85 (Stochastic event graph) *A timed event graph is a stochastic event graph if the holding times, the firing times and the lag times are all random variables defined on a common probability space.*

Different levels of generality can be considered. The most general situation that will be considered in Chapters 7 and 8 is the case when the sequences $\{\alpha_i(k)\}_{k \in \mathbb{Z}}$, $i = 1, \dots, |\mathcal{P}|$, and $\{\beta_j(k)\}_{k \in \mathbb{Z}}$, $j = 1, \dots, |\mathcal{Q}|$, are jointly stationary and ergodic sequences of nonnegative and integrable random variables defined on a common probability space $(\Omega, \mathbb{F}, \mathbb{P})$. Similarly, the lag times $w_i(k - \mu_i)$, $1 \leq k \leq \mu_i$, are assumed to be finite and integrable random variables defined on $(\Omega, \mathbb{F}, \mathbb{P})$.

More specific situations will also be considered, like for instance the case when the sequences $\{\alpha_l(k)\}_{k \in \mathbb{N}}$, $l = 1, \dots, |\mathcal{P}|$, and $\{\beta_i(k)\}_{k \in \mathbb{N}}$, $i = 1, \dots, |\mathcal{Q}|$, are mutually independent sequences of independent and identically distributed (i.i.d.)

random variables. For instance, all these variables could be exponentially distributed, with a parameter that depends on l or i , namely

$$\mathbb{P}[\alpha_l(k) \leq x] = \begin{cases} 1 - \exp(-a_l x) & \text{if } x \geq 0; \\ 0 & \text{otherwise,} \end{cases} \quad (2.44)$$

and

$$\mathbb{P}[\beta_i(k) \leq x] = \begin{cases} 1 - \exp(-b_i x) & \text{if } x \geq 0; \\ 0 & \text{otherwise,} \end{cases} \quad (2.45)$$

where $a_j > 0$ and $b_j > 0$. Another particular case arises when all the sequences are constant and deterministic, and the case with *constant timing* is thus a special (and degenerate) case of the i.i.d. situation.

2.6 MODELING ISSUES

In this section some issues related to the modeling of Petri nets will be described briefly.

2.6.1 Multigraphs

Multigraphs are graphs in which more than one arc between two nodes is allowed. The fact that in this chapter no such multigraphs have been considered is twofold.

The first reason is that the modeling power of Petri nets with multiple arcs and Petri nets with single arcs is the same [108]. This ‘modeling power’ is defined in terms of ‘reachability’, as discussed in §2.4.1. Petri nets with multiple arcs can straightforwardly be represented by Petri nets with single arcs, as shown in Figure 2.29. Note, however, that in the second of these figures a conflict situation

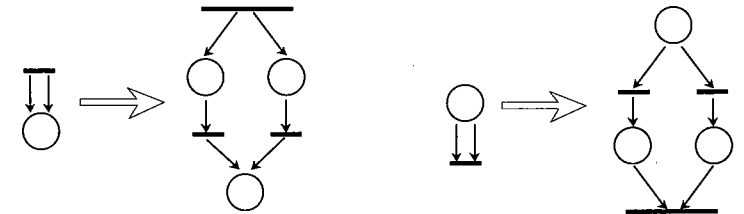


Figure 2.29: The change of multiple arcs into single arcs

has arisen. In order that this single arc representation of the originally multiple arc from place to transition behaves in the same way as this multiple arc, it is necessary that the two rival transitions receive tokens alternately.

The second reason is that it is not at all clear how to obtain equations for timed Petri nets in which there are multiple arcs. Observe that such nets are not event graphs, as is directly seen from Figure 2.29. It seems that some transitions fire in the long run twice as often as some other transitions; the $2k$ -th enabling of such a ‘fast’ transition is caused by the k -th firing (approximately) of such a ‘slow’ transition.

2.6.2 Places with Finite Capacity

For the rule for enabling transitions it has tacitly been assumed that each place can accommodate an unlimited number of tokens. For the modeling of many physical systems it is natural to consider an upper limit for the number of tokens that each place can hold. Such a Petri net is referred to as a finite capacity net. In such a net, each place has an associated capacity K_i , being the maximum number of tokens that p_i can hold at any time. For a transition in a finite capacity net to be enabled, there is the additional condition that the number of tokens in each $p_i \in \sigma(q_j)$ cannot exceed its capacity after the firing of q_j .

In the discussion to come we confine ourselves to event graphs, although the extension to Petri nets is quite straightforward, see [96]. Suppose that place p_i has a capacity constraint K_i , then the finite capacity net will be 'remodeled' as another event graph, without capacity constraints. If $\pi(p_i) \cap \sigma(p_i) \neq \emptyset$, then there is a loop. The number of tokens in a loop before and after the firing is the same and hence the capacity constraint is never violated (provided the initial number of tokens was admissible). Assume now that $\pi(p_i) \cap \sigma(p_i) = \emptyset$. Add another place p'_i to the net. This new place will have $\mu_{p'_i} = K_i - \mu_{p_i}$ tokens. Add an arc from $\sigma(p_i)$ to p'_i and an arc from p'_i to $\pi(p_i)$. The number of tokens in this new circuit is constant according to Theorem 2.37. The liveness of the event graph is not influenced by the addition of such a new circuit, see Theorem 2.38. An example is provided in Figure 2.30 where $K_i = 3$. It is easily verified that this newly constructed Petri net, without

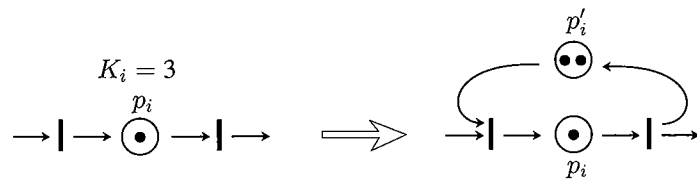


Figure 2.30: Node with and without a capacity constraint

capacity constraints, behaves in exactly the same way in terms of possible firing sequences as the original finite capacity net. In this sense the nets are 'equivalent'.

2.6.3 Synthesis of Event Graphs from Interacting Resources

This aim of this subsection is to show how, starting from the physical understanding of 'resources', and some assumptions, one can build up event graphs in a somewhat systematic way. This approach leads to a subclass of event graphs (essentially restricted by the kind of initial marking they can receive) for which the issue of initial conditions is transparent.

2.6.3.1 General Observations

The exercise of modeling is more an art than an activity which obeys rigid and precise rules. For example, the degree of details retained in a model is a matter

of appraisal with respect to future uses of the model. Models may be equivalent in some respects but they may differ in the physical insights they provide. These observations are classical in conventional system theory: it is well known that different state space realizations, even with different dimensions, may yield the same input-output behavior, but some of these realizations may capture a physical meaning of the state variables whereas others may not.

To be more specific, a clear identification of what corresponds to 'resources' in an abstract Petri net model may not be crucial if available resources are given once and for all and if the problem only consists in evaluating the performance of a specified system, but it may become a fundamental issue when resource quantities enter the decision variables, e.g. in optimal resource sizing at the design stage. It has already been noticed that, in an event graph, although the number of tokens in each circuit is invariant during the evolution of the system, this is not generally the case of the *total number of tokens* in the system, even if the graph is strongly connected and autonomous. In this case, it is unclear how tokens and physical resources are related to each other.

Figure 2.31 shows the simplest example of this type for which there are only two distinct situations for the distribution of tokens, the total number of tokens being either one or two. Indeed, if we redraw this event graph as in Figure 2.32, it becomes possible to interpret the two tokens as two resources circulating in the

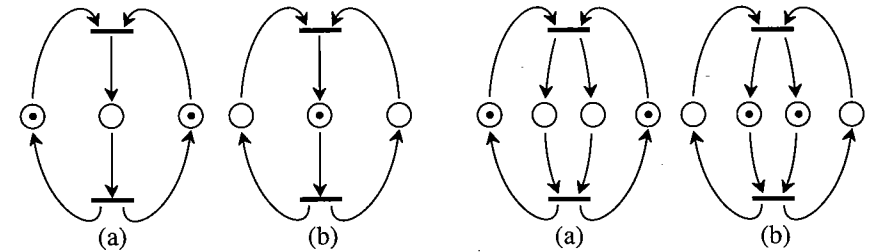


Figure 2.31: Merging two resources

Figure 2.32: An alternative model

system sometimes alone (case (a)) and sometimes jointly (case (b)). The problem with the former model of Figure 2.31 is that, when they stay together (case (b)), the two resources are represented by a single token. Obviously, these two models are equivalent if one is only interested in the number of transition firings within a certain time (assuming that some holding times have been defined properly in corresponding places), but the difference becomes relevant if one is willing to play with individual resource quantities in order to 'optimize' the design.

As long as modeling is an art, what is described hereafter should be considered as a set of practical guidelines—rather than as a rigorous theory—which should help in trying to construct models which capture as much physical meaning as possible. The dynamic systems we have in mind are supposed to involve the combined evolution of several interacting resources designed to achieve some specific overall task or service. Our approach is in three stages:

1. we first describe the evolution of each type of resource individually;
2. then we describe the mechanism of interaction between resources;
3. finally, we discuss the problem of initialization.

2.6.3.2 State Evolution of Individual Resources

The word 'resource' should be understood in the broadest sense: a machine, a tool, a part, a channel, a transportation link, a position in a storage or in a buffer, etc. are all resources. Of course, in practice, it is worthwhile modeling a resource explicitly as long as it is a 'scarce' resource, that is a resource the limited availability of which may have some influence on the evolution of the system at some time. The capacity of a buffer may be large enough for the buffer to behave as if it were infinite. Again, it is a matter of feeling to decide whether an explicit model of the finite capacity is *a priori* needed or not.

The evolution of each resource in the system is modeled by a sequence of 'stages'. What is considered a 'stage' is a matter of judgment since several *consecutive* stages may be aggregated into a single stage. For example, considering the evolution of a part in a workshop, traveling to the next machine and then waiting for the temperature to be cool enough before entering this next machine may be considered a single stage. We assume that

- the nature and the order of the stages experienced by each type of resource are known in advance;
- for a given resource, every stage is preceded (followed)—if it is not the first (last) stage—by a *single* other stage.

Consequently, for a given type of resource, if we draw a graph such that nodes (drawn as places in a Petri net) represent stages, and arcs indicate precedence of the upstream stages over the downstream stages, then this graph is a path (i.e. it has neither circuits nor branches) and it reflects the total order of stages.

Obviously, in a workshop, this assumes that a 'deterministic scheduling' of operations has been defined *a priori*. For example, a machine is supposed to work on a specified sequence of parts in a specified order, each operation being represented by a particular stage. If the stage 'idle' is possible between two such operations (i.e. the machine may deliver the part it holds without receiving its next part—hence it stays alone for a while), then additional stages should be introduced in between. Notice that

- the stage 'idle' is *not* represented by a *unique* node, but that a node of this type is introduced if necessary between any two successive operations;
- a machine may be idle, while still holding the part it was working on, because there is no storage available downstream, and the next operation of this machine cannot be undertaken before the downstream machine can absorb the present part. In this case, it is not necessary to introduce an additional stage, that is, the stage 'working' and the stage 'waiting for downstream delivery' need not be distinguished.

For a part, it is also assumed that the sequence of machines visited is specified in advance, but notice that storages visited should be considered as an operation of the same type as machines visited. Therefore, a storage position, if it is common to several types of parts, must know in advance the order in which it will be visited by these different parts, which is certainly a restrictive constraint in the modeling phase, but this is the price to pay for remaining eventually in the class of event graphs.

Finally, the evolution of a resource is simply a path (in other words, this evolution is represented by a serial automaton). For convenience, each arc will be 'cut' by a bar resembling a transition, which will serve later on for synchronization purposes. Each stage receives a sequence of holding times representing *minimal* times spent in this stage by the successive resources. For example, waiting in a storage should involve a minimal time equal to 0. At any time, the present stage of a given resource is represented by a token marking the corresponding place. Transitions represent changes of stages and they are instantaneous.

Resources may enter the system and leave it after a while (nonreusable resources) or they may revisit the same stages indefinitely because they are 'recycled' (reusable resources). For example, raw materials come in a workshop and leave after a transformation, whereas machines may indefinitely resume their work on the same repetitive sequences of parts. Sometimes, nonreusable resources are tightly associated with reusable resources so that it is only important to model these reusable resources: for example, parts may be fixed on pallets that are recycled after the parts leave the workshop. For reusable resources, we introduce an additional stage called the 'recycling stage', and we put an arc from the last stage (of an elementary sequence) to this recycling stage and another arc from the recycling stage to the first stage. Hence we obtain a circuit. Physically, the recycling stage might represent a certain 'reconditioning' operation (possibly involving other resources too), and therefore it might receive a nonzero holding time (transportation time, set-up time, etc.). However, it will be preferable to suppose that the recycling stage of any resource corresponds to an abstract operation which involves only this resource, and which is immediate (holding time 0). The positioning of this recycling stage with respect to the true reconditioning operation (before or after it) is left to the appraisal of the user in each specific situation. Indeed, each stage along the circuit of the reusable resource where this resource stands alone may be a candidate to play the role of the recycling stage, a remark which should be kept in mind when we speak of canonical initialization later on (this is related to the issue of the point at which one starts a periodic behavior).

2.6.3.3 Synchronization Mechanism

Any elementary operation may involve only one particular type of resource, or it may also involve several different resources. For example, a part waiting in a storage involves both the part and the storage position (whereas an idle machine involves only this machine). It should be realized that, so far, the same physical operation, as long as it involves n different resources simultaneously, has been represented by n different places.

If two stages belonging to two distinct resource paths (or circuits) correspond to the same operation, we must express that these stages are entered and left simultaneously by the two resources (moreover, the holding times of these stages should be the same). This may be achieved by putting two synchronization circuits, one connecting the 'beginning' transitions, the other connecting the 'end' transitions, as indicated in Figure 2.33. In order to comply with the standard event graph representation, we have put new places—represented here in grey color—over the arcs of these circuits. However, these grey places do not represent 'stages' as other places do. They are never marked with tokens and they have holding times equal to 0. Then, it is realized that these circuits involving no tokens and having a total holding time equal to 0 express simultaneity of events, that is entering or leaving the considered stage by anyone of the two resources precedes the same type of event achieved by the other resource and vice versa.

Remark 2.86 What we have just done here is nonstandard from the point of view of Petri net theory, although it is mathematically correct for the purpose of expressing simultaneity. Indeed, having a *live* Petri net which includes circuits with no tokens seems to contradict Theorem 2.38. More specifically, every transition having a 'grey place' upstream (which will never receive tokens) will never fire if we stick to the general rule about how transition firings are enabled. We propose the following (tricky) adaptation of this rule to get out of this contradiction: '*in a timed Petri net, a transition may "borrow" tokens to enable its own firing during a duration of 0 time units, that is, provided that it can "return" the same amount of tokens immediately*'. This condition is satisfied for transitions preceded and followed by (the same number of) 'grey places' since tokens may be 'borrowed' in upstream 'grey places' only at the epoch of firing (since those tokens are then immediately available—holding time 0), and, for the same reason, tokens produced in downstream 'grey places' can immediately be 'returned' after firing (they are immediately available when produced).

Mathematically, consider the pair of transitions at the left-hand side of Figure 2.33 and let $x_1(k)$ and $x_2(k)$ denote their respective daters (see Definition 2.57). The 'grey circuit' translates into the following two inequalities:

$$x_1(k) \geq x_2(k) \quad \text{and} \quad x_2(k) \geq x_1(k),$$

which imply equality: this is exactly what we want and everything is consistent.

On the contrary, the reader should think of what happens if one of the holding times put on 'grey places' is strictly positive and there are still no tokens in the initial marking of these places. ■

To avoid this discussion, two alternative solutions can be adopted. The first one consists in merging the simultaneous transitions as shown by Figure 2.34, which removes the synchronization circuits and the 'grey places'. We then come up with a representation similar to that of Figure 2.32. A further step towards simplification would be to merge the places and arcs in between so as to arrive at a representation similar to that of Figure 2.31, but then the resource interpretation would be obscured.

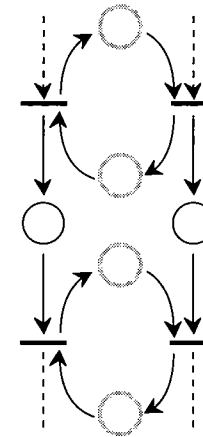


Figure 2.33: Synchronization mechanism

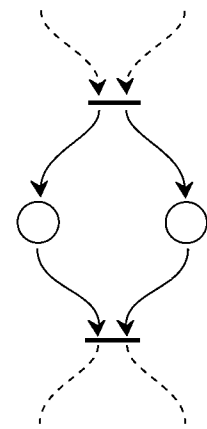


Figure 2.34: Alternative representation of synchronization

The second solution involves the introduction of fake transitions x'_1 and x'_2 upstream the real transitions to be synchronized. This mechanism is explained by Figure 2.35 with the equations proving that the firing times of x_1 and x_2 (denoted after the name of the transitions) are equal.

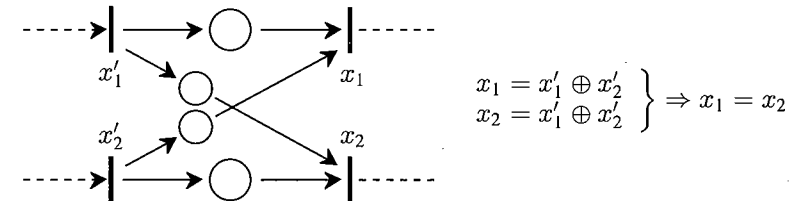


Figure 2.35: Another synchronization mechanism

Notice that the holding times of two synchronized stages, which were equal by construction, can now be different—e.g. one of the two may be taken to be 0—since only the greater holding time is important. All the above considerations extend to the case of n , rather than 2, resources simultaneously. It is realized that, with this approach, every transition has as many output arcs as it has input arcs. A problem arises with resources which are physically merged into a single product (assembly) or split up into two or more products (disassembly). That is, the end nodes of a resource path do not correspond to inlet or outlet nodes of the system. There is a choice of considering that a cluster of two resources that enter the system separately is still made of two resources traveling together (which amounts to doubling all arcs and places after the assembly operation) or to accept that the number of output arcs

at some transition (representing the beginning of an assembly operation) be less than the number of input arcs (which means that a new type of resource starts its life at that point of the system). Dual considerations apply to the problem of disassembly.

2.6.3.4 Initialization

As already indicated, during the evolution of the system the present stage of each resource will be marked on the graph by a token in a certain place along the path or the circuit of that resource. Two resources of the same type, e.g. two machines performing exactly the same sequence of operations, may use the same path or circuit as long as they never need to be distinguished. For example, a storage with n positions, and which is dedicated to a single type of stored resource, will be represented by a circuit with two stages, 'occupied' and 'available' (the only distinguishable stages of a single position), and with n tokens the distribution of which in the circuit indicates how many positions are available at any time. For a storage accommodating several types of resources, we refer the reader to the example at the end of this section.

Epochs at which resources move from one stage to the next one will be given by the data attached to the transition in between. We now define a canonical initial condition. For reusable resources, it corresponds to all tokens put at the corresponding recycling stages. As discussed earlier, these recycling stages are supposed to involve a single type of resource each, and a holding time equal to 0 (therefore, it is irrelevant to know when the tokens had been put there). For any nonreusable resource, since it passes through the system, we first complete its path by adding an inlet transition (upstream from the first place) and an outlet transition (downstream from the last place) so as to attach the epochs of inputs and outputs to these transitions (unless one of these transitions is already represented because the resource participates in an assembly operation, or because it is issued from a disassembly operation). The canonical initial condition for nonreusable resources corresponds to their paths being *empty*: all tokens must be introduced at the inlet transitions *after* the origin of time. Observe that the canonical initial condition is compatible in the sense of Definition 2.61.

From this given canonical initial condition, and given a sequence of epochs at all input transitions at which tokens are introduced into the system (see Definition 2.78), tokens will evolve within the system (whereas other tokens will leave) according to the general rules of timed event graphs, and they will reach some positions at some other given epoch. Obviously, all situations thus obtained, which we may call 'reachable conditions', are also acceptable 'initial conditions' by changing the origin of time to the present time. Such a candidate to play the role of an initial condition obviously fulfills some constraints (see hereafter), and it is not defined only by the positions of tokens (called the 'initial marking'): the time already spent by each token of the initial marking in this position before the (new) origin of time is also part of the definition of the 'initial condition' (at least for places where the holding time is nonzero). Alternatively, the first epoch at which each token of the initial marking may be consumed must be given: this corresponds to the notion of lag time introduced in Definition 2.49. Necessary, but maybe not sufficient, requirements of

reachability from the canonical initial condition can be stated for all places included between any pair of synchronized (Figure 2.33) or merged (Figure 2.34) transitions. Say there are n such places, then the same number of tokens must mark all these places, say p tokens per place, and there must exist p n -tuples of tokens (with one token per place) with the same difference between their holding times and their lag times. This assumption that there must exist exactly the same number p of tokens in each of the n places makes this notion of an 'initial condition reachable from the canonical initial condition' even more restrictive than the notion of a 'compatible initial condition' of Definition 2.61.

For example, if we return to the graph of Figure 2.32, with the interpretation of two reusable resources, position (a) corresponds to the canonical initial condition, position (b) is another acceptable initial marking. Suppose now that we add an additional exemplary of the resource represented by the left-hand circuit. Figure 2.36a represents an initial marking which we can interpret, but Figure 2.36b does not, although it is perfectly correct from the abstract point of view of event graphs. Moreover, it may be noted that the graph of Figure 2.36b is not reducible

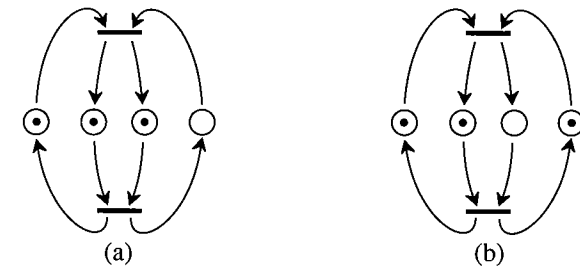


Figure 2.36: Two plus one resources

to something similar to Figure 2.31.

2.6.3.5 An Example

We consider two types of parts, say ♠ and ♥, which can wait in separate buffers of large capacities before being heated individually by a furnace. The furnace deals with parts ♠ and ♥ alternately. The parts can then wait in a stove having the room for three parts; each one of either type, the purpose being to maintain the temperature of the parts until they are assembled in pairs (♠, ♥) by a machine. Finally, parts leave the workshop. A part must stay in the furnace if it cannot enter the stove. In the same way, parts can leave the stove in pairs (♠, ♥) only when the assembly machine can handle them.

Figure 2.37 represents the stage sequences of parts ♠ and ♥ as vertical paths on the left-hand and right-hand sides, respectively. In the middle, from top to bottom, the three circuits represent the stage sequences of the furnace, of the stove and of the machine. Each stage is labeled by a letter and Table 2.1 gives the interpretation of these stages together with their holding times.

Table 2.1: An example: stage interpretation

Stage label	Stage interpretation	Holding time
A	Part ♠ waits in buffer	0
B	Part ♠ stays in furnace	α_1
C	Part ♠ waits in stove	0
D	Part ♠ is assembled with part ♥	α_2
E	Part ♥ waits in buffer	0
F	Part ♥ stays in furnace	α_3
G	Part ♥ waits in stove	0
H	Part ♥ is assembled with part ♠	α_2
I	Furnace waits for part of type ♠	0
J	Furnace holds part of type ♠	α_1
K	Furnace waits for part of type ♥	0
L	Furnace holds part of type ♥	α_3
M	One position in stove waits for part of type ♠	0
N	One position in stove holds part of type ♠	0
O	One position in stove waits for part of type ♥	0
P	One position in stove holds part of type ♥	0
Q	Machine assembles a pair (♠, ♥)	α_2
R	Machine waits for a pair (♠, ♥)	0

The transitions which are synchronized are connected by a dotted line. The initial marking assumes that the furnace starts with a part ♠ and that two positions out of the three in the stove accept a part ♠ first.

Figure 2.38 shows the reduced event graph obtained by merging the synchronized transitions and the stages representing the same operation. However, we keep multiple labels when appropriate so that the order of multiplicity of places between synchronized transitions is still apparent. Single labels on places in circuits indicate possible recycling stages. Observe that, for any transition, the total number of labels of places upstream balances the total number of labels of places downstream. Indeed, the Petri net of Figure 2.37 (with transitions connected by dotted lines merged together) is strictly conservative (Definition 2.35) whereas that of Figure 2.38 is conservative (Definition 2.36) with weights (used in that definition) indicated by the number of labels.

2.7 NOTES

Graph theory is a standard course in many mathematical curricula. The terminology varies, however. Here we followed the terminology as used in [67] as much as possible. Karp's algorithm, the subject of §2.3.2, was first published in [73]. Since the 1970s, a lot of attention has been paid to the theory of Petri nets. This theory has the potential of being suitable as an excellent modeling aid in many fields of application. Its modeling power is larger than the one

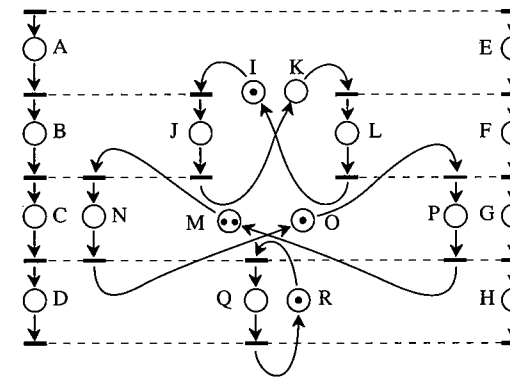


Figure 2.37: Expanded model

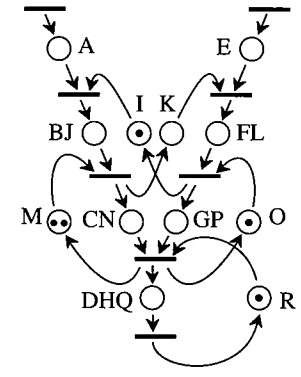


Figure 2.38: Reduced model

of automata. If one were to allow inhibitor arcs, see [1], then the modeling power is the same as that of Turing machines. A 'system theory' including maximization, addition as well as negation (which is what an inhibitor arc represents), has not yet been developed. An excellent overview of the theory of Petri nets is given in [96], where also many other references can be found. The section on timed Petri nets, however, is rather brief in [96]. Some material on cycle times can be found in [47] and [115], which is also discussed in Chapter 9. A good, though somewhat dated, introduction to Petri nets is [108]. Other sources which contain a lot of material on Petri nets are [30] and [29]. For a recent discussion on modeling power related to Petri nets, see [82]. Section 2.5 on equations for timed Petri nets is mainly based on [39] for the constant timing case and on [11] for the general case. Equivalence of systems represented by different graphs is also discussed in [84]. This reference, however, deals with systolic systems, in which there is a fixed clock frequency. Some of the results obtained there also seem plausible within the timed event graph setting. Relationships between graphs and binary dynamic systems are described in [25].

In [92] a novel scheme, called *kanban*, is described and analyzed for the coordination of tokens in an event graph. The essence is that the introduction of a new circuit with enough tokens regulates the 'speed' of the original event graph and it controls the number of tokens in various places.

For a recent development in continuous Petri nets, where the the number of tokens is real rather than integer valued, see [53] and [99]. In the latter reference some results given in Chapter 3 for 'discrete' Petri nets are directly extended to continuous ones.

PART II

Algebra

CHAPTER 3

Max-Plus Algebra

3.1 INTRODUCTION

In this chapter we systematically revisit the classical algebraic structures used in conventional calculus and we substitute the idempotent semifield $\mathbb{R}_{\max}$ (the set of real numbers endowed with the operations max and plus) for the field of scalars. The purpose is to provide the mathematical tools needed to study linear dynamical systems in $(\mathbb{R}_{\max})^n$.

This chapter is divided in three main parts. In the first part we study linear systems of equations and polynomial functions in $\mathbb{R}_{\max}$. In the second part we consider a more advanced topic which can be skipped in a first reading. This topic is the problem of the linear closure of $\mathbb{R}_{\max}$ and its consequences for solving systems of linear and polynomial equations (the linear closure is the extension of a set in such a way that any nondegenerated linear equation has one and only one solution). The third part is concerned with a max-plus extension of the Perron-Frobenius theory. It gives conditions under which event graphs reach a periodic behavior and it characterizes their periodicities. It can be seen as a more advanced topic of the spectral theory of max-plus matrices given in the first part of this chapter.

We first introduce the algebraic structure $\mathbb{R}_{\max}$ and we study its basic properties.

3.1.1 Definitions

Definition 3.1 (Semifield) A semifield $\mathcal{K}$ is a set endowed with two operations $\oplus$ and $\otimes$ such that:

- the operation $\oplus$ is associative, commutative and has a zero element ε ;
- the operation $\otimes$ defines a group on $\mathcal{K}_* \stackrel{\text{def}}{=} \mathcal{K} \setminus \{\varepsilon\}$, it is distributive with respect to $\oplus$ and its identity element e satisfies $\varepsilon \otimes e = e \otimes \varepsilon = \varepsilon$.

We say that the semifield is

- idempotent if the first operation is idempotent, that is, if $a \oplus a = a, \forall a \in \mathcal{K}$;
- commutative if the group is commutative.

Theorem 3.2 The zero element ε of an idempotent semifield is absorbing for the second operation, that is $\varepsilon \otimes a = a \otimes \varepsilon = \varepsilon, \forall a \in \mathcal{K}$.

Proof We have that

$$\varepsilon = \varepsilon e = \varepsilon(\varepsilon \oplus e) = \varepsilon^2 \oplus \varepsilon = \varepsilon^2,$$

and then,

$$\forall a \in \mathcal{K}_* \quad \varepsilon = \varepsilon e = \varepsilon a^{-1} a = \varepsilon(a^{-1} \oplus \varepsilon) a = \varepsilon a^{-1} a \oplus \varepsilon^2 a = \varepsilon^2 a = \varepsilon a.$$

■

Definition 3.3 (The algebraic structure $\mathbb{R}_{\max}$) The symbol $\mathbb{R}_{\max}$ denotes the set $\mathbb{R} \cup \{-\infty\}$ with $\max$ and $+$ as the two binary operations $\oplus$ and $\otimes$, respectively.

We call this structure the max-plus algebra. Sometimes this is also called an ordered group. We remark that the natural order on $\mathbb{R}_{\max}$ may be defined using the $\oplus$ operation

$$a \leq b \quad \text{if} \quad a \oplus b = b.$$

Definition 3.4 (The algebraic structure $\overline{\mathbb{R}}_{\max}$) The set $\mathbb{R} \cup \{-\infty\} \cup \{+\infty\}$ endowed with the operations $\max$ and $+$ as $\oplus$ and $\otimes$ and with the convention that $(-\infty) + \infty = -\infty$ is denoted $\overline{\mathbb{R}}_{\max}$. The element $+\infty$ is denoted $\top$.

Theorem 3.5 The algebraic structure $\mathbb{R}_{\max}$ is an idempotent commutative semifield.

The proof is straightforward. If we compare the properties of $\oplus$ and $\otimes$ with those of $+$ and $\times$, we see that:

- we have lost the symmetry of addition (for a given a , an element b does not exist such that $\max(b, a) = -\infty$ whenever $a \neq -\infty$);
- we have gained the idempotency of addition;
- there are no zero divisors in $\mathbb{R}_{\max}$ ($a \oplus b = -\infty \Rightarrow a = -\infty$ or $b = -\infty$).

If we try to make some algebraic calculations in this structure, we soon realize that idempotency is as useful as the existence of a symmetric element in the simplification of formulæ. For example, the analogue of the binomial formula

$$(a + b)^n = \binom{n}{0} a^n + \binom{n}{1} a^{n-1} b + \cdots + \binom{n}{n-1} a b^{n-1} + \binom{n}{0} b^n$$

is $n \max(a, b) = \max(na, nb)$, which is much simpler. On the other hand, we now face the difficulty that the $\max$ operation is no longer cancellative, e.g. $\max(a, b) = b$ does not imply that $a = -\infty$.

3.1.2 Notation

First of all, to emphasize the analogy with conventional calculus, $\max$ has been denoted $\oplus$, and $+$ has been denoted $\otimes$. We also introduce the symbol $\not\circ$ for the conventional $-$ (the inverse operation of $+$ which plays the role of multiplication, that is, the 'division'). Hence $a \not\circ b$ means $a - b$. Another notation for $a \not\circ b$ is the two-dimensional display notation

$$\frac{a}{b}.$$

We will omit the sign $\otimes$ if this does not lead to confusion. To prevent mistakes, we use ε and e for the 'zero' and the 'one', that is, the neutral elements of $\oplus$ and $\otimes$, respectively, namely $-\infty$ and 0 . To get the reader acquainted with this new notation, we propose the following table.

$\mathbb{R}_{\max}$ notation	Conventional notation	=
$2 \oplus 3$	$\max(2, 3)$	3
$1 \oplus 2 \oplus 3 \oplus 4 \oplus 5$	$\max(1, 2, 3, 4, 5)$	5
$2 \otimes 3 = 5$	$2 + 3$	5
$2 \oplus \varepsilon$	$\max(2, -\infty)$	2
$\varepsilon = \varepsilon \otimes 2$	$-\infty + 2$	$-\infty$
$(-1) \otimes 3$	$-1 + 3$	2
$e \otimes 3$	$0 + 3$	3
$3^2 = 2^3 = 3 \otimes 3 = 2 \otimes 2 \otimes 2$	$3 \times 2 = 2 \times 3 = 3 + 3 = 2 + 2 + 2$	6
$e = e^2 = 2^0$	$0 \times 2 = 2 \times 0$	0
$(2 \otimes 3) \not\circ (2 \oplus 3)$	$(2 + 3) - \max(2, 3)$	2
$(2 \oplus 3)^3 = 2^3 \oplus 3^3$	$3 \times \max(2, 3) = \max(3 \times 2, 3 \times 3)$	9
$6 \not\circ e$	$6 - 0$	6
$e \not\circ 3$	$0 - 3$	-3
$\sqrt[3]{8}$	$8/2$	4
$\sqrt[5]{15}$	$15/5$	3

There is no distinction, hence there are risks of confusion, between the two systems of notation as far as the power operation is concerned. As a general rule, a formula is written in one system of notation. Therefore if, in a formula, an operator of the max-plus algebra appears explicitly, then usually all the operators of this formula are max-plus operators.

3.1.3 The min Operation in the Max-Plus Algebra

It is possible to derive the min operation from the two operations $\otimes$ and $\oplus$ as follows:

$$\min(a, b) = \frac{ab}{a \oplus b}.$$

Let us now prove the classical properties of the min by pure rational calculations in the max-plus algebra.

- $-\min(a, b) = \max(-a, -b)$:

$$\frac{\frac{e}{ab}}{\frac{e}{a \oplus b}} = \frac{\frac{a \oplus b}{ab}}{\frac{e}{ab}} = \frac{\frac{e}{b}}{\frac{e}{a}} ;$$

- $-\max(a, b) = \min(-a, -b)$:

$$\frac{e}{a \oplus b} = \frac{\frac{e}{ab}}{\frac{e}{a \oplus b}} = \frac{\frac{e}{a} \otimes \frac{e}{b}}{\frac{e}{a} \oplus \frac{e}{b}} ;$$

- $\min(a, \min(b, c)) = \min(\min(a, b), c)$:

$$\frac{a \frac{bc}{b \oplus c}}{a \oplus \frac{bc}{b \oplus c}} = \frac{abc}{ab \oplus ac \oplus bc}$$

and the symmetry of the formula with respect to a, b and c proves the result;

- $\max(c, \min(a, b)) = \min(\max(c, a), \max(c, b))$:

$$\left\{ c \oplus \frac{ab}{a \oplus b} = \frac{(c \oplus a)(c \oplus b)}{(c \oplus a) \oplus (c \oplus b)} \right\} \Leftrightarrow \left\{ \frac{ca \oplus cb \oplus ab}{a \oplus b} = \frac{(c \oplus a)(c \oplus b)}{a \oplus b \oplus c} \right\}$$

$$\Leftrightarrow \{(ca \oplus cb \oplus ab)(a \oplus b \oplus c) = (c \oplus a)(c \oplus b)(a \oplus b)\} .$$

To check the last identity, we consider the expressions in both sides as polynomials in c and we first remark that the coefficient of c^2 , namely $a \oplus b$, is the same in both sides. The coefficient of c^0 , namely $ab(a \oplus b)$, also is the same in both sides. Now, considering the coefficient of c , it is equal to $(a \oplus b)^2 \oplus ab$ in the left-hand side, and to $(a \oplus b)^2$ in the right-hand side: these two expressions are clearly always equal.

- $\min(c, \max(a, b)) = \max(\min(c, a), \min(c, b))$:

$$\left\{ \frac{c(a \oplus b)}{c \oplus a \oplus b} = \frac{ca}{c \oplus a} \oplus \frac{cb}{c \oplus b} \right\} \Leftrightarrow \left\{ \frac{a \oplus b}{c \oplus a \oplus b} = \frac{a}{c \oplus a} \oplus \frac{b}{c \oplus b} \right\} .$$

The latter identity is amenable to the same verification as earlier.

3.2 MATRICES IN $\mathbb{R}_{\max}$

In this section we are mainly concerned with systems of linear equations. There are two kinds of linear systems in $\mathbb{R}_{\max}$ for which we are able to compute solutions: $x = Ax \oplus b$ and $Ax = b$ (the general system being $Ax \oplus b = Cx \oplus d$). We also study the spectral theory of matrices. There exist good notions of eigenvalue and eigenvector but there is often only one eigenvalue: this occurs when the precedence graph associated with the matrix is strongly connected (see Theorem 2.14).

3.2.1 Linear and Affine Scalar Functions

Definition 3.6 (Linear function) The function $f : \mathbb{R}_{\max} \rightarrow \mathbb{R}_{\max}$ is linear if it satisfies

$$f(c) = c \otimes f(e) , \quad \forall c \in \mathbb{R}_{\max} .$$

Thus any linear function is of the form $y = f(c) = a \otimes c$, where $a = f(e)$. The graph of such a function consists of a straight line with slope equal to one and which intersects the y -axis at a (see Figure 3.1).

Definition 3.7 (Affine function) The function $f : \mathbb{R}_{\max} \rightarrow \mathbb{R}_{\max}$, $f(c) = ac \oplus b$, $a \in \mathbb{R}_{\max}$, $b \in \mathbb{R}_{\max}$ is called affine.

Observe that, as usual, $b = f(\varepsilon)$ and $a = \lim_{c \rightarrow \infty} f(c)/c$, but here the limit is reached for a finite value of x (see Figure 3.2).

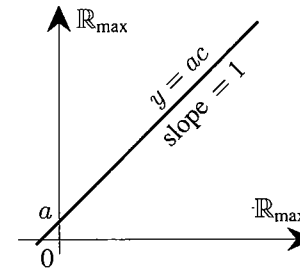


Figure 3.1: A linear function

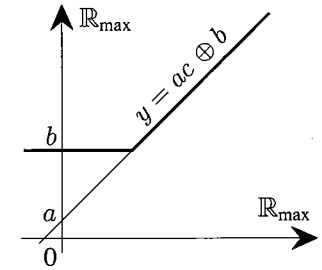


Figure 3.2: An affine function

In the primary school, the first algebraic problem that we have to solve is to find the solution of a scalar affine equation.

Definition 3.8 (Affine equation) The general scalar affine equation is

$$ax \oplus b = a'x \oplus b' . \quad (3.1)$$

Indeed, since $\oplus$ has no inverse, Equation (3.1) cannot be reduced to the usual form $ax \oplus b = \varepsilon$, which motivates the definition above.

Theorem 3.9 The solution of the general scalar affine equation is obtained as follows:

- if $((a' < a) \text{ and } (b < b')) \text{ or } ((a < a') \text{ and } (b' < b))$ (3.2) hold true, then the solution is unique and it is given by $x = (b \oplus b') \not\# (a \oplus a')$;
- if $a \neq a'$, $b \neq b'$, and (3.2) does not hold, no solutions exist in $\mathbb{R}_{\max}$;

- if $a = a'$ and $b \neq b'$, the solution is nonunique and all solutions are given by $x \geq (b \oplus b') \not\leq a$;
- if $a \neq a'$ and $b = b'$, the solution is nonunique and all solutions are given by $x \leq b \not\leq (a \oplus a')$;
- if $a = a'$ and $b = b'$, all $x \in \mathbb{R}$ are solutions.

The proof is straightforward from the geometric interpretation of affine equations as depicted in Figure 3.3.

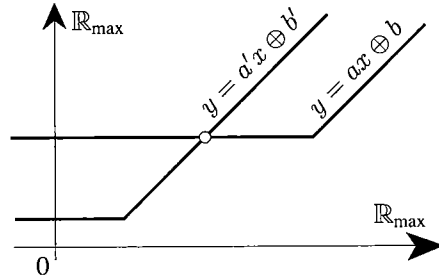


Figure 3.3: An affine equation

In practice, it is better to simplify (3.1) before solving it. For example, if $a > a'$ and $b' > b$, then $ax \oplus b = a'x \oplus b' \Leftrightarrow ax = b'$. Let us give all different kinds of simplified equations that may come up.

Definition 3.10 (Canonical form of an affine equation) An affine equation is in canonical form if it is in one of the simplified forms:

- $ax = b$;
- $ax \oplus b = \varepsilon$;
- $ax \oplus b = ax$;
- $ax \oplus b = b$.

3.2.2 Structures

The moduloid structure is a kind of module structure which, in turn, is a kind of vector space, that is, a set of vectors with an internal operation and an external operation defined over an idempotent semifield.

Definition 3.11 (Moduloid) A moduloid $\mathcal{M}$ over an idempotent semifield $\mathcal{K}$ (with operations $\oplus$ and $\otimes$, zero element ε and identity element e) is a set endowed with

- an internal operation also denoted $\oplus$ with a zero element also denoted ε ;

- an external operation defined on $\mathcal{K} \times \mathcal{M}$ with values in $\mathcal{M}$ indicated by the simple juxtaposition of the scalar and vector symbols;

which satisfies the following properties:

- $\oplus$ is associative, commutative;
- $\alpha(x \oplus y) = \alpha x \oplus \alpha y$;
- $(\alpha \oplus \beta)x = \alpha x \oplus \beta x$;
- $\alpha(\beta x) = (\alpha\beta)x$;
- $ex = x$;
- $\varepsilon x = \varepsilon$;

for all $\alpha, \beta \in \mathcal{K}$ and all $x, y \in \mathcal{M}$.

We will only be concerned with some special cases of such a structure.

Example 3.12 $(\mathbb{R}_{\max})^n$ is a moduloid over $\mathbb{R}_{\max}$. Its zero element is $(\varepsilon, \dots, \varepsilon)'$. ■

Other examples of moduloids will be presented later on.

Definition 3.13 (Idempotent algebra) A moduloid with an additional internal operation also denoted $\otimes$ is called an idempotent algebra if $\otimes$ is associative, if it has an identity element also denoted e , and if it is distributive with respect to $\oplus$.

This idempotent algebra is the main structure in which the forthcoming system theory is going to be developed.

Example 3.14 Let $(\mathbb{R}_{\max})^{n \times n}$ be the set of $n \times n$ matrices with coefficients in $\mathbb{R}_{\max}$ endowed with the following two internal operations:

- the componentwise addition denoted $\oplus$;
- the matrix multiplication already used in Chapters 1 and 2 denoted $\otimes$:

$$(A \otimes B)_{ij} = \bigoplus_{k=1}^n A_{ik} \otimes B_{kj} ;$$

and the external operation:

- $\forall \alpha \in \mathbb{R}_{\max}, \forall A \in (\mathbb{R}_{\max})^{n \times n}, \alpha A = (\alpha A_{ij})$.

The set $(\mathbb{R}_{\max})^{n \times n}$ is an idempotent algebra with

- the zero matrix, again denoted ε , which has all its entries equal to ε ;
- the identity matrix, again denoted e , which has the diagonal entries equal to e and the other entries equal to ε . ■

3.2.3 Systems of Linear Equations in $(\mathbb{R}_{\max})^n$

In this subsection we are mainly interested in systems of linear equations. To define such systems, we use matrix notation. A linear mapping $(\mathbb{R}_{\max})^n \rightarrow (\mathbb{R}_{\max})^n$ will be represented by a matrix as it was done in the previous chapters. With the max-plus algebra, the general system of equations is

$$Ax \oplus b = Cx \oplus d,$$

where A and C are $n \times n$ matrices and b and d are n -vectors. This system can be put in canonical form in the same way as we have done in the scalar case.

Definition 3.15 (Canonical form of a system of affine equations) *The system $Ax \oplus b = Cx \oplus d$ is said to be in canonical form if A, C, b , and d satisfy*

- $C_{ij} = \varepsilon$ if $A_{ij} > C_{ij}$, and $A_{ij} = \varepsilon$ if $A_{ij} < C_{ij}$;
- $d_i = \varepsilon$ if $b_i > d_i$, and $b_i = \varepsilon$ if $b_i < d_i$.

Example 3.16 Consider the system

$$\begin{pmatrix} 3 & 2 \\ \varepsilon & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \oplus \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 4 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \oplus \begin{pmatrix} e \\ 3 \end{pmatrix},$$

which can be simplified as follows:

$$\begin{pmatrix} \varepsilon & 2 \\ \varepsilon & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \oplus \begin{pmatrix} 1 \\ \varepsilon \end{pmatrix} = \begin{pmatrix} 4 & \varepsilon \\ 1 & \varepsilon \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \oplus \begin{pmatrix} \varepsilon \\ 3 \end{pmatrix},$$

which implies

$$\left. \begin{array}{l} 2x_2 \oplus 1 = 4x_1 \\ 2x_2 = 1x_1 \oplus 3 \end{array} \right\} \Rightarrow 4x_1 = 1x_1 \oplus 3 \Rightarrow 4x_1 = 3 \Rightarrow x_1 = -1 \Rightarrow x_2 = 1.$$

This system has a solution. In general, a linear system may or may not have a solution. Moreover, even if a solution exists, it may be nonunique. ■

There are two classes of linear systems for which we have a satisfactory theory, namely,

- $x = Ax \oplus b$;
- $Ax = b$.

Let us study the former case first.

3.2.3.1 Solution of $x = Ax \oplus b$

Theorem 3.17 *If there are only circuits of nonpositive weight in $G(A)$, there is a solution to $x = Ax \oplus b$ which is given by $x = A^*b$. Moreover, if the circuit weights are negative, the solution is unique.*

The reader should recall the definition of A^* as given by (1.22).

Proof If A^*b does exist, it is a solution; as a matter of fact,

$$A(A^*b) \oplus b = (e \oplus AA^*)b = A^*b.$$

Existence of A^*b . The meaning of $(A^*)_{ij}$ is the maximum weight of all paths of any length from j to i . Thus, a necessary and sufficient condition for the existence of $(A^*)_{ij}$ is that no strongly connected components of $G(A)$ have a circuit with positive weight. Otherwise, there would exist a path from j to i of arbitrarily large weight for all j and i belonging to the strongly connected component which includes the circuit of positive weight (by traversing this circuit a sufficient number of times).

Uniqueness of the solution. Suppose that x is a solution of $x = Ax \oplus b$. Then x satisfies

$$\begin{aligned} x &= b \oplus Ab \oplus A^2x, \\ x &= b \oplus Ab \oplus \dots \oplus A^{k-1}b \oplus A^kx, \end{aligned} \quad (3.3)$$

and thus $x \geq A^*b$. Moreover, if all the circuits of the graph have negative weights, then $A^k \rightarrow \varepsilon$ when $k \rightarrow \infty$. Indeed, the entries of A^k are the weights of the paths of length k which necessarily traverse some circuits of A a number of times going to ∞ with k , but the weights of these circuits are all negative. Using this property in Equation (3.3) for k large enough, we obtain that $x = A^*b$. ■

Remark 3.18 If the maximum circuit weight is zero, a solution does exist, but there is no uniqueness anymore. For example, the equation $x = x \oplus b$ admits the solution $x = a^*b = b$ but all $x > b$ are solutions too. ■

Example 3.19 Consider the two-dimensional equation

$$x = \begin{pmatrix} -1 & 2 \\ -3 & -1 \end{pmatrix} x \oplus \begin{pmatrix} e \\ 2 \end{pmatrix}.$$

Then,

$$b = \begin{pmatrix} e \\ 2 \end{pmatrix}, \quad Ab = \begin{pmatrix} 4 \\ 1 \end{pmatrix}, \quad A^2b = \begin{pmatrix} 3 \\ 1 \end{pmatrix}, \quad A^3b = \begin{pmatrix} 3 \\ e \end{pmatrix}, \quad A^4b = \begin{pmatrix} 2 \\ e \end{pmatrix} \dots$$

Thus,

$$x = \begin{pmatrix} e \\ 2 \end{pmatrix} \oplus \begin{pmatrix} 4 \\ 1 \end{pmatrix} \oplus \begin{pmatrix} 3 \\ e \end{pmatrix} \oplus \begin{pmatrix} 2 \\ e \end{pmatrix} \oplus \dots = \begin{pmatrix} 4 \\ 2 \end{pmatrix}.$$

This is the unique solution. ■

Theorem 3.20 If $G(A)$ has no circuit with positive weight, then

$$A^* = e \oplus A \oplus \dots \oplus A^{n-1},$$

where n is the dimension of matrix A .

Proof All the paths of length greater than or equal to n are necessarily made up of a circuit and of a path with length strictly less than n . Therefore, because the weights of circuits are nonpositive by assumption, we have

$$\forall m \geq n, \quad A^m \leq e \oplus \dots \oplus A^{n-1}.$$

■

Returning to Example 3.19, we remark that $A^2b, A^3b, \dots$ are less than $b \oplus Ab$.

3.2.3.2 Solution of $Ax = b$

The second class of linear systems for which we can obtain a general result consists of the systems $Ax = b$. However, we must first consider the problem in $\mathbb{R}_{\max}$ rather than in $\mathbb{R}_{\min}$, and second, we must somewhat weaken the notion of 'solution'. A *subsolution* of $Ax = b$ is an x which satisfies $Ax \leq b$, where the order relation on the vectors can also be defined by $x \leq y$ if $x \oplus y = y$.

Theorem 3.21 Given an $n \times n$ matrix A and an n -vector b with entries in $\mathbb{R}_{\max}$, the greatest subsolution of $Ax = b$ exists and is given by

$$-x_j = \max_i (-b_i + A_{ij}).$$

For reasons that will become apparent in §4.4.4 and §4.6.2, the vector form of this formula can be written $e\phi x = (e\phi b)A$.

Proof We have that

$$\begin{aligned} \{Ax \leq b\} &\Leftrightarrow \left\{ \bigoplus_j A_{ij}x_j \leq b_i, \quad \forall i \right\} \\ &\Leftrightarrow \{x_j \leq b_i - A_{ij}, \quad \forall i, j\} \\ &\Leftrightarrow \left\{ x_j \leq \min_i (b_i - A_{ij}), \quad \forall j \right\} \\ &\Leftrightarrow \left\{ -x_j \geq \max_i (-b_i + A_{ij}), \quad \forall j \right\}. \end{aligned}$$

Conversely, it can be checked similarly that the vector x defined by $-x_j = \max_i (-b_i + A_{ij}), \forall j$, is a subsolution. Therefore, it is the greatest one. ■

As a consequence, in order to attempt to solve the system $Ax = b$, we may first compute its greatest subsolution and then check by inspection whether it satisfies the equality.

Example 3.22 Let us compute the greatest subsolution of the following equality:

$$\begin{pmatrix} 2 & 3 \\ 4 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 6 \\ 7 \end{pmatrix}.$$

According to the preceding considerations, let us first compute

$$(e\phi b)A = \begin{pmatrix} -6 & -7 \end{pmatrix} \begin{pmatrix} 2 & 3 \\ 4 & 5 \end{pmatrix} = \begin{pmatrix} -3 & -2 \end{pmatrix}.$$

Then the greatest subsolution is $(x_1, x_2) = (3, 2)$; indeed,

$$\begin{pmatrix} 2 & 3 \\ 4 & 5 \end{pmatrix} \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 5 \\ 7 \end{pmatrix} \leq \begin{pmatrix} 6 \\ 7 \end{pmatrix}.$$

It is easily verified that the second inequality would not be satisfied if we increase x_1 and/or x_2 . Therefore, the first inequality cannot be reduced to an equality. ■

3.2.4 Spectral Theory of Matrices

Given a matrix A with entries in $\mathbb{R}_{\max}$, we consider the problem of existence of eigenvalues and eigenvectors, that is, the existence of (nonzero) λ and x such that

$$Ax = \lambda x. \quad (3.4)$$

The main result is as follows.

Theorem 3.23 If A is irreducible, or equivalently if $G(A)$ is strongly connected, there exists one and only one eigenvalue (but possibly several eigenvectors). This eigenvalue is equal to the maximum cycle mean of the graph (see § 2.3.2):

$$\lambda = \max_{\zeta} \frac{|\zeta|_w}{|\zeta|},$$

where ζ ranges over the set of circuits of $G(A)$.

Proof

Existence of x and λ . Consider matrix $B = A\phi\lambda \stackrel{\text{def}}{=} (e\phi\lambda)A$, where $\lambda = \max_{\zeta} |\zeta|_w/|\zeta|$. The maximum circuit weight of $G(B)$ is e . Hence B^* and B^+ exist. Matrix B^+ has some columns with diagonal entries equal to e . To prove this claim, pick a node k of a circuit ξ such that $\xi \in \arg \max_{\zeta} |\zeta|_w/|\zeta|$. The maximum weight of paths from k to k is e . Therefore we have $e = B_{kk}^+$. Let $B_{\cdot k}$ denote the k -th column of B . Then, since, generally speaking, $B^+ = BB^*$ and $B^* = e \oplus B^+$ (e the identity matrix), for that k ,

$$B_{\cdot k}^+ = B_{\cdot k}^* \Rightarrow BB_{\cdot k}^* = B_{\cdot k}^+ = B_{\cdot k}^* \Rightarrow AB_{\cdot k}^* = \lambda B_{\cdot k}^*.$$

Hence $x = B_{\cdot k}^* = B_{\cdot k}^+$ is an eigenvector of A corresponding to the eigenvalue λ . The set of nodes of $G(A)$ corresponding to nonzero entries of x is called the *support* of x .

Graph interpretation of λ . If λ satisfies Equation (3.4), there exists a nonzero component of x , say x_{i_1} . Then we have $(Ax)_{i_1} = \lambda x_{i_1}$ and there exists an index i_2 such that $A_{i_1 i_2} x_{i_2} = \lambda x_{i_1}$. Hence $x_{i_2} \neq \varepsilon$ and $A_{i_1 i_2} \neq \varepsilon$. We can repeat this argument and obtain a sequence $\{i_j\}$ such that $A_{i_{j-1} i_j} x_{i_j} = \lambda x_{i_{j-1}}$, $x_{i_j} \neq \varepsilon$ and $A_{i_{j-1} i_j} \neq \varepsilon$. At some stage we must reach an index i_l already encountered in the sequence since the number of nodes is finite. Therefore, we obtain a circuit $\beta = (i_l, i_m, \dots, i_{l+1}, i_l)$. By multiplication along this circuit, we obtain

$$A_{i_l i_{l+1}} A_{i_{l+1} i_{l+2}} \dots A_{i_m i_l} x_{i_{l+1}} x_{i_{l+2}} \dots x_{i_m} x_{i_l} = \lambda^{m-l+1} x_{i_l} x_{i_{l+1}} \dots x_{i_m}.$$

Since $x_{i_j} \neq \varepsilon$ for all i_j , we may simplify the equation above which shows that λ^{m-l+1} is the weight of the circuit of length $m-l+1$, or, otherwise stated, λ is the average weight of circuit β . Observe that this part of the proof did not use the irreducibility assumption.

If A is irreducible, all the components of x are different from ε . Suppose that the support of x does not cover the whole graph. Then, there are arcs going from the support of x to other nodes because the graph $G(A)$ has only one strongly connected component. Therefore, the support of Ax is larger than the support of x , which contradicts Equation (3.4).

Uniqueness in the irreducible case. Consider any circuit $\gamma = (i_1, \dots, i_p, i_1)$ such that its nodes belong to the support of x (here any node of $G(A)$). We have

$$A_{i_2 i_1} x_{i_1} \leq \lambda x_{i_2}, \quad \dots, \quad A_{i_p i_{p-1}} x_{i_{p-1}} \leq \lambda x_{i_p}, \quad A_{i_1 i_p} x_{i_p} \leq \lambda x_{i_1}.$$

Hence, by the same argument as in the paragraph on the graph interpretation of λ , we see that λ is greater than the average weight of γ . Therefore λ is the maximum cycle mean and thus it is unique. ■

It is important to understand the role of the support of x in the previous proof. If $G(A)$ is not strongly connected, the support of x is not necessarily the whole set of nodes and, in general, there is no unique eigenvalue (see Example 3.26 below).

Remark 3.24 The part of the proof on the graph interpretation of λ indeed showed that, for a general matrix A , any eigenvalue is equal to some cycle mean. Therefore the maximum cycle mean is equal to the maximum eigenvalue of the matrix. ■

Example 3.25 (Nonunique eigenvector) With the only assumption of Theorem 3.23 on irreducibility, the uniqueness of the eigenvector is not guaranteed as is shown by the following example:

$$\begin{pmatrix} 1 & e \\ e & 1 \end{pmatrix} \begin{pmatrix} e \\ -1 \end{pmatrix} = \begin{pmatrix} 1 \\ e \end{pmatrix} = 1 \begin{pmatrix} e \\ -1 \end{pmatrix},$$

and

$$\begin{pmatrix} 1 & e \\ e & 1 \end{pmatrix} \begin{pmatrix} -1 \\ e \end{pmatrix} = \begin{pmatrix} e \\ 1 \end{pmatrix} = 1 \begin{pmatrix} -1 \\ e \end{pmatrix}.$$

The two eigenvectors are obviously not 'proportional'. ■

Example 3.26 (A not irreducible)

- The following example is a trivial counterexample to the uniqueness of the eigenvalue when $G(A)$ is not connected:

$$\begin{pmatrix} 1 & \varepsilon \\ \varepsilon & 2 \end{pmatrix} \begin{pmatrix} e \\ \varepsilon \end{pmatrix} = 1 \begin{pmatrix} e \\ \varepsilon \end{pmatrix}, \quad \begin{pmatrix} 1 & \varepsilon \\ \varepsilon & 2 \end{pmatrix} \begin{pmatrix} \varepsilon \\ e \end{pmatrix} = 2 \begin{pmatrix} \varepsilon \\ e \end{pmatrix}.$$

- In the following example $G(A)$ is connected but not strongly connected. Nevertheless there is only one eigenvalue:

$$\begin{pmatrix} 1 & e \\ \varepsilon & e \end{pmatrix} \begin{pmatrix} e \\ \varepsilon \end{pmatrix} = 1 \begin{pmatrix} e \\ \varepsilon \end{pmatrix},$$

but

$$\begin{pmatrix} 1 & e \\ \varepsilon & e \end{pmatrix} \begin{pmatrix} a \\ e \end{pmatrix} = \lambda \begin{pmatrix} a \\ e \end{pmatrix}$$

has no solutions because the second equation implies $\lambda = e$, and then the first equation has no solutions for the unknown a .

- In the following example $G(A)$ is connected but not strongly connected and there are two eigenvalues:

$$\begin{pmatrix} e & e \\ \varepsilon & 1 \end{pmatrix} \begin{pmatrix} e \\ \varepsilon \end{pmatrix} = e \begin{pmatrix} e \\ \varepsilon \end{pmatrix}, \quad \begin{pmatrix} e & e \\ \varepsilon & 1 \end{pmatrix} \begin{pmatrix} e \\ 1 \end{pmatrix} = 1 \begin{pmatrix} e \\ 1 \end{pmatrix}.$$

More generally, consider the block triangular matrix

$$F = \begin{pmatrix} A & \varepsilon \\ B & C \end{pmatrix},$$

where $G(A)$ and $G(C)$ are strongly connected, and $G(C)$ is downstream of $G(A)$. Let λ_A and λ_C be the eigenvalues of blocks A and C , respectively, and let x_A and x_C be the corresponding eigenvectors. Observe that $\begin{pmatrix} \varepsilon & x_C \end{pmatrix}^t$ is an eigenvector of F for the eigenvalue λ_C . In addition, if $\lambda_A > \lambda_C$, the expression $(C \not\lambda_A)^*$ is well-defined. The vector

$$\begin{pmatrix} x_A \\ (C \not\lambda_A)^* (B \not\lambda_A) x_A \end{pmatrix}$$

is an eigenvector of F for the eigenvalue λ_A . In conclusion, F has two eigenvalues if the upstream m.s.c.s. is 'slower' than the downstream one. Clearly this kind of result can be generalized to a decomposition into an arbitrary number of blocks. This generalization will not be considered here (see [62]).

3.2.5 Application to Event Graphs

Consider an autonomous event graph with n transitions, that is, an event graph without sources, with constant holding times and zero firing times. In §2.5.2 we saw that it can be modeled by

$$x(k) = \bigoplus_{i=0}^M A(i)x(k-i), \quad (3.5)$$

where the $A(i)$ are $n \times n$ matrices with entries in $\mathbb{R}_{\max}$. We assume that the event graph (in which transitions are viewed as the nodes and places as the arcs) is strongly connected. In §2.5.4, it was shown that an equation in the standard form

$$\bar{x}(k+1) = \bar{A}\bar{x}(k) \quad (3.6)$$

can also describe the same physical system. A new event graph (with a different number of transitions in general) can be associated with (3.6). In this new event graph, each place has exactly one token in the initial marking. Therefore, in this graph, the length of a circuit or path can either be defined as the total number of arcs, or as the total number of tokens in the initial marking along this circuit or path.

We refer the reader to the transformations explained in §2.5.2 to §2.5.4 to see that some transitions in the two graphs can be identified to each other and that the circuits are in one-to-one correspondence. Since the original event graph is assumed to be strongly connected, the new event graph can also be assumed to be strongly connected, provided unnecessary transitions (not involved in circuits) be canceled. Then $\bar{A}$ is irreducible. Hence there exists a unique eigenvalue λ , and at least one eigenvector $\bar{x}$. By starting the recurrence in (3.6) with the initial value $\bar{x}(0) = \bar{x}$, we obtain that $\bar{x}(k) = \lambda^k \bar{x}$ for all k in $\mathbb{N}$. Therefore, a token leaves each transition every λ units of time, or, otherwise stated, the throughput of each transition is $1/\lambda$.

It was shown that λ can be evaluated as the maximum cycle mean of $G(\bar{A})$, that is, as the maximum ratio 'weight divided by length' over all the circuits of $G(\bar{A})$. The purpose of the following theorem is to show that λ can also be evaluated as the same maximum ratio over the circuits of the *original* graph, provided the length of an arc be understood as the number of tokens in the initial marking of the corresponding place (the weight is still defined as the holding time).

Let us return to (3.5). The graph $G(A(i))$ describes the subgraph of the original graph obtained by retaining only the arcs corresponding to places marked with i tokens. Since $A(i)$ is an $n \times n$ matrix, all original nodes are retained in this subgraph which is however not necessarily connected. Consider the following $n \times n$ matrix:

$$B(\lambda) \stackrel{\text{def}}{=} \bigoplus_{i=0}^M \lambda^{-i} A(i),$$

where λ is any real number.

Remark 3.27 For a given value of λ , a circuit of $B(\lambda)$ can be defined by a sequence of nodes (in which the last node equals the first). Indeed, once this sequence of nodes

is given, the arc between a pair of successive nodes (a, b) is selected by the argument $i_\lambda(a, b)$ of the $\max_i$ which is implicit in the expression of $(B(\lambda))_{ba}$. If this $i_\lambda(a, b)$ is not unique, it does not matter which one is selected since any choice leads to the same weight. Therefore, the set of circuits of $G(B(\lambda))$ is a subset of the set of circuits of the original graph (a circuit of the original graph can be specified by a given sequence of nodes *and* by a mapping $(a, b) \mapsto i(a, b)$ in order to specify one of the possible parallel arcs between nodes a and b). The set of circuits of $G(B(\lambda))$ is thus changing with λ . However, for any given value of λ , if we are only interested in the maximum circuit weight of $G(B(\lambda))$ (or in the maximum cycle mean, assuming that the length of a circuit is defined as the number of arcs in $G(B(\lambda))$), the maximum can be taken over the whole set of circuits of the original graph (this set is independent of λ). Indeed, the additional circuits thus considered do not contribute to the maximum since they correspond to choices of $i(a, b)$ which are not optimal. ■

Theorem 3.28 *We assume that*

1. $G(B(e))$ is strongly connected;
2. $G(A(0))$ has circuits of negative weight only;
3. there exists at least one circuit of $G(B(\lambda))$ containing one token.

Then, there exist a vector x and a unique scalar λ satisfying $x = B(\lambda)x$. The graph interpretation of λ is

$$\lambda = \max_{\zeta} \frac{|\zeta|_w}{|\zeta|_t}, \quad (3.7)$$

where ζ ranges over the set of circuits of the original event graph and $|\zeta|_t$ denotes the number of tokens in circuit ζ .

Proof To solve the equation $x = B(\lambda)x$, we must find λ and x such that e is an eigenvalue of the matrix $B(\lambda)$. The graph $G(B(e))$ being strongly connected, $G(B(\lambda))$ is also strongly connected for any real value of λ and therefore $B(\lambda)$ admits a unique eigenvalue $\Lambda(\lambda)$. Owing to the graph interpretation of $\Lambda(\lambda)$, $\Lambda(\lambda) = \max_{\zeta} |\zeta|_w / |\zeta|_t$, where ζ ranges over the set of circuits of $G(B(\lambda))$. However, Remark 3.27 showed that we can as well consider that ζ ranges over the set of circuits of the original graph. Hence, in conventional notation, we have

$$\begin{aligned} \Lambda(\lambda) &= \max_{\zeta} \left(\frac{1}{|\zeta|_t} \sum_{(a,b) \in \zeta} (B(\lambda))_{ba} \right) \\ &= \max_{\zeta} \left(\frac{1}{|\zeta|_t} \sum_{(a,b) \in \zeta} \max_{i \in \{1, \dots, M\}} ((A(i))_{ba} - i \times \lambda) \right) \\ &= \max_{\zeta} \left(\frac{1}{|\zeta|_t} \max_{i(a,b)} \sum_{(a,b) \in \zeta} ((A(i(a,b)))_{ba} - i(a,b) \times \lambda) \right). \end{aligned} \quad (3.8)$$

If we assume that there exists a λ such that $\Lambda(\lambda) = e = 0$, then, for any circuit ζ of the original graph, and thus for any mapping $i(\cdot, \cdot)$ which completes the specification of the circuit, we have

$$\lambda \geq \frac{\sum_{(a,b) \in \zeta} (A(i(a,b)))_{ba}}{\sum_{(a,b) \in \zeta} i(a,b)},$$

and the equality is obtained for some circuit ζ . This justifies the interpretation (3.7) of λ .

Let us now prove that $\Lambda(\lambda) = e$ has a unique solution. Because of (3.8), and since, according to Remark 3.27, the mappings $i(\cdot, \cdot)$ can be viewed as ranging in a set independent of λ , $\Lambda(\cdot)$ is the upper hull of a collection of affine functions (of λ). Each affine function has a nonpositive slope which, in absolute value, equals the number of tokens in a circuit divided by the number of arcs in the circuit. Therefore, Λ is a nonincreasing function of λ . Moreover, due to the third assumption of the theorem, there is at least one strictly negative slope. Hence $\lim_{\lambda \rightarrow -\infty} \Lambda(\lambda) = +\infty$. On the other hand, owing to the second assumption, and since the affine functions with zero slope stem necessarily from the circuits of $A(0)$, $\lim_{\lambda \rightarrow +\infty} \Lambda(\lambda) < 0$. Finally Λ is a convex nonincreasing function which decreases from $+\infty$ to a strictly negative value, and thus its graph crosses the x -axis at a single point. ■

It is easy to see that if we start the recurrence (3.5) with $x(0) = x, x(1) = \lambda \otimes x, \dots, x(M) = \lambda^M \otimes x$, then $x(k) = \lambda^k \otimes x$ for all k . Hence $1/\lambda$ is the throughput of the system at the periodic regime. At the end of this chapter, we will give conditions under which this regime is asymptotically reached, and conditions under which it is reached after a finite time, whatever the initial condition is.

3.3 SCALAR FUNCTIONS IN $\mathbb{R}_{\max}$

In this section we discuss nonlinear real-valued functions of one real variable, considered as mappings from $\mathbb{R}_{\max}$ into $\mathbb{R}_{\max}$. We classify them in polynomial, rational and algebraic functions in the max-plus algebra sense.

3.3.1 Polynomial Functions $\mathcal{P}(\mathbb{R}_{\max})$

Polynomial functions are a subset of piecewise linear functions (in the conventional sense) for which we have the analogue of the fundamental theorem of algebra. This set is not isomorphic to the set of formal polynomials of $\mathbb{R}_{\max}$, that is, the set of finite sequences endowed with a product which is the sup-convolution of sequences.

3.3.1.1 Formal Polynomials and Polynomial Functions

Definition 3.29 (Formal polynomials) We consider the set of finite real sequences of any length

$$p = (p(k), \dots, p(i) \dots p(n)) \quad , \quad k, i, n \in \mathbb{N} \quad , \quad p(i) \in \mathbb{R}_{\max} \quad .$$

If the extreme values k and n are such that $p(k)$ and $p(n)$ are different from ε , then $\text{val}(p) \stackrel{\text{def}}{=} k$ is called the valuation of p , and $\text{deg}(p) \stackrel{\text{def}}{=} n$ is called the degree of p . This set is endowed with the following two internal operations:

- componentwise addition $\oplus$;
- sup-convolution $\otimes$ of sequences, that is,

$$(p \otimes q)(l) \stackrel{\text{def}}{=} \bigoplus_{\substack{i+j=l \\ \text{val}(p) \leq i \leq \text{deg}(p) \\ \text{val}(q) \leq j \leq \text{deg}(q)}} p(i)q(j) \quad ,$$

and with the following external operation involving scalars in $\mathbb{R}_{\max}$:

- multiplication of all the elements of the sequence p by the same scalar of $\mathbb{R}_{\max}$,

We thus define $\mathbb{R}_{\max}[\gamma]$ which is called the set of formal polynomials.

Note that if the polynomial γ is defined by

$$\gamma(k) = \begin{cases} e & \text{if } k = 1 ; \\ \varepsilon & \text{otherwise,} \end{cases}$$

then any polynomial of $\mathbb{R}_{\max}[\gamma]$ can be written as $p = \bigoplus_{l=k}^n p(l)\gamma^l$.

Let us give a list of definitions related to the notion of formal polynomials.

Definition 3.30

Polynomial functions: associated with a formal polynomial p , we define the polynomial function by

$$\hat{p} : \mathbb{R}_{\max} \rightarrow \mathbb{R}_{\max} \quad , \quad c \mapsto \hat{p}(c) = p(k)c^k \oplus \dots \oplus p(n)c^n \quad .$$

The set of polynomial functions is denoted $\mathcal{P}(\mathbb{R}_{\max})$.

Support: the support $\text{supp}(p)$ is the set of indices of the nonzero elements of p , that is, $\text{supp}(p) = \{i \mid k \leq i \leq n, p(i) \neq \varepsilon\}$.

Monomial: a formal polynomial reduced to a sequence of one element is called a monomial¹.

Head monomial: the monomial of highest degree one can extract from a polynomial p , that is $p(n)\gamma^n$, is called the head monomial.

Tail monomial: the monomial of lowest degree out of p , that is, $p(k)\gamma^k$, is called the tail monomial.

¹We do not make the distinction between formal monomials and monomial functions because, unlike polynomials—see Remark 3.34 below—a formal monomial is in one-to-one correspondence with its associated function.

Full Support: we say that a formal polynomial has a full support if

$$p(i) \neq \varepsilon, \quad \forall i: k \leq i \leq n.$$

The following two theorems are obvious.

Theorem 3.31 The set of formal polynomials $\mathbb{R}_{\max}[\gamma]$ is an idempotent algebra.

Remark 3.32 Because we can identify scalars with monomials of degree 0, this idempotent algebra can be viewed as the idempotent semiring obtained by considering the two internal operations only, since the external multiplication by a scalar is amenable to the internal multiplication by a monomial. ■

Theorem 3.33 The set of polynomial functions $\mathcal{P}(\mathbb{R}_{\max})$ endowed with the two internal operations:

- pointwise addition denoted $\oplus$, that is, $(\widehat{p} \oplus \widehat{q})(c) \stackrel{\text{def}}{=} \widehat{p}(c) \oplus \widehat{q}(c)$;
- pointwise multiplication denoted $\otimes$, that is, $(\widehat{p} \otimes \widehat{q})(c) \stackrel{\text{def}}{=} \widehat{p}(c) \otimes \widehat{q}(c)$,

and the external operation over $\mathbb{R}_{\max} \times \mathcal{P}(\mathbb{R}_{\max})$, namely,

- $(b\widehat{p})(c) \stackrel{\text{def}}{=} b \otimes \widehat{p}(c)$,

is an idempotent algebra.

The same remark as above applies to this idempotent algebra too.

Polynomial functions are convex piecewise linear integer-sloped nondecreasing functions (see Figure 3.4). Indeed the monomial $p(i)c^i$ is nothing but the conventional affine function $ic + p(i)$. Owing to the meaning of addition of monomials, polynomial functions are thus upper hulls of such affine functions.

Remark 3.34 There is no one-to-one correspondence between formal polynomials and polynomial functions. For example,

$$\forall c, \quad c^2 \oplus 2 = (c \oplus 1)^2 = c^2 \oplus 1c \oplus 2.$$

The monomial $1c$ is dominated by $c^2 \oplus 2$. In other words, $1c$ does not contribute to the graph of $c^2 \oplus 1c \oplus 2$ (see Figure 3.5), and thus, two different formal polynomials are associated with the same function. ■

The following lemma should be obvious.

Lemma 3.35 (Evaluation homomorphism) Consider the mapping $\mathcal{F} : \mathbb{R}_{\max}[\gamma] \rightarrow \mathcal{P}(\mathbb{R}_{\max}), p \mapsto \widehat{p}$. Then, $\mathcal{F}$ is a homomorphism between the algebraic structures defined in Theorems 3.31 and 3.33. It will be referred to as the evaluation homomorphism.

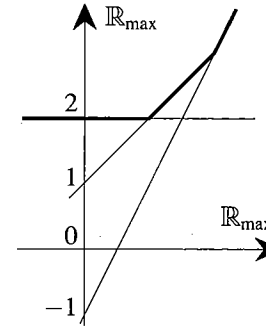


Figure 3.4: The graph of $y = (-1)c^2 \oplus 1c \oplus 2$

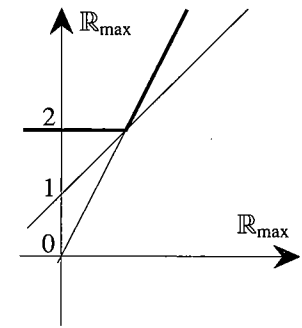


Figure 3.5: The graph of $y = (c \oplus 1)^2$

Unlike in conventional algebra, the evaluation homomorphism is not a one-to-one correspondence, as shown in Remark 3.34. More precisely, it is surjective but not injective. This is why it is important to distinguish between formal objects and their associated numerical functions.

Remark 3.36 The mapping $\mathcal{F}$ is in fact closely related to the Fenchel transform [58]. In convexity theory [119], with a numerical function f over a Hilbert space V , the Fenchel transform $\mathcal{F}_e$ associates a numerical function $\mathcal{F}_e(f)$ over the dual Hilbert space V^* as follows:

$$\forall c \in V^*, \quad [\mathcal{F}_e(f)](c) = \sup_{z \in V} (\langle c, z \rangle - f(z)),$$

where $\langle \cdot, \cdot \rangle$ denotes the duality product over $V^* \times V$. If we consider the formal polynomial p as a function from $\mathbb{N}$ into $\mathbb{R}_{\max}$, $p : l \mapsto p(l)$ (the domain of which can be extended to the whole $\mathbb{N}$ by setting $p(l) = \varepsilon$ if $l < \text{val}(p)$ or $l > \text{deg}(p)$), then,

$$\widehat{p}(c) = \max_{l \in \mathbb{N}} (lc + p(l)) = \mathcal{F}_e(-p)(c). \quad (3.9)$$

Before studying the properties of $\mathcal{F}$ and the equivalence relation it induces in $\mathbb{R}_{\max}[\gamma]$, let us first give some terminology related to convexity that we will use later on.

Definition 3.37

Convex set in a disconnected domain: we say that a subset $F \subset E = \mathbb{N} \times \mathbb{R}$ is convex if, for all $\mu \in [0, 1]$ and for all $x, y \in F$ for which $\mu x + (1 - \mu)y \in E$, it follows that $\mu x + (1 - \mu)y \in F$.

Hypograph and epigraph of a real function $f : X \rightarrow \mathbb{R}$: these are the sets defined respectively by

$$\text{hypo}(f) \stackrel{\text{def}}{=} \{(x, y) \mid x \in X, y \in \mathbb{R}, y \leq f(x)\},$$

$$\text{epi}(f) \stackrel{\text{def}}{=} \{(x, y) \mid x \in X, y \in \mathbb{R}, y \geq f(x)\}.$$

Convex (concave) mapping: a function $f : X \rightarrow \mathbb{R}$ is convex (respectively concave) if its epigraph (respectively its hypograph) is convex.

Extremal point of a convex set F : it is a point which is not a convex combination of two other distinct points of F .

Theorem 3.38 Consider the equivalence relation

$$\forall p, q \in \mathbb{R}_{\max}[\gamma], \quad p \stackrel{\mathcal{F}}{=} q \Leftrightarrow \widehat{p} = \widehat{q} \Leftrightarrow q \in \mathcal{F}^{-1}(\widehat{p}).$$

1. With a given $p \in \mathbb{R}_{\max}[\gamma]$, we associate two other elements of $\mathbb{R}_{\max}[\gamma]$ denoted $p_1^\sharp$ and $p_2^\sharp$, such that, for all $l \in \mathbb{N}$,

$$p_1^\sharp(l) \stackrel{\text{def}}{=} \max_{\substack{0 \leq \mu \leq 1 \\ i, j \in \mathbb{N}}} (\mu p(i) + (1 - \mu)p(j)), \quad (3.10)$$

subject to $l = \mu i + (1 - \mu)j$,

$$p_2^\sharp(l) \stackrel{\text{def}}{=} \min_{c \in \mathbb{R}_{\max}} (\widehat{p}(c) - lc). \quad (3.11)$$

Then $p_1^\sharp = p_2^\sharp$ (denoted simply $p^\sharp$) and $p^\sharp$ belongs to the same equivalence class as p of which it is the maximum element. The mapping $l \mapsto p^\sharp(l)$ is the concave upper hull of $l \mapsto p(l)$. Hence $\text{hypo}(p^\sharp)$ is convex.

2. Let now $p^\flat \in \mathbb{R}_{\max}[\gamma]$ be obtained from p by canceling the monomials of p which do not correspond to extremal points of $\text{hypo}(p^\sharp)$. Then $p^\flat$ belongs to the same equivalence class as p of which it is the minimum element.
3. Two members p and q of the same equivalence class have the same degree and valuation. Moreover $p^\sharp = q^\sharp$ and $p^\flat = q^\flat$.

Proof

1. Using (3.10) with the particular values $\mu = 1$, hence $l = i$, we first prove that $p_1^\sharp \geq p$ for the pointwise conventional order (which is also the natural order associated with the addition in $\mathbb{R}_{\max}[\gamma]$). Combining (3.9) (written for $p_1^\sharp$) with (3.10), we obtain

$$\begin{aligned} \widehat{p_1^\sharp}(c) &= \max_l \left(lc + \max_{0 \leq \mu \leq 1, i, j \in \mathbb{N}} (\mu p(i) + (1 - \mu)p(j)) \right) \\ &\quad \text{subject to } l = \mu i + (1 - \mu)j \\ &= \max_{0 \leq \mu \leq 1} \left(\mu \max_i (ic + p(i)) + (1 - \mu) \max_j (jc + p(j)) \right) \\ &= \max_{0 \leq \mu \leq 1} (\mu \widehat{p}(c) + (1 - \mu)\widehat{p}(c)) \\ &= \widehat{p}(c). \end{aligned}$$

This shows that $p_1^\sharp$ belongs to the same equivalence class as p and that it is greater than any such p , hence it is the maximum element in this equivalence class.

On the other hand, combining (3.11) with (3.9), we get

$$p_2^\sharp(l) = \min_{c \in \mathbb{R}_{\max}} \left(\max_{m \in \mathbb{N}} (mc + p(m)) - lc \right).$$

By choosing particular value $m = l$, it is shown that $p_2^\sharp \geq p$. Since $\mathcal{F}$ is a homomorphism, it preserves the order, and thus $\widehat{p_2^\sharp} \geq \widehat{p}$. But, if we combine (3.9) (written for $p_2^\sharp$) with (3.11), we get

$$\widehat{p_2^\sharp}(c) = \max_{l \in \mathbb{N}} \left(lc + \min_{c' \in \mathbb{R}_{\max}} (\widehat{p}(c') - lc') \right).$$

By picking the particular value $c' = c$, it is shown that $\widehat{p_2^\sharp} \leq \widehat{p}$. Hence, we have proved that $\widehat{p_2^\sharp} = \widehat{p}$ and that $p_2^\sharp \geq p$. Therefore $p_2^\sharp$ is the maximum element in the equivalence class of p . Since the maximum element is unique (see § 4.3.1), it follows that $p_1^\sharp = p_2^\sharp$.

From (3.11), it is apparent that $p_2^\sharp$ is concave as the lower hull of a family of affine functions. Hence, since it is greater than p , it is greater than its concave upper hull, but (3.10) shows that indeed it coincides with this hull.

2. It is now clear that the equivalence class of p can be characterized by $p^\sharp$, or equivalently by its hypograph which is a convex set. Since a convex set is fully characterized by the collection of its extreme points, this collection is another characterization of the class. Since $p^\flat$ has precisely been defined from this collection of extreme points, it is clearly an element of the same class and the minimum one (dropping any further monomial would change the collection of extreme points and thus the equivalence class).
3. In particular, the head and tail monomials of a given p correspond to members of the collection of extreme points. Therefore, all elements of an equivalence class have the same degree and valuation. ■

Definition 3.39 (Canonical forms of polynomial functions) According to the previous theorem, we may call $p^\sharp$ and $p^\flat$ the concavified polynomial and the skeleton of p , respectively.

- The skeleton $p^\flat$ is also called the minimum canonical form of the polynomial function $\widehat{p}$;
- the concavified polynomial $p^\sharp$ is also called the maximum canonical form of $\widehat{p}$.

Figure 3.6 illustrates these notions. It should be clear that necessarily $p^\sharp$ has full support.

Example 3.40 For $p = \gamma^2 \oplus \gamma \oplus 2$, we have $p^\flat = \gamma^2 \oplus 2$. For $p = \gamma^3 \oplus 3$, we have $p^\sharp = \gamma^3 \oplus 1\gamma^2 \oplus 2\gamma \oplus 3$. ■

Lemma 3.41 A formal polynomial p of valuation k and degree n is a maximal canonical form of a polynomial function $\hat{p}$, that is, $p = p^\sharp$, if and only if p has full support (hence $p(l) \neq \varepsilon$, for $l = k, \dots, n$), and

$$\frac{p(n-1)}{p(n)} \geq \frac{p(n-2)}{p(n-1)} \geq \dots \geq \frac{p(k)}{p(k+1)}. \quad (3.12)$$

Proof The fact that a maximal canonical form must have full support has been established earlier and this ensures that the ‘ratios’ in (3.12) are well defined. In the proof of Theorem 3.38 it has also been shown that $p^\sharp$ is the concave upper hull of the function $l \mapsto p(l)$ (concavity in the sense of Definition 3.37). Conversely, if p is concave, it is equal to its own concave upper hull, and thus $p = p^\sharp$. Now, (3.12) simply expresses that the slopes of the lines defined by the successive pairs of points $((l-1, p(l-1)), (l, p(l)))$ are decreasing with l , which is obviously a necessary and sufficient condition for p to be concave. ■

3.3.1.2 Factorization of Polynomials

Let us now show that polynomial functions and concave formal polynomials can be factored into a product of linear factors.

Definition 3.42 (Corners, multiplicity) The nonzero corners of a polynomial function $\hat{p}$ are the abscissae of the extremal points of the epigraph of $\hat{p}$. Since $\hat{p}$ is convex, at such a corner, the (integer) slope increases by some integer which is called the multiplicity of the corner. A corner is called multiple if the multiplicity is larger than one.

The zero corner exists if the least slope appearing in the graph of $\hat{p}$ is nonzero: the multiplicity of this zero corner is then equal to that (integer) slope.

Figure 3.7 shows a nonzero corner of multiplicity 2.

Theorem 3.43 (Fundamental theorem of algebra)

1. Any formal polynomial of the form $p = p(n) \bigotimes_{i=1}^n (\gamma \oplus c_i)$ (where some c_i may be equal to ε) satisfies $p = p^\sharp$.
2. Conversely, if a formal polynomial $p = p(k)\gamma^k \oplus p(k+1)\gamma^{k+1} \oplus \dots \oplus p(n)\gamma^n$ is such that $p = p^\sharp$, then it has full support, the numbers $c_i \in \mathbb{R}_{\max}$ defined by

$$c_i \stackrel{\text{def}}{=} \begin{cases} p(n-i)/p(n-i+1) & \text{for } 1 \leq i \leq n-k; \\ \varepsilon & \text{for } n-k < i \leq n, \end{cases} \quad (3.13)$$

are such that

$$c_1 \geq c_2 \geq \dots \geq c_n, \quad (3.14)$$

and p can be factored as $p = p(n) \left(\bigotimes_{i=1}^n (\gamma \oplus c_i) \right)$.

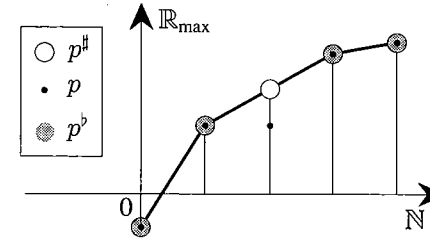


Figure 3.6: The functions p , $p^\sharp$ and $p^\flat$

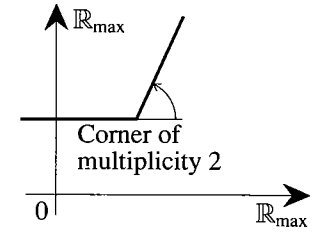


Figure 3.7: Corner of a polynomial function

3. Any polynomial function $\hat{p}$ can be factored as

$$\hat{p}(c) = p(n) \left(\bigotimes_{i=1}^n (c \oplus c_i) \right),$$

where the c_i are the zero and/or nonzero corners of the polynomial function $\hat{p}$ repeated with their order of multiplicity. These corners are obtained by Equation (3.13) using the maximum canonical form.

Proof

1. Let $p = \bigotimes_{i=1}^n p(n)(\gamma \oplus c_i)$ and assume without loss of generality that the c_i have been numbered in such a way that (3.14) holds. We consider the nontrivial case $p(n) \neq \varepsilon$. By direct calculation, it can be seen that $p(n-k) = p(n)\sigma_k$, where σ_k is the k -th symmetric product of the corners c_i , that is,

$$\sigma_1 = \bigoplus_{i=1}^n c_i, \quad \sigma_2 = \bigoplus_{i \neq j=1}^n c_i c_j, \dots$$

Owing to our assumption on the ordering of the c_i , it is clear that $\sigma_k = \bigotimes_{i=1}^k c_j$. Therefore,

$$\frac{p_{n-k}}{p_{n-k+1}} = \frac{\sigma_k}{\sigma_{k-1}} = c_k \leq c_{k-1} = \frac{p_{n-k+1}}{p_{n-k+2}}.$$

Thus $p = p^\sharp$ by Lemma 3.41.

2. If $p = p^\sharp$, the c_i can be defined by (3.13) unambiguously and then (3.14) follows from Lemma 3.41. The fact that p can be factored as indicated is checked as previously by direct calculation.
3. From the preceding considerations, provided that we represent $\hat{p}$ with the help of its maximum canonical form, the factored form can be obtained if we define the c_i with Equation (3.13). To complete the proof, it must be shown that any

such c_i is a corner in the sense of Definition 3.42 and that, if a c_i appears k_i times, then the slope jumps by k_i at c_i . To see this, rewrite the factored form in conventional notation, which yields

$$\hat{p}(c) = \sum_{i=1}^n \max(c, c_i) .$$

Each elementary term $c \oplus c_i$ has a graph represented in Figure 3.2 with a slope discontinuity equal to one at c_i . If a c_i appears k_i times, the term $k_i \times \max(c, c_i)$ causes a slope discontinuity equal to k_i . All other terms with $c_j \neq c_i$ do not cause any slope discontinuity at c_i . ■

Example 3.44 The formal polynomial $\gamma^2 \oplus 3\gamma \oplus 2$ is a maximum canonical form because $c_1 = 3\neq e = 3 \geq c_2 = 2\neq 3 = -1$, and therefore it can be factored into $(\gamma \oplus 3)(\gamma \oplus (-1))$. The formal polynomial $(\gamma^2 \oplus 2)^\# = \gamma^2 \oplus 1\gamma \oplus 2$ can be factored into $(\gamma \oplus 1)^2$. ■

3.3.2 Rational Functions

In this subsection we study rational functions in the $\mathbb{R}_{\max}$ algebra. These functions are continuous, piecewise linear, integer-sloped functions. We give the multiplicative form of such functions, which completely defines the points where the slope changes. Moreover, we show that the Euclidean division and the decomposition into simple elements is not always possible.

3.3.2.1 Definitions

Definition 3.45 Given $p(0), \dots, p(n), q(0), \dots, q(m) \in \mathbb{R}_{\max}$, $p(n)$ and $q(m) \neq e$, the rational function $\hat{r}$, associated with these coefficients, is given by

$$\hat{r} : \mathbb{R}_{\max} \rightarrow \mathbb{R}_{\max} , \quad c \mapsto \hat{r}(c) = \frac{p(0) \oplus \dots \oplus p(n)c^n}{q(0) \oplus \dots \oplus q(m)c^m} .$$

Such a function is equal to the difference of two polynomial functions (see Figure 3.8): hence it is still continuous, piecewise linear, integer-sloped, but it is neither convex nor increasing anymore.

Definition 3.46 The corners of the numerator are called zero corners or root corners, and the corners of the denominator are called pole corners.

Using the fundamental theorem of algebra, we can write any rational function $\hat{r}$ as

$$\hat{r}(c) = a \frac{\bigotimes_{i=1}^n (c \oplus c_i)}{\bigotimes_{j=1}^m (c \oplus d_j)} ,$$

where the zero and pole corners are possibly repeated with their order of multiplicity. At a zero corner of multiplicity k_i , the change of slope is k_i (unless this zero corner coincides with some pole corner). At a pole corner of multiplicity l_j , the change of slope is $-l_j$ (see Figure 3.9).

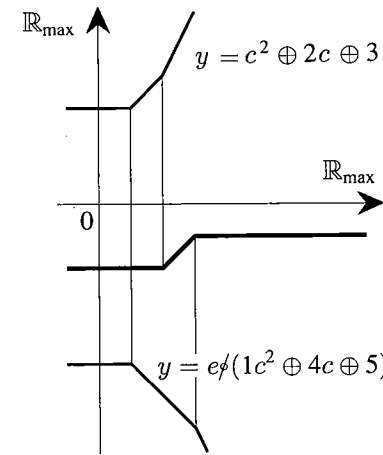


Figure 3.8: Graph of $y = (c^2 \oplus 2c \oplus 3) \neq (1c^2 \oplus 4c \oplus 5)$

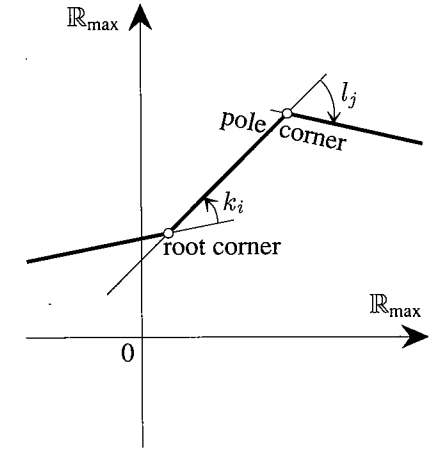


Figure 3.9: Root and pole corners

3.3.2.2 Euclidean Division

In general, a polynomial $\hat{p}$ cannot be expressed as $\hat{b}\hat{q} \oplus \hat{r}$ with $\deg(\hat{r}) < \deg(\hat{b})$ for some given polynomial $\hat{b}$ as shown by Example 3.47. Nevertheless, sometimes we can obtain such a decomposition, as shown by Example 3.48.

Example 3.47 The equation $c^2 \oplus e = \hat{q}(c \oplus 1) \oplus \hat{r}$ has no solutions. Indeed, $\hat{q}$ must be of degree 1 and $\hat{r}$ of degree 0; thus $q = q(1)\gamma \oplus q(0)$, $r = r(0)$, with $q(0), q(1), r(0) \in \mathbb{R}_{\max}$. By identifying the coefficients of degree 2 in both sides, we must have $q(1) = e$. Now, since the maximal canonical form in the left-hand side is $c^2 \oplus c \oplus e$, by considering the coefficient of degree 1 in the right-hand side, we must have $q(0) \oplus 1q(1) \leq e$, which contradicts $q(1) = e$. ■

Example 3.48 For $\hat{p}(c) = c^2 \oplus 3$ and $\hat{b} = c \oplus 1$, we have

$$c^2 \oplus 3 = (c \oplus 1)^2 \oplus 3 = (c \oplus 1)(c \oplus 1.5) \oplus 3 .$$

As in conventional algebra, this issue of the Euclidean division leads to solving a triangular system of linear equations. However, in $\mathbb{R}_{\max}$, the difficulty dwells in the fact that a triangular system of linear equations with nonzero diagonal elements may have no solutions.

Example 3.49 The system $(x_1 = 1, x_1 \oplus x_2 = e)$ has no solutions in $\mathbb{R}_{\max}$. ■

3.3.2.3 Decomposition of a Rational Function into Simple Elements

Definition 3.50 A proper rational function $\hat{r}/\hat{q}$ is a rational function which satisfies $\deg(\hat{r}) < \deg(\hat{q})$.

In general, it is not possible to express a rational function $\hat{p}/\hat{q}$ as $\hat{s} \oplus \hat{r}/\hat{q}$, where $\hat{r}/\hat{q}$ is proper and $\hat{s}$ is a polynomial function. Nevertheless, given a proper rational function, we may attempt to decompose it into simple elements.

Definition 3.51 A proper rational function $\hat{r}$ is decomposable into simple elements if it can be written as

$$\hat{r} = \bigoplus_{i=1}^n \bigoplus_{k=1}^{K_i} a_{ik} \not/c \oplus c_i)^k,$$

where the a_{ik} are constants.

Such a decomposition is not always possible.

Example 3.52 We first consider a rational function for which the decomposition into simple elements is possible:

$$\frac{c \oplus 1}{(c \oplus e)^2} = \frac{c \oplus e}{(c \oplus e)^2} \oplus \frac{1}{(c \oplus e)^2} = \frac{e}{c \oplus e} \oplus \frac{1}{(c \oplus e)^2}.$$

The rational function $(c \oplus e)/c \oplus 1)^2$, however, cannot be decomposed. Indeed, if such a decomposition exists, we would have

$$\frac{c \oplus e}{(c \oplus 1)^2} = \frac{a}{c \oplus 1} \oplus \frac{b}{(c \oplus 1)^2}.$$

Then $a(c \oplus 1) \oplus b = c \oplus e$, hence $a = e$, and also $a1 \oplus b = 1 \oplus b = e$, which is impossible. ■

The graph of a proper rational function which can be decomposed into simple elements is necessarily nonincreasing because it is the upper hull of nonincreasing functions. But a rational function with the degree of the numerator lower than the degree of the denominator is not always nonincreasing: this depends on the relative ordering of its pole and zero corners.

Example 3.53 The function $y = 2(c \oplus e)/c \oplus 1)^2$ is proper but not monotonic (see Figure 3.10). ■

However, being nonincreasing is a necessary but not a sufficient condition to be decomposable into simple elements as shown by the following example.

Example 3.54 The function $\hat{r}(c) = e/c^2 \oplus c$, the graph of which is displayed in Figure 3.11, cannot be decomposed into simple elements. Indeed,

$$\left\{ \hat{r}(c) = \frac{a}{c} \oplus \frac{b}{c \oplus e} \right\} \Rightarrow \{a(c \oplus e) \oplus bc = e\} \Rightarrow \left\{ \begin{array}{l} a \oplus b = e \\ a = e \end{array} \right\},$$

which is impossible in $\mathbb{R}_{\max}$. ■

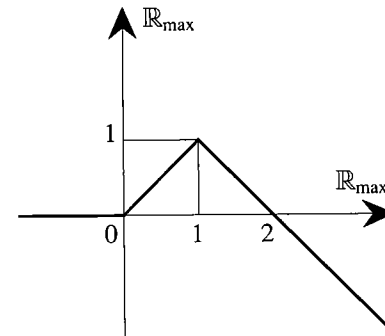


Figure 3.10: Graph of $y = 2(c \oplus e)/c \oplus 1)^2$

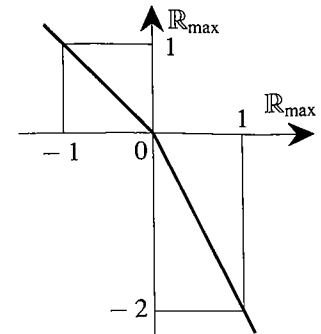


Figure 3.11: Graph of $y = e/c^2 \oplus c$

Theorem 3.55 A proper rational function has a simple element decomposition if and only if it has general root corners and special pole corners which are at the intersection of a zero-sloped line and a negative-integer-sloped line (Figure 3.12).

Proof If $\hat{r}$ is decomposable, it can be written as $\hat{r} = \bigoplus r_i$ with

$$\hat{r}_i(c) = \bigoplus_{k=1}^{K_i} a_{ik} \not/c \oplus c_i)^k.$$

Reducing the right-hand side to a common denominator $(c \oplus c_i)^{K_i}$, we obtain $\hat{r}_i = \hat{p}(c \oplus c_i)/c \oplus c_i)^{K_i}$, where $\hat{p}$ is a polynomial function of degree $K_i - 1$. The polynomial function $\hat{p}$ is characterized by the fact that the abscissæ of its corners are greater than c_i . Therefore $\hat{r}_i(c)$ is constant on the left-hand side of c_i , has a pole corner of order K_i at c_i , and a root corner on the right-hand side of c_i . Conversely, a function having this shape can easily be realized by an $\hat{r}_i$. The proof is completed by considering the fact that $\hat{r}$ is the supremum of a finite number of such $\hat{r}_i$. ■

3.3.3 Algebraic Equations

Definition 3.56 (Polynomial equation) Given two polynomial functions $\hat{p}$ and $\hat{q}$ of degree n and m , respectively, the equality $\hat{p}(c) = \hat{q}(c)$ is called a polynomial equation. Solutions to this equation are called roots. The degree of the polynomial equation is the integer $\max(n, m)$.

Some polynomial equations have roots, some do not. For example, $c^n = a$ has the root $c = \sqrt[n]{a}$, that is, $c = a/n$ in conventional algebra. On the other hand, the equation $\hat{p}(c) = \varepsilon$ has no roots when p is a general polynomial of degree $n \geq 1$.

3.3.3.1 Canonical Form of an Equation

Before studying equations, it is useful to write them in their simplest form. An equation $\hat{p}(c) = \hat{q}(c)$ can generally be simplified even if it is in the form $\hat{p}^b(c) = \hat{q}^b(c)$.

Indeed, if two monomials $p^b(k)\gamma^k$ and $q^b(k)\gamma^k$ of the same degree appear simultaneously, and if $p^b(k) < q^b(k)$, we can further simplify the equation by canceling the monomial $p^b(k)\gamma^k$.

Example 3.57 The equation $c^2 \oplus 3c \oplus 2 = 3c^2 \oplus 2c \oplus e$ can be reduced to $3c \oplus 2 = 3c^2$ (see Figure 3.13). ■

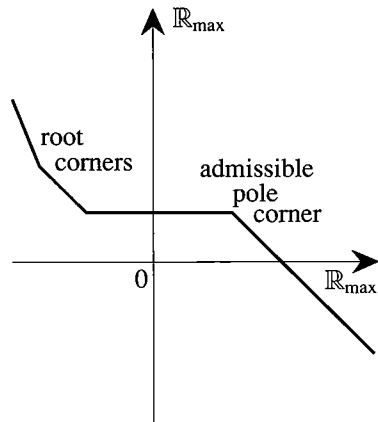


Figure 3.12: A rational function decomposable into simple elements

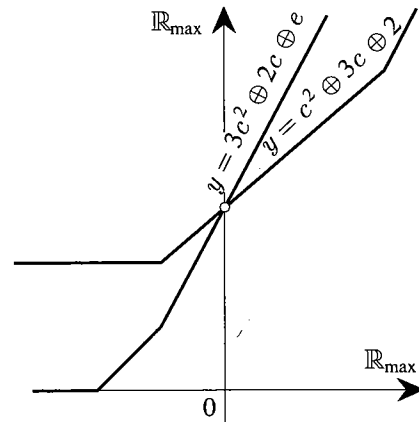


Figure 3.13: Equation $c^2 \oplus 3c \oplus 2 = 3c^2 \oplus 2c \oplus e$

Definition 3.58 If there exist two identical monomials on both sides of the equation, we say that the equation is degenerated. The nondegenerated equation $\hat{p}(c) = \hat{q}(c)$ is in minimal canonical form if $(p \oplus q)^b = p \oplus q$.

In the case of a degenerated equation, there is a segment of solutions.

3.3.3.2 Solving Polynomial Equations

Theorem 3.59 Any root of the nondegenerated equation $\hat{p}(c) = \hat{q}(c)$ is a corner of $\widehat{p \oplus q}$.

Proof Let us take the canonical form of the equation. Any root of $\hat{p}(c) = \hat{q}(c)$ is the solution of $p(k)c^k = q(l)c^l$ for some different k and l and thus it is a corner of $\widehat{p \oplus q}$ because the equation is in minimal canonical form. ■

The converse of this theorem is not true, i.e. a corner is not always a root of the equation $\hat{p}(c) = \hat{q}(c)$.

Example 3.60 The polynomial equation $3c \oplus 2 = 3c^2$ has the corner $c = -1$ which is not a root ($3(-1) \oplus 2 = 2$, $3(-1)^2 = 1$). ■

Now we give a characterization of the situation where the polynomial equation of degree n has exactly n roots.

Theorem 3.61 Suppose that n is even (the case when n is odd is similar). Let

$$\hat{p}(c) = p(0) \oplus p(2)c^2 \oplus \cdots \oplus p(2k)c^{2k},$$

$$\hat{q}(c) = p(1)c \oplus p(3)c^3 \oplus \cdots \oplus p(2k-1)c^{2k-1},$$

and suppose that $\hat{p}(c) = \hat{q}(c)$ is a nondegenerated equation in canonical form, then $c_i \stackrel{\text{def}}{=} p(i-1) \not\leq p(i)$, $i = 1, \dots, n$, are the n roots of the equation.

Proof No corner of $\widehat{p \oplus q}$ is multiple. Hence each one is obtained as the intersection of two monomials of consecutive degrees. Therefore, these monomials belong to different sides of the equation. Thus the corresponding corners are roots of the equation.

Conversely if the equation has n roots, $\widehat{p \oplus q}$ has n corners which therefore are distinct. But each of these corners is a root, hence these corners are characterized by the intersection of two monomial functions of consecutive degrees, the monomials being in different sides of the equation. ■

3.4 SYMMETRIZATION OF THE MAX-PLUS ALGEBRA

We have seen that the theory of linear system of equations in the max-plus algebra is not satisfactory at all, not even in the scalar case. We now extend $\mathbb{R}_{\max}$ to a larger set $\mathbb{S}$ for which $\mathbb{R}_{\max}$ can be viewed as the positive part. The construction is similar to the construction of $\mathbb{Z}$ as an extension of $\mathbb{N}$ in conventional algebra, but we cannot avoid some complications coming from the idempotency of the max operator. With this new set, we can generalize the notion of an equation that we call a balance. Then all nondegenerated scalar linear balances have a unique solution. Thus a linear closure of $\mathbb{R}_{\max}$ has been achieved.

3.4.1 The Algebraic Structure $\mathbb{S}$

A natural approach to our problem is to embed the max-plus algebra into a structure in which every nontrivial scalar linear equation has at least one solution. In particular we would like to have a solution to the equation $a \oplus x = \varepsilon$, that is, we would like to find a symmetric element to a . But this not possible, as shown by the following theorem.

Theorem 3.62 Every idempotent group is reduced to the zero element.

Proof Assume that the group $(G, \oplus)$ is idempotent with zero element ε . Let b be the symmetric element of $a \in G$. Then

$$a = a \oplus \varepsilon = a \oplus (a \oplus b) = (a \oplus a) \oplus b = a \oplus b = \varepsilon.$$

Nevertheless we can adapt the idea of the construction of $\mathbb{Z}$ from $\mathbb{N}$ to build a 'balancing' element rather than a symmetric one. This is the purpose of the following subsections.

3.4.1.1 The Algebra of Pairs

Let us consider the set of pairs $\mathbb{R}_{\max}^2$ endowed with the natural idempotent semifield structure:

$$\begin{aligned}(x', x'') \oplus (y', y'') &= (x' \oplus y', x'' \oplus y'') , \\ (x', x'') \otimes (y', y'') &= (x' y' \oplus x'' y'', x' y' \oplus x'' y'') ,\end{aligned}$$

with $(\varepsilon, \varepsilon)$ as the zero element and (e, e) as the identity element.

Let $x = (x', x'')$ and define the *minus* sign as $\ominus x = (x'', x')$, the *absolute value* of x as $|x| = x' \oplus x''$ and the *balance operator* as $x^\bullet = x \ominus x = (|x|, |x|)$. Clearly, these operators have the following properties:

1. $a^\bullet = (\ominus a)^\bullet$;
2. $a^{\bullet\bullet} = a^\bullet$;
3. $ab^\bullet = (ab)^\bullet$;
4. $\ominus(\ominus a) = a$;
5. $\ominus(a \oplus b) = (\ominus a) \oplus (\ominus b)$;
6. $\ominus(a \otimes b) = (\ominus a) \otimes b$.

These properties allow us to write $a \oplus (\ominus b) = a \ominus b$ as usual.

3.4.1.2 Quotient Structure

Definition 3.63 (Balance relation) Let $x = (x', x'')$ and $y = (y', y'')$. We say that x *balances* y (which is denoted $x \nabla y$) if $x' \oplus y'' = x'' \oplus y'$.

It is fundamental to notice that ∇ is *not* transitive and thus is not an equivalence relation. For instance, consider $(e, 1) \nabla (1, 1)$, $(1, 1) \nabla (1, e)$, but $(e, 1) \not\nabla (1, e)$! Since ∇ cannot be an equivalence relation, it is not possible to define the quotient structure of $\mathbb{R}_{\max}^2$ by means of ∇ (unlike in conventional algebra in which $\mathbb{N}^2 / \nabla \simeq \mathbb{Z}$). However, we can introduce the equivalence relation $\mathcal{R}$ on $\mathbb{R}_{\max}^2$ closely related to the balance relation, namely,

$$(x', x'') \mathcal{R} (y', y'') \Leftrightarrow \begin{cases} x' \oplus y'' = x'' \oplus y' & \text{if } x' \neq x'', y' \neq y'' , \\ (x', x'') = (y', y'') & \text{otherwise.} \end{cases}$$

It is easy to check that $\mathcal{R}$ is compatible with the addition and multiplication of $\mathbb{R}_{\max}^2$, the balance relation ∇ , and the $\ominus$, $|\cdot|$ and $^\bullet$ operators.

Definition 3.64 The symmetrized algebra $\mathbb{R}_{\max}^2 / \mathcal{R}$ of $\mathbb{R}_{\max}$ is called $\mathbb{S}$.

We distinguish three kinds of equivalence classes:

$$\begin{aligned}\overline{(t, -\infty)} &= \{(t, x'') \mid x'' < t\} , & \text{called positive elements;} \\ \overline{(-\infty, t)} &= \{(x', t) \mid x' < t\} , & \text{called negative elements;} \\ \overline{(t, t)} &= \{(t, t)\} , & \text{called balanced elements.}\end{aligned}$$

By associating $\overline{(t, -\infty)}$ with $t \in \mathbb{R}_{\max}$, we can identify $\mathbb{R}_{\max}$ with the semifield of positive or zero classes denoted $\mathbb{S}^\oplus$. The set of negative or zero classes (of the form $\ominus x$ for $x \in \mathbb{S}^\oplus$) will be denoted $\mathbb{S}^\ominus$. This set is not stable by multiplication and thus it is not a semifield. The set of balanced classes (of the form $x^\bullet$) is denoted $\mathbb{S}^\bullet$; it is also isomorphic to $\mathbb{R}_{\max}$. This yields the decomposition

$$\mathbb{S} = \mathbb{S}^\oplus \cup \mathbb{S}^\ominus \cup \mathbb{S}^\bullet . \quad (3.15)$$

The element ε is the *only* element common to $\mathbb{S}^\oplus$ and $\mathbb{S}^\ominus$ and $\mathbb{S}^\bullet$. This decomposition of $\mathbb{S}$ should be compared with $\mathbb{Z} = \mathbb{N}^+ \cup \mathbb{N}^-$. This notation allows us to write $3 \ominus 2$ instead of $\overline{(3, -\infty)} \oplus \overline{(-\infty, 2)}$. We thus have $3 \ominus 2 = \overline{(3, 2)} = \overline{(3, -\infty)} = 3$. More generally, calculations in $\mathbb{S}$ can be summarized as follows:

$$\begin{aligned}a \ominus b &= a , & \text{if } a > b ; \\ b \ominus a &= \ominus a , & \text{if } a > b ; \\ a \ominus a &= a^\bullet .\end{aligned} \quad (3.16)$$

Because of its importance, we introduce the notation $\mathbb{S}^\vee$ for the set $\mathbb{S}^\oplus \cup \mathbb{S}^\ominus$ and $\mathbb{S}_*^\vee = \mathbb{S}^\vee \setminus \{\varepsilon\}$. The elements of $\mathbb{S}^\vee$ are called *signed* elements. They are either positive, negative or zero.

Theorem 3.65 The set $\mathbb{S}_*^\vee = \mathbb{S} \setminus \mathbb{S}^\bullet$ is the set of all invertible elements of $\mathbb{S}$.

Proof The obvious identity $t \otimes (-t) = (\ominus t) \otimes (\ominus -t) = e$ for $t \in \mathbb{R}_{\max} \setminus \{\varepsilon\}$ implies that every nonzero element of $\mathbb{S}^\vee$ is invertible. Moreover, the absorbing properties of the balance operator show that $\mathbb{S}^\vee$ is absorbing for the product. Thus, $x^\bullet y \neq e$ for all $y \in \mathbb{S}$ since $e \notin \mathbb{S}^\bullet$. ■

Remark 3.66 Thus, in $\mathbb{S}$, with each element $a \in \mathbb{S}^\vee$, we can associate an element $\ominus a$ such that $b = a \ominus a \in \mathbb{S}^\bullet$ but in general $b \neq \varepsilon$. This is the main difference with the usual symmetrization. Here the whole set $\mathbb{S}^\bullet$ plays the role of the usual zero element. ■

3.4.2 Linear Balances

Before solving general linear balances, we need to understand the meaning of the generalization of equations in $\mathbb{R}_{\max}$ by balances in $\mathbb{S}$. This can be done by studying the properties of balances.

Theorem 3.67 The relation ∇ satisfies the following properties:

1. $a \nabla a$;

2. $a \nabla b \Leftrightarrow b \nabla a$;
3. $a \nabla b \Leftrightarrow a \oplus b \nabla \varepsilon$;
4. $\{a \nabla b, c \nabla d\} \Rightarrow a \oplus c \nabla b \oplus d$;
5. $a \nabla b \Rightarrow ac \nabla bc$.

Proof Let us prove Property 5. Obviously, $a \nabla b \Leftrightarrow a \oplus b \in \mathbb{S}^\bullet$ and, since $\mathbb{S}^\bullet$ is absorbing, $(a \oplus b)c = ac \oplus bc \in \mathbb{S}^\bullet$, i.e. $ac \nabla bc$. ■

Although ∇ is not transitive, when some variables are signed, we can manipulate balances in the same way as we manipulate equations.

Theorem 3.68

1. **Weak substitution** If $x \nabla a$, $cx \nabla b$ and $x \in \mathbb{S}^\vee$, we have $ca \nabla b$.
2. **Weak transitivity** If $a \nabla x$, $x \nabla b$ and $x \in \mathbb{S}^\vee$, we have $a \nabla b$.
3. **Reduction of balances** If $x \nabla y$ and $x, y \in \mathbb{S}^\vee$, we have $x = y$.

Proof

1. We have either $x \in \mathbb{S}^\oplus$ or $x \in \mathbb{S}^\ominus$. Assume for instance that $x \in \mathbb{S}^\oplus$, that is, $x = \overline{(x', \varepsilon)}$. With the usual notation, $x' \oplus a'' = a'$ and $c'x' \oplus b'' = c''x' \oplus b'$. Adding $c'a'' \oplus c''a''$ to the last equality, we get

$$c'x' \oplus c'a'' \oplus c''a'' \oplus b'' = c''x' \oplus c'a'' \oplus c''a'' \oplus b' ,$$

which yields, by using $x' \oplus a'' = a'$,

$$c'a' \oplus c''a'' \oplus b'' = c''a' \oplus c'a'' \oplus b' ,$$

that is, $ca \nabla b$.

2. This is a consequence of the weak substitution for $c = e$.
3. This point is trivial but is important in order to derive equalities from balances. ■

The introduction of these new notions is justified by the fact that any linear balance (which is not degenerated) has one and only one solution in $\mathbb{S}^\vee$.

Theorem 3.69 Let $a \in \mathbb{S}_*^\vee$ and $b \in \mathbb{S}^\vee$, then $x = \ominus a^{-1}b$ is the unique solution of the balance

$$ax \oplus b \nabla \varepsilon , \quad (3.17)$$

which belongs to $\mathbb{S}^\vee$.

Proof From the properties of balances it follows that $ax \oplus b \nabla \varepsilon \Leftrightarrow x \nabla \ominus a^{-1}b$. Then using the reduction property and the fact that $\ominus a^{-1}b \in \mathbb{S}^\vee$, we obtain $x = \ominus a^{-1}b$. ■

Remark 3.70 If $b \notin \mathbb{S}^\vee$, we lose the uniqueness of signed solutions. Every x such that $|ax| \leq |b|$ (i.e. $|x| \leq |a^{-1}b|$) is a solution of the balance (3.17). If $a \notin \mathbb{S}^\vee$, we again lose uniqueness. Assume $b \in \mathbb{S}^\vee$ (otherwise, the balance holds for all values of x), then every x such that $|ax| \geq |b|$ is a solution. ■

Remark 3.71 We can describe all the solutions of (3.17). For all $t \in \mathbb{R}_{\max}$, we obviously have $at^\bullet \nabla \varepsilon$. Adding this balance to $ax \oplus b \nabla \varepsilon$, where x is the unique signed solution, we obtain $a(x \oplus t^\bullet) \oplus b \nabla \varepsilon$. Thus,

$$x_t = x \oplus t^\bullet \quad (3.18)$$

is a solution of (3.17). If $t \geq |x|$, then $x_t = t^\bullet$ is balanced. Conversely, it can be checked that every solution of (3.17) may be written as in (3.18). Finally the unique signed solution x is also the least solution. ■

Remark 3.72 Nontrivial linear balances (with data in $\mathbb{S}^\vee$) always have solutions in $\mathbb{S}$; this is why $\mathbb{S}$ may be considered as a linear closure of $\mathbb{R}_{\max}$. ■

3.5 LINEAR SYSTEMS IN $\mathbb{S}$

It is straightforward to extend balances to the vector case. Theorems 3.67 and 3.68 still hold when a, b, x, y and c are matrices with appropriate dimensions, provided we replace 'belongs to $\mathbb{S}^\vee$ ' by 'every entry belongs to $\mathbb{S}^\vee$ '. Therefore, we say that a vector or a matrix is *signed* if all its entries are signed.

We now consider a solution $x \in \mathbb{R}_{\max}$ of the equation

$$Ax \oplus b = Cx \oplus d . \quad (3.19)$$

Then the definition of the balance relation implies that

$$(A \ominus C)x \oplus (b \ominus d) \nabla \varepsilon . \quad (3.20)$$

Conversely, assuming that x is a positive solution of (3.20), we obtain

$$Ax \oplus b \nabla Cx \oplus d ,$$

with $Ax \oplus b$ and $Cx \oplus d \in \mathbb{S}^\oplus$. Using Theorem 3.68, we obtain

$$Ax \oplus b = Cx \oplus d .$$

Therefore we have the following theorem.

Theorem 3.73 The set of solutions of the general system of linear equations (3.19) in $\mathbb{R}_{\max}$ and the set of positive solutions of the associated linear balance (3.20) in $\mathbb{S}$ coincide.

Hence, studying Equation (3.19) is reduced to solving linear balances in $\mathbb{S}$.

Remark 3.74 The case when a solution x of (3.20) has some negative and some positive entries is also of interest. We write $x = x^+ \oplus x^-$ with $x^+, x^- \in (\mathbb{S}^\oplus)^n$. Partitioning the columns of A and C according to the sign of the entries of x , we obtain $A = A^+ \oplus A^-$, $C = C^+ \oplus C^-$, so that $Ax = A^+x^+ \oplus A^-x^-$ and $Cx = C^+x^+ \oplus C^-x^-$. We can thus claim the existence of a solution in $\mathbb{R}_{\max}$ to the new problem

$$A^+x^+ \oplus C^-x^- \oplus b = A^-x^- \oplus C^+x^+ \oplus d.$$

The solution of nondegenerated problems is not unique, but the set of solutions forms a single class of $\mathbb{R}_{\max}^2$ (for the equivalence relation $\mathcal{R}$). ■

3.5.1 Determinant

Before dealing with general systems, we need to extend the determinant machinery to the $\mathbb{S}$ -context. We define the signature of a permutation σ by

$$\text{sgn}(\sigma) = \begin{cases} e & \text{if } \sigma \text{ is even;} \\ \ominus e & \text{otherwise.} \end{cases}$$

Then the determinant of an $n \times n$ matrix $A = (A_{ij})$ is given (as usual) by

$$\bigoplus_{\sigma} \text{sgn}(\sigma) \bigotimes_{i=1}^n A_{i\sigma(i)},$$

and is denoted either $|A|$ or $\det(A)$. The transpose of the matrix of cofactors is denoted $A^{\natural}$ ($(A^{\natural})_{ij} \stackrel{\text{def}}{=} \text{cof}_{ji}(A)$). The classical properties of the determinant are still true.

Theorem 3.75 *The determinant has the following properties:*

linearity:

$$|(u_1, \dots, \lambda u_i \oplus \mu v_i, \dots, u_n)| = \lambda |(u_1, \dots, u_i, \dots, u_n)| \oplus \mu |(u_1, \dots, v_i, \dots, u_n)|;$$

antisymmetry:

$$|(u_{\sigma(1)}, \dots, u_{\sigma(n)})| = \text{sgn}(\sigma) |(u_1, \dots, u_n)|;$$

and consequently

$$|(u_1, \dots, v, \dots, v, \dots, u_n)| \nabla \varepsilon;$$

expansion with respect to a row:

$$|A| = \bigoplus_{k=1}^n a_{ik} \text{cof}_{ik}(A);$$

transposition: $|A| = |A'|$.

A direct consequence is that some classical proofs lead to classical identities in this new setting. Sometimes weak substitution limits the scope of this approach.

Theorem 3.76 *For an $n \times n$ matrix A with entries in $\mathbb{S}$, we have*

Cramer formula: $AA^{\natural} \nabla |A|e$, and if $|A|$ is signed, then the diagonal of $AA^{\natural}$ is signed;

recursive computation of the determinant:

$$|A| = \left| \begin{pmatrix} F & G \\ H & a_{nn} \end{pmatrix} \right| = |F|_{a_{nn}} \ominus HF^{\natural}G$$

for a partition of matrix A where a_{nn} is a scalar;

Cayley-Hamilton theorem: p being the characteristic polynomial of matrix A , i.e.

$$p(\lambda) \stackrel{\text{def}}{=} |A \ominus \lambda e|, \text{ we have } p(A) \nabla \varepsilon.$$

Remark 3.77 We define the positive determinant of a matrix A , denoted $|A|^+$, by the sum of terms $\bigotimes_i A_{i\sigma(i)}$ where the sum is limited to even permutations, and a negative determinant, denoted $|A|^-$, by the same sum limited to odd permutations. The matrix of positive cofactors is defined by

$$(A^{\natural+})_{ij} = \begin{cases} |A^{ji}|^+ & \text{if } i+j \text{ is even,} \\ |A^{ji}|^- & \text{if } i+j \text{ is odd,} \end{cases}$$

where A^{ji} denotes the matrix derived from A by deleting row j and column i . The matrix of negative cofactors $A^{\natural-}$ is defined similarly. With this notation, Theorem 3.76 can be rewritten as follows:

$$AA^{\natural+} \oplus |A|^- e = AA^{\natural-} \oplus |A|^+ e.$$

This formula does not use the $\ominus$ sign and is valid in any semiring. The symmetrized algebra appears to be a natural way of handling such identities (and giving proofs in an algebraic way). ■

3.5.2 Solving Systems of Linear Balances by the Cramer Rule

In this subsection we study solutions of systems of linear equations with entries in $\mathbb{S}$. We only consider the solutions belonging to $(\mathbb{S}^\vee)^n$, that is, we only consider signed solutions. Indeed, in a more general setting we cannot hope to have a result of uniqueness; see Remark 3.70. We can now state the fundamental result for the existence and uniqueness of signed solutions of linear systems.

Theorem 3.78 (Cramer system) *Let A be an $n \times n$ matrix with entries in $\mathbb{S}$, $|A| \in \mathbb{S}_*^\vee$, $b \in \mathbb{S}^n$ and $A^{\natural}b \in (\mathbb{S}^\vee)^n$. Then, in $\mathbb{S}^\vee$ there exists a unique solution of*

$$Ax \nabla b, \tag{3.21}$$

and it satisfies

$$x \nabla A^{\natural}b / |A|. \tag{3.22}$$

Proof By right-multiplying the identity $AA^{\flat} \nabla |A|e$ by $|A|^{-1}b$, we see that x is a solution. Let us now prove uniqueness. The proof is by induction on the size of the matrix. It is based on Gauss elimination in which we manage the balances using weak substitution. Let us prove (3.22) for the last row, i.e. $|A|x_n \nabla (A^{\flat}b)_n$. Developing $|A|$ with respect to the last column, $|A| = \bigoplus_{k=1}^n a_{kn} \text{cof}_{kn}(A)$, we see that at least one term is invertible, say $a_{1n} \text{cof}_{1n}(A)$. We now partition A , b and x in such a way that the scalar a_{1n} becomes a block:

$$A = \begin{pmatrix} H & a_{1n} \\ F & G \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ B \end{pmatrix}, \quad x = \begin{pmatrix} X \\ x_n \end{pmatrix}.$$

Then $Ax \nabla b$ can be written as

$$HX \oplus a_{1n}x_n \nabla b_1, \quad (3.23)$$

$$FX \oplus Gx_n \nabla B. \quad (3.24)$$

Since $|F| = (\ominus e)^{n+1} \text{cof}_{1n}(A)$ is invertible, we can apply the induction hypothesis to (3.24). This implies that

$$X \nabla |F|^{-1} F^{\flat} (B \ominus Gx_n).$$

Using the weak substitution property, we can replace $X \in (\mathbb{S}^{\vee})^{n-1}$ in Equation (3.23) to obtain

$$|F|^{-1} H F^{\flat} (B \ominus Gx_n) \oplus a_{1n}x_n \nabla b_1,$$

that is,

$$x_n (|F| a_{1n} \ominus H F^{\flat} G) \nabla |F| b_1 \ominus H F^{\flat} B.$$

Here we recognize the developments of $|A|$ and $(A^{\flat}b)_n$, therefore

$$x_n \nabla (A^{\flat}b)_n \not\neq |A|.$$

Since the same reasoning can be applied to the entries of x other than n , this concludes the proof. ■

Remark 3.79 Let us write D_i for the determinant of the matrix obtained by replacing the i -th column of A by the column vector b ; then $D_i = (A^{\flat}b)_i$. Assume that $D \stackrel{\text{def}}{=} |A|$ is invertible, then Equation (3.22) is equivalent to

$$x_i \nabla D^{-1} D_i, \quad \forall i.$$

If $A^{\flat}b \in (\mathbb{S}^{\vee})^n$, then by using the reduction of balances (see Theorem 3.68), we obtain

$$x_i = D^{-1} D_i,$$

which is exactly the classical Cramer formula. ■

Example 3.80 The $\mathbb{R}_{\max}$ equation

$$\begin{pmatrix} e & -4 \\ 3 & 2 \end{pmatrix} \otimes \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \oplus \begin{pmatrix} 1 \\ -5 \end{pmatrix} = \begin{pmatrix} -1 & 1 \\ \varepsilon & 2 \end{pmatrix} \otimes \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \oplus \begin{pmatrix} 2 \\ 7 \end{pmatrix}, \quad (3.25)$$

corresponds to the balance

$$\begin{pmatrix} e & \ominus 1 \\ 3 & 2 \bullet \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \nabla \begin{pmatrix} 2 \\ 7 \end{pmatrix}. \quad (3.26)$$

Its determinant is $D = 4$.

$$D_1 = \left| \begin{pmatrix} 2 & \ominus 1 \\ 7 & 2 \bullet \end{pmatrix} \right| = 8, \quad D_2 = \left| \begin{pmatrix} e & 2 \\ 3 & 7 \end{pmatrix} \right| = 7,$$

$$A^{\flat}b = \begin{pmatrix} D_1 \\ D_2 \end{pmatrix} = \begin{pmatrix} 8 \\ 7 \end{pmatrix} \in (\mathbb{S}^{\vee})^2.$$

The system is invertible and has a unique solution. Thus $(x_1 = D_1 \not\neq D = 8 - 4 = 4, x_2 = D_2 \not\neq D = 7 - 4 = 3)$ is the unique positive solution in $\mathbb{S}$ of the balance (3.26). Hence it is the unique solution in $\mathbb{R}_{\max}$ of Equation (3.25). ■

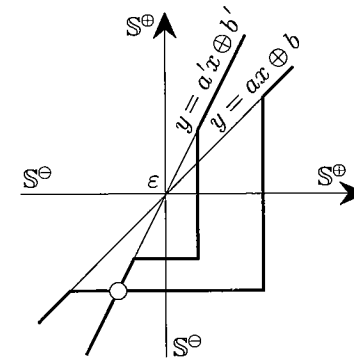


Figure 3.14: A (2, 2) linear system of equations

Example 3.81 In the two-dimensional case the condition $A^{\flat}b \in (\mathbb{S}^{\vee})^n$ has a very clear geometric interpretation (see Figure 3.14) as the intersection of straight lines in $\mathbb{S}$. First we can choose an exponential scaling of the x_1 and x_2 axes. The exponential maps $\mathbb{S}^{\oplus}$ to $\mathbb{R}^+$ and $\mathbb{S}^{\ominus}$ to $\mathbb{R}^-$, if we identify $\mathbb{S}^{\ominus}$ with $i\pi + \mathbb{R} \cup \{-\infty\}$. We do not represent the balance axis in this representation. Therefore the straight line $ax_2 \oplus bx_1 \oplus c \nabla \varepsilon$ is a broken line (in the usual sense) composed of four segments:

- two of them are symmetric with respect of the origin; they correspond to the contribution of $ax_2 \oplus bx_1$ to the balance (they belong to the conventional line $ax_2 + bx_1 = 0$);
- the horizontal segment corresponds to the contribution of $ax_2 \oplus c$ to the balance;
- the vertical segment corresponds to the contribution of $bx_1 \oplus c$ to the balance.

Then it is easy to see that two such lines have one and only one point of intersection, or there exists a complete segment of solutions. This latter case is called singular and is not further considered here. ■

Remark 3.82 The invertibility of $|A|$ is *not* a necessary condition for the existence of a signed solution to the system $Ax \nabla b$ for some value of b . Let us consider

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

Because $|A| = \varepsilon$, the matrix A is not invertible. Let $t \in \mathbb{S}^\vee$ be such that $|b_i| \leq |t|$ for all i , and let $x = (t \quad \ominus t \quad \varepsilon)'$. Then $Ax \nabla b$. But in this case, the signed solution is not unique. ■

Remark 3.83 As already noticed, in [65] for example, determinants have a natural interpretation in terms of assignment problems. So the Cramer calculations have the same complexity as $n + 1$ assignment problems, which can be solved using flow algorithms. ■

3.6 POLYNOMIALS WITH COEFFICIENTS IN $\mathbb{S}$

We have built the linear closure $\mathbb{S}$ of $\mathbb{R}_{\max}$. Therefore any linear equation in $\mathbb{S}$ has in general a solution in this set. The purpose of this section is to show that $\mathbb{S}$ is almost algebraically closed, that is, any 'closed' polynomial equation of degree n in $\mathbb{S}$ has n solutions. The term 'closed' refers to the fact that the class of formal polynomials having the same polynomial function defined on $\mathbb{S}$ is not always closed (in a topological sense that will be made precise later). We will see that $\mathbb{S}$ is algebraically closed only for the subset of 'closed' polynomial equations. Moreover, we will see that any polynomial function can be transformed into a closed function by modifying it in a finite number of points.

3.6.1 Some Polynomial Functions

We can generalize the notions of formal polynomials and of polynomial functions to the set $\mathbb{S}$. We restrict our study to polynomial functions in $\mathcal{P}(\mathbb{S}^\vee) \stackrel{\text{def}}{=} \mathcal{F}(\mathbb{S}^\vee[\gamma])$, where $\mathbb{S}^\vee[\gamma]$ denotes the class of formal polynomials with coefficients in $\mathbb{S}^\vee$ and $\mathcal{F}$ is the straightforward extension of the evaluation homomorphism introduced in Lemma 3.35. We will see that such functions assume values in $\mathbb{S}^\vee$ when their argument ranges in $\mathbb{S}^\vee$, except perhaps at a finite number of points. For the more general class of polynomial functions with coefficients in $\mathbb{S}$, denoted $\mathcal{P}(\mathbb{S})$, it may happen that the function takes a balanced value on a continuous set of points of $\mathbb{S}^\vee$ (in which case we say that we have a balanced facet). Because we are mainly interested in the analogy with conventional polynomials, we do not deal with this latter situation.

To get a better understanding of polynomial functions on $\mathbb{S}$ we study some particular cases. Let us start by plotting the graphs of polynomial functions of degree one, two and three (see Figure 3.15). We must study the graphs over each

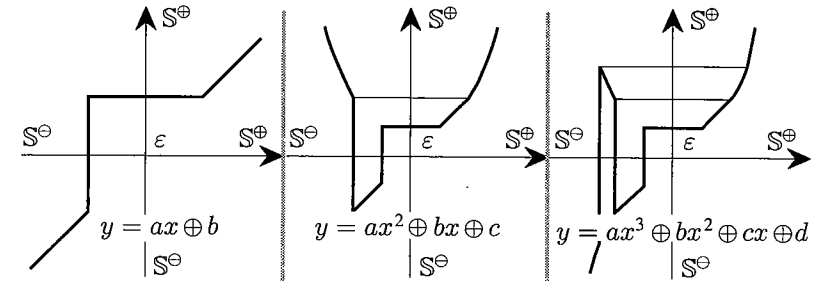


Figure 3.15: Polynomial functions of degree one, two and three

of the three components $\mathbb{S}^+$, $\mathbb{S}^-$ and $\mathbb{S}^*$. The value of the function itself, however, is an element of $\mathbb{S}$ which also belongs to one of these components. Therefore the plot is quite complicated. In order to simplify the study, only the plot over $\mathbb{S}^\vee$ is considered. Moreover we will use the exponential system of coordinates that we have discussed in the previous section. Figure 3.15 shows the points of discontinuity of the three component functions. These discontinuities always appear at abscissæ which are symmetric with respect to corner abscissæ. At these discontinuities the polynomial functions take balanced values (in the graph, we see sign changes). They correspond to what we call 'roots' of polynomial functions. If a polynomial is of degree n , it has in general n corners and n points in $\mathbb{S}^\vee$ where the polynomial function takes balanced values.

Let us study the polynomial functions of degree two in more detail. We consider

$$\hat{p}(c) = p(0) \oplus p(1)c \oplus p(2)c^2, \quad p(i) \in \mathbb{S}^\vee, \quad i = 0, 1, 2.$$

The polynomial function $|\hat{p}| \in \mathcal{P}(\mathbb{R}_{\max})$ is defined as $|\hat{p}|(c) = |p(0)| \oplus |p(1)|c \oplus |p(2)|c^2$. We will later prove, but it should be clear, that if c is a root of $\hat{p}$, then $|c|$ is a corner of $|p|$. Owing to Theorem 3.43, $|\hat{p}|$ can be factored into the product of two linear polynomials. We have the following four possibilities:

1. If $|p(1)| \not\leq |p(2)|$, then $|\hat{p}|$ has two distinct corners $c_1 = |p(1)| \not\leq |p(2)|$ and $c_2 = |p(0)| \not\leq |p(1)|$. It can be checked that $\hat{p} = p(2)(c \oplus |p(1)| \not\leq |p(2)|)(c \oplus |p(0)| \not\leq |p(1)|)$ and that this factorization is unique. In addition, the moduli of the roots are c_1 and c_2 .
2. If $|p(1)| \leq |p(2)|$, then $|\hat{p}|$ has a corner $c_1 = |p(0)| \not\leq |p(2)|^{1/2}$ of multiplicity 2 and the roots of $\hat{p} = p(2)(c \oplus |p(1)| \not\leq |p(2)|)(c \oplus |p(0)| \not\leq |p(1)|)$ have modulus c_1 .
3. If $|p(1)| \leq |p(2)|$ and $p(2)p(0) \in \mathbb{S}^+$, then $|\hat{p}|$ has a corner $c_1 = |p(0)| \not\leq |p(2)|^{1/2}$ of multiplicity 2 and $\hat{p}$ cannot be factored. Indeed at c_1 and $\ominus c_1$, $\hat{p}$ has signed values. We have $\hat{p} = p(2)(c^2 \oplus |p(0)| \not\leq |p(2)|)$.

4. If $|p(1)p(2)| < |p(0)p(1)|$ and $p(2)p(0) \in \mathbb{S}^\ominus$, then the polynomial $|\widehat{p}|$ has a corner $c_1 = |p(0)p(2)|^{1/2}$ of multiplicity 2. We have $\widehat{p} = p(2)(c \oplus c_1)(c \ominus c_1)$ and thus $\widehat{p}$ has been factored.

This discussion suggests that if the corners of $|\widehat{p}|$ (now a polynomial function of degree n) are distinct, then there are n roots; but if $|\widehat{p}|$ has multiple corners, then we are not guaranteed to have a factorization in linear factors. We now study the situation in more detail.

3.6.2 Factorization of Polynomial Functions

We can consider formal polynomials $\mathbb{S}[\gamma]$ and polynomial functions $\mathcal{P}(\mathbb{S})$ with coefficients in $\mathbb{S}$ as we have done for polynomials with coefficients in $\mathbb{R}_{\max}$. They define algebras. The following mapping from $\mathcal{P}(\mathbb{S})$ into $\mathcal{P}(\mathbb{R}_{\max})$

$$p = p(k)\gamma^k \oplus \dots \oplus p(n)\gamma^n \mapsto |p| = |p(k)|\gamma^k \oplus \dots \oplus |p(n)|\gamma^n$$

is a surjective morphism.

Definition 3.84 (Root) The root of a polynomial function $\widehat{p} \in \mathcal{P}(\mathbb{S})$ is an element $c \in \mathbb{S}^\vee$ such that $\widehat{p}(c) \nabla \varepsilon$.

Remark 3.85 It should be clear that the computation of the linear factors of a polynomial yields its roots. Indeed we have

$$\bigotimes_i (c \ominus c_i) \nabla \varepsilon \Leftrightarrow c \ominus c_i \nabla \varepsilon .$$

Lemma 3.86 If, for a polynomial $\widehat{p} \in \mathcal{P}(\mathbb{S}^\vee)$, c is a root of $\widehat{p}(c)$, then $|c|$ is a corner of $|\widehat{p}|$.

Proof If c is a root, then $\widehat{p}(c)$ is balanced and hence $|\widehat{p}(c)| = |\widehat{p}(|c|)|$ is a quantity which is achieved by two distinct monomials of $\widehat{p}$ since, by assumption, the coefficients of $\widehat{p}$ belong to $\mathbb{S}^\vee$. Thus $|c|$ is a corner of $|\widehat{p}|$. ■

For the same reason as in the $\mathbb{R}_{\max}$ case, the mapping

$$\mathcal{F} : \mathbb{S}[\gamma] \rightarrow \mathcal{P}(\mathbb{S}) , \quad p \mapsto \widehat{p} ,$$

is not injective. In order to study the set valued-inverse mapping $\mathcal{F}^{-1}$, we introduce the notion of a closed polynomial function.

Definition 3.87 (Closed polynomial function) We say that the polynomial function $\widehat{p} \in \mathcal{P}(\mathbb{S}^\vee)$ is closed if $\mathcal{F}^{-1}(\widehat{p})$ admits a maximum element denoted $p^\sharp \in \mathcal{P}(\mathbb{S})$. This element is called the maximum representative of $\widehat{p}$.

Example 3.88 The polynomial function $c^2 \ominus e$ is closed because it has the same graph as $c^2 \oplus e^\bullet c \ominus e$. The polynomial function $c^2 \oplus e$ is not closed because its graph is different from that of any polynomial $\widehat{p}_a = c^2 \oplus ac \oplus e$ with $a \in \{e, \ominus e, e^\bullet\}$ although it is the same for all $a < e$. ■

The notion of closed polynomial function is relevant because the inverse of the evaluation homomorphism is simple for this class, and because this class is exactly the set of polynomial functions which can be factored into linear factors.

Theorem 3.89 A polynomial function $\widehat{p} \in \mathcal{P}(\mathbb{S}^\vee)$ can be factored in linear factors if and only if it is closed.

Proof

Any polynomial which can be factored is closed. Let us suppose that the degree of $\widehat{p}$ is n , its valuation 0, and the coefficient of its head monomial is e (the proof can be generalized to avoid these assumptions). Let us suppose that the degree of $\widehat{p}$ is n , its valuation 0, and the coefficient of its head monomial is e (the proof can be adapted to the general case). Let $c_i, i = 1, \dots, n$, denote the roots of $\widehat{p}$ numbered according to the decreasing order of their modulus. If the c_i are all distinct, we have

$$\widehat{p}(c) = \bigoplus_{i=0}^n m_i \quad \text{with} \quad m_i \stackrel{\text{def}}{=} \left(\bigotimes_{j=1}^i c_j \right) c^{n-i} .$$

Because

$$m_i > m_j , \quad \forall j > i , \quad \forall c : c_{i+1} < c < c_i ,$$

we cannot increase the coefficient of a monomial without changing the graph of $\widehat{p}$ and therefore it is closed. The situation is more complicated when there is a multiple root because then at least three monomials take the same value in modulus at this root. To understand what happens at such a multiple root, let us consider the case when $\widehat{p}$ has only the roots e or $\ominus e$, that is, $\widehat{p}(c) = (c \oplus e)^{n-m} (c \ominus e)^m$. The expansion of this polynomial gives five kinds of polynomials:

- $c^n \oplus c^{n-1} \oplus \dots \oplus e$;
- $c^n \ominus c^{n-1} \oplus c^{n-2} \dots \oplus e$;
- $c^n \ominus c^{n-1} \oplus c^{n-2} \dots \ominus e$;
- $c^n \oplus e^\bullet c^{n-1} \oplus e^\bullet c^{n-2} \dots \oplus e$;
- $c^n \oplus e^\bullet c^{n-1} \oplus e^\bullet c^{n-2} \dots \ominus e$.

By inspection, we verify that we cannot increase any coefficient of these polynomials without changing their graphs. Therefore, they are closed.

We remark that some polynomials considered in this enumeration do not belong to $\mathcal{P}(\mathbb{S}^\vee)$. For example, it is the case of $(c \ominus e)^2 (c \oplus e)$. Some other polynomials

have their coefficients in $\mathbb{S}^\bullet$ and do not seem to belong to the class that we study here, but they have other representatives in $\mathcal{P}(\mathbb{S}^\vee)$. For example, we have

$$(c \ominus e)(c \oplus e) = c^2 \oplus e^\bullet c \ominus e = c^2 \ominus e.$$

Any closed polynomial can be factored. If $\widehat{p}$ is closed, let p denote its maximum representative. Its coefficients $c_i = p(n-i)p(n-i+1)$ are nonincreasing with i in modulus. Indeed, if that were not the case, there would exist i and k such that $|c_{i-k}| > |c_{i+1}| > |c_i|$. Then it would be possible to increase p_{n-i} while preserving the inequalities $|c_{i-k}| > |c_{i+1}| > |c_i|$. Because this operation would not change the graph of $\widehat{p}$, we would have contradicted the maximality of p .

Now, if the $|c_i|$ are strictly decreasing with i , we directly verify by expansion that $\widehat{p}(c) = \bigotimes_i (c \oplus c_i)$. If the c_i are simply nonincreasing, we will only examine the particular case when the c_i have their modulus equal to e . There are four subcases:

1. $\widehat{p}(e) \nabla \varepsilon$ and $\widehat{p}(\ominus e) \nabla \varepsilon$;
2. $\widehat{p}(e) \nabla \varepsilon$ and $\widehat{p}(\ominus e) \nabla \varepsilon$;
3. $\widehat{p}(e) \nabla \varepsilon$ and $\widehat{p}(\ominus e) \nabla \varepsilon$;
4. $\widehat{p}(e) \nabla \varepsilon$ and $\widehat{p}(\ominus e) \nabla \varepsilon$.

The first case can appear only if n is even and

$$\widehat{p}(c) = c^n \oplus c^{n-2} \oplus \dots \oplus e,$$

which contradicts the fact the c_i are nonincreasing. The other cases correspond to a factorization studied in the first part of the proof. ■

Corollary 3.90 A sufficient condition for $\widehat{p} \in \mathcal{P}(\mathbb{S}^\vee)$ to be closed is that $|\widehat{p}|$ has distinct corners.

Example 3.91

- $c^2 \oplus e$ is always positive and therefore cannot be factored;
- $c^2 \oplus c \oplus e$ is closed and can be factored into $(c \oplus e)^2$;
- $(\gamma \ominus e)(\gamma \oplus e)^2 = (\gamma \oplus e)(\gamma \ominus e)^2 = \gamma^3 \oplus e^\bullet \gamma^2 \oplus e^\bullet \gamma \oplus e$ is a maximum representative of a closed polynomial;
- $\gamma^3 \oplus e^\bullet \gamma^2 \oplus e\gamma \oplus e$ is not a maximum representative of a closed polynomial. ■

In the following theorem the form of the inverse function of $\mathcal{F}$, which is a set-valued mapping, is made precise.

Theorem 3.92 The set $\mathcal{F}^{-1}(\widehat{p})$, with $\widehat{p} \in \mathcal{P}(\mathbb{S}^\vee)$, admits the minimum element $p^\flat \in \mathbb{S}^\vee[\gamma]$ and the maximum element $p^\sharp \in \mathbb{S}[\gamma]$ which satisfy

- $p^\flat \leq \mathcal{F}^{-1}(\widehat{p}) \leq p^\sharp$;
- $p^\flat, p^\sharp \in \mathcal{F}^{-1}(\widehat{p})$;
- $|p^\flat| = |p|^\flat$;
- $|p^\sharp| = |p|^\sharp$.

Proof The proof follows from the epimorphism property of $p \mapsto |p|$, from the corresponding result in $\mathbb{R}_{\max}$, and from the previous result on closed polynomial functions. ■

Example 3.93 For $\widehat{p} = c^2 \oplus 1c \oplus e$ we have $p^\flat = p^\sharp = \gamma^2 \oplus 1\gamma \oplus e$. For $\widehat{p} = c^2 \oplus (-1)c \oplus e$ we have $p^\flat = \gamma^2 \oplus e$, but $p^\sharp$ does not exist in this case. ■

3.7 ASYMPTOTIC BEHAVIOR OF A^k

In this section we prove a max-plus algebra analogue of the Perron-Frobenius theorem, that is, we study the asymptotic behavior of the mapping $k \mapsto A^k$, where A is an $n \times n$ matrix with entries in $\mathbb{R}_{\max}$. We can restrict ourselves to the case when $\mathcal{G}(A)$ contains at least a circuit since, otherwise, A is nilpotent, that is, $A^k = \varepsilon$ for k sufficiently large.

We suppose that the maximum cycle mean is equal to e . If this is not the case, the matrix A is normalized by dividing all its entries by the maximum cycle mean λ . Then the behavior of the general recurrent equation is easily derived from the formula $A^k = \lambda^k(\lambda^{-1}A)^k$. Therefore, in this section all circuits have nonpositive weights and some do have a weight equal to $e = 0$ which is also the maximum cycle mean. We recall that, in this situation, e is the maximum eigenvalue of A (see Remark 3.24).

3.7.1 Critical Graph of a Matrix A

Definition 3.94 For an $n \times n$ normalized matrix A , the following notions are defined:

Critical circuit: a circuit ζ of the precedence graph $\mathcal{G}(A)$ is called critical if it has maximum weight, that is, $|\zeta|_w = e$.

Critical graph: the critical graph $\mathcal{G}^c(A)$ consists of those nodes and arcs of $\mathcal{G}(A)$ which belong to a critical circuit of $\mathcal{G}(A)$. Its nodes constitute the set $\mathcal{V}^c$.

Saturation graph: given an eigenvector y associated with the eigenvalue e , the saturation graph $S(A, y)$ consists of those nodes and arcs of $\mathcal{G}(A)$ such that $A_{ij}y_j = y_i$ for some i and j with $y_i, y_j \neq \varepsilon$.

Cyclicity of a graph: the cyclicity of a m.s.c.s. is the gcd (greatest common divisor) of the lengths of all its circuits. The cyclicity $c(\mathcal{G})$ of a graph $\mathcal{G}$ is the lcm (least common multiple) of the cyclicities of all its m.s.c.s.'s.

Example 3.95 Consider the matrix

$$A = \begin{pmatrix} e & e & \varepsilon & \varepsilon \\ -1 & -2 & \varepsilon & \varepsilon \\ \varepsilon & -1 & -1 & e \\ \varepsilon & \varepsilon & e & e \end{pmatrix}.$$

- Its precedence graph $G(A)$ has three critical circuits $\{1\}$, $\{3, 4\}$, $\{4\}$.
- Its critical graph is the precedence graph of the matrix

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

- Matrix A has the eigenvector $(e \ -1 \ -2 \ -2)'$ associated with the eigenvalue e . The corresponding saturation graph is the precedence graph of the matrix

$$S = \begin{pmatrix} e & \varepsilon & \varepsilon & \varepsilon \\ -1 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & -1 & \varepsilon & e \\ \varepsilon & \varepsilon & e & e \end{pmatrix}.$$

- The cyclicity of the critical graph is 1. Indeed, the critical graph has two m.s.c.s.'s with nodes $\{1\}$ and $\{3, 4\}$, respectively. The second one has two critical circuits, $\{4\}$ and $\{3, 4\}$, of length 1 and 2, respectively. The cyclicity of the first m.s.c.s. is 1, the cyclicity of the second m.s.c.s. is $\gcd(1, 2) = 1$. Therefore the cyclicity of $G^c(A)$ is $\text{lcm}(1, 1) = 1$. ■

Let us give now some simple results about these graphs which will be useful in the following subsections.

Theorem 3.96 Every circuit of $G^c(A)$ is critical.

Proof If this were not the case, we could find a circuit ζ , composed of subpaths ζ_i of critical circuits γ_i , with a weight different from e . If this circuit had a weight greater than e , it would contradict the assumption that the maximum circuit weight of $G(A)$ is e . If the weight of ζ were less than e , the circuit ζ' composed of the union of the complements of ζ_i in γ_i would be a circuit of weight greater than e and this would also be a contradiction. ■

Corollary 3.97 Given a pair of nodes (i, j) in $G^c(A)$, all paths connecting i to j in $G^c(A)$ have the same weight.

Proof If there exists a path p from i to j in $G^c(A)$, it can be completed by a path p' from j to i also in $G^c(A)$ to form a critical circuit. If there exists another path p'' from i to j in $G^c(A)$, the concatenations of p and p' on the one hand, and of p'' and p' on the other hand, form two critical circuits with the same weight. Hence p and p'' must have the same weight. ■

Theorem 3.98 For each node in a saturation graph, there exists a circuit upstream in this graph. The circuits of any saturation graph belong to $G^c(A)$.

Proof Indeed, if i is one of its nodes, there exists another node j , upstream with respect to i , such that $y_i = A_{ij}y_j$; $y_i, y_j \neq \varepsilon$. The same reasoning shows that there exists another node upstream with respect to j , etc. Because the number of nodes of $S(A, y)$ is finite, the path $(i, j, \dots)$ obtained by this construction contains a circuit. A circuit $(i_0, i_1, \dots, i_k, i_0)$ of a saturation graph $S(A, y)$ satisfies

$$y_{i_1} = A_{i_1 i_0} y_{i_0}, \dots, y_{i_0} = A_{i_0 i_k} y_{i_k}.$$

The multiplication of all these equalities shows that the weight of the circuit $(i, i_1, \dots, i_k, i)$ is e . ■

Example 3.99 In Example 3.95, node 2 has the critical circuit (indeed a loop) $\{1\}$ upstream. ■

3.7.2 Eigenspace Associated with the Maximum Eigenvalue

In this subsection we describe the set of eigenvectors of matrix A associated with the eigenvalue e . Clearly, this set is a moduloid. We characterize a nonredundant set of generators of this moduloid as a subset of the columns of A^+ .

Here the word 'eigenvector' must be understood as 'eigenvector associated with the eigenvalue e '. We will use the notation A_{ij}^p for $(A^p)_{ij}$ and A_{ij}^+ for $(A^+)_{ij}$.

Theorem 3.100 If y is an eigenvector of A , it is also an eigenvector of A^+ . It is the linear combination of the columns $A_{\cdot i}^+$, $i \in \mathcal{V}^c$. More precisely,

$$y = \bigoplus_{i \in \mathcal{V}^c} y_i A_{\cdot i}^+. \quad (3.27)$$

Proof The first part of the theorem is trivial. Let us prove Formula (3.27). Consider two nodes i and j in the same m.s.c.s. of the saturation graph $S(A, y)$. There exists a path $(i, i_1, \dots, i_k, j)$ which satisfies

$$y_{i_1} = A_{i_1 i} y_i, \dots, y_j = A_{j i_k} y_{i_k}.$$

Therefore, $y_j = w y_i$ with

$$w = A_{j i_k} \cdots A_{i_1 i} \leq A_{ji}^+,$$

and we have

$$A_{lj}^+ y_j = A_{lj}^+ w y_i \leq A_{lj}^+ A_{ji}^+ y_i \leq A_{li}^+ y_i, \quad \forall l. \quad (3.28)$$

We could have chosen i in a circuit of the saturation graph according to Theorem 3.98. This i will be called $i(j)$ in the following. We have

$$\begin{aligned} y_l &= \bigoplus_{j \in S(A, y)} A_{lj}^+ y_j && \text{(by definition of } S(A, y) \text{)} \\ &\leq \bigoplus_{j \in S(A, y)} A_{li(j)}^+ y_{i(j)} && \text{(by (3.28))} \\ &\leq \bigoplus_{i \in \mathcal{V}^c} A_{li}^+ y_i, \quad \forall l, \end{aligned}$$

where the last inequality stems from the fact that $i(j)$, belonging to a circuit of a saturation graph, belongs also to $\mathcal{V}^c$ by Theorem 3.98. The reverse inequality is derived immediately from the fact that y is an eigenvector of A^+ . ■

Theorem 3.101 *Given a matrix A with maximum circuit weight e , any eigenvector associated with the eigenvalue e is obtained by a linear combination of N_A^c columns of A^+ , where N_A^c denotes the number of m.s.c.s.'s of $G^c(A)$. More precisely we have*

1. the columns $A_{\cdot i}^+, i \in \mathcal{V}^c$, are eigenvectors;
2. if nodes i and j belong to the same m.s.c.s. of $G^c(A)$, then $A_{\cdot i}^+$ and $A_{\cdot j}^+$ are 'proportional';
3. no $A_{\cdot i}^+$ can be expressed as a linear combination of columns $A_{\cdot j}^+$ which only makes use of nodes j belonging to m.s.c.s.'s of $G^c(A)$ distinct from $[i]$.

Proof The first statement has already been proved in Theorem 3.23. Consider now the second statement: since $A^+ A^+ \leq A^+$ and $A_{ij}^+ A_{ji}^+ = e$, if nodes i and j belong to the same m.s.c.s. of $G^c(A)$, hence to the same critical circuit by Theorem 3.96, we have

$$A_{ij}^+ A_{ji}^+ \leq A_{ii}^+ = A_{ii}^+ A_{ij}^+ A_{ji}^+ \leq A_{ij}^+ A_{ji}^+, \quad \forall l,$$

which shows that

$$A_{ij}^+ A_{ji}^+ = A_{ii}^+, \quad \forall l.$$

This result, together with (3.27), show that N_A^c columns of A^+ are sufficient to generate all eigenvectors.

The third statement of the theorem claims that we cannot further reduce this number of columns. Otherwise, one column of A^+ , say i , could be expressed as a linear combination of other columns of A^+ selected in other m.s.c.s.'s of $G^c(A)$. Let K denote the set of columns involved in this linear combination. Then, construct a matrix B as follows. Let $J = K \cup \{i\}$. Matrix B is obtained from A^+ by deleting all rows and columns with indices out of J . By construction, $B_{kk} = e, \forall k$, and the weights of all circuits of $G(B)$ are less than or equal to e . The linear combination of the columns of A^+ is preserved when restricting this combination to matrix B . Owing to the multilinearity and antisymmetry of the determinant, and

to the decomposition of any permutation into circular permutations, $\det B = e^{\cdot}$. Since $\bigotimes_k B_{kk} = e$, there must exist another permutation ς such that $\bigotimes_k B_{k\varsigma(k)} = e$. Stated differently, there must exist a critical circuit connecting some m.s.c.s.'s of $G^c(A)$ and this yields a contradiction. ■

Example 3.102 If we return to Example 3.95, matrix A^+ is equal to

$$\begin{pmatrix} e & e & \varepsilon & \varepsilon \\ -1 & -1 & \varepsilon & \varepsilon \\ -2 & -1 & e & e \\ -2 & -1 & e & e \end{pmatrix}.$$

Nodes 1, 3, 4 belong to $G^c(A)$ which has two m.s.c.s.'s, namely $\{1\}$ and $\{3,4\}$. Columns 1 and 3 are independent eigenvectors. Column 4 is equal to column 3. ■

3.7.3 Spectral Projector

In this subsection we build the spectral projector on the eigenspace (invariant mod-
uloid) associated with the eigenvalue e .

Definition 3.103 (Spectral projector) A matrix Q satisfying $AQ = QA = Q^2$ is called a spectral projector of A associated with the eigenvalue e .

Theorem 3.104 The matrices

$$Q_i \stackrel{\text{def}}{=} A_{\cdot i}^+ A_{i \cdot}^+, \quad i \in \mathcal{V}^c, \quad (3.29)$$

are spectral projectors of A .

Proof The properties $AQ_i = Q_i A = Q_i$ follow from the fact that the columns $A_{\cdot i}^+, i \in \mathcal{V}^c$, are eigenvectors of A (from which we deduce by transposition that the rows of A^+ such that $A_{ii}^+ = e$ are left eigenvectors). Let us prove that $Q_i^2 = Q_i$, that is,

$$A_{\cdot i}^+ A_{i \cdot}^+ = \bigoplus_k A_{\cdot i}^+ A_{ik}^+ A_{ki}^+ A_{i \cdot}^+.$$

This relation is true because $A_{ii}^+ = e$ implies $\bigoplus_k A_{ik}^+ A_{ki}^+ = e$. ■

Theorem 3.105 The matrix $Q \stackrel{\text{def}}{=} \bigoplus_{i \in \mathcal{V}^c} Q_i$, where the matrices Q_i are defined by (3.29), is a spectral projector.

Proof The only nontrivial fact to prove is that $Q^2 = Q$. This relation will be proved if we prove that $Q_i Q_j \leq Q_i \oplus Q_j$. This last inequality is true because it means that the greatest weight of the paths connecting a pair of nodes and traversing i and j is less than the maximum weight of the paths connecting the same pair of nodes and traversing either i or j . ■

Example 3.106 Continuing with Example 3.95, we obtain two elementary spectral projectors:

$$Q_1 = \begin{pmatrix} e \\ -1 \\ -2 \\ -2 \end{pmatrix} \begin{pmatrix} e & e & \varepsilon & \varepsilon \end{pmatrix} = \begin{pmatrix} e & e & \varepsilon & \varepsilon \\ -1 & -1 & \varepsilon & \varepsilon \\ -2 & -2 & \varepsilon & \varepsilon \\ -2 & -2 & \varepsilon & \varepsilon \end{pmatrix},$$

$$Q_2 = \begin{pmatrix} \varepsilon \\ \varepsilon \\ e \\ e \end{pmatrix} \begin{pmatrix} -2 & -1 & e & e \end{pmatrix} = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ -2 & -1 & e & e \\ -2 & -1 & e & e \end{pmatrix},$$

$$Q = Q_1 \oplus Q_2 = \begin{pmatrix} e & e & \varepsilon & \varepsilon \\ -1 & -1 & \varepsilon & \varepsilon \\ -2 & -1 & e & e \\ -2 & -1 & e & e \end{pmatrix}.$$

■

3.7.4 Convergence of A^k with k

In this subsection we give a necessary and sufficient condition for the convergence of the powers of matrix A . To achieve this goal, we equip $\mathbb{R}_{\max}$ with the topology: when $n \rightarrow +\infty$,

$$x_n \rightarrow x \Leftrightarrow |x_n - x|_e \stackrel{\text{def}}{=} |\exp(x_n) - \exp(x)| \rightarrow 0.$$

The purpose of this topology is to simplify the study of the convergence towards ε . Indeed, because $\exp(\varepsilon) = e$, $\lim_n |x_n - \varepsilon|_e = 0$ and $\lim_n |x'_n - \varepsilon|_e = 0$ imply that $\lim_n |x_n - x'_n|_e = 0$. This property, which is not true with respect to the usual absolute value in $\mathbb{R}$, is useful for the asymptotic cyclicity notion that we will introduce later on. We first recall the following result on the diophantine linear equation (see [33]).

Lemma 3.107 For all p and n which are coprime and for all $q \geq (p-1)(n-1)$, there exist two integers $a(q)$ and $b(q)$ such that $q = a(q)p + b(q)n$.

Theorem 3.108 A necessary and sufficient condition to have $\lim_{k \rightarrow \infty} A^k = Q$ is that the cyclicity—see Definition 3.94—of each m.s.c.s. of $G^c(A)$ is equal to 1.

Proof Let us prove the sufficient condition first. Consider a node i in $G^c(A)$. For any other node j , there exists a path from i to j of maximum weight (possibly equal to ε). If there happens to be more than one such path, we take the one with the least length and call this length $p(i, j)$ which is less than $n-1$ if A is an $n \times n$ matrix. If the maximum weight is ε , we consider that the length is equal to 1. By Lemma 3.107 and the assumption on cyclicity, there exists some integer $M(i)$ such that, for all m greater than $M(i)$, there exists a critical circuit of length m in

$[i]_{G^c(A)}$ (the m.s.c.s. of $G^c(A)$ to which i belongs). Therefore, because the maximum weight of the circuits is e , any maximum-weight path from i to j of length q greater than $M(i) + p$ is composed of a critical circuit of length $q - p$ traversing i and a maximum-weight path from i to j (of any length). Therefore $A_{ji}^q = A_{ji}^+$ for all q greater than $p + M(i)$ and this holds for all i in $G^c(A)$. Since $A_{ii}^+ = e$, we also have $A_{ji}^q = A_{ji}^+ A_{ii}^+$.

Consider now another node l which does not belong to $G^c(A)$. Let i be a node in $G^c(A)$, let q be large enough and let $p \leq n$ be such that $A_{il}^p = A_{il}^+$ (such a p exists because circuits have weights less than e , hence the lengths of maximum-weight paths between any two nodes do not need to exceed n). We have

$$A_{jl}^q \geq A_{ji}^{q-p} A_{il}^p = A_{ji}^+ A_{il}^+,$$

where the inequality is a consequence of the matrix product in $\mathbb{R}_{\max}$ whereas the equality arises from the previous part of the proof and from the property of p . If we have a strict inequality, it means that the paths with maximum weight from j to l do not traverse i , and since this is true for any i in $G^c(A)$, these paths do not traverse $G^c(A)$. On the other hand, for q large enough, they must traverse some circuits which therefore have a strictly negative weight. When q increases, these paths have weights arbitrarily close to ε . Finally, this situation is possible only if there is no node of $G^c(A)$ located downstream of l in $G(A)$. In this case $A_{il}^+ = \varepsilon$ for all i in $G^c(A)$ and therefore

$$\lim_{q \rightarrow \infty} A_{jl}^q = \varepsilon = \bigoplus_{i \in V^c} A_{ji}^+ A_{il}^+.$$

By collecting all the above results, we have proved that

$$\lim_{q \rightarrow \infty} A_{jl}^q = \bigoplus_{i \in V^c} A_{ji}^+ A_{il}^+, \quad \forall j, l \in G(A).$$

Conversely, suppose that the above limit property holds true and that at the same time the cyclicity of $G^c(A)$ is strictly greater than 1. Let us consider a node $i \in G^c(A)$ (hence $A_{ii}^+ = e$). We have

$$\exp(A_{ii}^{k \times d}) = \exp(e) = \exp(A_{ii}^{k \times d + 1}) + \eta,$$

where η can be arbitrarily small because of the assumed limit. But $A_{ii}^{k \times d + 1} = A_{ii}^p$ for $0 \leq p \leq n$ (again because circuits have nonpositive weights). Therefore, $A_{ii}^{k \times d + 1}$ can assume values out of a finite set. From the relation above, it should be clear that $A_{ii}^{k \times d + 1} = e$. This means that there exists a circuit of length $k \times d + 1$. But the gcd of kd and $k \times d + 1$ is 1, which is a contradiction. ■

Theorem 3.109 Suppose that $G(A)$ is strongly connected. Then there exists K such that

$$\forall k \geq K, \quad A^k = Q,$$

if and only if the cyclicity of each m.s.c.s. of $G^c(A)$ is equal to 1.

Proof The proof is similar to the previous one. The only difference lies in the second part of the 'if' part. Under the assumption that $G(A)$ is strongly connected, a path of maximum weight from l to j with length large enough necessarily crosses $G^c(A)$. Therefore, for q large enough we have

$$A_{jl}^q = A_{ji}^+ A_{il}^+,$$

where i belongs to $G^c(A)$. ■

Example 3.110

- Using Example 3.95 once more, we have

$$A^2 = \begin{pmatrix} e & e & \varepsilon & \varepsilon \\ -1 & -1 & \varepsilon & \varepsilon \\ -2 & -2 & e & e \\ \varepsilon & -1 & e & e \end{pmatrix},$$

$$A^3 = A^4 = \dots = \begin{pmatrix} e & e & \varepsilon & \varepsilon \\ -1 & -1 & \varepsilon & \varepsilon \\ -2 & -1 & e & e \\ -2 & -1 & e & e \end{pmatrix}.$$

Therefore A^n , $n \geq 3$, is equal to $Q_1 \oplus Q_2$ given in Example 3.106.

- In the previous example the periodic regime is reached after a finite number of steps. This is true for the submatrix associated with the nodes of the critical graph but it is not true in general for the complete matrix. To show this, take the example

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

- In the previous example there is an entry which goes to ε . When all entries converge to a finite number, the periodic regime can be reached also, but the time needed may be arbitrarily long. Consider the matrix

$$A = \begin{pmatrix} -\eta & -1 \\ e & e \end{pmatrix}.$$

The matrix A^k converges to the matrix

$$\begin{pmatrix} -1 & -1 \\ e & e \end{pmatrix}$$

if η is a small positive number. But we have to wait for a power of order $1/\eta$ to reach the asymptote. ■

3.7.5 Cyclic Matrices

In this subsection we use the previous theorem to describe the general behavior of the successive powers of matrix A which turns out to be essentially cyclic.

Definition 3.111

Cyclicity of a matrix: a matrix A is said to be cyclic if there exist d and M such that $\forall m \geq M$, $A^{m+d} = A^m$. The least such d is called the cyclicity of matrix A and A is said to be d -cyclic.

Asymptotic cyclicity: a matrix A is said to be asymptotically cyclic if there exists d such that, for all $\eta > 0$, there exists M such that, for all $m \geq M$, $\sup_{ij} |(A^{m+d})_{ij} - (A^m)_{ij}| \leq \eta$. The least such d is called the asymptotic cyclicity of matrix A and A is said to be d -asymptotically cyclic.

Theorem 3.112 Any matrix is asymptotically cyclic. The asymptotic cyclicity d of matrix A is equal to the cyclicity ρ of $G^c(A)$. Moreover if $G(A)$ and $G(A^\rho)$ are connected, the matrix is ρ -cyclic.

Proof This result is already proved in the case $\rho = 1$. For any matrix A , if we consider $B = A^\rho$, then the asymptotic cyclicity ρ' of B is equal to 1. Indeed, the nodes of the critical graph of B are a subset of $G^c(A)$, and around each such node there exists a loop. The necessary and sufficient conditions of convergence of the powers of a matrix can be applied to B (see Theorems 3.108 and 3.109). They show the convergence (possibly in a finite number of stages) of $B^k = A^{k \times \rho}$ to the spectral projector Q associated with B . Because any m can be written $h + k \times \rho$, $A^m = A^{h+k \times \rho} = A^h B^k$ converges to $A^h Q$ when k goes to infinity. This is equivalent to saying that matrix A is d -asymptotically-cyclic (or d -cyclic in the case of finite convergence), with $d \leq \rho$.

Let us prove that the asymptotic cyclicity d of matrix A is greater than or equal to ρ (hence it is equal to ρ). The proof when the matrix is not only asymptotically cyclic but cyclic is similar. Consider a node of the m.s.c.s. $l = [i]_{G^c(A)}$ of $G^c(A)$ and let ρ_l denote its cyclicity. By definition of d , we have

$$\exp(A_{ii}^{k \times \rho_l}) = \exp(e) = \exp(A_{ii}^{k \times \rho_l + d}) + \eta,$$

for η arbitrarily small and for k large enough. Therefore $A_{ii}^{k \times \rho_l} = A_{ii}^{k \times \rho_l + d}$ and there is a circuit of length d in the m.s.c.s. l of $G^c(A)$. Therefore ρ_l divides d . But this is true for all m.s.c.s.'s of $G^c(A)$ and therefore d is divided by the lcm of all the ρ_l which is ρ . ■

Example 3.113 The matrix

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

has cyclicity 2. Indeed,

$$A^{2n} = \begin{pmatrix} -2n & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon \\ \varepsilon & \varepsilon & e \end{pmatrix}, \quad A^{2n+1} = \begin{pmatrix} -(2n+1) & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e \\ \varepsilon & e & \varepsilon \end{pmatrix}.$$

3.8 NOTES

The max-plus algebra is a special case of a more general structure which is called a dioid structure. This is the topic of the next chapter. Nevertheless the max-plus algebra, and the algebras of vector objects built up on it, are important examples of dioids for this book because they are perfectly tailored to describe synchronization mechanisms. They were also used to compute paths of maximum length in a graph in operations research. This is why they are sometimes called 'path algebra' [67].

Linear systems of equations in the max-plus algebra were systematically studied in [49]. Some other very interesting references on this topic are [67], [130]. In these references the Gauss elimination algorithm can be found. It was not discussed here. Linear dependence has been studied in [49], [65], [93], [126] and [62]. Several points of view exist but none of them is completely satisfactory. Moreover the geometry of linear manifolds in the max-plus algebra is not well understood (on this aspect, see [126]).

The only paper that we know on a systematic study of polynomial and rational functions in the max-plus algebra is [51]. In this paper one can find some results on rational functions not detailed in this book.

The symmetrization of the max-plus algebra was discussed earlier in [109] and [110]. The presentation given here is based on these references. This symmetrization is more deeply studied in [62]. The reference [65] has been an important source of ideas even though symmetrization has been avoided in this paper. The proof of the Cramer formula is mainly due to S. Gaubert and M. Akian. Relevant references are [118], [124], [93].

The attempt made here to discuss polynomials in $\mathbb{S}$ is new. It could give a new insight into the eigenvalue problem. Because of the lack of space this discussion has not been continued here.

The section on the max-plus Perron-Frobenius theorem is a new version of the report [37]. The proof is mainly due to M. Viot. Some other relevant references are [64], [49], [127].

CHAPTER 4

Dioids

4.1 INTRODUCTION

In previous chapters, the set $\mathbb{R} \cup \{-\infty\}$ (respectively $\mathbb{R} \cup \{+\infty\}$) endowed with the max (respectively the min) operation as addition and the usual addition as multiplication has appeared as a suitable algebraic structure for obtaining 'linear' models of some discrete event systems. In Chapter 5 it will be shown that another slightly more complex structure is also appropriate for the same class of systems.

All these algebraic structures share some common features that will be studied in the present chapter. However, there is yet no universal name nor a definite set of axioms everybody agrees upon in this general field. We refer the reader to the notes at the end of this chapter where some related works are briefly discussed. Here we adopt the following point of view: we introduce a first 'minimal' set of axioms according to what seems to be the intersection of axioms generally retained in the works alluded to above, and also according to what seems to be appropriate for the linear system theory we are going to develop in Chapter 5. Starting from this minimal set of axioms, we derive some basic results. To obtain further results we may need to introduce some additional assumptions or properties, which we do only when necessary: it is then clear where this added structure is really needed.

We use the word 'dioid' as the generic name for the algebraic structure studied in this chapter. The linguistic roots of this name and its introduction in the literature are discussed in the notes section.

Dioids are structures that lie somewhere between conventional linear algebra and semilattices endowed with an internal operation generally called multiplication. With the former, it shares combinatorial properties such as associativity and commutativity of addition, associativity of multiplication, distributivity of multiplication with respect to addition, and of course the existence of zero and identity elements. With the latter, it shares the features of an ordered structure (adding is then simply taking the upper bound) endowed with another 'compatible' operation. Therefore one may expect that the results of linear algebra which depend only on combinatorial properties will generalize to dioids. A typical case is the Cayley-Hamilton theorem. On the other hand, since neither addition nor multiplication are invertible in general dioids (in this respect the max-plus algebra is special since the structure associated with $+$ is a group), one appeals to the classical theory of residuation in lattice structures to provide alternative notions of inversion of the basic operations and of other order-preserving mappings. This yields a way to 'solve' some equations in a certain sense related to the order structure even if there is no solution in a more