KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT DER TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELEKTROTECHNIEK-ESAT
Kard. Mercierlaan 94, 3001 Leuven (Heverlee)

CONTINUOUS-TIME MATRIX ALGORITHMS, SYSTOLIC ALGORITHMS AND ADAPTIVE NEURAL NETWORKS

Promotoren:
Prof. Dr. Ir. J. Vandewalle,
Dr. Ir. M. Moonen.

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de toegepaste wetenschappen

door

Jeroen DEHAENE

Oktober 1995

KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT DER TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELEKTROTECHNIEK-ESAT
Kard. Mercierlaan 94, 3001 Leuven (Heverlee)

CONTINUOUS-TIME MATRIX ALGORITHMS, SYSTOLIC ALGORITHMS AND ADAPTIVE NEURAL NETWORKS

Jury:
Prof. W. Dutré, vice-decaan, voorzitter,
Prof. J. Vandewalle, promotor,
Dr. M. Moonen, promotor,
Prof. A. Bultheel,
Prof. B. De Moor,
Dr. F. Gaston (Univ. of Birmingham, UK),
Prof. U. Helmke (Univ. Würzburg, Germany),
Prof. P. Van Dooren (U.C.L., Louvain-la-Neuve)

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de toegepaste wetenschappen

door

Jeroen DEHAENE

U.D.C. 519.61

Oktober 1995

Voorwoord

Bij het beëindigen van dit werk, houd ik eraan enkele mensen te bedanken.

In de eerste plaats bedank ik mijn promotoren, Prof. J. Vandewalle en Dr. M. Moonen. Joos Vandewalle wil ik bedanken omdat ik sterk apprecieer hoe hij vanuit zijn eigen interesse voor zowel fundamentele als toegepaste wetenschappen, mij en anderen de kans en de vrijheid geeft om naar ingenieursnormen sterk theoretisch onderzoek te verrichten, en tegelijk aandacht blijft vragen voor de toepasbaarheid. Marc Moonen verdient mijn dank omdat hij de hand had in enkele beslissende wendingen, omdat zijn constructieve kritiek heeft bijgedragen tot de begrijpbaarheid van deze tekst en omdat de dreiging van zijn perfectionistisch oordeel mijn spontane afkeer voor formele details heeft kunnen kleinkrijgen.

Verder dank ik de andere leden van het leescomité, Prof. B. De Moor en Prof. A. Bultheel. Bart De Moor moet ik ook bedanken om zijn bulderend enthousiasme, dat zo luid weergalmt dat er ook in dit werk van alles meetrilt, en omdat het dankzij gelijke interesses en ondanks een verschil in stijl, aangenaam samenwerken is met hem. Adhemar Bultheel wil ik bedanken voor de zorgvuldigheid waarmee hij dit werk ondanks de examentijd heeft nagelezen.

Mijn dank gaat ook uit naar de voorzitter van de jury, Prof. W. Dutré, voor zijn bereidwillige medewerking.

I would also like to thank the other members of the jury, Dr. F. Gaston (University of Birmingham), Prof. U. Helmke (Universität Würzburg) and Prof. P. Van Dooren (Université Catholique de Louvain, Louvain-la-Neuve), for their cooperation. I owe special thanks to Fiona Gaston, who has made some of my sentences sound more English (I hope this one is all right). I thank Uwe Helmke for the enthusiasm with which he promotes differential geometry in the world of engineers. I also want to apologize to the readers, and especially the members of the jury, for descriptions with too many words or notations with too many subscripts, where I simply wanted to say 'this here on this figure'.

I also thank Prof. R.W. Brockett (Harvard University) for being interested in my work. And I am looking forward to visiting him and his research group. I

am grateful to Prof. J.A. Nossek (Technische Universität München) for having invited us in his special session at MTNS '93 in Regensburg, Germany.

Verder wil ik ook een hele rij collega's bedanken. I want to thank Yi Cheng for the fruitful and pleasant cooperation. Ik had verschillende nuttige discussies met Filiep Vanpoucke, met wie ik reeds vele voetsporen deelde, Johan Suykens en Yves Moreau, van de neurale-netwerkengroep, Lieven Vandenberghe, die indruk maakte op mij als verse beginneling door de inhoud en de stijl van zijn doctoraat, Johan David, die mij ook in contact bracht met het werk van Brockett, Lieven De Lathauwer en Peter Van Overschee. I also thank Johan David, Jairo Espinosa and Yves Moreau for the shared walks for a cup of tea or coffee.

Ik wil ook de secretaressen van het eerste en het laatste uur bedanken, Ann Deforce, Ingrid Tokka, Rita De Wolf, Rebecca Crabbé en Ida Tassens, en dan mag ik ook het organisatorische werk van Bart Motmans niet vergeten. Tenslotte wil ik ook de gehele systeemploeg bedanken.

Ik dank ook het Nationaal Fonds voor Wetenschappelijk Onderzoek, voor de financiële steun.

Van de vele andere mensen, die mijn dank wel langs minder officiële kanalen zullen vernemen, wens ik tenslotte nog speciaal mijn ouders en ook mijn broers te vermelden voor de steun die ik kreeg en krijg op mijn weg.

Abstract

In the domain of 'continuous-time matrix algorithms', matrix based algorithms are studied from the viewpoint of continuous-time systems theory and differential geometry. We put emphasis on formulas for tracking decompositions of a time-varying matrix, and present them as tools for the design and analysis of matrix algorithms. We define a class of continuous-time matrix algorithms with a uniform parallel signal flow graph. We derive algorithms for recursive least-squares estimation, belonging to this class, which are continuous-time limits of known systolic algorithms. Some of them are candidates for analog realization. For algorithms for subspace tracking, belonging to the same class, we present new analysis results based on continuous-time concepts. From these algorithms we also derive new fully pipelined systolic algorithms, inheriting the main properties of their continuous-time counterparts. We reinterpret the presented continuous-time adaptive signal processing algorithms as adaptation laws for neural networks and derive other adaptation laws, based on isospectral algorithms, for neural memory applications. Finally, we illustrate the relevance of the differential geometric approach for the design of optimization algorithms, by studying an application in robust control theory, the computation of the structured singular value.

Abstract

Het domein van de 'tijdscontinue matrixalgoritmen' bestudeert matrixgebaseerde algoritmen vanuit het standpunt van de tijdscontinue systeemtheorie en de differentiaalmeetkunde. We benadrukken formules voor het volgen van ontbindingen van een tijdsafhankelijke matrix, die gebruikt kunnen worden voor het ontwerp en de analyse van matrixalgoritmen. We definiëren een klasse van tijdscontinue matrixalgoritmen met een uniform parallelle signaalgrafe. We leiden algoritmen voor recursieve kleinste-kwadratenschatting af, die tot deze klasse behoren, en die tijdscontinue limieten zijn van gekende systolische algoritmen. Sommige komen in aanmerking voor analoge realisatie. Voor algoritmen voor deelruimtevolgen, die tot dezelfde klasse behoren, geven we een nieuwe analyse, op basis van tijdscontinue concepten. Uit deze algoritmen worden ook nieuwe volledig 'gepipelinede' systolische algoritmen afgeleid, die de belangrijkste eigenschappen van hun tijdscontinue tegenhangers overerven. We herinterpreteren de voorgestelde tijdscontinue signaalverwerkingsalgoritmen als adaptatiewetten voor neurale netwerken en leiden ook andere adaptatiewetten af, gebaseerd op isospectrale algoritmen, voor neurale geheugentoepassingen. Tenslotte, illustreren we de relevantie van de differentaalmeetkundige benadering voor het ontwerp van optimisatie-algoritmen, aan de hand van een toepassing in robuuste controletheorie, de berekening van de gestructureerde singuliere waarde.

Notation

notation	**meaning**	**def./p.**
A^T	transpose of matrix A	
A^{-T}	inverse transpose of matrix A	
A^H	Hermitian transpose of matrix A	
c^*	complex conjugate of c	
$A_{\alpha,\beta}$	submatrix of A with row indices in α and column indices in β. Subscripts $_{\leqslant i}$, $_{<i}$, $_{\geqslant i}$ or $_{>i}$ denote indices (strictly) less or (strictly) greater than i. A subscript $_{\cdot}$ denotes the set of all indices. Single entries are denoted by the corresponding lower-case symbol if no confusion is possible: $a_{i,j} = A_{i,j}$.	
$\dot{x}$	time derivative of x	10
$\langle A, B\rangle$	standard inner product of matrices A and B	2.7,3.27
$\|A\|_F$	Frobenius norm of matrix A	2.7
$[X, Y]$	(Lie) bracket of matrices X and Y	2.10
$X.*Y$	elementwise matrix product	2.9
$\angle(x, y)$	angle between vectors x and y	
$\langle f, g\rangle_{(t)}$	inner product of functions $f, g \in L_{2,\lambda}(-\infty, t]$ or generalized matrix product of vector functions $f, g \in (L_{2,\lambda}(-\infty, t])^n$.	6.2
$\|f\|_{(t)}$	norm of function $f \in L_{2,\lambda}(-\infty, t]$	6.2
$y \parallel \{x_1, \ldots, x_n\}$		78
$y \perp \{x_1, \ldots, x_n\}$		78
$\mathbb{C}$	set of complex numbers	
$\mathcal{C}_m(t)$	cross product subspace of dimension m	7.12
$\mathcal{C}_A$	space of $n \times n$-matrices commutable with $A \in \mathbb{R}^{n\times n}$	2.8
c_A	commutator of matrix A	2.9
$\mathrm{cf}(A)$	Cholesky factor of matrix A	3.1
$\mathrm{col}(A)$	column space of matrix A	
$\mathrm{D}\Phi\|_X$	derivative of a map Φ at X	31

d_A	elementwise multiplication with matrix A	2.9
$\mathrm{diag}\{a_1,\ldots,a_n\}$	diagonal matrix with diagonal entries $a_1,\ldots,a_n$	
$\mathrm{diag}[A_1,\ldots,A_n]$	block diagonal matrix with block diagonal entries $A_1,\ldots,A_n$	
$\mathrm{diag}(A)$	diagonal part of matrix A	2.9
$\dim\mathcal{S}$	dimension of set $\mathcal{S}$	
$E\{x\}$	expectation of random variable x	
$E^{(i,j)}$	matrix of standard basis	2.6
$e_{R,S}$	equivalence transformation with matrices R and S	2.9
$\mathcal{G}_{diag}(n)$	group of nonsingular diagonal $n\times n$-matrices	2.24
$\mathsf{GL}(n)$	general linear group of $n\times n$-matrices	2.24
$\mathcal{G}_{low}(n)$	group of nonsingular lower triangular $n\times n$-matrices	2.24
$\mathcal{G}_{lows}(n)$	group of lower triangular $n\times n$-matrices with unit diagonal	2.24
$\mathcal{G}_{upp}(n)$	group of nonsingular upper triangular $n\times n$-matrices	2.24
$\mathcal{G}_{upp}(n)^+$	group of upper triangular $n\times n$-matrices with positive diagonal	2.24
$\mathcal{G}_{upps}(n)$	group of upper triangular $n\times n$-matrices with unit diagonal	2.24
$\mathfrak{g}_{diag}(n)$	Lie algebra of $\mathcal{G}_{diag}(n)$	2.25
$\mathfrak{gl}(\mathrm{n})$	Lie algebra of $\mathsf{GL}(n)$	2.25
$\mathfrak{g}_{low}(n)$	Lie algebra of $\mathcal{G}_{low}(n)$	2.25
$\mathfrak{g}_{lows}(n)$	Lie algebra of $\mathcal{G}_{lows}(n)$	2.25
$\mathfrak{g}_{upp}(n)$	Lie algebra of $\mathcal{G}_{upp}(n)$	2.25
$\mathfrak{g}_{upp}(n)^+$	Lie algebra of $\mathcal{G}_{upp}(n)^+$	2.25
$\mathfrak{g}_{upps}(n)$	Lie algebra of $\mathcal{G}_{upps}(n)$	2.25
$G^{i\|j}$	Givens rotation affecting components i and j	2.16
$\mathrm{grad}_x f$	gradient of f at x	3.21
$\mathcal{I}_A$	isospectral manifold through matrix A	2.17
$\mathrm{icf}(A)$	inverse Cholesky factor of matrix A	3.1
id	identity map	
$\mathrm{ilf}(A)$	inverse L-factor of matrix A	3.1
$\mathrm{Im}\,c$	imaginary part of a complex number c	
$\mathrm{inv}(A)$	inverse of matrix A	3.1
$\mathrm{irf}(A)$	inverse R-factor of matrix A	3.1
$\mathrm{iuf}(A)$	inverse U-factor of matrix A	3.1
$L_{2,\lambda}(-\infty,t]$	set of square integrable functions with exponential weight over $(-\infty,t]$	6.2
l_A	left multiplication with A	2.9
$\mathrm{lf}(A)$	L-factor of matrix A	3.1
$\mathrm{low}(A)$	lower triangular part of matrix A	2.9
$\mathrm{lowh}(A)$	lower triangular half of matrix A	2.9
$\mathrm{lows}(A)$	strictly lower triangular part of matrix A	2.9
N_c	continuous-time information matrix	6.5
N_d	discrete-time information matrix	6.12

$\mathrm{null}(A)$	null space of matrix A	
$\mathcal{O}(a)$	orbit of element a	25
$\mathsf{O}(n)$	orthogonal group of $n \times n$-matrices	2.24
$\mathrm{O}()$	order of magnitude	
$\mathcal{Q}$	group of structured unitary matrices	11.2
$\mathfrak{q}$	Lie algebra of $\mathcal{Q}$	11.4
$\mathrm{qf}(A)$	Q-factor of matrix A	3.1
$\mathbb{R}$	set of real numbers	
r_A	right multiplication with A	2.9
$\mathrm{Re}\, c$	real part of a complex number c	
$\mathrm{rf}(A)$	R-factor of matrix A	3.1
$\mathcal{S}_{diag}(n)$	space of diagonal $n \times n$-matrices	2.8
$\mathcal{S}_{low}(n)$	space of lower triangular $n \times n$-matrices	2.8
$\mathcal{S}_{lows}(n)$	space of strictly lower triangular $n \times n$-matrices	2.8
$\mathcal{S}_{skew}(n)$	space of skew symmetric $n \times n$-matrices	2.8
$\mathcal{S}_{symm}(n)$	space of symmetric $n \times n$-matrices	2.8
$\mathcal{S}_{upp}(n)$	space of upper triangular $n \times n$-matrices	2.8
$\mathcal{S}_{upps}(n)$	space of strictly upper triangular $n \times n$-matrices	2.8
$\mathsf{SL}(n)$	special linear group of $n \times n$-matrices	2.24
$\mathsf{SO}(n)$	special orthogonal group of $n \times n$-matrices	2.24
$\mathfrak{sl}(\mathrm{n})$	Lie algebra of $\mathsf{SL}(n)$	2.25
$\mathfrak{so}(\mathrm{n})$	Lie algebra of $\mathsf{SO}(n)$	2.25
s_R	similarity transformation with matrix R	2.9
$\mathrm{span}\{v_1, \ldots, v_n\}$	span of vectors $v_1, \ldots, v_n$	
$\mathrm{St}(m, n)$	Stiefel manifold of $n \times m$-matrices	2.21
$\mathrm{Stab}(a)$	Stabilizer of element a	25
$T_X\mathcal{M}$	tangent space to a manifold $\mathcal{M}$ at $X \in \mathcal{M}$	2.3
$\mathrm{Tr}(A)$	trace of matrix A	
$\mathsf{U}(n)$	group of unitary $n \times n$-matrices	3.28
$\mathrm{uf}(A)$	U-factor of matrix A	3.1
$\mathrm{upp}(A)$	upper triangular part of matrix A	2.9
$\mathrm{upph}(A)$	upper triangular half of matrix A	2.9,3.31
$\mathrm{upps}(A)$	strictly upper triangular part of matrix A	2.9
$\mathrm{vec}(A)$	Vector representation of matrix A	2.5
$\mathbb{Z}$	set of integers	
$\boldsymbol{\Delta}$	uncertainty structure of a structured singular value problem	11.1
$\delta_{i,j}$	Kronecker-δ, $\delta_{i,j} = [i = j]$	
$\mu_{\boldsymbol{\Delta}}$ or μ	structured singular value	11.1,11.2
$\rho(A)$	spectral radius of matrix A	11.2
$\pi(A)$	projection of matix A on $\boldsymbol{\Delta}$	11.5
$\rho_{\mathrm{skew}}(A)$	projection of matrix A on $\mathcal{S}_{skew}(n)$, parallel to $\mathcal{S}_{upp}(n)$	2.9
$\rho_{\mathrm{upp}}(A)$	projection of matrix A on $\mathcal{S}_{upp}(n)$, parallel to $\mathcal{S}_{skew}(n)$	2.9

Acronyms

EVD	EigenValue Decomposition
GK	Gentleman-Kung
LCMV	Linearly Constrained Minimum Variance
LMS	Least Mean Squares
MM	Moonen-McWhirter
RLS	Recursive Least Squares
SVD	Singular Value Decomposition

Contents

Chapter 1

Introduction

The unifying theme of this work is the theory of **continuous-time matrix algorithms**. A number of recent papers have considered matrix problems and algorithms from the viewpoint of *continuous-time systems theory and differential geometry*. We will refer to this viewpoint as the 'continuous-time approach'. Next to offering an elegant framework, shedding new light on known algorithms, and often unifying seemingly different results, this approach has also inspired new theoretical results and new discrete-time algorithms.

We give an *introduction* to 'continuous-time matrix algorithms', we present *new contributions* with applications in adaptive signal processing, robust control, and neural network associative memories, and we try to *evaluate* the merits and the limitations of the approach, on the basis of the presented material.

One of the contributions of this work is to establish a link between continuous-time algorithms and discrete-time **systolic algorithms**. We show how known systolic arrays have simple continuous-time counterparts, with neural network interpretations. Exploiting the natural parallelism of some continuous-time algorithms, we also derive new systolic algorithms.

An important motivation for this work, was the study of **adaptation laws for neural networks**. Since by making generalizations our work has gradually grown out of this field, most results are first presented in a more general context, and only afterwards reinterpreted in the context of neural networks.

Section 1.1, gives a general overview of this work. Section 1.2 gives a chapter by chapter overview.

1.1 General overview

Figure 1.1 summarizes the main ideas from a theoretical point of view, and considers four aspects of the central topic of continuous-time matrix algorithms.

The *first* aspect is the use of the continuous-time approach, for the design of both continuous-time and discrete-time algorithms. For all applications considered, we will *derive several continuous-time algorithms.* Basic tools are the use of gradients of functions defined over a manifold, and the use of derivatives of linear algebra related maps, like matrix decompositions, to design, transform, and analyze algorithms.

Although the *analog realization* of these algorithms is appealing and has received much recent interest for instance in the area of analog adaptive signal processing, it is only feasible for simple algorithms. We will discuss high-level aspects of the analog realization of recursive least-squares algorithms.

However, continuous-time algorithms can also be useful for the *design of discrete-time algorithms.* A new contribution of this work lies in the design of *systolic algorithms* through the continuous-time approach. Many continuous-time algorithms turn out to have simple uniform parallel signal flow graphs (consisting of simple, often identical cells). We show how some of these algorithms are continuous-time limits of systolic algorithms and derive new systolic algorithms, inheriting properties from their continuous-time counterparts. However there is no automatic method to turn continuous-time algorithms and their simple signal flow graphs into parallel discrete-time algorithms.

Next to the design of parallel algorithms, the continuous-time view is useful for the design of *discrete-time optimization algorithms*, especially with regular equality constraints. To define algorithms that 'live' in a space defined by an equality constraint, differential geometric concepts are useful, and the continuous-time view is never far away. Moreover continuous-time gradient algorithms can be helpful for exploratory experiments, as often they are much easier to program (using standard integration packages) than discrete-time algorithms that preserve the constraint.

The *second* aspect is the analysis of algorithms. Concepts from differential geometry and systems theory can be helpful in many ways for the analysis of algorithms, like the analysis of the stability equilibrium points and the corresponding convergence behavior. We concentrate on the analysis of adaptive signal processing algorithms. We prove results on *convergence* and *tracking performance* by using continuous-time tools such as singular value tracking and canonical angle tracking. Similar ideas are also applied to the analysis of neural network adaptation laws for memory applications.

As a *third* aspect, we claim that the continuous-time approach is an insightful

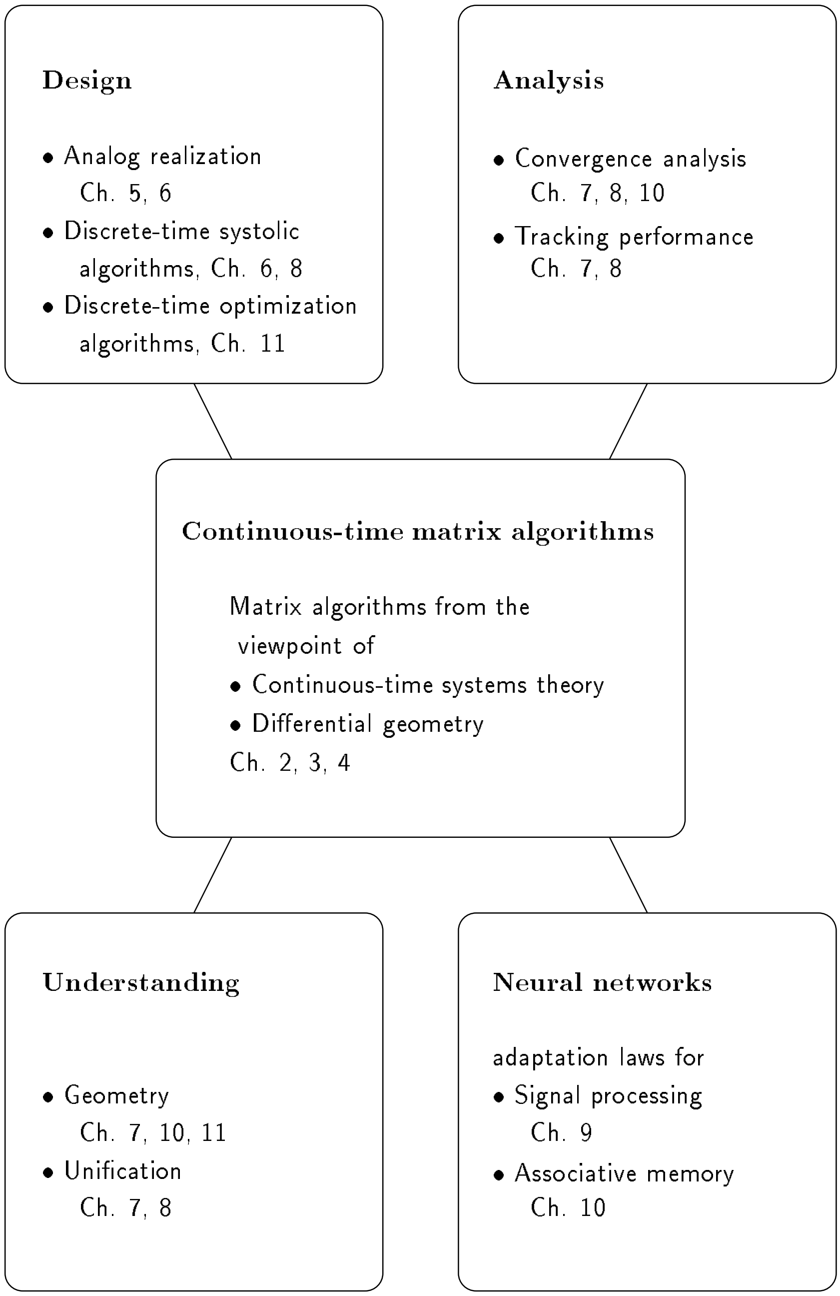

Figure 1.1: General overview of the main ideas of this work.

approach, leading to the *understanding* of algorithms and to inspiration in their design and analysis. We put a special emphasis on geometric interpretations. The appreciation of this more subjective aspect can of course depend on personal taste. Nevertheless it is clear, that continuous-time algorithms and their derivations often take surprisingly different and simpler forms than their discrete-time counterparts. And the mere fact that the viewpoint is different can only foster inspiration. The continuous-time viewpoint often offers a *unifying* viewpoint for different algorithms. We will see how seemingly different algorithms for subspace tracking take the same form in the continuous-time limit.

As a *fourth* aspect we consider neural network interpretations. As mentioned above, the study of adaptation laws for artificial neural networks has been a motivation since the beginning of our work, but as our research work has grown out of this field, we present some of these results as special cases and other interpretations.

The general idea borrowed from the field of neural networks is the following way of computing with a dynamical system. On a *local* scale (neurons, weights), the system shows simple and often uniform behavior, all elements evolving by the same law (or a small number of different laws) using information available through the network. On a *global* scale this results in some interesting behavior such as memory, control, or various types of information processing. One of the main problems is that the relation between even the simplest local behavior and the resulting global behavior can be very complex. In this work we study a class of systems, which can be realized with with simple and uniform local behavior on a network, and at the same time allow for a rigorous mathematical study of the global behavior. We consider applications in adaptive signal processing and neural associative memory.

Finally, figure 1.2 gives a general overview from the applications point of view. This corresponds more to the division into chapters, as discussed in the next section. The applications considered are diverse. It is impossible to go into details for all applications, and we will avoid deviating too far from the main topic of this work, being the application of ideas from continuous-time systems theory and differential geometry to the design and analysis of algorithms. To support the relevance of this viewpoint, in chapters 8, 10 and 11 we also pay attention to issues outside these domains.

1.2 Chapter by chapter overview

Figure 1.3 shows how the different chapters can be grouped together.

Chapters 2 to 4 are an introduction to continuous-time matrix algorithms. Although most of the results in this introduction are not new, we give some new

Recursive least-squares estimation
Ch. 6, 9
- Continuous-time algorithms
- Link with known systolic algorithms
- Algorithms for analog realization

Subspace tracking
Ch. 7, 8, 9
- Continuous-time algorithms
- New systolic algorithms
- Analysis

Associative memory
Ch. 10
- New continuous-time adaptation laws

Structured singular value (optimization)
Ch. 11
- New continuous-time and discrete-time algorithms

Figure 1.2: Overview from the applications point of view.

interpretations and put a special emphasis on the results of section 3.1. The organization of the text should allow readers who are only interested in concrete results of later chapters, to skip the introduction and only use results as they are needed.

Chapter 2 introduces the basic concepts. We give a more precise content to the notion of 'continuous-time algorithms'. We summarize some definitions and properties of matrix space as a Euclidean vector space, and we consider important manifolds in matrix space.

Chapter 3 introduces the basic tools for the design and analysis of matrix algorithms. We put the emphasis on a number of lemmas for tracking matrix decompositions, and give a geometric interpretation, considering matrix decompositions as nonlinear coordinate transformations of matrix space. Another important tool is the use of gradients of functions over Riemannian manifolds. We give definitions and examples and show the relevance of the decomposition

Ch. 1	Introduction	
Ch. 2	Definitions and concepts	Continuous-time matrix
Ch. 3	Tools	algorithms (introduction)
Ch. 4	Examples	
Ch. 5	Parallel realization	Parallel algorithms
Ch. 6	Recursive least-squares estimation	Adaptive signal processing
Ch. 7	Continuous-time subspace tracking	
Ch. 8	Discrete-time subspace tracking	
Ch. 9	Neural network interpretations	Neural networks
Ch. 10	Associative memory	
Ch. 11	Structured singular value	Optimization
Ch. 12	Conclusion	

Figure 1.3: Chapter by chapter overview

tracking lemmas for deriving gradients. In this chapter we also consider generalizations to the case of complex matrices.

Chapter 4 discusses the QR flow and the double bracket flow. These are two continuous-time algorithms that have played a central part in the evolution of the field, and are also relevant in further chapters. We also introduce a new square root version of the QR flow.

Chapter 5 is a short chapter, describing a class of continuous-time matrix algorithms, with a uniform parallel signal flow graph that can be easily derived from the formal description of the algorithm. The main continuous-time algorithms of chapters 6 to 10 fit in this class. These algorithms have neural network interpretations (chapter 9) and some of them are continuous-time limits of systolic algorithms (chapters 6 and 8).

Chapters 6 to 9 consider adaptive signal processing applications. The signal processing operations considered are classical ones, with many well known applications.

Chapter 6 considers continuous-time recursive least-squares estimation. Three algorithms are derived using the tools of chapter 2 for matrix decomposition tracking. The obtained algorithms fit in the class introduced in chapter 5, and can be interpreted as continuous-time versions of systolic algorithms. We also consider high-level aspects of analog realization. Neural network interpretations are discussed in chapter 9. The relations between these continuous-time algorithms and their systolic counterparts will be exploited to derive new systolic algorithms in chapter 8.

Chapter 7 considers the problem of subspace tracking. A class of continuous-time algorithms is considered, containing continuous-time versions of a number of known discrete-time algorithms. Three continuous-time algorithms fit in the class defined in chapter 5, and have a simple signal flow graph. Special attention is paid to one of them. A new analysis is given of the convergence and tracking behavior, using the tools of chapter 3 to track singular values and canonical angles between subspaces.

In chapter 8 one of the continuous-time subspace tracking algorithms of chapter 7 is converted into a discrete-time algorithm with a similar signal flow graph, that serves as a starting point for the derivation of high throughput systolic algorithms, inheriting the main properties of the continuous-time algorithms. The systolic realizations obtained are compared with existing related discrete-time algorithms.

Chapters 9 and 10 consider continuous-time algorithms as a framework for studying neural network adaptation laws.

Chapter 9 summarizes some elementary concepts of neural networks and discusses neural interpretations for the algorithms of chapters 5 to 7, and some related new ones.

In chapter 10 we consider continuous-time adaptation laws for recurrent neural networks, used as associative memories. We derive new adaptation laws which can be implemented on the network. They store patterns in the memory that are shown by a teaching signal. In an appendix, three closely related models of neural networks for associative memory applications are compared.

In **chapter 11** we derive continuous-time and discrete-time gradient based optimization algorithms for calculating lower-bounds of the structured singular value of a complex matrix. We derive a steepest ascent algorithm and a conjugate gradient algorithm, that live in a manifold of structured unitary matrix. Gradients are obtained using the theory of chapters 2 and 3. We compare our algorithm with an existing algorithm and illustrate our results by numerical experiments.

Finally, in **chapter 12** we conclude by summarizing our experience, trying to assess the merits and limitations of the continuous-time approach on the basis of the presented material. We also discuss open problems and further work and summarize the main contributions of this work.